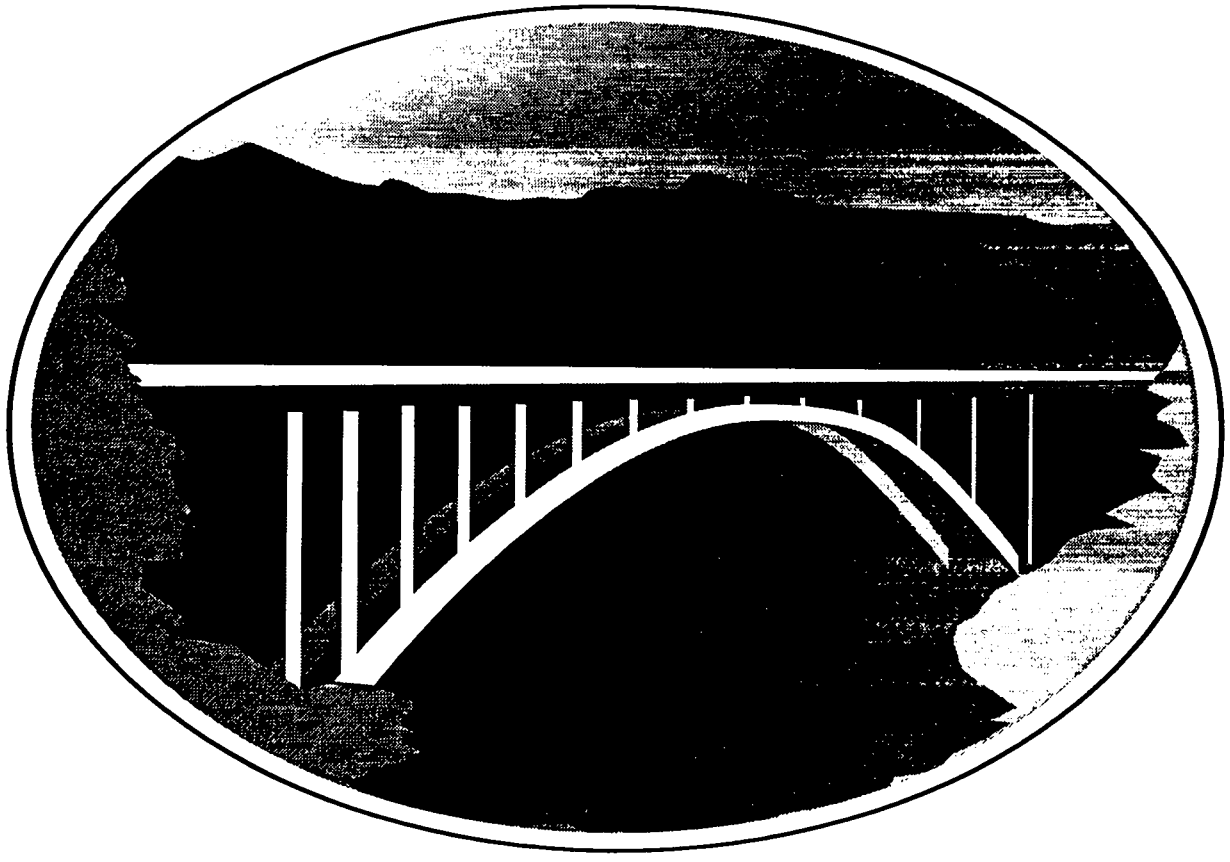
LA-12740-M
MANUAL

THE NJOY NUCLEAR DATA PROCESSING SYSTEM VERSION 91



Los Alamos
NATIONAL LABORATORY

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36.

Cover illustration by Kathi Parker, Group CIC-1

This work was supported by the US Department of Energy, Office of Energy Research.

An Affirmative Action/Equal Opportunity Employer

LA-12740-M
Manual

UC-413
Issued: October 1994

The NJOY Nuclear Data Processing System Version 91

R. E. MacFarlane
D. W. Muir

Los Alamos
NATIONAL LABORATORY
Los Alamos, New Mexico 87545

CONTENTS

The NJOY Nuclear Data Processing System, Version 91

by

R. E. MacFarlane and D. W. Muir

ABSTRACT

The NJOY nuclear data processing system is a comprehensive computer code package for producing pointwise and multigroup cross sections and related quantities from evaluated nuclear data in the ENDF format, including the latest US library, ENDF/B-VI. The NJOY code can work with neutrons, photons, and charged particles, and it can produce libraries for a wide variety of particle transport and reactor analysis codes.

I. INTRODUCTION

The NJOY nuclear data processing system[1, 2] is a comprehensive computer code package for producing pointwise and multigroup nuclear cross sections and related quantities from evaluated nuclear data in the ENDF format. The U.S. Evaluated Nuclear Data Files (ENDF) have progressed through a number of versions, notably ENDF/B-III, ENDF/B-IV, and ENDF/B-V.[3] The latest version, ENDF/B-VI,[4] has recently become available. The ENDF format is also being used in other modern libraries such as the JEF-I and JEF-II libraries produced in Europe. These libraries represent the underlying nuclear data from a physics viewpoint, but practical calculations usually require special libraries for particle transport codes or reactor core physics codes. This is the mission of NJOY—to take the basic data from the nuclear data library and convert it into the forms needed for applications.

A. The Modules of NJOY

The NJOY code consists of a set of modules, each performing a well-defined processing task. Each of these modules is essentially a separate computer program.

They are linked to one another by input and output files and a few common constants.

NJOY directs the flow of data through the other modules and contains a library of common functions and subroutines used by the other modules.

RECONR reconstructs pointwise (energy-dependent) cross sections from ENDF resonance parameters and interpolation schemes.

BROADR Doppler-broadens and thins pointwise cross sections.

UNRESR computes effective self-shielded pointwise cross sections in the unresolved energy range.

HEATR generates pointwise heat production cross sections (KERMA factors) and radiation-damage-production cross sections.

THERMR produces cross sections and energy-to-energy matrices for free or bound scatterers in the thermal energy range.

GROUPR generates self-shielded multigroup cross sections, group-to-group scattering matrices, photon production matrices, and charged-particle cross sections from pointwise input.

GAMINR calculates multigroup photoatomic cross sections, KERMA factors, and group-to-group photon scattering matrices.

ERRORR computes multigroup covariance matrices from ENDF uncertainties.

COVR reads the output of ERRORR and performs covariance plotting and output formatting operations.

MODER converts ENDF "tapes" back and forth between formatted (that is, ASCII, EBCDIC, etc.) and blocked binary modes.

DTFR formats multigroup data for transport codes that accept formats based on the DTF-IV code.

CCCCR formats multigroup data for the CCCC standard interface files ISOTXS, BRKOXS, and DLAYXS.

MATXSR formats multigroup data for the newer MATXS material cross-section interface file, which works with the TRANSX code to make libraries for many particle transport codes.

RESXSR prepares pointwise cross sections in a CCCC-like form for thermal flux calculators.

ACER prepares libraries in ACE format for the Los Alamos continuous-energy Monte Carlo code MCNP.

POWR prepares libraries for the EPRI-CELL and EPRI-CPM codes.

WIMSR prepares libraries for the thermal reactor assembly codes WIMS-D and WIMS-E.

PLOTR makes plots of cross sections and perspective views of distributions for both pointwise and multigroup data.

MIXR is used to combine cross sections into elements or other mixtures, mainly for plotting.

PURR is used to prepare unresolved-region probability tables for the MCNP continuous-energy Monte Carlo code.

The methods used in these modules and instructions on how to use them are given in subsequent chapters of this report. For convenience in making updates, each of these chapters is numbered separately and contains its own references. The chapters on the modules are followed by three appendices, which give information on the group structures, weight functions, and test problems.

B. Data Flow in NJOY

The modules of NJOY can be linked in a number of different ways to prepare libraries for various nuclear applications. The following brief summary illustrates the general flow of data in the code.

RECONR reads an ENDF tape* and produces a common energy grid for all reactions (the union grid) such that all cross sections can be obtained to within a specified tolerance by linear interpolation. Resonance cross sections are constructed using a method of choosing the energy grid that incorporates control over the number of significant figures generated and a resonance integral criterion to reduce the number of grid points generated for some materials.. Summation cross sections (for example, total, inelastic) are reconstructed from their parts. The resulting pointwise cross sections are written onto a "point-ENDF" (PENDF) tape for future use. BROADR reads a PENDF tape and Doppler-broadens the data using the accurate point-kernel method. The union grid allows all resonance reactions to be broadened simultaneously, resulting in a great savings of processing time. After broadening and thinning, the summation cross sections are again reconstructed from their parts. The results are written out on a new PENDF tape for future use. UNRESR produces effective self-shielded pointwise cross sections, versus energy and background cross section, in the unresolved range. This is done for each temperature produced by BROADR, using the average resonance parameters from the ENDF evaluation. The results are added to the PENDF tape using a special format.

If desired, additional special kinds of data can be added to the PENDF tape. HEATR computes energy-balance heating, KERMA, and damage energy using

*The word "tape" is synonymous with "file" in this manual, and it does not imply a physical tape. This usage is consistent with ENDF usage; the term "ENDF tape" is traditional.

reaction kinematics or applying conservation of energy. The ENDF photon production files can be used in this step, when available. Comparisons of momentum and energy calculations with the photon files can be used to find energy-balance errors in the evaluations. For ENDF-6 data, charged-particle distributions in File 6 are used directly to compute accurate heating and damage parameters. The energy-balance heating results are added to the PENDF tape using ENDF reaction numbers in the 300 series; kinematic KERMA uses the special identifier 443, and damage results use the special identifier 444. THERMR produces pointwise cross sections in the thermal range. Energy-to-energy incoherent inelastic scattering matrices can be computed for free-gas scattering or for bound scattering using a precomputed scattering law $S(\alpha,\beta)$ in ENDF format. The secondary angle and energy grids are determined adaptively so as to represent the function to a desired precision by linear interpolation; the angular representation is then converted to one based on equally probable angles. Coherent-elastic scattering from crystalline materials can be computed using internal lattice information, or for ENDF-6 format files, using data from the evaluation directly. The scheme used provides a detailed representation of the delta functions in energy and angle. Incoherent-elastic scattering for hydrogenous materials is represented using equally probable angles computed analytically or using ENDF-6 parameters. The results for all the processes are added to the PENDF tape using special formats and special reaction numbers between 221 and 250. GROUPR processes the pointwise cross sections produced by the modules described above into multigroup form. The weighting function for group averaging can be taken to be the Bondarenko form, or it can be computed from the slowing-down equation for a heavy absorber in a light moderator (see Appendices A and B for information on group structures and weight functions). Self-shielded cross sections, scattering matrices, photon production matrices, and charged-particle matrices are all averaged in a unified way, the only difference being in the function that describes the "feed" into a secondary group g' with Legendre order ℓ from initial energy E. The feed function for two-body scattering is computed using a center-of-mass (CM) Gaussian integration scheme, which provides high accuracy even for small Legendre components of the scattering matrix. Special features are included for delayed neutrons, coupled energy-angle distributions (either from THERMR or from ENDF-6 evaluations using File 6), discrete scattering angles arising from thermal coherent reactions, and charged-particle elastic scattering. Prompt fission is treated with a full group-to-group matrix. The results are written in a special "groupwise-ENDF" format (GENDF) for use by the output formatting modules. GAMINR uses a specialized version of GROUPR to compute photoatomic cross sections and group-to-group matrices. Coherent and incoherent atomic form factors are processed in order to extend the

useful range of the results to lower energies. Photon heat production cross sections are also generated. The results are saved on a GENDF tape.

The covariance module, ERRORR, can either produce its own multigroup cross sections using the methods of GROUPR or start from a precomputed set. The cross sections and ENDF covariance data are combined in a way that includes the effects of deriving one cross section from several others. Special features are included to process covariances for data given as resonance parameters or ratios (for example, fission $\bar{\nu}$). The COVR module uses the output from ERRORR and the widely available DISSPLA* plotting software to make publication-quality plots of covariance data; it also provides output in the efficient BOXER format, and it provides a site for user-supplied routines to prepare covariance libraries for various sensitivity systems.

MODER is usually used at the beginning of an NJOY job to convert ENDF library files into binary mode for calculational efficiency, or at the end of a job to obtain a printable version of a result from ENDF, PENDF, or GENDF input. It can also be used to extract desired materials from a multimaterial library, or to combine several materials into a new ENDF, PENDF, or GENDF file. DTFR is a simple reformatting code that produces cross-section tables acceptable to many discrete-ordinates transport codes. It also converts the GROUPR fission matrix to χ and $\bar{\nu}\sigma_f$ and prepares a photon-production matrix, if desired. The user can define edit cross sections that are any linear combination of the cross sections on the GENDF tape. This makes complex edits such as gas production possible. DTFR also contains its own plotting package for the cross sections, P_0 scattering matrix, and photon production matrix. This module has become somewhat obsolete with the advent of the MATXS/TRANSX system and PLOTR.

A number of other interface file formats are available from NJOY. The CCCCR module is a straightforward reformatting code that supports all the options of the CCCC-IV[5] file specification. In the cross-section file (ISOTXS), the user can choose either isotope χ matrices or isotope χ vectors collapsed using any specified flux. The BRKOXS file includes the normal self-shielding factors plus self-shielding factors for elastic removal. The DLAYXS provides delayed-neutron data for reactor kinetics codes. Note that some of the cross sections producible with NJOY are not defined in the CCCC files. For that reason, we have introduced the new CCCC-type material cross section file MATXS. The MATXSR module reformats GENDF data for neutrons, photons, and charged particles into the MATXS format, which is suitable for input to the TRANSX (transport cross

*The DISSPLA plotting package is a product of ISSCO, Integrated Software Systems Corporation, 10505 Sorrento Valley Rd., San Diego, CA 92121.

section) code.[6] TRANSX can produce libraries for a variety of particle transport codes, such as ANISN,[7] ONEDANT,[8] TWODANT,[9] and DIF3D.[10] The MATXS format uses efficient packing techniques and flexible naming conventions that allow it to store all NJOY data types. A companion module, RESXSR, formats pointwise data into a CCCC-like format for use in thermal flux calculators.

Pointwise data can also be fed directly into the ACER module. This module prepares cross sections and scattering laws in ACE format (A Compact ENDF) for the MCNP code.[11] All the cross sections are represented on a union grid for linear interpolation by taking advantage of the representation used in RECONR and BROADR. "Laws" for describing scattering and photon production are very detailed, providing a faithful representation of the ENDF-format evaluation with few approximations. The data are organized for random access for purposes of efficiency. A future version of MCNP will be able to handle self-shielding in the unresolved-energy range using probability tables. The PURR module in this release of NJOY can be used to prepare these tables.

Another alternate path for multigroup data is to use the POWR module to produce libraries for the power reactor codes EPRI-CELL or EPRI-CPM*. Similarly, the WIMSR module can be used to prepare libraries for the thermal reactor assembly codes WIMS-D and WIMS-E.[12]

At the end of any NJOY run, the PLOTR module can be used to view the results or the original ENDF data. PLOTR can prepare 2-D plots with the normal combinations of linear and log axes, including legend blocks or curve tags, titles, and so on. Several curves can be compared on one plot (for example, pointwise data can be compared with multigroup results), and experimental data points with error bars can be superimposed, if desired. PLOTR can also produce 3-D perspective plots of ENDF and GENDF angle or energy distributions. The DISSPLA plotting system is required. The MIXR module can be used to combine isotopes into elements for plotting and other purposes. It only works for simple cross sections at the present time.

C. Computer Implementation

NJOY is written in simple FORTRAN-77 with a few survivals from earlier times. As far as possible, features that are likely to be machine dependent (such as timers, date functions, and input/output) are isolated in special subroutines. Some Hollerith variables survive in places where they are appropriate, such as the

*EPRI-CELL and EPRI-CPM are proprietary products of the Electric Power Research Institute (EPRI), 3412 Hillview Avenue, Palo Alto, CA 94304.

CCCC modules. In other places, character variables are used.

For consistency and convenience, the NJOY modules make use of a set of common functions and subroutines located in the NJOY module. They include customized free-format input, special file open, close, and reposition routines, dynamic storage allocation calls, utility programs for ENDF-format files, fatal error and warning message calls, and uniform time and date routines. See the NJOY chapter of this report for more details.

NJOY is usually loaded with each module being a separate "segment" or "overlay." The NJOY module is then the main segment or overlay, and it controls the execution of the other modules. This is possible because each module is basically an independent program, both logically and physically. Changes or repairs in one module rarely have side effects in other modules. The segmented structure thereby leads to economy in memory requirements.

NJOY is heavily commented. Each module starts with a long block of comment cards that gives a brief description of the module and then gives the current user input instructions. Users should always check the input instructions in the current version of the NJOY source code rather than the instructions summarized in this manual—changes may have been made. Furthermore, each function or subroutine starts with a block of comment cards that describes its function and special requirements. Additional lines of comment cards are used inside each procedure to block off its major components.

Typography conventions for FORTRAN differ from place to place. On most machines at Los Alamos and at most other Cray installations, FORTRAN text is given in lowercase. On the other hand, many IBM and VAX installations customarily use uppercase. In order to avoid translation problems, NJOY avoids using mixed-case text for comments or for labels on graphs. (Note that the plotting modules use shift characters to change between uppercase and lowercase.) In this report, FORTRAN text and variable names are printed using an uppercase typewriter font.

D. Version Control

A strict procedure for version control has been adopted for NJOY. In addition to providing a simple method for making changes and repairs to the code, it provides the documentation required for formal Quality Assurance (QA) programs. The central feature of this procedure is a new code called UPD.[13] UPD is similar in concept to the UPDATE code[14] found on CDC and Cray systems, or to the UPEML code[15] included with NJOY 89 as released by the code centers, except that it has been simplified in concept and written in basic FORTRAN in order to be adaptable to as many systems as possible.

Like UPDATE, the UPD code uses a "source" file (called SRC) that contains the FORTRAN cards of the code separated into groups by *DECK cards. However, there is no binary program library. Instead, UPD works directly from the source file. Also like UPDATE, corrections are specified by giving "idents" (in the file UPN) containing commands like "*I *addr*" or "*D *addr*" (insert and delete, respectively) followed by the new or changed text. The addresses (*addr*) for corrections have forms like "NJOY.6" or "RECONR.111,222." Updates can be made on top of earlier updates; for example, "*D UP32.45,MATXSR.1234."

The revision procedure requires that updates be made in sequence: UP1, UP2, UP3, ... The corresponding NJOY versions would be called 91.1, 91.2, 91.3,.... Each ident can only contain changes for one module, and it must start with a comment card giving the date and the reason for the change. For example,

```
*IDENT UP22
*/ GROUPR -- 21 OCT 91 -- FIX BUG IN CHARGED-PARTICLE ELASTIC
*D GROUPR.1126,1127
 ...
```

Once finished, tested, dated, and released, a given ident cannot be changed directly. It can only be changed by using a subsequent new ident in the UPN file. This assures that an NJOY version number such as 91.22 uniquely defines every line in the code.

The next-to-the-last ident in the update directives file fixes the version information in the NJOY source and changes the version string printed on the "banner" at the top of every NJOY run. For example,

```
*IDENT VERS
*/ VERSION 91.22 -- 21 OCT 91
*D NJOY.10
C     *       VERSION 91.22 -- 21 OCT 91
*D NJOY.301
      DATA VERS/'91.22'/
```

Therefore, any given copy of the NJOY source code is uniquely labeled, and every NJOY run is uniquely traceable to a specific version of the source code.

The version ident would normally be followed by machine-dependent changes for a given laboratory and computer system. For example, the first few lines of the machine-dependent update for the CTSS operating system used on Cray computers at Los Alamos follow:

```
*IDENT CTSS
*D NJOY.3
   PROGRAM NJOY(INPUT,OUTPUT,TTY,TAPE5=INPUT,TAPE6=OUTPUT,TAPE7=TTY)
*I NJOY.122
C     ***ENABLE FILE REPLACEMENT ON EXECUTE LINE
      CALL FILEREP
*I NJOY.308
      CALL MACH(MX)
*D NJOY.357
C     REQUEST TRACEBACK AND DROP FILE
      CALL TRCBK(NSYSO,0)
      CALL TRCBK(NSYSE,0)
      CALL EXITA(1)
*D NJOY.725
      CALL SECOND(TIME)
   ...
```

Note that these changes affect the system I/O units, fatal error handling, and elapsed time function. Several machine-dependent ident packages are included in the NJOY distribution.

UPD also provides a limited conditional capability for defining machine dependencies through the *IF, *ELSEIF, *ELSE, and *ENDIF directives. The conditions for the IF and ELSEIF commands are set using the "*SET name" directive in the UPD input file. In NJOY 91, the main use for this feature is "SW", which turns on the features for "short-word" machines (that is, REAL*8, DLOG, MULT=2, etc.). A list of all the conditional options that can be set for NJOY 91.0 is given below. Check the comment cards at the start of your current version of NJOY to see if others have been introduced.

Set	Function
SW	short-word machine (32-bit words)
DISS	activate DISSPLA plotting
WORDIO	use word-indexed binary IO in ACER

UPD produces either a "CPL" file for input to a compiler, or a "LST" file that provides a convenient listing for the code, complete with UPD identifiers for each line. See the UPD report[13] for a more complete description.

E. Test Problems

Another important part of the NJOY revision control procedure is the set of standard test problems used to validate each new version. This kind of systematic testing is also a key part of any QA program. The NJOY test problems also act as examples in helping new users to operate the code system. Brief descriptions of the current set of test problems follow. See Appendix C for details.

Problem 1: Process one ENDF/B-V isotope through pointwise and multigroup modules. It tests heating and damage calculations, thermal calculations for free-gas carbon and carbon bound in graphite, and multigroup averaging. The full PENDF tape is included in the test comparisons. ENDF/B-V Tape 511* and ENDF/B-III thermal tape T322 are required.

Problem 2: Process one ENDF/B-IV isotope for a practical CCCC library. It tests resonance reconstruction, Doppler-broadening to several temperatures, self-shielded unresolved cross sections, self-shielded multigroup cross sections, and the CCCC files ISOTXS, BRKOXS, and DLAYXS. Tape 404 is required.

Problem 3: Process photon interaction cross sections into both DTF and MATXS format. The problem tests photoatomic cross section linearization in RECONR, multigroup averaging in GAMINR, and output formatting in DTFR and MATXSR. The DLC7E library† is required.

Problem 4: ERROR is tested, including the calculation of covariances for fission $\bar{\nu}$. Tape 511 is used.

Problem 5: This run tests COVR, including the plotting capability. Tape 511 is used. This is a fairly expensive calculation, and when DISSPLA plotting is enabled, it produces a large number of covariance graphs.

Problem 6: Includes a number of 2-D sample problems for PLOTR, and one 3-D case. Plots with special characters, error bars, curve tags, and legend blocks are demonstrated.

Problem 7: Prepares an ACE-format library for a fissionable material.

An NJOY "release" is normally made after a number of idents have been inserted into UPN. For example, released versions might be 91.0, 91.13, 91.38, etc. When the number of idents becomes cumbersome, a new resequenced version is normally prepared; for example, 91.99 might be followed by 93.0. Intermediate versions would be experimental. Each ident is normally tested as it is written to make sure that it performs correctly. The entire suite of test problems is run when each new "release" is made, and the results are saved to provide a benchmark

*ENDF tapes are available from the National Nuclear Data Center (NNDC) at the Brookhaven National Laboratory, Upton, NY 11973.

†The DLC7E photon library is available from the Radiation Shielding Information Center (RSIC) at the Oak Ridge National Laboratory, Box 2008, Oak Ridge, TN 37831.

for testing future releases. Each difference between the test results for subsequent versions should be understood.

F. History and Acknowledgments

NJOY was started as a successor to MINX[16] late in 1973 (it was called MINX-II then). The current name was chosen in late 1974 to be evocative of "MINX plus". The first goals were to add a photon production capability like that in LAPHAN0,[17] to add a photon interaction capability like GAMLEG,[18] to provide an easy link to the Los Alamos 30-group libraries of the day using DTF[19] format, and to merge in the capabilities of ETOPL[20] to produce libraries for the MCNP Monte Carlo code. Most of the work was done by MacFarlane; Rosemary Boicourt joined the project in 1975. First, the RESEND[21] and SIGMA1[22] modules of MINX were converted to use union grids, and a new method of resonance reconstruction was developed. These steps led to RECONR and BROADR. UNRESR, which was based on methods from ETOX,[23] was moved over from MINX with only a few changes. Next, a completely new multigroup averaging program, GROUPR, was developed around the unifying concept of the "feed function," which handled neutron- and photon-production cross sections in a parallel manner. The CM Gaussian integration for discrete two-body scattering was developed. DTFR was developed as the first NJOY output module. The first versions of the NJOY utility codes were introduced; the new concepts of "structured programming" inspired some of the features of the new NJOY code.

Major influences during this period included Don Harris, Raphe LaBauve, Bob Seamon, and Pat Soran at Los Alamos, and Chuck Weisbin at the Oak Ridge National Laboratory (ORNL). Odelli Ozer at Brookhaven (and later EPRI) helped with RESEND, and Red Cullen at Livermore and John Hancock at Los Alamos helped with the Doppler broadening module. In those days, the development of NJOY was supported by the US Fast Breeder Reactor and Weapons Programs.

Code development continued during 1975. The ERRORR module was added for calculating covariances from ENDF/B files. The ACER module was created by borrowing heavily from ETOPL and Chuck Forrest's MCPOINT code. Rich Barrett joined the project, and he did most of the work in creating a new CCCC module for NJOY that had several advances over the MINX version and met the CCCC-III standards.[24] By the end of the year, HEATR had also been added to the code (with ideas from Doug Muir). HEATR gave NJOY most of the capabilities of the original KERMA factor code, MACK.[25]

During 1976, free-format input and dynamic data storage were added to NJOY, GAMINR was written to complete the original NJOY goal of processing photon interaction cross sections, and the MATXSR module was designed and written, primarily by Rich Barrett. This completed the capability to construct fully coupled cross sections for neutron-photon-heating problems. A major new effort was writing the THERMR module to improve upon the thermal moderator scattering cross sections then produced using the FLANGE-II[26] and HEXSCAT[27] codes, and starting the POWR module to produce cross sections for the EPRI-CELL and EPRI-CPM codes used by the US electric utility companies. This work was funded by the Electric Power Research Institute (EPRI).

The first release of NJOY to the Radiation Shielding Information Center (RSIC) at Oak Ridge and to the National Energy Software Center (NESC) at Argonne was made in the summer of 1977. This version was tested and converted for IBM machines by R. Q. Wright (ORNL). Also, TRANSX was developed during 1977, the MATXS1 30 x 12 library was produced based on ENDF/B-IV, the flux calculator was added to NJOY to support the EPRI library work, and the first version of the EPRI-CELL library was made and used.

A second release of NJOY was made in 1978.[28] In addition, further improvements were made for preparing EPRI cross sections, the MATXS/TRANSX system was improved, and a thermal capability was added to the MCNP Monte Carlo code using cross sections from THERMR as processed by ACER.

In 1979, the radiation damage calculation was added to HEATR, and the GROUPR flux calculator was further improved. In 1980, a plotting option was added to ERRORR. During this period, NJOY had become more stable. Changes usually consisted of small improvements or bug fixes instead of major new capabilities. Starting in this period, NJOY received some support from the US Magnetic Fusion Energy Program, mostly for covariance work and TRANSX related library support.

In 1981 and 1982, improvements included the momentum-conservation method for radiative capture in HEATR. Analytic $\psi\chi$ broadening was added to RECONR for some cases, and the integral criteria for resonance reconstruction with significant-figure control were installed in RECONR. Several new capabilities were added to ERRORR, and the COVR module was added to NJOY to handle both ERRORR plotting and covariance library output (this covariance-related work was done by Doug Muir). In addition, CCCCR was updated to the CCCC-IV[5] standards. A major new release called NJOY (10/81) was made to the code centers, and the first two volumes[2] of a new NJOY report were written and published.

European users began to make important contributions about this time. Enrico Sartori of the NEA Data Bank at Saclay in France, Margarete Mattes of the University of Stuttgart in Germany, and Sandro Pelloni of the Paul Scherer Institute in Switzerland deserve mention.

Another major release, NJOY 6/83, was made in 1983. By this time, NJOY was in use in at least 20 laboratories in the US and around the world. Small improvements continued, such as the kinematic KERMA calculation in HEATR. Volume IV of the NJOY report was published in 1985, and Volume III appeared in 1987.

The next big set of improvements in NJOY was associated with the introduction of the ENDF-6 format. This required significant changes in RECONR to support new resonance formats like Reich-Moore and Hybrid R-Function (implemented with help from Charlie Dunford of Brookhaven), in HEATR to implement direct calculations of KERMA and damage from charged-particle and recoil distributions in File 6, in THERMR to support new formats for File 7, and in GROUPR to support the group-to-group transfer matrices using energy-angle data from File 6. The PLOTR module was also developed during this period. The result was the release of NJOY 89[29] in time for processing the new ENDF/B-VI library and the JEF-II library (which was also in ENDF-6 format).

During 1989 and 1990, initial processing of ENDF/B-VI and JEF-II exposed a number of small problems that had to be fixed. In addition, the ACER module was rewritten to clean it up, to add capabilities to produce ACE dosimetry and photoatomic libraries, and to provide for convenient generation of files in several different formats for users away from Los Alamos. A MIXR module was added to NJOY, mostly to allow elemental cross sections to be reconstructed from ENDF/B-VI isotopes for plotting purposes. A new technique was introduced into GROUPR and all the output modules for multigroup data that provided for more efficient processing of fission and photon production matrices with lots of low-energy groups. Major revisions were made to the MATXS format to allow for charged-particle cross sections, to pack matrices with lots of low-energy groups more efficiently, and to make inserting and extracting new materials easier. The WIMSR module, which had been under development for a number of years in cooperation with WIMS users in Canada and Mexico, was introduced into NJOY. Finally, the PURR module for generating unresolved-region probability tables for use with MCNP, which had also been under development for many years, was formally added to the code.

The result of this year-and-a-half of work is NJOY 91, as described in this

manual.

G. References

1. R. E. MacFarlane and R. M. Boicourt, "NJOY: A Neutron and Photon Processing System," *Trans. Am. Nucl. Soc.* 22, 720 (1975).

2. R. E. MacFarlane, D. W. Muir, and R. M. Boicourt, "The NJOY Nuclear Data Processing System, Volume 1: User's Manual," Los Alamos National Laboratory report LA-9303-M, Vol. I (ENDF-324), (May 1982). R. E. MacFarlane, D. W. Muir, and R. M. Boicourt, "The NJOY Nuclear Data Processing System, Volume II: The NJOY, RECONR, BROADR, HEATR, and THERMR Modules," Los Alamos National Laboratory report LA-9303-M, Vol. II (ENDF-324) (May 1982). R. E. MacFarlane and D. W. Muir, "The NJOY Nuclear Data Processing System, Volume III: The GROUPR, GAMINR, and MODER Modules," Los Alamos National Laboratory report LA-9303-M, Vol. III (ENDF-324) (October 1987). D. W. Muir and R. E. MacFarlane, "The NJOY Nuclear Data Processing System, Volume IV: The ERRORR and COVR Modules," Los Alamos National Laboratory report LA-9303-M, Vol. IV (ENDF-324) (December 1985).

3. R. Kinsey, Ed., "ENDF-102, Data Formats and Procedures for the Evaluated Nuclear Data File, ENDF," Brookhaven National Laboratory report BNL-NCS-50496 (ENDF-102) 2nd Edition (ENDF/B-V) (October 1979).

4. P. F. Rose and C. L. Dunford, Eds., "ENDF-102, Data Formats and Procedures for the Evaluated Nuclear Data File, ENDF-6," Brookhaven National Laboratory report BNL-NCS-44945 (July 1990).

5. R. D. O'Dell, "Standard Interface Files and Procedures for Reactor Physics Codes, Version IV," Los Alamos Scientific Laboratory report LA-6941-MS (September 1977).

6. R. E. MacFarlane, "TRANSX-CTR: A Code for Interfacing MATXS Cross-Section Libraries to Nuclear Transport Codes for Fusion Systems Analysis," Los Alamos National Laboratory report LA-9863-MS (February 1984).

7. W. W. Engle, Jr., "A Users Manual for ANISN, A One-Dimensional Discrete Ordinates Transport Code with Anisotropic Scattering," Oak Ridge Gaseous Diffusion Plant Computing Technology Center report K-1693 (1967).

8. R. Douglas O'Dell, Forrest W. Brinkley, Jr., Duane R. Marr, and Raymond E. Alcouffe, "Revised User's Manual for ONEDANT: A Code Package for One-Dimensional, Diffusion-Accelerated, Neutral-Particle Transport," Los Alamos National Laboratory report LA-9184-M, Rev. (December 1989).

9. R. E. Alcouffe, F. W. Brinkley, D. R. Marr, and R. D. O'Dell, "User's Guide for TWODANT: A Code Package for Two-Dimensional, Diffusion-Accelerated, Neutral-Particle Transport," Los Alamos National Laboratory report LA-10049 Revised (February 1990).

10. K. L. Derstine, "DIF3D: A Code to Solve One-, Two-, and Three-Dimensional Finite-Difference Diffusion Theory Problems," Argonne National Laboratory report ANL-82-64 (April 1984).

11. Judith F. Briesmeister, Ed., "MCNP—A General Monte Carlo Code for Neutron and Photon Transport, Version 3A," Los Alamos National Laboratory report LA-7396-M, Rev. 2 (September 1986).

12. J. R. Askew, F. J. Fayers, and P. B. Kemshell, "A General Description of the Lattice Code WIMS," *J. Brit. Nucl. Energy Soc.* 5, 564 (1966).

13. R. E. MacFarlane and D. C. George, "UPD: A Portable Version-Control Program," Los Alamos National Laboratory report LA-12057-MS (April 1991).

14. "UPDATE, VERSION 1, Reference Manual," Revision 11/23/81, Control Data Corporation report 60449900 (November 1981).

15. T. A. Mehlhorn and M. F. Young, "UPEML Version 2.0: A Machine-Portable CDC Update Emulator," Sandia National Laboratories (Albuquerque) report SAND87-0679 (August 1987).

16. C. R. Weisbin, P. D. Soran, R. E. MacFarlane, D. R. Harris, R. J. LaBauve, J. S. Hendricks, J. E. White, and R. B. Kidman, "MINX, A Multigroup Interpretation of Nuclear X-Sections from ENDF/B," Los Alamos Scientific Laboratory report LA-6486-MS (ENDF-237) (September 1976).

17. D. J. Dudziak, R. E. Seamon, and D. V. Susco, "LAPHAN0: A Multigroup Photon-Production Matrix and Source Code for ENDF," Los Alamos Scientific Laboratory report LA-4750-MS (ENDF-156) (January 1972).

18. K. D. Lathrop, "GAMLEG—A FORTRAN Code to Produce Multigroup Cross Sections for Photon Transport Calculations," Los Alamos Scientific Laboratory report LA-3267 (April 1965).

19. K. D. Lathrop, "DTF-IV, A FORTRAN Program for Solving the Multigroup Transport Equation with Anisotropic Scattering," Los Alamos Scientific Laboratory report LA-3373 (November 1965).

20. R. J. LaBauve, C. R. Weisbin, R. E. Seamon, M. E. Battat, D. R. Harris, P. G. Young, and M. M. Klein, "PENDF: A Library of Nuclear Data for Monte Carlo Calculations Derived from Data in the ENDF/B Format," Los Alamos Scientific Laboratory report LA-5687 (October 1974).

21. O. Ozer, Ed. "Description of the ENDF/B Processing Codes and Retrieval Subroutines," Brookhaven National Laboratory report BNL-50300 (ENDF 11) (June 1971).

22. D. E. Cullen, "Program SIGMA1 (Version 74-1)," Lawrence Livermore Laboratory report UCID-16426 (January 1974).

23. R. E. Schenter, J. L. Baker, and R. B. Kidman, "ETOX, A Code to Calculate Group Constants for Nuclear Reactor Calculations," Battelle Northwest Laboratory report BNWL-1002 (1969).

24. B. M. Carmichael, "Standard Interface Files and Procedures for Reactor Physics Codes, Version III," Los Alamos Scientific Laboratory report LA-5486-MS (1974).

25. M. A. Abdou, C. W. Maynard, and R. Q. Wright, "MACK: A Computer Program to Calculate Neutron Energy Release Parameters (Fluence to Kerma Factors) and Multigroup Neutron Reaction Cross Sections from Nuclear Data

in ENDF Format," Oak Ridge National Laboratory report ORNL-TM-3994 (1973).

26. H. C. Honek and D. R. Finch, "FLANGE-II (Version 71-1), A Code to Process Thermal Neutron Data for an ENDF/B Tape," E. I. DuPont de Nemours and Co. Savannah River Laboratory report DP-1278 (1971).

27. Y. D. Naliboff and J. U. Koppel, "HEXSCAT, A Coherent Elastic Scattering of Neutrons by Hexagonal Lattices," General Atomic report GA-6026 (1964).

28. R. E. MacFarlane, R. J. Barrett, D. W. Muir, and R. M. Boicourt, "NJOY: A Comprehensive ENDF/B Processing System," in A Review of Multigroup Nuclear Cross-Section Processing, Proceedings of a Seminar-Workshop, Oak Ridge, Tennessee, March 14-16, 1978," Oak Ridge National Laboratory report ORNL/RSIC-41 (October 1978).

29. R. E. MacFarlane, "Introducing NJOY 89," in Proceedings of the Seminar on NJOY and Themis, 20-21 June 1989, OECD/NEA Data Bank, Saclay, France (1989). See also "Nuclear Theory and Application Progress Report, January 1, 1989—April 1, 1990," pp. 21-38, Los Alamos National Laboratory report LA-11972-PR (December 1990).

II. NJOY

The NJOY module of the NJOY system has two elements. First, it contains the main program for the system, which directs the data flow through the other modules. Second, it contains a library of commonly used subroutines and functions for use in the other modules. This manual describes NJOY version 91.91.

A. The NJOY Program

This is the main program of the NJOY system. It starts by reading an input option parameter and the ENDF version value. Then it sets up a loop that simply reads a module name in free format and calls in the requested module. The first card read by any module contains the unit numbers for the various input and output files. In this way, the output of one module can be assigned to be the input of another module, thereby linking the modules to perform the desired processing task. An example of the linking procedure is given below:

```
  [mount an ENDF tape as TAPE20]
0
5
RECONR
20 21
  [input lines for RECONR]
GROUPR
20 21 0 22
  [input lines for GROUPR]
DTFR
22 23 21
  [input lines for DTFR]
STOP
  [DTF-format lines written on TAPE23]    .
```

An important feature of a good modular system is that there be a minimum of interactions between the modules in order to reduce side effects. In the NJOY system, modules communicate only by the input and output units specified, as shown above, and through two common blocks of constants:

```
COMMON/MAINIO/NSYSI,NSYSO,NSYSE,NTTY
COMMON/UTIL/NPAGE,IVERF    .
```

The main program sets the page length (NPAGE) for blocked binary files and assigns the unit numbers for system input and output. NJOY expects these numbers to be less than 10 (the normal choice is 5 for input and 6 for output). In a time-sharing environment, it is often helpful to have a short print for the terminal

while still preserving the long listing for the system printer. Such an option is provided by IOPT=1. This option changes the input and output to the terminal (see NTTY). Fatal error messages and warning messages will appear on the output file; they may also appear on the terminal or a special error unit (see NSYSE). The final common parameter is IVERF, which should be between 3 and 6 to process ENDF/B-III, -IV, -V, or -VI evaluations. The input instructions for the NJOY module are given as comment cards at the beginning of the module. They are reproduced here for the convenience of the user.

```
*                                                                *
*---INPUT SPECIFICATIONS (FREE FORMAT)----------------------------*
*                                                                *
* CARD1       INPUT OPTION                                       *
*    IOPT     0 FOR CARD INPUT AND FULL OUTPUT.                  *
*             1 FOR TERMINAL INPUT WITH SHORT OUTPUT ON TERMINAL *
* CARD2                                                          *
*    IVERF    ENDF/B VERSION NUMBER (3 THRU 6 ONLY)              *
* CARD 3                                                         *
*    MOPT     SIX CHARACTER MODULE NAME, E.G. RECONR.  THE       *
*             STRING MAY ALSO BE DELIMITED BY *, E.G., *RECONR*. *
*             REPEAT MOPT FOR EACH MODULE DESIRED, AND USE       *
*             STOP OR *STOP* TO TERMINATE PROGRAM.               *
*                                                                *
* SEE THE COMMENTS AT THE START OF EACH MODULE FOR               *
* ITS SPECIFIC INPUT INSTRUCTIONS.                               *
*                                                                *
```

The method used for invoking NJOY is system dependent, and it also depends on whether the user wants batch or interactive mode. For example, under the CTSS system used on the Cray computers at Los Alamos, batch jobs can be started with the command

```
NJOY
```

where the default files are INPUT and OUTPUT, or

```
NJOY INPUT=TEST1,OUTPUT=OUT1
```

with explicit input and output file names. For interactive jobs, the appropriate command is

```
NJOY INPUT=TTY    .
```

The code prompts the user with "?" on the terminal. The correct response is "1" for IOPT. NJOY will then begin prompting the user for input with messages on the terminal.

For UNIX systems, the standard I/O redirection syntax is used to control the system files:

```
NJOY <TEST1 >OUT1    .
```

The default input and output units are the terminal, so interactive runs can be started with a simple "NJOY". As above, this would cause NJOY to prompt for IOPT with a question mark.

B. Interface Files

Another requirement of a good modular system is that the input and output files be in a common format so that modules can work with each other's output in a flexible way. Since NJOY is basically an ENDF processing code, ENDF-compatible formats were chosen for linking modules together. "Input" and "output" modules can be specified to communicate with other formats (the "outside world"). In the example above, TAPE23 is an example of such an external file. The other tapes* in the example are ENDF-type files, and the sequence shown is fairly typical. If the user in the example needs data at a higher temperature, the RECONR point-ENDF, or PENDF, tape (TAPE21) can be run through BROADR to produce a new Doppler-broadened PENDF tape for GROUPR. Many other combinations are possible simply by rearranging the sequence of module names and changing the unit numbers that link them. These common-format files also provide for convenient restarts at many points in the calculational sequence. For example, if a user is trying to produce pointwise cross sections at 300 K, 600 K, and 900 K and runs out of time while working on 900 K, he can save the partially completed PENDF tape and restart from 600 K. Multigroup modules use specially constructed groupwise-ENDF formats (GENDF) that are compatible with the multigroup output modules. A GENDF tape from GROUPR can be saved in the NJOY data library, run through CCCCR to produce one output format, and then run through MATXSR for another output format.

In NJOY, unit numbers from 20 through 99 are used for storing results or linking modules, units 10 through 19 are reserved for scratch files, which will be destroyed after a module has completed its job, and units below 10 are reserved for the system. Negative unit numbers indicate binary mode.

There are special utility routines to open, close, and reposition files. These

*The word "tape" will be used throughout this report as a synonym for "file"; an actual physical tape is not implied. This is consistent with ENDF custom, where the phrase "ENDF tape" is traditional. Furthermore, in most NJOY installations, the actual files on the machine will have names like "TAPE23."

routines automatically handle the NJOY conventions on positive or negative unit numbers, scratch files, and so on. They are written using standard FORTRAN-77 calls.

OPENZ(LUN,NEW)

Open the unit=ABS(LUN). If LUN≥0, use coded (formatted) mode, and if LUN≤0, use binary mode. Destroy on close or job termination if 10≤LUN≤20. If NEW=1, destroy the file on this unit (if it exists) and open a new file.

CLOSZ(LUN)

Close the file with unit=ABS(LUN). Do nothing if LUN=0 or if LUN is a scratch file.

REPOZ(NTAPE)

Rewind the file with unit=ABS(NTAPE). Do nothing if NTAPE=0.

SKIPRZ(LUN,NREC)

Skip NREC records forward or backwards. Caution: Some systems have a call for this option; others can use loops of backspace and dummy reads as given in the NJOY code. Both these operations work well for systems that use "linked-list" data structures for I/O files. On some systems, however, backspace is implemented as a rewind followed by forward dummy reads to the desired location. In such cases (for example, VAX), SKIPRZ must be recoded to avoid calling backspace repeatedly.

C. Free Format Input

For a card-input program, free-form input is convenient, but in a time-sharing environment, it is almost essential. Therefore, a subroutine FREE has been included among the NJOY utilities to provide a standardized free-format input capability.

FREE(NIN,Z,NZA,NCW)

Read free-form input from NIN into the array Z. On input, NZA is the number of words desired; on return, it is the number of words found. NCW is the number of Hollerith characters for each word (blank fill to right).

All numbers read from the input cards are returned as real in array Z. The calling program can convert selected numbers to integer mode as required using the standard FORTRAN nearest-integer function, NINT. Hollerith variables are returned in integer form using the internal N-bit code of the machine. If NCW is larger than the number of characters per word, successive locations of Z will be used.

Fields on the input cards are delimited by any character not used for another purpose (+, −, number, E, H, *, R , /). For exponent fields, the E must be

present, and spaces are not allowed before the E. Decimal points are not required after numbers. Text or Hollerith fields may use nHstring or *string*, or the field may be a single word (NCW characters of less) starting with a letter. The character / terminates the input for one call to FREE (it may involve more than one card) leaving any unread variables unchanged. This feature is often used to default variables from the right. The nR specification causes the number following R to be repeated n times. Comments can be given after the /, and lines with a $ in column 1 are also taken to be comments. Some input examples follow.

legal	illegal
12 12. 1.2E1	1.2+1 E1
U235 *U235* 4HU235	4RU235 (does not right-justify)
5R1.0 3R1.1E6	

Other examples will be found in input samples throughout this report.

NJOY modules normally use FREE in a way that takes advantage of its capability to default fields from the right. For example,

```
DIMENSION Z(10)
...
NZ=4
Z(2)=0.
Z(3)=5
Z(4)=1
CALL FREE(NIN,Z,NZ,4)
AA=Z(1)
BB=Z(2)
II=NINT(Z(3))
JJ=NINT(Z(4))
...
```

No default is provided for AA; it must be given. If the input card is terminated after the first entry with /, the remaining values will all be left with their default values, and NZ will be returned as 1. On the other hand, the input card

```
1.E5 22. 4/
```

will assign the value 22.0 to BB, the value 4 to II and leave JJ set to its default value of 1. NZ will be returned as 3.

FREE contains one parameter that may have to be changed when converting between different machines: MACHWD is the number of Hollerith characters in a machine word (10 on CDC, 8 on Cray, 4 on IBM). The rest of the character

handling is incorporated into **SUBROUTINE PACKC**, which inserts characters into words using FORTRAN-77 character operations, and **FUNCTION LETTER**, which tests whether a given character is a letter to find undelimited Hollerith strings.

D. ENDF Input-Output

The ENDF format for evaluated nuclear data is well documented elsewhere,[1, 2] but for the convenience of the reader, some features of the format will be described here.

ENDF tapes are subdivided internally into "materials" (MAT), "files" (MF), and "sections" (MT). A MAT contains all data for a particular evaluation for an element or isotope (for example, MAT1276 is an evaluation for 8-O-16 in ENDF/B-IV). A file contains a particular type of data for that MAT: MF=3 is a cross-section versus energy data; MF=15 contains secondary photon energy distributions. A section refers to a particular reaction [for example, MT=2 is elastic scattering and MT=107 is the (n, α) reaction]. Every record contains the current MAT, MF, and MT values. Two materials are separated by a record with MAT=0 (the material-end or MEND record). Two files are separated by a record with MF=0 (the file-end or FEND record). Two sections are separated by a record with MT=0 (the section-end or SEND record). Finally, the tape is terminated with a record with MAT $= -1$ (tape-end or TEND record).

NJOY has a set of utility subroutines for locating desired positions on an ENDF tape.

`FINDF(MAT,MF,MT,NIN)`

> Search NIN backward or forward for the first record with this MAT, MF, MT. Issue a fatal error message if the record is not found.

`TOSEND(NIN,NOUT,NSCR,A)`

`TOFEND(NIN,NOUT,NSCR,A)`

`TOMEND(NIN,NOUT,NSCR,A)`

`TOTEND(NIN,NOUT,NSCR,A)`

> Skip forward past the next SEND, FEND, MEND, or TEND card on NIN. If NOUT and/or NSCR are nonzero, copy the records. Input and output files must be in the same mode.

The data on an ENDF tape are written in 7 different kinds of "structures", each of which has a binary and a formatted form (the words "coded," "formatted", and "BCD" will often be used interchangeably even though the actual representation might be ASCII or display code). The structures are: (1) TAPEID, a Hollerith

title for the tape; (2) CONT, a control record (includes SEND, FEND, MEND, and TEND); (3) LIST, a list of data items; (4) HOLL, a list of Hollerith words; (5) TAB1, a one-dimensional tabulation of data pairs; (6) TAB2, a two-dimensional tabulation control record; and (7) DICT, a directory or an index (it used to be called the "dictionary") to the sections found in the MAT. It should be noted that HOLL is a special case of LIST and DICT is a special case of CONT.

In binary mode, each structure is written as a single logical record as follows:

`TAPEID[MAT,MF,MT/A(I),I=1,17]` *

where MAT=tape number, MF=MT=0, and the Hollerith data are 16A4,A2;

`CONT[MAT,MF,MT/C1,C2,L1,L2,N1,N2]`

`LIST[MAT,MF,MT/C1,C2,L1,L2,N1,N2/A(I),I=1,N1]`

`HOLL[MAT,MF,MT/C1,C2,L1,L2,N1,N2/A(I),I=1,N1]`

where MF=1, MT=451, and each line of Hollerith characters is stored in A as 16A4,A2;

`TAB1[MAT,MF,MT/C1,C2,L1,L2,N1,N2/NBT(I),JNT(I),I=1,N1/X(I),Y(I),I=1,N2]`

where NBT and JNT are the interpolation table and Y(X) is the one-dimensional tabulation;

`TAB2[MAT,MF,MT/C1,C2,L1,L2,N1,N2/NBT(I),JNT(I),I=1,N1]`

where the interpolation table is to be used to control a series of N2 LIST or TAB1 structures that follow; and

`DICT[MAT MF,MT/0.,0.,MFS,MTS,NCS,MODS]`

where there is a record for each section in the material (MFS, MTS) giving the card count (NCS) for that section. For ENDF/B-V, MODS indicates the revision number for that section.

The ENDF procedure manuals[1, 2] explain how these structures are combined to represent various physical quantities.

In order to make these records practical, limits have been established that keep the record length below approximately 10 000 words. In BCD mode, each structure is broken up into many card images, each containing 6 data words, followed by MAT, MF, MT, and a line sequence number. There is no intrinsic limit to the length of a data structure written in BCD form because a program reading the data can normally be coded to use the data in "pages" of reasonable size. The MINX code[3] (the predecessor of NJOY) was forced to use BCD formats to handle the large tabulations found on PENDF tapes. Analysis showed that this code used

*In ENDF/B manuals, the slash is used as a logical divider. Replace it with a comma and add parentheses when constructing a FORTRAN I/O list.

more than 50% of its running time coding and decoding BCD formats. In order to eliminate this waste, a blocked binary format was developed for the ENDF data structures. A structure is divided up into several logical records of intermediate length (typically about 300 words), each having the following form:

```
[MAT,MF,MT,NB,NW/A(I),I=1,NW]
```

where NB is the number of words remaining in the data structure (the last record has NB=0). This type of record is compatible with the official ENDF binary record, but is also adaptable to paging methods. The page size can be chosen to optimize input/output rates for a particular computer system.

A set of utility subroutines has been devised to handle both blocked-binary and paged-BCD input and output.

`CONTIO(NIN,NOUT,NSCR,A,NB,NW)`

Read/write a control record to/from A (NB=0, NW=6). CONTIO uses ASEND, AMEND, etc. for END cards.

`LISTIO(NIN,NOUT,NSCR,A,NB,NW)`

Read/write the first record or page of a list record to/from A. If NB is not zero, continue with MOREIO, as illustrated in Examples 1 and 2 on the next page.

`TAB1IO(NIN,NOUT,NSCR,A,NB,NW)`

Read/write the first record or page of a TAB1 structure. If NB is not zero, use MOREIO.

`TAB2IO(NIN,NOUT,NSCR,A,NB,NW)`

Read/write a TAB2 structure (NB=0).

`MOREIO(NIN,NOUT,NSCR,A,NB,NW)`

Read/write continuation records or pages to/from the array A. Returns NB=0 after processing the last record or page.

`TPIDIO(NIN,NOUT,NSCR,A,NB,NW)`

Read/write the Hollerith tape identification record to/from array A (NB=0, NW=17).

`HDATIO(NIN,NOUT,NWCR,A,NB,NW)`

Read/write the first record or page of the Hollerith descriptive data (MF1, MT451) to/from A, taking account of the 16A4,A2 format needed in BCD mode. If NB is not zero, use MOREIO.

`DICTIO(NIN,NOUT,NSCR,A,NB,NW)`

Read/write the entire material directory to/from A. On entry, NW is the number of entries in the dictionary. MOREIO is not used.

LINEIO(NIN,NOUT,NSCR)

This routine is used by some of the other routines in this group to read and/or write a line of ENDF text containing real numbers. It writes blank fields at the ends of short lines.

HOLLIO(NIN,NOUT,NSCR)

This routine is used by HDATIO to read and/or write lines of Hollerith information using the 17-word 16A4,A2 format.

CXFP(X,F,S,N)

This routine is used by some of the other ENDF/B routines to prepare formatted output without the normal FORTRAN "E". Floating-point numbers are output as ±1.23456±NN or ±1.234567±N, depending on the size of the exponent.

ASEND(NOUT,NSCR)

AFEND(NOUT,NSCR)

AMEND(NOUT,NSCR)

ATEND(NOUT,NSCR)

Write an "end" record on the desired units.

In these calling sequences, the unit numbers can be positive, negative, or zero. Positive numbers mean BCD mode, negative numbers mean blocked-binary mode, and zero means the file corresponding to this position in the calling sequence is not used. All of these routines use one area of labeled common

```
COMMON/CONT/C1,C2,L1,L2,N1,N2,MAT,MF,MT,NSH,NSP,NSC
```

where C1 through MT have their usual ENDF meanings, NSH is the sequence number for NIN, NSP is the sequence number for NOUT, and NSC is the sequence number for NSCR. Two examples may help to make clear the use of these routines.

Example 1. Read All Data

```
      LOC=1
      CALL TAB1IO(NIN,0,0,A(1),NB,NW)
  110 IF (NB.EQ.0) GO TO 120
      LOC=LOC+NW
      CALL MOREIO(NIN,0,0,A(LOC),NB,NW)
      GO TO 110
  120 (process data in A)
```

Example 2. Paging

```
      CALL TAB1IO(NIN,0,0,A(1),NB,NW)
  110 (process this page of data in A)
      IF (NB.EQ.0) GO TO 120
      CALL MOREIO(NIN,0,0,A(1),NB,NW)
      GO TO 110
  120 CONTINUE
```

When NIN is BCD, paging is automatic. Positive and negative unit numbers can be mixed in TPIDIO, CONTIO, LISTIO, etc., when mode conversion is desired.

E. Buffered Binary Scratch Storage

During the execution of a program, large amounts of data often need to be stored in mass storage temporarily. In order to make such scratch storage as efficient as possible, NJOY includes a pair of utility subroutines that automatically buffer such data through fast memory to disk.

```
LOADA(I,A,NA,NTAPE,BUF,NBUF)

FINDA(I,A,NA,NTAPE,BUF,NBUF)
```

> Load or find data in a buffered binary scratch file. Here I is the data point number (I must increase, except I=1 causes a rewind and I<0 flushes the fast memory buffer to mass storage), A is the array containing data to be stored or the destination of data to be read, NA is the number of words to be transmitted (must be the same for all I), NTAPE is the logical unit number of the disk file, BUF is the fast memory buffer array, and NBUF is the length of the buffer array.

When a point is to be saved, LOADA stores it in BUF. When BUF becomes full, it is automatically dumped to disk. When a point is to be retrieved, FINDA checks to see whether the desired point is in BUF. If not, it reads through the disk until the desired point is in memory. It then returns the desired point. When NA is small with respect to NBUF, using LOADA/FINDA reduces the number of I/O operations dramatically.

Sometimes it is necessary to find a particular part of the buffered data. In such cases, use

```
SCANA(E,IP,NP,NA,NTAPE,BUF,NBUF) ,
```

where E is a value for the first of the NA words, and IP points to part of the data whose first word is either equal to E or is the first value less than E.

F. Dynamic Storage Allocation

In many large computer codes, storage requirements may change continually throughout the execution of a problem. If maximum use is to be made of the available memory, it is necessary to reallocate and repack storage in response to the requirements of the calculation. In NJOY, these functions are handled by the STORAG package of six subroutines.

STORAG(IAMAX,NIDMAX,IPR,A)

Initialize variably dimensioned dynamic storage allocation system for the container array A. IAMAX = length of container array. NIDMAX = maximum number of data identifiers that will be needed at one time. IPR = print flag (normally 0, use 1 to suppress most routine messages).

RESERV(ID,NWORDS,INDEX,A)

Reserve NWORDS in A for the data set identified by ID. ID is a Hollerith name with up to 4 characters. Space will be allocated at the top of A if possible. If insufficient space is available, A will be repacked, and another attempt to reserve space will be made. If NWORDS=-1, repack A and assign all available words to this ID. INDEX points to the first word for data set ID in A.

RELEAS(ID,NWORDS,A)

Release all but NWORDS of the space assigned to ID in A. NWORDS=0 deletes this ID. If NWORDS is less than zero, this ID and all ID entries above it are deleted. Note that repacking of A only takes place when the released space is really needed (see RESERV).

FINDEX(ID,INDEX,A)

Find the index for the data set ID. Using FINDEX is good practice if there is any chance that A might have been repacked since RESERV was called.

USAG(A)

Output a usage report for variably-dimensioned storage. The report may be an underestimate if the option of reserving storage with NWORDS=-1 was used, followed by a release with NWORDS=0. USAG has no way of knowing how much of the memory was actually used.

MOVE(A1,A2,NWORDS)

This routine is used by RESERV to move data in memory while repacking.

The NWORDS=-1 option in RESERV is useful when the number of words in a data set is not known in advance—an example:

```
NW=-1
CALL RESERV('SIG',NW,LSIG,A)
READ(NIN) NW,(A(LSIG+I-1),I=1,NW)
CALL RELEAS('SIG',NW,A)
```

STORAG prints out routine messages (if IPR=0) so that the user can monitor the use of memory. The following example from THERMR illustrates several characteristics of STORAG.

```
 1  STORAG  10/20000
 2  ID SCR   1/2050
 3  ID BUFO  2/3050
 4  ID BUFN  3/4050
 5  ID STK   4/4110
 6  ID FL    5/19963
 7  XX FL    406
 8  XX STK    -1
 9  ID E     4/4095
    ...
10  USAGE 4516/20000
```

In line 1, STORAG is initialized with 20 000 words of memory for up to 10 identifiers. In lines 2, 3, 4, and 5, space is reserved for SCR, BUFO, BUFN, and STK. The number before the slash is the ordinal number assigned to the identifier, and the second number is the total amount of storage used so far. In line 7, space for FL was reserved with NWORDS=-1. Therefore, 20 000 words less the STORAG table were allocated. If repacking had been necessary, a "REPACKING" message would have appeared here. The program determined that only 406 words were needed for FL, and the remainder of the storage was released in line 7. The maximum storage used to this point was 4110 + 406 = 4516. Further on, the code was finished with STK and FL, and both were released by a single call with NWORDS=-1 as indicated by line 8. Finally, line 9 shows a new identifier being assigned. Note that position 4 in the STORAG table was reused. The usage message on line 10 gives the maximum amount of storage used by this module compared to the available storage.

The STORAG system is compact and easy to use. The overhead required to use it is very small, unless frequent repacking is required.

G. ENDF/B Utility Routines

There are several operations performed on ENDF/B data that are needed in so many other modules that it is practical to put them into the NJOY level.

TERP1(X1,Y1,X2,Y2,X,Y,I)

> Interpolate for $y(x)$ between $y_1(x_1)$ and $y_2(x_2)$ using the ENDF interpolation law I (I=1 means y=y_1, I=2 means y is linear in x, I=3 means y is linear in $\ln(x)$, I=4 means $\ln(y)$ is linear in x, and I=5 means $\ln(y)$ is linear in $\ln(x)$).

TERPA(Y,X,XNEXT,IDIS,A,IP,IR)

Interpolate for $y(x)$ in the TAB1 structure in array A. The routine searches for the correct interpolation range starting from IP and IR (initialize to 2 and 1 for first call). It returns XNEXT, the next x value in the tabulation. IDIS is set to 1 if there is a discontinuity at XNEXT; it is zero otherwise).

```
GETY1(X,XNEXT,IDIS,Y1,ITAPE,A)
```

```
GETY2(X,XNEXT,IDIS,Y2,ITAPE,A)
```

Find $y(x)$ in a TAB1 structure starting at the current location on ITAPE by paging the data through array A. GETY1 and GETY2 are identical for occasions when two different tapes are being searched at the same time. XNEXT and IDIS behave as in TERPA. The array A must be at least NPAGE+50 words in length. These routines are normally used to retrieve cross sections from MF=3.

```
GRAL(XL,YL,XH,YH,X1,X2,I)
```

This function returns the integral from x_1 to x_2 of an ENDF function with interpolation law I (see TERP1). XL, YL, XH, and YH are the low and high limits of the interpolation panel.

```
INTEGA(F,X1,X2,A,IP,IR)
```

Integrate the TAB1 function stored in A from x_1 to x_2. The routine automatically determines the correct interpolation law for each panel or fraction of a panel and uses GRAL to compute each part of the integral. Set IP=2 and IR=1 on the first call to INTEGA. In subsequent calls, the previous values of IP and IR will usually provide a good starting point for searching in the TAB1 structure.

H. Math Routines

Several mathematics routines are also included in the NJOY modules for use by other modules.

```
LEGNDR(X,P,NP)
```

Generate Legendre polynomials $P_\ell(x)$. The $\ell = 0$ value is in P(1), the $\ell = 1$ value in P(2), etc. NP is the maximum Legendre order produced, so the largest index for P is NP+1.

```
E1(X)
```

Compute the first-order exponential integral function $E_1(x)$.

```
GAMI(A,X)
```

Compute the incomplete gamma function $\gamma(a, x)$.

```
GAM1(A)
```

Compute the gamma function $\Gamma(a)$.

I. CCCC I/O Routines

The CCCC standards[4] were established before FORTRAN-77 was developed. In those days, input/output routines tended to be machine dependent. In order to help improve the portability of reactor physics codes, a set of standard I/O procedures were defined. The idea was that if every CCCC-compliant code used these routines, only the routines would have to be adapted when moving to a new computer system. The rest of the code would be completely transportable. The CCCC-compatible modules of NJOY (namely, CCCCR, MATXSR, and RESXSR) only need very simple versions of these procedures.

`REED(N,IREC,X,L,MODE)`

Read a CCCC binary record from unit `N`. `IREC` is the index for the record. If `IREC` is not the next record, the routine will use `SKIPRZ` to move backward or forward in the file before executing the read operation. If `IREC=1`, the file on unit `N` will be rewound. The parameter `L` gives the number of words to be read in the array `X`. `MODE` is ignored.

`RITE(N,IREC,X,L,MODE)`

Write a CCCC binary record on unit `N`. The records must be written sequentially, and `IREC=1` rewinds the file before the first record is written. `L` is the number of words from array `X` to be written. `MODE` is ignored.

J. System-Dependent Routines

As much as possible, actions that are likely to be machine dependent have been put into subroutines or functions in the NJOY level. The I/O routines have already been discussed. Some other places with possible machine problems follow.

`BANNER`

This subroutine prints the NJOY banner on the output file. It includes a user field `LAB`, which should be changed to properly identify the user's installation. It also includes a variable `MX`, which can be used to indicate which machine was used at large computing centers; at Los Alamos, use `CALL MACH(MX)`.

`TIMER(TIME)`

Returns the run time in seconds. The meaning of this number may vary from system to system. It might be central-processor (CP) time, or at some installations, it may include other factors, such as I/O time or memory charges. This makes it difficult to compare NJOY runs on different systems. This routine will have to be revised for many systems.

`DATER(HDATE)`

Returns the date as an 8-character string in one of the forms mm/dd/yy or ddmmyy; for example, 11/15/90 or 03jun91. The call to the system date routine will normally have to be changed to adapt `DATER` to the local system. It may also be necessary to convert between integer and character formats for the date.

`WCLOCK(HTIME)`

Returns the "wall clock" time. This is the time of day that the NJOY run started, and it shouldn't be confused with the elapsed CP time for the run as returned by TIMER. The time is represented as an 8-character string in the form hh:mm:ss; for example, 12:13:47. The user is free to use a 24-hour convention for time.

SIGFIG

Because of the many comparisons and searches that it makes, NJOY often has to match two numbers that are different only in the few least significant bits. This routine is intended to make such numbers exactly equal to each other by truncating the numbers to a given number of digits and removing any low-significance junk resulting from nonterminating binary fractions. This problem is not so common on short-word-length machines, but it might still be necessary to convert this routine for some machines.

K. Error and Warning Messages

Starting with version 91.0, NJOY has a pair of standard routines for printing fatal error messages and warning messages. This helps to enforce consistency in the messages, insulates other subroutines from the complexities of the system (for example, I/O units, "console", "standard error file"), and provides a site for machine-dependent error handling, including such things as saving "drop files" and generating trace-back listings.

ERROR(FROM,MESS1,MESS2)

This subroutine should result in a fatal error exit and must be adjusted to reflect the local system. Special features such as traceback information or saving files for later analysis can be performed here. FROM is a character string containing the name of the procedure that called ERROR, and MESS1 and MESS2 are two 60-character strings containing messages describing the error. MESS2 is not printed if it is empty.

MESS(FROM,MESS1,MESS2)

This routine is for nonfatal warning messages. FROM is the routine that called it. It prints FROM, MESS1, and MESS2 (if not empty), and returns to the calling routine.

The actual error messages produced by functions and subroutines in the NJOY module are listed below, including explanations of the meaning of the errors and suggested steps to alleviate them.

ERROR IN NJOY***ILLEGAL OPTION

Only 0 or 1 is allowed. Fix the input file and start again.

ERROR IN NJOY***ILLEGAL ENDF/B VERSION NUMBER

Only 3 through 6 are allowed. ENDF/B-III data are actually processed with IVERF=4.

ERROR IN NJOY***ILLEGAL MODULE NAME

Check spelling, and check for missing (/) or incorrect item counts in the preceding module. Only the first four characters of each name are used. This error message is generated directly by NJOY instead of using ERROR; for interactive input, the user is given a chance to recover from typing in a bad module name.

ERROR IN OPENZ***ILLEGAL UNIT NUMBER

ERROR IN CLOSZ***ILLEGAL UNIT NUMBER

Units less than 10 are reserved for the system.

ERROR IN TOMEND***MODE CONVERSION NOT ALLOWED

ERROR IN TOFEND***MODE CONVERSION NOT ALLOWED

ERROR IN TOSEND***MODE CONVERSION NOT ALLOWED

Input and output units must both be binary or both be BCD. Check the signs of the unit numbers in the input file.

ERROR IN FINDF***MAT---MF---MT---NOT ON TAPE

Desired section cannot be found. Either the wrong tape was mounted, or there is a mistake in the input deck.

ERROR IN SCANA***INITIAL IP NE 0

Must be called with IP=0.

ERROR IN SCANA***DID NOT FIND ENERGY ---

Energy requested is greater than the highest energy in the LOADA/FINDA file.

ERROR IN GRAL***X2 LT X1

The integration interval is bad.

ERROR IN STORAG***STORAGE EXCEEDED

There is not enough storage allocated to hold even the directory table.

ERROR IN RESERV***STORAGE EXCEEDED. NEED---MORE WORDS FOR ID---

Container array is not large enough to hold desired data, even after repacking. The message gives an estimate of the additional storage required. This usually means that the size of the container array will have to be increased. See the instructions for the module that reported the error for more information.

ERROR IN RESERV***ID---ALREADY DEFINED

An ID must be released before being reassigned. This message probably indicates that there is a coding error in NJOY.

ERROR IN RESERV***POINTER SEQUENCE ERROR

The directory at the start of the container array has probably been clobbered. The normal source of this problem is a coding error that used A(I) as a normal array with small values for I. Normally, a pointer is needed, as in A(ILOC+I).

ERROR IN RESERV*EXCEEDED MAXIMUM NUMBER OF ID-S**

See **NIDMAX** in **STORAG**.

ERROR IN RESERV*REQUESTED RESERVE OF ZERO WORDS**

Check coding that called **RESERV**.

ERROR IN RELEAS*ID---NOT DEFINED**

Check coding and spelling.

ERROR IN RELEAS*ATTEMPT TO RELEASE MORE WORDS THAN STORED**

Is self-explanatory. Check coding.

ERROR IN FINDEX*ID---NOT DEFINED**

Check coding and spelling.

ERROR IN SIGFIG*IX IS BAD**

ERROR IN SIGFIG*TOO MANY INNER ITERATIONS**

The code has trouble truncating the number of significant figures.

MESSAGE FROM E1---X.LE.0, RESULT=+1.E38

The x value is illegal for the $E_1(x)$ function.

MESSAGE FROM GAMI---SECOND ARGUMENT NEGATIVE, RESULT=-1E38

MESSAGE FROM GAMI---ARGUMENTS FORCE RESULT OF +1E38

Parameters are illegal for incomplete gamma function $\gamma(a,x)$.

MESSAGE FROM GAM1---ABS(ARG) TOO LARGE, RESULT=1E38

MESSAGE FROM GAM1---ARG NOT POSITIVE, RESULT=1E38

The x value is illegal for the gamma function $\Gamma(x)$.

L. Coding Details

The NJOY routine of the NJOY code starts with a "program" statement. This type of statement is not standard in FORTRAN-77, and it may have to be changed or omitted entirely for particular systems. As an example, the UPCTSS file of changes for the CTSS operating system for Cray computers uses the program statement to assign the system I/O units.

The code continues with a large block of comment cards that gives a short description of the NJOY module and specifications for the user's input lines. (The term "card" is used out of respect for the past; this usage should not be taken to imply that a real card that can be "folded, spindled, or mutilated" has to be used.) These blocks of comments cards occur at the beginning of every NJOY module. It is a good idea to check the input instructions in the comment cards for the current version in order to see whether there have been any changes from the input instructions reproduced in this manual.

The first step in the executable portion of the code is to set the constants in the two global common arrays MAINIO and UTIL. The quantity NPAGE is the page size used for blocked-binary ENDF records. It must be a multiple of 6 (since most ENDF card images have 6 fields) and 17 (since Hollerith lines use the format "16A4,A2"); therefore, a value of 306 was chosen. If a programmer should desire to use larger pages, the line

```
NPAGE=(NPAGE/102)*102
```

assures that the common divisors of 6 and 17 are preserved. The next few lines of code define the system I/O units. They often have to be changed when moving from system to system, and some systems require that explicit OPEN statements be given for the system units. In addition, the unit assignments change between batch and interactive input modes (see IOPT). Examples of some alternate versions of this coding will be found in the various machine-dependent UPD files included with the NJOY distribution.

The next step is to print out the NJOY "banner," and to read in the ENDF version number. Note that for interactive mode, a message is written to the user's terminal "prompting" for the desired data. **NJOY** now starts an infinite loop, reading in module names, and executing the requested module, until the name "STOP" is read. Note that giving NJOY an incorrect module name is not fatal in interactive mode; the code prompts the user for a new choice.

Subroutine **BANNER** is simple, but there are a few items that might have to be changed when making a new version or transporting the code to a new system. Updating **IVERS** was discussed in connection with UPD and the NJOY version-control system. The **LAB** name can be changed at every NJOY site. Some large computer systems, such as the Los Alamos National Laboratory facility, balance loads by moving jobs to different computers. In this case, the user can call subroutine **MACH** to fill in the **MX** parameter. The NJOY banner will then indicate which machine was used for the run.

The default versions of **ERROR** and **MESS** simply write the one or two lines of message characters to the appropriate units. The **LENR** function is used to measure the real length of a FORTRAN-77 character string by removing trailing blanks. By changing **NSYSE**, messages can be directed to the user's terminal, to a standard error unit, or left to appear on the output file only. The default fatal error exit "**STOP 77**" can be changed to cause a "traceback" to the subroutine that called **ERROR**, and backward through the stack of subroutines that called it. Also, some systems can cause a drop file or "core" file to be generated for post-mortem analysis with a debugging program. As discussed in Section I of this report, these kinds of changes can be introduced using a machine-dependent ident in UPD.

Subroutine **FREE** was introduced into NJOY before the more standard **READ*** command became widely available, and since **FREE** interfaces to the code differently than **READ***, NJOY has stayed with its tried and true method for reading free-format input. The main loop over input cards goes through statement number 100. Each card is read in as 80 separate characters. The code then loops through these characters looking for the start and end of the various types of fields allowed by the **FREE** syntax. Numeric fields are turned into single-precision floating-point numbers. String fields are turned into either single-precision or double-precision Hollerith values, depending on the values of **NCW** and **MACHWD**. Subroutine **PACKC** is used to insert characters into the accumulating Hollerith variables, and function **LETTER** is used to check characters to see if they are numeric or alphabetic.

The next four routines in NJOY provide a uniform way of handling input and output files. The unit numbers obey the following convention: zero means do

nothing, negative means a binary unit, and positive means a coded unit (BCD, ASCII, EBCDIC, etc.). Unit numbers between 10 and 19 are scratch files, which will be automatically destroyed at the end of the job. Therefore, REPOZ simply takes the absolute value of the unit number and calls rewind if it is nonzero. Similarly, CLOSZ takes the absolute value of the unit number and closes it if nonzero and not a scratch file. Subroutine SKIPRZ skips forward with dummy reads of the correct type for the sign of the unit, and it skips backward with backspace. Note that this can be very inefficient for some systems. For example, the VAX executes backspace by first rewinding and then skipping forward by $n-1$ records. Making successive calls to backspace as done in the "DO 130" loop would be very expensive. It is better to rewind before calling SKIPRZ; then only forward skips would be required to get to the desired record. Finally, subroutine OPENZ is used to open files with the desired characteristics (that is, binary or formatted, new or old, scratch or permanent). Standard FORTRAN-77 statements are used. Note that the file names are constructed to have the forms TAPE20, TAPE21, etc. Scratch files will have whatever names are standard for the system being used.

Subroutines TIMER, DATER, and WCLOCK provide standard interfaces to system routines that normally have different names and return different kinds of answers on different computer systems. The forms of the desired outputs are indicated in the comment cards, and some examples of implementing these functions on various computer systems are included as UPD machine-dependent idents in the NJOY distribution.

As discussed in Section D of this chapter, NJOY uses files in ENDF-like formats for most communications between modules. The routines CONTIO, LISTIO, TAB1IO, TAB2IO, MOREIO, TPIDIO, HDATIO, and DICTIO are used to read, write, copy, or convert the mode of binary or formatted records on these interface files. They all have similar structures. First, records are read using binary or coded commands, depending on the sign of the unit number. Then the records are written back out, once again using a mode that depends on the sign of the unit number. Unit numbers can be zero, in which case that unit is not used. Note that all fields in records (except the identification fields like MAT, MF, and MT) are read and written in single-precision real form in A. When these numbers are read, they are converted into integers for the L1H, L2H, N1H, and N2H fields using the FORTRAN nearest-integer function, NINT. If programmers choose to use a number in the array A directly, they should be careful to make the same conversion. Subroutine LINEIO is used by several of these routines to read or write a line of floating-point numbers. It has two special features. First, any empty fields at the end of a line are filled in with blanks using FORTRAN variable-format operations. These kinds

of operations are not included in the ANSII standard, even though many compilers allow them for backward compatibility. The second feature is the construction of floating-point numbers without the normal FORTRAN "E" using subroutine CXFP. Subroutine TABLIO is used by TAB1IO and TAB2IO to read and write ENDF-style interpolation tables. Subroutine HOLLIO is used to read or write lines of Hollerith information. Note that Hollerith data is represented using 17 fields per card, but all other ENDF data uses 6 fields per card.

Subroutine CXFP simply breaks up the single-precision floating-point number X into a fraction part F, a sign part S, and an exponent part N. The number of digits in the fraction part depends on the number of digits in the sign part in order to maximize the precision of the formatted representation of X. Subroutine LINEIO does the actual work of writing these three values into the 11-column fields on the output "cards."

As discussed above, ENDF tapes are divided into materials, files, and sections by special end cards with names like MEND, FEND, SEND, and TEND. The subroutines TOSEND, TOFEND, TOMEND, and TOTEND can be used to move from the current location to one of these end cards, optionally copying the records as they go. Since these routines do not know the actual structure of the records, they cannot change the mode of the data while copying. They simply read in a record, write it out again if desired, and watch for the specified end card.

The parallel set of routines ASEND, AFEND, AMEND, and ATEND write one of the requested end cards to one or two different output files using the mode defined by the sign of the unit number. The NJOY convention is to use blanks for the first 66 columns of formatted end cards in order to make them easy to see on ENDF listings. The tapes as received from the National Nuclear Data Center (NNDC) of the Brookhaven National Laboratory always have these fields filled in with numerical values of zero. The MODER module automatically changes ENDF tapes from the BNL convention to the NJOY one.

Subroutine FINDF is used very frequently in NJOY modules to search through an ENDF-type tape for a desired section (MAT, MF, MT). Since the three numbers that describe sections are always arranged in order, it is possible to search both up and down. On entry, the routine reads the first card and decides if it has to read up or down to find the desired section. It then continues by reading records in the proper direction until it comes to the desired section. It then backs up by one record so that the next read after the call to FINDF will read the first record of the desired section. If the desired section is not found, a fatal error message will be issued. FINDF can go into an infinite loop if the MAT number requested is

smaller than the number for the first material on the tape (thus FINDF is moving backwards) and the number in the MAT field on the tape identification record is larger than the material number for the first material. Some systems have an "IF (BOI)" function that can be used to detect when the tape is at the "beginning of information."

The LOADA/FINDA system was discussed in Section II-E. The data representation consists of a fairly large buffer array BUF of length NBUF. This buffer contains a number of component blocks, each of length NA. Therefore, it is easy to compute the location of any block using modular arithmetic on the block index I. If the location is currently in the buffer array, the block can be read or written. Otherwise, the associated scratch file NTAPE must be repositioned before being read or written. The files NTAPE have to be written sequentially, but they can be searched in any order. Subroutine SCANA takes advantage of this to locate the block whose first element is closest to the input parameter E.

Subroutine TERP1 is used to interpolate between two points X1,Y1 and X2,Y2 using ENDF interpolation law I. The results for Y at X are given by simple formulas. No tests are made for values that give illegal arguments for the FORTRAN log and exponent functions. Therefore, fatal arithmetic errors from TERP1 are fairly common. Going slightly out of order, GRAL is a routine that computes integrals inside the endpoints of a segment represented using the ENDF interpolation laws. Once again, simple formulas are used, and it is also possible to get arithmetic errors from the log and exponent functions with this routine.

It is usually necessary to interpolate or integrate functions that involve more than one panel of ENDF interpolated data. Most commonly, this occurs for data in TAB1 format. If the TAB1 record is small enough so that it can fit into an array A in memory, subroutine TERPA can be used to search for the panel that contains X, and to interpolate for Y at X. Similarly, INTEGA can be used to compute the integral from X1 to X2 by integrating over all the panels and parts of panels within these limits. TERPA uses TERP1, and INTEGA uses GRAL.

NJOY works with some very large TAB1 records that cannot fit into memory. For these cases, GETY1 and GETY2 can be used to interpolate for Y at X. Array A only has to be big enough for one page of data (plus a little extra space for parameters like C1H, N2H, and TAB1 interpolation tables). If the desired X is not in memory, new pages of data are read from ITAPE until the desired value is found. The subroutine then uses TERP1 to compute Y. Note that zero is returned if X is outside the range of the table, and XNEXT=1.E10 when the last point is returned. GETY1 and GETY2 are almost identical so that numbers can be retrieved independently

from two different input tapes at the same time.

Subroutine SIGFIG is used to control the precision of numbers in several NJOY modules. This is a difficult problem because the true representation of numbers in the machine is as non-terminating binary fractions. In addition, different representations for floating-point numbers are used in different computer systems, and significant-figure truncation can be machine-dependent.

The NJOY variable-dimensioned storage package keeps its directory and all the data in a large container array, IA. The format for the directory is given in the following table:

I	IA(I)	Meaning
1	IAMAX	length of array IA
2	NIDMAX	maximum number of data IDs
3	NID	current number of data IDs
4	IPR	print flag
5	0	not used
6	NUSED	max number of words used
7	ID	data ID name #1
8	3*NIDMAX+7	pointer to data for ID #1
9	NWORDS	length of data for ID #1
10	ID	data ID name #2
11	INDEX	pointer to data for ID #2
12	INDEX	length of data for ID #2
...	...	continue for additional IDs

The first step is to call STORAG, which initializes the directory table for the given container array IA(IAMAX), maximum number of IDs expected NIDMAX, and print flag IPR. All the ID directory fields are initialized to zero.

Subroutine RESERV can then be used to allocate storage for a particular data block. The data-block names ID are limited to 4 characters so that single-precision variables can be used on all machines. RESERV first tries to add the new name at the top of the ID table (that is, after ID NID). If successful, the new data pointer will be equal to the data pointer for ID NID plus the length of the data block for ID NID. The directory fields for the new ID are filled in, NID is incremented, and the pointer to the new data block is returned. The attempt to reserve a block can be unsuccessful because there are no ID slots left (NID=NIDMAX) or because there is not enough space left in the container array. In either case, the routine drops down to statement number 130 to repack the container array. It does this by going through the directory and searching for unused ranges of words in IA. Unused ranges can result from entire IDs being released (see RELEAS) or from part of the words in an ID being released. RESERV fills in the empty spaces by moving

the contents of IA down with subroutine MOVE, and it updates the directory to describe the new locations of the data blocks.

Subroutine RELEAS is used to change directory entries to indicate that all or some of the words for a given ID or range of IDs have been released. Releasing an entire ID causes the name to be cleared; releasing just part of an ID causes NWORDS to be adjusted. Note that no data are moved by a call to RELEAS. Data are repacked only when necessary in order to improve efficiency (see RESERV).

Subroutine USAG prints out a usage report at the end of a module. These messages can be useful when trying to minimize the amount of memory used by NJOY. Unfortunately, the STORAG package often reserves storage using NWORDS=-1, which requests that all of the remaining storage be allocated to an ID. If this ID is later released using NWORDS=0, there is no way for USAG to determine how much storage was really used, and the reported value may be too small.

The data pointer for a new data block is reported when it is reserved, but if repacking takes place, many of the pointers may change. Therefore, it is good practice to call FINDEX whenever a pointer to a data block is needed. It simply scans through the directory in IA until it finds the requested ID and returns the associated pointer.

Subroutine LEGNDR is used to compute Legendre polynomials by recursion. The formulas are as follows:

$$P_0(x) = 1., \tag{1}$$

$$P_1(x) = x, \tag{2}$$

$$P_{\ell+1)}(x) = \frac{(2\ell-1)xP_\ell(x) - (\ell-1)P_{\ell-2}}{\ell}, \tag{3}$$

where P_0 is stored in P(1), P_1 is stored in P(2), and so on.

There are several math routines that are needed in various NJOY modules; they are placed in this level of the code for convenience. The functions E1, GAMI, and GAM1 were taken from a Los Alamos National Laboratory math library and rewritten using the NJOY coding standards. These routines are often available on standard math libraries at large computer installations; it may be advantageous to replace the calls to the internal versions with the optimized versions from the external math library.

The standard CCCC binary sequential I/O routines REED and RITE are used in CCCCR, MATXSR, and RESXSR. They are described in detail in CCCCR, Chapter XIV of this manual.

M. References

1. R. Kinsey, "ENDF-102, Data Formats and Procedures for the Evaluated Nuclear Data Files, ENDF," Brookhaven National Laboratory report BNL-NCS-50496 (ENDF 102), 2nd. Ed. (ENDF/B-V) (October 1979).

2. P. F. Rose and C. L. Dunford, Eds., "ENDF-102, Data Formats and Procedures for the Evaluated Nuclear Data File, ENDF-6," Brookhaven National Laboratory report BNL-NCS-44945 (July 1990).

3. C. R. Weisbin, P. D. Soran, R. E. MacFarlane, D. R. Harris, R. J. LaBauve, J. S. Hendricks, J. E. White, and R. B. Kidman, "MINX, A Multigroup Interpretation of Nuclear X-Sections from ENDF/B," Los Alamos Scientific Laboratory report LA-6486-MS (ENDF-237) (1976).

4. R. D. O'Dell, "Standard Interface Files and Procedures for Reactor Physics Codes, Version IV," Los Alamos Scientific Laboratory report LA-6941-MS (September 1977).

III. RECONR

The RECONR module is used to reconstruct resonance cross sections from resonance parameters and to reconstruct cross sections from ENDF/B nonlinear interpolation schemes. The output is written as a pointwise-ENDF (PENDF) tape with all cross sections on a unionized energy grid suitable for linear interpolation to within a specified tolerance. Redundant reactions (for example, total or inelastic) are reconstructed to be exactly equal to the sum of their parts at all energies. The resonance parameters are removed from File 2, and the material directory is corrected to reflect all changes. RECONR has the following features:

- Efficient use of dynamic storage allocation and a special stack structure allow large problems to be run on small machines.
- The unionized grid improves the accuracy, usefulness, and ENDF/B compatibility of the output. All summation cross sections are preserved on the union grid.
- A correct directory of the output tape is provided.
- Approximate $\psi\chi$ Doppler broadening may be used in some cases to speed up reconstruction for narrow-resonance materials.
- A resonance-integral criterion is added to the normal linearization criterion in order to reduce the number of points added to the tabulation to represent "unimportant" resonances.
- All ENDF-6 formats are handled except Generalized R-Matrix parameters, energy-dependent scattering radius, and the calculation of angular distributions from resonance parameters.

This manual describes RECONR in versions of NJOY through 91.91.

A. ENDF/B Cross Section Representations

A typical cross section derived from an ENDF/B evaluation is shown in Fig. 1. The low-energy cross sections are "smooth". They are described in File 3 (see Section II.D for a review of ENDF/B nomenclature) using cross-section values given on an energy grid with a specified law for interpolation between the points. In the resolved resonance range, resonance parameters are given in File 2, and the cross sections for resonance reactions have to be obtained by adding the contributions of all the resonances to "backgrounds" from File 3. At still higher energies comes the unresolved region where explicit resonances are no longer defined. Instead, the cross section is computed from statistical distributions of the resonance parameters given in File 2 and backgrounds from File 3 (or optionally taken directly from

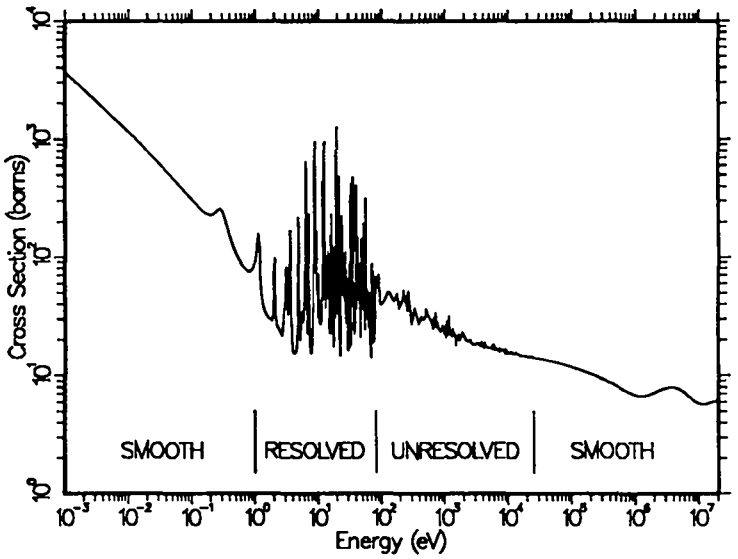


Figure 1: A typical cross section reconstructed from an ENDF/B evaluation using RECONR. The smooth, resolved, and unresolved energy regions use different representations of the cross sections. This is the total cross section for ^{235}U from ENDF/B-V.

File 3 as for smooth cross sections). Finally, at the highest energies, the smooth File 3 representation is used again.

For light and medium-mass isotopes, the unresolved range is usually omitted. For the lightest isotopes, the resolved range is also omitted, the resonance cross sections being given directly in the "smooth" format. In addition, several different resonance representations are allowed (for example, Single-Level Breit-Wigner, Multilevel Breit-Wigner, Reich-Moore, and so on). It is the purpose of RECONR to take all of these separate representations and produce a simple cross section versus energy representation like the one shown in Fig. 1.

B. Unionization and Linearization Strategy

Several of the cross sections found in ENDF/B evaluations are summation cross sections (for example, total, inelastic, and sometimes (n,2n) or fission), and it is important that each summation cross section be equal to the sum of its parts. However, if the partial cross sections are represented with nonlinear interpolation schemes, the sum cannot be represented by any simple interpolation law. A typical case is the sum of elastic scattering (MT=2 interpolated linearly to represent a constant) and radiative capture (MT=102 interpolated log-log to represent 1/v).

The total cross section cannot be represented accurately by either scheme unless the grid points are very close together. This effect leads to significant balance errors in multigroup transport codes and to splitting problems in continuous-energy Monte Carlo codes.

Furthermore, the use of linear-linear interpolation (that is, σ linear in E) can be advantageous in several ways. The data can be plotted easily, they can be integrated easily, cross sections can be Doppler-broadened efficiently (see BROADR), and, finally, linear data can be retrieved efficiently in continuous-energy Monte Carlo codes.

Therefore, RECONR puts all cross sections on a single unionized grid suitable for linear interpolation. As described in more detail below, RECONR makes one pass through the ENDF/B material to select the energy grid, and then a second pass to compute cross sections on this grid. Each cross section on the PENDF tape (except for the summation cross sections) is exactly equal to its ENDF/B value. The summation cross sections are then obtained by adding up the partial cross sections at each grid point.

While RECONR is going through the reactions given in the ENDF/B evaluation, it also checks the reaction thresholds against the Q value and atomic weight ratio to the neutron A (AWR in the file) given for the reaction. If the condition

$$\text{threshold} \geq \frac{A+1}{A} Q \tag{1}$$

is not satisfied, the threshold energy is moved up to satisfy the condition (an informative message is printed if the change exceeds 0.1%).

If desired, the unionized grid developed from the ENDF/B file can be supplemented with "user grid points" given in the input data. The code automatically adds 1.E-5 eV, 0.0253 eV, and 20 MeV to the grid if they are not already present.

There are special problems with choosing the energy grid in the unresolved range. In some cases, the unresolved cross section is represented using resonance parameters that are independent of energy. The cross sections are not constant, however, but have a shape determined by the energy variation of neutron wave number, penetrability factors, and so on. RECONR handles this case by choosing a set of energies (about 10 per decade) to be used to calculate the cross sections; the set of energies gives a reasonable approximation to the result intended. For evaluations that use energy-dependent resonance parameters, it is supposed to be sufficient to compute the unresolved cross sections at the given energies and to use interpolation on the cross sections to obtain the appropriate values at other

energies. However, some evaluations carried over from earlier versions of ENDF/B were not evaluated using this convention, and cross sections computed using cross-section interpolation are not sufficiently accurate. RECONR detects such cases by looking for large steps between the points of the given energy grid. It then adds additional energy grid points using the same 10-per-decade rule used for energy-independent parameters.

C. Linearization and Reconstruction Methods

Linearization (LUNION) and resonance reconstruction (RESXS) both function by inserting new energy grid points between the points of an original grid using an "inverted stack". The general concepts involved are illustrated with a simple example shown in Fig. 2.

The stack is first primed with two starting values. For linearization, they will be two adjacent points on the original union grid. For reconstruction, they will usually be the peaks or half-height energies of resonances. The stack is said to be inverted because the lower energy is at the "top" (I=2).

This interval or panel is now divided into two parts, and the cross section computed at the intermediate point is compared to the result of linear interpolation between the adjacent points. If the two values do not agree within various criteria, the top of the stack is moved up one notch (I=3), and the new value is inserted (I=2). The code then repeats the checking process for the new (smaller) interval at the top of the stack. The top of the stack rises until convergence is achieved for the top interval. The top energy and cross section are then saved on a scratch file, the stack index is decremented, and the checks are repeated. This process is continued with the top of the stack rising and falling in response to the complexity of the cross section until the entire panel ΔE has been converged (I=1). The stack is then reprimed with the bounds of the next panel. The process continues until the entire energy range for linearization or reconstruction has been processed.

This stack logic enables a panel to be subdivided into parts as small as $\Delta E/2^n$ where n is the stack size (currently 20), and several different cross sections (elastic, capture, fission) can easily be stored in arrays of this size. By contrast, RESEND used several arrays 500 words long and sometimes ran out of storage while subdividing between resonances.

Intervals are subdivided differently for linearization and resonance reconstruction. In the latter case, the interval is simply divided in half as in RESEND. For linearization, the method developed by D. R. Harris for MINX[1] is used. Analytic formulas are used to choose the optimum intermediate point; this turns out

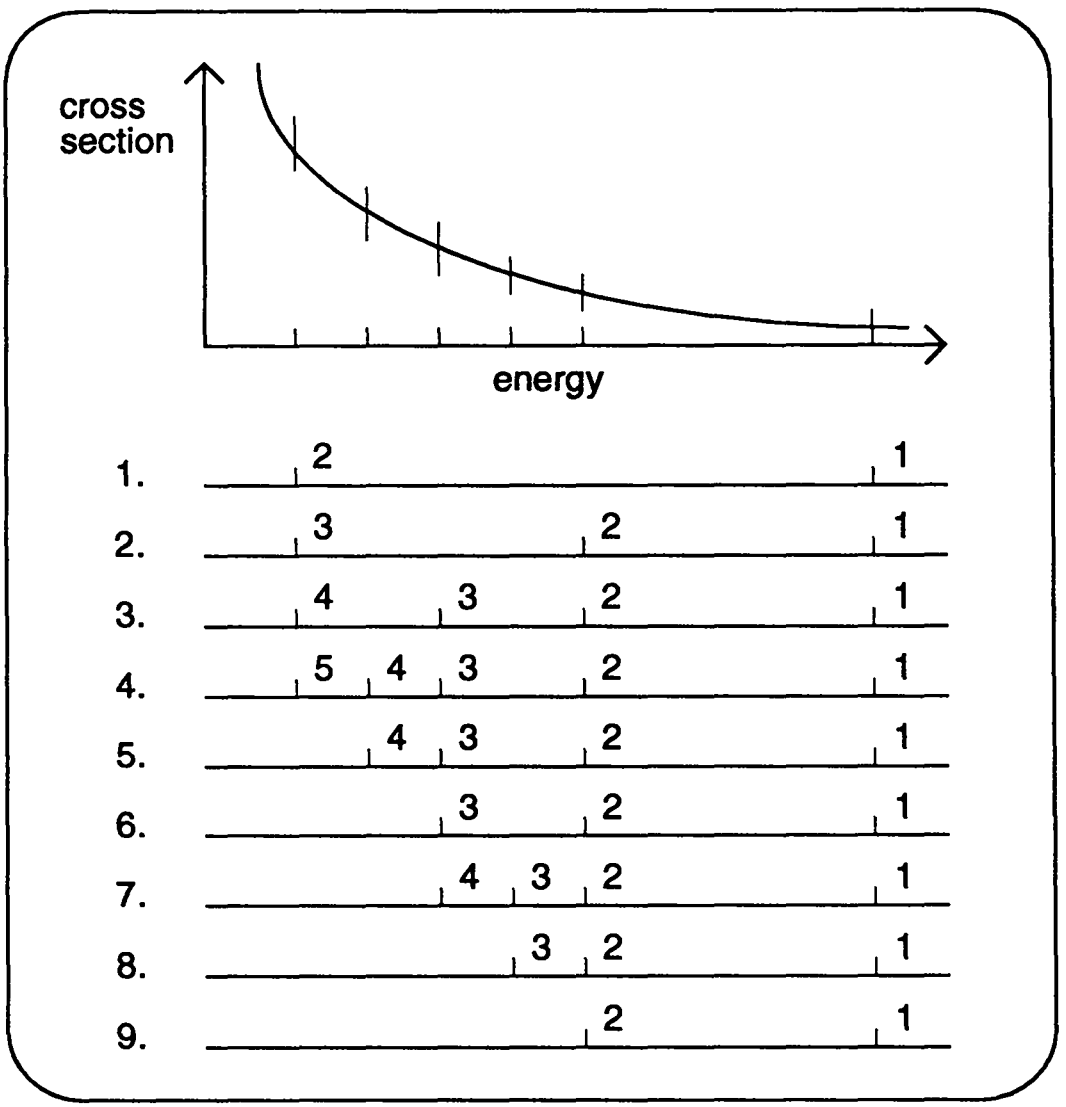


Figure 2: Inverted-stack method used in RECONR and several other places in NJOY. Line 1 shows the two initial points (the lower energy is higher in the stack). In line 2, a new point has been calculated at the midpoint, but the result was not converged, and the new point has been inserted in the stack. In line 3, the midpoint of the top panel has been checked again, found to be not converged, and inserted into the stack. The same thing happens in line 4. In line 5, the top panel is found to be converged, and the top point (5) has been written out. The same thing happens in line 6. In line 7, the top panel is tested and found to be not converged. The midpoint is added to the stack. Finally, in line 8, the top panel is found to be converged, and the top point is written out. This leaves two points in the stack (see line 9). Note that the energy points come off the stack in the desired order of increasing energy, and that only one point has to be moved up in the stack as each new result is inserted.

to be the energy where the slope of the actual interpolation function equals the slope of the linear approximation. Formulas are provided for each of the nonlinear ENDF/B interpolation laws: σ linear in $\ln(E)$; $\ln(\sigma)$ linear in E; and $\ln(\sigma)$ linear in $\ln(E)$.

The convergence criterion used for linearization is that the linearized cross section at the intermediate point is within the fractional tolerance **ERR** (or a small absolute value **ERRLIM**) of the actual cross section specified by the ENDF law. More complicated criteria are used for resonance reconstruction.

There are two basic problems that arise if a simple fractional tolerance test is used to control resonance reconstruction. First, as points are added to the energy grid, adjacent energy values may become so close that they will be rounded to the same number when a formatted output file is produced or when the machine-dependent limit for decimal single-precision accuracy is reached. It clearly makes no sense to continue to add grid points after this limit is reached. Through the use of dynamic format reconstruction, the energy resolution available for formatted NJOY output is 7 significant figures (that is, $\pm 1.234567 \pm n$) rather than the usual 5 or 6 (see Section II.D). On "short-word" machines (32–36 bits per word), the limit set by precision is also about 7 significant figures. On "long-word" machines (typically 60–64 bits per word), binary output files can be used, and NJOY can produce up to 15 significant figures if necessary.

Significant-figure control is implemented as follows: each intermediate energy is truncated to **NDIGIT** significant figures before the corresponding cross sections are computed, and if the resulting number is equal to either of the adjacent values, the interval is declared to be converged. Thus, no identical energies are produced, but an unpredictable loss in accuracy results. The error in the area of this interval is certainly less than $0.5 \times \Delta\sigma \times \Delta E$, so this value is added to an error estimate and a count of panels truncated by the significant figure check is incremented for a later informative diagnostic message.

The second basic problem alluded to above is that a very large number of resonance grid points arise from straightforward linear reconstruction of the resonance cross section of some isotopes. Many of these points come from narrow, weak, high-energy resonances, which do not need to be treated accurately in many applications. As an example, the capture and fission resonance integrals important for thermal reactors must be computed with a 1/E flux weighting. If the resonance reconstruction tolerance is set high (say 1%) to reduce the cost of processing, the resonance integrals will be computed to only 1% accuracy. However, if the reconstruction tolerance were set to a smaller value, like 0.1%, and if the high-energy

resonances (whose importance is reduced by the 1/E weight and the 1/v trend of the capture and fission cross sections) were treated with less accuracy than the low-energy resonances, then it is likely that one could achieve an accuracy much better than 1% with an overall reduction in the number of points (hence computing cost). Since 1/E weighting is not realistic in all applications (for example, in fast reactors), user control of this "thinning" operation must be provided.

Based on these arguments, the following approach was chosen to control the problem of very large files. First, panels are subdivided until the elastic, capture, and fission cross sections are converged to within **ERRMAX**, where **ERRMAX** $\geq$ **ERR**. These two tolerances are normally chosen to form a reasonable band, such as 10% and 0.5%, to ensure that all resonances are treated at least roughly (for example, for plotting). If the resonance integral (1/E weight) in some panel is large, the panel is further subdivided to achieve an accuracy of **ERR** (say 0.5%). However, if the contribution to the resonance integral from any one interval gets small, the interval will be declared converged, and the local value of the cross section will end up with some intermediate accuracy. Once again, the contribution to the error in the resonance integral should be less than $0.5 \times \Delta\sigma \times \Delta E$. This value is added into an accumulating estimate of the error, and a count of panels truncated by the resonance integral check is incremented.

The problem with this test is that RECONR does not know the value of the resonance integral in advance, so the tolerance parameter **ERRINT** is not the actual allowed fractional error in the integral. Instead, it is more like the resonance integral error per grid point (barns/point). Thus, a choice of **ERRINT=ERR/10000** with **ERR**=0.001 would limit the integral error to about 0.001 barn if 10000 points resulted from reconstruction. Since important resonance integrals vary from a few barns to a few hundred barns, this is a reasonable choice. The integral check can be suppressed by setting **ERRINT** very small or **ERRMAX=ERR**.

When resonance reconstruction is complete, RECONR provides a summary of the possible resonance integral error due to significant figure reduction and the integral check over several coarse energy bands. A typical example follows:

```
                ESTIMATED MAXIMUM ERROR DUE TO
                RESONANCE INTEGRAL CHECK (ERRMAX, ERRINT)
                AND SIGNIFICANT FIGURE TRUNCATION (NDIGIT)

  UPPER       ELASTIC     PERCENT ERROR     CAPTURE     PERCENT ERROR
  ENERGY      INTEGRAL   RES-INT SIG-FIG    INTEGRAL   RES-INT SIG-FIG
1.55E+02
4.96E+02     7.45E+00     .000   0.000     2.82E-02     .009   0.000
1.63E+03     5.94E+00     .000   0.000     1.71E-01     .002   0.000
5.20E+03     4.12E+00     .000   0.000     6.80E-03     .149   0.000
```

```
1.73E+04    6.49E+00    .000    .000    1.61E-02    .134    .008
5.62E+04    9.66E+00    .001    .000    1.74E-02    .200    .097
1.78E+05    4.20E+00    .004    .009    1.19E-02    .216   4.284
4.00E+05    3.50E+00    .008    .007    5.64E-03    .257   5.265

POINTS ADDED BY RESONANCE RECONSTRUCTION = 12309
POINTS AFFECTED BY RESONANCE INTEGRAL CHECK =   6969
POINTS AFFECTED BY SIGNIFICANT FIGURE REDUCTION =  1262
POINTS REMOVED BY BACKTHINNING =   201
FINAL NUMBER OF RESONANCE POINTS = 12749
```

The last band covers the unresolved range, if present. The parameter **NDIGIT** and the parameters **ERRMAX** and **ERRINT**, taken together, should be considered as adjustment "knobs" that can increase or decrease the errors in their respective columns to get an appropriate balance of accuracy and economy for a particular application.

D. Resonance Representations

RECONR uses the resonance formulas as implemented in the original RESEND code[2] with four changes: a more efficient calculation of multilevel Breit-Wigner cross sections developed by C. Lubitz of the Knolls Atomic Power Laboratory (General Electric Co.) and coded by P. Rose of the National Nuclear Data Center (NNDC) at Brookhaven National Laboratory, the addition of competitive widths introduced for ENDF/B-V, a $\psi\chi$ Doppler-broadening calculation for single-level Breit-Wigner and Adler-Adler resonance shapes, and a capability to process either the multilevel multichannel R-matrix Reich-Moore parameters or the multilevel single-channel Hybrid R-Function parameters based on the work of M. Bhat and C. Dunford of the NNDC. An expanded discussion of the following formulas can be found in the ENDF/B-V and ENDF-6 format manuals.[3, 4]

1. Single-Level Breit-Wigner Representation (SLBW) The subroutine that computes single-level Breit-Wigner cross sections (**CSSLBW**) uses

$$\begin{aligned}\sigma_n &= \sigma_p \\ &\quad + \sum_\ell \sum_r \sigma_{mr} \Big\{ \Big[\cos 2\phi_\ell - (1 - \frac{\Gamma_{nr}}{\Gamma_r}) \Big] \psi(\theta, x) \\ &\quad + \sin 2\phi_\ell \, \chi(\theta, x) \Big\} , \qquad (2) \\ \sigma_f &= \sum_\ell \sum_r \sigma_{mr} \frac{\Gamma_{fr}}{\Gamma_r} \psi(\theta, x) , \qquad (3)\end{aligned}$$

$$\sigma_\gamma = \sum_\ell \sum_r \sigma_{mr} \frac{\Gamma_{\gamma r}}{\Gamma_r} \psi(\theta, x) \text{ , and} \tag{4}$$

$$\sigma_p = \sum_\ell \frac{4\pi}{k^2}(2\ell+1)\sin^2\theta_\ell \text{ ,} \tag{5}$$

where σ_n, σ_f, σ_γ, and σ_p are the neutron (elastic), fission, radiative capture, and potential scattering components of the cross section arising from the given resonances. There can be "background" cross sections in File 3 that must be added to these values to account for competitive reactions such as inelastic scattering or to correct for the inadequacies of the single-level representation with regard to multilevel effects or missed resonances. The sums extend over all the ℓ values and all the resolved resonances r with a particular value of ℓ. Each resonance is characterized by its total, neutron, fission, and capture widths ($\Gamma, \Gamma_n, \Gamma_f, \Gamma_\gamma$), by its J value (AJ in the file), and by its maximum value (SMAX= σ_{mr}/Γ_r in the code)

$$\sigma_{mr} = \frac{4\pi}{k^2} g_J \frac{\Gamma_{nr}}{\Gamma_r} \text{ ,} \tag{6}$$

where g_J is the spin statistical factor

$$g_J = \frac{2J+1}{4I+2} \text{ ,} \tag{7}$$

and I is the total spin SPI given in File 2, and k is the neutron wave number, which depends on incident energy E and the atomic weight ratio to the neutron for the isotope A (AWRI in the file), as follows:

$$k = (2.196771\times10^{-3})\frac{A}{A+1}\sqrt{E} \text{ .} \tag{8}$$

There are two different characteristic lengths that appear in the ENDF resonance formulas: first, there is the "scattering radius" $\hat{a}$, which is given directly in File 2 as AP; and second, there is the "channel radius" a, which is given by

$$a = 0.123\, A^{1/3} + 0.08 \text{ .} \tag{9}$$

If the File 2 parameter NAPS is equal to one, a is set equal to $\hat{a}$ in calculating penetrabilities and shift factors (see below). The ENDF-6 option to enter an energy-dependent scattering radius is not supported. The neutron width in the equations for the SLBW cross sections is energy dependent due to the penetration factors P_ℓ; that is,

$$\Gamma_{nr}(E) = \frac{P_\ell(E)\,\Gamma_{nr}}{P_\ell(|E_r|)} \text{ ,} \tag{10}$$

where

$$P_0 = \rho \,, \tag{11}$$

$$P_1 = \frac{\rho^3}{1+\rho^2} \,, \tag{12}$$

$$P_2 = \frac{\rho^5}{9+3\rho^2+\rho^4} \,, \tag{13}$$

$$P_3 = \frac{\rho^7}{225+45\rho^2+6\rho^4+\rho^6} \,, \text{ and} \tag{14}$$

$$P_4 = \frac{\rho^9}{11025+1575\rho^2+135\rho^4+10\rho^6+\rho^8} \,, \tag{15}$$

where E_r is the resonance energy and ρ=ka depends on the channel radius or the scattering radius as specified by NAPS. The phase shifts are given by

$$\phi_0 = \hat{\rho} \,, \tag{16}$$

$$\phi_1 = \hat{\rho} - \tan^{-1}\hat{\rho} \,, \tag{17}$$

$$\phi_2 = \hat{\rho} - \tan^{-1}\frac{3\hat{\rho}}{3-\hat{\rho}^2} \,, \tag{18}$$

$$\phi_3 = \hat{\rho} - \tan^{-1}\frac{15\hat{\rho}-\hat{\rho}^2}{15-6\hat{\rho}^2} \,, \text{ and} \tag{19}$$

$$\phi_4 = \hat{\rho} - \tan^{-1}\frac{105\hat{\rho}-10\hat{\rho}^3}{105-45\hat{\rho}^2+\hat{\rho}^4} \,, \tag{20}$$

where $\hat{\rho}$=$k\hat{a}$ depends on the scattering radius. The final components of the cross section are the actual line shape functions ψ and χ. At zero temperature,

$$\psi = \frac{1}{1+x^2} \,, \tag{21}$$

$$\chi = \frac{x}{1+x^2} \,, \tag{22}$$

$$x = \frac{2(E-E_r')}{\Gamma_r} \,, \tag{23}$$

and

$$E_r' = E_r + \frac{S_\ell(|E_r|)-S_\ell(E)}{2(P_\ell(|E_r|)}\,\Gamma_{nr}(|E_r|) \,, \tag{24}$$

in terms of the shift factors

$$S_0 = 0 \,, \tag{25}$$

$$S_1 = -\frac{1}{1+\rho^2} \,, \tag{26}$$

$$S_2 = -\frac{18+3\rho^2}{9+3\rho^2+\rho^4} \,, \tag{27}$$

$$S_3 = -\frac{675+90\rho^2+6\rho^4}{225+45\rho^2+6\rho^4+\rho^6}\ , \text{ and} \tag{28}$$

$$S_4 = -\frac{44100+4725\rho^2+270\rho^4+10\rho^6}{11025+1575\rho^2+135\rho^4+10\rho^6+\rho^8}\ . \tag{29}$$

To go to higher temperatures, define

$$\theta = \frac{\Gamma_r}{\sqrt{\frac{4kTE}{A}}}\ , \tag{30}$$

where k is the Boltzmann constant and T is the absolute temperature. The line shapes ψ and χ are now given by

$$\psi = \frac{\sqrt{\pi}}{2}\,\theta\,\mathrm{Re}W(\frac{\theta x}{2},\frac{\theta}{2})\ , \tag{31}$$

and

$$\chi = \frac{\sqrt{\pi}}{2}\,\theta\,\mathrm{Im}W(\frac{\theta x}{2},\frac{\theta}{2})\ , \tag{32}$$

in terms of the complex probability function (see **QUICKW**, **WTAB**, and **W**, which came from the MC^2 code[5])

$$W(x,y) = \mathrm{e}^{-z^2}\,\mathrm{erfc}(-iz) = \frac{i}{\pi}\int_{-\infty}^{\infty}\frac{\mathrm{e}^{-t^2}}{z-t}\,dt\ , \tag{33}$$

where z=x+iy. The $\psi\chi$ method is not as accurate as kernel broadening (see BROADR) because the backgrounds (which are sometimes quite complex) are not broadened, and terms important for energies less than about $16kT/A$ are neglected; However, the $\psi\chi$ method is less expensive than BROADR. The current version of RECONR includes Doppler-broadening for the single-level Breit-Wigner and Adler-Adler representations only.

2. Multilevel Breit-Wigner Representation (MLBW) The Lubitz-Rose method used for calculating multilevel Breit-Wigner cross sections (**CSMLBW**) is formulated as follows:

$$\sigma_n(E) = \frac{\pi}{k^2}\sum_{\ell}\sum_{s=|I-\frac{1}{2}|}^{I+\frac{1}{2}}\sum_{J=|l-s|}^{l+s} g_J\,|1-U_{nn}^{\ell sJ}(E)|^2\ , \tag{34}$$

with

$$U_{nn}^{\ell J}(E) = \mathrm{e}^{2i\phi_\ell} - \sum_r \frac{i\Gamma_{nr}}{E_r'-E-i\Gamma_r/2}\ , \tag{35}$$

where the other symbols are the same as those used above. Expanding the complex operations gives

$$\sigma_n(E) = \frac{\pi}{k^2}\sum_{\ell}\sum_{s=|I-\frac{1}{2}|}^{I+\frac{1}{2}}\sum_{J=|l-s|}^{l+s} g_J\left\{\left(1-\cos 2\phi_\ell - \sum_r \frac{\Gamma_{nr}}{\Gamma_r}\frac{2}{1+x_r^2}\right)^2 + \left(\sin 2\phi_\ell + \sum_r \frac{\Gamma_{nr}}{\Gamma_r}\frac{2x_r}{1+x_r^2}\right)^2\right\}, \quad (36)$$

where the sums over r are limited to resonances in spin sequence ℓ that have the specified value of s and J. Unfortunately, the s dependence of Γ is not known. The file contains only $\Gamma_J=\Gamma_{s_1 J}+\Gamma_{s_2 J}$. It is assumed that the Γ_J can be used for one of the two values of s, and zero is used for the other. Of course, it is important to include both channel-spin terms in the potential scattering. Therefore, the equation is written in the following form:

$$\sigma_n(E) = \frac{\pi}{k^2}\sum_{\ell}\left[\sum_J g_J\left\{\left(1-\cos 2\phi_\ell - \sum_r \frac{\Gamma_{nr}}{\Gamma_r}\frac{2}{1+x_r^2}\right)^2 + \left(\sin 2\phi_\ell + \sum_r \frac{\Gamma_{nr}}{\Gamma_r}\frac{2x_r}{1+x_r^2}\right)^2\right\} + 2D_\ell(1-\cos 2\phi_\ell)\right], \quad (37)$$

where the summation over J now runs from

$$||I-\ell|-\frac{1}{2}| \rightarrow I+\ell+\frac{1}{2}, \quad (38)$$

and D_ℓ gives the additional contribution to the statistical weight resulting from duplicate J values not included in the new J sum; namely,

$$D_\ell = \sum_{s=|I-\frac{1}{2}|}^{I+\frac{1}{2}}\sum_{J=|l-s|}^{l+s} g_J - \sum_{J=||I-\ell|-\frac{1}{2}|}^{I+\ell+\frac{1}{2}} g_j \quad (39)$$

$$= (2\ell+1) - \sum_{J=||I-\ell|-\frac{1}{2}|}^{I+\ell+\frac{1}{2}} g_j . \quad (40)$$

A case where this correction would appear is the ℓ=1 term for a spin-1 nuclide. There will be 5 J values: 1/2, 3/2, and 5/2 for channel spin 3/2; and 1/2 and 3/2 for channel spin 1/2. All five contribute to the potential scattering, but the file will only include resonances for the first three.

The fission and capture cross sections are the same as for the single-level option. The $\psi\chi$ Doppler-broadening cannot be used with this formulation of the MLBW representation.

3. Adler-Adler Representation (AA) The multilevel Adler-Adler representation is defined for ℓ=0 only. It is useful for fissionable materials. The total cross sections are given by

$$\begin{aligned}\sigma_t(E) &= \frac{4\pi}{k^2}\sin^2\phi_0 \\ &+ \frac{\pi\sqrt{E}}{k^2}\left\{\sum_r \frac{1}{\nu_r}\left[(G_r\cos 2\phi_0 + H_r\sin 2\phi_0)\,\psi(\theta,x)\right.\right. \\ &+ \left.\left.(H_r\cos 2\phi_0 - G_r\sin 2\phi_0)\,\chi(\theta,x)\right]\right. \\ &+ \left. A_1 + \frac{a_2}{E} + \frac{A_3}{E^2} + \frac{A_4}{E^3} + B_1E + B_2E^2\right\}, \end{aligned} \tag{41}$$

where

$$x = \frac{\mu_r - E}{\nu_r}, \tag{42}$$

and where ν_r is the resonance half-width (corresponds to $\Gamma/2$ in the Breit-Wigner notation), μ_r is the resonance energy, G_r is the symmetric total parameter, H_r is the asymmetric total parameter, and the A_i and B_i are coefficients of the total background correction.

The fission and capture cross section both use the form

$$\begin{aligned}\sigma_x(E) &= \frac{\pi\sqrt{E}}{k^2}\left\{\sum_r \frac{1}{\nu_r}\left[G_r\psi(\theta,x) + H_r\chi(\theta,x)\right]\right. \\ &+ \left. A_1 + \frac{A_2}{E} + \frac{A_3}{E^2} + \frac{A_4}{E^4} + B_1E + B_2E^2\right\}, \end{aligned} \tag{43}$$

where the values of G, H, A_i, and B_i appropriate for the desired reaction are used.

Doppler-broadening can be applied as for the SLBW case, except note that Γ_r in Eq. (24) must be replaced with $2\nu_r$. Doppler-broadened Adler-Adler cross sections are more accurate than SLBW cross sections because the background is smoother. However, cross sections below about $16kT/A$ will still be inaccurate.

4. Reich-Moore Representation (RM) The Reich-Moore representation is a multilevel formulation with two fission channels; hence, it is useful for both structural and fissionable materials. The cross sections are given by

$$\sigma_t = \frac{2\pi}{k^2}\sum_\ell\sum_J g_J\left\{(1 - \mathrm{Re}\,U_{nm}^{\ell J}) + 2d_{\ell J}[1 - \cos(2\phi_\ell)]\right\}, \tag{44}$$

$$\sigma_n = \frac{\pi}{k^2}\sum_\ell\sum_J g_J\left\{|1 - U_{nn}^{\ell J}|^2 + 2d_{\ell J}[1 - \cos(2\phi_\ell)]\right\}, \tag{45}$$

$$\sigma_f = \frac{4\pi}{k^2}\sum_{\ell}\sum_{J} g_J \sum_{c} |\mathcal{I}_{nc}^{\ell J}|^2 \text{ , and} \tag{46}$$

$$\sigma_\gamma = \sigma_t - \sigma_n - \sigma_f \text{ ,} \tag{47}$$

where $\mathcal{I}_{nc}$ is an element of the inverse of the complex R-matrix and

$$U_{nn}^{\ell J} = e^{2i\phi_\ell}\left[2\mathcal{I}_{nn} - 1\right] . \tag{48}$$

The elements of the R-matrix are given by

$$R_{nc}^{\ell J} = \delta_{nc} - \frac{i}{2}\sum_{r}\frac{\Gamma_{nr}^{1/2}\Gamma_{cr}^{1/2}}{E_r - E - \frac{i}{2}\Gamma_{\gamma r}} . \tag{49}$$

In these equations, "c" stands for the fission channel, "r" indexes the resonances belonging to spin sequence (ℓ, J), and the other symbols have the same meanings as for SLBW or MLBW. Of course, when fission is not present, σ_f can be ignored. The R-matrix reduces to an R-function, and the matrix inversion normally required to get $\mathcal{I}_{nn}$ reduces to a simple inversion of a complex number.

As in the MLBW case, the summation over J runs from

$$||I - \ell| - \frac{1}{2}| \rightarrow I + \ell + \frac{1}{2} . \tag{50}$$

The term $d_{\ell J}$ in the expressions for the total and elastic cross sections is used to account for the possibility of an additional contribution to the potential scattering cross section from the second channel spin. It is unity if there is a second J value equal to J, and zero otherwise. This is just a slightly different approach for making the correction discussed in connection with Eq. (40). Returning to the I=1, ℓ=1 example given above, d will be one for J=1/2 and J=3/2, and it will be zero for J=5/2.

ENDF-6 Reich-Moore evaluations can contain a parameter LAD that indicates that these parameters can be used to compute an angular distribution for elastic scattering if desired (an approximate angular distribution is still given in File 4 for these cases). The current version of RECONR does not compute angular distributions.

5. Hybrid R-Function Representation (HRF) The Hybrid R-Function representation, treats elastic scattering as a multilevel cross section using formulas similar to those given above for the Reich-Moore format in the case where fission

is absent. The other reactions are treated with formulas similar to those of the SLBW method. The main use for this format is to provide a better representation of competitive reactions than is provided by any of the other formats described above. This treatment can include a background R-function, tabulated charged-particle penetrabilities, and optical model phase shifts. Following the Reich-Moore notation, the elastic cross section is given by

$$\sigma_n = \frac{\pi}{k^2} \sum_{\ell} \sum_{s=|I-\frac{1}{2}|}^{I+\frac{1}{2}} \sum_{J=|l-s|}^{l+s} g_J |1 - U_{nn}^{\ell s J}|^2 \ , \tag{51}$$

where the U function is given by the scalar version of Eq. (48):

$$U_{nn}^{\ell s J} = \mathrm{e}^{2i\phi_\ell} \left[\frac{2}{R_{nn}^{\ell s J}} - 1 \right] . \tag{52}$$

The R function itself is given by

$$R_{nn}^{\ell s J} = 1 - \frac{i}{2} \sum_{r} \frac{\Gamma_{nr}}{E_r - E - \frac{i}{2}\Gamma_{\gamma r}} - i\, P_{\ell s J} R_{\ell s J}^0 \ , \tag{53}$$

where $R_{\ell s J}^0$ is a (complex) background R function and $P_{\ell s J}$ is a penetrability factor. The background R function can either be read in or set to zero. The penetrability and shift factors are computed from the scattering radius or channel radius as for SLBW. The phase shifts $\phi_{\ell s J}$ can be computed from the scattering radius as before, or the (complex) phase shifts can be read in from an optical model calculation.

Note that resonance parameters are given explicitly for all three quantum numbers ℓ, s, and J. No correction to the potential scattering cross section from repeated J values is needed.

Elastic angular distributions can also be computed from HRF parameters if the LAD parameter is set; however, RECONR does not yet support the calculation of angular distributions. Infinitely dilute cross sections in the unresolved-energy range are computed in `CSUNR1` or `CSUNR2` using average resonance parameters and probability distributions from File 2. With the approximations used, these cross sections are not temperature dependent; therefore, the results are a good match to resolved resonance data generated using `TEMPR>0`. The formulas used are based on the single-level approximation with interference.

$$\sigma_n(E) \ = \ \sigma_p + \frac{2\pi^2}{k^2} \sum_{\ell,J} \frac{g_J}{\overline{D}} [\overline{\Gamma}_n^2 R_n - 2\overline{\Gamma}_n \sin^2 \phi_\ell] \ , \tag{54}$$

$$\sigma_x(E) = \frac{2\pi^2}{k^2}\sum_{\ell,J}\frac{g_J}{\overline{D}}\overline{\Gamma}_n\overline{\Gamma}_x R_x \text{ , and} \tag{55}$$

$$\sigma_p = \frac{4\pi}{k^2}\sum_{\ell}(2\ell+1)\sin^2\phi_\ell \text{ ,} \tag{56}$$

where x stands for either fission or capture, $\overline{\Gamma}_i$ and $\overline{D}$ are the appropriate average widths and spacing for the ℓ,J spin sequence, and R_i is the fluctuation integral for the reaction and sequence (see **GNRL**). These integrals are simply the averages taken over the chi-square distributions specified in the file; for example,

$$\overline{\Gamma}_n\overline{\Gamma}_f R_i = \left\langle\frac{\Gamma_n\Gamma_f}{\Gamma}\right\rangle$$

$$= \int dx_n P_\mu(x_n)\int dx_f P_\nu(x_f)\int dx_c P_\lambda(x_c) \tag{57}$$

$$\times \frac{\Gamma_n(x_n)\,\Gamma_f(x_f)}{\Gamma_n(x_n)+\Gamma_f(x_f)+\Gamma_\gamma+\Gamma_c(x_c)} \text{ ,} \tag{58}$$

where $P_\mu(x)$ is the chi-square distribution for μ degrees of freedom. The integrals are evaluated with the quadrature scheme developed by R. Hwang for the MC^2-2 code[6] giving

$$R_f = \sum_i W_i^\mu \sum_j W_j^\nu \sum_k W_k^\lambda \frac{Q_i^\mu Q_j^\nu}{\overline{\Gamma}_n Q_i^\mu + \overline{\Gamma}_f Q_j^\nu + \Gamma_\gamma + \overline{\Gamma}_c Q_k^\lambda} \text{ .} \tag{59}$$

The W_i^μ and Q_i^μ are the appropriate quadrature weights and values for μ degrees of freedom, and Γ_γ is assumed to be constant (many degrees of freedom). The competitive width $\overline{\Gamma}_c$ is assumed to affect the fluctuations, but a corresponding cross section is not computed. The entire competitive cross section is supposed to be in the File 3 total cross section as a smooth background.

It should be noted that the reduced average neutron width (AMUN) is given in the file, and

$$\overline{\Gamma}_n = \Gamma_n^0\sqrt{E}\,V_\ell(E) \text{ ,} \tag{60}$$

where the penetrabilities for the unresolved region are defined as

$$V_0 = 1 \text{ ,} \tag{61}$$

$$V_1 = \frac{\rho^2}{1+\rho^2} \text{ , and} \tag{62}$$

$$V_2 = \frac{\rho^4}{\rho+3\rho^2+\rho^4} \text{ .} \tag{63}$$

Other parameters are defined as for SLBW.

Unresolved parameters can be given as independent of energy, with only fission widths dependent on energy, or as fully energy dependent. The first two options are processed in CSUNR1, and the last one is processed in CSUNR2.

E. Code Description

The flow of this module is controlled by the RECONR routine. The first step is to read cards 1, 2, and 3 of the user's input. The TAPEID record of the input tape (NENDF) is read and printed, then the new TAPEID record is written to the output tape (NPEND). RECONR is now ready to enter the loop over the desired materials.

For each material, STORAG is used to allocate space for the energy nodes and for scratch storage (ENODE, SCR), and RUIN is called to read cards 4 through 7 of the user's input. Subroutine RUIN automatically adds the ENDF energy limits of 1E-5 eV and 20 MeV and the thermal energy 0.0253 eV to any energy grid points entered by the user. If the reconstruction temperature (TEMPR) is greater than zero, a table of ψ and χ functions is generated (the W table; see WTAB and QUICKW). The FINDF utility subroutine is then used to find the first card of File 1 (MF=1,MT=451) for the desired material.

File 1 on the input ENDF tape is examined to obtain certain constants and flags and to analyze the directory (ANLYZD). Subroutine ANLYZD determines which reactions should be considered "redundant"; that is, the reactions that are sums of other reactions and will be included on the output PENDF tape. The total cross section (MT=1 for neutrons, MT=501 for photons) will always be included; the nonelastic cross section (MT=3) will be included if it is needed for photon production (that is, MF=12, MT=3 is found); the inelastic cross section (MT=4) will be included if sections with MT in the range 51–91 occur in the file, and the total fission reaction (MT=18) will be called redundant if the partial fission representation (MT=19, 20, 21, 38) is found. Space for the new material directory is then reserved (MFS, MTS, NCS). Section identification and card counts will be entered into these arrays as they are determined.

The next step is to read File 2, which contains resolved and unresolved resonance parameters (if any). The array RES is assigned to contain the File 2 data and RDFIL2 is called to read them. While the resonance parameters are being stored, RECONR adds each resonance energy to its list of energy nodes (ENODE). In the unresolved energy range, RECONR uses the energies of tabulated parameters or fission widths if available. If the evaluation uses energy-independent parameters, or if the energy steps between the nodes are too large, RDFIL2 creates additional node energies at a density of approximately 10 points per decade. Note that regions where the unresolved representation for an element overlaps the resolved or smooth ranges are found and marked by negative energy values. The energy nodes are sorted into order and duplications are removed. When control is returned to RECONR, any unused space in the RES array is released to be made available for other uses.

If unresolved data is present, subroutine GENUNR is called to compute the infinitely-dilute unresolved average cross sections on the unresolved energy grid using CSUNR1 or CSUNR2. Any backgrounds on File 3 are included, except in regions of resolved-unresolved or unresolved-smooth overlap. The computed cross sections are arranged in the order required by the special section with MF=2 and MT=152, which is written onto the PENDF tape by RECOUT. Using the normal ENDF style, this format is defined by the following:

```
[MAT,2,152/ZA,AWR,LSSF,0,0,INTUNR]HEAD
[MAT,2,152/0.,0.,5,1,NW,NUNR/
          E1,STOT1,SELAS1,SFIS1,SCAP1,STRN1,
          E2,STOT2,SELAS2,SFIS2,SCAP2,STRN2,
          ...
          ENUNR,STOTNUNR,....]LIST
```

where NW=6+6*NUNR. The definitions of the energy and cross section entries are fairly obvious, except STRN stands for the current-weighted total cross section. This format is specialized to "infinite dilution." The more general form used for self-shielded effective cross sections will be described in the UNRESR and GROUPR chapters of this manual.

The subroutine LUNION is used to linearize and unionize the ENDF data. Space is reserved for two buffers to be used by LOADA/FINDA and for the linearization stack (X and Y). The length of the stack (NDIM) determines the smallest possible subdivision of each panel (energy points as close as 2^{-NDIM} times the panel width can be generated). Since the number of energies in the union grid may soon exceed the capacity of any reasonable memory array, the existing list of energy nodes is copied to binary scratch storage (the LOADA/FINDA). This storage system

consists of the buffers BOLD and BNEW and the scratch units IOLD and INEW. The energy grid points will "ping-pong" back and forth between units 14 and 15 as the union grid is built up. Subroutine LUNION now starts with MT=2 and checks each reaction in sequence to determine whether the current grid (on IOLD) is sufficient to represent the reaction to within the desired tolerance using linear interpolation. If not, RECONR uses ISLIN1 to select the optimum points to be added to the new grid (on INEW). The units INEW and IOLD are swapped, and the next MT is processed. When all nonredundant reactions have been examined, the list of energies in LOADA/FINDA storage is the desired linearized and unionized grid. The storage used is released.

This grid is used as the starting point for resonance reconstruction in RESXS. Subroutine RESXS first reserves space for the LOADA/FINDA buffers BUFR and BUFG, the linearization stack (X and Y), and the partial cross sections (SIG). The length of the stack (NDIM) determines the smallest possible subdivision of a panel between two nodes (energy points as close a $2^{-\text{NDIM}}$ times the panel width can be generated). Subroutine RESXS then examines the grid on NGRID (IOLD from LUNION) panel by panel. Grid points are added and cross sections computed until the convergence criteria discussed in Section C are satisfied. The cross sections are copied to NOUT using LOADA, and RESXS continues to the next panel. This procedure is continued until all panels are converged. The result is a tape (NOUT) containing the energy grid in the resonance region and the total, elastic, fission, and capture cross sections at each energy point.

Unionization is obtained automatically in the resonance region since the three partials are computed simultaneously in SIGMA. This routine calls CSNORP if there are no resonance parameters, CSSLBW for single-level Breit-Wigner parameters, CSMLBW for multilevel Breit-Wigner parameters, CSAA for multilevel Adler-Adler parameters, CSRMAT for Reich-Moore parameters, CSHYB for Hybrid R-Function parameters, and SIGUNR for unresolved resonance parameters. This last routine retrieves the cross sections from the table prepared by GENUNR. A special feature of RECONR is the ability to reconstruct the cross sections at TEMPR by $\psi\chi$ broadening if single-level Breit-Wigner (SLBW) or Adler-Adler (AA) parameters are given. The Doppler-broadened resonance shapes are obtained using QUICKW (see description in UNRESR) in CSSLBW or CSAA and the linearization procedure proceeds as before.

The resonance cross sections on NGRID are merged with the ENDF cross sections in EMERGE. First, the background grid from LUNION is merged with the resonance grid from RESXS and written onto the LOADA/FINDA file, which will accumulate

the total cross section and any other redundant reactions required (IOLD/INEW). A loop is then set up over all nonredundant reactions. For each grid point, the ENDF background cross section is obtained by interpolation. If this grid point has a resonance contribution on NRES, it is added. The resulting net cross section at this point is added into the appropriate redundant cross sections on IOLD/INEW and also saved on NGRID. When all the energies for this reaction have been processed, the cross sections on NGRID are converted into a TAB1 record and written on NSCR. This loop is continued until all reactions have been processed. When EMERGE is finished, NSCR contains cross sections for all the nonredundant reactions, and IOLD contains the redundant summation reactions.

Control now passes to RECOUT, which writes the new File 1 comments and dictionary. It also writes a default version of the section with MF=2 and MT=151 that gives no resonance parameters. The upper limit of the resolved energy range, ERESH, is added to the "C2" field of the third card so that BROADR knows not to broaden into the unresolved energy range. For materials with unresolved data, a specially formatted section (MF=2, MT=152) is written containing the infinitely-dilute unresolved cross sections. This section can be used by BROADR and GROUPR to correct for resolved-unresolved overlap effects, if necessary. Subroutine RECOUT then steps through the reactions on NSCR and IOLD. Redundant reactions are converted to TAB1 records and inserted in the correct order. Nonredundant reactions are simply copied. Finally, a MEND record is added and control is returned to RECONR.

Now RECONR either directs that this process be repeated for another isotope or writes a TEND record and terminates. The result is a new tape in ENDF format containing the desired pointwise cross sections. Note that only Files 1, 2, 3, 10, and 13 are included for neutron tapes. Only Files 1 and 23 are included for photon tapes.

F. Input Instructions

The input instructions for each module are given in the code as comment cards at the beginning of the source code for each module. The RECONR instructions are reproduced here for the convenience of the reader.

```
*---INPUT SPECIFICATIONS (FREE FORMAT)-------------------------*
*                                                             *
* CARD 1                                                      *
*    NENDF    UNIT FOR ENDF/B TAPE                           *
*    NPEND    UNIT FOR PENDF TAPE                            *
* CARD 2                                                      *
*    LABEL    66 CHARACTER LABEL FOR NEW PENDF TAPE          *
```

```
*            DELIMITED WITH *, ENDED WITH /.                    *
* CARD 3                                                         *
*    MAT      MATERIAL TO BE RECONSTRUCTED                       *
*    NCARDS   NUMBER OF CARDS OF DESCRIPTIVE DATA FOR NEW MF1    *
*             (DEFAULT=0.)                                       *
*    NGRID    NUMBER OF USER ENERGY GRID POINTS TO BE ADDED.     *
*             (DEFAULT=0.)                                       *
* CARD 4                                                         *
*    ERR      FRACTIONAL RECONSTRUCTION TOLERANCE USED WHEN      *
*             RESONANCE-INTEGRAL ERROR CRITERION (SEE ERRINT)    *
*             IS NOT SATISFIED.                                  *
*    TEMPR    RECONSTRUCTION TEMPERATURE (DEG KELVIN)            *
*             (DEFAULT=0.)                                       *
*    NDIGIT   NO. SIGNIFICANT DIGITS (DEFAULT=7)                 *
*    ERRMAX   FRACTIONAL RECONSTRUCTION TOLERANCE USED WHEN      *
*             RESONANCE-INTEGRAL ERROR CRITERION IS SATISFIED    *
*             (ERRMAX.GE.ERR, DEFAULT=20.*ERR)                   *
*    ERRINT   MAXIMUM RESONANCE-INTEGRAL ERROR (IN BARNS)        *
*             PER GRID POINT (DEFAULT=ERR/10000)                 *
*             (NOTE: THE MAX CROSS SECTION DIFFERENCE FOR        *
*              LINEARIZATION, ERRLIM, AND FOR RECONSTRUCTION,    *
*              ERRMIN, ARE ALSO TIED TO ERRINT.  TO GET MAXIMUM  *
*              ACCURACY, SET ERRINT TO A VERY SMALL NUMBER.      *
*              FOR ECONOMICAL PRODUCTION, USE THE DEFAULTS.)     *
* CARD 5                                                         *
*    CARDS    NCARDS OF DESCRIPTIVE COMMENTS FOR MT451           *
*             EACH CARD DELIMITED WITH *, ENDED WITH /.          *
* CARD 6                                                         *
*    ENODE    USERS ENERGY GRID POINTS                           *
*       CARDS 3, 4, 5, 6 MUST BE INPUT FOR EACH MATERIAL DESIRED *
*       MAT=0/ TERMINATES EXECUTION OF RECONR.                   *
*                                                                *
******************************************************************
```

A sample input for processing two isotopes from ENDF/B-IV Tape 407 follows (the line numbers are for reference only and are not part of the input):

```
 1.  0
 2.  4
 3.  RECONR
 4.  20 -21
 5.  *PENDF TAPE FOR U-235 AND PU239 FROM T407*/
 6.  1261 2/
 7.  .005/
 8.  *92-U-235 FROM T407*/
 9.  *PROCESSED WITH NJOY*/
10.  1264 2/
11.  .005/
12.  *94-PU-239 FROM T407*/
13.  *PROCESSED WITH NJOY*/
14.  0/
```

Card 4 tells RECONR that the input ENDF tape (Tape 407) will be on unit 20, and that the output PENDF tape will be on unit 21. Because of the minus sign, the PENDF tape will be in NJOY blocked-binary mode. The first MAT number (1261 for U-235) is given on Card 6, and the second one (1264 for Pu-239) is on card 10. In both cases, the reconstruction tolerance is 0.5% and the default integral thinning parameters are taken. Note the "0/" in line 14 that terminates the RECONR portion of the run. The resulting PENDF tape will contain the desired TAPEID card, followed by U235, a MEND card, PU-239, a MEND card, and a TEND card.

G. Error Messages

`ERROR IN RECONR***ILLEGAL NSUB FOR RECONR`

RECONR only processes sublibraries that contain cross section data. Check whether the right input ENDF input tape was mounted.

`ERROR IN RUINA***ILLEGAL NDIGIT`

Value must be between 1 and 15 on a long-word machine, or between 1 and 7 on a short-word machine.

`ERROR IN ANLYZD***TOO MANY REDUNDANT REACTIONS`

Increase the size of `MTR(10)` in `/RECON/` and increase `MTMAX=10` in `RECONR`.

`ERROR IN RDFILE2***STORAGE IN ENODE EXCEEDED`

Too many energy nodes including the user's nodes and the energies from MF=2. Increase `NODMAX` in `RECONR`.

`ERROR IN RDFIL2***STORAGE IN A EXCEEDED`

Too much resonance data. The main container array is too small. Increase `/RSTORE/` and `JX` in `RECONR` or decrease buffer sizes `NBUFG`, `NBUFR`, or `NBUF`.

`ERROR IN RDFIL2***STORAGE IN EUNR EXCEEDED.`

The limit `MAXUNR=200` has been exceeded.

`ERROR IN RDFIL2***MLBW FICTITIOUS J VALUES NOT ALLOWED.`

This reflects an error in the evaluation.

`ERROR IN RDFIL2***ENERGY-DEPENDENT SCATTERING RADIUS NOT CODED.`

This option is not coded, nor is it allowed in ENDF/B-VI. Other national files may have different procedures.

`MESSAGE FROM RDFIL2***CALCULATION OF ANGULAR DISTRIBUTION NOT...`

This option is not yet available in RECONR.

`ERROR IN RDFIL2***HYBRID COMPETING REACTIONS NOT YET ADDED`

This option is not yet available in RECONR.

ERROR IN LUNION***ILL BEHAVED THRESHOLD

The routine is having trouble adjusting the threshold to agree with the Q value. Check the points near the threshold for this evaluation.

ERROR IN LUNION***EXCEEDED STACK

Increase length of linearization stack NDIM (currently 20).

ERROR IN ISLIN1***X1=--- GE X2=---

The interpolation interval is bad.

ERROR IN ISLIN1***XM=--- LE X1=---

ERROR IN ISLIN1***XM=--- GE X2=---

Desired point is outside the interpolation range.

ERROR IN RESXS***STACK EXCEEDED

Increase length of reconstruction stack NDIM (currently 20).

ERROR IN SIGMA***GENERAL R-MATRIX NOT INSTALLED.

This option is not yet available in RECONR.

ERROR IN SIGMA***ILLEGAL OPTION.

There is a problem with the ENDF tape.

ERROR IN CSMLBW***NOT CODED FOR TEMPERATURE GT 0 DEG K

The $\psi\chi$ Doppler-broadening option is only coded for single-level Breit-Wigner and Adler-Adler resonance parameters. Use TEMPR=0. only.

ERROR IN CSRMAT***NOT CODED FOR TEMPERATURE GT 0 DEG K

The $\psi\chi$ Doppler-broadening option is only coded for single-level Breit-Wigner and Adler-Adler resonance parameters. Use TEMPR=0. only.

ERROR IN CSHYBR***DOPPLER BROADENING NOT PROVIDED FOR HYBRID

The $\psi\chi$ Doppler-broadening option is only coded for single-level Breit-Wigner and Adler-Adler resonance parameters. Use TEMPR=0. only.

ERROR IN CSAA***BAD LI VALUE

There is an error in the evaluation format.

MESSAGE FROM EMERGE---NEGATIVE ELASTIC CROSS SECTIONS FOUND

Negative elastic cross sections can occur for SLBW evaluations.

ERROR IN RECOUT***FOR MF --- MT ---

Indexing and pair count for this section do not make sense.

H. Input-Output Units

The following logical units are used:

10	NSCR1 in RECONR, NOUT in LUNION, and NIN in EMERGE. Contains copy of nonredundant sections from original ENDF tape.
11	NSCR2 in RECONR; NGRID in LUNION, RESXS, and EMERGE. Contains union grid for ENDF tape (not counting resonances).
12	NSCR3 in RECONR, NOUT in RESXS, and NRES in EMERGE. Contains resonance grid and cross sections.
13	NSCR4 in RECONR is used for two separate purposes. In RESXS it is a binary scratch file NSCR used for the unthinned resonance data. In EMERGE and RECOUT, it is NMERGE and contains the nonredundant reactions on the union grid.
14/15	IOLD/INEW in LUNION. Are used locally only to accumulate union grid for ENDF cross sections. Destroy after use.
14/15	IOLD/INEW in EMERGE. Are used locally only to accumulate summation cross sections on union grid.
20-99	User's choice for NENDF and NPEND to link RECONR with other NJOY modules.
5,6,7	See the NJOY chapter for a description of the I/O units.

Note that 11, 12, 14, and 15 are always binary. Unit 10 has the same mode as NENDF. Unit 13 is binary when used in RESXS, and it has the same mode as NPEND elsewhere. NPEND can have a different mode than NENDF.

I. Storage Allocation

Storage allocation in RECONR is sensitive to (1) the amount of resonance parameter data, (2) the size of the resonance reconstruction stack, (3) the use of $\psi\chi$ broadening, and (4) the sizes of the LOADA/FINDA buffers. Other storage requirements are minor.

Buffer sizes can be reduced or increased at will. The result is a storage/speed tradeoff with no change in capability or accuracy. See NBUFG=2000, NBUFR=2000, and NBUF=2000 in RECONR.

The $\psi\chi$ broadening option requires 7688 words of additional storage. Therefore, the container array in /RSTORE/ can be reduced significantly if $\psi\chi$ is not required. No code changes are needed—just avoid TEMPR greater than zero.

Resonance reconstruction in RESXS uses 5×NDIM words. The parameter NDIM determines the smallest subdivision of a panel that can be obtained. Using NDIM=20

allows points to be generated with spacing as small as one-millionth of the panel size (2^{20}).

J. References

1. C. R. Weisbin, P. D. Soran, R. E. MacFarlane, D. R. Harris, R. J. LaBauve, J. S. Hendricks, J. E. White, and R. B. Kidman, "MINX: A Multigroup Interpretation of Nuclear X-Sections from ENDF/B," Los Alamos Scientific Laboratory report LA-6486-MS (ENDF-237) (1976).

2. O. Ozer, "RESEND: A Program to Preprocess ENDF/B Materials With Resonance Files into Pointwise Form," Brookhaven National Laboratory report BNL-17134 (1972).

3. R. Kinsey, Ed., "ENDF-102, Data Formats and Procedures for the Evaluated Nuclear Data File, ENDF," Brookhaven National Laboratory report BNL-NCS-50496 (ENDF-102), 2nd Edition (ENDF/B-V) (1979).

4. P. F. Rose and C. L. Dunford, Eds., "ENDF-102, Data Formats and Procedures for the Evaluated Nuclear Data File, ENDF-6," available from the National Nuclear Data Center, Brookhaven National Laboratory, as a limited distribution informal report BNL-NCS-44945 (July 1990).

5. B. J. Toppel, A. L. Rago, and D. M. O'Shea, "MC^2, A Code to Calculate Multigroup Cross Sections," Argonne National Laboratory report ANL-7318 (1967).

6. H. Henryson II, B. J. Toppel, and C. G. Stenberg, "MC^2-2: A Code to Calculate Fast Neutron Spectra and Multigroup Cross Sections," Argonne National Laboratory report ANL-8144 (ENDF-239) (1976).

IV. BROADR

BROADR generates Doppler-broadened cross sections in PENDF format starting from piecewise linear cross sections in PENDF format. The input cross sections can be from RECONR or from a previous BROADR run. The code is based on SIGMA1[1] by D. E. Cullen. The method is often called "kernel broadening" because it uses a detailed integration of the integral equation defining the effective cross section. It is a fully accurate method, treating all resonance and nonresonance cross sections including multilevel effects. BROADR has the following features:

- An alternate calculation is used for low energies and high temperatures that corrects a numerical problem of the original SIGMA1. (This problem has been corrected in another way in later versions of SIGMA1.)
- Variable dimensioning is used, which allows the code to be run on large or small machines with full use of whatever storage is made available.
- All low-threshold reactions are broadened in parallel on a union grid. This makes the code run faster than the original SIGMA1.
- The union grid is constructed adaptively to give a linearized representation of the broadened cross section with tolerances consistent with those used in RECONR. Energy points may be added to or removed from the input grid as required for the best possible representation.
- Binary input and output can be used. This roughly halves the time required for a typical run on some computer systems, and it allows the full accuracy of the machine to be used.
- The summation cross sections—total, nonelastic, and sometimes fission or (n,2n)—are reconstructed to equal the sum of their parts.
- The file directory (actually an index to the reactions present) is updated.

A. Doppler-Broadening Theory

The effective cross section for a material at temperature T is defined to be that cross section that gives the same reaction rate for stationary target nuclei as the real cross section gives for moving nuclei. Therefore,

$$\rho v \overline{\sigma}(v,T) = \int d\mathbf{v}' \rho \, |\mathbf{v} - \mathbf{v}'| \, \sigma(|\mathbf{v} - \mathbf{v}'|) \, P(\mathbf{v}', T) \, , \tag{1}$$

where $\mathbf{v}$ is the velocity of the incident particles, $\mathbf{v}'$ is the velocity of the target, ρ is the density of target nuclei, σ is the cross section for stationary nuclei, and $P(\mathbf{v}', T)$ is the distribution of target velocities in the laboratory system. For many

cases of interest, the target motion is isotropic and the distribution of velocities can be described by the Maxwell-Boltzmann function

$$P(\mathbf{v}',T)\,d\mathbf{v}' = \frac{\alpha^{3/2}}{\pi^{3/2}}\exp(-\alpha v'^2)\,d\mathbf{v}' \,, \tag{2}$$

where $\alpha = M/(2kT)$, k is Boltzmann's constant, and M is the target mass.

Equation 1 can be partially integrated in terms of the relative speed $V = |\mathbf{v}-\mathbf{v}'|$ to give the standard form of the Doppler-broadened cross section:

$$\overline{\sigma}(v) = \frac{\alpha^{1/2}}{\phi^{1/2}\,v^2}\int_0^\infty dV\,\sigma(V)\,V^2\left\{e^{-\alpha(V-v)^2} - e^{-\alpha(V+v)^2}\right\} . \tag{3}$$

It is instructive to break this up into two parts:

$$\overline{\sigma}(v) = \sigma^*(v) - \sigma^*(-v) \,, \tag{4}$$

where

$$\sigma^*(v) = \frac{\alpha^{1/2}}{\pi^{1/2}v^2}\int_0^\infty dV\,\sigma(V)\,V^2\,e^{-\alpha(V-v)^2} \,. \tag{5}$$

The exponential function in Eq. (5) limits the significant part of the integral to the range

$$v - \frac{4}{\sqrt{\alpha}} < V < v + \frac{4}{\sqrt{\alpha}}.$$

For $\sigma^*(-v)$, the integral depends only on velocities satisfying

$$0 \le V < \frac{4}{\sqrt{\alpha}}.$$

These results can be converted to energy units using

$$E_m = \frac{1}{2}m(\frac{4}{\sqrt{\alpha}})^2 = \frac{16kT}{A}.$$

Some examples are given in Table I. Doppler-broadening effects will be important below this energy and for any features such as resonances, thresholds, or artificial discontinuities in evaluations that are not slowly varying with respect to $2\sqrt{E_m E}$. As an example, for ^{235}U at 100 eV, Doppler effects are important for features smaller than about 0.8 eV.

The numerical evaluation of Eq. (5) developed for SIGMA1 assumes that the cross section can be represented by a piecewise linear function of energy to acceptable accuracy. This is just the form of the NJOY PENDF tapes (see RECONR).

Table 1: Energy Parameter for Effective Doppler-Broadening

Target	Temperature	Energy Parameter (E_m)
H_2	300 K	0.2 eV
U-235	300 K	0.0017 eV
U-235	1.0 keV	69 eV

Defining the reduced variables $y = \sqrt{\alpha x}$ and $x = \sqrt{\alpha V}$, the cross section becomes

$$\sigma(x) = \sigma_i + s_i(x^2 - x_i^2) \ , \tag{6}$$

with slope $s_i = (\sigma_{i+1} - \sigma_i)/(x_{i+1}^2 - x_i^2)$. Eq. (5) can now be written as

$$\sigma^*(y) = \frac{1}{\pi^{1/2} y^2} \sum_{i=0}^{N} \int_{x_i}^{x_{i+1}} \sigma(x)\, x^2 \mathrm{e}^{-(x-y)^2} dx = \sum_i \{A_i \left[\sigma_i - s_i x_i^2\right] + B_i s_i\} \ , \tag{7}$$

where

$$\begin{aligned} x_0 &= 0 \ , \\ x_{N+1} &= \infty \ , \\ A_i &= \frac{1}{y^2} H_2 + \frac{2}{y} H_1 + H_0 \ , \text{ and} \\ B_i &= \frac{1}{y^2} H_4 + \frac{4}{y} H_3 + 6H_2 + 4yH_1 + y^2 H_0 \ , \end{aligned}$$

and where H_n is shorthand for $H_n(x_i - y, x_{i+1} - y)$. The extrapolations to zero and infinity assume a constant cross section ($s_0 = s_N = 0$). The H functions are the incomplete probability integrals defined by

$$H_n(a,b) = \frac{1}{\sqrt{\pi}} \int_a^b z^n \, \mathrm{e}^{-z^2} \, dz \ . \tag{8}$$

These functions can be computed in two ways. First,

$$H_n(a,b) = F_n(a) - F_n(b) \ , \tag{9}$$

where

$$F_n(a) = \frac{1}{\sqrt{\pi}} \int_a^\infty z^n \, \mathrm{e}^{-z^2} \, dz \ . \tag{10}$$

These functions satisfy a recursion relation that can be used to obtain

$$F_0(a) = \frac{1}{2}\mathrm{erfc}(a)\,, \tag{11}$$

$$F_1(a) = \frac{1}{2\sqrt{\pi}}\exp(-a^2)\,, \text{ and} \tag{12}$$

$$F_n(a) = \frac{n-1}{2}F_{n-2}(a) + a^{n-1}F_1(a)\,, \tag{13}$$

where $\mathrm{erfc}(a)$ denotes the complementary error function

$$\mathrm{erfc}(a) = \frac{2}{\sqrt{\pi}}\int_a^\infty \mathrm{e}^{-z^2}dz\,. \tag{14}$$

However, when $F_n(a) \approx F_n(b)$, the difference in Eq. (8) may lose significance. In such cases, $H_n(a,b)$ can be computed by a second method based on a direct Taylor expansion of the defining integral.[2] Write

$$H_n(a,b) = \frac{1}{\sqrt{\pi}}\int_0^b z^n\mathrm{e}^{-z}dz - \frac{1}{\sqrt{\pi}}\int_0^a z^n\mathrm{e}^{-z^2}dz = G_n(b) - G_n(a)\,. \tag{15}$$

But by Taylor's Theorem,

$$G_n(b) - G_n(a) = \frac{b-a}{1!}G_n'(a) + \ldots + \frac{(b-a)^m}{m!}G_n^{(m)}(a) + \ldots\,. \tag{16}$$

Also,

$$G_n^{(m)}(x) = \frac{d^{m-1}}{dx^{m-1}}[x^n\mathrm{e}^{-x^2}] = \mathrm{e}^{-x^2}P_n^m(x)\,, \tag{17}$$

where $P_n^m(x)$ is a polynomial with recursion relation

$$P_n^m(x) = \frac{d}{dx}P_n^{m-1}(x) - 2xP_n^{m-1}(x)\,, \tag{18}$$

with $P_n^1 = x^n$. From this point, it is straightforward to generate terms until the desired number of significant figures is obtained.

When interpreting BROADR output, it is useful to remember several important features of the Doppler-broadening process. A $1/v$ cross section remains unchanged. Contrary to "popular knowledge", the area under a resonance does not remain unchanged unless $E \gg kT/A$. In fact, each resonance develops a new $1/v$ tail. Finally, a constant cross section (for example, elastic scattering) develops a $1/v$ tail at low energies after Doppler-broadening. These effects are shown in Figs. 1, 2, and 3; they can be best understood by noting that the Doppler process

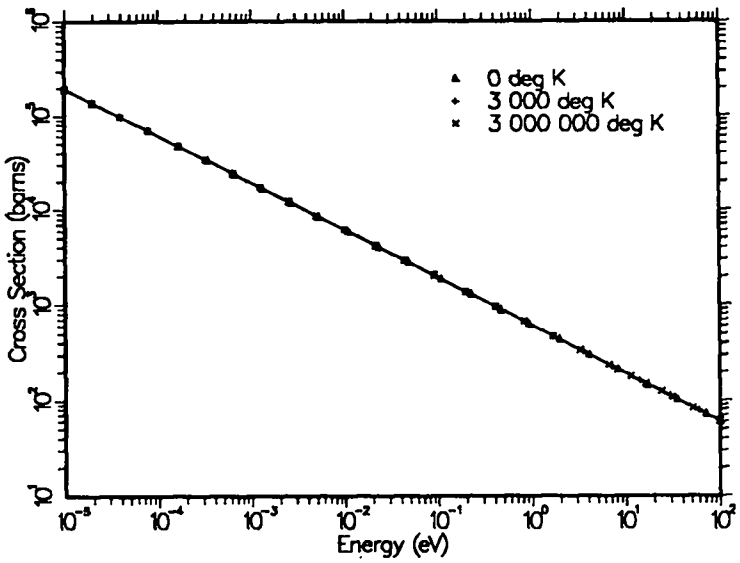


Figure 1: The (n,α) cross section for ^{10}B from ENDF/B-V for three different temperatures showing that a $1/v$ cross section is invariant under Doppler-broadening.

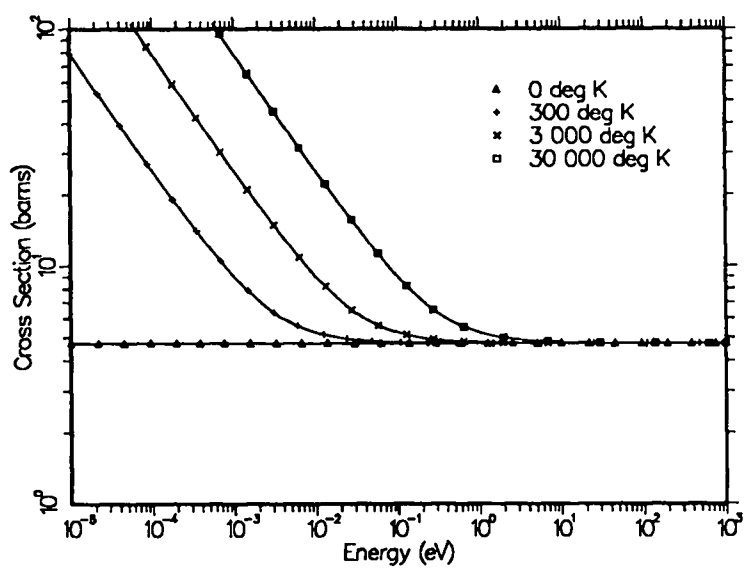


Figure 2: The elastic cross section for carbon from ENDF/B-V showing that Doppler-broadening a constant cross section adds a $1/v$ tail.

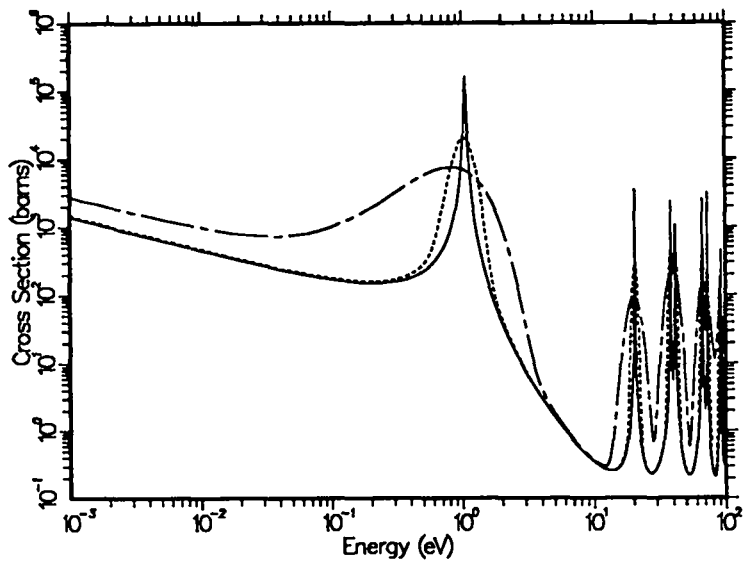


Figure 3: The (n,γ) cross section for ^{240}Pu from ENDF/B-V for several temperatures showing the effects of Doppler broadening on resonances. The temperatures are 0 K (solid), 30 000 K (dotted), and 300 000 K (dash-dot). The higher resonances behave in the classical manner even at 30 000 K; note that the line shape returns to the asymptotic value in the wings of the resonance. All resonances at 300 000 K (and to a lesser extent the first resonance for 30 000 K) show the additional $1/v$ component that appears when kT/A is large with respect to the resonance energy.

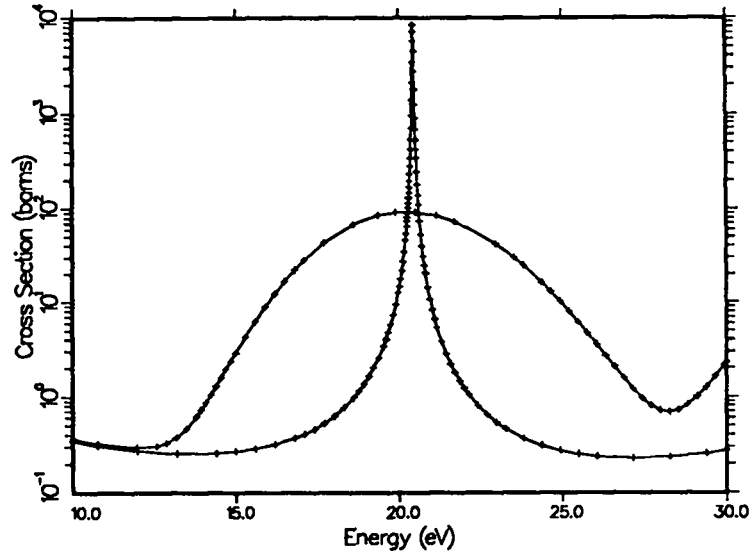


Figure 4: An expanded plot of the 20 eV resonance from Fig. 3 showing both thinning and "thickening" of the energy grid produced adaptively by BROADR. The two curves show the capture cross section at 0 K and 300 000 K. Note that the high-temperature curve has fewer points than the 0 K curve near the peak at 20 eV and more points in the wings near 15 eV and 25 eV. Clearly, using the 0 K grid to represent the broadened cross section in the wings of this resonance would give poor results.

preserves reaction rate $v\sigma(v)$ according to Eq. (1), and a finite reaction rate is expected for $T > 0$ even as $v \to 0$.

Earlier versions of BROADR and SIGMA1 assumed that the input energy grid from RECONR could also be used to represent the Doppler-broadened cross section before thinning. The grid was then thinned to take advantage of the smoothing effect of Doppler broadening. Unfortunately, this assumption is inadequate. The reconstruction process in RECONR places many points near the center of a resonance to represent its sharp sides. After broadening, the cross section in this energy region becomes rather smooth; the sharp sides are moved out to energies where RECONR provides few points. At still higher energies, the resonance line shape returns to its asymptotic value, and the RECONR grid is adequate once more. The more recent versions of BROADR check the cross section between points of the incoming energy grid, and add additional grid points if they are necessary to represent the broadened line shape to the desired accuracy. This effect is illustrated in Fig. 4.

B. Data-Paging Methodology

A piecewise linear representation of a reaction cross section of a resonance material may require a very large number of energy points. For example, ENDF/B-IV U-238 (MAT1262) requires 57 400 points for the total cross section for 0.5% precision (ERRMAX=err). It is impractical to load all these points into memory simultaneously. However, the discussion following Eq. (5) in the theory section shows that only a limited energy range around the point of interest is required.

The strategy used is to stage the cross-section data into three "pages" of NPAGE points each. Points in the center page can then be broadened using the NPAGE or more points on each side of the point of interest. If $v - 4/\sqrt{\alpha}$ and $v + 4/\sqrt{\alpha}$ are both included in the three-page range, accurate broadening can be performed. If not, a diagnostic warning is printed; the user should repeat the calculation with a smaller temperature step or a larger page size.

There are many different reaction cross sections for each material. However, the cross sections for high velocities are normally smooth with respect to $32kT/A$ for any temperatures outside of stellar photospheres; therefore, they do not show significant Doppler effects. The code uses the input value THNMAX, or the upper limit of the resolved-resonance energy range, or the lowest threshold (typically>100 keV) as a breakpoint. No Doppler-broadening or energy-grid reconstruction is performed above that energy. Furthermore, the A_i and B_i factors in Eq. (7) depend only on the energy (or velocity) values and not on the cross sections. Since the A_i and B_i are expensive to compute, the code computes them only once for the points of a unionized energy grid. The sum of Eq. (7) is accumulated for all the non-threshold reactions simultaneously. This trick helps make BROADR several times faster than the original SIGMA1.

C. Coding Details

The code begins by reading the user's input (see Section D). Storage is then allocated for the LOADA/FINDA buffers (IBUF0 and IBUFN) and for the scratch storage (ISCR). The buffer length NBUF can be changed at will (currently NBUF is 1000).

The input PENDF tape is searched for the desired material (MAT1). If the restart option is set (ISTART=1), the temperatures less than or equal to TEMP1 for MAT1 are assumed to have been broadened previously, and they are copied to the output file. In either case, the files for TEMP1 are copied to a scratch file on unit NSCR1 (currently set to 10).

Next, NSCR1 is rewound and examined reaction by reaction. The energy grid from the total cross section (MT1) is saved on scratch storage using LOADA. If the input tape has not been through RECONR, the BROADR module will still run, but at possibly reduced accuracy. The next low-threshold reaction (that is, the next reaction with a threshold less than EMIN, which is currently 1 eV) is located on NSCR1. The energy points are retrieved from scratch file IOLD (12 or 13) using FINDA, the cross sections for this reaction are computed on this grid, and the results are stored on scratch file INEW (13 or 12) using LOADA. The units for IOLD and INEW are then exchanged, and the entire process is repeated for the next low-threshold reaction.

The final result of this process is a list of NREAC low-threshold-reaction types in MTR (usually MT2, MT18, and MT102), the threshold value for the first high-threshold reaction (or the input value) in THNMAX, and scratch file IOLD containing the energy grid and all the low-threshold reactions (there are N2IN points).

Now that the number of reactions to be broadened simultaneously is known (NREAC), storage for data paging can be assigned. The total amount of storage available is NAMAX-2*NBUF-NWSCR-40. The value of NAMAX should be as large as possible (current value is 30 000). This space is divided up into the largest possible page size, NPAGE. An overflow region NSTACK is also allocated. Subroutine STORAG is used to allocate three pages for energies (E), three pages for each reaction cross section (S), one extended page for the broadened energy grid (EB), and three extended pages for the broadened cross section (SB). This system is designed to use the available storage with maximum efficiency.

The cross sections on IOLD are now broadened by FILE3 (see below) and the results are written on scratch unit INEW using LOADA.

The directory from NSCR1 is revised to reflect any thinning or thickening and written on the output PENDF tape (NOUT). Note that the new temperature is written into the first word of the Hollerith data record to simplify later searching.

The broadened cross sections are now converted into ENDF TAB1 records and merged with the unbroadened cross sections on NSCR1. The total cross section [and sometimes nonelastic, fission, and (n,2n)] is reconstructed to equal the sum of its parts. The new Doppler-broadened "MAT" on NOUT is a legal PENDF file with the same MAT number as the original data but with a new temperature.

The process is now repeated for each of the NTEMP2 final temperatures TEMP2 requested. Note that after each step INEW contains the new data and IOLD contains the previous data. If the "bootstrap" option is set (ISTRAP=1), these units are interchanged. For this option, TEMP2(IT) is always obtained from TEMP2(IT-1). Because of the thinning effect of Doppler-broadening, the broadening runs faster at each step. The accumulation of error is usually not a problem. For ISTRAP=0, TEMP1 is used for the starting temperature every time.

The broadening and energy-grid reconstruction are directed by FILE3. The routine loads data into the appropriate memory pages from scratch file IOLD and then either calls BROADN to broaden it (with thinning or thickening of the grid as necessary) or calls THINB to thin it without broadening. The results are written onto scratch file INEW.

In BROADN, the energy grid points just loaded into E by FILE3 are converted to the dimensionless variables x and y [see Eq. (6)]. An adaptive reconstruction of the Doppler broadened cross section is then performed for the energy range in the center page using an inverted stack algorithm like the one described for RECONR. The upper limit of each panel is taken to be a point from the input grid, but in order to allow for thinning, up to NMAX=10 of the input grid points can be skipped before the actual upper limit is selected. In addition, the energy of the upper limit cannot be more than STEP=2.0 times the energy of the preceding point. The cross sections are now computed at the midpoint of the top panel in the stack using BSIGMA. If the results differ from the values obtained by interpolation by more than the specified tolerance, the new point is added to the stack, and the tests are repeated. Otherwise, the top point in the stack is converged. A backward check is made to see if some of the previous points can be removed based on the new value, the new value is stored in the output array, and the height of the stack is reduced by one. The routine now tries to subdivide the new panel at the top of the stack in the same way. When the stack has been reduced to one element, a new upper limit is chosen from the input energy grid as described above, and the entire process is repeated. The reconstruction logic in BROADR uses the same integral tests as RECONR. Refer to the RECONR chapter for more details.

Subroutine BSIGMA is used to calculate the actual broadened cross section at an energy point using the data in the three pages. First, the routine locates the energy panel containing the desired energy EN. It then loops over intervals below the current point adding in contributions to $\bar{\sigma}$ from the $V-v$ term of Eq. (5) until the contributions to the cross section become small. If the lower limit of the bottom page is reached before convergence, a warning message is issued. The

routine then loops over intervals above the current point until convergence. Once again, a warning is issued if necessary. Finally, the low-energy term [the one involving $V+v$ in Eq. (5)] is added, if applicable.

Subroutine THINB is provided for cases where the input cross section set is to be thinned only. This routine uses the original SIGMA1 method. The first input point is always kept. The routine then loops over higher energy values. For each grid point, all the points from there back to the last accepted point are checked for their deviation from a straight line. If they all can be removed without violating the specified tolerance, the interval is extended to the next higher point and the tests are repeated. If any point in the range is too far from the linear approximation, the last point in the range is accepted as an output point, and the testing process is repeated starting from this new lower limit. The procedure terminates when all of the points in the middle page have been thinned, and control is returned to FILE3 to get the next page of data.

Subroutine HUNKY has been modified from the original SIGMA1 version to implement the alternate $H_n(a,b)$ calculation when necessary (see HNABB). When using the direct method, F_n values from the previous step are used in the difference of Eq. (9), and FUNKY is called to get the new values. The A_i and B_i of Eq. (7) are related to the S1 and S2 here.

Subroutine FUNKY evaluates $F_n(a)$ by the recursion formula of Eq. (13) using a rational approximation to the reduced complementary error function.[2]

Function HNABB implements the alternate calculation described by Eqs. (15)-(18). The series expansion is continued until about six significant figures are guaranteed (see EPS and HNABB). Currently, HNABB is called when only four significant figures are reliable in HUNKY (see TOLER in HUNKY).

D. User Input

The following input instructions have been copied from the comment cards at the start of BROADR.

```
*---INPUT SPECIFICATIONS (FREE FORMAT)-------------------------*
*                                                              *
* CARD 1                                                       *
*    NIN      INPUT PENDF TAPE                                 *
*    NOUT     OUTPUT PENDF TAPE                                *
* CARD 2                                                       *
*    MAT1     MATERIAL TO BE PROCESSED                         *
*    NTEMP2   NUMBER OF FINAL TEMPERATURES (MAXIMUM=10)        *
*    ISTART   RESTART (0 NO, 1 YES, DEFAULT 0)                 *
*    ISTRAP   BOOTSTRAP (0 NO, 1 YES, DEFAULT 0)               *
```

```
*    TEMP1    STARTING TEMPERATURE FROM NIN (DEFAULT=0.K)          *
* CARD 3                                                           *
*    ERRTHN   FRACTIONAL TOLERANCE FOR THINNING                    *
*    THNMAX   MAX. ENERGY FOR BROADENING AND THINNING              *
*             (DEFAULT=1 MEV)                                      *
*    ERRMAX   FRACTIONAL TOLERANCE USED WHEN INTEGRAL CRITERION    *
*             IS SATISFIED (SAME USAGE AS IN RECONR)               *
*             (ERRMAX.GE.ERRTHN, DEFAULT=20*ERRTHN)                *
*    ERRINT   PARAMETER TO CONTROL INTEGRAL THINNING               *
*             (USAGE AS IN RECONR) (DEFAULT=ERRTHN/10000)          *
*             SET VERY SMALL TO TURN OFF INTEGRAL THINNING.        *
*      (A GOOD CHOICE FOR THE CONVERGENCE PARAMETERS               *
*       ERRTHN, ERRMAX, AND ERRINT IS THE SAME SET OF              *
*       VALUES USED IN RECONR)                                     *
* CARD 4                                                           *
*    TEMP2    FINAL TEMPERATURES (DEG KELVIN)                      *
* CARD 5                                                           *
*    MAT1     NEXT MAT NUMBER TO BE PROCESSED WITH THESE           *
*             PARAMETERS.  TERMINATE WITH MAT1=0.                  *
*                                                                  *
*---INPUT OPTIONS-------------------------------------------------*
*                                                                  *
* THE OUTPUT TAPE WILL CONTAIN THE NTEMP2 FINAL TEMPERATURES       *
* SPECIFIED.  IT IS NECESSARY TO HAVE TEMP1.LE.TEMP2(1).           *
* IF TEMP2.EQ.TEMP1, THE DATA WILL BE THINNED ONLY.                *
*                                                                  *
* RESTART    CONTINUE BROADENING AN EXISTING PENDF TAPE.  ALL      *
*            TEMPERATURES ARE COPIED THROUGH TEMP1.  ADDITIONAL    *
*            FINAL TEMPERATURES ARE ADDED BY STARTING WITH THE     *
*            DATA AT TEMP1.                                        *
*                                                                  *
* BOOTSTRAP  IF BOOTSTRAP IS NOT REQUESTED, EACH FINAL TEMPERA-    *
*            TURE IS GENERATED BY BROADENING DIRECTLY FROM         *
*            TEMP1 TO TEMP2.  IF BOOTSTRAP IS REQUESTED, EACH      *
*            FINAL TEMPERATURE IS BROADENED FROM THE PRECEDING     *
*            TEMPERATURE.  THIS OPTION IS FASTER DUE TO THE        *
*            THINNING IN THE PREVIOUS STEP.  HOWEVER, ERRORS       *
*            ACCUMULATE.                                           *
*                                                                  *
* THNMAX     THE UPPER LIMIT FOR BROADENING AND THINNING IS THE    *
*            LOWEST OF THE INPUT VALUE OF THNMAX, THE LOWEST       *
*            REACTION THRESHOLD, OR THE START OF THE UNRESOLVED    *
*            RANGE.  IF THERE IS RESOLVED-UNRESOLVED OVERLAP,      *
*            THE OVERLAP REGION IS INCLUDED IN THE BROADENING.     *
*            A NEGATIVE VALUE OF THNMAX WILL OVERRIDE THE          *
*            RESOLVED AND THRESHOLD LIMITS.  THIS ALLOWS THE       *
*            FIRST FEW THRESHOLD REACTIONS TO BE BROADENED IF      *
*            DESIRED.  THE MAGNITUDE OF THNMAX MUST BE CHOSEN      *
*            TO KEEP THE NUMBER OF BROADENABLE REACTIONS LESS      *
*            THAN OR EQUAL TO THE MAXIMUM OF NTT (10).             *
*                                                                  *
********************************************************************
```

Note that TEMP1 need not occur on NOUT if ISTART=0. The restart option (ISTART=1) enables the user to add new temperatures to the end of an existing PENDF tape. This option is also useful if a job runs out of time while processing, for example, the first temperature. The job can be restarted from the NOUT. The first four temperatures will be copied to the new NOUT and broadening will continue for temperature five. The bootstrap option speeds up the code by using the broadened result for TEMP2(i-1) as the starting point to obtain TEMP2(i). The THNMAX parameters can be used to speed up a calculation or to prevent the broadening of inappropriate data such as sharp steps or triangles in an evaluated cross section (for example, ENDF/B-V lead).

The following example prepares a single output tape containing Am-241 and Am-243 from ENDF/B-IV at two temperatures each. The line numbers are for reference only; they are not part of the input.

```
1.  0
2.  4
3.  BROADR
4.  20 21
5.  1056 2 0 1/
6.  .001/
7.  300. 1200.
8.  1057/
9.  0/
```

On line 4, unit 20 should contain a RECONR-generated BCD PENDF tape of 0 K cross sections for the two isotopes. Four materials will be generated on unit 21 with 0.1% accuracy. Best results are obtained when the error tolerance ERRTHN and the optional integral-thinning controls ERRMAX and ERRINT are the same as those used for the RECONR run. Note that "bootstrap" was requested, so the input data for the 1200 K case will be the output from the 300 K case.

E. Error Messages

ERROR IN BROADR*NIN AND NOUT MUST BE SAME MODE**

Use coded to coded, or blocked binary to blocked binary. The latter is much faster due to the several tape copies performed in BROADR.

ERROR IN BROADR*TOO MANY TEMPERATURES**

The number of temperatures is limited by the statement TEMP2(10).

ERROR IN BROADR*TOO MANY LOW THRESHOLD REACTIONS**

The current limit is set by the statement NTT=10 in BROADR. Check TT, MTR, and NTT in BROADR, TT in FILE3, and SBT in BROADN. Too many reactions might also strain the total storage (see /BSTORE/A(30000) and NAMAX=30000 in BROADR).

MESSAGE FROM BROADR---NO BROADENABLE REACTIONS

No low threshold reactions were found.

ERROR IN BROADR***STORAGE EXCEEDED

Insufficient storage to update directory. Increase NWSCR in BROADR.

MESSAGE FROM STOUNX---SIGMA ZERO DATA REMOVED FROM UNRESOLVED

The input PENDF tape already contained a special unresolved section in File 2. It has been removed. Rerun UNRESR if necessary.

MESSAGE FROM BSIGMA---BROADENING TRUNCATED AT A=----

The page is too small for the temperature difference requested. Increase total storage available or repeat the calculation with smaller temperature steps and ISTRAP=1. The normal maximum size of A is 4.0 and A is inversely proportional to $T_i - T_{i-1}$.

F. Input/Output Units

The following units are used for input and output by BROADR.

10	NSCR1 in BROADR. Contains the ENDF/B data at the initial temperature.
12/13	IOLD/INEW in BROADR. Contains union grid and low threshold reactions.
20-26	User's choice for NIN and NOUT to link with other modules.

Units 12 and 13 will always be binary. Unit 10 will have the same mode as NIN and NOUT (binary mode is recommended).

G. Storage Allocation

All storage is divided in the most efficient way possible. The container array in /BSTORE/ and NAMAX should be made as large as possible. The value of NBUF can be increased or decreased at will—large values will give faster execution. The value for NWSCR depends on the size of the ENDF/B dictionary, and 1 000 words is sufficient for all current evaluations.

H. References

1. D. E. Cullen, "Program SIGMA1 (Version 77-1): Doppler Broaden Evaluated Cross Sections in the Evaluated Nuclear Data File/Version B (ENDF/B) Format," Lawrence Livermore National Laboratory report UCRL-50400, Vol. 17, Part B (1977).
2. M. Abromowitz and I. Stegun, *Handbook of Mathematical Functions* (Dover Publications, New York, 1965).

V. UNRESR

The UNRESR module is used to produce effective self-shielded cross sections for resonance reactions in the unresolved energy range. In ENDF-format evaluations, the unresolved range begins at an energy where it is difficult to measure individual resonances and extends to an energy where the effects of fluctuations in the resonance cross sections become unimportant for practical calculations. As described in the ENDF format manual,[1] resonance information for this energy range is given as average values for resonance widths and spacings together with distribution functions for the widths and spacings. This representation can be converted into effective cross sections suitable for codes that use the background cross section method, often called the Bondarenko method,[2] using a method originally developed for the MC^2 code[3] and extended for the ETOX code.[4] This unresolved-resonance method has the following features:

- Flux-weighted cross sections are produced for the total, elastic, fission, and capture cross sections, including competition with inelastic scattering.
- A current-weighted total cross section is produced for calculating the effective self-shielded transport cross section.
- Up to 10 values of temperature and 10 values of σ_0 are allowed.
- The energy grid used is consistent with the grid used by RECONR.
- The computed effective cross sections are written on the PENDF tape in a specially defined section (MF2,MT152) for use by other modules.
- The accurate quadrature scheme from the MC^2-2 code[5] is used for computing averages over the ENDF statistical distribution functions.

A. Theory

In the unresolved energy range, it is not possible to define precise values for the cross sections of the resonance reactions $\sigma_x(E)$, where x stands for the reaction type (total, elastic, fission, or capture). It is only possible to define average values. Of course, these average values should try to preserve the reaction rate:

$$\overline{\sigma}_{0x}(E^*) = \frac{\int_{E_1}^{E_2} \sigma_x(E)\,\phi_0(E)\,dE}{\int_{E_1}^{E_2} \phi_0(E)\,dE}\,, \tag{1}$$

where $\phi_0(E)$ is the scalar flux, E^* is an effective energy in the range $[E_1, E_2]$, and the range $[E_1, E_2]$ is large enough to hold many resonances but small with respect

to slowly varying functions of E. In order to calculate effective values for the transport cross section, it is necessary to compute the current-weighted total cross section also. It is given by

$$\overline{\sigma}_{1t}(E^*) = \frac{\int_{E_1}^{E_2} \sigma_x(E)\,\phi_1(E)\,dE}{\int_{E_1}^{E_2} \phi_1(E)\,dE}\,, \tag{2}$$

where the P_1 component of the neutron flux, $\phi_1(E)$, is proportional to the neutron current. To proceed farther, it is necessary to choose a model for the shape of $\phi_\ell(E)$ in the vicinity of E^*. The model used in UNRESR is based on the B_0 approximation for large homogeneous systems and narrow resonances:

$$\phi_\ell(E) = \frac{C(E)}{[\,\Sigma_t(E)\,]^\ell}\,, \tag{3}$$

where $C(E)$ is a slowly varying function of E , and $\Sigma_t(E)$ is the macroscopic total cross section for the system. In order to use this result in Eq. (1), it is further assumed that the effects of other isotopes in the mixture can be approximated by a constant called σ_0 in the range $[E_1, E_2]$, or

$$\phi_\ell(E) = \frac{C(E)}{[\,\sigma_0 + \sigma_t(E)\,]^\ell}\,. \tag{4}$$

Therefore, the effective cross sections in the unresolved range are represented by

$$\overline{\sigma}_{0x}(E^*) = \frac{\int_{E_1}^{E_2} \dfrac{\sigma_x(E)}{\sigma_0 + \sigma_t(E)} C(E)\,dE}{\int_{E_1}^{E_2} \dfrac{1}{\sigma_0 + \sigma_t(E)} C(E)\,dE}\,, \tag{5}$$

with x being t for total, e for elastic, f for fission, and γ for capture, and

$$\overline{\sigma}_{1t}(E^*) = \frac{\int_{E_1}^{E_2} \dfrac{\sigma_x(E)}{[\,\sigma_0 + \sigma_t(E)\,]^2} C(E)\,dE}{\int_{E_1}^{E_2} \dfrac{1}{[\,\sigma_0 + \sigma_t(E)\,]^2} C(E)\,dE}\,. \tag{6}$$

This equation can also be written in the equivalent form

$$\overline{\sigma}_{1t}(E^*) = \frac{\int_{E_1}^{E_2} \dfrac{1}{\sigma_0 + \sigma_t(E)} C(E)\,dE}{\int_{E_1}^{E_2} \dfrac{1}{[\,\sigma_0 + \sigma_t(E)\,]^2} C(E)\,dE} - \sigma_0\,. \tag{7}$$

The parameter σ_0 in Eq. (4) deserves more discussion. It can be looked on as a parameter that controls the depth of resonance dips in the flux. When σ_0 is large with respect to the peak cross sections of resonances in $\sigma_t(E)$, the shape of the flux is essentially $C(E)$. For smaller values of σ_0, dips will develop in the flux that correspond to peaks in σ_t. These dips will cancel out part of the reaction rate in the region of the peaks, thus leading to self-shielding of the cross section. Analysis shows that it is possible to use this single parameter to represent the effects of admixed materials or the effects of neutron escape from an absorbing region. See the GROUPR chapter of this manual for additional details.

The cross sections that appear in the above integrals can be written as the sum of a resonant part and a smooth part as follows:

$$\sigma_x(E) = b_x + \sigma_{Rx}(E) = b_x + \sum_s \sum_r \sigma_{xsr}(E-E_{sr}) \,, \tag{8}$$

where s is an index to a spin sequence, r is an index to a particular resonance in that spin sequence, and E_{sr} is the center energy for that resonance. The smooth part b_x can come from a smooth background given in the ENDF file, and it also includes the potential scattering cross section σ_p for the elastic and total cross sections (x=t and x=e). In terms of the smooth and resonant parts, the effective cross sections become

$$\overline{\sigma}_{0x}(E^*) = b_x + \frac{\displaystyle\int_{E_1}^{E_2} \frac{\sigma_{Rx}(E)}{\overline{\sigma}+\sigma_{Rt}(E)} C(E)\,dE}{\displaystyle\int_{E_1}^{E_2} \frac{1}{\overline{\sigma}+\sigma_{Rt}(E)} C(E)\,dE} \,, \tag{9}$$

and

$$\overline{\sigma}_{1t}(E^*) = \frac{\displaystyle\int_{E_1}^{E_2} \frac{1}{\overline{\sigma}+\sigma_{Rt}(E)} C(E)\,dE}{\displaystyle\int_{E_1}^{E_2} \frac{1}{[\overline{\sigma}+\sigma_{Rt}(E)]^2} C(E)\,dE} - \sigma_0 \,, \tag{10}$$

where $\overline{\sigma} = b_t + \sigma_0$. It is convenient to transform the denominators of Eqs. (9) and (10) into

$$\int \frac{1}{\overline{\sigma}+\sigma_t} C\,dE = \frac{1}{\overline{\sigma}} \left\{ \int C\,dE - \int \frac{\sigma_t}{\overline{\sigma}+\sigma_t} C\,dE \right\} \,, \tag{11}$$

and

$$\int \frac{1}{[\overline{\sigma}+\sigma_t]^2} C\,dE = \frac{1}{\overline{\sigma}^2} \left\{ \int C\,dE - \int \frac{\sigma_t}{\overline{\sigma}+\sigma_t} C\,dE - \int \frac{\overline{\sigma}\sigma_t}{[\overline{\sigma}+\sigma_t]^2} C\,dE \right\} \,. \tag{12}$$

Furthermore, since $C(E)$ is assumed to be a slowly-varying function of E, it can be pulled out through all integrals and dropped. The average cross sections become

$$\overline{\sigma}_{0x} = b_x + \frac{\overline{\sigma} I_{0x}}{1 - I_{0t}} , \tag{13}$$

and

$$\overline{\sigma}_{1t} = b_t + \frac{\overline{\sigma} I_{1t}}{1 - I_{0t} - I_{1t}} . \tag{14}$$

The last equation can also be written in the form

$$\overline{\sigma}_{1t} = \overline{\sigma} \left[\frac{1 - I_{0t}}{1 - I_{0t} - I_{1t}} \right] - \sigma_0 . \tag{15}$$

The average cross sections are thereby seen to depend on two types of "fluctuation integrals:"

$$I_{0x} = \frac{1}{E_2 - E_1} \int_{E_1}^{E_2} \frac{\sigma_{Rx}(E)}{\overline{\sigma} + \sigma_{Rt}(E)} dE , \tag{16}$$

and

$$I_{1t} = \frac{1}{E_2 - E_1} \int_{E_1}^{E_2} \frac{\overline{\sigma} \sigma_{Rt}(E)}{[\overline{\sigma} + \sigma_{Rt}(E)]^2} dE , \tag{17}$$

where x can take on the values t, n, f, or γ. Note that $I_{1t} \leq I_{0t}$, the difference increasing as σ_0 decreases from infinity.

Inserting the actual sums over resonances into the formula for I_{0x} gives

$$I_{0x} = \frac{1}{E_2 - E_1} \int_{E_1}^{E_2} \frac{\sum_{sr} \sigma_{xsr}(E - E_{sr})}{\overline{\sigma} + \sum_{sr} \sigma_{tsr}(E - E_{sr})} dE . \tag{18}$$

If the resonances were widely separated, only the "self" term would be important, and one would obtain

$$I_{0x} = \sum_{sr} \frac{1}{E_2 - E_1} \int_{E_1}^{E_2} \frac{\sigma_{xsr}(E - E_{sr})}{\overline{\sigma} + \sigma_{tsr}(E - E_{sr})} dE . \tag{19}$$

Since the range of integration is large with respect to the width of any one resonance, the variable of integration can be changed to $\xi = E - E_{sr}$, and the limits on ξ can be extended to infinity. For any one sequence, the interval $E_2 - E_1$ is equal to the average spacing of resonances in that sequence times the number of resonances in the interval. Therefore,

$$I_{0x}^I = \sum_s \frac{1}{D_s} \frac{1}{N_s} \sum_r \int_{-\infty}^{\infty} \frac{\sigma_{xsr}(\xi)}{\overline{\sigma} + \sigma_{tsr}(\xi)} d\xi \tag{20}$$

where D_s is the average spacing, and the "I" superscript indicates that this is the "isolated resonance" result. Because there are assumed to be many resonances in the interval, the sum over resonances can be changed to a multiple integration over some characteristic set of parameters (such as widths) times the probability of finding a resonance with some particular values of the parameters:

$$\frac{1}{N}\sum_{r\in s} f_r = <f>_s = \int d\alpha P_s(\alpha) \int d\beta P_s(\beta) \cdots f(\alpha,\beta,\cdots) \; . \tag{21}$$

In the following text, this multiple integral (up to four fold) will be abbreviated by writing the α integral only. The final results for isolated resonances are as follows:

$$I^I_{0x} = \sum_s \frac{1}{D_s} \int P(\alpha) \int_{-\infty}^{\infty} \frac{\sigma_{xs\alpha}(\xi)}{\overline{\sigma} + \sigma_{ts\alpha}(\xi)} \, d\xi \, d\alpha \; , \tag{22}$$

and

$$I^I_{1t} = \sum_s \frac{1}{D_s} \int P(\alpha) \int_{-\infty}^{\infty} \frac{\overline{\sigma}\sigma_{ts\alpha}(\xi)}{[\overline{\sigma} + \sigma_{ts\alpha}(\xi)]^2} \, d\xi \, d\alpha \; . \tag{23}$$

If the effects of overlap are too large to be neglected, overlap corrections to the isolated resonance result can be constructed using the continued-fraction generator

$$\frac{1}{a+b} = \frac{1}{a}\left(1 - \frac{b}{a+b}\right) \; . \tag{24}$$

Starting with the I_0 integrals,

$$\begin{aligned} \frac{\sum_{sr}\sigma_{xsr}}{\overline{\sigma} + \sum_{sr}\sigma_{tsr}} &= \sum_{sr} \frac{\sigma_{xsr}}{\overline{\sigma} + \sigma_{tsr}} \Big\{ 1 \\ &\quad - \sum_{r'\neq r} \frac{\sigma_{tsr'}}{\overline{\sigma} + \sum\sigma_{tsr}} - \sum_{s'\neq s}\sum_{r'} \frac{\sigma_{ts'r'}}{\overline{\sigma} + \sum\sigma_{tsr}} \Big\} \; . \end{aligned} \tag{25}$$

Expand the second term in the braces to get

$$\begin{aligned} \frac{\sum_{sr}\sigma_{xsr}}{\overline{\sigma} + \sum_{sr}\sigma_{tsr}} &= \sum_{sr} \frac{\sigma_{xsr}}{\overline{\sigma} + \sigma_{tsr}} \Big\{ 1 \\ &\quad - \sum_{r'\neq r} \frac{\sigma_{tsr'}}{\overline{\sigma} + \sigma_{tsr} + \sigma_{tsr'}} \\ &\qquad \Big\{ 1 - \sum_{\substack{r''\neq r \\ r''\neq r'}} \frac{\sigma_{tsr''}}{\overline{\sigma} + \sum\sigma_{tsr}} - \sum_{s'\neq s}\sum_{r'} \frac{\sigma_{ts'r'}}{\overline{\sigma} + \sum\sigma_{tsr}} \Big\} \\ &\quad - \sum_{s'\neq s}\sum_{r'} \frac{\sigma_{ts'r'}}{\overline{\sigma} + \sum\sigma_{tsr}} \Big\} \; . \end{aligned} \tag{26}$$

Neglecting the products of three *different* resonances in sequence s gives

$$\begin{aligned}\frac{\sum_{sr}\sigma_{xsr}}{\overline{\sigma}+\sum_{sr}\sigma_{tsr}} &= \sum_{sr}\frac{\sigma_{xsr}}{\overline{\sigma}+\sigma_{tsr}} \\ &\times \left\{1-\sum_{r'\neq r}\frac{\sigma_{tsr'}}{\overline{\sigma}+\sigma_{tsr}+\sigma_{tsr'}}\right\} \\ &\times \left[1-\sum_{s'\neq s}\sum_{r'}\frac{\sigma_{ts'r'}}{\overline{\sigma}+\sum\sigma_{tsr}}\right] . \end{aligned} \tag{27}$$

The factor before the opening brace is the isolated resonance result, the factor in braces is the in-sequence overlap correction, and the factor in brackets is the sequence-sequence overlap correction. Note that recursion can be used to refine the sequence-sequence correction to any desired accuracy. Similarly, the I_1 integral requires

$$\begin{aligned}\frac{\sum_{sr}\overline{\sigma}\sigma_{xsr}}{[\overline{\sigma}+\sum_{sr}\sigma_{tsr}]^2} &= \sum_{sr}\frac{\overline{\sigma}\sigma_{xsr}}{[\overline{\sigma}+\sigma_{tsr}]^2}\Big[1 \\ &- \sum_{r'\neq r}\frac{\sigma_{tsr'}}{\overline{\sigma}+\sum\sigma_{tsr}} - \sum_{s'\neq s}\sum_{r'}\frac{\sigma_{ts'r'}}{\overline{\sigma}+\sum\sigma_{tsr}}\Big]^2 . \end{aligned} \tag{28}$$

Once more, we expand the fraction and neglect terms that will result in products of three or more different resonances in the same sequence. The result is

$$\begin{aligned}\frac{\sum_{sr}\overline{\sigma}\sigma_{xsr}}{[\overline{\sigma}+\sum_{sr}\sigma_{tsr}]^2} &= \sum_{sr}\frac{\overline{\sigma}\sigma_{xsr}}{[\overline{\sigma}+\sigma_{tsr}]^2} \\ &\times \left\{1-2\sum_{r'\neq r}\frac{\sigma_{tsr'}}{\overline{\sigma}+\sigma_{tsr}+\sigma_{tsr'}}+\sum_{r'\neq r}\left(\frac{\sigma_{tsr'}}{\overline{\sigma}+\sigma_{tsr}+\sigma_{tsr'}}\right)^2\right\} \\ &\times \left[1-\sum_{s'\neq s}\sum_{r'}\frac{\sigma_{ts'r'}}{\overline{\sigma}+\sum\sigma_{tsr}}\right] . \end{aligned} \tag{29}$$

Once again, in-sequence and sequence-sequence overlap terms have been factored out.

The next step is to substitute these results back into the fluctuation integrals I_0 and I_1. The integrals over energy and the sums over different resonances in each sequence can be handled as described above for isolated resonances. This procedure will result in three different kinds of integrals. The first kind includes the isolated resonance integrals already considered above

$$\begin{aligned} B_{xs} &= \frac{1}{E_2-E_1}\int_{E_1}^{E_2}\sum_{r}\frac{\sigma_{xsr}}{\overline{\sigma}+\sigma_{tsr}}\,dE \\ &= \frac{1}{D_s}\int P(\alpha)\int_{-\infty}^{\infty}\frac{\sigma_{xs\alpha}(\xi)}{\overline{\sigma}+\sigma_{ts\alpha(\xi)}}\,d\xi\,d\alpha , \end{aligned} \tag{30}$$

and

$$D_{ts} = \frac{1}{E_2 - E_1}\int_{E_1}^{E_2}\sum_r \frac{\overline{\sigma}\sigma_{tsr}}{[\overline{\sigma}+\sigma_{tsr}]^2}\,dE$$
$$= \frac{1}{D_s}\int P(\alpha)\int_{-\infty}^{\infty}\frac{\overline{\sigma}\sigma_{xs\alpha}(\xi)}{[\overline{\sigma}+\sigma_{ts\alpha}(\xi)]^2}\,d\xi\,d\alpha\ . \qquad (31)$$

Note that $D_t \le B_t$, the difference increasing as σ_0 decreases from infinity.

The next kind are the in-sequence overlap integrals. The sum over r' is replaced by integrals over the probabilities of finding each partial width and the probability of finding a resonance r' at a distance η from resonance r.

$$V_{0xs} = \frac{1}{E_2 - E_1}\int_{E_1}^{E_2}\sum_r\sum_{r'\neq r}\frac{\sigma_{xsr}}{\overline{\sigma}+\sigma_{tsr}}\frac{\sigma_{tsr'}}{\overline{\sigma}+\sigma_{tsr}+\sigma_{tsr'}}\,dE$$
$$= \frac{1}{D_s^2}\int P(\alpha)\int P(\beta)\int\int\Omega(\eta)\frac{\sigma_{xs\alpha}(\xi)}{\overline{\sigma}+\sigma_{ts\alpha}(\xi)}$$
$$\frac{\sigma_{ts\beta}(\xi-\eta)}{\overline{\sigma}+\sigma_{ts\alpha}(\xi)+\sigma_{ts\beta}(\xi-\eta)}\,d\eta\,d\xi\,d\beta\,d\alpha\ , \qquad (32)$$

where $\xi = E - E_{sr}$ and $\eta = E_{sr'} - E_{sr}$. Similarly,

$$V_{1ts} = \frac{1}{E_2 - E_1}\int_{E_1}^{E_2}\sum_r\sum_{r'\neq r}\frac{\overline{\sigma}\sigma_{tsr}}{[\overline{\sigma}+\sigma_{tsr}]^2}\Big\{2\frac{\sigma_{tsr'}}{\overline{\sigma}+\sigma_{tsr}+\sigma_{tsr'}}$$
$$-\Big(\frac{\sigma_{tsr'}}{\overline{\sigma}+\sigma_{tsr}+\sigma_{tsr'}}\Big)^2\Big\}\,dE$$
$$= \frac{1}{D_s^2}\int P(\alpha)\int P(\beta)\int\int\Omega(\eta)\frac{\overline{\sigma}\sigma_{ts\alpha}(\xi)}{[\overline{\sigma}+\sigma_{ts\alpha}(\xi)]^2}$$
$$\Big\{2\frac{\sigma_{ts\beta}(\xi-\eta)}{\overline{\sigma}+\sigma_{ts\alpha}(\xi)+\sigma_{ts\beta}(\xi-\eta)}$$
$$-\Big[\frac{\sigma_{ts\beta}(\xi-\eta)}{\overline{\sigma}+\sigma_{ts\alpha}(\xi)+\sigma_{ts\beta}(\xi-\eta)}\Big]^2\Big\}\,d\eta\,d\xi\,d\beta\,d\alpha\ . \qquad (33)$$

The final class of integrals includes the sequence-sequence overlap corrections. They are simplified by noting that the positions of resonances in different spin sequences are uncorrelated. Therefore, $\Omega(\eta)$=1, and the integral of the product reduces to the product of the integrals.

Using the results and definitions from above, the fluctuation integrals become

$$I_{0x} = \sum_s A_{xs}\ , \qquad (34)$$

$$A_{xs} = (B_{xs} - V_{0xs})\Big[1 - \sum_{s'\neq s}A_{ts'}\Big]\ , \qquad (35)$$

and

$$I_{1t} = \sum_s (D_{ts} - V_{1ts}) \Big[1 - \sum_{s' \neq s} A_{ts'} \Big]^2 , \tag{36}$$

where Eq. (35) provides a recursive definition of the A_{ts} for the sequence-sequence corrections as well as the normal value of A_{xs}.

These equations are formally exact for the sequence-sequence overlaps, but in-sequence overlaps only include the interactions between pairs of resonances. Three different approximations to this result are currently in use.

1. The MC²/ETOX Approximation The MC² and ETOX codes use similar approximations to the results above, except that MC² does not include a calculation of the current-weighted total cross section. Both codes explicitly neglect the in-sequence overlap corrections. This approximation was based on the assumption that resonance repulsion would reduce the overlap between resonances in a particular spin sequence, leaving the accidental close spacing of resonances in different sequences as the dominant overlap effect. In addition, both codes stop the recursion of Eq. (35) at $A_t = B_t$. Thus,

$$I_{0x} = \sum_s B_{xs} \Big(1 - \sum_{s' \neq s} B_{ts'} \Big) , \tag{37}$$

and

$$I_{1t} = \sum_s D_{ts} \Big(1 - \sum_{s' \neq s} B_{ts'} \Big)^2 . \tag{38}$$

The equations for the effective cross sections in the MC²/ETOX approximation become

$$\overline{\sigma}_{0x} = b_x + \frac{\overline{\sigma} \sum_s B_{xs} \Big(1 - \sum_{s' \neq s} B_{ts'} \Big)}{1 - \sum_s B_{ts} \Big(1 - \sum_{s' \neq s} B_{ts'} \Big)} , \tag{39}$$

and

$$\overline{\sigma}_{1t} = b_t + \frac{\overline{\sigma} \sum_s D_{ts} \Big(1 - \sum_{s' \neq s} B_{ts'} \Big)^2}{1 - \sum_s B_{ts} \Big(1 - \sum_{s' \neq s} B_{ts'} \Big) - \sum_s D_{ts} \Big(1 - \sum_{s' \neq s} B_{ts'} \Big)^2} , \tag{40}$$

or

$$\overline{\sigma}_{1t} = \overline{\sigma} \left[\frac{1 - \sum_s B_{ts} \Big(1 - \sum_{s' \neq s} B_{ts'} \Big)}{1 - \sum_s B_{ts} \Big(1 - \sum_{s' \neq s} B_{ts'} \Big) - \sum_s D_{ts} \Big(1 - \sum_{s' \neq s} B_{ts'} \Big)^2} \right] - \sigma_0 . \tag{41}$$

These are the equations that are used in the UNRESR module of NJOY. Note that the equation in the ETOX code and report corresponding to Eq. (41) is incorrect. The following equation was used in the ETOX code:

$$\overline{\sigma}_{1t} = \overline{\sigma}\left[\frac{1-\sum_s B_{ts}\left(1-\sum_{s'\neq s} B_{ts'}\right)}{1-\sum_s C_{ts}\left(1-\sum_{s'\neq s} C_{ts'}\right)}\right] - \sigma_0 \,, \tag{42}$$

with $C_{ts} = B_{ts} + D_{ts}$.

2. The MC^2-2 Approximation The MC^2-2 code includes the in-sequence overlap corrections, which the authors found to be more important than previously thought. It uses additional approximations to obtain the equivalent of

$$\overline{\sigma}_{0x} = b_x + \overline{\sigma}\sum_s \frac{B_{xs} - V_{0xs}}{1 - B_{ts} + V_{0ts}} \,. \tag{43}$$

The additional approximations used are

1. Set $A_{ts} = B_{ts} - V_{0ts}$ (first-order sequence-sequence overlap),
2. Neglect the factor $(1 - \sum_{s'\neq s} A_{ts'})$ in the denominator, and
3. Use the approximation $1 - \sum_i f_i \approx \prod_i (1 - f_i)$ on the numerator and denominator.

These simplifications result in a loss of accuracy for the sequence-sequence overlap correction at relatively low values of σ_0. The $\overline{\sigma}_{1t}$ term is not calculated.

3. The UXSR Approximation The experimental UXSR module was developed at Oak Ridge (with some contributions from Los Alamos) based on coding from the Argonne National Laboratory (ANL) in an attempt to include the sophisticated in-sequence overlap corrections from MC^2-2 without approximating the sequence-sequence corrections so badly. It also implemented a calculation of the current-weighted total cross section, which was omitted in MC^2-2. The additional cost of using the full expressions for Eqs. (34) and (36) is minimal, and effective cross sections can be computed for lower values of σ_0 when in-sequence overlap is small (*e.g.*, ^{238}U).

Now that expressions have been chosen for computing the cross sections in terms of the isolated-resonance integrals, it is necessary to select an efficient numerical method for computing them. The resonant parts of the cross sections are given by

$$\sigma_{xsr}(E-E_{sr}) = \left[\sigma_m \frac{\Gamma_x}{\Gamma}\psi(\theta,X)\right]_{sr} , \tag{44}$$

and

$$\sigma_{tsr}(E-E_{sr}) = \left[\sigma_m\{\cos 2\phi_\ell\,\psi(\theta,X) + \sin 2\phi_\ell\,\chi(\theta,X)\}\right]_{sr} , \tag{45}$$

where x takes on the values γ, f, or c for capture, fission, or competition, and

$$\sigma_m = \frac{4\pi g_J}{k^2}\frac{\Gamma_n}{\Gamma} , \tag{46}$$

$$\theta = \sqrt{\frac{A}{4kTE_0}}\,\Gamma , \tag{47}$$

$$X = \frac{2(E-E_0)}{\Gamma} , \tag{48}$$

$$g_J = \frac{2J+1}{2(2I+1)} , \text{ and} \tag{49}$$

$$k = 2.196771\times 10^{-3}\frac{A}{1+A}\sqrt{E} . \tag{50}$$

The functions ψ and χ are the symmetric and antisymmetric components of the broadened resonance line shape:

$$\psi(\theta,X) = \frac{\theta\sqrt{\pi}}{2}\mathrm{Re}W\left(\frac{\theta X}{2},\frac{\theta}{2}\right) , \tag{51}$$

and

$$\chi(\theta,X) = \theta\sqrt{\pi}\,\mathrm{Im}W\left(\frac{\theta X}{2},\frac{\theta}{2}\right) , \tag{52}$$

where

$$W(x,y) = \exp[-(x+iy)^2]\,\mathrm{erfc}[-i(x+iy)] \tag{53}$$

is the complex probability integral. The methods for computing ψ and χ are well known (see QUIKW).

The first integral needed is

$$\begin{aligned} B_{xs} &= \frac{1}{D_s}\int P(\alpha)\int \frac{\sigma_{xs\alpha}(\xi)}{\overline{\sigma}+\sigma_{ts\alpha}(\xi)}\,d\xi\,d\alpha \\ &= \frac{1}{D_s}\int P(\alpha)\int \frac{\sigma_m(\Gamma_x/\Gamma)\psi(\theta,X)}{\overline{\sigma}+\sigma_m\{\cos 2\phi_\ell\,\psi(\theta,X)+\sin 2\phi_\ell\,\chi(\theta,X)\}}\,d\xi\,d\alpha \\ &= \frac{1}{D_s}\int P(\alpha)\frac{\Gamma_x}{2\cos 2\phi_\ell}\int \frac{\psi(\theta,X)}{\beta+\psi(\theta,X)+\tan 2\phi_\ell\,\chi(\theta,X)}\,dX\,d\alpha , \end{aligned} \tag{54}$$

where

$$\beta = \frac{\overline{\sigma}}{\sigma_m \cos 2\phi_\ell} \ . \tag{55}$$

The second integral needed is

$$\begin{aligned} B_{ts} &= \frac{1}{D_s} \int P(\alpha) \int \frac{\sigma_{ts\alpha}(\xi)}{\overline{\sigma} + \sigma_{ts\alpha}\xi)} \, d\xi \, d\alpha \\ &= \frac{1}{D_s} \int P(\alpha) \frac{\Gamma}{2} \int \frac{\psi(\theta, X) + \tan 2\phi_\ell \, \chi(\theta, X)}{\beta + \psi(\theta, X) + \tan 2\phi_\ell \, \chi(\theta, X)} \, dX \, d\alpha \ . \end{aligned} \tag{56}$$

Both of these integrals can be expressed in terms of the basic J integral:

$$\begin{aligned} B_{xs} &= \frac{1}{D_s} \int P(\alpha) \frac{\Gamma}{\cos 2\phi_\ell} J(\beta, \theta, \tan 2\phi_\ell, 0) \, d\alpha \ , \text{ and} \\ B_{ts} &= \frac{1}{D_s} \int P(\alpha) \, \Gamma \, J(\beta, \theta, \tan 2\phi_\ell, \tan 2\phi_\ell) \, d\alpha \ , \end{aligned} \tag{57}$$

where

$$J(\beta, \theta, a, b) = \frac{1}{2} \int_{-\infty}^{\infty} \frac{\psi(\theta, X) + b \, \chi(\theta, X)}{\beta + \psi(\theta, X) + a \, \chi(\theta, X)} \, dX \ . \tag{58}$$

The D integral can be handled in the same way, but only total reaction is required.

$$\begin{aligned} D_{ts} &= \frac{1}{D_s} \int P(\alpha) \int \frac{\overline{\sigma} \sigma_{ts\alpha}(\xi)}{[\overline{\sigma} + \sigma_{ts\alpha}(\xi)]^2} \, d\xi \, d\alpha \\ &= \frac{1}{D_s} \int P(\alpha) \frac{\Gamma}{2} \int \frac{\beta\psi(\theta, X) + \tan 2\phi_\ell \, \chi(\theta, X)}{[\beta + \psi(\theta, X) + \tan 2\phi_\ell \, \chi(\theta, X)]^2} \, dX \, d\alpha \\ &= \frac{1}{D_s} \int P(\alpha) \, \Gamma \, K(\beta, \theta, \tan 2\phi_\ell, \tan 2\phi_\ell) \, d\alpha \ , \end{aligned} \tag{59}$$

where

$$K(\beta, \theta, a, b) = \frac{1}{2} \int_{-\infty}^{\infty} \frac{\beta[\psi(\theta, X) + b \, \chi(\theta, X)]}{[\beta + \psi(\theta, X) + a \, \chi(\theta, X)]^2} \, dX \ . \tag{60}$$

A method for computing J, including the interference effects, has been developed by Hwang for MC^2-2.[5] However, this method was not available in the days when MC^2 and ETOX were developed. Therefore, UNRESR uses only $J(\beta, \theta, 0, 0)$ and $K(\beta, \theta, 0, 0)$ in computing the isolated-resonance fluctuation integrals. A direct integration is used over most of the X range, but the part of the integral arising from large X is handled using analytic integrations of the asymptotic forms of the arguments (see AJKU).

The final step is to do the n-fold integration over the probability distributions for the resonance widths. This integration has been abbreviated as a single integration over α in the above equations. The method used was originally developed for MC^2-2 and is based on Gauss-Jacobi quadratures. A set of 10 quadrature

points and weights is provided for each of the χ^2 probability distributions with 1 through 4 degrees of freedom. These quadratures convert the n-fold integral into an n-fold summation. The value of n can be as large as 4 when Γ_n, Γ_f, Γ_γ, and Γ_c (competitive width) are all present.

Although UNRESR neglects the effects of overlap between resonances in the same spin sequence and the effects of interference in the elastic and total cross sections, it still gives reasonable results for the background cross section values needed for most practical problems. Modern evaluations are steadily reducing the need for accurate unresolved calculations by extending the resolved resonance range to higher and higher energies. Ultimately, UNRESR should be replaced by UXSR when that approach is fully developed and tested. Another alternative is to generate self-shielded effective cross sections from ladders of resonances chosen statistically (see PURR).

B. Implementation

In implementing this theory in UNRESR, there are special considerations involving the choice of an incident energy grid, what to do if the unresolved range overlaps the resolved range or the range of smooth cross sections, the choice of the σ_0 grid, how to interpolate on σ_0, and how to communicate the results to other modules.

1. Choice of Energy Grid. The same logic is used to choose the incident energy grid in UNRESR and RECONR. It is complicated, because of the several different representations available for unresolved data, and because of the existence of evaluations that have been carried over from previous versions of ENDF/B. Unresolved-resonance range self-shielded cross sections are expensive to compute; therefore, some effort has been made to avoid using any more energy grid points than are necessary.

For evaluations that give energy-independent unresolved-resonance parameters, there is still an energy dependence to the cross sections. Because this dependence is normally somewhere between constant and a $1/v$ law, a fairly coarse grid with about 10 points per decade should be sufficient to allow the cross sections to be computed reliably using linear-linear interpolation.

If the evaluation uses energy-dependent parameters, the normal reaction would be to use the energies that were provided by the evaluator and to obtain intermediate cross sections by interpolation. Unfortunately, some of the evaluations carried over from earlier days contain some energy intervals that are quite large

(for example, steps by factors of 10). The evaluators for these isotopes assumed that the user would use parameter interpolation and compute the cross sections at a number of intermediate energies in these long steps. UNRESR will detect such evaluations and add additional energy points in the large energy steps using an algorithm similar to the one used for the cases with energy-independent parameters.

The final energy grid can be observed by scanning the printed output from UNRESR.

2. Resolved-Unresolved Overlap. Elemental evaluations include separate energy ranges in MF2/MT151 for each of the isotopes of the element, and these energy ranges do not have to be the same for each isotope. This means that the lower end of an unresolved range can easily overlap the resolved range from another isotope, and the upper end of the unresolved range for an isotope can overlap the smooth range of another isotope. These overlap regions are detected by UNRESR as the resonance data are read in, and they are marked by making the sign of the incident energy value negative.

3. Choosing a σ_0 Grid. There are two factors to consider, namely, choosing values that will represent the shape adequately, and limiting the range of σ_0 to the region where the theory is valid. The $\sigma_x(\sigma_0)$ curves start out decreasing from the infinite dilution value as $1/\sigma_0$ as σ_0 decreases from infinity (1×10^{10} in the code). The curve eventually goes through an inflection point at some characteristic value of σ_0, becomes concave upward, and approaches a limiting value at small σ_0 that is smaller than the infinite-dilution value. Decade steps are often used, but the user should try to select values that include the inflection point and not waste values on the $1/\sigma_0$ region. Half-decade values are useful near the inflection point (*e.g.*, 100, 300, 1000 for ^{235}U). The grid interval chosen should be consistent with the interpolation method used (see below).

Choosing the lower limit for σ_0 is a more difficult problem. As shown in the theory section, resonance overlap effects are developed as a series in $1/\sigma_0$, and the series is truncated after only one step of recursion in Eq. (35). This means that the overlap correction should be most accurate for large σ_0 and gradually get worse as σ_0 decreases. Experience shows that the correction can actually get large enough to produce negative cross sections for small σ_0. (This problem can also show up as a failure in interpolation when a log scheme has been selected.) For isotopes that have relatively narrow resonances spaced relatively widely, such as

^{238}U, UNRESR gives reasonable results to σ_0 values as low as 0.1. For materials with stronger overlap, such as ^{235}U, a lower limit around 100 is more reasonable. A few of the heavy actinides in ENDF/B-VI have evaluations that break down for σ_0 values lower than 1000. This problem is not too serious in practice. The fertile materials, which appear in large concentrations in reactors, allow the necessary small values of σ_0. The fissile materials have to be more dilute, and the larger σ_0 limit needed for them is not usually a problem.

The UXSR approximation discussed above allows one to reach somewhat smaller σ_0 values.

4. Interpolating on σ_0. It turns out that these functions are difficult to interpolate because they have a limited radius of convergence. Although approximate schemes have been developed based on using functions of similar shape such as the tanh function,[6] better results can be obtained by using different interpolation schemes for the low- and high-σ_0 ranges. The TRANSX-CTR code[7] used interpolation in $1/\sigma_0$ for high σ_0, Lagrangian interpolation of $\ln\sigma_x$ *vs* $\ln\sigma_0$, for intermediate values, and a σ_0^2 extrapolation for very low σ_0. Unfortunately, schemes like this sometimes give ridiculous interpolation excursions when the polynomials are not suitable. Therefore, TRANSX-2[8] has had to revert to using simple linear interpolation, which is always bounded and predictable, but which requires relatively fine σ_0 grids.

5. Communicating Results to Other Modules. NJOY tries to use ENDF-like files for all communications between the different calculational modules. Because the unresolved effective cross sections were originally derived from the resonance parameters in File 2, it seemed reasonable to create a new section in File 2 to carry the unresolved cross sections onto other modules, and a special MT number of 152 was selected for this purpose. RECONR creates an MT152 that contains only the infinitely-dilute unresolved cross sections. UNRESR overwrites it with self-shielded unresolved cross sections. GROUPR can then use these cross sections in its calculation of the multigroup constants. The format used for MT152 is given below using the conventional ENDF style.

```
[MAT,2,152/ ZA, AWR, LSSF, 0, 0, INTUNR ] HEAD
[MAT,2,152/ TEMZ, 0, NREAC, NSIGZ, NW, NUNR/
            SIGZ(1), SIGZ(2),...,SIGZ(NSIGZ),
            EUNR(1),
            SIGU(1,1,1), SIGU(1,2,1),...,SIGU(1,NSIGZ,1),
            SIGU(2,1,1),...
            ...
```

```
SIGU(NREAC,1,1),...,SIGU(NREAC,NSIGZ,1),
EUNR(2),...
  <continue for energies through EUNR(NUNR)>
...SIGU(NREAC,NSIGZ,NUNR) ] LIST
```

where NREAC is always 5 (for the total, elastic, fission, capture, and current-weighted total reactions, in that order), NSIGZ is the number of σ_0 values, NUNR is the number of unresolved energy grid points, and NW is given by

```
NW=NSIGZ+NUNR*(1+5*NSIGZ) .
```

C. User Input

The following summary of the user input instructions was copied from the comment cards at the beginning of the UNRESR module in the NJOY 91.0 source file.

```
*---INPUT SPECIFICATIONS (FREE FORMAT)-------------------------*
*                                                              *
* CARD 1                                                       *
*   NENDF   UNIT FOR ENDF/B TAPE                               *
*   NIN     UNIT FOR INPUT PENDF TAPE                          *
*   NOUT    UNIT FOR OUTPUT PENDF TAPE                         *
* CARD 2                                                       *
*   MATD    MATERIAL TO BE PROCESSED                           *
*   NTEMP   NO. OF TEMPERATURES (MAX. NO. ALLOWED=9)           *
*   NSIGZ   NO. OF SIGMA ZEROES (MAX. NO. ALLOWED=8)           *
*   IPRINT  PRINT OPTION (0=MIN, 1=MAX) (DEFAULT=0)            *
* CARD 3                                                       *
*   TEMP    TEMPERATURES IN KELVIN (INCLUDING ZERO)            *
* CARD 4                                                       *
*   SIGZ    SIGMA ZERO VALUES (INCLUDING INFINITY)             *
*       CARDS 2, 3, 4 MUST BE INPUT FOR EACH MATERIAL DESIRED  *
*       MATD=0/ TERMINATES EXECUTION OF UNRESR.                *
*                                                              *
****************************************************************
```

Card 1, as usual, specifies the input and output units for the module. The input PENDF tape on NIN must have been processed through RECONR and BROADR. It will contain default versions of the special unresolved section with MF=2 and MT=152 that gives the infinitely-dilute unresolved cross sections for each temperature. The output PENDF file NOUT will contain revised sections MF2,MT152 that give effective cross sections *vs* σ_0 for all temperatures.

Card 2 is used to specify the material desired (MATD), the number of temperatures and background cross sections desired (NTEMP and NSIGZ), and the print option (IPRINT). The actual temperature and σ_0 values are given on Cards 3 and

4. Temperatures should be chosen to be adequate for the planned applications. The temperature dependence of the effective cross sections is usually monotonic and fairly smooth. Polynomial interpolation schemes using T is often used, and roughly uniform spacing for the temperature grid (or spacing that increases modestly as T increases) is usually suitable.

The choice of a grid for σ_0 is more difficult. The curves of cross section versus σ_0 have an inflection point, and it is important to choose the grid to represent the inflection point fairly well. A log spacing for σ_0 is recommended. Very small values of σ_0 should not be used. These considerations are discussed in more detail in the "Implementation" section above. Unfortunately, the best choice for a grid can only be found by trial and error.

D. Output Example

The following portion of UNRESR output is for ^{238}U from ENDF/B-VI.

```
**************************************************************************************

UNRESR...CALCULATION OF UNRESOLVED RESONANCE CROSS SECTIONS                 1494.1S

                                                              STORAGE   8/ 20000

UNIT FOR INPUT ENDF/B TAPE ..................    -21
UNIT FOR INPUT PENDF TAPE ...................    -23
UNIT FOR OUTPUT PENDF TAPE ..................    -24
TEMPERATURES ............................. 3.00E+02
                                           4.00E+02
                                             ...
SIGMA ZERO VALUES ........................ 1.00E+10
                                           1.00E+04
                                             ...
                                           3.00E+01
                                           1.00E+01
                                           3.00E+00
                                           1.00E+00
PRINT OPTION (0 MIN., 1 MAX.) .............       1

MAT = 9237    TEMP =  3.00E+02                                              1494.1S
ENERGY =  1.0000E+04
  1.545E+01  1.544E+01 ... 1.401E+01  1.348E+01  1.314E+01  1.302E+01
  1.474E+01  1.472E+01 ... 1.337E+01  1.289E+01  1.259E+01  1.248E+01
  0.000E+00  0.000E+00 ... 0.000E+00  0.000E+00  0.000E+00  0.000E+00
  7.154E-01  7.149E-01 ... 6.372E-01  5.925E-01  5.576E-01  5.426E-01
  1.545E+01  1.543E+01 ... 1.323E+01  1.264E+01  1.231E+01  1.219E+01
ENERGY =  1.2000E+04
  1.512E+01  1.511E+01 ... 1.393E+01  1.344E+01  1.313E+01  1.301E+01
  1.445E+01  1.444E+01 ... 1.332E+01  1.288E+01  1.259E+01  1.248E+01
  0.000E+00  0.000E+00 ... 0.000E+00  0.000E+00  0.000E+00  0.000E+00
```

```
  6.634E-01  6.631E-01 ... 6.027E-01  5.651E-01  5.348E-01  5.215E-01
  1.512E+01  1.510E+01 ... 1.323E+01  1.266E+01  1.233E+01  1.221E+01
 ...

ENERGY =  1.4903E+05
  1.139E+01  1.139E+01 ... 1.137E+01  1.135E+01  1.129E+01  1.124E+01
  1.125E+01  1.125E+01 ... 1.124E+01  1.121E+01  1.115E+01  1.111E+01
  0.000E+00  0.000E+00 ... 0.000E+00  0.000E+00  0.000E+00  0.000E+00
  1.334E-01  1.334E-01 ... 1.346E-01  1.347E-01  1.332E-01  1.317E-01
  1.213E+01  1.213E+01 ... 1.211E+01  1.203E+01  1.183E+01  1.166E+01
GENERATED CROSS SECTIONS AT 31 POINTS                              1507.8S

MAT = 9237    TEMP =  4.00E+02                                     1507.8S
ENERGY =  1.0000E+04
  1.545E+01  1.544E+01 ... 1.449E+01  1.404E+01  1.373E+01  1.361E+01
  1.474E+01  1.473E+01 ... 1.382E+01  1.340E+01  1.311E+01  1.300E+01
  0.000E+00  0.000E+00 ... 0.000E+00  0.000E+00  0.000E+00  0.000E+00
  7.154E-01  7.152E-01 ... 6.759E-01  6.485E-01  6.247E-01  6.138E-01
  1.545E+01  1.544E+01 ... 1.387E+01  1.328E+01  1.291E+01  1.277E+01
...
GENERATED CROSS SECTIONS AT 31 POINTS                              1523.6S

                      ...                          USAGE  11118/ 20000

                                                                   1523.6S
********************************************************************************
```

The display of the cross section table for each energy has σ_0 reading across (in decreasing order) and reaction type reading down (in the order of total, elastic, fission, capture, and current-weighted total). Two σ_0 values were removed from the table to make it narrower for this report.

E. Coding Details

The code starts by reading in the user's input and output unit numbers and opening the files that will be needed during the UNRESR run. After printing the introductory timer line, the storage system is initialized and space is reserved for an array SCR, which will be used for reading in ENDF records. As is usual for NJOY, SCR is stored in the common container array A starting at the pointer ISCR. UNRESR now prints out the user's unit numbers on the output listing and calls UWTAB to prepare internal tables used by UW to compute the real and imaginary parts of the complex probability integral.

The next step is to read in the TAPEID records of the ENDF and PENDF tapes. The loop over materials starts at statement number 110 by reading in the user's values for the ENDF MAT number, number of temperatures NTEMP, number of σ_0 values NSIGZ, and print flag IPRINT. If this is not the end of the material loop

(MAT=0), the actual values of the temperatures and background cross sections are read into the arrays TEMP and SIGZ. The input quantities are echoed to the output listing to help detect input errors.

The code then begins a loop over the requested temperatures. It writes the current values of material ID, temperature, and time on the output listing, it reads the resonance parameters from the section with MT=151 in File 2 of the ENDF tape using RDUNF2, and it reads the background cross sections from File 3 for each of the resonance reactions using RDUNF3. It also starts to build an output record to be written in MT=152 on the new PENDF tape in the array starting at IB.

The next loop is over all the energy grid points at this temperature. The grid points were determined in RDUNF2, and the NUNR points are stored in the array EUNR. The background cross sections are stored in an array SB(IE,IX) starting at pointer ISB. The energy index takes on NUNR values, and the reaction index IX takes on four values. The calculation of the actual effective cross sections takes place in UNRESL. The results for each energy appear in the array SIGU(IX,IS), where IS takes on NSIGZ values. Each UNRESL array is stored into the accumulating output array B and printed on the output listing. When the energy loop is complete, the unused part of the array B is released.

At this point, UNRESR checks to see if there is a previous version of MT152 on the PENDF tape. If so, these new data will replace it. If not, a new section with MT=152 will be created. In either case, a new section MT451 in File 1 is generated containing the current temperature and the correct entry in the directory for the PENDF tape. Finally, the new MT152 for this temperature is copied onto the output tape from the B array, and the rest of the contents of this temperature on the old PENDF tape are copied to the new PENDF tape. After writing a report on the number of resonance points generated and the amount of computer time used, UNRESR branches back to continue the temperature loop. When the last temperature has been processed, the code closes the files, writes a final report on the output listing, and terminates.

Note that UNRESR takes special branches for materials with no resonance parameters or materials with no unresolved parameters. Therefore, the user can freely request an UNRESR run even when it is not appropriate, and no damage will be done.

The subroutine RDUNF2 is used to read in the unresolved resonance parameters from File 2 of the input ENDF tape. It is very similar to the corresponding coding in RECONR. The array starting at ISCR is used to read in ENDF record, the resonance parameter data are stored in the array starting at IARRY, and the final

grid of energy values is stored in EUNR. Note that RDUNF2 will add extra energy nodes for evaluations with energy-independent parameters or for energy-dependent evaluations that have energy steps larger than the factor WIDE, which is currently set to 3.0. The subroutine also discovers resolved-unresolved or unresolved-smooth overlap, flags those energy values, and prints messages to the user on the output listing.

Subroutine ILIST is used to insert a new energy into a list of energies in ascending order. It is used to manage the accumulation of the grid of energy nodes.

Subroutine RDUNF3 is used to read in the background cross sections in the unresolved range from File 3. The unresolved energy grid determined by RDUNF2 is used for the background cross sections.

The main calculation of the effective cross sections for the unresolved range is performed in subroutine UNRESL. The calculation is done in two passes: first, the potential scattering cross section is computed; second, the unresolved cross sections are computed. The passes are controlled by the parameter ISPOT. In both cases, resonance parameter data stored into the main container array A by RDUNF2 is used. The coding is similar to that used in RECONR down to the point where the ETOX statistical averages are computed. The three loops ending at statement number 420 carry out the n-fold quadrature represented as integrals over α in the text. They account for variations in the fission width, neutron width, and competitive width. The capture width is assumed to be constant. Note that AJKU is called in the innermost loop to compute the J and K integrals in XJ and XK, respectively. The K integral returned by this routine is actually $J + K$ in our notation. The final quantities are computed in the loops that end at statement number 460. Note that TK is equivalent to $B_{ts} + D_{ts}$ in our notation. Similarly, TL is equivalent to B_{xs}, and TJ is equivalent to B_{ts}. The last step in this subroutine is to compute the average cross sections by summing over spin sequences. The loop ending at statement 530 computes XJ as $\sum_s B_{ts}$ and XK $\sum_s C_{ts}$. With these quantities available, it is easy to finish the calculation of the effective cross sections as given by Eq. (39) and Eq. (41).

The subroutines UUNFAC, INTRF, and INTR are similar to corresponding routines from RECONR and don't require further discussion here.

Subroutine AJKU is used to compute the J and K integrals without interference corrections.

The subroutines QUIKW, UWTAB, and UW implement a package for computing the complex probability integral efficiently. It was originally developed at ANL

for the MC^2 code. When it is used, a pair of 62 × 62 tables for the real and imaginary parts of the complex probability integral are precomputed using **UWTAB** and **UW**. Values of W for small arguments are obtained by interpolating in these precomputed tables. Values of W for larger arguments are obtained using various asymptotic formulas.

F. Error Messages

ERROR IN RECONR*ILLEGAL NSUB FOR RECONR**

RECONR only processes sublibraries that contain cross section data. Check whether the right ENDF input tape was mounted.

ERROR IN UNRESR*MODE CONVERSION BETWEEN NIN AND NOUT**

Input and output files must both be in ASCII mode (positive unit numbers), or they must both be in binary mode (negative unit numbers).

ERROR IN UNRESR*TOO MANY TEMPERATURES**

The number of temperatures is limited to 10 (this limit holds for all versions after 91.14).

ERROR IN UNRESR*TOO MANY SIGMA ZEROES**

The number of σ_0 values is limited to 10 (this limit holds for all versions after 91.14).

ERROR IN UNRESR*STORAGE EXCEEDED**

It will be necessary to reduce the total number of temperature and σ_0 values requested, or to increase the size of the main container array **A(20000)** in common **USTORE** and the size limit **DATA NAMAX/2000/.**

ERROR IN RDUNF2*STORAGE EXCEEDED**

It will be necessary to increase the size of the main container array **A(20000)** in common **USTORE** and the size limit **DATA NAMAX/2000/.**

ERROR IN UNRESL*STORAGE EXCEEDED**

The code is currently limited to 30 spin sequences. For more spin sequences, it will be necessary to increase the dimensions of several arrays in this subroutine.

G. References

1. P. F. Rose and C. L. Dunford, Eds., "ENDF-102, Data Formats and Procedures for the Evaluated Nuclear Data File, ENDF-6," available from the National Nuclear Data Center, Brookhaven National Laboratory, as a limited distribution informal report BNL-NCS-44945 (July 1990).

2. I. I. Bondarenko, Ed., *Group Constants for Nuclear Reactor Calculations* (Consultants Bureau, New York, 1964).

3. B. J. Toppel, A. L. Rago, and D. M. O'Shea, "MC^2, A Code to Calculate Multigroup Cross Sections," Argonne National Laboratory report ANL-7318 (June 1967).

4. R. E. Schenter, J. L. Baker, and R. B. Kidman, "ETOX, A Code to Calculate Group Constants for Nuclear Reactor Calculations," Battelle Northwest Laboratory report BNWL-1002 (1962).

5. H. Henryson II, B. J. Toppel, and C. G. Stenberg, "MC^2-2: A Code to Calculate Fast Neutron Spectra and Multigroup Cross Sections," Argonne National Laboratory report ANL-8144 (ENDF 239) (June 1976).

6. R. B. Kidman, "Cross Section Structure Factor Interpolation Schemes," Hanford Engineering Development Laboratory report HEDL-TME-71-40 (1971).

7. R. E. MacFarlane, "TRANSX-CTR: A Code for Interfacing MATXS Cross-Section Libraries to Nuclear Transport Codes for Fusion Systems Analysis," Los Alamos National Laboratory report LA-9863-MS (February 1984).

8. R. E. MacFarlane, "TRANSX 2: A Code for Interfacing MATXS Cross-Section Libraries to Nuclear Transport Codes," Los Alamos National Laboratory report LA-12312-MS (July 1992).

VI. HEATR

The HEATR module generates pointwise heat production cross sections and radiation damage energy production for specified reactions and adds them to an existing PENDF tape. The heating and damage numbers can then be easily group averaged, plotted, or reformatted for other purposes. An option of use to evaluators checks ENDF/B files for neutron/photon energy-balance consistency. The advantages of HEATR include

- Heating and damage are computed in a consistent way.
- All ENDF/B neutron and photon data are used.
- ENDF/-6 charged-particle distributions are used when available.
- Kinematic checks are available to improve future evaluations.
- Both energy-balance and kinematic KERMA factors can be produced.

This chapter describes HEATR in version 91.91.

A. Theory of Nuclear Heating

Heating is an important parameter of any nuclear system. It may represent the product being sold—as in a power reactor—or it may affect the design of peripheral systems such as shields and structural components.

Nuclear heating can be conveniently divided into neutron heating and photon heating (see Fig. 1). The neutron heating at a given location is proportional to the local neutron flux and arises from the kinetic energy of the charged products of a neutron induced reaction (including both charged secondary particles and the recoil nucleus itself). Similarly, the photon heating is proportional to the flux of secondary photons transported from the site of previous neutron reactions. It is also traceable to the kinetic energy of charged particles (for example, electron–positron pairs and recoil induced by photoelectric capture).

Heating, therefore, is often described by the KERMA[1] (Kinetic Energy Release in Materials) factors $k_{ij}(E)$ defined such that the heating rate in a mixture is given by

$$H(E) = \sum_i \sum_j \rho_i k_{ij}(E) \phi(E) \,, \tag{1}$$

where ρ_i is the number density of material i, $k_{ij}(E)$ is the KERMA factor for material i and reaction j at incident energy E, and $\phi(E)$ is the neutron or photon scalar flux at E. KERMA is used just like a microscopic reaction cross section except that its units are energy $\times$ cross section (eV-barns for HEATR).

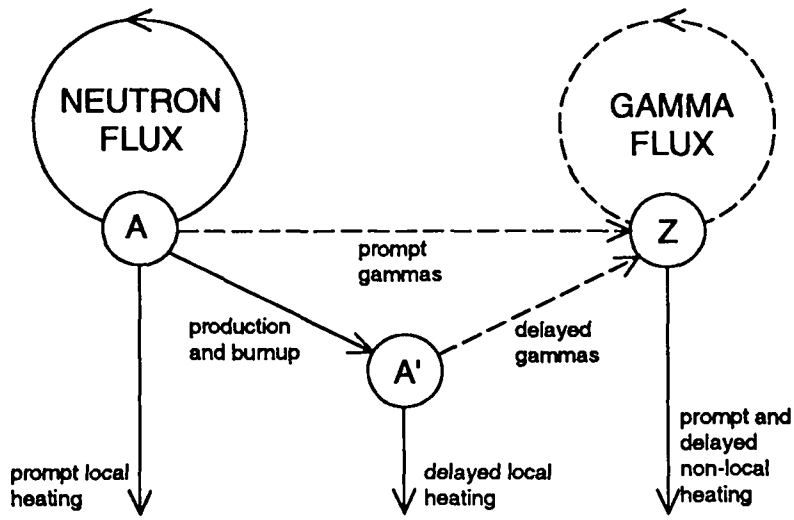


Figure 1: Components of nuclear heating. HEATR treats the prompt local neutron heating only. Gamma heating is computed in GAMINR. Delayed local heating and photon production are not treated by NJOY, and they must be added at a later stage.

The "direct method" for computing the KERMA factor is

$$k_{ij}(E) = \sum_{\ell} \overline{E}_{ij\ell}(E)\sigma_{ij}(E) \ , \tag{2}$$

where the sum is carried out over all charged products of the reaction including the recoil nucleus, and $\overline{E}_{ij\ell}$ is the total kinetic energy carried away by the ℓth species of secondary particle. These kinds of data are now becoming available for some materials with the advent of ENDF/B-VI, but earlier ENDF/B versions did not include the detailed spectral information needed to evaluate Eq. (2).

For this reason, NJOY computes KERMA factors for many materials by the "energy-balance method."[2] The energy allocated to neutrons and photons is simply subtracted from the available energy to obtain the energy carried away by charged particles:

$$k_{ij}(E) = \left(E + Q_{ij} - \overline{E}_{ijn} - \overline{E}_{ij\gamma}\right) \sigma_{ij}(E) \ , \tag{3}$$

where Q_{ij} is the mass-difference Q value for material i and reaction j, $\overline{E}_n$ is the total energy of secondary neutrons including multiplicity, and $\overline{E}_\gamma$ is the energy of secondary photons including photon yields.

This method is well suited for use with ENDF/B-V, which contains neutron and

photon spectral data, but not the particle spectra required by the direct method. The disadvantage of this method is that the KERMA factor sometimes depends on a difference between large numbers. In order to obtain accurate results, extreme care must be taken with the evaluation to ensure that photon and neutron yields and average energies are consistent. In fact, the lack of consistency in ENDF/B-V often reveals itself as negative KERMA factors.[3]

However, a negative KERMA factor is not always the defect it seems to be. It must be remembered that heating has both neutron and photon components. A negative KERMA might indicate that too much energy has been included with the photon production in the evaluation. This will result in excessive photon heating if most of the photons stay in the system. However, the negative KERMA will have just the right magnitude to cancel this excess heating. The energy-balance method guarantees conservation of total energy in large homogeneous systems.

In this context, large and homogeneous means that most neutrons and photons stay in their source regions. It is clear that energy-balance errors in the evaluation affect the spatial distribution of heat and not the total system heating when the energy-balance method is employed.

A final problem with the energy-balance method occurs for the elemental evaluations in ENDF/B. Isotopic Q values and cross sections are not available in the files. It will usually be possible to define adequate cross sections, yields, and spectra for the element. However, it is clear that the available energy should be computed with an effective Q given by

$$\overline{Q} = \frac{\sum_i \rho_i \sigma_i Q_i}{\sum_i \rho_i \sigma_i} , \tag{4}$$

where ρ_i is the atomic fraction of isotope i in the element. This number is energy dependent and can be represented only approximately by the single constant Q allowed in ENDF/B. HEATR allows the user to input an auxiliary energy-dependent Q for elements.

For elastic and discrete-level inelastic scattering, the neutron KERMA factor can be evaluated directly without reference to photon data. For other reactions, conservation of momentum and energy can be used to estimate the KERMA or to compute minimum and maximum limits for the heating. HEATR includes an option that tests the energy-balance KERMA factors against these kinematic limits, thereby providing a valuable test of the neutron–photon consistency of the evaluation. If the energy-balance heating numbers for a particular isotope

should fail these tests, and if the isotope is important for a "small" system, an improved evaluation is probably required. The alternative of making *ad hoc* fixes to improve the local heat production is dangerous because the faults in the neutron and/or photon data revealed by the tests may lead to significant errors in neutron transport and/or photon dose and nonlocal energy deposition.

In practice, an exception to this conclusion must be made for the radiative capture reaction (n,γ). The difference between the available energy E+Q and the total energy of the emitted photons is such a small fraction of E+Q that it is difficult to hold enough precision to get reasonable recoil energies. Moreover, the emitted photons cause a component of recoil whose effect is not normally included in evaluated capture spectra. Finally, the "element problem" cited above is especially troublesome for capture because the available energy may change by several MeV between energies dominated by resonances in different isotopes of the element, giving rise to many negative or absurdly large heating numbers. These problems are more important for damage calculations (see below) where the entire effect comes from recoil and the compensation provided by later deposition of the photon energy is absent.

For these reasons, HEATR estimates the recoil due to radiative capture using conservation of momentum. The recoil is the vector sum of the "kick" caused by the incident neutron and the kicks due to the emission of all subsequent photons. Assuming that all photon emission is isotropic and that the directions of photon emission are uncorrelated, the photon component of recoil depends on the average of E_γ^2 over the entire photon spectrum

$$E_R = \frac{E}{A+1} + \frac{\overline{E_\gamma^2}}{2(A+1)mc^2} , \tag{5}$$

where mc^2 is the neutron mass energy. The second term is important below 25–100 keV. This formula gives an estimate that works for both isotopes and elements and has no precision problems. However, it does not explicitly conserve energy, and isotopes with bad capture photon data can still cause problems.

B. Theory of Damage Energy

Damage to materials caused by neutron irradiation is an important design consideration in fission reactors and is expected to be an even more important problem in fusion power systems. There are many radiation effects that may cause damage, for example, direct heating, gas production (*e.g.*, helium embrittlement), and the production of lattice defects.

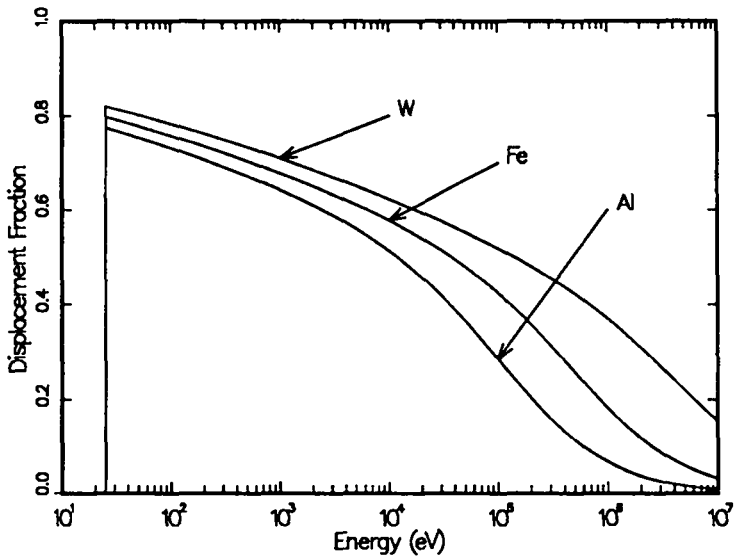


Figure 2: Examples of the portion of the primary recoil energy that is available to cause lattice displacements in metallic lattices. The remaining energy leads to electronic excitation. The quantity plotted is $P(E)$ from Eq. (36) divided by E. The 25 eV cutoff is also discussed in connection with Eq. (36).

A large cluster of lattice defects can be produced by the primary recoil nucleus of a nuclear reaction as it slows down in a lattice. It has been shown that there is an empirical correlation between the number of displaced atoms (DPA, displacements per atom) and various properties of metals, such as elasticity. The number of displaced atoms depends on the total available energy E_a and the energy required to displace an atom from its lattice position E_d. Since the available energy is used up by producing pairs,

$$\text{DPA} = \frac{E_a}{2E_d} \ . \tag{6}$$

The values of E_d used in practice are chosen to represent the empirical correlations, and a wide range of values is found in the literature[4, 5, 6] (see Table 1 for some examples). The energy available to cause displacements is what HEATR calculates. It depend on the recoil spectrum and the partition of recoil energy between electronic excitations and atomic motion. The partition function used was given by Robinson[7] based on the electronic screening theory of Lindhard[8] (see Fig. 2).

The damage output from HEATR is the damage energy production cross section (eV-barns). As in Eq. (1), multiplying by the density and flux gives eV/s. Dividing by $2E_d$ gives displacements/s.

Table 1: Typical Values for the Atomic Displacement Energy Needed to Compute DPA.[6]

Element	E_d in eV	Element	E_d in eV
Be	31	Co	40
C	31	Ni	40
Mg	25	Cu	40
Al	27	Zr	40
Si	25	Nb	40
Ca	40	Mo	60
Ti	40	Ag	60
V	40	Ta	90
Cr	40	W	90
Mn	40	Au	30
Fe	40	Pb	25

C. Computation of KERMA Factors By Energy Balance

The ENDF/B files do not usually give photon production data for all partial reactions. Summation reactions such as nonelastic (MT=3) and inelastic (MT=4) are often used. It is still possible to compute partial KERMA factors for those summation reactions by reordering Eq. (3) as follows:

$$k_{ij} = \sum_{j \in J} k_{ij}^{n}(E) - \sum_{\ell \in J} \overline{E}_{i\ell\gamma}\sigma_{i\ell}(E) \; , \tag{7}$$

where j runs over all neutron partials contained in J, and ℓ runs over all photon partials in J. The total KERMA is well defined, but partial KERMAS should be used only with caution.

HEATR loops through all the neutron reactions on the ENDF/B tape. If energy balance is to be used, it computes the neutron contributions needed for the first term. These are

$$k_{ij}^{n}(E) = \Big[E + Q_{ij} - \overline{E}_{ijn}(E)\Big]\sigma_{ij}(E) \; . \tag{8}$$

The Q value is zero for elastic and inelastic scattering. For (n,n′) particle reactions represented by scattering with an LR flag set, Q is the ENDF "C1" field from MF=3. For most other reactions, Q is the "C2" field from MF=3. HEATR allows users to override any Q value with their own numbers.

There are some special problems with obtaining the Q value for fission. First, the fission Q value given in the C1 field of MF3 includes delayed contributions,

and second, the Q value for fission is energy dependent. The energy dependence can be seen by writing the mass balance equation for the reaction. In energy units,

$$Q = m_A + m_n - m_{fp} - \overline{\nu} m_n \ , \tag{9}$$

where m_A is the mass energy of the target, m_n is the mass energy of the neutron, m_{fp} is the total mass energy of the fission products, and $\overline{\nu}$ is the fission neutron yield. Both m_{fp} and $\overline{\nu}$ are energy dependent. It is also necessary to account for the energy dependence of neutrino emission. Values for some of these terms and estimates for the energy dependence are given in MF1/MT458 and the discussion of that section in ENDF-102. We have chosen to ignore the energy dependence of delayed beta and gamma emission because we don't yet treat it in subsequent codes. Therefore, the following corrections are made to get an effective Q for fission in HEATR:

$$Q' = Q - 8.07E6(\overline{\nu}(E) - \overline{\nu}(0)) + .057E + .100E - E_{\beta D} - E_{\gamma D} \ , \tag{10}$$

where $E_{\beta D}$ and $E_{\gamma D}$ are the delayed beta and gamma energies from MT458. The term $.057E$ is an estimate of the energy dependence of the term m_{fp}, and the term $.100E$ is an estimate of the energy dependence of neutrino emission.

The $\overline{E}_n$ value as used in Eq. (8) is defined to include multiplicity. The multiplicity is either implicit—for example, 2 for (n,2n)—or is retrieved from the ENDF/B file (fission $\overline{\nu}$). The average energy per neutron is computed differently for discrete two-body reactions and continuum reactions.

For elastic and discrete inelastic scattering (MT=2, 51–90),

$$\overline{E}_n = \frac{E}{(A+1)^2}\left(1 + 2Rf_1 + R^2\right) \ , \tag{11}$$

where f_1 is the center-of-mass (CM) average scattering cosine from MF=4 and R is the effective mass ratio. For elastic scattering, R=A, but for threshold scattering,

$$R = A\sqrt{1 - \frac{(A+1)S}{AE}} \ , \tag{12}$$

where S is the negative of the C2 field from MF=3.

For continuum scattering, the average energy per neutron is computed from the secondary neutron spectrum g in MF=5 using

$$\overline{E}_n(E) = \int_0^U E' g(E, E')\, dE' \ , \tag{13}$$

where U is defined in MF=5. If g is tabulated (LAW=1 or LAW=5), the integral is carried out analytically for each panel by making use of the ENDF/B interpolation laws. For the simple analytic representations (LAW=7, 9, or 11), the average energies are known.[9]

The neutron cross sections required by Eq. (8) are obtained from an existing PENDF file (see RECONR and BROADR).

When the neutron sum in Eq. (7) is complete, the code processes the photon production files. If the evaluation does not include photon data, HEATR returns only the first sum. This is equivalent to assuming that all photon energy is deposited locally consistent with the fact that there will be no contribution to the photon transport source from this material. The same result can be forced by using the LOCAL parameter (see "User Input" below).

Discrete photon yields and energies are obtained from MF=12 or 13. Continuum photon data are obtained from MF=15, and the average photon energy and $\overline{E_\gamma^2}$ are computed. For radiative capture, the photon term becomes

$$E_\gamma \sigma_\gamma = \left(E + Q - \frac{E}{A+1} + \frac{\overline{E_\gamma^2}}{2(A+1)mc^2} Y_\gamma \right) \sigma_\gamma \ , \tag{14}$$

where Y_γ is the capture photon yield from MF=12. This corrects the capture contribution from Eq. (8) by conservation of momentum. For other reactions, Eq. (8) is sufficient, and the product of $\overline{E}_\gamma$, Y_γ, and σ_γ is subtracted from the neutron contribution.

D. Kinematic Limits

As an option provided mainly as an aid to evaluators, HEATR will compute the kinematic maximum and minimum KERMA factors and compare them with the energy-balance results. The formulas are as follows. For elastic scattering (MT=2), the expected recoil energy is

$$\overline{E}_R = \frac{2AE}{(A+1)^2}(1 - f_1) \ . \tag{15}$$

For discrete-inelastic scattering (MT=51–90), the photon momentum is neglected to obtain

$$\overline{E}_R = \frac{2AE}{(A+1)^2} \left[1 - f_1 \sqrt{1 - \frac{(A+1)E_\gamma}{AE}} \right] - \frac{E_\gamma}{A+1} \ , \tag{16}$$

where E_γ=−C2 from MF=3. For continuum inelastic scattering (MT=91), secondary neutrons are assumed to be isotropic in the laboratory system (LAB) giving

$$\overline{E}_R = \frac{E - E_n}{A} , \tag{17}$$

and

$$\overline{E}_\gamma = \frac{(A-1)E - (A+1)\overline{E}_n}{A} , \tag{18}$$

where $\overline{E}_\gamma$ is the average photon energy expected for this representation. For radiative capture (MT=102),

$$\overline{E}_R = \frac{E}{A+1} + E_K \tag{19}$$

and

$$\overline{E}_\gamma = Q + \frac{AE}{A+1} - E_K , \tag{20}$$

where

$$E_K = \frac{1}{2M_R c^2}\left[\frac{AE}{A+1} + Q\right]^2 \left\{1 - \frac{1}{M_R c^2}\left[\frac{AE}{A+1} + Q\right]\right\} , \tag{21}$$

with

$$M_R c^2 = (939.512 \times 10^6)(A+1) - Q \tag{22}$$

being the mass energy in eV.

For two-body scattering followed by particle emission (MT=51–91, LR flag set), a minimum and maximum can be defined:

$$(E'_R + E_x)_{\text{min}} = \overline{E}_R , \text{ and} \tag{23}$$

$$(E'_R + E_x)_{\text{max}} = \overline{E}_R + Q + (E_\gamma)_{\text{max}} , \tag{24}$$

where $\overline{E}_R$ is the value from Eq. (16) or (17), Q is the C2 field from File 3, and $(E_\gamma)_{\text{max}}$ is the negative of the C2 field from File 3. In these equations, E'_R is the recoil energy and E_x is the energy of the charged product. For absorption followed by particle emission (MT=103–120),

$$(E_R + E_x)_{\text{min}} = \frac{E}{A+1-x} , \tag{25}$$

$$(E_\gamma)_{\text{max}} = Q + \frac{A-x}{A+1-x} E , \text{ and} \tag{26}$$

$$(E_R + E_x)_{\text{max}} = E + Q , \tag{27}$$

where Q is the C2 field from MF=3 and x is the particle mass ratio (x=1 gives a minimum for all reactions). For (n,2n) reactions,

$$(E_R)_{\min} = 0 \text{ , and} \tag{28}$$

$$(E_R)_{\max} = \frac{E + \overline{E}_n}{A-1} \text{ ,} \tag{29}$$

and for (n,3n) reactions,

$$(E_R)_{\min} = 0 \text{ , and} \tag{30}$$

$$(E_R)_{\max} = \frac{E + 2\overline{E}_n}{A-2} \text{ .} \tag{31}$$

For both (n,2n) and (n,3n), if $(E_R)_{\max}$ is greater than E_R, it is set equal to E_R. In addition, these formulas are not used for A<10; the following case is used instead. For other neutron continuum scattering reactions (MT=22–37),

$$(E_R + E_x)_{\min} = 0 \text{ , and} \tag{32}$$

$$(E_R + E_x)_{\max} = E + Q - \overline{E}_n \text{ ,} \tag{33}$$

where Q is the C2 field from File 3. Finally, for fission (MT=18–21, 38), the limits are

$$(E_R)_{\min} = E + Q - \frac{1}{2}\overline{E}_n - 15{\times}10^6\,\text{eV} \text{ , and} \tag{34}$$

$$(E_R)_{\max} = E + Q - \overline{E}_n \text{ ,} \tag{35}$$

where Q is the prompt fission Q-value less neutrinos. It is determined by taking the total (less neutrinos) value from File 3 and subtracting the delayed energy obtained from MF1, MT458.

These values are intended to be very conservative. Note that E_K is only significant at very low neutron energy. In order to reduce unimportant error messages, a tolerance band is applied to the above limits. If all checks are satisfied, the resulting KERMA factors should give good local heating results even when 99.8% of the photons escape the local region. More information on using the kinematic checks to diagnose energy-balance problems in evaluations will be found in "Diagnosing Energy-Balance Problems" Section VI.I.

The upper kinematic limit can also be written out to the output tape as MT=443 if desired. It is similar to the KERMA factors generated by the MACK code,[1] and it is sometimes preferable to the energy-balance KERMA for calculating

local heating for evaluations with severe energy-balance problems. The kinematic value in MT443 is useful for plots (see the examples in this report).

E. Computation of Damage Energy

The formulas used for calculating damage energy are derived from the same sources as the heating formulas given above, except in this case, the effects of scattering angle do not result in simple factors like f_1 because the Robinson partition function is not linear. Instead, it is calculated as follows:

$$P(E) = \frac{E_R}{1 + F_L(3.4008\epsilon^{1/6} + 0.40244\epsilon^{3/4} + \epsilon)} , \tag{36}$$

if $E_R \geq 25.0$ eV, and zero otherwise. In Eq. (36), E_R is the primary recoil energy,

$$\epsilon = \frac{E_R}{E_L} , \tag{37}$$

$$E_L = 30.724 Z_R Z_L \left(Z_R^{2/3} + Z_L^{2/3}\right)^{1/2} (A_R + A_L)/A_L \text{ , and} \tag{38}$$

$$F_L = \frac{0.0793 Z_R^{2/3} Z_L^{1/2} (A_R + A_L)^{2/3}}{\left(Z_R^{2/3} + Z_L^{2/3}\right)^{3/4} A_R^{3/2} A_L^{1/2}} , \tag{39}$$

and Z_i and A_i refer to the charge and atomic number of the lattice nuclei (L) and the recoil nuclei (R). The function behaves like E_R at low recoil energies and then levels out at higher energies. Therefore, the damage-energy production cross section is always less than the heat production cross section. See Fig. 2 for examples.

For elastic and two-body discrete-level inelastic scattering,

$$E_R(E, \mu) = \frac{AE}{(A+1)^2}\left(1 - 2R\mu + R^2\right) , \tag{40}$$

where the "effective mass" is given by

$$R = \sqrt{1 - \frac{(A+1)(-Q)}{AE}} , \tag{41}$$

and μ is the CM scattering cosine. The damage energy production cross section is then obtained from

$$D(E) = \sigma(E) \int_{-1}^{1} f(E, \mu)\, P(E_R[E, \mu])\, d\mu , \tag{42}$$

where f is the angular distribution from the ENDF/B File 4. This integration is performed with a 20-point Gauss-Legendre quadrature. Discrete-level reactions with LR flags to indicate, for example, (n,n')α reactions, are treated in the same way at present. The additional emitted particles are ignored.

Continuum reaction like (n,n') give a recoil spectrum

$$E_R(E,E',\mu) = \frac{1}{A}\left(E - 2\sqrt{EE'}\mu + E'\right) , \tag{43}$$

where E' is the secondary neutron energy, μ is the laboratory cosine, and the photon momentum has been neglected. The damage becomes

$$D(E) = \sigma(E) \int_0^\infty dE' \int_{-1}^1 d\mu\, f(E,\mu)\, g(E,E')\, P(E_R[E,E',\mu]) , \tag{44}$$

where g is the secondary energy distribution from File 5. In the code, the angular distribution is defaulted to isotropic, and a 4-point Gaussian quadrature is used for the angular integration. For analytic representations of g, an adaptive integration to 5% accuracy is used for E'; for tabulated File 5 data, a trapezoidal integration is performed using the energy grid of the file. The same procedure is used for (n,2n), (n,3n), etc., but it is not realistic for reactions like (n,n'p) or (n,n'α). The neutron in these types of reactions can get out of the nucleus quite easily; thus, much of the energy available to secondary particles is typically carried away by the charged particles.[10] HEATR treats these reactions in the same way as (n,p) or (n,α).

The recoil for radiative capture must include the momentum of the emitted photons below 25–100 keV giving

$$E_R = \frac{E}{A+1} - 2\sqrt{\frac{E}{A+1}}\sqrt{\frac{E_\gamma^2}{2(A+1)mc^2}}\cos\phi + \frac{\overline{E_\gamma^2}}{2(A+1)mc^2} , \tag{45}$$

where ϕ is the angle between the incident neutron direction and emitted photon direction. If subsequent photons are emitted in a cascade, each one will add an additional term of $\overline{E_\gamma^2}$ and an additional angle. A complete averaging of Eq. (45) with respect to $P(E_R)$ would be very difficult and would require angular correlations not present in ENDF/B. However, damage calculations are still fairly crude, and an estimate for the damage obtained by treating the neutron "kick" and all the photon kicks independently should give a reasonable upper limit because

$$\int_{-1}^1 D(E_R)\, d\cos\phi \le D\left(\frac{E}{A+1}\right) + \sum_\gamma D\left(\frac{\overline{E_\gamma^2}}{2M_R c^2}\right) . \tag{46}$$

The actual formula used in the code is

$$D(E) = D\left(\frac{E}{A+1}\right) + D\left(\frac{1}{2M_R c^2}\left[\frac{AE}{A+1}+Q\right]^2\right) + \sum_\gamma D\left(\frac{\overline{E_\gamma^2}}{2M_R c^2}\right) - D\left(\frac{1}{2M_R c^2}\left[\frac{AE}{A+1}+Q\right]^2\right) , \quad (47)$$

where the first line is computed in the neutron section, and the second line is computed in the photon section. This form also provides a reasonable default when no photons are given.

Finally, for the (n,particle) reactions, the primary recoil is given by

$$E_R = \frac{1}{A+1}\left(E^* - 2\sqrt{aE^*E_a}\cos\phi + aE_a\right) , \quad (48)$$

where a is the mass ratio of the emitted particle to the neutron, E^* is given by

$$E^* = \frac{A+1-a}{A+1}E , \quad (49)$$

and the particle energy E_a is approximated as being equal to the smaller of the available energy

$$Q + \frac{AE}{A+1} , \quad (50)$$

or the Coulomb barrier energy

$$\frac{1.029\times 10^6\, zZ}{a^{1/3}+A^{1/3}} \text{ in eV} , \quad (51)$$

where z is the charge of the emitted particle and Z is the charge of the target. A more reasonable distribution would be desirable,[10] but this one has the advantage of eliminating an integration, and most results are dominated by the kick imparted by the incident neutron anyway. The angular distribution for the emitted particle is taken as isotropic in the lab. At high incident energies, direct interaction processes would be expected to give rise to a forward-peaked distribution, thereby reducing the damage. However, the importance of this effect is also reduced by the dominance of the neutron kick.

Figure 3 gives a typical result for a damage energy production calculation, showing the separate contributions of elastic, inelastic, and absorption processes.

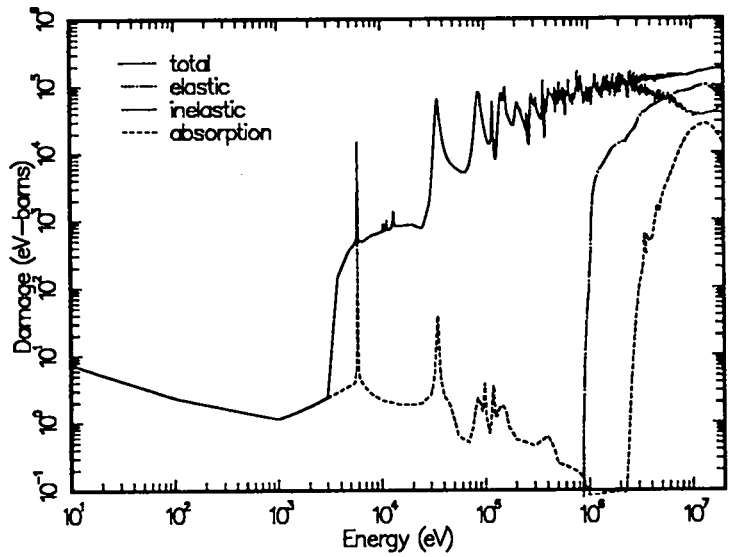


Figure 3: Components of radiation damage energy production for ^{27}Al from ENDF/B-VI. Note that capture dominates at very low energies, then elastic dominates, and finally inelastic begins to contribute at very high energies.

F. Heating and Damage from File 6

A number of the new evaluations in ENDF/B-VI include complete energy-angle distributions for all of the particles produced by a reaction, including the residual nucleus. In these cases, HEATR can compute the contributions to KERMA by calculating the average energy in the spectrum of each outgoing charged particle or residual nucleus and using Eq. (2).

A fully-used section of File 6 contains subsections for all of the particles and photons produced by the reaction, including the recoil nucleus. There are a number of different schemes used to represent the energy-angle distributions for these outgoing particles. The most important ones for HEATR follow:

- *No distribution.* In this case, the subsection is inadequate for use in heating and damage calculations. A warning message is issued.
- *Two-body angular distribution.* These are basically the same as distributions in File 4.
- *Recoil distribution.* This particle is a recoil nucleus from a two-body reaction. Its angular distribution is assumed to be the complement of the angular distribution for the first subsection in this section.
- *CM Kalbach distribution.* This format is often used by Los Alamos evaluations, and transformation to the laboratory frame is required. The looping order for the data is E, E', μ.

- *LAB Legendre distribution.* This format is used in most of the Oak Ridge evaluations for ENDF/B-VI. It is already in the laboratory frame, and the angular information can be simply ignored.
- *LAB angle-energy distribution.* This format is used for the ^{9}Be evaluation of ENDF/B-VI by Livermore. The looping order is E, μ, E'.

The normal procedure is to loop through all of these subsections. The subsections producing neutrons are skipped because they do not contribute to heating or to damage. Subsections describing charged particles and residual nuclei are processed into heating and damage contributions. Finally, the photon subsection is processed for the photon energy check, even though is does not affect either heating or damage.

For "two-body" sections, the emitted particle energy is given by

$$E' = \frac{A'E}{A+1}\left(1 + 2R\mu + R^2\right) , \tag{52}$$

where

$$R = \sqrt{\frac{A(A+1-A')}{A'}} , \tag{53}$$

and A' is the ratio of the mass of the outgoing particle to that of the incident particle. The heating is obtained by doing a simple integral over μ, and the damage is computed using the integral over μ given in Eq. (42). In both cases, the integrals are performed using either a 20-point Gauss-Legendre quadrature (for Legendre representations) or a trapezoidal integration (for tabulated data).

For "recoil" sections, the code backs up to the particle distribution and calculates the recoil using the same method described above with the sign of μ changed.

For laboratory distributions that use the E, E', μ ordering, the angular part can be ignored, and the heating and damage become

$$K(E) = \int g(E \rightarrow E')E'\,dE' , \tag{54}$$

and

$$D(E) = \int g(E \rightarrow E')P(E')\,dE' , \tag{55}$$

where $g(E \rightarrow E')$ is the angle-integrated energy distribution from File 6, and $P(E')$ is the damage partition function. Trapezoidal integration is used for the continuum, and the integrand is simply added into the sum for the delta functions (if any).

Heating for subsections that use the ordering E, μ, E' is computed using the formula

$$K(E) = \int \left\{ \int g(E \rightarrow E', \mu) E' \, dE' \right\} d\mu \ , \tag{56}$$

where an inner integral is performed using trapezoidal integration for each value in the μ grid. The results are then used in a second trapezoidal integration over μ. The damage integral is performed at the same time in a parallel manner.

The problem is somewhat more difficult for subsections represented in the center-of-mass frame. The definitions for $K(E)$ and $D(E)$ are the same as those given above, except that the quantity $g(E \rightarrow E')$ has to be generated in the lab system. The methods used to do the transformation are basically the same in HEATR and GROUPR. The first step is to set up an adaptive integral over E'. The first value needed to prime the stack is obtained by calling H6CM with E'=0. It returns the corresponding value of g in the lab system and a value for EPNEXT. The second value for the stack is computed for E=EPNEXT. The routine then subdivides this interval until 2% convergence is achieved, accumulating the contributions to the heating and damage integrals as it goes. It then moves up to a new panel. This process continues until the entire range of E' has been covered.

The key to this process is H6CM. As described in more detail in the GROUPR chapter of this manual, it performs integrals of the form

$$g_L(E \rightarrow E'_L) = \int_{\mu_{\min}}^{+1} g_C(E \rightarrow E'_C, \mu_C) \, J \, d\mu_L \ , \tag{57}$$

where L and C denote the laboratory and center-of-mass systems, respectively, and J is the Jacobian for the transformation. The contours in the E'_C,μ_C frame that are used for these integrals have constant E'_L. The limiting cosine $\mu_{\min}$ depends on kinematic factors and the maximum possible value for E'_C in the File 6 tabulation.

The ENDF/B-VI library contains a few abbreviated versions of File 6 that contain an energy-angle distribution for neutron emission, but no recoil or photon data. In these cases, the code computes $\overline{E}_n$ from File 6 ignoring any angular information, and it uses this value to compute a contribution to the heating by the energy-balance method. The damage contribution is set to zero. It is assumed that the photons corresponding to this reaction are present in MF12 or MF13, and that their effects on the heating will be subtracted later.

G. User Input

The input instructions that follow were reproduced from the comment cards in the current version of HEATR.

```
*---INPUT SPECIFICATIONS (FREE FORMAT)-------------------------*
*                                                              *
* CARD 1                                                       *
*    NENDF    UNIT FOR ENDF/B TAPE                             *
*    NIN      UNIT FOR INPUT PENDF TAPE                        *
*    NOUT     UNIT FOR OUTPUT PENDF TAPE                       *
* CARD 2                                                       *
*    MATD     MATERIAL TO BE PROCESSED                         *
*    NPK      NUMBER OF PARTIAL KERMAS DESIRED (DEFAULT=0)     *
*    NQA      NUMBER OF USER Q VALUES (DEFAULT=0)              *
*    NTEMP    NUMBER OF TEMPERATURES TO PROCESS                *
*             (DEFAULT=0, MEANING ALL ON PENDF)                *
*    LOCAL    0/1=GAMMA RAYS TRANSPORTED/DEPOSITED LOCALLY     *
*             (DEFAULT=0)                                      *
*    IPRINT   PRINT (0 MIN, 1 MAX, 2 CHECK) (DEFAULT=0)        *
* CARD 3      FOR NPK GT 0 ONLY                                *
*    MTK      MT NUMBERS FOR PARTIAL KERMAS DESIRED            *
*             TOTAL (MT301) WILL BE PROVIDED AUTOMATICALLY.    *
*             PARTIAL KERMA FOR REACTION MT IS MT+300          *
*             AND MAY NOT BE PROPERLY DEFINED UNLESS           *
*             A GAMMA FILE FOR MT IS ON ENDF TAPE.             *
*             SPECIAL VALUES ALLOWED--                         *
*               303   NON-ELASTIC (ALL BUT MT2)                *
*               304   INELASTIC (MT51 THRU 91)                 *
*               318   FISSION (MT18 OR MT19, 20, 21, 38)       *
*               401   DISAPPEARANCE (MT102 THRU 120)           *
*               443   TOTAL KINEMATIC KERMA (HIGH LIMIT)       *
*             DAMAGE ENERGY PRODUCTION VALUES--                *
*               444   TOTAL                                    *
*               445   ELASTIC (MT2)                           *
*               446   INELASTIC (MT51 THRU 91)                *
*               447   DISAPPEARANCE (MT102 THRU 120)          *
*          CARDS 4 AND 5 FOR NQA GT 0 ONLY                     *
* CARD 4                                                       *
*    MTA      MT NUMBERS FOR USERS Q VALUES                    *
* CARD 5                                                       *
*    QA       USER SPECIFIED Q VALUES (EV)                     *
*               (IF QA.GE.99.E6, READ IN VARIABLE QBAR         *
*                  FOR THIS REACTION)                          *
* CARD 5A     VARIABLE QBAR (FOR REACTIONS WITH QA FLAG ONLY)  *
*    QBAR      TAB1 RECORD GIVING QBAR VERSUS E (1000 WORDS MAX)*
*                                                              *
****************************************************************
```

Card 1 specifies the input and output units for HEATR. They are all ENDF-type files. The input PENDF tape has normally been through RECONR and BROADR, but it is possible to run HEATR directly on an ENDF tape in order to do kinematic checks. In this case, the results in the resonance range should be ignored.

On Card 2, the default value for NPK is zero, which instructs the code to process the energy-balance total KERMA (MT=301) only. Most often, the user will want to include MT443 and MT444 also (NPK=2). The kinematic KERMA computed when MT443 is requested is very useful for judging the energy-balance consistency of an evaluation (see the subsection on "Diagnosing Energy-Balance Problems" below). It can also be used instead of the energy-balance value in MT301 when local heating effects are important and the evaluation scores poorly in an energy-balance check. Damage energy production cross sections (MT444) should be computed for important structural materials; this expensive calculation can be omitted for other materials.

When kinematic checks are desired, a number of additional MTK values can be included. They can be determined by checking the evaluation to see what partial KERMA factors are well defined. For old-style evaluations that do not use File 6, look for the MT values used in Files 12 and 13. Many evaluations use only MT3 and MT102 (or 3, 18, and 102 for fissionable materials); in these cases, the only MTK values that make sense are 302, 303, and 402 (or 302, 303, 318, and 402 for fissionable materials). Caution: in many evaluations, MT102 is used at low energies and taken to zero at some breakpoint. MT3 is used at higher energies. In these evaluations, the partial KERMA MT402 does not make sense above the breakpoint, and MT3 does not make sense below it.

More complicated photon-production evaluations may include MT4 and/or discrete-photon data in MT51–90. In these cases, the user can request MTK=304. The same kind of energy-range restriction discussed for MT102 can occur for the inelastic contributions. Other evaluations give additional partial reactions that can be used to check the photon production and energy-balance consistency of an evaluation in detail. Unfortunately, HEATR can only handle 6 additional reactions at a time. Multiple runs may be necessary in complex cases.

Note that several special MTK values are provided for the components of the damage-energy production cross section. They were used to prepare Fig. 3, and they may be of some interest to specialists, but they are not needed for normal data libraries.

In a few cases in the past, it has been necessary to change the Q values that are normally retrieved from the ENDF tape. In addition, it is sometimes necessary to replace the single Q value supplied in MF3 with an energy-dependent Q function for an element. One example of the former occurred for ^{16}O for ENDF/B-V. The first inelastic level (MT51) decays by pair production rather than the more normal mode of photon emission. In order to get the correct heating, it was necessary to

change the Q value by giving Card 4 and Card 5 as follows:

```
51
-5.0294E6
```

That is, the Q value is increased by twice the electron energy of 0.511 eV. Another example is the sequential (n,2n) reaction for ^{9}Be in ENDF/B-V. It is necessary to include 4 changes to the Q values:

```
46 47 48 49/
4R-1.6651E6/
```

The next example illustrates using energy-dependent Q values for elemental titanium. Set NQA equal to 3 and give the following values on Cards 4, 5, and 5a:

```
16 103 107/
99E6 99E6 99E6/
0. 0. 0 0 1 8
8 2
8.0E6 -8.14E6 9.0E6 -8.14E6 1.1E7 -8.38E6
1.2E7 -8.74E6 1.3E7 -1.03E7 1.4E7 -1.091E7
1.5E7 -1.11E7 2.0E7 -1.125E7/
0. 0. 0 0 1 9
9 2
1.0e-5 1.82E5 4.0E6 1.82E5 5.0E6 -1.19E6
6.0E6 -2.01E6 7.0E6 -2.20E6 8.0E6 -2.27E6
1.4E7 -2.35E6 1.7E6 -2.43E6 2.0E7 -2.37E6/
0. 0. 0 0 1 9
9 2
1.0e-5 2.182E6 6.0E6 2.182E6 7.0E6 2.10E6
8.0E6 -3.11E5 9.0E6 -9.90e% 1.0E7 -1.20E6
1.1E7 -1.27E6 1.4E6 -1.32E6 2.0E7 -1.48E6/
```

The next parameter on Card 2 is NTEMP. For normal runs, use zero, and all the temperatures on the input PENDF tape will be processed. For kinematic check runs, use NTEMP=1. The LOCAL parameter suppresses the processing of the photon-production files, if any. The photon energy appears in the KERMA factors as if the photons had very short range. A useful way to use the IPRINT parameter is to set it to zero for normal runs, which produce heating and damage values at all temperatures, and to use IPRINT=2 for the energy-balance check run, which is performed for the first temperature on NIN only.

Card 3 gives the partial KERMA and damage selection MT numbers. Note that the user does not include MT301 in this list. It is always inserted as the first value automatically. Giving MT301 in this list will cause an informative message to be issued.

Cards 4, 5, and 5a give the user's changes to the ENDF Q values. The way in which to use these cards was described above in connection with NQA on Card 2.

H. Reading HEATR Output

When full output and/or kinematic checks have been requested, HEATR loops through the reactions found in Files 3, 12, and 13. For each reaction, it prints out information about the energies, yields, cross sections, and contributions to heating. The energy grid used is a subset of the PENDF grid. At present, decade steps are used below 1 eV, factor-of-two steps are used from 1 eV to 100 keV, quarter-lethargy steps are used above 100 keV, and approximately 1 MeV steps are used above 2 MeV. An example of this printout for elastic scattering in ENDF/B-VI ^{27}Al is shown below:

```
NEUTRON HEATING FOR MT  2   QO =  0.0000E+00     Q =  0.0000E+00
          E          EBAR         YIELD          XSEC        HEATING
  1.0000E-05   9.3053E-06   1.0000E+00   1.5004E+01   1.0424E-05
  1.1406E-04   1.0614E-04   1.0000E+00   4.5996E+00   3.6449E-05
  1.1406E-03   1.0614E-03   1.0000E+00   1.8953E+00   1.5019E-04
  1.1912E-02   1.1085E-02   1.0000E+00   1.4027E+00   1.1609E-03
  1.2812E-01   1.1922E-01   1.0000E+00   1.3521E+00   1.2036E-02
     ...
  4.3750E+02   4.0710E+02   1.0000E+00   1.3473E+00   4.0950E+01
  8.8750E+02   8.2584E+02   1.0000E+00   1.3472E+00   8.3068E+01
  3.0000E+03   2.7916E+03   1.0000E+00   1.3443E+00   2.8018E+02
  6.0000E+03   5.5832E+03   1.0000E+00   1.6952E+00   7.0663E+02
  1.2000E+04   1.1166E+04   1.0000E+00   1.1130E+00   9.2791E+02
  2.4000E+04   2.2333E+04   1.0000E+00   6.0689E-01   1.0119E+03
     ...
```

Note the identification and Q information printed on the first line; `Q` is the ENDF Q value from File 3, and `QO` is the corresponding mass-difference Q value needed for Eq. (3). The `EBAR`, `YIELD`, and `XSEC` columns contain $\overline{E}_n$, Y, and σ, respectively. The `HEATING` column is just $(E+Q-Y\overline{E}_n)\sigma$. The results are similar for discrete inelastic levels represented using File 4. The heating due to the associated photons will be subtracted later while MF12 or MF13 is being processed. However, if an LR flag is set, the residual nucleus from the (n,n′) reaction breaks up by emitting additional particles. This extra breakup energy changes the `QO` value. An example of such a section for ^{27}Al(n,n$_{25}$)p follows:

```
NEUTRON HEATING FOR MT 75   QO = -8.2710E+06     Q = -1.0750E+07
          E          EBAR         YIELD          XSEC        HEATING
  1.2000E+07   7.8653E+05   1.0000E+00   8.2242E-02   2.4199E+05
  1.3000E+07   1.7116E+06   1.0000E+00   8.0121E-02   2.4176E+05
  1.4000E+07   2.6427E+06   1.0000E+00   5.9282E-02   1.8296E+05
  1.5000E+07   3.5864E+06   1.0000E+00   4.1834E-02   1.3147E+05
  1.6000E+07   4.5096E+06   1.0000E+00   2.8880E-02   9.2977E+04
  1.7000E+07   5.4335E+06   1.0000E+00   1.9867E-02   6.5472E+04
  1.8000E+07   6.3848E+06   1.0000E+00   1.3677E-02   4.5739E+04
  1.9000E+07   7.2944E+06   1.0000E+00   9.4771E-03   3.2550E+04
```

```
 2.0000E+07    8.2479E+06    1.0000E+00    6.6142E-03    2.3025E+04
```

Starting with ENDF/B-VI, discrete-inelastic sections may also be given in File 6. Such sections contain their own photon production data, and the **HEATING** column will represent the entire recoil energy as in Eq. (16). (See below for detailed discussion of ENDF/B-VI output.)

For continuum reactions that use MF4 and MF5, such as (n,n′) or (n,2n), the neutron part of the display looks like this:

```
NEUTRON HEATING FOR MT 16   Q0 = -1.3057E+07     Q = -1.3057E+07
          E          EBAR          YIELD          XSEC        HEATING
 1.4000E+07    1.9960E+05    2.0000E+00    2.4000E-02    1.3051E+04
 1.5000E+07    6.6850E+05    2.0000E+00    1.2320E-01    7.4659E+04
 1.6000E+07    1.0855E+06    2.0000E+00    2.0710E-01    1.5987E+05
 1.7000E+07    1.4308E+06    2.0000E+00    2.6510E-01    2.8667E+05
 1.8000E+07    1.6379E+06    2.0000E+00    3.0300E-01    5.0518E+05
 1.9000E+07    1.7659E+06    2.0000E+00    3.3000E-01    7.9567E+05
 2.0000E+07    1.8755E+06    2.0000E+00    3.5000E-01    1.1172E+06
```

Once again, the photon effects will be subtracted later.

Absorption reactions such as (n,γ) or (n,p), lead to similar displays, but the particle **EBAR** columns will always be set to zero (no emitted neutrons). An example from ^{27}Al follows:

```
NEUTRON HEATING FOR MT103   Q0 = -1.8278E+06     Q = -1.8278E+06
          E          EBAR          YIELD          XSEC        HEATING
 2.5000E+06    0.0000E+00    1.0000E+00    3.2800E-05    2.2048E+01
 3.0000E+06    0.0000E+00    1.0000E+00    1.3300E-03    1.5590E+03
 3.5000E+06    0.0000E+00    1.0000E+00    1.0100E-02    1.6889E+04
 4.0000E+06    0.0000E+00    1.0000E+00    6.9667E-03    1.5133E+04
 4.5000E+06    0.0000E+00    1.0000E+00    1.7000E-02    4.5427E+04
 5.0000E+06    0.0000E+00    1.0000E+00    2.3300E-02    7.3912E+04
    ...
 1.7000E+07    0.0000E+00    1.0000E+00    5.5200E-02    8.3751E+05
 1.8000E+07    0.0000E+00    1.0000E+00    4.7800E-02    7.7303E+05
 1.9000E+07    0.0000E+00    1.0000E+00    4.0200E-02    6.9032E+05
 2.0000E+07    0.0000E+00    1.0000E+00    3.2200E-02    5.8514E+05
```

If File 6 is present (which happens for evaluations in ENDF-6 format only), each reaction will be divided into subsections, one for each emitted particle. The neutron subsections are simply skipped, because the neutrons do not contribute to KERMA or damage. The subsection for each charged particle or residual nucleus will give the incident energy, average energy for the emitted particle, cross section, heating contribution, and damage contribution as follows:

```
FILE SIX HEATING FOR MT 28, PARTICLE =  1001     Q =  -1.0180E+07
          E          EBAR          YIELD          XSEC       HEATING
 1.1000E+07   5.2780E+05   1.0000E+00   5.4397E-05   2.8711E+01
 1.2000E+07   1.2849E+06   1.0000E+00   9.9355E-04   1.2767E+03
 1.3000E+07   1.8296E+06   1.0000E+00   1.6900E-02   3.0920E+04
    ...
 1.8000E+07   3.9423E+06   1.0000E+00   1.8130E-01   7.1473E+05
 1.9000E+07   4.2382E+06   1.0000E+00   2.0571E-01   8.7185E+05
 2.0000E+07   4.4673E+06   1.0000E+00   2.2625E-01   1.0107E+06

FILE SIX HEATING FOR MT 28, PARTICLE = 25055     Q =  -1.0180E+07
          E          EBAR          YIELD          XSEC       HEATING
 1.1000E+07   2.0152E+05   1.0000E+00   5.4397E-05   1.0962E+01
 1.2000E+07   2.2984E+05   1.0000E+00   9.9355E-04   2.2836E+02
 1.3000E+07   2.5561E+05   1.0000E+00   1.6900E-02   4.3198E+03
    ...
 1.8000E+07   3.7970E+05   1.0000E+00   1.8130E-01   6.8840E+04
 1.9000E+07   4.0300E+05   1.0000E+00   2.0571E-01   8.2901E+04
 2.0000E+07   4.2481E+05   1.0000E+00   2.2625E-01   9.6114E+04

FILE SIX HEATING FOR MT 28, PARTICLE =     0     Q =  -1.0180E+07
          E          EBAR          YIELD          XSEC       HEATING
 1.1000E+07   7.6981E+04   1.0695E-01   5.4397E-05   0.0000E+00
 1.2000E+07   1.3750E+05   4.8214E-01   9.9355E-04   0.0000E+00
 1.3000E+07   1.5672E+05   6.4870E-01   1.6900E-02   0.0000E+00
    ...
 1.8000E+07   8.4220E+05   1.3540E+00   1.8130E-01   0.0000E+00
 1.9000E+07   1.0182E+06   1.5674E+00   2.0571E-01   0.0000E+00
 2.0000E+07   1.1886E+06   1.7566E+00   2.2625E-01   0.0000E+00
```

Note that the last subsection in this example was for emitted photons. Photons do not contribute to the KERMA or damage, but this information is needed later for the photon energy check.

After all the sections corresponding to MT numbers in File 3 have been processed (using the File 4, File 4/5, or File 6 method as appropriate), the photon production sections in Files 12 and 13 are processed, if present. File 12 data are usually present for radiative capture (MT=102), at least at low energies. Simple files normally give a tabulated photon spectrum. The display gives the average energy for this spectrum in the EBAR column and the negative contribution to the heating computed with Eq. (14) in the HEATING column. The EDAM column contains the $\overline{E_\gamma^2}$ term needed to compute the photon contribution to the damage, which is given in the DAMAGE column. See Eq. (47). The display also has an extra line for each incident energy containing the percent error "--- PC" between the total photon energy as computed from File 12 and the value $E+Q-E/(A+1)$ computed from File 3. As discussed above, HEATR does not guarantee energy balance in large systems if this error occurs. The following example shows some large errors

due to mistakes in the ENDF/B-V evaluation for ^{55}Mn. Two columns labeled **EDAM** and **XSEC** have been removed to show the **HEATING** and **DAMAGE** columns. The text has also been shifted to the left of its normal position.

```
PHOTON ENERGY (FROM YIELDS) MF12, MT102
          E       EBAR/ERR          EGAM ...       YIELD       HEATING        DAMAGE
 1  CONTINUUM GAMMAS
 1.0000E-05    4.5088E+06    2.4237E+02 ... 2.4791E+00   -4.8477E+09    5.4975E+04
 1.0000E-05      53.7 PC
 1.0703E-04    4.5088E+06    2.4237E+02 ... 2.4791E+00   -1.4819E+09    1.6806E+04
 1.0703E-04      53.7 PC
 1.2520E-03    4.5088E+06    2.4237E+02 ... 2.4791E+00   -4.3347E+08    4.9159E+03
 1.2520E-03      53.7 PC
 1.4292E-02    4.5088E+06    2.4237E+02 ... 2.4791E+00   -1.2830E+08    1.4551E+03
 1.4292E-02      53.7 PC
   ...
 1.3571E+04    4.4522E+06    2.3887E+02 ... 2.4836E+00   -7.3869E+03   -1.2406E-01
 1.3571E+04      51.8 PC
 2.7142E+04    4.3957E+06    2.3536E+02 ... 2.4881E+00   -3.7037E+05   -1.6487E+01
 2.7142E+04      49.9 PC
 5.4287E+04    4.2826E+06    2.2834E+02 ... 2.4972E+00   -2.5604E+04   -2.5340E+00
 5.4287E+04      46.0 PC
 1.0858E+05    4.0563E+06    2.1430E+02 ... 2.5154E+00   -1.3327E+04   -2.7289E+00
 1.0858E+05      38.3 PC
```

Many sections MF12, MT102 give multiplicities for the production of discrete photons. In these cases, HEATR prints out data for all of the parts, and it provides a sum at the end. The balance error is printed with the sum. The following example shows a case with discrete photons. The last two columns have been removed (**HEATING**, **DAMAGE**), and the text has been shifted to the left.

```
PHOTON ENERGY (FROM YIELDS) MF12, MT102
          E       EBAR/ERR          EGAM          EDAM          XSEC          YIELD
 1   7.7260E+06 EV GAMMA
 1.0000E-05    7.7260E+06    1.1448E+03    8.9780E+02    1.1677E+01    3.0000E-01
 1.1406E-04    7.7260E+06    1.1448E+03    8.9780E+02    3.4574E+00    3.0000E-01
 1.1406E-03    7.7260E+06    1.1448E+03    8.9780E+02    1.0934E+00    3.0000E-01
   ...
 2.0000E+07    2.7005E+07    1.1448E+03    8.9780E+02    1.0000E-03    3.0000E-01
 2   7.6950E+06 EV GAMMA
 1.0000E-05    7.6950E+06    1.1356E+03    8.9091E+02    1.1677E+01    5.0000E-02
 1.1406E-04    7.6950E+06    1.1356E+03    8.9091E+02    3.4574E+00    5.0000E-02
   ...
 2.0000E+07    2.6974E+07    1.1356E+03    8.9091E+02    1.0000E-03    5.0000E-02
 3   6.8630E+06 EV GAMMA
 1.0000E-05    6.8630E+06    9.0330E+02    7.1515E+02    1.1677E+01    1.2000E-03
 1.1406E-04    6.8630E+06    9.0330E+02    7.1515E+02    3.4574E+00    1.2000E-03
 1.1406E-03    6.8630E+06    9.0330E+02    7.1515E+02    1.0934E+00    1.2000E-03
   ...
89   3.1000E+04 EV GAMMA
```

```
1.0000E-05   3.1000E+04   1.8430E-02   0.0000E+00   1.1677E+01   2.8884E-01
1.0000E-05      0.0 PC
1.1406E-04   3.1000E+04   1.8430E-02   0.0000E+00   3.4574E+00   2.8884E-01
1.1406E-04      0.0 PC
1.1406E-03   3.1000E+04   1.8430E-02   0.0000E+00   1.0934E+00   2.8884E-01
1.1406E-03      0.0 PC
1.1912E-02   3.1000E+04   1.8430E-02   0.0000E+00   3.3832E-01   2.8884E-01
1.1912E-02      0.0 PC
  ...
2.0000E+07   3.1000E+04   1.8430E-02   0.0000E+00   1.0000E-03   2.8884E-01
2.0000E+07      0.0 PC
```

In this case (^{27}Al), the capture energy production checks out perfectly for the sum of all 89 discrete photons.

Other sections using either File 12 or File 13 generate displays similar to the following:

```
PHOTON ENERGY (FROM XSECS) MF13, MT  3
            E           EBAR          XSEC         ENERGY       HEATING
    1  CONTINUUM GAMMAS
    2.0000E+05   3.6753E+06   4.2076E-03   1.5464E+04  -1.5464E+04
    4.0500E+05   3.3863E+06   5.2873E-03   1.7904E+04  -1.7904E+04
    6.0031E+05   3.1097E+06   6.4478E-03   2.0051E+04  -2.0051E+04
    8.0182E+05   2.0089E+06   9.3236E-02   1.8730E+05  -1.8730E+05
    1.0000E+06   9.2622E+05   1.7859E-01   1.6541E+05  -1.6541E+05
    1.2000E+06   9.6151E+05   2.8329E-01   2.7239E+05  -2.7239E+05
   ...
```

Note that the photon $\overline{E}\sigma$ is simply subtracted from the HEATING column for each incident energy.

If the partial KERMA MTK=443 was requested in the user's input, HEATR will print out a special section that tests the total photon energy production against the kinematic limits (see Section VI.D above for the formulas used). An example follows:

```
PHOTON ENERGY PRODUCTION CHECK
            E      EV-BARNS          MIN          MAX
    1.0000E-05   9.0215E+07   9.0187E+07   9.0200E+07
    1.1406E-04   2.6712E+07   2.6704E+07   2.6708E+07
    1.1406E-03   8.4479E+06   8.4453E+06   8.4466E+06
    1.1912E-02   2.6138E+06   2.6130E+06   2.6134E+06
    1.2812E-01   7.9895E+05   7.9871E+05   7.9883E+05
    1.2812E+00   2.5211E+05   2.5203E+05   2.5207E+05
    2.6875E+00   1.7420E+05   1.7415E+05   1.7417E+05
    5.5000E+00   1.2186E+05   1.2182E+05   1.2184E+05
    1.1406E+01   8.4662E+04   8.4636E+04   8.4648E+04
    2.4062E+01   5.8231E+04   5.8213E+04   5.8222E+04
    4.9375E+01   4.0614E+04   4.0601E+04   4.0607E+04
```

```
  1.0000E+02    2.8522E+04    2.8514E+04    2.8518E+04
...
  8.0000E+06    3.8972E+06    3.7964E+06    4.4880E+06
  9.0000E+06    4.4782E+06    4.4401E+06    5.4986E+06
  1.0000E+07    4.9645E+06    5.2078E+06    6.6176E+06
  1.1000E+07    5.3712E+06    6.1302E+06    7.8374E+06   ----
  1.2000E+07    5.3212E+06    5.8699E+06    8.0618E+06
  1.3000E+07    5.0984E+06    5.9333E+06    8.5312E+06   ----
  1.4000E+07    4.7415E+06    5.7172E+06    8.6605E+06   ----
  1.5000E+07    4.0795E+06    4.7419E+06    8.0409E+06   ----
  1.6000E+07    3.2521E+06    3.6806E+06    7.2734E+06   ----
  1.7000E+07    2.8079E+06    2.8418E+06    6.6927E+06
  1.8000E+07    2.7492E+06    2.2915E+06    6.2850E+06
  1.9000E+07    2.9626E+06    1.9674E+06    6.2330E+06
  2.0000E+07    3.4419E+06    1.7938E+06    6.1994E+06
```

The low and high kinematic limits will be the same at low energies where only kinematics affect the calculations. They may be the same for all energies for ENDF/B-VI evaluations that provide complete distributions for all outgoing charged particles and recoil nuclei. Normally, the limits diverge above the threshold for continuum reactions. Note that HEATR marks lines where the computed value goes more than a little way outside the limits with the symbols `++++` or `----`. It is often convenient to extract these numbers from the output listing and plot them (see Fig. 4). Although the energy grid is a little coarse, such plots can often be useful (see below).

The last part of a full HEATR output listing is a tabulation of the computed KERMA and damage factors on the normal coarse energy grid. Columns are provided for the total KERMA and for each of the partial KERMA results requested with `MTK` values in the user's input. If kinematic checks were requested, the check values are written just above and below the corresponding partial KERMA values. In addition, `LOW` and `HIGH` messages are written just above or just below the kinematic limits in every column where a significant violation of the limits occurs. Caution: if summation reactions (MT3, MT4) were used to define the photon production over some parts of the energy range, the partial KERMA results may not make sense at some energies. For example, consider the common pattern in ENDF/B-V where MT102 is used for capture at low energies, but at higher energies, it is set to zero, and the capture contribution is included in MT3 (nonelastic). Clearly, the partial KERMA MT402 doesn't make sense above this breakpoint. The following example shows part of the final KERMA listing for ENDF/B-V ^{55}Mn. The damage column was removed.

```
FINAL KERMA FACTORS
            E          301          302          303          402          443
```

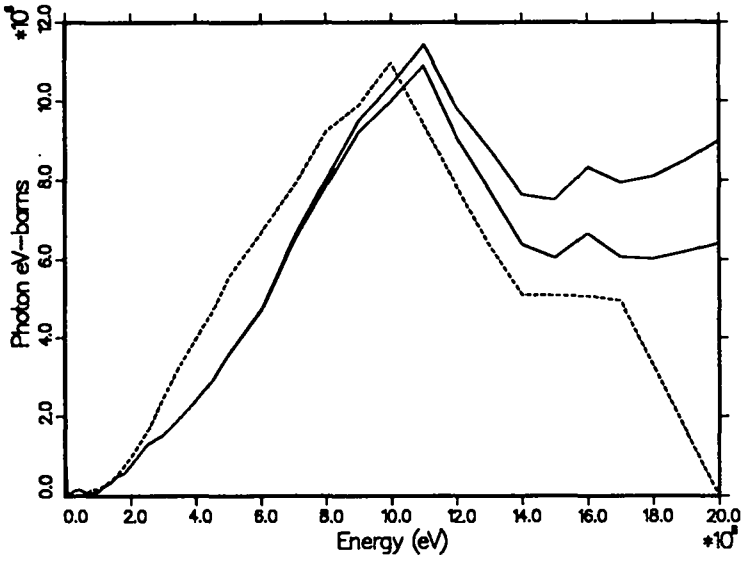


Figure 4: Example of a plot comparing the total photon energy production for ^{55}Mn from ENDF/B-V.1 (dashed) with the kinematic limits (solid).

```
          MIN    1.2400E-04   3.7775E-06   1.2022E-04   1.2022E-04
1.0000E-05       4.0068E+05   3.7775E-06   4.0068E+05   4.0068E+05   4.0068E+05
          MAX    3.3820E+05   3.7775E-06   3.3820E+05   3.3820E+05
                   HIGH                      HIGH         HIGH

          MIN    4.0650E-04   1.3174E-05   3.9333E-04   3.9333E-04
1.0703E-04       1.2248E+05   1.3174E-05   1.2248E+05   1.2248E+05   1.2248E+05
          MAX    1.0338E+05   1.3174E-05   1.0338E+05   1.0338E+05
                   HIGH                      HIGH         HIGH

...

          MIN    3.2373E+04   2.3497E+04   8.8759E+03   4.1114E+01
6.0031E+05       1.0017E+05   2.3497E+04   7.6673E+04   2.9899E+04   3.2375E+04
          MAX    3.2375E+04   2.3497E+04   8.8781E+03   4.3366E+01
                   HIGH                      HIGH         HIGH

                   LOW                       LOW
          MIN    7.3041E+04   5.9403E+04   1.3638E+04   4.4748E+01
8.0182E+05      -2.4355E+04   5.9403E+04  -8.3758E+04   2.4987E+04   7.3043E+04
          MAX    7.3043E+04   5.9403E+04   1.3640E+04   4.6676E+01
                                                         HIGH

                   LOW                       LOW
          MIN    9.8973E+04   7.7075E+04   2.1898E+04   4.7777E+01
1.0000E+06       3.7682E+04   7.7075E+04  -3.9393E+04   2.1917E+04   9.8974E+04
```

```
          MAX    9.8974E+04    7.7075E+04    2.1900E+04    4.9509E+01
                                                             HIGH

                   LOW                         LOW
          MIN    1.1397E+05    7.7800E+04    3.6168E+04    4.9760E+01
1.2000E+06       9.5321E+04    7.7800E+04    1.7521E+04    1.9482E+04    1.1397E+05
          MAX    1.1397E+05    7.7800E+04    3.6169E+04    5.1335E+01
                                                             HIGH

                   LOW                         LOW
          MIN    1.4632E+05    1.0192E+05    4.4402E+04    5.3005E+01
1.4000E+06       9.8251E+04    1.0192E+05   -3.6667E+03    1.8208E+04    1.4632E+05
          MAX    1.4632E+05    1.0192E+05    4.4403E+04    5.4511E+01
                                                             HIGH

...

          MIN    2.3650E+05    1.6907E+05    6.7426E+04    1.7607E+02
1.9000E+07       7.1384E+06    1.6907E+05    6.9693E+06    1.3503E+04    2.5435E+06
          MAX    2.5435E+06    1.6907E+05    2.3744E+06    1.7939E+02
                   HIGH                        HIGH          HIGH

          MIN    2.4001E+05    1.8261E+05    5.7406E+04    1.4423E+02
2.0000E+07       9.2369E+06    1.8261E+05    9.0543E+06    1.0908E+04    2.8385E+06
          MAX    2.8385E+06    1.8261E+05    2.6559E+06    1.4701E+02
                   HIGH                        HIGH          HIGH
```

The following subsection discusses how to analyze the "check" output of HEATR in order to diagnose energy-balance errors in ENDF-format evaluations. The examples are drawn from ENDF/B-V testing.[3]

I. Diagnosing Energy-Balance Problems

The analysis should start with MT102, because if it is wrong, the guarantee energy conservation for large systems breaks down. If the display for MF12,MT102 shows messages of the form "--- PC", there may be a problem. If these messages only show up at the higher energies, and if the size of the error increases with energy, it is probable that the evaluator has used a thermal spectrum over the entire energy range (this is very common). Of course, the total photon energy production from radiative capture should equal

$$\frac{A}{A+1}E+Q\ , \tag{58}$$

where the rest of the total energy E+Q is carried away by recoil. If only a thermal spectrum is given, the E term is being neglected, and errors will normally appear above about 1 MeV. The E term can be included in evaluations that use

tabulated data by giving E-dependent spectra in File 15; and it can be included for evaluations that use discrete photons by setting the "primary photon" flags in File 12 properly. In practice, the capture cross sections above 1 MeV are often comparatively small due to the $1/v$ tendency of capture, and the errors introduced by neglecting the E term can be ignored.

If the MT102 errors show up at low energies, there is probably an error in the average photon yield from File 12, in the average energy computed from File 15, or both. In the ^{55}Mn case shown above, the yield had been incorrectly entered. In addition, the spectrum didn't agree with the experimental data because the bin boundaries were shifted. Each case must be inspected in detail to find the problems.

The next common source of energy-balance errors in ENDF files arises from the representation used for inelastic scattering. Typically, the neutron scattering is described in detail using up to 40 levels for the (n,n') reaction. However, the photon production is often described using MF13/MT3 or MF13/MT4 and rather coarse energy resolution. As a result, it is possible to find photons for (n,n_1) being produced for incident neutron energies slightly below the MT51 threshold! These photons would lead to a spike of negative KERMA factors. A more common effect of the coarse grid used for photon production is to lead to an underestimate or overestimate of the photon production by not following the detailed shape of the inelastic cross section. The HEATR "kinematic KERMA" is correct in this range since only two-body reactions are active. Therefore, a plot of MT301 and MT443 on the same frame normally shows these effects in detail. Figure 5 is an example of such a plot. Figure 6 shows both the inelastic cross section from File 3 and the photon production cross section from File 13 to demonstrate the mismatch in the energy grids that contributes to the energy-balance errors. These kinds of errors are best removed by changing to a representation that uses File 12 to give photon production yields for the separate reactions MT51, MT52, etc. This representation makes full use of the File 3 cross sections, and as long as each section of File 12 conserves energy, the total inelastic reaction is guaranteed to conserve energy, even at the finest energy resolution.

A method that is frequently used by evaluators of photon production files is to select a number of nonelastic photon spectra on a fairly coarse incident-energy grid using theory or experiment, and then to readjust the photon yield on this energy grid so as to conserve energy at each grid point. However, the results do not, in general, conserve energy at intermediate points. If a very coarse energy grid is used for File 13, quite large deviations between MT301 and MT443 can

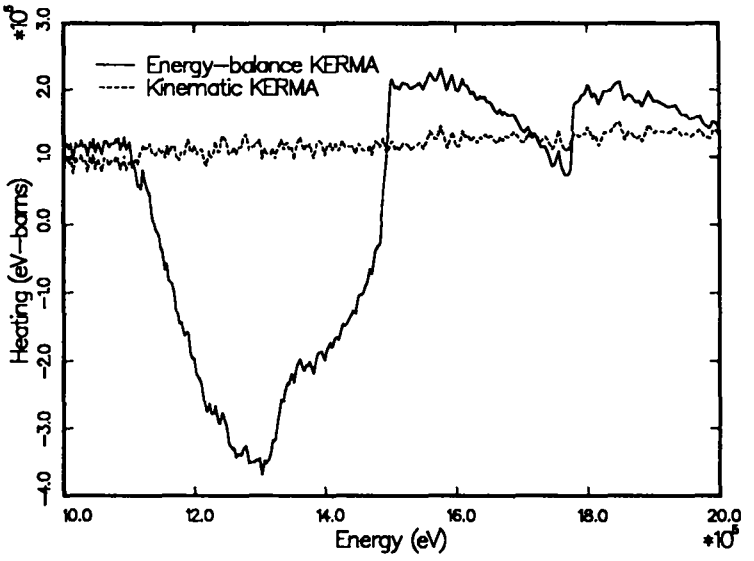


Figure 5: Comparison of MT301 with MT443 for the region of the discrete-inelastic thresholds for ^{59}Co from ENDF/B-V.2. Note the large region of negative KERMA. The best way to remove this kind of problem is by using yields in File 12, MT51, 52, 53, ...to represent the photon production.

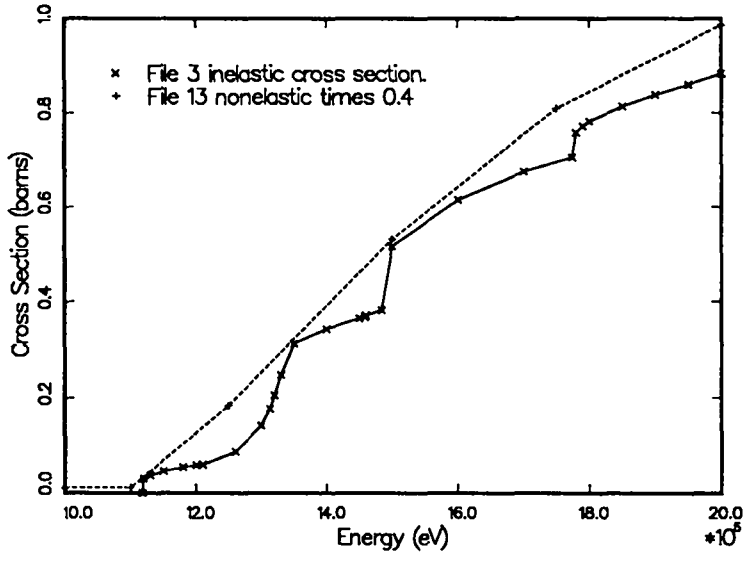


Figure 6: Plot showing the mismatch between the energy grids used for File 3 and File 13 in the region of the thresholds for discrete-inelastic scattering levels for the case shown in Fig. 5. The cross and ex symbols show the actual grid energies in the evaluation.

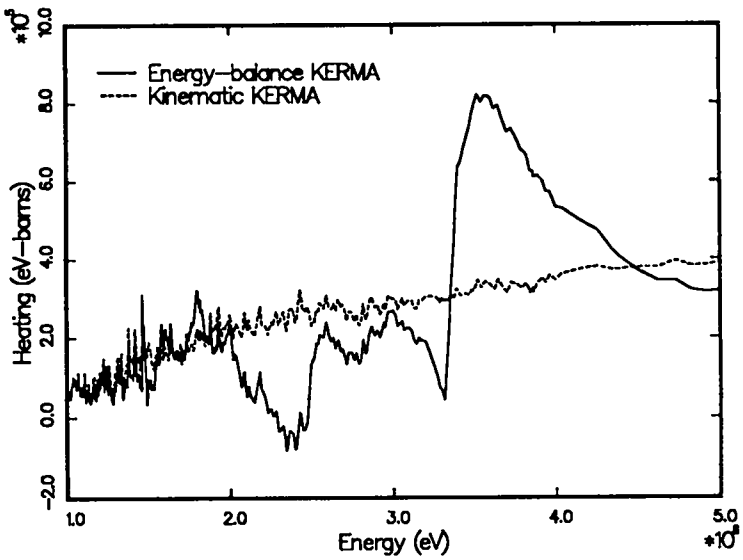


Figure 7: Typical energy-balance problems between points where balance is satisfied. Discrete photons were used below about 2 MeV, and energy balance is reasonably good there. The energy points in MF13 for the continuum part are at 2, 3, and 5 MeV, and the balance is also good at those energies. Clearly, a grid in File 13 that used steps of about 0.25 MeV between 2 and 4 MeV would reduce the size of the deviations substantially and remove the negative KERMA factors.

result. Figure 7 shows such a case. The solution to this kind of violation of energy balance is to add intermediate points in Files 13 and 15 until the magnitude of the deviations is small enough for practical calculations.

Especially large energy-balance errors of this type are caused by interpolating across the minimum formed by the decreasing capture heating and the increasing inelastic heating. Figure 8 shows a dramatic example using a photon energy production comparison.

For energies above the threshold for continuum reactions like (n,n′) or (n,2n), it is difficult to use the results of the kinematic checks to fix evaluations. The representation of Eqs. (17) and (18) for continuum inelastic scattering is very rough. Comparison to other more accurate methods suggests that a CM formula would be better here,[11] even though the ENDF file says "lab." Most other reactions give very wide low and high limits. Two exceptions are (n,2n) and (n,3n). If they dominate the cross section, the kinematic limits will be fairly close together. In the 14 MeV range, energy errors could be in the photon data, the neutron data, or both. The best way to eliminate balance errors is to construct a new evaluation

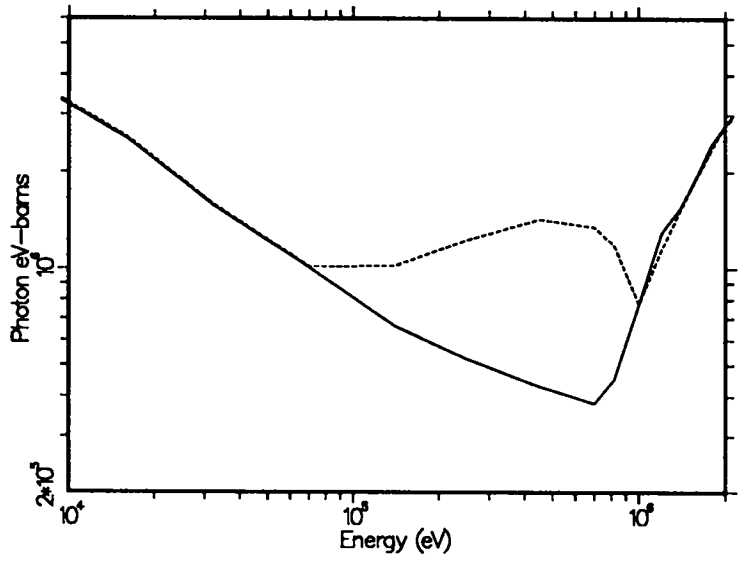


Figure 8: Computed photon energy production (dashed) compared with the kinematic value (solid) for ^{93}Nb from ENDF/B-V. The original File 13 has grid points at 100 keV and 1 MeV. Interpolating across that wide bin gives a photon production rate that is much too large for energies in the vicinity of a few hundred keV. This will result in a large region of negative heating numbers. Since this is just the region of the peak flux in a fast reactor, niobium-clad regions could be cooled instead of heated!

based on up-to-date nuclear model codes.

J. Coding Details

The main program starts by reading user's input and locating the desired material on the PENDF tape. The main loop over temperature goes through statement number 160. For each temperature, HINIT is called to examine the directory. Flags are set if MF=12 or 13 is present, if MT=18 or 19 is used, and if MT458 is present (see MGAM, MT19, and MT458). The flags MT103, MT104, MT105, MT106 and MT107 are set if the corresponding particle production levels are present. The MT numbers used for the levels depend on whether the input file used version 6 format or one of the earlier formats. For example, MT103 is set if MT700–749 is found for ENDF-6 data, or if MT600-619 is found for earlier versions. The code also checks to see if the corresponding angular distribution data are present (see NMISS4). If any are not present, the code will assume they are isotropic. Note that HINIT also collects a list of the File 6 MT numbers in MT6(I6). The statements just before statement number 155 retrieve the delayed fission energy value from MF1/MT458 QDEL for later use when calculating the heating from prompt fission.

The next step in HINIT is to make a copy of File 6 on a scratch tape (if any sections of File 6 were found), and to make a standardized copy of the ENDF tape using HCONVR. This subroutine also saves the grid of the total cross section (MT=1) on the LOADA/FINDA scratch file that will be used to accumulate the KERMA factors, damage, and kinematic checks (if requested). Note also MT303, which tells which of the requested edits is for nonelastic heating, MT=303. This is used later for writing out the photon energy production check.

Now NHEAT is called. Its basic function is to loop over the "nonredundant" reactions in File 3, and to accumulate the corresponding contributions to the partial heating and partial damage values into the appropriate elements of the C array on the LOADA/FINDA file. Redundant reactions are reactions that duplicate or include effects that can be obtained from another MT number. They are determined using a set of IF statements just after the entry to the reaction loop at statement number 105. The structure of the C array depends on whether kinematic checks are being accumulated or not and whether photon production files are present. If neither occurs, the structure has NPK+1 elements as follows:

Element	Contents
1	energy
2	total heating
3	value for first partial
...	
NPK+1	value for last partial

where NPK is the number of partial KERMA or damage values being accumulated, including the total. If checks are being accumulated, the C array has the following 3*NPK+1 elements:

Element	Contents
1	energy
2	total heating
3	value for first partial
...	
2+NPK	lower kinematic limit for total
3+NPK	lower kinematic limit for first partial
...	
2+2*NPK	upper kinematic limit for total
3+2*NPK	upper kinematic limit for first partial
...	
3*NPK+1	upper kinematic limit for last partial

If photon production files are present in the evaluation, the total length of the C array increases by the following three words (LEN is the old length from above plus three):

Element	Contents
LEN-2	photon capture correction
LEN-1	total photon eV-barns
LEN	total energy yield for "subtot"

Back inside the loop over nonredundant reactions, subroutine GETY1 is initialized for this reaction. The code checks to see if this section uses File 6 for its distributions; if so, it finds the corresponding section in File 6. It is now possible to select the appropriate Q value and particle yield, and to initialize the appropriate calculational routine. This routine will be SIXBAR for all reactions described in File 6, DISBAR for two-body reactions using File 4 (including charged-particle reactions in the 600 or 700 series of MT numbers), CONBAR for continuum reactions represented using File 5, and CAPDAM for the neutron disappearance reactions (MT=102, 103, etc.) and the charged-particle continuum reactions from the 600 or 700 series of MT numbers. The last step before beginning the energy loop for this reaction is to call INDX, which determines which element of the C array is to receive the heating or damage contribution from this reaction (see below).

The energy loop in NHEAT goes through statement number 190. For each energy, FINDA is called to retrieve the current values for the energy [see C(1)] and the partial heating and damage values as accumulated so far. On the first pass through the scratch file, the list of energies to be used for printing information on the listing is established in ELIST using a few IF statements based on the range of the energy variable E. For each energy, the corresponding cross section is retrieved

using GETY1 and the appropriate $\overline{E}$ and damage numbers are computed by calling GETSIX, DISBAR, CONBAR, or CAPDAM. The heating contribution is computed from the appropriate formula, and the heating and damage numbers are summed into the C array at location INDEX. If requested, the kinematic limits on the heating are computed and summed into the C array at INDEX+NPK and INDEX+2*NPK. The completed results for this energy and reaction are written out onto the LOADA/FINDA scratch file, and the energy loop is continued.

When the energy loop is complete, the subroutine jumps to the next section (or subsection in the case of File 6) and repeats the entire energy loop for that reaction (or particle from File 6).

Subroutine INDX is used to select what element of the C array is to receive the heating or damage contribution for a section with a particular MT number. The meaning of each element of the C array is obtained from the MTP array. Normally, a reaction MT contributes to the partial heating element with MTP(I)=MT+300. But it can also contribute to several other elements of C, such as nonelastic (MT303), inelastic (MT304), etc. Therefore, INDX returns the count of reactions contributed to by MT in NMT and the indexes for the C array in IMT(NMT).

Subroutine CAPDAM is used to compute the damage energy for neutron capture (or disappearance) reactions; that is, for MT=102, 103, etc. On the initialization entry (EE=0), the routine sets up various kinematics parameters, such as ZX and AX to describe the outgoing particle, and initializes DF. In order to save time, the routine only calculates the damage on a grid that increases by steps of 10%. Intermediate values are obtained by interpolation (see EL, DAML, EN, and DAMN). The values at the grid points are computed using

$$D\left(\frac{E}{A+1}\right)+D\left(\frac{1}{2M_Rc^2}\left[\frac{AE}{A+1}+Q\right]^2\right) \tag{59}$$

for radiative capture (the corrections for multiple photon emission will be made later), or using Eq. (42) with E_R from Eq. (48) and a 4-point Gauss-Legendre quadrature. The angular distribution for particle emission is taken to be isotropic.

Subroutine DISBAR is used by NHEAT to compute the average secondary energy and damage energy for elastic scattering (MT=2), discrete-inelastic scattering (MT=51–90), or discrete-level particle production (MT=600–648, 650–698, etc. for ENDF/B-VI, or MT=700–717, 720–737, etc. for earlier versions). It starts by initializing HGTFLE (which is very similar to GETFLE in the GROUPR module), determining kinematic parameters like AWP (the mass ratio to the neutron for the emitted particle), and initializing DF. In order to save time, it only computes the

heating and damage on a grid based on steps by a factor of 1.1 and the ENEXT values from HGTFLE. On a normal entry, it interpolates between these values (see EL, CL, DAML, EN, CN, and DAMN). When the desired EE exceeds EN, the old high values are moved down to the low positions, and new high values are calculated. The calculation of CN follows Eq. (11). The calculation of DAMN uses Eq. (42) with a 20-point Gauss-Legendre quadrature (see NQ, QP, and QW).

Function DF is used to compute the damage partition function given in Eq. (36). The constants that depend on the recoil atom or particle type and lattice type (see ZR, AR, ZL, AL) are computed in an initialization call with E=0. Thereafter, it can be called with any other value of E.

Similarly, CONBAR computes the average secondary energy and damage energy for continuous distributions described in File 5. Analytic representations use simple formulas coded into ANABAR or a combination of adaptive and Gaussian quadrature in ANADAM. Tabulated data are interpolated from the File 5 table using TABBAR or integrated using trapezoidal and Gaussian quadratures in TABDAM. As usual, the routine is initialized by calling it with E=0. The secondary-particle yield is either chosen from the MT number, or HGTYLD is initialized. The desired section of File 5 is located on the input ENDF tape, and the kinematic constants are computed. The reactions with MT=22, 28, 32, 33, and 34 will be treated using the CAPDAM method; if MTD has one of these values (see MTT), CAPDAM is initialized. As is the case for GETSED in the GROUPR module, this routine can handle some sections of File 5 that contain multiple subsections, but the analytic subsections must come first. The loop on subsections goes through statement number 104. As each analytic subsection is read, appropriate data are stored in memory using pointers in the array LOC. Only the first energy is read and stored for a tabulated subsection (LF=1). The idea is to have only two energy values in memory at a time in order to save storage; the second subsection will be read during the first normal entry to the subroutine. The final step in the initialization pass is to initialize DF. For a normal entry into CONBAR, the energy-dependent fission yield is retrieved, if needed, and the loop over subsections is entered (see "DO 420 IK=1,NKTOT"). Each subsection in File 5 starts with a fractional-probability record. The desired value for energy E is computed by interpolation using the standard NJOY utility routine TERPA. For analytic subsections, the routine jumps to statement number 400 and uses ANABAR to compute $\overline{E_n}$, and ANADAM or CAPDAM to compute the damage energy. Note that in order to save time, ANADAM is only calculated on a fairly coarse grid based on steps by a factor of 1.5. The intermediate values are obtained by interpolation using TERP1. For tabulated subsections, EBAR and DAME values are normally obtained by interpolation (see ELO, FLO, DLO, EHI, FHI, and DHI).

However, for the first entry, or whenever E reaches EHI, the high data are moved into the low positions, new high data are read from the File 5 subsection, and the values for heating and damage are computed at EHI using TABBAR and either TABDAM or CAPDAM.

Subroutine HGTYLD is similar to GETYLD in the GROUPR module. It finds the required section on the ENDF tape and reads the entire LIST or TAB1 record into memory. On normal entries, it either computes the yield using the polynomial formula with constants from the LIST record, or it uses TERPA to interpolate for the yield in the TAB1 data.

Subroutine ANABAR is used to compute the average energy for a neutron described by an analytic subsection of File 5. The simple formulas used are tabulated in the ENDF format manual.[9] Similarly, ANADAM is used to compute the damage energy for an analytic subsection of File 5. Only LF=9 (the Simple Maxwellian Distribution) is supported; the routine returns zero for other laws. Note that a statement function is defined to compute the secondary energy distribution for this law (see SED). For each incident energy, the spectrum temperature THETA is retrieved using TERPA, and an adaptive integration stack is initialized with points at four secondary energies, namely, 1., $.5(E-U)$, θ, and $E-U$, where U is a parameter that sets the maximum possible value of E'. The adaptive procedure proceeds to solve Eq. (44) by subdividing this starting grid until trapezoidal integration can be used on each panel. The inner integral over emission cosine μ is performed using a 4-point Gauss-Legendre quadrature for each point on the adaptive grid. The statement function GETSED is used to compute $g(E')$, and DF is used to compute the partition function.

Subroutine TABBAR is used to compute the mean energy of the emitted neutron for a tabulated subsection of File 5. It can also be used for a tabulated subsection of File 6. This option is flagged by LAW negative. The trick is to set the "stride" or "cycle" through the file to be larger than 2 (see NCYC). The angular part of the $g(E \rightarrow E')$ table is skipped, and only the E' and g values are retrieved. For File 5, this routine only works for laws 1 and 5; others cause a fatal error message to be issued. In both of these cases, the integral over E' needed to compute the average energy is done analytically for each panel in the input data using a different formula for each interpolation scheme INT.

Subroutine TABDAM is used to compute the damage energy for a tabulated subsection of File 5. The integration that is needed is given in Eq. (44). The energy grid of the tabulation is assumed to be good enough to allow trapezoidal integration to be used for E', and a 4-point Gauss-Legendre quadrature is used

for μ.

Subroutine SIXBAR is used to compute charged-particle average energy and damage energy represented by using a subsection of File 6. In some evaluations, File 6 sections occur that contain only neutron data and no charged-particle or recoil spectra. In these cases, SIXBAR is used to compute the neutron average energy for the energy-balance method, and no damage energy is computed. As is common with NJOY subroutines, SIXBAR is initialized by calling it with E=0. The first step is to read in the TAB1 record that contains the particle yield and identity (ZAP and AWP).

If the particle described by the subsection is a neutron, and assuming that there is more than one subsection (see N6), the routine skips over the neutron subsection and returns EBAR=-1 as a flag to the calling program. On some subsequent entry, the particle will not be a neutron, and the code jumps to statement number 200 to check whether the particle is a recoil nucleus. If it is, the routine backs up to the first subsection again, which it assumes is the particle causing the recoil. Note that this puts some restrictions on the form of File 6. When the code finally arrives at statement number 210, it is ready to start processing the current subsection. It reads in the parameters for laws 3 and 6, or the TAB2 record and the data for the first energy point for the other laws. With the data in place, it computes the corresponding values for mean energy and damage energy using GETSIX and returns.

In the special case where the section contains only a single subsection that describes a neutron, the data stored in memory will be the data for that subsection, and the subroutine TABBAR with a negative value for the law is used to produce the low values.

On a normal entry (E>0), SIXBAR checks to see whether E is in the current interpolation range. If it is, the code jumps to statement number 400. For the analytic laws (LAW=3 and LAW=6), it uses a direct call to GETSIX to compute the mean energy and damage energy. For the tabulated laws, it interpolates for the results using the low and high data (see ELO, FLO, DLO, EHI, FHI, and DHI). On the first entry, or whenever E increases to EHI, the code moves the high data to the low positions, and then it reads in the data for the next energy and computes a new set of high values for mean energy and damage using GETSIX or TABBAR.

Subroutine GETSIX is used to compute the mean energy and damage energy for one particular incident energy in a subsection of File 6. The method used depends on the value of LAW and the reference frame for the subsection. The first case in the coding is for LAW=1 with data in the CM system.

This case uses Eqs. (54) and (55) with an adaptive integration over E'. The integration stack is contained in the arrays X and Y. It is primed with X(2)=0, and H6CM is called to compute Y(2) and the next grid point EPNEXT. The first panel is completed by calculating Y(1) and X(1)=EPNEXT. The panel is then divided in halt and the midpoint is tested to see if it is within TOL=.02 (*i.e.*, 2%) of the linearly interpolated value. If not, the midpoint is inserted in X and Y, and the new top panel [that is X(2)−X(3)] is tested. This continues until convergence is achieved in the top panel. The contributions to the heating and damage are added into the accumulating integrals at statement number 190, and I is decremented so that the process can be repeated for the next panel down. When I decreases to one, the current value of EPNEXT is used to start the next higher E' panel. This loop over panels continues until the entire E' range has been integrated.

The next special case is for tabulated distributions that use E, E', μ ordering in the lab system. The angular part is ignored. a simple loop over the NEP points in $g(E \rightarrow E')$ is started with "DO 415." Trapezoidal integration is used for each panel for both heating and damage (H and D). If ND>0, the first ND entries are discrete energies, and the values of the integrand at those energies are added into H and D. Finally, H and D are copied into EBAR and DAME.

The block of coding starting at statement number 450 is used to compute particle mean energies for the emitted particles from two-body reactions, or to compute the mean recoil energy for a two-body reaction (see IREC>0). The calculation follows Eq. (52). Note that the kinematic factors include AWP, the mass ratio of the emitted particle to the incident particle. The parameter BETA here is the same as R in Eq. (53). If the angular distribution in File 6 is in Legendre form, the heating and damage integrals are performed using a 20-point Gauss-Legendre quadrature (see NQ, QP, and QW). The quadrature loop starts with "DO 470". If the angular distribution is tabulated as $f(\mu)$ *vs* μ, a trapezoidal integration is used for both heating and damage (see "DO 490").

The final option in GETSIX is for lab distributions that use E, μ, E' ordering. See Eq. (56). The inner integrals are computed using trapezoidal integration in the "DO 540" loop. The outer integral over μ also uses trapezoidal integration on the results of the inner integrals for each μ grid point. See "DO 550."

Subroutine H6CM is used by GETSIX to compute the lab distribution $g(E \rightarrow E'_L)$ of Eq. (57) using the CM data in File 6. This subroutine uses H6DDX, SEPH, and H6PSP to retrieve the CM tabulated or phase-space data from the file. These three routines are basically the same as F6CM, F6DDX, SEPE, and F6PSP in GROUPR. See Chapter VIII of this manual for more details.

Subroutine GHEAT is used to correct the heating and damage values accumulated during the pass through the neutron sections. It loops through all of the reactions in File 12 and File 13 using two ENDF-type tapes. One is the input PENDF tape, which is used to retrieve cross sections for use with the photon multiplicities in File 12. The other is a version of the input ENDF tape that has been passed through HCONVR to put the photon data in a standard form (see the GROUPR chapter of this manual for a more detailed discussion of CONVER). This scratch tape is used to retrieve the File 12 and File 13 data. It is very common to find reaction MT3 (nonelastic) in File 12, but this reaction has been removed from the PENDF tape because it is redundant; that is, it is equal to MT1−MT2. Therefore, two passes are made through the File 12 data for MT3, an addition pass with MT1 from the PENDF tape, and a subtraction pass with MT2 from the PENDF tape. Once the desired sections on the two tapes have been found, the subroutines GAMBAR, CAPDAM, and DISGAM are initialized.

The energy loop for GHEAT goes through statement number 190. For each energy, FINDA is used to retrieve the partial KERMA factors as computed from the pass through the neutron files. The yield or cross section is retrieved using GETY1 into the variable Y. If necessary, the corresponding cross section X is retrieved using GETY2. For cases where an energy-dependent Q is available, it is retrieved using TERP1 on the data stored at LQX. The next two lines correct the energy of "primary" photons (LP=2).

For radiative capture represented in File 12 (MT102), GAMBAR, DISGAM, and/or CAPDAM are called to return $\overline{E}_\gamma$ and $\overline{E_\gamma^2}/(2m_R c^2)$ for this photon spectrum or discrete photon and to correct the heating and values in the C array using Eq. (14) and the second line of Eq. (47). The capture contribution to the total photon eV-barns is added into C(NPKK-1) and the photon energy yield is loaded into C(NPKK) for each subsection. When the last subsection is reached, the capture energy check is made using this subtotal. Note that the capture error is loaded into C(NPKK-2) for later use in calculating the kinematic limits for photon energy production.

For other photon-production reactions, the photon eV-barns contributions is subtracted from the energy-balance heating positions, added into the total photon energy value in C(NPKK-1), and added into C(NPKK) for the subtotal for a section with multiple subsections. After all the corrections have been completed for this energy, the revised values are written out using LOADA. The code then moves on to the next reaction and repeats the entire process.

When the reaction loop has been finished, GHEAT checks to see if it can print out a photon energy production check. It can do this if kinematic checks have

been requested and if MT303 was requested in the user's list of partial KERMA calculations. The code reads through the LOADA/FINDA file one more time. For each energy in ELIST, it prints out the total photon eV-barns from C(NPKK-1) and the kinematic limits ELO and EHI. If the limits are violated by more than 10%, alarms consisting of the strings ++++ or ---- are printed after the eV-barns values.

Subroutine GAMBAR is used to compute the mean energy for continuous photon spectra and the photon recoil correction for capture. When called with E=0, it locates the desired section of File 15 on the ENDF tape and reads in the first incident energy. On a normal entry, it checks to see if E is in the range of the data already computed (ELO, EHI, etc.), and if so, it interpolates for the desired results. If not (or on the first real entry), it moves the high data down to the low positions, reads in the next energy from File 15, prepares new values at the new EHI, and checks the energy range again. The photon EBAR is returned by TABBAR, and the corrections to the heating value (ESQB) and damage value (ESQD) from photon production are generated using TABSQR.

Subroutine TABSQR is used to compute the average recoil energy

$$\frac{\overline{E_\gamma^2}}{2M_R c^2} \tag{60}$$

for radiative capture for a tabulated subsection of File 15. The corresponding damage energy is computed at the same time. The basic secondary-energy integral is over the panels defined by the grid points given in File 15. Inside each panel, the integral is computed using a 4-point Gauss-Legendre quadrature.

Subroutine DISGAM is used to compute the $\overline{E_\gamma^2}$ and corresponding damage energy for a discrete capture photon. The rest-mass constant is computed by calling DISGAM once with E=0.

Subroutine HOUT writes the new PENDF tape with the desired thermal MT numbers added. It also correct the directory in MF1/MT451, and it prepares the output listing for printing. The first step is to loop through the partial KERMA factors requested and to write the data on the LOADA/FINDA file onto a scratch tape in ENDF File 3 format. While the first partial is being prepared, the code matches energies in C(1) against the energy list for printing in ELIST. When a match is found, the partial KERMA factors are checked against the kinematic limits, and the variables KLO or KHI are set if any of the comparisons are out of bounds. The KERMA factors, kinematic limits, and error flags are then printed on the output listing. When all of the new sections for File 3 have been prepared, the code updates the contents of the File 1 directory. It then loops through the

rest of the input PENDF tapes copying sections to the output and inserting the new sections in the appropriate places.

If the ENDF/B material includes photon production data, the energy carried away by photons is subtracted from the accumulating KERMA factors in **GHEAT**. The damage cross section is also corrected for photon momentum. First, a scratch file is prepared containing MF=12 and 13. Transition probability arrays are converted using **CONVER** if present. A loop is set up over all reactions in File 12 and File 13. Tabulated energy distributions are integrated using **GAMBAR** (both $\overline{E}_\gamma$ and $\overline{E^2_\gamma}$ are computed for MT=102). In order to avoid requiring MF3/MT3 pointwise data, the code uses MT1–MT2 to compute the nonelastic neutron cross section if it is needed.

K. Error Messages

ERROR IN HEATR***REQUESTED TOO MANY KERMA MTS

6 values in addition to MT=301 are allowed with kinematic checks; otherwise, 25 can be requested. See NPKMAX=28. When checks are requested, the number of words needed is 3*NPK+7; otherwise, NPK+3 are needed.

ERROR IN HEATR***REQUESTED TOO MANY Q VALUES

Limited to 31 only (see NQAMAX).

ERROR IN HEATR***MODE CONVERSION NOT ALLOWED BETWEEN NIN AND NOUT

Both units must be BCD (positive) or blocked binary (negative).

ERROR IN HEATR***TOO MANY MF6 REACTIONS

A maximum of 320 reactions are allowed. See the arrays MT6 and MT6G.

MESSAGE FROM HEATR---MT301 ALWAYS CALCULATED

MT301 was given in the input list of partial KERMA factors. This is not necessary; it is always inserted automatically.

MESSAGE FROM HINIT---MF4 MISSING, ISOTROPY ASSUMED...

Cross sections were found for charged-particle levels in the 600 or 700 series of MT numbers, but no corresponding angular distributions were found. Isotropy is assumed to enable the calculation to proceed, but this evaluation should be upgraded to include the proper sections of File 4.

MESSAGE FROM HINIT---MT18 IS REDUNDANT...

If MT19 is present, MT18 will be ignored.

MESSAGE FROM HINIT---MT19 HAS NO SPECTRUM...

In some evaluations, the partial fission reactions MT19, 20, 21, and 38 are given in File 3, but no corresponding distributions are given. In these cases, it is assumed that MT18 should be used for the fission neutron distributions.

MESSAGE FROM NHEAT---CHANGED Q FROM --- TO ---

The fission Q value is adjusted from the total (non-neutrino) value given in File 3 to a prompt value using the delayed neutron energy from MF1/MT458.

ERROR IN NHEAT***BINDING ENERGY FOR SEQUENTIAL ...

The user must enter special Q values for the ENDF/B evaluation for Be-9. See the discussion in Section VI.G.

ERROR IN NHEAT***STORAGE EXCEEDED

Insufficient storage for diagnostic energy grid. See ELIST(100) and ILMAX.

ERROR IN CONBAR***NKTOT GT NKMAX

More than 12 subsections found. See NKMAX and D1, D2, E1, E2, and LOC, all dimensioned 12.

ERROR IN CONBAR***TABULATED SUBSECTION MUST BE LAST

Required by organizational problems. If the tabulation is last, it can be read with the data for only two energies in memory at one time. This situation is satisfied in versions IV and V. Other evaluations may have to be modified.

ERROR IN CONBAR***INSUFFICIENT STORAGE FOR RAW ENDF DATA

Main container array is too small. Increase the size of array A(12000) in common block /HSTORE/ and adjust NAMAX (These variables are in HEATR).

ERROR IN HGTYLD***ILLEGAL LND

Assumes a maximum of six time groups for delayed neutrons.

ERROR IN HGTYLD***STORAGE EXCEEDED

Increase NWMAX in NHEAT. Currently 2500.

ERROR IN TABBAR***CODED FOR LF=1 AND LF=5 ONLY

Self-explanatory. Should not occur.

MESSAGE FROM SIXBAR---MT nnn DESCRIBES NEUTRONS ONLY...

Some ENDF/B-VI evaluations contain energy-angle distributions for neutrons in File 6 but do not contain the associated recoil distribution. The code has to use the energy-balance method for this reaction, and the damage is set to zero. Most of these cases are for fissionable materials, where these approximations are of little consequence.

MESSAGE FROM SIXBAR---NO DISTRIBUTION FOR MT --- ...

The ENDF-6 format allows the evaluator to describe a subsection of File 6 with "LAW=0"; that is, no distribution is given. Such sections are fine for giving particle yields for gas production and similar applications, but they are not adequate for computing heating and damage.

ERROR IN H6DDX***ILLEGAL LANG

The allowed values for the angular law flag are 1, 2, and 11–15.

ERROR IN H6DDX***TABULATED ANG. DIST. NOT CODED

This version does not handle tabulated angular distributions for the CM system.

ERROR IN SEPH***DOMINANT ISOTOPE NOT KNOWN FOR ...

The Kalbach-systematics approach to computing angular distributions for particle emission requires the separation energy as computed by the liquid drop model. If the target for an evaluation is an element, it is necessary to choose a dominant isotope that adequately represents the effect for this element. Dominant isotopes for materials often evaluated as elements are given in IF statements in this routine. If the desired value is missing, it must be added, and NJOY will have to be recompiled. See the corresponding routines in GROUPR and ACER as well.

ERROR IN H6PSP***3, 4 OR 5 PARTICLES ONLY

The phase-space law is defined for 3, 4, or 5 particles only.

ERROR IN HGTFLE***LAB DISTRIBUTION CHANGED TO CM...

ENDF procedures require that two-body reactions be described in the CM system. Some earlier evaluations claim to be in the lab system. However, they are for relatively heavy targets, and changing to the CM frame will cause only a small change in the results.

ERROR IN HGTFLE***DESIRED ENERGY ABOVE HIGHEST ENERGY GIVEN

Fault in the evaluation.

ERROR IN GETCO***LIMITED TO 21 LEGENDRE COEFFICIENTS

Normal ENDF limit.

ERROR IN GETCO***LAB TO CM CONVERSION NOT CODED

Discrete scattering data should be in the CM system already.

MESSAGE FROM HCONVR---GAMMA PROD PATCH MADE FOR MT ---

This reflects some problems in the old ENDF-III evaluations for Cl and K, which were also carried over to later ENDF versions.

ERROR IN GHEAT***LO=2 NOT CODED

Will not occur since LO=2 data have been transformed to LO=1 format by HCONVR.

MESSAGE FROM GHEAT---NO FILE 12 FOR THIS MATERIAL

Information only.

ERROR IN GAMBAR***REQUESTED ENERGY AT HIGHEST ENERGY GIVEN

Some fault in MF15 data.

ERROR IN GAMBAR***STORAGE EXCEED IN A

Increase container array /HSTORE/ and parameter NAMAX in HEATR.

ERROR IN GAMBAR***REQUESTED ENERGY GT HIGHEST GIVEN

Probably reflects an error in the evaluation.

L. Storage Allocation

Variably dimensioned dynamic storage allocation is used for most data. Storage requirements are dominated by the length of File 5 or File 15 for the evaluation. The size of common /HSTORE/ and the parameter NAMAX in HEATR may be adjusted accordingly. The LOADA/FINDA buffer size NBUF may be decreased or increased at will. The code is currently dimensioned as follows:

100	coarse grid points
31	auxiliary Q values
25	partial KERMAS (7 when kinematic limits are requested)
12000	total storage

M. References

1. M. A. Abdou, C. W. Maynard, and R. Q. Wright, "MACK: A Computer Program to Calculate Neutron Energy Release Parameters (Fluence-to-Kerma Factors) and Multigroup Reaction Cross Sections from Nuclear Data in ENDF Format," Oak Ridge National Laboratory report ORNL-TM-3994 (July 1973).

2. D. W. Muir, "Gamma Rays, Q-Values, and Kerma Factors," Los Alamos Scientific Laboratory report LA-6258-MS (March 1976).

3. R. E. MacFarlane, "Energy Balance of ENDF/B-V," Trans. Am. Nucl. Soc. 33, 681 (1979). See also R. E. MacFarlane, "Energy Balance of ENDF/B-V.2," Minutes of the Cross Section Evaluation Working Group (available from the National Nuclear Data Center, Brookhaven National Laboratory, Upton, NY) (May 1984).

4. T. A. Gabriel, J. D. Amburgy, and N. M. Greene, "Radiation-Damage Calculations: Primary Knock-On Atom Spectra, Displacement Rates, and Gas Production Rates," *Nucl. Sci. Eng.* 61, 21 (1976).

5. D. G. Doran, "Neutron Displacement Cross Sections for Stainless Steel and Tantalum Based on a Linhard Model," *Nucl. Sci. Eng.* 49, 130 (1972).

6. L. R. Greenwood and R. K. Smither, "Displacement Damage Calculations with ENDF/B-V," in Proceedings of the Advisory Group Meeting on Nuclear Data for Radiation Damage Assessment and Reactor Safety Aspects, October 12-16, 1981, IAEA, Vienna, Austria (October 1981).

7. M. T. Robinson, in *Nuclear Fusion Reactors* (British Nuclear Energy Society, London, 1970).

8. J. Lindhard, V. Nielsen, M. Scharff, and P. V. Thomsen, *Kgl. Dansk, Vidensk. Selsk, Mat-Fys. Medd.* 33 (1963).

9. P. F. Rose and C. L. Dunford, Eds., "ENDF-102, Data Formats and Procedures for the Evaluated Nuclear Data File, ENDF-6," Brookhaven National Laboratory report BNL-NCS-44945 (July 1990).

10. R. E. MacFarlane, D. W. Muir, and F. M. Mann, "Radiation Damage Calculations with NJOY," *J. Nucl. Materials* 122 and 123, 1041 (1984).

11. R. E. MacFarlane and D. G. Foster, Jr., "Advanced Nuclear Data for Radiation Damage Calculations," *J. Nucl. Materials* 122 and 123, 1047 (1984).

VII. THERMR

The THERMR module generates pointwise neutron scattering cross sections in the thermal energy range and adds them to an existing PENDF tape. The cross sections can then be group-averaged, plotted, or reformatted in subsequent modules. THERMR works with either the original ENDF/B-III thermal format[1] and data files[2] (which were also used for ENDF/B-IV and -V), or the new ENDF-6 format.[3] Coherent elastic cross sections are generated for crystalline materials using either parameters given in an ENDF-6 format evaluation or an extended version of the method of HEXSCAT[4]. Incoherent elastic cross sections for non-crystalline materials such as polyethylene and ZrH can be generated either from parameters in an ENDF-6 format file or by direct evaluation using parameters included in the THERMR coding. Inelastic cross sections and energy-to-energy transfer matrices can be produced for a gas of free atoms, or for bound scatterers when ENDF $S(\alpha,\beta)$ scattering functions are available. This function has previously been performed using FLANGE-II.[5] THERMR has the following features:

- The energy grid for coherent elastic scattering is produced adaptively so as to represent the cross section between the sharp Bragg edges to a specified tolerance using linear interpolation.
- The secondary energy grid for inelastic incoherent scattering is produced adaptively so as to represent all structure with linear interpolation.
- Incoherent cross sections are computed by integrating the incoherent matrix for consistency.
- Free-atom incoherent scattering is normalized to the Doppler broadened elastic scattering cross section in order to provide an approximate representation of resonance scattering and to preserve the correct total cross section.
- Discrete angle representations are used to avoid the limitations of Legendre expansions.
- Hard-to-find parameters for the ENDF/B-III evaluations are included in the THERMR code for the user's convenience.
- ENDF-6 format files can be processed. This gives the evaluator more control over the final results, because all parameters needed to compute the cross sections are contained in the file.

A. Coherent Elastic Scattering

In crystalline solids consisting of coherent scatters—for example, graphite—the so-called "zero-phonon term" leads to interference scattering from the various planes of atoms of the crystals making up the solid. There is no energy loss in

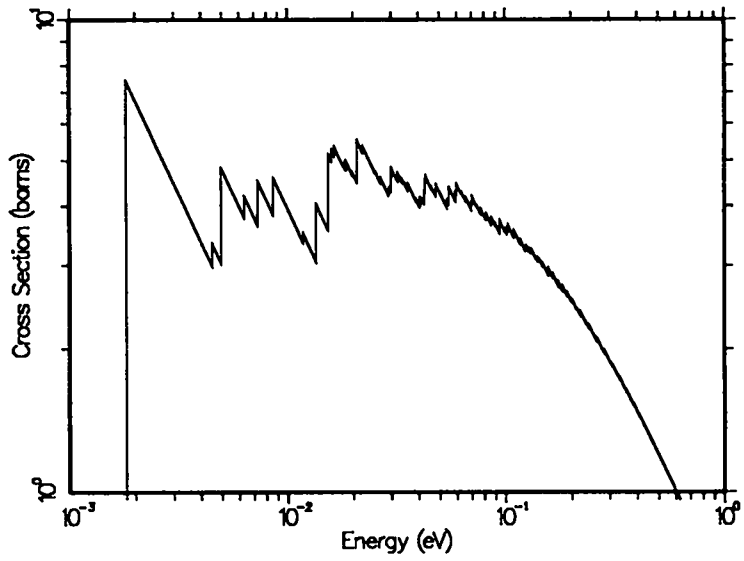


Figure 1: Typical behavior of the coherent elastic scattering cross section for a crystalline material as computed by THERMR. This cross section is for graphite at 300 K.

such scattering, and the ENDF term for the reaction is coherent elastic scattering. The cross section may be represented as follows:

$$\sigma^{\text{coh}}(E, E', \mu) = \frac{\sigma_c}{E} \sum_{E_i<E} f_i \, \mathrm{e}^{-2WE_i} \, \delta(\mu - \mu_0) \, \delta(E - E') \, , \tag{1}$$

where

$$\mu_0 = 1 - \frac{E_i}{E} \, , \tag{2}$$

and the integrated cross section is given by

$$\sigma^{\text{coh}} = \frac{\sigma_c}{E} \sum_{E_i>E} f_i \, \mathrm{e}^{-2WE_i} \, . \tag{3}$$

In these equations, E is the incident neutron energy, E' is the secondary neutron energy, μ is the scattering cosine in the laboratory (LAB) reference system, σ_c is the characteristic coherent cross section for the material, W is the effective Debye-Waller coefficient (which is a function of temperature), the E_i are the so-called "Bragg edges", and the f_i are related to the crystallographic structure factors.

It can be seen from Eq. (3) and the example in Fig. 1 that the coherent elastic cross section is zero before the first Bragg edge, E_1 (typically about 2 to 5 meV). It then jumps sharply to a value determined by f_1 and the Debye-Waller term.

At higher energies, the cross section drops off as $1/E$ until $E=E_2$. It then takes another jump and resumes its $1/E$ drop-off. The sizes of the steps in the cross section gradually get smaller, and at high energies there is nothing left but an asymptotic $1/E$ decrease (typically above 1 to 2 eV).

For evaluations in the new ENDF-6 format, the section MF7/MT2 contains the quantity $E\sigma^{\rm coh}(E)$ as a function of energy and temperature. The energy dependence is given as a histogram with breaks at the Bragg energies. The cross section is easily recovered from this representation by dividing by E. The E_i are easily found as the tabulation points of the function, and the f_i for a point can be obtained by subtracting the value at the previous point.

For evaluations using the older ENDF/B-III thermal format, it is necessary to compute the E_i and f_i in THERMR. The methods used are based on HEXSCAT[4] and work only for the hexagonal materials graphite, Be, and BeO. The Bragg edges are given by

$$E_i = \frac{\hbar^2 \tau_i^2}{8m} , \tag{4}$$

where τ_i is the length of the vectors of one particular "shell" of the reciprocal lattice, and m is the neutron mass. The f_i factors are given by

$$f_i = \frac{\pi^2 \hbar^2}{2mNV} \sum_{\rm shell} |F(\tau)|^2 , \tag{5}$$

where the shell sum extends over all reciprocal lattice vectors of the given length, N is the number of atoms in the unit cell, and F is the crystallographic structure factor. The calculation works by preparing a sorted list of precomputed τ_i and f_i values. As τ_i gets large, the values of τ_i get more and more closely spaced. In order to save storage and run time, a range of τ values can be lumped together to give a single effective τ_i and f_i. This device washes out the Bragg edges at high energies while preserving the proper average cross section and angular dependence. The current grouping factor is 5% (see EPS in SIGCOH).

Lattice constants (given in SIGCOH for graphite, Be, and BeO), form factor formulas (see FORM), Debye-Waller coefficients, and methods for computing reciprocal lattice vectors were borrowed directly from HEXSCAT.

The energy grid for E is obtained adaptively (see COH). A panel extending from just above one Bragg edge to just below the next higher edge is subdivided by successive halving until linear interpolation is within a specified fractional tolerance (TOL) of the exact cross section at every point. This procedure is repeated for every

panel from the first Bragg edge to the specified maximum energy for the thermal treatment (**EMAX**).

The code usually computes and writes out the cross section of Eq. (3), the average over μ of Eq. (1), which is sometimes called the P_0 cross section. Subsequent modules can deduce the correct discrete scattering angles μ_0 from the location of the Bragg edges E_i and the factors f_i from the cross-section steps at the Bragg edges (see GROUPR). Legendre cross sections can also be computed by making a small change to the code. It is not necessary to give the P_1, P_2, and P_3 cross sections explicitly as was done in some earlier codes or in File 4 of the ENDF thermal tapes.

B. Incoherent Inelastic Scattering

In ENDF/B notation, the thermal incoherent scattering cross section is given by

$$\sigma^{\text{inc}}(E,E',\mu) = \frac{\sigma_b}{2kT}\sqrt{\frac{E'}{E}}\,e^{-\beta/2}S(\alpha,\beta)\,, \tag{6}$$

where E is the initial neutron energy, E' is the energy of the scattered neutron, μ is the scattering cosine in the laboratory system, σ_b is the characteristic bound incoherent cross section for the nuclide, T is the Kelvin temperature, β is the dimensionless energy transfer,

$$\beta = \frac{E'-E}{kT}\,, \tag{7}$$

α is the dimensionless momentum transfer,

$$\alpha = \frac{E'+E-2\mu\sqrt{EE'}}{AkT}\,, \tag{8}$$

k is Boltzmann's constant, and A is the ratio of the scatter mass to the neutron mass. The bound scattering cross section is usually given in terms of the characteristic free cross section, σ_f,

$$\sigma_b = \sigma_f\frac{(A+1)^2}{A^2}\,. \tag{9}$$

The scattering law $S(\alpha,\beta)$ describes the binding of the scattering atom in a material. For a free gas of scatterers with no internal structure

$$S(\alpha,\beta) = \frac{1}{\sqrt{4\pi\alpha}}\exp\left\{-\frac{\alpha^2+\beta^2}{4\alpha}\right\}\,. \tag{10}$$

For binding in solids and liquids, $S(\alpha,\beta)$ for a number of important moderator materials is available in ENDF/B File 7 format. The scattering law is given as tables of S versus α for various values of β. Values of S for other values of α and β can be obtained by interpolation. The scattering law is normally symmetric in β and only has to be tabulated for positive values, but for materials like ortho-hydrogen and parahydrogen of interest for cold moderators at neutron scattering facilities, this is not true. These kinds of materials are identified by the ENDF-6 LASYM option, and THERMR assumes that the scattering law is given explicitly for both positive and negative values of β.

If the α or β required is outside the range of the table in File 7, the differential scattering cross section can be computed using the short collision time (SCT) approximation

$$\sigma^{\mathrm{SCT}}(E,E',\mu) = \frac{\sigma_b}{2kT}\frac{\sqrt{E'/E}}{\sqrt{4\pi\,|\alpha|\,T_{\mathrm{eff}}/T}}\exp\left\{-\frac{(|\alpha|-\beta)^2}{4\,|\alpha|}\frac{T}{T_{\mathrm{eff}}}-\frac{\beta+|\beta|}{2}\right\}, \quad (11)$$

where T_{eff} is the effective temperature for the SCT approximation. These temperatures are available[2] for the older ENDF/B-III evaluations; they are usually somewhat larger than the corresponding Maxwellian temperature T. For the convenience of the user, the values of T_{eff} for the common moderators are included as defaults (see input instructions). For the new ENDF-6 format, the effective temperatures are included in the data file. However, there is a complication. Some evaluations give $S(\alpha,\beta)$ for a molecule or compound (in the ENDF/B-III files, these cases are BeO and C_6H_6). The corresponding SCT approximation must contain terms for both atoms. The two sets of bound cross sections and effective temperatures are included in the data statements in THERMR, and they can be given in the new ENDF-6 format if desired.

THERMR expects the requested temperature T to be one of the temperatures included on the ENDF/B thermal file, or within a few degrees of that value (296 K is used if 300 K is requested). Intermediate temperatures should be obtained by interpolating between the resulting cross sections and not by interpolating $S(\alpha,\beta)$.

The secondary energy grid for incoherent scattering is obtained adaptively (see CALCEM). A stack is first primed with the point at zero and the first point above zero that can be derived from the positive and negative values of β from the evaluator's β grid using Eq. (7). (For free-gas scattering, the β grid is taken to have the two entries 0. and 25.). This interval is then subdivided by successive halving until the cross section obtained by linear interpolation is within the specified tolerance of the correct cross section (from SIGL). The next highest energy derived from

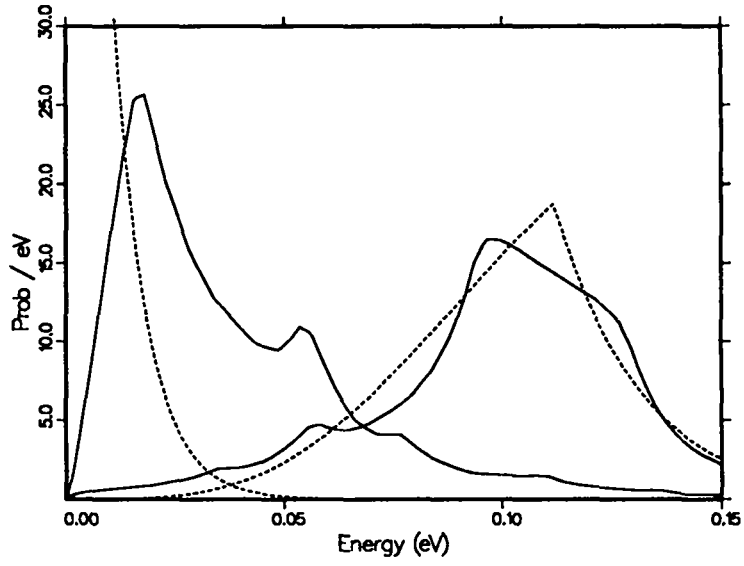


Figure 2: Adaptive reconstruction of two of the emission curves for graphite at 300 K (E=.00016 eV to the left, and E=.1116 eV to the right). Note the presence of excitation features from the phonon frequency spectrum for both upscatter and downscatter. The breaks in the curves are due to β interpolation in $S(\alpha,\beta)$ and not to the tolerances in the reconstruction process. The dashed curves are the corresponding free gas results.

the β grid is then calculated, and the subdivision process is repeated for this new interval. This process is continued until the β grid has been exhausted. Excess points with zero cross section are removed before writing the spectrum into File 6. This procedure is sure to pick up all the structure in the evaluation; giving points related to the β grid avoids excessive work in trying to fit sharp corners introduced by breaks in the interpolation of $S(\alpha,\beta)$. Figure 2 shows how the procedure picks up features resulting from the sharp excitation features in the graphite phonon frequency distribution. The result of this adaptive reconstruction is easily integrated by the trapezoid rule to find the incoherent cross section at energy E.

The cross section for one particular $E \rightarrow E'$ is the integral over the angular variable of Eq. (6). The angular dependence is obtained by adaptively subdividing the cosine range until the actual angular function (see SIG) is represented by linear interpolation to within a specified tolerance. The integral under this curve is used in calculating the secondary-energy dependence as described above. Rather than providing the traditional Legendre coefficients, THERMR divides the angular range into equally probable cosine bins and then selects the single cosine in each

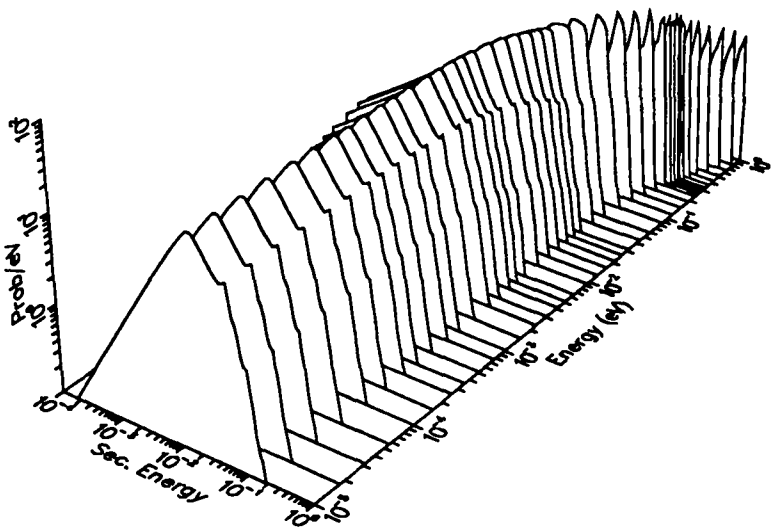


Figure 3: Neutron distribution for incoherent inelastic scattering from graphite (T = 300 K).

bin that preserves the average cosine in the bin. These equally probable cosines can be converted to Legendre coefficients easily when producing group constants, and they are suitable for direct use in Monte Carlo codes. For strongly peaked functions, such as scattering for $E \gg kT$ when the result begins to look "elastic", all the discrete angles will be bunched together near the scattering angle defined by ordinary kinematics. This behavior cannot be obtained with ordinary P_3 Legendre coefficients. Conversely, if such angles are converted to Legendre form, very high orders can be used. If a direct calculation of Legendre components is desired, reverse the sign of **NNL** in **CALCEM**.

The incident energy grid is currently stored directly in the code (see **EGRID** in **CALCEM**). The choice of grid for $\sigma^{\text{inc}}(E)$ is not critical since the cross section is a slowly varying function of E. However, the energy grid would seem to be important for the emission spectra. In order to demonstrate the problem, two perspective plots of the full energy distribution of incoherent inelastic scattering from graphite at 300 K are shown in Figs. 3 and 4. The second plot is simply an expansion of the high-energy region of the distribution.

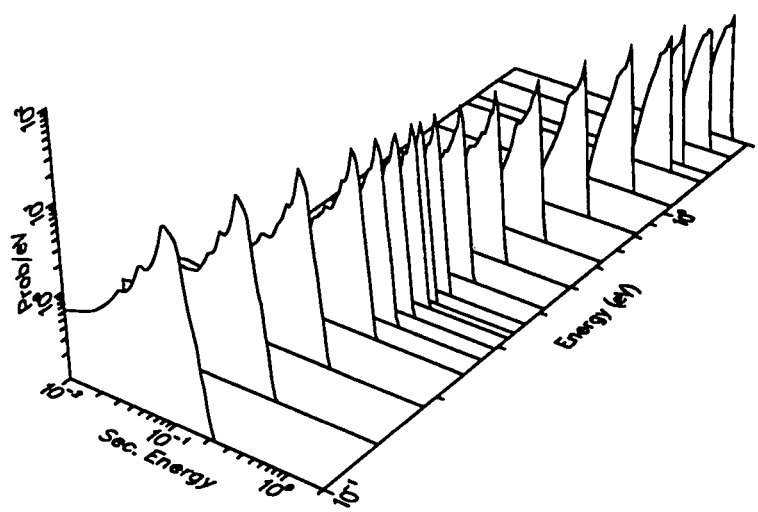


Figure 4: Expanded view of the high-energy region of the graphite incoherent inelastic distribution.

It is clear that $\sigma^{\text{inc}}(E,E')$ for one value of E' is a very strongly energy-dependent function for the higher incident energies. However, as shown in Fig. 4, the shape of the secondary energy distribution changes more slowly, with the peak tending to follow the line $E'=E$. This behavior implies that a relatively coarse incident energy grid might prove adequate if a suitable method is used to interpolate between the shapes at adjacent E values. One such interpolation scheme is implemented in GROUPR. The use of discrete angles is especially suitable for this interpolation scheme.

Strictly speaking, the scattering law for free-gas scattering given in Eq. (10) is only applicable to scatterers with no internal structure. However, many materials of interest in reactor physics have strong scattering resonances in the thermal range (for example, ^{240}Pu and ^{135}Xe). The Doppler broadened elastic cross section produced by BROADR is formally correct for a gas of resonant scatterers, but the cross section resulting from Eq. (10) is not. In order to allow for resonance scattering in a way that at least provides the correct total cross section, THERMR renormalizes the free-atom scattering to the broadened elastic cross section. The secondary energy distribution will still be incorrect.

C. Incoherent Elastic Scattering

In hydrogenous solids, there is an elastic (no energy loss) component of scattering arising from the zero-phonon term that can be treated in the incoherent approximation because of the large incoherent cross section of hydrogen. The ENDF term for this process is incoherent elastic scattering, and it is found in the materials polyethylene and zirconium hydride. The differential cross section is given by

$$\sigma^{\text{iel}}(E,E',\mu) = \frac{\sigma_b}{2} e^{-2WE(1-\mu)}\,\delta(E-E')\ , \tag{12}$$

where σ_b is the characteristic bound cross section and W is the Debye-Waller coefficient. The energy grid of the elastic cross section is used for E, and the average cross section and equally probable angles are computed using

$$\sigma^{\text{iel}}(E) = \frac{\sigma_b}{2}\left\{\frac{1-e^{-4WE}}{2WE}\right\}\ , \tag{13}$$

and

$$\begin{aligned}\bar{\mu}_i &= \frac{N}{2WE}\left[e^{-2WE(1-\mu_i)}(2WE\mu_i-1)\right. \\ &- \left. e^{-2WE(1-\mu_{i-1})}(2WE\mu_{i-1}-1)\right]/(1-e^{-4WE})\ ,\end{aligned} \tag{14}$$

where

$$\mu_i = 1 + \frac{1}{2WE} \ln \left[\frac{1 - e^{-4WE}}{N} + e^{-2WE(1-\mu_{i-1})} \right] \tag{15}$$

is the upper limit of one equal probability bin and $\bar{\mu}_i$ is the selected discrete cosine in this bin. Here N is the number of bins and μ_0 is -1.

The characteristic bound cross sections and the Debye-Waller coefficients can be read from MF7/MT2 of an ENDF-6 format evaluation, or obtained directly from data statements in the code for the older format.

D. Coding Details

The THERMR procedure begins with the reading of the user's input. The required ENDF tape (**NENDF**) is only used for MF7 data; it can be set to zero if only free-gas scattering is needed. Similarly, **MATDE** is the material number for the File 7 tape and can be set to zero for free-atom problems. The ENDF File 7 format only gives "$M_0 \sigma_{f0}$", the product of the free scattering cross section for the principal scatterer and the number of principal scatterer atoms in the molecule. As a result, THERMR needs the parameter **NATOM** to obtain the effective microscopic cross section (for example, for H in H_2O, use **NATOM=2**). For ENDF/B-III format files, default parameters are supplied for mixed moderators (BeO and benzine) and effective temperatures, if needed.

Continuing, THERMR finds the desired material on the input PENDF and ENDF tapes. It will automatically loop over **NTEMP** materials on **NIN**. The input tape must have been through BROADR. The elastic cross section at the current temperature is saved on a **LOADA/FINDA** scratch file to be used for normalizing free-atom scattering if necessary. For ENDF-6 format materials, the parameters for the elastic calculation are read in using **RDELAS**. Next, **THERMR** computes elastic and/or inelastic cross sections by calls to **COH**, **IEL**, and **CALCEM**. Finally, the results are written onto the output PENDF tape by **TPEND**.

Some alteration of ENDF/B formats and conventions was required to accommodate thermal cross sections. The incoherent inelastic cross sections fit well into MF=3 using MT=**MTREF** (see user input). The coherent or incoherent elastic cross section (if present) uses **MTREF+1**. Other modules of NJOY expect that thermal MT numbers will be between 221 and 250. The incoherent energy-to-energy matrix is stored in MF6 (coupled angle-energy distributions). Before the introduction of the ENDF-6 format, the ENDF File 6 formats were not well-suited to this application because secondary angle and energy were not tightly coupled as required by the physics of the problem. Therefore, three new formats were defined

for File 6: LTT=5 for discrete-angle inelastic transfer cross sections, LTT=6 for discrete-angle elastic data, and LTT=7 for coherent elastic reactions.

The format for LTT=5 follows in the notation of ENDF-102[1]:

```
MAT,6,MT [ ZA, AWR, 0, LTT, 0, 0 ] CONT  LTT=5
MAT,6,MT [ T, 0., 0, 1, NNE / NNE, 2 ] TAB2
MAT,6,MT [ 0., EN, 0, 0, NEP*(NL+1), NL+1 /
           EP(1), PP(1), EPM(1,1), ...
           EP(2), PP(2), ... ] LIST
   ... repeat the LIST for the NNE values of EN ...
MAT,6,0  [ 0., 0., 0, 0, 0, 0 ] SEND
```

There is a list record for each of the NNE values of incident energy. Each list record gives the normalized secondary energy distribution as NEP value of PP vs. E', and for each value of E', the record gives NL equally probable cosines EPM.

Similarly, the format for LTT=6 is as follows:

```
MAT,6,MT [ ZA, AWR, 0, LTT, 0, 0 ] CONT  LTT=6
MAT,6,MT [ T, 0., 0, 1, NNE / NNE, 2 ] TAB2
MAT,6,MT [ 0., EN, 0, 0, NU+2, NU+2 /
           EN, 1., U(1), U(2), ... ] LIST
   ... repeat the LIST for the NNE values of EN ...
MAT,6,0  [ 0., 0., 0, 0, 0, 0 ] SEND
```

Here, there is just a set of NU equally probable cosines given for each incident energy. Note that this format was designed to look like that for LTT=5 with NEP=1.

Finally, the format for LTT=7 is as follows:

```
MAT,6,MT [ ZA, AWR, 0, LTT, 0, 0 ] CONT  LTT=7
MAT,6,MT [ 0,, 0., 0, 0, 0, NBRAGG ] CONT
MAT,6,0  [ 0., 0., 0, 0, 0, 0 ] SEND
```

In this case, all the important information is in File 3 under MT=MTREF+1. For convenience, the number of Bragg edges used is given here in File 6 as NBRAGG.

In subroutine COH, the energy grid is determined adaptively and stored into the same LOADA/FINDA scratch file used for the elastic cross section. The elastic

cross section is converted to the coherent grid using Lagrangian interpolation (see TERP). The structure of the record stored on the scratch file is

[energy / static elastic / incoherent inelastic / coherent elastic].

Coherent cross sections at a given energy E are computed by SIGCOH. If this is the first entry (E=0) for an ENDF-III type material, the appropriate lattice constants are selected and the Debye-Waller coefficient is obtained for the desired temperature by interpolation. Then the reciprocal lattice wave vectors and structure factors are computed, sorted into shells, and stored for later use. On a normal entry (E>0), the stored list is used to compute the cross section. For ENDF-6 format materials, the initialization step is used to organize the data already read from MF7/MT2 by RDELAS, and subsequent entries are used to compute the cross section.

Incoherent elastic cross sections are computed in subroutine IEL. The appropriate bound cross sections and Debye-Waller coefficients are either extracted from the data already read from an ENDF-6 format MF7/MT2 by RDELAS, or they are extracted from data statements in IEL and then adjusted to the specified temperature using TERP or TERPA. The angle-integrated cross section is computed analytically on the grid of the static elastic cross section and written back onto the LOADA/FINDA scratch file in the same slot used for coherent elastic as described above (both never occur simultaneously in the same material). The discrete equally probable cosines are cast into LTT=7 format and written onto a scratch tape for use by TPEND.

Incoherent cross sections and matrices are generated in CALCEM. On the first entry, the ENDF/B scattering law is read in or parameters are set for free-atom scattering. For ENDF-6 files, the effective temperatures for the SCT approximation are read in. For the older formats, these numbers were either read in or set to default values during the user input process. On subsequent entries, the adaptive loop to determine the secondary energy grid is carried out. The required cross sections and discrete cosines are returned by SIGL, which uses SIG to compute the differential cross sections. Because the spectrum curve will have discontinuities in slope at energies corresponding to the break points of the β grid, it is important to use these energies as the starting points for the adaptive reconstruction. The first panel starts at $E' = 0$ and ends at the first energy greater than zero that can be derived from the β grid. This will normally be a negative β value corresponding to E'<E. These two energies and their corresponding cross sections are loaded into an inverted stack like the one used in RECONR. Next, the top interval in the stack is divided in half, and new cross sections are computed at this midpoint. If

the new cross section is not within the desired tolerance of the value obtained by linear interpolation between the adjacent points, the new value is inserted into the stack. Otherwise, the top value in the stack is converged and can be saved to the location where the spectrum is accumulating. Each time the stack gets down to a single element, a new point is calculated from the next β value in the evaluator's β grid, and the subdivision process is continued. During this reconstruction process, the integrated cross section is computed by adding in each trapezoid. In addition, note is taken of the last nonzero cross section value in order to remove excess zero values from the end of the record. The σ vs. E' curve is complete when the β grid has been exhausted (the highest positive value). The result is put directly into the modified MF=6 format and written onto a scratch file.

When all the desired incident energies have been processed, the incoherent cross section is calculated on the File 3 energy grid by interpolating in the table of values computed by the reconstruction process. The results are stored on the **LOADA/FINDA** tape. If free-atom scattering has been selected, the elastic cross section is stored in the incoherent slot.

Incoherent inelastic scattering cross sections and discrete cosines are computed in **SIGL**. The stack for the adaptive reconstruction of the angular distribution for a given $E \rightarrow E'$ is primed with $\mu=-1$, $\mu=+1$, and the angle for static scattering. The top interval on the stack is subdivided by halving until the actual cross section computed by **SIG** is within a specified tolerance of a linear interpolate. As each panel is converged, its area is added to the accumulating cross section. On convergence, the fraction of the cross section corresponding to each equally probable bin is computed, and the linearization process is repeated to find the bin boundaries and discrete cosines. Note that Legendre coefficients can be computed in this routine from the discrete cosines.

Subroutine **SIG** is used to compute the actual double-differential cross section for a given value of E, E', and μ. This is done using $S(\alpha,\beta)$ (with the possible use of the SCT approximation for large values of α or β), or using the free-gas scattering law. Normally, $s(\alpha,\beta)$ is symmetrical in β and results are extracted from the table at **ISAB** using $|\beta|$. However, this is not true for materials like orthohydrogen and parahydrogen. In these cases, the LASYM parameter is set, and explicit $S(\alpha,\beta)$ values are given for both negative and positive β. For liquids, the presence of diffusion leads to a singularity for $\beta = 0$ and small α (the quasi-elastic scattering peak). The normal ENDF interpolation laws do not represent $S(\alpha,\beta)$ well in this region. Therefore, THERMR tries to determine if the material is a liquid by looking at the small-α behavior of the $\beta = 0$ curve (see **CLIQ**). It

then extrapolates low-β curves to low-α values using a β^2/α law.

Finally, **TPEND** is called to prepare the output tape. The File 1 directory is updated to account for the new sections that are being added. File 3 is located and the cross sections stored on the **LOADA/FINDA** scratch file are retrieved, formatted, and written to the output tape. Note that the elastic cross section in MT=2 and the total cross section in MT=1 are not changed from their static values, nor is the union grid updated. As a result, MT=221–250 must be considered supplemental. Subsequent modules could ignore them or use them in place of the static values. Also note that it is possible to run THERMR several times with different values of **MTREF**. The result would be one PENDF tape containing static cross sections and cross sections for several different binding states that can be selected at will (for example, MT2 for static hydrogen, MT221 for free hydrogen, MT222 for hydrogen in water, and MT223 and MT224 for hydrogen in polyethylene, all on one PENDF tape).

File 6 matrices are read from a scratch tape (**NSCR**) in ENDF format, normalized, and written back onto the final tape. Since free incoherent scattering was set equal to elastic scattering in **CALCEM**, the approximate resonance correction of the matrix is now complete.

E. Using the ENDF/B-III Thermal Data Files

The thermal data files originally prepared for ENDF/B-III were also used for ENDF/B-IV and ENDF/B-V.* Table 1 summarizes the contents of these data tapes.

For most of these evaluations, THERMR will produce cross sections appropriate for the major scattering atom bound in a particular material, for example, hydrogen bound in water, or Zr bound in ZrH. In these cases, the cross sections are combined later (for example, hydrogen bound in water would be combined with free-gas oxygen, and Zr bound in ZrH would be combined with H bound in ZrH). The treatment to be used for the secondary scattering atoms for each evaluation is indicated in the table.

In two cases—BeO and Benzine—the scattering laws $S(\alpha,\beta)$ for the two component atoms have been combined into a single scattering law normalized to be used with the cross section of the primary scattering atom. For these cases, THERMR produces a cross section for the molecule or compound directly. In making a

*The data files and the reference manual[2] are available from the National Nuclear Data Center, Brookhaven National Laboratory, Upton, NY 11973. Request Tapes 320 through 325 and report ENDF-269.

Table 1: Moderator Materials on the ENDF/B-III Thermal Data Files Showing Their MAT Numbers and Distribution Tape Numbers

Material	MAT	Tape	MTs	elastic	secondary
Be	1064	321	231,232	coh	
BeO	1099	321	233,234	coh	none
C(graphite)	1065	322	229,230	coh	
C(polyethylene)	1114	322	223	iel	free C
C_6H_6	1095	325	227		none
$D(D_2O)$	1004	320	228		free O
$H(H_2O)$	1002	320	222		free O
$Zr(ZrH_n)$	1096	323	235,236	iel	1097
$H(ZrH_n)$	1097	323	225,226	iel	1096

macroscopic cross section for BeO, the user would multiply the thermal BeO cross section from THERMR by the atomic density of Be, taking care not to add any additional thermal contribution for the oxygen.

THERMR labels the thermal cross sections that it generates with specially defined MT numbers. The particular numbers shown in the table are recognized by the reaction naming logic in MATXSR. Note that two numbers are defined for materials that have both inelastic and elastic components; the first number is for inelastic, and the second for elastic.

F. Input Instructions

The following input instructions have been copied from the comment cards in THERMR:

```
*---INPUT SPECIFICATIONS (FREE FORMAT)-------------------------*
*                                                              *
*  CARD 1                                                      *
*     NENDF      ENDF/B TAPE FOR MF7 DATA                      *
*     NIN        OLD PENDF TAPE                                *
*     NOUT       NEW PENDF TAPE                                *
*  CARD 2                                                      *
*     MATDE      MATERIAL DESIRED ON ENDF TAPE                 *
*     MATDP      MATERIAL DESIRED ON PENDF TAPE                *
*     NBIN       NUMBER OF EQUI-PROBABLE ANGLES                *
*     NTEMP      NUMBER OF TEMPERATURES                        *
*     IINC       INELASTIC OPTIONS                             *
*                   0     NONE                                 *
*                   1     COMPUTE AS FREE GAS                  *
*                   2     RESERVED                             *
```

```
*                  3     RESERVED                                    *
*                  4     READ S(A,B) AND COMPUTE MATRIX              *
*     ICOH       ELASTIC OPTIONS                                     *
*                  0     NONE                                        *
*                  1     GRAPHITE                                    *
*                  2     BERYLLIUM                                   *
*                  3     BERYLLIUM OXIDE                             *
*                 11     POLYETHYLENE                                *
*                 12     H(ZRH)                                      *
*                 13     ZR(ZRH)                                     *
*     NATOM      NUMBER OF PRINCIPAL ATOMS                           *
*     MTREF      MT FOR INELASTIC REACTION (201-250 ONLY)            *
*     IPRINT     PRINT OPTION (0=MINIMUM, 1=MAXIMUM,                 *
*                2=MAX. NORMAL + INTERMEDIATE RESULTS)               *
*                (DEFAULT=0)                                         *
*  CARD 3                                                            *
*     TEMPR      TEMPERATURES (KELVIN)                               *
*  CARD 4                                                            *
*     TOL        TOLERANCE                                           *
*     EMAX       MAXIMUM ENERGY FOR THERMAL TREATMENT                *
*                (FOR TEMPERATURES GREATER THAN 3000,                *
*                EMAX AND THE ENERGY GRID ARE SCALED BY              *
*                TEMP/300.  FREE GAS ONLY.)                          *
*                                                                    *
**********************************************************************
```

The following sample problem illustrates the production of thermal cross sections for hydrogen in water.

```
0
5
THERMR
20 21 22 /
1002 1301 8 1 4 0 2 222 0
300.
.01 4.6
STOP
```

It is assumed that ENDF/B Tape 320 is mounted on unit 20, and that a previously prepared PENDF tape for ^{1}H (MAT1301) is mounted on unit 21. The thermal cross sections for hydrogen in water will be written on unit 24 using MTREF=222. Note that the parameter NATOM is set to 2 because the water molecule H_2O contains two hydrogen atoms. In addition, ICOL=0 for hydrogen in water. The higher-energy parts of the neutron emission curves for this example are shown in Fig. 5. The sharp peak at E=E' is quasi-elastic scattering broadened by diffusion.

A calculation of both free and graphite cross sections for ENDF/B-V carbon would go as follows:

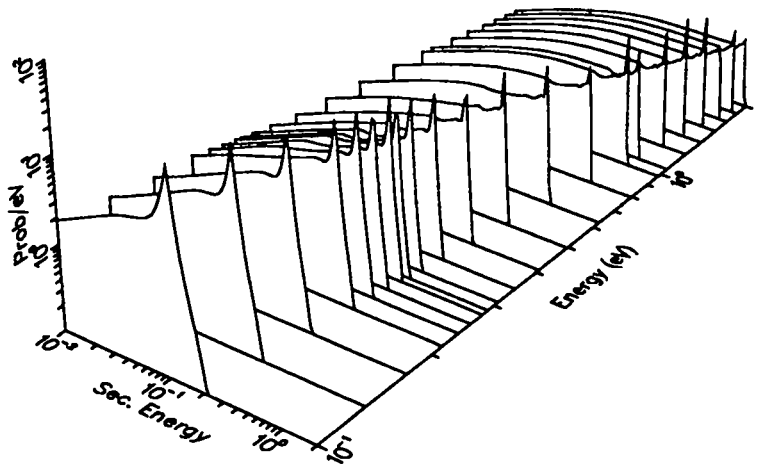


Figure 5: Expanded view of the high-energy region of the incoherent inelastic distribution for hydrogen bound in water. Note the sharp quasi-elastic peak at $E=E'$. The slightly ragged representation of the peak is caused by excessive thinning in PLOTR.

```
0
5
RECONR
20 22 / TAPE20 IS ENDF/B-V TAPE 511
/
1306 1/
.005/
*6-C-NAT FROM TAPE 511*/
0/
BROADR
22 23
1306 1/
.005/
300.
0/
THERMR
0 23 24
0 1306 8 1 1 0 1 221 0
300.
.01 1.2
THERMR
26 24 25 / TAPE26 IS ENDF/B TAPE 322
1065 1306 8 1 4 1 1 229 0
300.
.01 1.2
STOP
```

First, ENDF/B-V Tape 511 must be mounted on unit 20. At Los Alamos, this is done by copying it to a file named TAPE20 in the user's local file space. Similarly, ENDF/B Tape 322 must be copied to TAPE26. Next, RECONR is run to linearize the evaluation, and BROADR is run to prepare 300 K cross sections. The first THERMR run is for free-gas scattering (MTREF=221), and the second run is for carbon bound in graphite (MTREF=229). Note that NATOM is now 1, and that 8 discrete angles were requested in both cases. For graphite, ICOH is set to 1 in order to request the calculation of coherent elastic scattering like that shown in Fig. 1. The coherent results will use MF=3 and MT=230. Distributions will be calculated to 1% accuracy for energies up to 1.2 eV. The final PENDF tape will be TAPE25.

G. Error Messages

ERROR IN THERMR***NIN=0

An input PENDF tape is required.

ERROR IN THERMR***MODE CONVERSION NOT ALLOWED

NIN and NOUT must both be binary or both be coded.

ERROR IN THERMR***ILLEGAL REFERENCE MT

Restricted to MT=201–250.

ERROR IN THERMR***DESIRED MATERIAL NOT ON PENDF TAPE

Check input instructions against contents of thermal tape.

ERROR IN THERMR***DESIRED TEMPERATURE NOT ON TAPE

Check input instructions against contents of thermal tape.

ERROR IN THERMR***IINC=2 NOT PROGRAMMED

ERROR IN THERMR***IINC=3 NOT PROGRAMMED

These were intended for future options.

ERROR IN RDELAS***DESIRED TEMP NOT FOUND

The temperatures requested in THERMR must be the same as the temperatures on the input PENDF tape.

ERROR IN COH***TOO MANY LEGENDRE ORDERS

The code currently computes only P_0, but NL=1 in COH can be changed if desired. Code is currently limited to 6 (P_5). If more coefficients are desired, increase NLMAX and the dimensions of the variables S, EJ, and EX in COH, CALCEM, and TPEND.

ERROR IN SIGCOH***STORAGE EXCEEDED

Not enough room for lattice factors. Increase /TSTORE/ and NAMAX in THERMR.

ERROR IN SIGCOH***ILLEGAL LAT

Only three lattices are coded so far. To add others, insert the constants in SIGCOH and form factor formulas in FORM.

ERROR IN IEL***UNKNOWN MATERIAL IDENTIFIER

Only three options are coded so far. To add others, insert data statements for the Debye-Waller integrals and values for the bound cross sections.

ERROR IN CALCEM***NL TOO LARGE FOR BINNING

Increase NLMAX (now 17) and the dimensions of Y and YT.

ERROR IN CALCEM***STORAGE EXCEEDED

Increase NWSCR in THERMR. This may cause a STORAG error that requires /TSTORE/ and NAMAX to be increased as well.

ERROR IN CALCEM***DESIRED TEMPERATURE NOT FOUND

Any temperature requested in THERMR must be on the input PENDF tape.

ERROR IN CALCEM***ONLY 2 SCT ATOMS ALLOWED

For mixed moderators, such as BeO and Benzine, the SCT contribution from each atom must be included. The code only allows for 2.

ERROR IS SIG***ILLEGAL OPTION

Only tabulated $S(\alpha,\beta)$ and free gas are coded at this time.

ERROR IN SIGL***NO LEGAL SOLUTION

MESSAGE FROM SIGL***DISC=-ffff, SET TO ABS VALUE....

ERROR IN SIGL***NO LEGAL SOLUTION (QUADRATIC PATH)

The code has trouble solving the equation for the boundary of a bin.

ERROR IN TPEND***DID NOT FIND TEMP ON NIN

Temperatures requested for THERMR are not consistent with those on the input PENDF tape.

ERROR IN TPEND***STORAGE EXCEEDED

Increase NWSCR in THERMR.

ERROR IN TPEND***CROSS SECTION = 0

Thermal cross section of zero cannot be used to normalize the distribution.

H. Input/Output Units

The following logical units are used.

10/11	IOLD/INEW in THERMR. Also used in COH, READEM, CALCEM, and TPEND. Used for the LOADA/FINDA scratch file that saves the energy grid and reaction cross sections.
12	NSCR in THERMR. Also used in CALCEM and TPEND. Contains the scattering matrix before normalization.
13	NSCR2 in THERMR and TPEND. Contains data from NIN that are to be simply copied to NOUT.
20-99	User's choice for NENDF, NIN, NOUT, and NREAD (IINC=2 only) to link with other modules. No mode conversion between NIN and NOUT allowed.

Units 10 and 11 are always binary. Units 12 and 13 have the same mode as NIN and NOUT. The user can choose the modes for NENDF, NIN, NOUT, except NIN and NOUT must have the same mode.

I. Storage Allocation

The storage allocated in THERMR is for the LOADA/FINDA buffers and a scratch array. The value of NBUF my be changed at will; larger values increase I/O efficiency. The variable NWSCR controls the maximum size of the TAB1 records of $\sigma(E \rightarrow E')$ versus E' for incoherent scattering. Hence it interacts with TOL. The linearization stack (STK) in COH is controlled by IMAX and the number of Legendre components requested (always 1 in the standard version). The current value of IMAX (20) is sufficient to divide each panel into parts as small as one-millionth of the panel size. The length of the list of lattice factors (FL) in SIGCOH is controlled by the size of the ENDF/B File 7, and /TSTORE/ must be big enough for the problem.

J. References

1. R. Kinsey, Ed., "ENDF-102, Data Formats and Procedures for the Evaluated Nuclear Data File, ENDF," Brookhaven National Laboratory report BNL-NCS-50496 (ENDF-102) 2nd Edition (ENDF/B-V) (October 1979).

2. J. U. Koppel and D. H. Houston, "Reference Manual for ENDF Thermal Neutron Scattering Data," General Atomic report GA-8774 revised and reissued as ENDF-269 by the National Nuclear Data Center, Brookhaven National Laboratory (1978).

3. P. F. Rose and C. L. Dunford, "ENDF-102, Data Formats and Procedures for the Evaluated Nuclear Data File, ENDF," Brookhaven National Laboratory report BNL-NCS-44945 (July 1990).

4. Y. D. Naliboff and J. U. Koppel, "HEXSCAT: Coherent Scattering of Neutrons by Hexagonal Lattices," General Atomic report GA-6026 (1964).

5. H. C. Honeck and D. R. Finch, "FLANGEII (Version 71-1), A Code to Process Thermal Neutron Data for an ENDF/B Tape," Savannah River Laboratory report DP-1278 (ENDF-152) (1971).

VIII. GROUPR

GROUPR produces self-shielded multigroup cross sections, anisotropic group-to-group scattering matrices, and anisotropic photon production matrices for neutrons from ENDF/B-IV, V, or VI evaluated nuclear data. With ENDF-6 format files, incoming and outgoing charged particles can also be handled. Special features are provided for ratio quantities (for example, $\overline{\mu}$, $\overline{\nu}$, or photon yield), inverse velocity, delayed neutron spectra by time group, and anisotropic thermal neutron scattering. Fission is represented as a group-to-group matrix for full generality. Scattering matrices and photon production matrices may be self-shielded if desired.

The Bondarenko narrow-resonance weighting scheme[1] is usually used. Optionally, a weighting flux can be computed for various mixtures of heavy absorbers with light moderators. An accurate pointwise solution of the integral slowing down equation is used. This option is normally called on to account for intermediate resonance effects in the epithermal range.

Neutron data and photon-production data are processed in a parallel manner using the same weight function and quadrature scheme. This helps to assure consistent cross sections for coupled neutron-photon problems. Two-body scattering is computed with a center-of-mass (CM) Gaussian quadrature, which gives accurate results even for small Legendre components of the group-to-group matrix.

User conveniences include free-form input and complete control over which reactions are processed. The neutron group structure, photon group structure, and weight function can each be read in or set to one of the internal options. Output can be printed and/or written to an output "groupwise-ENDF" (GENDF) file for further processing by a formatting module (DTFR, CCCCR, MATXSR), by the covariance module (ERRORR), or by the MCNP continuous-energy Monte Carlo module (ACER).

This chapter describes GROUPR in version 91.91.

A. Multigroup Constants

Multigroup constants are normally used by computer codes that calculate the distributions of neutrons and/or photons in space and energy, and that compute various responses to these distributions, such as criticality, dose to personnel, or activation of materials. These distributions are solutions of the neutral particle

transport equation.*

$$\begin{aligned} \mu\frac{\partial}{\partial x}\phi(x,\mu,E) \;+\; & \sigma_t(x,E)\,\phi(x,\mu,E) \\ = \; & \int d\Omega' \int dE'\,\sigma_X(x,E'\!\to\!E,\Omega'\!\to\!\Omega)\,\phi(x,\mu',E') \\ + \; & Q(x,\mu,E)\,, \end{aligned} \tag{1}$$

where the flux ϕ is allowed to vary with position x, direction Ω with polar cosine μ, and energy E. Similarly, the macroscopic total cross section σ_t varies with position and energy. The right-hand side of the equation contains the source due to transfers from other directions Ω' and energies E' (as described by the macroscopic transfer cross section σ_X), and a fixed or external source Q.

The macroscopic cross sections (in units of cm^{-1}) in Eq. (1) can be calculated from microscopic cross sections for the component isotopes or elements (in barns) using

$$\sigma_t(x,E) = \sum_i \rho_i(x)\,\sigma_t^i(T[x],E)\,, \tag{2}$$

where ρ_i is the number density for a constituent (in $\text{barns}^{-1}\text{cm}^{-1}$), which may vary with position, and T is the temperature, which may also vary with position. A similar formula holds for σ_X.

The transfer cross section σ_X (which includes both scattering and fission processes) is normally assumed to depend only on the cosine of the scattering angle, $\mu_0 = \Omega\cdot\Omega'$. This allows σ_X to be expanded using Legendre polynomials

$$\sigma_X(x,E'\!\to\!E,\Omega'\!\to\!\Omega) = \sum_{\ell=0}^{\infty}\frac{2\ell+1}{4\pi}\,\sigma_{X\ell}(x,E'\!\to\!E)\,P_\ell(\mu_0)\,. \tag{3}$$

Application of the addition theorem and integration over azimuthal angle then gives

$$\begin{aligned} \mu\frac{\partial}{\partial x}\phi(x,\mu,E) \;+\; & \sigma_t(x,E)\,\phi(x,\mu,E) \\ = \; & \sum_{\ell=0}^{\infty}\frac{2\ell+1}{2}P_\ell(\mu)\int \sigma_{X\ell}(x,E'\!\to\!E)\,\phi_\ell(x,E')\,dE' \\ + \; & Q(x,\mu,E)\,, \end{aligned} \tag{4}$$

where

$$\phi_\ell(x,E) = \int P_\ell(\mu)\,\phi(x,\mu,E)\,d\mu\,. \tag{5}$$

*The following development uses a notation based on Bell and Glasstone[2], where the lower-case sigma is used for both macroscopic and microscopic cross sections, depending on the context. One-dimensional slab geometry is used throughout for simplicity.

The desired responses are then given by

$$R(x) = \int \sigma_r(x,E)\,\phi_0(x,E)\,dE\,, \tag{6}$$

where σ_r is the reaction cross section for the response. The next step is to integrate Eqs. (4) and (6) over a range of energies chosen to lie in group g. The results are

$$\begin{aligned}\mu\frac{\partial}{\partial x}\phi_g(x,\mu) \;+\; &\sum_{\ell=0}^{\infty} P_\ell(\mu)\,\sigma_{t\ell g}(x)\,\phi_{\ell g}(x) \\ &= \sum_{\ell=0}^{\infty}\frac{2\ell+1}{2}P_\ell(\mu)\sum_{g'}\sigma_{X\ell g'\to g}(x)\,\phi_{\ell g'}(x) + Q_g(x,\mu)\,,\end{aligned} \tag{7}$$

and

$$R(x) = \sum_g \sigma_{rg}(x)\,\phi_{0g}(x)\,, \tag{8}$$

where

$$\phi_{\ell g}(x) = \int_g \phi_\ell(x,E)\,dE\,, \tag{9}$$

$$\sigma_{t\ell g}(x) = \frac{\int_g \sigma_t(x,E)\,\phi_\ell(x,E)\,dE}{\int_g \phi_\ell(x,E)\,dE}\,, \tag{10}$$

$$\sigma_r(x) = \frac{\int_g \sigma_r(x,E)\,\phi_0(x,E)\,dE}{\int_g \phi_0(x,E)\,dE}\,, \tag{11}$$

and

$$\sigma_{X\ell g'\to g} = \frac{\int_g dE \int_{g'} dE'\,\sigma_{X\ell}(x,E'\to E)\,\phi_\ell(x,E)}{\int_g \phi_\ell(x,E)\,dE}\,. \tag{12}$$

The last three equations provide the fundamental definitions for the multigroup cross sections and the group-to-group matrix. Note that the values of the group constants depend upon the basic energy-dependent cross sections obtained from ENDF/B by way of the RECONR, BROADR, UNRESR, HEATR, and THERMR modules of NJOY, and the shape of ϕ within the group.

B. Group Ordering

Since neutrons normally lose energy in scattering, the scattering source into group g depends on the flux at higher-energy groups g' and the cross section for transferring neutrons from g' to g. For this reason, Eq. (7) is usually solved by

sweeping from high energies to low energies. (Any thermal upscatter or fission is handled by iteration.) Data libraries for use with transport codes normally number the groups such that group 1 is the highest-energy group, and all the scattering matrix elements that transfer neutrons into group 1 are given first, followed by those for scattering into group 2, and so on.

However, in ENDF/B the evaluated nuclear data are always given in order of increasing incident energy, and secondary neutron distributions are described by giving emission spectra for given incident energies. Therefore, GROUPR numbers its groups such that group 1 is the lowest-energy group, and it calculates that scattering out of group 1, followed by the scattering out of group 2, and so on.

The "backward" energy-group numbering convention used by GROUPR is a possible source of confusion in interpreting output produced by the various modules of NJOY. All group indices printed by GROUPR or written to the GROUPR output file use the increasing-energy order. The covariance modules and photon interaction module follow the GROUPR ordering convention. Output modules such as DTFR, CCCCR, and MATXSR invert the group order and rearrange the scattering matrices from the GROUPR outscatter organization to the "transport" inscatter form. Any group indices printed by these three output modules will be in the conventional transport decreasing-energy order.

C. Basic ENDF/B Cross Sections

The basic energy-dependent cross sections and energy-angle distributions needed for Eqs. (10), (11), and (12) are obtained from evaluated nuclear data in ENDF/B format[3, 4]. These data are indexed by material (MAT), type of information (MF), and reaction (MT). Materials can be single isotopes, elements, or compounds. Type of information includes energy-dependent cross section (MF=3), angular distributions (MF=4), secondary energy distributions (MF=5), and energy-angle distributions (MF=6). Reactions include the total (MT=1) required for Eq. (10), the partial scattering reactions that must be summed for Eq. (12) [that is, elastic (MT=2), discrete-level inelastic (MT=51–90), (n,2n) (MT=16), etc.], and the many partial reactions that can be used to calculate responses [for example, (n,2n) or (n,α) for activation, gas production, heat production (KERMA), radiation damage (DPA), etc.].

Before using GROUPR, the basic ENDF/B cross sections should have been converted into energy- and temperature-dependent pointwise cross sections in PENDF (pointwise ENDF) format using RECONR and BROADR. For heavy isotopes, un-

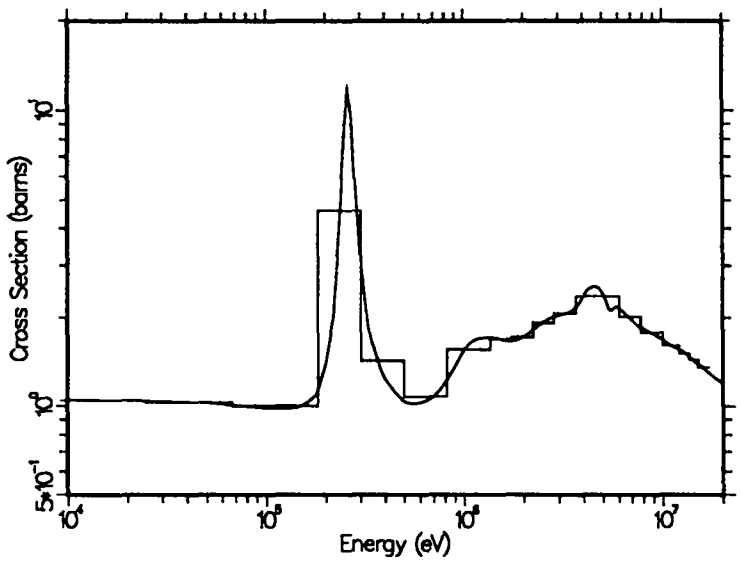


Figure 1: A comparison of the pointwise and multigroup representations for the total cross section of Li-7. The Los Alamos 30-group structure is shown.

resolved self-shielding data should have been added to the PENDF tape* using UNRESR. If needed, heat production cross sections (KERMA), radiation damage production (or DPA), and thermal upscatter data could have also been added to the PENDF tape using HEATR and THERMR. See the chapters on these other modules for more information.

The detailed methods used for evaluating Eqs. (10), (11), and (12) from ENDF and PENDF tapes are given below (see Sections VIII.K, VIII.L, and VIII.N). An example of an ENDF/B pointwise cross section compared with group-averaged cross sections from GROUPR is given in Fig. 1.

D. Weighting Flux

In general, the weighting flux ϕ is not known; it is, after all, the particle distribution being sought in the transport calculation. However, it is often possible to obtain fairly accurate group constants for a particular application if the shape of the flux is reasonably well known over the broad energy ranges of a particular few-group structure (for example, a fission spectrum, thermal Maxwellian, or $1/E$ slowing-down spectrum). Alternatively, one can use many small groups so that mistakes in guessing the shape inside the group are not very important. The key

*The term "tape" is used loosely in this report to refer to any input or output file. Of course, such files would usually be on disk storage in a modern system.

to using the multigroup method effectively is balancing the tradeoffs between the choice of weight function and the number of groups used for each different class of problem being solved.

In many cases of practical interest, the flux ϕ will contain dips corresponding to the absorption resonances of the various materials. In the reaction rate $\sigma(E) \times \phi(E)$, these dips clearly reduce (self-shield) the effect of the corresponding resonance. GROUPR provides two methods to estimate the effect of this self-shielding: the Bondarenko model and the flux calculator.

In the Bondarenko model[1], the narrow resonance (NR) approximation, and the B_N approximation for large systems[2] are invoked to obtain

$$\phi_\ell(E) = \frac{W_\ell(E)}{[\sigma_t(E)]^{\ell+1}} , \tag{13}$$

where ϕ_ℓ is the ℓ-th Legendre component of the angular flux, the $W_\ell(E)$ are smooth functions of energy (such as $1/E$+fission), and $\sigma_t(E)$ is the total macroscopic cross section for the material. GROUPR takes all of the W_ℓ to be equal to the single function $C(E)$, where $C(E)$ can be read in or set to one of several internally defined functions. It is further assumed that the important self-shielding effect of the flux can be obtained for isotope i by representing all the other isotopes with a constant "background cross section", σ_0. Therefore,

$$\phi^i_\ell(E) = \frac{C(E)}{[\sigma^i_t(E) + \sigma^i_0]^{\ell+1}} , \tag{14}$$

where σ^i_t is the microscopic total cross section for isotope i. The qualitative behavior of Eq. (14) is easy to understand. If σ_0 is larger than the tallest peaks in σ_t, the weighting flux ϕ is approximately proportional to the smooth weighting function $C(E)$. This is called infinite dilution; the cross section in the material of interest has little or no effect on the flux. On the other hand, if σ_0 is small with respect to σ_t, the weighting flux will have large dips at the locations of the peaks in σ_t, and a large self-shielding effect will be expected.

Each component material of a mixture has a different weight function. The macroscopic total cross sections are given by

$$\sigma_{t\ell g} = \sum_i \rho_i \, \sigma^i_{t\ell g}(\sigma^i_0, T) , \tag{15}$$

where

$$\sigma^i_{t\ell g}(\sigma_0,T)=\frac{\int_g \frac{\sigma^i_t(E,T)}{[\sigma_0+\sigma^i_t]^{\ell+1}}C(E)\,dE}{\int_g \frac{1}{[\sigma_0+\sigma^i_t]^{\ell+1}}C(E)\,dE}\,. \tag{16}$$

Similar equations are used for σ_R and $\sigma_{X\ell g\to g'}$. On the GROUPR level, σ_0 and T are simply parameters. Subsequent codes, such as 1DX[5] or TRANSX[6], can compute an appropriate value for σ_0, and then interpolate in tables of cross section versus σ_0 and T to get the desired self-shielded group constants.

The appropriate value for σ^i_0 is obvious when a single resonance material is mixed with a moderator material (for example, $^{238}UO_2$), because the admixed materials typically have a constant cross section in the energy range where the heavy isotopes have resonances. For a mixture of resonance materials, the normal procedure is to preserve the average of Eq. (14) in each group by using

$$\sigma^i_{0g}=\frac{1}{\rho_i}\sum_{j\neq i}\rho_j\,\sigma^j_{i0g}(\sigma^j_{0g},T)\,, \tag{17}$$

where the ρ_i are atomic densities or atomic fractions. Eq. (17) is solved by iteration. Interference between resonances in different materials is handled in an average sense only.

In the unresolved energy range (see UNRESR), the explicit dependence of cross section on energy is not known. The integrands are replaced by their expected values

$$\sigma_{x\ell g}(\sigma_0)=\frac{\int_g\left\langle\frac{\sigma_x}{[\sigma_0+\sigma_t]^{\ell+1}}\right\rangle W_\ell\,dE}{\int_g\left\langle\frac{1}{[\sigma_0+\sigma_t]^{\ell+1}}\right\rangle W_\ell\,dE}\,, \tag{18}$$

where the expected values are averages over the distributions of resonance position and width expected in the vicinity of energy E. The UNRESR module produces effective self-shielded point cross sections defined by

$$\langle\sigma_x\rangle_\ell=\frac{\left\langle\frac{\sigma_x}{[\sigma_0+\sigma_t]^{\ell+1}}\right\rangle}{\left\langle\frac{1}{[\sigma_0+\sigma_t]^{\ell+1}}\right\rangle}\,. \tag{19}$$

Substituting Eq. (19) into Eq. (18) gives an equation of the form of Eqs. (10) and (11), except that σ is replaced by $\langle\sigma\rangle$, and the flux is replaced by an average effective flux. This effective flux can be obtained by manipulating the effective

total cross section as follows:

$$\langle\sigma_t\rangle_\ell = \frac{\left\langle \frac{\sigma_0+\sigma_t-\sigma_0}{[\sigma_0+\sigma_t]^{\ell+1}} \right\rangle}{\left\langle \frac{1}{[\sigma_0+\sigma_t]^{\ell+1}} \right\rangle} = \frac{\left\langle \frac{1}{[\sigma_0+\sigma_t]^{\ell}} \right\rangle}{\left\langle \frac{1}{[\sigma_0+\sigma_t]^{\ell+1}} \right\rangle} - \sigma_0 \ , \tag{20}$$

from which

$$\left\langle \frac{1}{[\sigma_0+\sigma_t]^{\ell+1}} \right\rangle = \frac{\left\langle \frac{1}{[\sigma_0+\sigma_t]^{\ell}} \right\rangle}{\sigma_0+\langle\sigma_t\rangle_k} \ . \tag{21}$$

Eq. (21) defines a recursion relation which can be used to compute the effective flux to any order

$$\left\langle \frac{1}{[\sigma_0+\sigma_t]^{\ell+1}} \right\rangle = \prod_{k=0}^{\ell} \frac{1}{\sigma_0+\langle\sigma_t\rangle_k} \ , \tag{22}$$

This equation reduces to Eq. (14) in the resolved range. It is the formula used in **GENFLX** to compute $\phi_\ell(E)$ for the Bondarenko option.

When heterogeneity effects are important, the background cross section method can be extended as follows. In an infinite system of two regions (fuel and moderator), the neutron balance equations are

$$V_f\sigma_f\phi_f = (1-P_f)V_fS_f + P_mV_mS_m \ , \tag{23}$$

and

$$V_m\sigma_m\phi_m = P_fV_fS_f + (1-P_m)V_mS_m \ , \tag{24}$$

where V_f and V_m are the region volumes, σ_f and σ_m are the corresponding total macroscopic cross sections, S_f and S_m are the sources per unit volume in each region, P_f is the probability that a neutron born in the fuel will suffer its next collision in the moderator, and P_m is the probability that a neutron born in the moderator will suffer its next collision in the fuel. As usual, use is made of the reciprocity theorem,

$$V_f\sigma_fP_f = V_m\sigma_mP_m \ , \tag{25}$$

and the Wigner rational approximation to the fuel escape probability,

$$P_f = \frac{\sigma_e}{\sigma_e+\sigma_f} \ , \tag{26}$$

where σ_e is a slowly varying function of energy called the escape cross section, to obtain an equation for the fuel flux in the form

$$(\sigma_f+\sigma_e)\phi_f = \frac{\sigma_eS_m}{\sigma_m} + S_f \ . \tag{27}$$

In the limit where the resonances are narrow with respect to both fuel and moderator scattering, the source terms S_f and S_m take on their asymptotic forms of σ_p/E and σ_m/E respectively, and this equation becomes equivalent to the Bondarenko model quoted above with

$$\sigma_0^f = \frac{\sigma_e}{\rho_f} , \tag{28}$$

and

$$C(E) = \frac{\sigma_e + \sigma_p}{\rho_f E} . \tag{29}$$

Note that a large escape cross section (a sample that is small relative to the average distance to collision), corresponds to infinite dilution as discussed above. To illustrate the general case, consider a neutron traveling through a lump of uranium oxide with an energy close to a resonance energy. If the neutron scatters from an oxygen nucleus, it will lose enough energy so that it can not longer react with the uranium resonance. Similarly, if the neutron escapes from the lump, it can no longer react with the uranium resonance. The processes of moderator scattering and escape are equivalent in some way. Comparing Eq. (28) with Eq. (17) gives an "equivalence principle" that says that a lump of particular dimensions and a mixture of particular composition will have the same self-shielded cross sections when the narrow resonance approximation is valid. The effects of material mixing and escape can simply be added to obtain the effective σ_0 for a lump containing admixed moderator material. Therefore, Eq. (17) is extended to read

$$\sigma_{0g}^i = \frac{1}{\rho_i}\left\{\sigma_e + \sum_{j\neq i} \rho_j \, \sigma_{i0g}^j(\sigma_{0g}^j, T)\right\} , \tag{30}$$

where the escape cross section for simple convex objects (such as plates, spheres, or cylinders) is given by $(4V/S)^{-1}$, where V and S are the volume and surface area of the object, respectively. Many codes that use the background cross section method modify the escape cross section as defined above to correct for errors in the Wigner rational approximation ("Bell factor", "Levine factor"), or to correct for the interaction between different lumps in the moderating region ("Dancoff corrections"). These enhancements will not be discussed here.

E. Flux Calculator

This narrow-resonance approach is quite useful for practical fast reactor problems. However, for nuclear systems sensitive to energies from 1 to 500 eV, there are many broad- and intermediate-width resonances which cannot be self-shielded with sufficient accuracy using the Bondarenko approach. The GROUPR flux calculator is designed for just such problems.

Consider an infinite homogeneous mixture of two materials and assume isotropic scattering in the center-of-mass system. The integral slowing-down equation becomes

$$\begin{aligned}\sigma(E)\,\phi(E) &= \int_E^{E/\alpha_1} \frac{\sigma_{s1}(E')}{(1-\alpha_1)E'}\,\phi(E')\,dE' \\ &+ \int_E^{E/\alpha_2} \frac{\sigma_{s2}(E')}{(1-\alpha_2)E'}\,\phi(E')\,dE' \,. \end{aligned} \tag{31}$$

Furthermore, assume that material 1 is a pure scatterer with constant cross section and transform to the σ_0 representation. The integral equation becomes

$$\begin{aligned}[\sigma_0 + \sigma_{t2}(E)]\,\phi(E) &= \int_E^{E/\alpha_1} \frac{\sigma_0}{(1-\alpha_1)E'}\,\phi(E')\,dE' \\ &+ \int_E^{E/\alpha_2} \frac{\sigma_{s2}(E')}{(1-\alpha_2)E'}\,\phi(E')\,dE' \,. \end{aligned} \tag{32}$$

Finally, assume that the moderator (material 1) is light enough so that all the resonances of material 2 are narrow with respect to scattering from material 1. This allows the first integral to be approximated by its asymptotic form $1/E$. More generally, the integral is assumed to be a smooth function of E given by $C(E)$. In this way, material 1 can represent a mixture of other materials just as in the Bondarenko method. Fission source and thermal upscatter effects can also be lumped in $C(E)$. The integral equation has now been reduced to

$$[\sigma_0 + \sigma_t(E)]\,\phi(E) = C(E)\,\sigma_0 + \int_E^{E/\alpha} \frac{\sigma_s(E')}{(1-\alpha)E'}\,\phi(E')\,dE' \,. \tag{33}$$

This is the simplest problem that can be solved using the flux calculator. The results still depend on the single parameter σ_0, and they can be used easily by codes that accept Bondarenko cross sections.

For heterogeneous problems, when the narrow-resonance approximation fails, both S_f and S_m in Eq. (27) will show resonance features. To proceed further with the solution of this equation, it is necessary to eliminate the moderator flux that is implicit in S_m. As a sample case, consider a fuel pin immersed in a large region of water. The fission neutrons appear at high energies, escape from the pin, slow down in the moderator (giving a $1/E$ flux), and are absorbed by the resonances in the pin. In this limit, any dips in the moderator flux caused by resonances in the fuel are small. On the other hand, in a closely packed lattice, the flux in the moderator is very similar to the flux in the fuel, and resonance dips in the moderator flux become very evident. Intermediate cases can be approximated[7] by

assuming

$$\phi_m = (1-\beta)\,C(E) + \beta\phi_f \ , \tag{34}$$

where β is a heterogeneity parameter given by

$$\beta = \frac{V_f \sigma_e}{V_m \sigma_m} \ . \tag{35}$$

Note that $\beta \to 0$ gives the isolated rod limit and $\beta \to 1$ gives the close-packed lattice limit. This substitution reduces the calculation of the fuel flux to

$$(\sigma_f + \sigma_e)\,\phi_f = (1-\beta)\,C(E)\,\sigma_e + S_\beta \ , \tag{36}$$

where S_β is the source term corresponding to a homogeneous mixture of the fuel isotopes with the isotopes from the moderator region changed by the factor $\beta\sigma_e/\sigma_m$. If the fuel and moderator each consisted of a single isotope and for isotropic scattering in the center-of-mass system, the integral equation would become

$$\begin{aligned} [\sigma_0 + \sigma_t(E)]\,\phi_f(E) &= (1-\beta)\,C(E)\,\sigma_0 \\ &+ \int_E^{E/\alpha_m} \frac{\beta\sigma_0}{(1-\alpha_m)E'}\,\phi_f(E')\,dE' \\ &+ \int_E^{E/\alpha_f} \frac{\sigma_{sf}(E')}{(1-\alpha_f)E'}\,\phi_f(E')\,dE' \ , \end{aligned} \tag{37}$$

where σ_0 is σ_e divided by the fuel density (units are barns/atom), α_m and α_f are the maximum fractional energy change in scattering for the two isotopes, and $\sigma_{sf}(E')$ is the fuel scattering cross section.

This result has a form parallel to that of Eq. (33), but the solution depends on the two parameters β and σ_0. For any given data set, β must be chosen in advance. This might not be difficult if the data are to be used for one particular system, such as pressurized water reactors. The routine also has the capability to include one more moderator integral with a different α value and a constant cross section. The full equation is

$$\begin{aligned} [\sigma_0 + \sigma_t(E)]\,\phi_f(E) &= (1-\beta)\,C(E)\,\sigma_0 \\ &+ \int_E^{E/\alpha_3} \frac{\beta(1-\gamma)(\sigma_0 - \sigma_{am}}{(1-\alpha_3)E'}\,\phi_f(E')\,dE' \\ &+ \int_E^{E/\alpha_2} \frac{\sigma_{am} + \beta\gamma(\sigma_0 - \sigma_{am}}{(1-\alpha_2)E'}\,\phi_f(E')\,dE' \\ &+ \int_E^{E/\alpha_f} \frac{\sigma_{sf}(E')}{(1-\alpha_f)E'}\,\phi_f(E')\,dE' \ , \end{aligned} \tag{38}$$

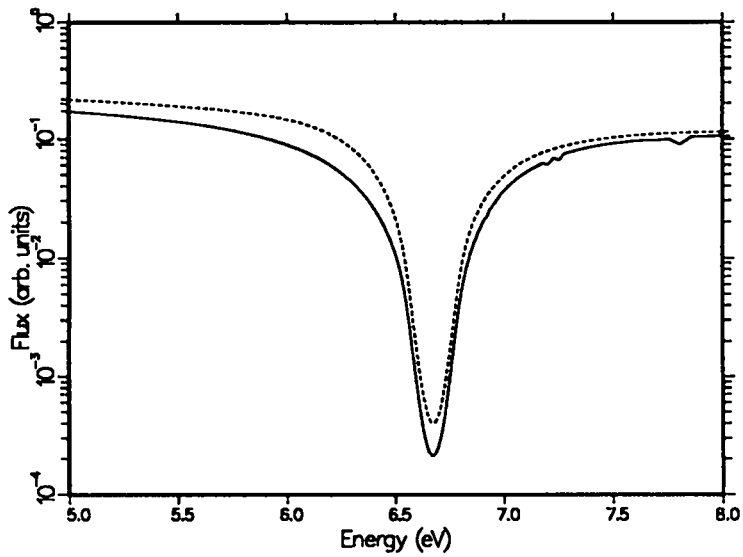


Figure 2: A comparison of the Bondarenko flux model (dashed) with a realistic computed flux (solid) for a ^{238}U oxide pin in water in the region of the 6.7 eV resonance.

where σ_{am} is the cross section of the admixed moderator (with energy loss α_2), and γ is the fraction of the admixed moderator that is mixed with the external moderator (which has energy loss α_3). This allows calculations with H_2O as the moderator and an oxide as the fuel. The flux calculator can thus obtain quite realistic flux shapes for a variety of fuel, admixed moderator, and external moderator combinations. An example comparing the Bondarenko flux with a more realistic computed flux is given in Fig. 2.

In practice, a fuel rod rarely contains only one resonance isotope. As an example, consider a mixture of a few percent of ^{239}Pu with ^{238}U as the major component. There will be a strong dip in the flux associated with the 6.7 eV ^{238}U resonance that will affect the flux in the region of the 7.8 eV ^{239}Pu resonance (the interference effect), and there will also be a dip in the flux corresponding to the 7.8 eV resonance (the self-shielding effect). This additional complication in the flux shape would be expected to change the group constants for ^{239}Pu since both features lie in the same group for typical group structures. However, the effect of the ^{239}Pu on the ^{238}U group constants should be minimal. This argument suggests that the full flux calculation be used for ^{238}U as a single resonance material. The resulting flux would then be used to estimate the flux to be used in averaging the ^{239}Pu

cross sections as follows:

$$\phi_{239}(E,\sigma_0) = \frac{\phi_{238}(E,50)}{\sigma_0 + \dfrac{\sigma_{239}(E)}{1 + \dfrac{\sigma_{238}(E)}{50}}} , \tag{39}$$

where the ^{238}U flux is characteristic of a background of 50 barns/atom, which is representative of many thermal reactor systems. This formula assumes that the effect of ^{239}Pu on the scattering source for the mixture is small, but it retains the absorption effects. The self-shielding of ^{239}Pu is treated in the narrow resonance approximation only. The GROUPR flux calculator includes an option to write out a file containing the calculated flux and cross section needed for this formula (e.g., for ^{238}U) and another option to skip the flux calculation and use the formula above to obtain the weighting flux (e.g., for ^{239}Pu).

The slowing-down integral equation of Eq. (33) or (37) is solved point by point (see subroutine **GENFLX**) using the total and elastic cross sections on the PENDF tape produced by RECONR. In order to keep this task within bounds, the flux is computed from the lower limit of the first group to a specified energy **EHI** or until **NFMAX** values have been computed. The flux at higher energies is continued using the Bondarenko model described above.

F. Fission Source

The fission source was included in the transfer cross section σ_X in the development above. It is usually convenient to separate fission from scattering. Assuming isotropy, the source of fission neutrons into group g is given by

$$S_g = \sum_{g'} \sigma_{fg'\to g}\, \phi_{0g'} , \tag{40}$$

where the group-to-group matrix for fission is defined as in Eq. (12), but with ℓ equal to zero. Most existing transport codes do not use this matrix form directly because the upscatter is expensive to handle and a reasonably accurate alternative exists. Except for relatively high neutron energies, the spectrum of fission neutrons is only weakly dependent on initial energy. Therefore, the fission source can be written

$$S_g = \chi_g \sum_{g'} \overline{\nu}_{g'}\, \sigma_{fg'}\, \phi_{0g'} , \tag{41}$$

where $\overline{\nu}_g$ is the fission neutron yield, σ_{fg} is the fission cross section, and χ_g is the average fission spectrum (the familiar "chi" vector), which can be defined by

$$\chi_g = \frac{\sum_{g'} \sigma_{fg'\to g}\, \phi_{0g'}}{\sum_{g}\sum_{g'} \sigma_{fg'\to g}\, \phi_{0g'}} . \tag{42}$$

The fission neutron yield is given by

$$\overline{\nu}_g = \sum_{g'} \sigma_{fg\to g'} \,/\, \sigma_{fg} . \tag{43}$$

Clearly, χ_g as given by Eq. (42) depends on the flux in the system of interest. The dependence is weak except for high incident energies, and a rough guess for ϕ_{0g} usually gives an accurate spectrum. When this is not the case, the problem can be iterated, or the full matrix representation can be used.

It is possible to take advantage of the weak energy dependence of the shape of the fission spectrum at low energies to reduce the time required to process fission and to reduce the size of the fission data on the output file. GROUPR determines a break energy from File 5 such that the fission spectrum is constant below this energy. It only has to make a single calculation of this spectrum χ_g^{LE}. Then it computes a fission neutron production cross section σ_{fPg}^{HE} for groups below the break energy using the normal cross section processing methods. A full matrix $\sigma_{fg'\to g}^{HE}$ is computed for groups above the break point. As an example, consider using the GROUPR 187-group structure and finding that there are 130 constant groups. The 187×187 matrix is reduced to a 57×187 matrix, a 187-element spectrum vector, and a 130-group production vector, for a total reduction in size of 68%. The effective fission matrix is given by

$$\sigma_{fg'\to g} = \sigma_{fg'\to g}^{HE} + \chi_g^{LE} \sigma_{fPg'}^{LE} . \tag{44}$$

The fission matrix computed by GROUPR represents the prompt part of fission only. The delayed component of fission is represented by a delayed-neutron yield $\overline{\nu}_g^D$, decay constants for six time groups, λ_i^D, and emission probability spectra for six time groups, χ_{ig}^D. Steady-state fission can be obtained using

$$\overline{\nu}_g^{SS} = \overline{\nu}_g + \overline{\nu}_g^D , \tag{45}$$

and

$$\chi_g^{SS} = \frac{\sum_{g'} \sigma_{fg' \to g} \phi_{0g'} + \chi_g^D \sum_{g'} \overline{\nu}_{g'}^D \sigma_{fg'} \phi_{0g'}}{\sum_g \sum_{g'} \sigma_{fg' \to g} \phi_{0g'} + \sum_{g'} \overline{\nu}_{g'}^D \sigma_{fg'} \phi_{0g'}} , \tag{46}$$

where

$$\chi_g^D = \sum_i \chi_{ig}^D . \tag{47}$$

Note that χ_g^D sums to unity, but the χ for each time group sums to the fraction of the delayed neutron yield that appears in that time group.

In the ENDF/B files, the total fission reaction is represented by MT=18. Important isotopes also give the partial fission reactions (n,f), (n,n'f), (n,2nf), and sometimes (n,3nf) using MT=19, 20, 21, and 38 respectively. The MT=18 representation is adequate for most fission reactor applications, but the partial reactions should be processed for applications with significant flux above 6 MeV. Caution: although the cross section for MT=18 equals the sum of its parts, the group-to-group fission matrix $\sigma_{fg \to g'}$ computed from MT=18 will not, in general, equal the sum of the partial matrices for MT=19, 20, 21, and 38 above the 6-MeV threshold for second-chance fission. The delayed neutron data are given in MT=455. Sample input instructions for processing the various combinations of fission reactions used in ENDF/B will be found in Section VIII.N. GROUPR outputs all the components of fission separately in order to give succeeding modules or codes complete flexibility.

G. Diffusion Cross Sections

The diffusion equation is very commonly used in reactor physics calculations. Starting with Eq. (7), use the Legendre expansion for ϕ_g in the derivative term, and make use of the recursion relation for $\mu P_\ell(\mu)$ and the orthogonality relation for the Legendre polynomials to obtain the transport equation in spherical harmonic form

$$\frac{n+1}{2n+1} \frac{\partial \phi_{n+1,g}}{\partial x} + \frac{n}{2n+1} \frac{\partial \phi_{n-1,g}}{\partial x} \sigma_{tng} \phi_{ng}$$

$$= \sum_{g'} \sigma_{sng' \to g} \phi_{ng'} + S_{fg} + Q_{ng} , \tag{48}$$

where the transfer term has been separated into a scattering term with cross section σ_s, and a fission source term S_f. When this set of equations is truncated at $n = N$,

the result is usually called the "P_N equations". Now, all terms with $n > 1$ are dropped, and Q is assumed to be isotropic. Thus,

$$\frac{\partial \phi_{1g}}{\partial x} + \sigma_{t0g}\,\phi_{0g} = \sum_{g'} \sigma_{s0g'\to g}\,\phi_{0g'} + S_f + Q_{0g} \ , \tag{49}$$

and

$$\frac{1}{3}\frac{\partial \phi_{0g}}{\partial x} + \sigma_{t1g}\,\phi_{1g} = \sum_{g'} \sigma_{s1g'\to g}\,\phi_{1g'} \ . \tag{50}$$

The second equation can be written in the form of Fick's Law as follows:

$$\phi_{1g} = -D_g \frac{\partial \phi_{0g}}{\partial x} \ , \tag{51}$$

$$D_g = \frac{1}{3}\frac{1}{\sigma_{t1g} - \sum_{g'} \sigma_{s1g'\to g}\,\phi_{1g'} / \phi_{1g}} \ , \tag{52}$$

where D_g is the diffusion constant. The term in the denominator of the second factor is the transport cross section for diffusion, σ_{trD}. Unfortunately, it depends on a fairly complete knowledge of the neutron current in the system, perhaps from a previous calculation. However, for many problems, σ_{trD} can be simplified by assuming that

$$\sum_{g'} \sigma_{s1g'\to g}\,\phi_{1g'} \approx \sum_{g'} \sigma_{s1g\to g'}\,\phi_{1g} \ , \tag{53}$$

with the result that

$$\sigma_{trD,g} = \sigma_{t1g} - \sum_{g'} \sigma_{s1g\to g'} \ , \tag{54}$$

or

$$\sigma_{trD,g} = \sigma_{t1g} - \overline{\mu}_g\,\sigma_{s0g} \ , \tag{55}$$

where $\overline{\mu}_g$ is the average scattering cosine for neutrons in group g. These forms depend only on the shape of the weighting flux within the group, as usual. Substituting for ϕ_{1g} in Eq. (49) gives

$$\frac{\partial}{\partial x}\left(-D_g \frac{\partial \phi_{0g}}{\partial x}\right) + \sigma_{0tg}\,\phi_{0g} = \sum_{g'} \sigma_{s0g'\to g}\,\phi_{0g'} + S_{fg} + Q_{0g} \ , \tag{56}$$

which is the standard diffusion equation in slab geometry. Neither the diffusion coefficient nor the transport cross section for diffusion is produced directly by GROUPR. However, the components such a $\sigma_{t\ell}$ and $\sigma_{s0g'\to g}$ are made available to subsequent modules.

H. Cross Sections for Transport Theory

The S_N (discrete ordinates) transport codes solve the equation

$$\begin{aligned} \mu\frac{\partial}{\partial x}\phi_g(\mu,x) \;+\; & \sigma_g^{SN}\phi_g(\mu,x) \\ = \; & \sum_{\ell=0}^{N}\frac{2\ell+1}{2}P_\ell(\mu)\sum_{g'}\sigma_{s\ell g'\to g}^{SN}(x)\,\phi_{\ell g'} \\ + \; & S_{fg}+Q_g(\mu,x)\;, \end{aligned} \tag{57}$$

where once again one-dimensional slab geometry has been used for simplicity.* By comparing Eq. (57) with Eq. (7), it is seen that the S_N equations require the following cross sections:

$$\sigma_{s\ell g'\to g}^{SN} = \sigma_{s\ell g'\to g}, \qquad g'\neq g\;, \tag{58}$$

and

$$\sigma_{s\ell g\to g}^{SN} = \sigma_{s\ell g\to g} - \sigma_{t\ell g} + \sigma_g^{SN}\;, \tag{59}$$

where σ_g^{SN} is not determined and can be chosen to improve the convergence of the S_N calculation. A particular choice of σ_g^{SN} gives rise to a "transport approximation", and various recipes are in use.

Consistent-P approximation:

$$\sigma_g^{SN} = \sigma_{t0g}\;. \tag{60}$$

Inconsistent-P approximation:

$$\sigma_g^{SN} = \sigma_{t,N+1,g}\;. \tag{61}$$

Diagonal transport approximation:

$$\sigma_g^{SN} = \sigma_{t,N+1,g} - \sigma_{s,N+1,g\to g}\;. \tag{62}$$

Bell-Hansen-Sandmeier (BHS) or extended transport approximation:

$$\sigma_g^{SN} = \sigma_{t,N+1,g} - \sum_{g'}\sigma_{s,N+1,g\to g'}\;. \tag{63}$$

*The following development is based on the work of Bell, Hansen, and Sandmeier[8].

Inflow transport approximation:

$$\sigma_g^{SN} = \sigma_{t,N+1,g} - \frac{\sum_{g'} \sigma_{s,N+1,g'\to g}\, \phi_{N+1,g'}}{\phi_{N+1,g}} \,. \tag{64}$$

The first two approximations are most appropriate when the scattering orders above N are small. The inconsistent option removes most of the delta function of forward scattering introduced by the correction for the anisotropy of the total scattering rate, and should normally be more convergent than the consistent option. The diagonal and BHS recipes make an attempt to correct for anisotropy in the scattering matrix and are especially effective for forward-peaked scattering. The BHS form is more often used, but the diagonal option can be substituted when BHS produces negative values. The inflow recipe makes the $N+1$ term of the P_N expansion vanish, but it requires a good knowledge of the $N+1$ flux moment from some previous calculation. Inflow reduces to BHS for systems in equilibrium by detail balance (i.e., the thermal region). In the absence of self-shielding (that is, $\sigma_0 \to \infty$), the distinction between σ_{t1} and σ_{t0} disappears, and so does the distinction between the inconsistent-P and consistent-P options. Also note that the inflow and BHS versions of σ_g are equivalent to the definitions of $\sigma_{trD,g}$ given in Eqs. (52) and (54), respectively, when $N=0$.

These transport-corrected cross sections are not computed directly by GROUPR, but the components needed are written to the GROUPR output file for use by subsequent modules.

I. Photon Production and Coupled Sets

Photon transport is treated with an equation similar to Eq. (7), except that the flux, cross sections, and groups are all referred to photon interactions and photon energies instead of to the corresponding neutron quantities. Methods of producing the photon interaction cross sections are described in GAMINR, Chapter IX of this manual. The external photon source Q_g depends on the neutron flux and the photon production cross sections through

$$Q_g(x,\mu) = \sum_{\ell=0}^{\infty} \frac{2\ell+1}{2} P_\ell(\mu) \sum_{g'} \sigma_{\gamma\ell g'\to g}(x)\, \phi_{\ell g'}(x) \,, \tag{65}$$

where $\sigma_{\gamma\ell g'\to g}$ is defined by Eq. (12) with X replaced by γ. The ENDF/B files define σ_γ using a combination of photon production cross sections (MF=13), photon yields (MF=12) with respect to neutron cross sections (MF=3), discrete lines

(MF=12 and 13), and continuous γ distributions (MF=15). Methods for working with these representations will be discussed in more detail below.

The low-energy groups for fission and capture normally have photon emission spectra whose shapes do not change with energy. The same method used for reducing the size of the fission matrix (see Section VIII.F) can be used for these photon production matrices. In mathematical form,

$$\sigma_{\gamma g' \to g} = \sigma^{HE}_{\gamma g' \to g} + s^{LE}_{\gamma} g \sigma^{LE}_{\gamma P g'} , \tag{66}$$

where $s^{LE}_{\gamma} g$ is the normalized emission spectrum, and $\sigma^{LE}_{\gamma P_g}$ is the associated production cross section.

For many practical problems, it is convenient to combine the neutron and photon transport calculations into a single application of Eq. (7) where the photons are treated as additional groups of low-energy neutrons. Since (γ,n) events are not usually very important, the downscattering structure (see Section B) of the transport calculation is preserved for both n$\to\gamma$ and n$\to$n events. Cross sections for this application are called "coupled sets". Coupled sets are not produced directly by GROUPR, but the n$\to$n and n$\to\gamma$ components are made available to the other modules where they can be combined with $\gamma\to\gamma$ cross sections from GAMINR. As an example, the MATXSR module can format data for the TRANSX code[6], which can, in turn, prepare coupled sets for use by standard transport codes such as ONEDANT[9]. The normal arrangement of data in a coupled set is shown schematically in Fig. 3.

J. Thermal Data

GROUPR uses the thermal data written onto the PENDF tape by the THERMR module. It does not process ENDF File-7 data directly. Three different representations of thermal scattering are used in ENDF.

For crystalline materials like graphite, coherent elastic (with zero energy change) scattering can take place. The cross section for this process shows well-defined Bragg edges at energies that correspond to the various lattice-plane spacings in the crystalline powder. As shown in THERMR, Chapter VII of this manual, the angular dependence of the coherent elastic cross section can be written as

$$\sigma^{coh}(E,\mu) = \sum_i \frac{b_i}{E} \delta(\mu - \mu_i) , \tag{67}$$

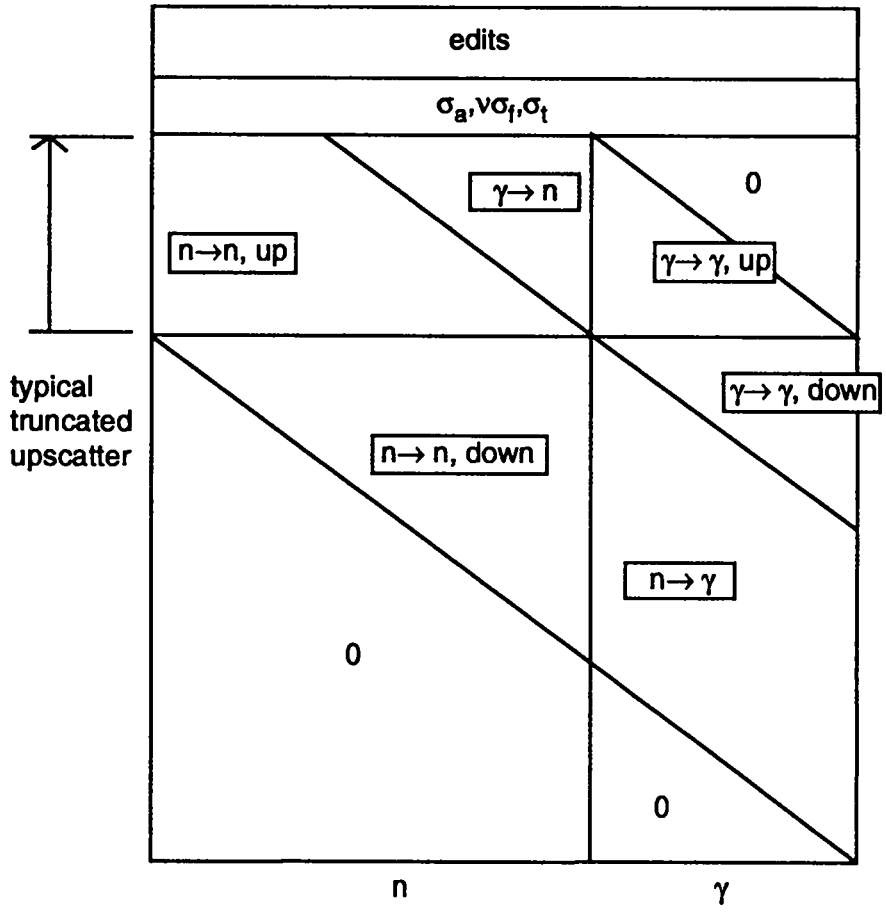


Figure 3: Arrangement of neutron (n) and photon (γ) cross sections in a coupled transport table. The group index increases from left to right, and the position index increases from top to bottom. The $\gamma \rightarrow$n and $\gamma \rightarrow \gamma$ upscatter blocks are normally empty.

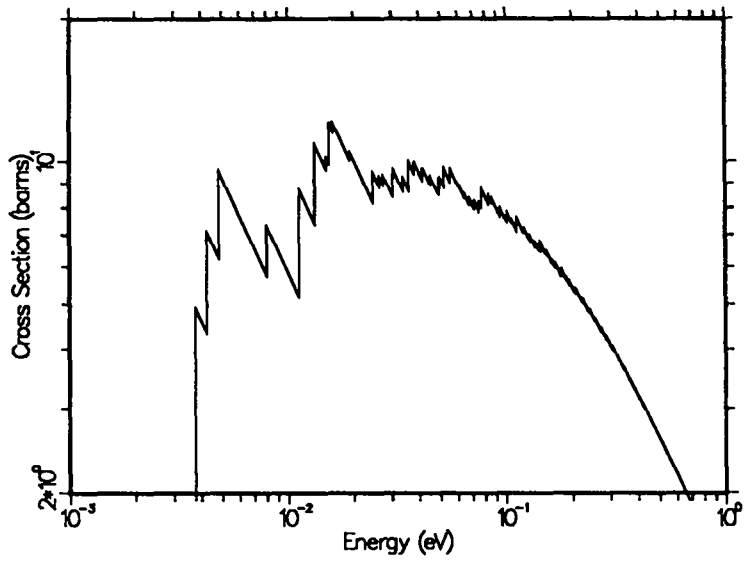


Figure 4: The coherent elastic scattering cross section for BeO showing the Bragg edges. The shape of $\sigma(E)$ between edges is $1/E$. Therefore, the function $E\sigma(E)$ is a stairstep function, where the height of each step depends on the structure factor for scattering from that set of lattice planes [see Eq. (69)].

where

$$\mu_i = 1 - 2\frac{E_i}{E} , \tag{68}$$

and where the E_i are the energies of the Bragg edges. THERMR integrates Eq. (67) over all angles, and writes the result to the PENDF tape. Clearly, the b_i can be recovered from

$$E\,\sigma^{coh}(E) = \sum_i{}' b_i , \tag{69}$$

where the primed sum is over all i such that $E_i < E$. It is only necessary to locate the steps in $E\,\sigma^{coh}(E)$. The size of the step gives b_i, and the E for the step gives E_i. The Legendre cross sections become

$$\sigma_\ell^{coh}(E) = \sum_i{}' \frac{b_i}{E} P_\ell(\mu_i) , \tag{70}$$

where any terms with $\mu_i < -1$ are omitted from the primed sum. An example of a pointwise cross section for coherent elastic scattering is given in Fig. 4.

For hydrogenous solids like polyethylene and zirconium hydride, the process of incoherent elastic scattering is important. Here the angular cross section is given

by

$$\sigma^{iel}(E,\mu) = \frac{\sigma_b}{2} \exp\left[-\frac{2EW}{A}(1-\mu)\right] , \tag{71}$$

(see THERMR, Chapter VIII of this manual). THERMR converts this into an integrated cross section, $\sigma^{iel}(E)$, and a set of N equally probable emission cosines, $\overline{\mu}_i$. These angles are present in File 6 on the PENDF tape. GROUPR can easily determine the Legendre components of the scattering cross section using

$$\sigma_\ell^{iel}(E) = \sigma^{iel}(E)\frac{1}{N}\sum_{i=1}^{N} P_\ell(\mu_i) . \tag{72}$$

The third thermal process is incoherent inelastic scattering. Here the neutron energy can either increase or decrease. The data from THERMR are given as a cross section in File 3 and an energy-angle distribution using a special form of File 6. The distribution is represented by sets of secondary-energy values E' for particular incident energies E. For each $E \rightarrow E'$, a scattering probability $f^{inc}(E \rightarrow E')$ and a set of equally probable cosines $\mu_i(E \rightarrow E')$ are given. The scattering probabilities for each value of E integrate to unity. Although the thermal scattering cross section is a smooth function of incident neutron energy, this is not true for the scattering from E to one particular final energy group g', since the differential cross section tends to peak along the line $E'=E$ and at energy-transfer values corresponding to well-defined excitations in the molecule or lattice. If interpolation between adjacent values of E were to be performed along lines of constant E', the excitation peaks and the $E=E'$ feature would produce double features in the intermediate spectrum, as shown in Fig. 5. To avoid this problem, while still using a relatively sparse incident energy grid, GROUPR interpolates between E and E' along lines of constant energy transfer. Of course, this breaks down at low values of E', because one of the spectra will go to zero before the other one does. In this range, GROUPR transforms the low-energy parts of the two spectra onto a "unit base," combines them in fractions that depend on E, and scales the result back out to the interpolated value of E' corresponding to E.

K. Generalized Group Integrals

In order to unify many formerly different processing tasks, GROUPR uses the concept of a generalized group integral

$$\sigma_g = \frac{\int_g \mathcal{F}(E)\,\sigma(E)\,\phi(E)\,dE}{\int_g \phi(E)\,dE} , \tag{73}$$

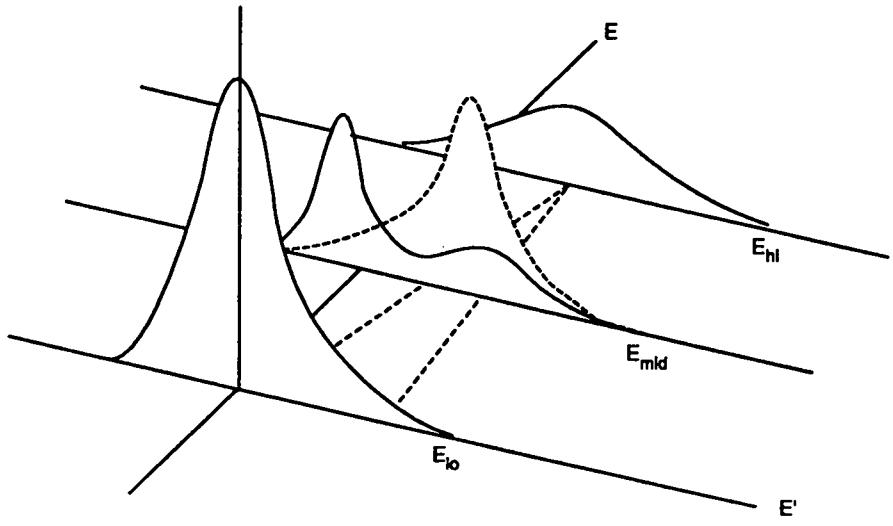


Figure 5: Illustration of thermal interpolation showing the double-humped curve resulting from simple Cartesian interpolation for a discrete excitation (solid) and the more realistic curve obtained by interpolating along lines of constant energy transfer (dashed).

where the integrals are over all initial neutron energies in group g, $\sigma(E)$ is a cross section at E, and $\phi(E)$ is an estimate of the flux at E. The function $\mathcal{F}(E)$ is called the "feed function". It alone changes for different data types. To average a neutron cross section, $\mathcal{F}$ is set to 1. To average a ratio quantity like $\overline{\mu}$ with respect to elastic scattering, $\mathcal{F}$ is set to $\overline{\mu}$. For photon production, $\mathcal{F}$ is the photon yield. For matrices, $\mathcal{F}$ is the ℓ-th Legendre component of the normalized probability of scattering into secondary energy group g' from initial energy E. This definition is clearly independent of whether the secondary particle is a neutron or a photon.

The question of integration grid or quadrature scheme is important for the evaluation of Eq. (73). Each factor in the integrands has its own characteristic features, and it is important to account for them all. First, a grid must be established for each factor. As an example, the grid of $\sigma(E)$ is generated in RECONR such that sigma can be obtained to within a given tolerance by linear interpolation. GROUPR contains a subroutine **GETSIG** which carries out this interpolation at E and also returns the next grid energy in **ENEXT**. Subroutines **GETFLX** and **GETFF** perform similar functions for the flux and feed function. It is now easy to generate a union grid for the three-factor integrand using the following FORTRAN:

```
...
CALL GETSIG(E,ENEXT,...)
CALL GETFLX(E,EN,...)
```

```
IF (EN.LT.ENEXT) ENEXT=EN
CALL GETFF(E,EN,...)
IF (EN.LT.ENEXT) ENEXT=EN
 ...
```

On occasion, there will be a discontinuity at ENEXT for one of the factors. In order to flag such a case, each "GET" routine also sets a discontinuity flag IDISC. The grid logic actually used throughout NJOY is as follows:

```
...
CALL GETSIG(E,ENEXT,IDISC,...)
CALL GETFLX(E,EN,IDIS,...)
IF (EN.EQ.ENEXT.AND.IDIS.GT.IDISC) IDISC=IDIS
IF (EN.LT.ENEXT) IDISC=IDIS
IF (EN.LT.ENEXT) ENEXT=EN
CALL GETFF(E,EN,IDIS,...)
IF (EN.EQ.ENEXT.AND.IDIS.GT.IDISC) IDISC=IDIS
IF (EN.LT.ENEXT) IDISC=IDIS
IF (EN.LT.ENEXT) ENEXT=EN
...
```

This union grid for the integrand in the numerator is used to subdivide the generalized group integral of Eq. (73) into "panels". The main program of GROUPR carries out the integrals with the following logic:

```
    ...
    ELO=EGN(IG)
    EHI=EGN(IG+1)
    ENEXT=EHI
230 CALL PANEL(ELO,ENEXT,...)
    IF (ENEXT.EQ.EHI) GO TO 240
    ELO=ENEXT
    ENEXT=EHI
    GO TO 230
240 CONTINUE
    ...
```

A panel is first defined by the energy bounds of group g. Subroutine PANEL is called to sum in the contributions from this panel. However, PANEL discovers that the integrand has a grid point at ENEXT less than EHI. It adds in the contributions for the smaller panel ELO to ENEXT and returns. GROUPR now sees that ENEXT is less than EHI, so it tries a new panel from the top of the last panel (ELO=ENEXT) to EHI. This loop continues until a panel with EHI as its upper bound is summed in. The integral for this group is then complete.

In this simple way, the algorithm accounts for the user's group structure and for all structure in the integrand. The method used for establishing the RECONR

grid makes this integration algorithm equivalent to adaptive integration as used in MINX[10]. It has the great advantage that no "stack" of intermediate results is carried along. This single-pass feature of the quadrature scheme allows many different integrals to be accumulated simultaneously within reasonable storage limits. In this way, GROUPR accumulates cross sections for all values of σ_0 simultaneously. Similarly, group-to-group cross sections are computed for all secondary energy groups and all Legendre orders simultaneously.

Any degree of complexity can be used for the integral over each subpanel. Because $\sigma(E)$ has been linearized, **PANEL** is based on trapezoidal integration. Both $\phi(E)$ and $R(E) = \sigma(E) \times \phi(E)$ are assumed to vary linearly across each panel. In some cases, the feed function is oscillatory over a certain energy range (see Two-Body Scattering, Section VIII.L). It is then convenient to integrate inside the panel using Lobatto quadrature[11] (note the variable **NQ** in **PANEL**). As discussed in more detail later, this method can obtain accurate results for an oscillatory function whose integral is small with respect to its magnitude. This behavior is characteristic of the higher Legendre components of two-body scattering cross sections.

The generalized integration scheme discussed here is also used by the GAMINR and ERRORR modules.

L. Two-Body Scattering

Elastic scattering (ENDF/B MT=2) and discrete inelastic neutron scattering (with MT=51–90) are both examples of two-body kinematics and are treated together by GROUPR. Some other cases that occur for charged particles or in File 6 will be discussed later. The feed function required for the group-to-group matrix calculation may be written

$$\mathcal{F}_{\ell g'}(E) = \int_{g'} dE' \int_{-1}^{+1} d\omega \, f(E \rightarrow E', \omega) \, P_\ell(\mu[\omega]) \, , \tag{74}$$

where $f(E \rightarrow E', \omega)$ is the probability of scattering from E to E' through a center-of-mass cosine ω and P_ℓ is a Legendre polynomial for laboratory cosine μ. The laboratory cosine corresponding to ω is given by

$$\mu = \frac{1 + R\omega}{\sqrt{1 + R^2 + 2R\omega}} \, , \tag{75}$$

and the cosine ω is related to secondary energy E' by

$$\omega = \frac{E'(1+A)^2/A' - E(1+R^2)}{2RE} , \tag{76}$$

where A' is the ratio of the emitted particle mass to the incident particle mass (A'=1 for neutron scattering). In Eqs. (75) and (76), R is the effective mass ratio

$$R = \sqrt{\frac{A(A+1-A')}{A'}}\sqrt{1 - \frac{(A+1)(-Q)}{AE}} , \tag{77}$$

where A is the ratio of target mass to neutron mass, and $-Q$ is the energy level of the excited nucleus (Q=0 for elastic scattering). Integrating Eq. (74) over secondary energy gives

$$\mathcal{F}_{\ell g'}(E) = \int_{\omega_1}^{\omega_2} f(E,\omega)\, P_\ell(\mu[\omega])\, d\omega , \tag{78}$$

where ω_1 and ω_2 are evaluated using Eq. (76) for E' equal to the upper and lower bounds of g', respectively.

The scattering probability is given by

$$f(E,\omega) = \sum_{\ell=0}^{L} f_\ell(E)\, P_\ell(\omega) , \tag{79}$$

where the Legendre coefficients are either retrieved directly from the ENDF/B File 4 or computed from File 4 tabulated angular distributions (see subroutines GETFLE and GETCO).

The integration in Eq. (78) is performed (see subroutine GETDIS) simultaneously for all Legendre components using Gaussian quadrature.[11] The quadrature order is selected based on the estimated polynomial order of the integrand. A reasonable estimate is given by

$$\mathrm{ND} + \mathrm{NL} + \log(300/A) , \tag{80}$$

where ND is the number of Legendre components desired for the feed function, and NL is the number of components required to represent $f(E,\omega)$. The log term approximates the number of additional components required to represent the center-of-mass to lab transformation.

The two-body feed function for higher Legendre orders is a strongly oscillatory function in some energy ranges. An example is shown in Fig. 6. Furthermore, the

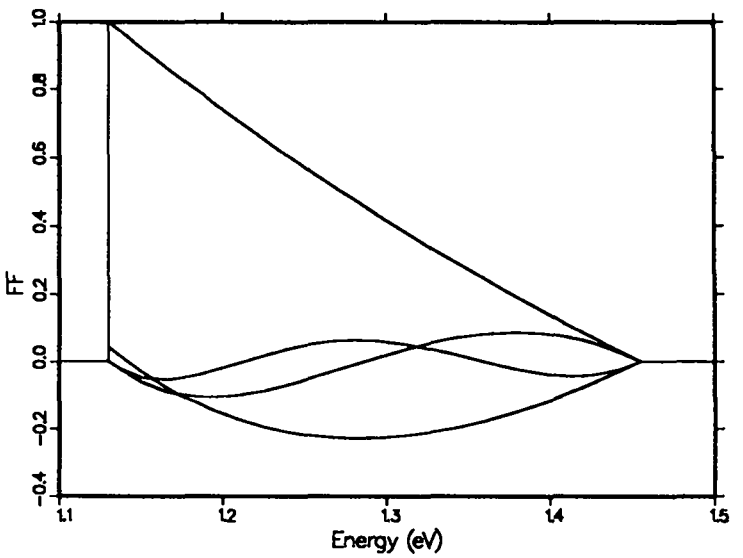


Figure 6: A typical feed function for two-body scattering showing the oscillations that must be treated correctly by the integration over incident energy.

integral of the oscillatory part is often small with respect to the magnitude of the function. Such functions are very difficult to integrate with adaptive techniques, which converge to some fraction of the integral of the absolute value. This is the reason that MINX[10] gives poor answers for small Legendre components of the scattering matrix. Gaussian methods, on the other hand, are capable of integrating such oscillatory functions exactly if they are polynomials. Since a polynomial representation of the feed function is fairly accurate, a Gaussian quadrature scheme was chosen. The scheme used is also well adapted to performing many integrals in parallel. In GROUPR, all Legendre components and all final groups are accumulated simultaneously (see **PANEL**).

The boundaries of the various regions of the feed function are called "critical points." Between critical points, the feed function is a smooth analytic function of approximately known polynomial order. It is only necessary to add these critical points to the incident energy grid of the feed function (the **ENEXT** variable) and to tell **PANEL** what quadrature order (**NQ**) to use. The critical points are determined in **GETFF** by solving Eq. (76) for the values of E for which $\omega = +1$ and $\omega = -1$ when E' is equal to the various group boundaries. This can be done by writing

$$\frac{E_g}{E} = \frac{1}{(1+A)^2}[R^2 \pm 2R + 1] , \tag{81}$$

substituting for E using Eq. (77), and then solving for R. The result is

$$E_{crit} = \frac{\frac{(A+1)(-Q)}{A}}{1 - \frac{A'F^2}{A(A+1-A')}} , \tag{82}$$

where

$$F = \frac{1 \pm \sqrt{D}}{1 + \frac{E_F}{(-Q)}} , \tag{83}$$

$$D = \left[\frac{A(A+1-A')}{A'}\left(1 + \frac{E_F}{(-Q)}\right) - 1\right]\frac{E_F}{(-Q)} , \tag{84}$$

and

$$E_F = \frac{1+A}{A+1-A'}E_g . \tag{85}$$

File 4 can also contain angular distributions for charged-particle emission through discrete levels (see MT=600–648 for protons, MT=650–698 for deuterons, and so on; the elastic case, MT=2, is discussed in the next section). Moreover, File 6 can contain angular distributions for discrete two-body scattering (see Law 3). It can also declare that a particular particle is the recoil particle from a two-body reaction (Law 4), in which case the appropriate angular distribution is obtained from the corresponding Law 3 subsection by complementing the angle. The representation of the angular distribution for these laws is almost identical to that in File 4, and the calculation is done in GETDIS using much of the File 4 coding. The effects on the kinematics of the difference in the mass of the emitted and incident particles are handled by the variable A' in the above equations.

M. Charged-Particle Elastic Scattering

Coulomb scattering only occurs in the elastic channel, and this calculation also uses the GETDIS subroutine. The problem with Coulomb scattering is that the basic Rutherford formula becomes singular at small angles. In practice, this singularity is removed by the screening effects of the atomic electrons. The forward transport of charged particles in this screening regime is usually handled by continuous-slowing-down theory by using a "stopping power". The new ENDF-6 format allows for three different ways to handle the large-angle scattering regime. First and most general is the nuclear amplitude expansion:

$$\begin{aligned} \sigma &= |\text{nucl} + \text{coul}|^2 \\ &= \sigma_{\text{nucl}} + \sigma_{\text{coul}} + \text{interference} \end{aligned} \tag{86}$$

The Coulomb term is given by the Rutherford formula, a Legendre expansion is defined for the nuclear term, and a complex Legendre expansion is defined for the interference term. This representation cannot be generated directly from experimental data; an R-matrix or phase-shift analysis is necessary.

A method very closely related to experiment (σ_{exp}) is the nuclear plus interference (NI) formula:

$$\sigma_{\text{NI}}(\mu, E) = \sigma_{\text{exp}}(\mu, E) - \sigma_{\text{coul}}(\mu, E), \tag{87}$$

where the function is only defined for angles with cosines $\mu<\mu_{\text{max}}$. The minimum angle is usually taken to be somewhere around 20 degrees (GROUPR uses μ_{max}=0.96). This function is still ill-behaved near the cutoff, and it must be tabulated. The third option is the residual cross section expansion:

$$\sigma_R(\mu, E) = (1 - \mu)[\sigma_{\text{exp}}(\mu, E) - \sigma_{\text{coul}}(\mu, E)]. \tag{88}$$

The $(1-\mu)$ term removes the pole at the origin. The residual is uncertain, but it is usually small enough that the entire curve can be fitted with Legendre polynomials without worrying about what happens at small angles. In practice, both the nuclear amplitude expansion and the nuclear plus interference representation are converted to the residual cross section representation in subroutine `CONVER`. As a result, `GETDIS` only has to cope with the one representation.

N. Continuum Scattering and Fission

In ENDF/B, scattering from many closely-spaced levels or multibody scattering such as (n,2n), (n,n'α), and fission is often represented using a separable function of scattering cosine and secondary neutron energy

$$f(E\rightarrow E', \mu) = F(E, \mu)\, g(E\rightarrow E') \,, \tag{89}$$

where F is the probability that a neutron will scatter through a laboratory angle with cosine μ irrespective of final energy E'. It is obtained from MF=4. Similarly, g is the probability that a neutron's energy will change from E to E' irrespective of the scattering angle, and it is given in MF=5. Continuum reactions are easily identified by MT numbers of 6–49 and 91. The secondary-energy distributions can be represented in the following ways:

Law	Description
1	Arbitrary tabulated function
5	General evaporation spectrum (Used for delayed neutrons only.)
7	Simple Maxwellian fission spectrum
9	Evaporation spectrum
11	Energy-dependent Watt fission spectrum
12	Energy-dependent spectrum of Madland and Nix

The feed functions for continuum scattering are simply

$$\mathcal{F}_{\ell g'}(E') = \int_{-1}^{+1} P_\ell(\mu)\, f(E,\mu)\, d\mu \int_{g'} g(E,E')\, dE' \,. \tag{90}$$

The first integral is returned by GETFLE [FLE for $f_\ell(E)$] as described above, and the second integral is returned by GETSED (SED for secondary-energy distribution).

For Law 1, g is given as a tabulated function of E' for various values of E. When $E_1 \le E < E_2$, the term $\int_{g'} g(E,E')\, dE'$ is obtained by interpolating between precomputed values of $\int_{g'} g(E_1,E')\, dE'$ and $\int_{g'} g(E_2,E')\, dE'$ in subroutine GETSED. Except in the case of fission, any apparent upscatter produced by the "stairstep" treatment near E=E' is added to the in-group scattering term (g'=g).

For Law 5, $g(x)$ is given versus $x = E'/\theta(E)$ and $\theta(E)$ is given vs. E in File 5. This secondary neutron distribution leads to the following group integral:

$$\int_{g'} g(E,E')\, dE' = \theta(E) \int_{E_1/\theta}^{E_2/\theta} g(x)\, dx \,, \tag{91}$$

with E_1 and E_2 being the lower and upper boundary energies for group g'.

For Law 7, the secondary-energy distribution is given by

$$g(E,E') = \frac{\sqrt{E'}}{I} \exp\left(-\frac{E'}{\theta(E)}\right) , \tag{92}$$

where the effective temperature $\theta(E)$ is tabulated in File 5 and the normalization factor is given by

$$I = \theta^{3/2}[\sqrt{\frac{\pi}{4}}\,\mathrm{erf}(x) - xe^{-x}] \,, \tag{93}$$

where

$$x = \frac{E-U}{\theta} \,. \tag{94}$$

Here U is a constant used to define the upper limit of secondary neutron energy $\theta \le E' \le E-U$. The desired group integral is given by

$$\int_{g'} g(E,E')\, dE' \;=\; \int_{E_1}^{E_2} g(E,E')\, dE'$$

$$= \frac{X_1 - X_2 - Y_1 + Y_2}{\sqrt{\pi/4} - Y - X} , \tag{95}$$

where

$$X = \sqrt{x}\, e^{-x} , \tag{96}$$

$$Y = \sqrt{\frac{\pi}{4}}\, \mathrm{rerfc}(\sqrt{x}) , \tag{97}$$

and where X_1, Y_1, X_2, and Y_2 refer to X and Y evaluated at E_1/θ and E_2/θ. The integral of Eq. (95) is computed in **ANASED** ("**ANA**" for analytic). The function "rerfc" is the reduced complementary error function[11].

For Law 9, the secondary-energy distribution is given by

$$g(E, E') = \frac{E'}{I} \exp\left(-\frac{E'}{\theta(E)}\right) , \tag{98}$$

where

$$I = \theta^2[1 - e^{-x}(1 + x)] . \tag{99}$$

Here x has the same meaning as above—see Eq. (94). The group integral is

$$\int_{g'} g(E, E')\, dE' = \frac{(1 + x_1)e^{-x_1} - (1 + x_2)e^{-x_2}}{1 - (1 + x)e^{-x}} , \tag{100}$$

where x_1 and x_2 refer to E_1/θ and E_2/θ, respectively. This result is also computed in **ANASED**.

For Law 11, the secondary-energy distribution is given by

$$g(E, E') = \frac{e^{-E'/a}}{I} \sinh(\sqrt{bE'}) , \tag{101}$$

where

$$I = \frac{1}{2}\sqrt{\frac{\pi a^3 b}{4}}\, e^{x_0}[\mathrm{erf}(\sqrt{x} - \sqrt{x_0}) + \mathrm{erf}(\sqrt{x} + \sqrt{x_0})] - ae^{-x \sinh(abx)} . \tag{102}$$

In this case, $a(E)$ and $b(E)$ are given in File 5,

$$x = \frac{E - U}{a} , \tag{103}$$

and

$$x_0 = \frac{ab}{4} . \tag{104}$$

The group integrals are given by

$$\int_{g'} g(E,E') = \frac{H(x_1,x_2,x_0)}{H(0,x,x_0)} , \tag{105}$$

where

$$\begin{aligned} H(x_1,x_2,x_0) &= H_2(\sqrt{x_1}-\sqrt{x_0},\sqrt{x_2}-\sqrt{x_0}) \\ &+ \sqrt{x_0}H_1(\sqrt{x_1}-\sqrt{x_0},\sqrt{x_2}-\sqrt{x_0}) \\ &- H_2(\sqrt{x_1}+\sqrt{x_0},\sqrt{x_2}+\sqrt{x_0}) \\ &+ \sqrt{x_0}H_1(\sqrt{x_1}+\sqrt{x_0},\sqrt{x_2}+\sqrt{x_0}) , \end{aligned} \tag{106}$$

and where

$$H_n(a,b) = \frac{1}{\pi}\int_a^b z^n e^{-z^2}\,dz . \tag{107}$$

The methods for computing H_n are described in BROADR, Chapter IV of this manual. When $\sqrt{x_2} < .01$, a short-cut calculation can be used for the numerator of Eq. (105)

$$\frac{4\sqrt{x_0}e^{-x_0}}{3\sqrt{\pi/4}}[x_2^{3/2}-x_1^{3/2}] . \tag{108}$$

For Law 12, the Madland-Nix option, the secondary-energy distribution is given by

$$g(E,E') = \frac{1}{2}[G(E',E_{fl})+G(E',E_{fh})] , \tag{109}$$

where

$$\begin{aligned} &G(E',E_f) \\ &\quad = \frac{1}{3\sqrt{E_f T_m}}\left[u_2^{3/2}E_1(u_2) - u_1^{3/2}E_1(u_1) + \gamma(3/2,u_2) - \gamma(3/2,u_1)\right] , \end{aligned} \tag{110}$$

$$u_1 = (\sqrt{E'}-\sqrt{E_f})^2/T_m \text{ , and} \tag{111}$$

$$u_2 = (\sqrt{E'}+\sqrt{E_f})^2/T_m , \tag{112}$$

and where E_{fl}, E_{fh}, and $T_m(E)$ are given in File 5. The special functions used are the first-order exponential integral, $E_1(x)$, and the incomplete gamma function, $\gamma(n,x)$. The group integrals of this function are very complex[12]. Let

$$\alpha = \sqrt{T_m} , \tag{113}$$

$$\beta = \sqrt{E_f} , \tag{114}$$

$$A = \frac{(\sqrt{E_1}+\beta)^2}{\alpha^2}, \tag{115}$$

$$B = \frac{(\sqrt{E_2}+\beta)^2}{\alpha^2}, \tag{116}$$

$$A' = \frac{(\sqrt{E_1}-\beta)^2}{\alpha^2}, \text{ and} \tag{117}$$

$$B' = \frac{(\sqrt{E_2}-\beta)^2}{\alpha^2}. \tag{118}$$

Then the integral over the range (E_1, E_2) of G is given by one of the following three expression, depending on the region of integration in which E_1 and E_2 lie.

Region I ($E_1 \geq E_f, E_2 > E_f$)

$$\begin{aligned}
&3\sqrt{E_f T_m}\int_{E_1}^{E_2} G(E', E_f)\,dE' \\
&\quad = [(\tfrac{2}{5}\alpha^2 B^{5/2} - \tfrac{1}{2}\alpha\beta B^2)E_1(B) - (\tfrac{2}{5}\alpha^2 A^{5/2} - \tfrac{1}{2}\alpha\beta A^2)E_1(A)] \\
&\quad - [(\tfrac{2}{5}\alpha^2 B'^{5/2} + \tfrac{1}{2}\alpha\beta B'^2)E_1(B') - (\tfrac{2}{5}\alpha^2 A'^{5/2} + \tfrac{1}{2}\alpha\beta A'^2)E_1(A')] \\
&\quad + [(\alpha^2 B - 2\alpha\beta B^{1/2})\,\gamma(3/2, B) - (\alpha^2 A - 2\alpha\beta A^{1/2})\,\gamma(3/2, A)] \\
&\quad - [(\alpha^2 B' + 2\alpha\beta B'^{1/2}\,\gamma(3/2, B') - (\alpha^2 A' + 2\alpha\beta A'^{1/2})\,\gamma(3/2, A')] \\
&\quad - \tfrac{3}{5}\alpha^2[\gamma(5/2, B) - \gamma(5/2, A) - \gamma(5/2, B') + \gamma(5/2, A')] \\
&\quad - \tfrac{3}{2}\alpha\beta\,[\mathrm{e}^{-B}(1+B) - \mathrm{e}^{-A}(1+A) + \mathrm{e}^{-B'}(1+B') - \mathrm{e}^{-A'}(1+A')]\,.
\end{aligned} \tag{119}$$

Region II ($E_1 < E_f, E_2 \leq E_f$)

$$\begin{aligned}
&3\sqrt{(E_f T_m)}\int_{E_1}^{E_2} G(E', E_f)\,dE' \\
&\quad = [(\tfrac{2}{5}\alpha^2 B^{5/2} - \tfrac{1}{2}\alpha\beta B^2)E_1(B) - (\tfrac{2}{5}\alpha^2 A^{5/2} - \tfrac{1}{2}\alpha\beta A^2)E_1(A)] \\
&\quad - [(\tfrac{2}{5}\alpha^2 B'^2)E_1(B') - (\tfrac{2}{5}\alpha^2 A'^{5/2} - \tfrac{1}{2}\alpha\beta A'^2)E_1(A')] \\
&\quad + [(\alpha^2 B - 2\alpha\beta B^{1/2})\,\gamma(3/2, B) - (\alpha^2 A - 2\alpha\beta A^{1/2})\,\gamma(3/2, A)] \\
&\quad - [(\alpha^2 B' - 2\alpha\beta B'^{1/2}\,\gamma(3, 2, B') - (\alpha^2 A' - 2\alpha\beta A'^{1/2})\,\gamma(3/2, A')] \\
&\quad - \tfrac{3}{5}\alpha\beta[\gamma(5/2, B) - \gamma(5/2, A) - \gamma(5/2, B') + \gamma(5/2, A')] \\
&\quad - \tfrac{3}{2}\alpha\beta\,[\mathrm{e}^{-B}(1+B) - \mathrm{e}^{-A}(1+A) - \mathrm{e}^{-B'}(1+B') + \mathrm{e}^{-A'}(1+A')]\,.
\end{aligned} \tag{120}$$

Region III ($E_1 < E_f, E_2 > E_f$)

$$
\begin{aligned}
&3\sqrt{E_f T_m}\int_{E_1}^{E_2} G(E', E_f)\, dE' \\
&\quad = [(\tfrac{2}{5}\alpha^2 B^{5/2} - \tfrac{1}{2}\alpha\beta B^2)E_1(B) - (\tfrac{2}{5}\alpha^2 A^{5/2} - \tfrac{1}{2}\alpha\beta A^2)E_1(A)] \\
&\quad - [(\tfrac{2}{5}\alpha^2 B'^{5/2} + \tfrac{1}{2}\alpha\beta B'^2)E_1(B') - (\tfrac{2}{5}\alpha^2 A'^{5/2} - \tfrac{1}{2}\alpha\beta A'^2)E_1(A')] \\
&\quad + [(\alpha^2 B - 2\alpha\beta B^{1/2})\,\gamma(3/2, B) - (\alpha^2 A - 2\alpha\beta A^{1/2})\,\gamma(3, 2, A)] \\
&\quad - [(\alpha^2 B' + 2\alpha\beta B'^{1/2})\,\gamma(3/2, B') - (\alpha^2 A' - 2\alpha\beta A'^{1/2})\,\gamma(3/2, A'_)] \\
&\quad - \tfrac{3}{5}\alpha^2[\gamma(5/2, B) - \gamma(5/2, A) - \gamma(5/2, B') + \gamma(5/2, A')] \\
&\quad - \tfrac{3}{2}\alpha\beta\,[e^{-B}(1+B) - e^{-A}(1+A) + e^{-B'}(1+B') + e^{-A'}(1+A') - 2]\,.
\end{aligned}
\tag{121}
$$

O. File 6 Energy-Angle Distributions

If the File 6 data are expressed as a continuous energy-angle distribution (Law 1) in the laboratory system, it is fairly easy to generate the multigroup transfer matrix. As usual for GROUPR, the task is to calculate the "feed function" (the Legendre moments for transferring to all possible secondary-energy groups starting with incident energy E). The E' integration is controlled by the GETMF6 subroutine, which calls **F6LAB** to generate the integrands. This routine expects data in Law-1 format, where the looping order is E, E', ℓ. The only problems here are handling the new ENDF-6 interpolation laws for two-dimensional tabulations (for example, unit base) and converting tabulated angle functions for $E \rightarrow E'$ into Legendre coefficients (which is done with a Gauss-Legendre quadrature of order 8).

If the File 6 data are given in the energy-angle form of Law 7 (where the looping order is E, μ, E'), GROUPR converts it to the Law 1 form using subroutine **LL2LAB**. It does this by stepping through an E' grid that is the union of the E' grids for all the different angles given. At each of these union E' values, it calculates the Legendre coefficients using a Gauss-Legendre quadrature of order 8, and checks back to see if the preceding point is still needed to represent the distribution to the desired degree of accuracy. Now **GETMF6** and **F6LAB** can be used to complete the calculation as above.

If the File 6 data are expressed in the CM system, or if the phase-space option is used, more processing is necessary to convert to the LAB system. This conversion is done for each incident energy E given in the file. The grid for laboratory

secondary energy E'_L is obtained by doing an adaptive reconstruction of the emission probability $p_{L\ell}(E, E'_L)$ such that each Legendre order can be expressed as a linear-linear function of E'_L. This part is done in subroutine **CM2LAB**. The values for $p_{L\ell}(E, E'_L)$ are obtained in subroutine **F6CM** by doing an adaptive integration along the contour E'_L=const in the E'_C, μ_C plane using μ_L as the variable of integration:

$$p_{L\ell}(E, E'_L) = \int_{\mu_{\min}}^{+1} p_C(E, E'_C, \mu_C)\, P_\ell(\mu_L)\, J\, d\mu_L \ , \tag{122}$$

where μ is a scattering cosine and L and C denote the laboratory and center-of-mass (CM) systems, respectively. The Jacobian of the transformation is given by

$$J = \sqrt{\frac{E'_L}{E'_C}} = \frac{1}{\sqrt{1 + c^2 - 2c\mu_L}} \ , \tag{123}$$

and the cosine transformation is given by

$$\mu_C = J(\mu_L - c) \ . \tag{124}$$

The constant c is given by

$$c = \sqrt{\frac{A'}{(A+1)^2}} \sqrt{\frac{E}{E'_L}} \ , \tag{125}$$

where A is the ratio of the atomic weight of the target to the atomic weight of the projectile, and A' is the ratio of the atomic weight of the emitted particle to the atomic weight of the projectile. The lower limit of the integral depends on the maximum possible value for the center-of-mass (CM) secondary energy as follows:

$$\mu_{\min} = \frac{1}{2c}\left(1 + c^2 - \frac{E'_{C\max}}{E'_L}\right), \tag{126}$$

where

$$E'_{L\max} = E\left(\sqrt{\frac{E'_{C\max}}{E}} + \sqrt{\frac{A'}{(A+1)^2}}\right)^2 . \tag{127}$$

An example of the integration contours for this coordinate transformation is given in Fig. 7.

The CM energy-angle distribution can be given as a set of Legendre coefficients or a tabulated angular distribution for each possible energy transfer $E \rightarrow E'$, as a "precompound fraction" $r(E, E')$ for use with the Kalbach-Mann[13] or Kalbach[14] angular distributions, or as parameters for a phase-space distribution. The first

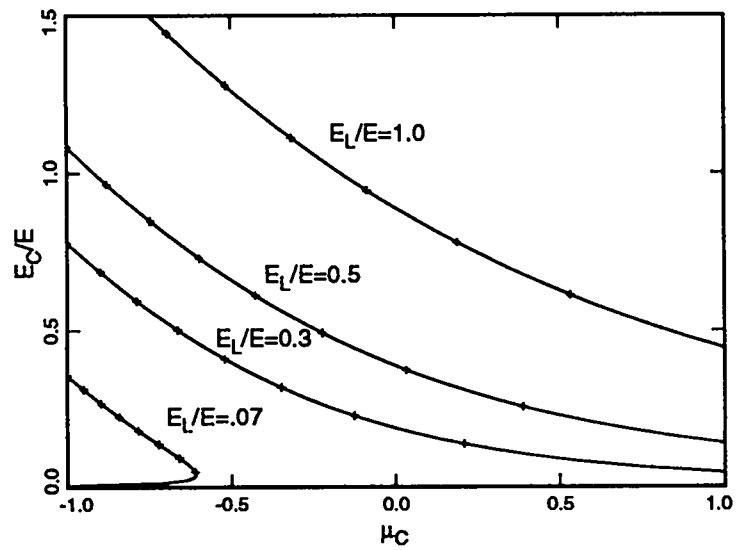


Figure 7: Coordinate mapping between CM and laboratory reference frames for A=2. The parameters E_L and E_C are the secondary energies in the lab and CM frames. The crosses on the curves are at μ_L values of -1.0, -.75, -.50, ..., .75, and 1.0. This figure also illustrates the advantage of integrating over μ_L for the contour in the lower-left corner. The values of E_C and μ_C are single-values functions of μ_L.

three options are processed using F6DDX, and the last using F6PSP. The Kalbach option leads to a very compact representation. Kalbach and Mann examined a large number of experimental angular distributions for neutrons and charged particles. They noticed that each distribution could be divided into two parts: an equilibrium part symmetric in μ, and a forward-peaked preequilibrium part. The relative amount of the two parts depended on a parameter r, the preequilibrium fraction, that varies from zero for low E' to 1.0 for large E'. The shapes of the two parts of the distributions depended most directly on E'. This representation is very useful for preequilibrium statistical-model codes like GNASH,[15] because they can compute the parameter r, and all the rest of the angular information comes from simple universal functions. More specifically, Kalbach's latest work says that

$$f(\mu) = \frac{a}{2\sinh(a)}\Big[\cosh(a\mu) + r\sinh(a\mu)\Big], \tag{128}$$

where a is a simple function of E, E', and B_b, the separation energy of the emitted particle from the liquid-drop model without pairing and shell terms. The separation energies are computed by function SEPE. This function has a problem for elemental evaluations, because it needs an A value for the calculation, and it is difficult to guess which A value is most characteristic of the element. A short table is included in the function, and an "ERROR IN SEPE" will result if the function is called with an element that doesn't appear in the table. Similar routines appear in HEATR and ACER (SEPH and SEP, respectively). A better long-range solution would be desirable.

The File 6 processing methods in GROUPR apply equally well to neutrons and charged particles. The effects on the kinematics due to the difference in mass between the incident particle and the emitted particle are handled by the variable A' in the above equations.

P. GENDF Output

The group constants produced by GROUPR are normally written to an output file in GENDF (groupwise ENDF) format for use by other modules of NJOY. For example, DTFR can be used to convert a GENDF material to DTF (or "transport") format; CCCCR produces the standard interface files[16] ISOTXS, BRKOXS, and DLAYXS; MATXSR produces a file in MATXS format;[6] and POWR produces libraries of the Electric Power Research Institute (EPRI) codes EPRI-CELL and EPRI-CPM. Other formats can easily be produced by new modules, and some functions such as group collapse, are conveniently performed di-

rectly in GENDF format. The GENDF format is also used for GAMINR output, and both ACER and ERRORR use GENDF input for some purposes.

Depending on the sign of NGOUT2 (see below), the GENDF file will be written in either formatted mode (i.e, BCD, ASCII, EBCDIC, etc.) or in the special NJOY blocked-binary mode. Conversion between these modes can be performed subsequently by the MODER module.

The GENDF material begins with a header record (MF=1, MT=451), but the format of this first section is different from MT=451 on an ENDF or PENDF tape. The section consists of a "CONT" record, containing

ZA	Standard 1000*Z+A value
AWR	Atomic weight ratio to neutron
0	Zero
NZ	Number of σ_0 values
-1	Identifies a GENDF-type data file
NTW	Number of words in title

and a single "LIST" record, containing

TEMPIN	Material temperature (Kelvin)
0.	Zero
NGN	Number of neutron groups
NGG	Number of photon groups
NW	Number of words in LIST
0	Zero
TITLE	Title from GROUPR input (NTW words)
SIGZ	Sigma-zero values (NZ words)
EGN	Neutron group boundaries, low to high (NGN+1 words)
EGG	Photon group boundaries, low to high (NGG+1 words).

For photoatomic GENDF files produced by the GAMINR module, the photon group structure is stored in NGN and EGN, and the number of photon groups is given as NGG=0. The word count is seen to be NW=NTW+NZ+NGN+1+NGG+1. The LIST record is followed by a standard ENDF file-end record (FEND). The normal ENDF section-end (SEND) is omitted.

This header is followed by a series of records for reactions. The ENDF ordering requirements are relaxed, and MF and MT values can occur in any order. Each section starts with a "CONT" record.

ZA	Standard 1000*Z+A value
AWR	Standard atomic weight ratio
NL	Number of Legendre components
NZ	Number of sigma-zero values
LRFLAG	Break-up identifier flag
NGN	Number of groups

It is followed by a series of LIST records, one for every incident-energy group with nonzero result,

TEMP	Material temperature (Kelvin)
0.	Zero
NG2	Number of secondary positions
IG2LO	Index to lowest nonzero group
NW	Number of words in LIST
IG	Group index for this record
A(NW)	Data for LIST (NW words),

where NW=NL*NZ*NG2. The last LIST record in the sequence is the one with IG=NGN. It must be given even if its contents are zero. The last LIST record is followed by a SEND record.

The contents of A(NW) change for various types of data. For simple cross section "vectors" (MF=3), NG2 is 2, and A contains the two FORTRAN arrays

```
FLUX(NL,NZ), SIGMA(NL,NZ)
```

in that order. For ratio quantities like fission $\overline{\nu}$, NG2 is 3, and A contains

```
FLUX(NL,NZ), RATIO(NL,NZ), SIGMA(NL,NZ).
```

For transfer matrices (MF=6, 16, 21, etc.), A contains

```
FLUX(NL,NZ), MATRIX(NL,NZ,NG2-1).
```

The actual secondary group indices for the last index of MATRIX are usually IG2LO, IG2LO+1, etc., using the GROUPR convention of labeling groups in order of increasing energy. If the low-energy part of the fission matrix (or the fission or capture photon production matrices) uses the special format described in Section F, the spectrum will be found in a LIST record with IG=0 and the production cross section will be found in a series of records with IG2LO=0. The group range for the spectrum ranges from IG2LO to IG2LO+NG2-1. For IG2LO=0, NG2 will be 2 as for a normal cross section, and the two values will be the flux for group IG and the corresponding production cross section.

Finally, for delayed neutron spectra (MF=5), NL is used to index the 6 time groups, NZ is 1, and there is only one incident energy record (IG=IGN). The array A contains

```
LAMBDA(NL), CHID(NL,NG2-1),
```

where LAMBDA contains the 6 delayed-neutron time constants and CHID contains the six spectra.

The GENDF material ends with a material-end (MEND) record, and the GENDF tape ends with a tape-end (TEND) record.

Q. Running GROUPR

The input instructions follow. They are reproduced from the comment cards at the beginning of the 91.0 version of the GROUPR module. Because the code changes from time to time, it is a good idea to check these comment cards in the current version to obtain up-to-date input instructions.

```
*---INPUT SPECIFICATIONS (FREE FORMAT)-----------------------------*
*                                                                  *
* CARD1                                                           *
*    NENDF   UNIT FOR ENDF/B TAPE                                 *
*    NPEND   UNIT FOR PENDF TAPE                                  *
*    NGOUT1  UNIT FOR INPUT GOUT TAPE (DEFAULT=0)                 *
*    NGOUT2  UNIT FOR OUTPUT GOUT TAPE (DEFAULT=0)                *
* CARD2                                                           *
*    MATB    MATERIAL TO BE PROCESSED                             *
*            MATB LT 0 IS A FLAG TO ADD MTS TO AND/OR REPLACE     *
*            INDIVIDUAL MTS ON NGOUT1.                            *
*    IGN     NEUTRON GROUP STRUCTURE OPTION                       *
*    IGG     GAMMA GROUP STRUCTURE OPTION                         *
*    IWT     WEIGHT FUNCTION OPTION                               *
*    LORD    LEGENDRE ORDER                                       *
*    NTEMP   NUMBER OF TEMPERATURES                               *
*    NSIGZ   NUMBER OF SIGMA ZEROES                               *
*    IPRINT  LONG PRINT OPTION (0/1=MINIMUM/MAXIMUM)              *
*            (DEFAULT=1)                                          *
* CARD3                                                           *
*    TITLE   RUN LABEL (UP TO 80 CHARACTERS DELIMITED BY *,       *
*            ENDED WITH /)  (DEFAULT=BLANK)                       *
* CARD4                                                           *
*    TEMP    TEMPERATURES IN KELVIN                               *
* CARD5                                                           *
*    SIGZ    SIGMA ZERO VALUES (INCLUDING INFINITY)               *
*                                                                 *
*          IF IGN=1, READ NEUTRON GROUP STRUCTURE (6A AND 6B)     *
* CARD6A                                                          *
*    NGN     NUMBER OF GROUPS                                     *
* CARD6B                                                          *
*    EGN     NGN+1 GROUP BREAKS (EV)                              *
*                                                                 *
*          IF IGG=1, READ GAMMA GROUP STRUCTURE (7A AND 7B)       *
* CARD7A                                                          *
*    NGG     NUMBER OF GROUPS                                     *
* CARD7B                                                          *
*    EGG     NGG+1 GROUP BREAKS (EV)                              *
*                                                                 *
*          WEIGHT FUNCTION OPTIONS (8A,8B,8C,8D)                  *
* CARD8A     FLUX CALCULATOR PARAMETERS (IWT.LT.0 ONLY)           *
*    EHI     BREAK BETWEEN COMPUTED FLUX AND BONDARENKO FLUX      *
*                (MUST BE IN RESOLVED RANGE)                      *
*    SIGPOT  ESTIMATE OF POTENTIAL SCATTERING CROSS SECTION       *
*    NFLMAX  MAXIMUM NUMBER OF COMPUTED FLUX POINTS               *
```

```
*    NINWT   TAPE UNIT FOR NEW FLUX PARAMETERS (DEFAULT=0)       *
*    JSIGZ   INDEX OF REFERENCE SIGMA ZERO IN SIGZ ARRAY         *
*            (DEFAULT=0)                                         *
*    ALPHA2   ALPHA FOR ADMIXED MODERATOR (DEF=0=NONE)           *
*    SAM      ADMIXED MODERATOR XSEC IN BARNS PER ABSORBER       *
*             ATOM (DEF=0=NONE)                                  *
*    BETA     HETEROGENEITY PARAMETER (DEF=0=NONE)               *
*    ALPHA3   ALPHA FOR EXTERNAL MODERATOR (DEF=0=NONE)          *
*    GAMMA    FRACTION OF ADMIXED MODERATOR CROSS SECTION IN     *
*              EXTERNAL MODERATOR CROSS SECTION (DEF=0)          *
* CARD8B     TABULATED (IWT=1 OR -1 ONLY)                        *
*    WGHT    READ WEIGHT FUNCTION AS TAB1 RECORD.                *
*            END WITH A /.                                       *
* CARD8C     ANALYTIC FLUX PARAMETERS (IWT=4 OR -4 ONLY)         *
*    EB      THERMAL BREAK (EV)                                  *
*    TB      THERMAL TEMPERATURE (EV)                            *
*    EC      FISSION BREAK (EV)                                  *
*    TC      FISSION TEMPERATURE (EV)                            *
* CARD8D     INPUT RESONANCE FLUX (IWT=0 ONLY)                   *
*    NINWT   TAPE UNIT FOR FLUX PARAMETERS                       *
*                                                                *
* CARD9                                                          *
*    MFD     FILE TO BE PROCESSED                                *
*    MTD     SECTION TO BE PROCESSED                             *
*    MTNAME  DESCRIPTION OF SECTION TO BE PROCESSED              *
*          REPEAT FOR ALL REACTIONS DESIRED                      *
*          MFD=0/ TERMINATES THIS TEMPERATURE/MATERIAL.          *
* CARD10                                                         *
*    MATD    NEXT MAT NUMBER TO BE PROCESSED                     *
*            MATD=0/ TERMINATES GROUPR RUN.                      *
*                                                                *
*---OPTIONS FOR INPUT VARIABLES-----------------------------------*
*                                                                *
*     IGN          MEANING                                       *
*     ---          -------                                       *
*      1           ARBITRARY STRUCTURE (READ IN)                 *
*      2           CSEWG 239-GROUP STRUCTURE                     *
*      3           LANL 30-GROUP STRUCTURE                       *
*      4           ANL 27-GROUP STRUCTURE                        *
*      5           RRD 50-GROUP STRUCTURE                        *
*      6           GAM-I 68-GROUP STRUCTURE                      *
*      7           GAM-II 100-GROUP STRUCTURE                    *
*      8           LASER-THERMOS 35-GROUP STRUCTURE              *
*      9           EPRI-CPM 69-GROUP STRUCTURE                   *
*     10           LANL 187-GROUP STRUCTURE                      *
*     11           LANL 70-GROUP STRUCTURE                       *
*     12           SAND-II 620-GROUP STRUCTURE                   *
*     13           LANL 80-GROUP STRUCTURE                       *
*     14           EURLIB 100-GROUP STRUCTURE                    *
*     15           SAND-IIA 640-GROUP STRUCTURE                  *
*     16           VITAMIN-E 174-GROUP STRUCTURE                 *
*     17           VITAMIN-J 175-GROUP STRUCTURE                 *
*                                                                *
```

```
*     IGG          MEANING                                          *
*     ---          -------                                          *
*      0           NONE                                             *
*      1           ARBITRARY STRUCTURE (READ IN)                    *
*      2           CSEWG 94-GROUP STRUCTURE                         *
*      3           LANL 12-GROUP STRUCTURE                          *
*      4           STEINER 21-GROUP GAMMA-RAY STRUCTURE             *
*      5           STRAKER 22-GROUP STRUCTURE                       *
*      6           LANL 48-GROUP STRUCTURE                          *
*      7           LANL 24-GROUP STRUCTURE                          *
*      8           VITAMIN-C 36-GROUP STRUCTURE                     *
*      9           VITAMIN-E 38-GROUP STRUCTURE                     *
*                                                                   *
*     IWT          MEANING                                          *
*     ---          -------                                          *
*      1           READ IN SMOOTH WEIGHT FUNCTION                   *
*      2           CONSTANT                                         *
*      3           1/E                                              *
*      4           1/E + FISSION SPECTRUM + THERMAL MAXWELLIAN      *
*      5           EPRI-CELL LWR                                    *
*      6           (THERMAL) -- (1/E) -- (FISSION + FUSION)         *
*      7           FAST REACTOR                                     *
*      8           THERMAL--1/E--FAST REACTOR--FISSION + FUSION     *
*      9           CLAW WEIGHT FUNCTION                             *
*     10           CLAW WITH T-DEPENDENT THERMAL PART               *
*     11           VITAMIN-E WEIGHT FUNCTION (ORNL-5505)            *
*     -N           COMPUTE FLUX WITH WEIGHT N                       *
*      0           READ IN RESONANCE FLUX FROM NINWT                *
*                                                                   *
*     MFD          MEANING                                          *
*     ---          -------                                          *
*      3           CROSS SECTION OR YIELD VECTOR                    *
*      5           FISSION CHI BY SHORT-CUT METHOD                  *
*      6           NEUTRON-NEUTRON MATRIX (MF4/5)                   *
*      8           NEUTRON-NEUTRON MATRIX (MF6)                     *
*     12           PHOTON PROD. XSEC (PHOTON YIELDS GIVEN, MF12)    *
*     13           PHOTON PROD. XSEC (PHOTON XSECS GIVEN, MF13)     *
*     16           NEUTRON-GAMMA MATRIX (PHOTON YIELDS GIVEN)       *
*     17           NEUTRON-GAMMA MATRIX (PHOTON XSECS GIVEN)        *
*     18           NEUTRON-GAMMA MATRIX (MF6)                       *
*        NOTE: IF NECESSARY, MFD=13 WILL AUTOMATICALLY CHANGE       *
*        TO 12 AND MFD=16 WILL AUTOMATICALLY CHANGE TO 17 OR 18.    *
*     21           PROTON PRODUCTION MATRIX (MF6)                   *
*     22           DEUTERON PRODUCTION (MF6)                        *
*     23           TRITON PRODUCTION (MF6)                          *
*     24           HE-3 PRODUCTION (MF6)                            *
*     25           ALPHA PRODUCTION (MF6)                           *
*     26           RESIDUAL NUCLEUS (A>4) PRODUCTION (MF6)          *
*     31           PROTON PRODUCTION MATRIX (MF4)                   *
*     32           DEUTERON PRODUCTION (MF4)                        *
*     33           TRITON PRODUCTION (MF4)                          *
*     34           HE-3 PRODUCTION (MF4)                            *
*     35           ALPHA PRODUCTION (MF4)                           *
```

```
*     36           RESIDUAL NUCLEUS (A>4) PRODUCTION (MF4)       *
*          NOTE: IF NECESSARY, MFD=21-26 WILL                    *
*          AUTOMATICALLY CHANGE TO 31-36.                        *
*     90+N         RADIOACTIVE NUCLIDE PRODUCTION FROM NTH       *
*                   SUBSECTION FOR MTD IN FILE 9.                *
*    100+N         RADIOACTIVE NUCLIDE PRODUCTION FROM NTH       *
*                   SUBSECTION FOR MTD IN FILE 10.               *
*                                                                *
*     MTD          MEANING                                       *
*     ---          -------                                       *
*     -N           PROCESS ALL MT NUMBERS FROM THE PREVIOUS      *
*                          ENTRY TO N INCLUSIVE                  *
*     221-250      RESERVED FOR THERMAL SCATTERING               *
*     258          AVERAGE LETHARGY                              *
*     259          AVERAGE INVERSE VELOCITY (M/SEC)              *
*                                                                *
*     AUTOMATIC REACTION PROCESSING OPTIONS                      *
*     ------------------------------------                       *
*       3/         DO ALL REACTIONS IN FILE3 ON INPUT PENDF      *
*       6/         DO ALL MATRIX REACTIONS IN ENDF DICTIONARY    *
*      13/         DO ALL PHOTON PRODUCTION CROSS SECTIONS       *
*      16/         DO ALL PHOTON PRODUCTION MATRICES             *
*      21/         DO ALL PROTON PRODUCTION MATRICES             *
*      22/         DO ALL DEUTERON PRODUCTION MATRICES           *
*      23/         DO ALL TRITON PRODUCTION MATRICES             *
*      24/         DO ALL HE-3 PRODUCTION MATRICES               *
*      25/         DO ALL ALPHA PRODUCTION MATRICES              *
*      26/         DO ALL A>4 PRODUCTION MATRICES                *
*                                                                *
******************************************************************
```

In these instructions, CARD1 defines the input and output units for GROUPR. The module requires both ENDF and PENDF input tapes, because the PENDF tapes produced by RECONR, BROADR, etc., do not contain angle (MF=4), energy (MF=5), or photon (MF=12,15) distributions. For materials that do not use resonance parameters to represent part of the cross section, it is possible to use a copy of the ENDF tape in place of the PENDF tape. The normal mode for GROUPR is to use NGOUT1=0; however, sometimes it is convenient to add a new material or reaction to an existing GENDF tape. The old GENDF tape is then mounted on unit NGOUT1, and the revised GENDF tape will be written to NGOUT2.

Card 2 selects the first material to be processed (MATB) and sets up the group structures, weighting option, Legendre order, and self-shielding parameters for all the materials to be processed in this run.

The names of the available group structures are given in the input instructions, and the energy bounds are tabulated in Appendix A of this report. Of course, it is always possible to read in an arbitrary group structure (see CARD6A through

CARD7B). The energies must be given in increasing order (note that this is opposite from the usual convention). Here is an example of the input cards for the conventional 4-group structure used in many thermal reactor codes:

```
4/ CARD6A
1E-5 .625 5530 .821E6 10E6 / CARD6B  .
```

These cards are read by the standard NJOY free-format input routine (FREE). Fields are delimited by space, and "/" terminates the processing of input on a card. Anything after the slash is a comment.

The available weight function options are listed in the input instructions under IWT, and they are plotted in Appendix B of this report. Just as in the case of group structures, an arbitrary weight function can be read in (see CARD8B). This function is presented to GROUPR as an ENDF/B "TAB1" record. This means that a count of $E, C(E)$ pairs and one or more interpolation schemes are given. For the case of a single interpolation scheme, the input cards would be

```
0. 0. 0 0 1 N N INT
E1 C(E1) E2 C(E2) ... / CARD8B,
```

where N is the count of E, C pairs, and INT specifies the interpolation law to be used for values of E between the grid values E_1, E_2,... according to the following "graph paper" schemes:

INT	Meaning
1	histogram
2	linear-linear
3	linear-log
4	log-linear
5	log-log

For example, I=5 specifies that $\ln C$ is a linear function of $\ln E$ in each panel $E_i \leq E < E_j$. Similarly, I=4 specifies that $\ln C$ is a linear function of E in each panel. As many physical lines as are needed can be used for "CARD8B", as long as the terminating slash is included.

One of the weighting options is a generalized "$1/E$+fission+thermal" function where the thermal temperature, fission temperature, and breakpoint energies (all in eV) are given on CARD8C. The weight function for the Los Alamos LIB-IV cross section library[17] used

```
.10 .025 820.3E3 1.4E6 / CARD8C  .
```

A plot of this function will be found in Appendix B.

The GROUPR flux calculator is selected by a negative sign on **IWT**. The additional **CARD8A** is then read. The calculator option used is determined by the number of parameters given and their values. The parameters **EHI** and **NFLMAX** are used to select the energy range for the flux calculation, and they also determine the cost in time and storage. The actual value for **SIGPOT** is not very critical—a number near 10 barns is typical for fissionable materials.

Nonzero values for **NINWT** and **JSIGZ** will cause the computed flux for a given fissionable isotope (such as ^{238}U) to be written out onto a file. This saved flux can be used as input for a subsequent run for a fissile material (such as ^{239}Pu) with **IWT**=0 to get an approximate correction for resonance-resonance interference. See Eq. (39).

Nonzero values for some of the last five parameters on **CARD8A** select the extended flux calculation of Eq. (37). The simplest such calculation is for an isolated pin containing a heavy absorber with an admixed moderator. For $^{238}UO_2$, the card might be

```
400 10.6 5000 0 0 .7768 7.5 / CARD8A  ,
```

where 7.5 barns is twice the oxygen cross section and α is computed from $[(A-1)/(A+1)]^2$ with $A = 15.858$. A more general case would be a PWR-like lattice of $^{238}UO_2$ fuel rods in water:

```
400 10.6 5000 0 0 .7768 7.5 .40 1.7E-7 .086 / CARD8A  ,
```

where .086 is computed using 3.75 barns for O and 40 barns for the two H atoms bound in water; that is,

$$\gamma = \frac{3.75}{2 * 20 + 3.75} = .086 . \tag{129}$$

A third example would compute the flux for a homogeneous mixture of ^{238}U and hydrogen

```
400 10.6 5000 0 0 0 0 1. 1.7E-7 / CARD8A  .
```

As a final example, consider a homogeneous mixture of uranium and water. This requires **BETA**=1 and **SAM**=0. Thus,

```
400 10.6 5000 0 0 .7768 0. 1. 1.7E-7 .086 / CARD8A  .
```

The maximum Legendre expansion order used for scattering matrices is set by LORD. The number of tables produced is LORD+1; that is, ℓ = 0, 1, ... LORD. When more than 1 value of σ_0 is requested, both the ℓ=0 and ℓ=1 components of the total cross section are produced.

Card 3 contains a short descriptive title that is printed on the listing and added to the output GENDF tape. Card 4 gives the NTEMP values of temperature for the run. They must be in ascending order, and if unresolved data are included on the PENDF tape, the temperatures in this list must match the first NTEMP values in MF=2, MT=153 from UNRESR (see STOUNR and GETUNR). Card 5 gives the σ_0 values for the run in descending order, starting with infinity (represented by 10^{10} barns).

This completes the description of the global input parameters for GROUPR. The rest of the input cards request reactions to be processed for the various temperatures and materials desired. Because of the many types of data that it can process, GROUPR does not have a completely automatic mode for choosing reactions to be processed. On the basic level, it asks the user to request each separate cross section or group-to-group matrix using the parameters MFD, MTD, and MTNAME. However, simplified input modes are also available. For example, the one "CARD9" containing

```
3/
```

will process the cross section "vectors" for all of the reaction MT numbers found on the PENDF tape.

For completeness, the full input for MATD, MFD, and MTNAME will be described first. Most readers can skip to the description of automated processing below. The value of MFD depends on the output desired (vector, matrix) and the form of the data on the ENDF evaluation. Simple cross section "vectors" σ_{xg} are requested using MFD=3 and the MTD numbers desired from the list of reactions available in the evaluation (check the directory in MF=1, MT=451 of the ENDF and PENDF tapes for the reactions available). A typical example would be

```
3   1 *TOTAL*/
3   2 *ELASTIC*/
3  16 *(N,2N)*/
3  51 *(N,NPRIME)FIRST*/
3 -66 *(N,NPRIME)NEXT*/
3  91 *(N,NPRIME)CONTINUUM*/
3 102 *RADIATIVE CAPTURE*/  .
```

The combinations of "3 51" followed by "3 −66" means process all the reactions from 51 through 66; that is, (n,n$'_1$), (n,n$'_2$), ..., (n,n$'_{16}$). If self-shielding is re-

quested, the following reactions will be processed using **NSIGZ** values of background cross section: total (MT=1), elastic (MT=2), fission (MT=18 and 19), radiative capture (MT=102), heat production (MT=301), kinematic KERMA (MT=443), and damage energy production (MT=444). The other File 3 reactions will be computed at $\sigma_0=\infty$ only. This list of reactions can be altered by small changes in **INIT** if desired.

There are several special options for **MTD** available when processing cross section vectors:

MTD	Option
259	Average inverse neutron velocity for group in s/m.
258	Average lethargy for group.
251	Average elastic scattering cosine $\overline{\mu}$ computed from File 4.
252	Continuous-slowing-down parameter $\overline{\xi}$ (average logarithmic energy decrement for elastic scattering) computed from File 4.
253	Continuous-slowing-down parameter $\overline{\gamma}$ (the average of the square of the logarithmic energy decrement for elastic scattering, divided by twice the average logarithmic energy decrement for elastic scattering) computed from File 4.
452	$\overline{\nu}$: the average total fission yield computed from MF=1 and MF=3.
455	$\overline{\nu}^D$: the average delayed neutron yield computed from MF=1 and MF=3.
456	$\overline{\nu}^P$: the average prompt fission neutron yield computed from MF=1 and MF=3.

There are also some special options for **MFD** that can be used when processing cross sections:

MFD	Option
12	Photon production cross section computed from File 12 and File 3.
13	Photon production cross section computed from File 13. Recent versions of GROUPR will automatically shift between 12 and 13, if necessary.
91–99	Isomer production cross sections computed from yields in File 9. **MFD**=91 is for the first state given in the file (usually the ground state), **MFD**=92 is for the second state given (usually the first isomeric state), etc.
101–109	Isomer production cross sections as above, except for data given as isomer production cross sections in File 10.

An example of the isomer production capability would be the radiative capture reaction of ENDF/B-V ^{109}Ag(n,γ) from Tape 532:

```
91 102 *(N,G) TO G.S.*/
92 102 *(N,G) TO ISOMER*/  .
```

The next class of reactions usually processed is the group-to-group neutron scattering matrices. The complete list of MTD values is most easily found under File 4 in the MF=1, MT=451 "dictionary" section of the evaluation. An example follows:

```
6   2 *ELASTIC MATRIX*/
6  16 *(N,2N) MATRIX*/
6  51 *(N,NPRIME)FIRST MATRIX*/
6 -66 *(N,NPRIME)NEXT MATRIX*/
6  91 *(N,NPRIME)CONTINUUM MATRIX*/  .
```

Using MFD=6 implies that File 4, or File 4 and File 5, will be used to generate the group-to-group matrix. The elastic matrix will be computed for NSIGZ values of background cross section, but the other reactions will be computed for σ_0=∞ only. The list of matrices to be self-shielded can be changed by making small changes to INIT if desired.

The case of fission is more complex. For the minor isotopes, only the total fission reaction is used, and the following input is appropriate for the prompt component:

```
3 18 *FISSION XSEC*/
6 18 *PROMPT FISSION MATRIX*/  .
```

For the important isotopes, partial fission reactions are given. They are really not needed for most fission reactor problems, and the input above is adequate. However, for problems where high-energy neutrons are important, the following input should be used:

```
3 18 *TOTAL FISSION*/
3 19 *(N,F)*/
3 20 *(N,NF)*/
3 21 *(N,2NF)*/
3 38 *(N,3NF)*/
6 19 *(N,F)*/
6 20 *(N,NF)*/
6 21 *(N,2NF)*/
6 38 *(N,3NF)*/  .
```

Note that "6 18" is omitted because it will, in general, be different from the sum of the partial matrices (see Section F). Some materials don't have data for (n,3nf); in

these cases, omit the two lines with MTD=38 from the input. The fission matrix is not self-shielded. Since resonance-to-resonance fission-spectrum variations are not described in ENDF/B, it is sufficient to self-shield the cross section and then to use the self-shielding factor for the cross section to self-shield the fission neutron production.

Delayed fission data are available for the important actinide isotopes, and the following input to GROUPR is used to process them:

```
3 455 *DELAYED NUBAR*/
5 455 *DELAYED SPECTRA*/  .
```

The line for MFD=5 automatically requests spectra for all six time groups of delayed neutrons. The six time constants are also extracted from the evaluation. As discussed in Section VIII.F, formatting modules such as DTFR and CCCCR must combine the prompt and delayed fission data written onto the GENDF tape in order to obtain steady-state fission parameters for use in transport codes.

Starting with the ENDF-6 format, neutron production data may also be found in File 6, and MFD=8 is used to tell the code to use MF6 for this MTD. When using full input, the user will have to check the File 1 directory and determine what subsections occur in File 6.

Photon production reactions can be found in the ENDF dictionary under MF=12 and 13. To request a neutron-to-photon matrix, add 4 to this number.* For example,

```
17   3 *NONELASTIC PHOTONS*/
16   4 *INELASTIC PHOTONS*/
16  18 *FISSION PHOTONS*/
16 102 *CAPTURE PHOTONS*/  .
```

Yields (MF=12) are normally used with resonance reactions (MT=18 or MT=102), or for low-lying inelastic levels (MT=51, 52, ...). MT=3 is often used by evaluators as a catch-all reaction at high energies where it is difficult to separate the source reactions in total photon emission measurements. In these cases, photon production cross sections from other reactions like MT=102 are normally set equal to zero at high energies. The general rule for photon emission is that the total production is equal to the sum of all the partial production reactions given in the evaluation. Starting with the ENDF-6 format, photon production may also appear in File 6.

*In recent versions of NJOY, GROUPR will automatically shift between 16 and 17 using data read from the ENDF dictionary by the CONVER subroutine. Thus, use of MFD=17 is no longer necessary.

Use MFD=18 to process these contributions. Since resonance-to-resonance variations in photon spectra are not given in ENDF/B, GROUPR does not normally self-shield the photon production matrices (although this can be done if desired by making a small change in INIT); instead, it is assumed that only the corresponding cross section needs to be shielded. Subsequent codes can use the cross section self-shielding factor with the infinite-dilution photon production matrix to obtain self-shielded photon production numbers.

This version of GROUPR can also generate group-to-group matrices for charged-particle production from neutron reactions and for all kinds of matrices for incident charged particles. The incident particle is determined by the input tape mounted. The identity of the secondary particle is chosen by using one of the following special MFD values:

For distributions given in File 6 (energy-angle):

MFD	Meaning
21	proton production
22	deuteron production
23	triton production
24	He-3 production
25	alpha production
26	residual nucleus (A>4) production

For distributions given in File 4 (angle only):

MFD	Meaning
31	proton production
32	deuteron production
33	triton production
34	He-3 production
35	alpha production
36	residual nucleus (A>4) production

If necessary, MFD=21–26 will automatically change to 31–36.

The user will normally process all reactions of interest at the first temperature (for example, 300 K). At higher temperatures, the threshold reactions should be omitted, because their cross sections do not change significantly with temperature except at the most extreme conditions. This means that only the following reactions should be included for the higher temperatures (if present): total (MT=1), elastic (MT=2), fission (MT=18), radiative capture (MT=102), heating (MT=301), kinematic KERMA (MT=443), damage (MT=444), and any thermal cross sections (MT=221–250). Only the elastic and thermal matrices should be included at the higher temperatures.

Warning: when using the explicit-input option, it is a fatal error to request a reaction that does not appear in the evaluation, cannot be computed from the evaluation, or was not added to the PENDF tape by a previous module. Reactions with thresholds above the upper boundary of the highest energy group will be skipped after printing a message on the output file.

Automated processing of essentially all reactions included in an ENDF/B evaluation is also available. As mentioned above, the single card

```
3/
```

will process all the reactions found in File 3 of the input PENDF tape. However, this list excludes thermal data (MT=221–250) and special options such as MTD=251–253, 258–259, and 452–456. If any of these reactions are needed, they should be given explicitly (see example below). Similarly, the single card

```
6/
```

will process the group-to-group matrices for all reactions appearing in File 4 of the ENDF/B tape, except for MT=103–107 and thermal scattering matrices (MT=221–250). If MT=18 and 19 are both present, only MT=19 will be processed into a fission matrix. For ENDF-6 evaluations, the "6/" option will also process every neutron-producing subsection in File 6. Photon production cross sections are requested using

```
13/
```

and photon-production matrices are requested with the single card

```
16/    .
```

In both cases, all reactions in both File 12 and File 13 will be processed without the need for using MFD=12 or MFD=17. For ENDF-6 libraries, this option will also process all photon-production subsections in File 6. There is no automatic option for delayed neutron data. An example of a processing run for a fissionable isotope with thermal cross sections follows:

```
3/
3 221/ THERMAL XSEC
3 229/ AV.INVERSE VELOCITY
3 455/ DELAYED NUBAR
5 455/ DELAYED SPECTRA
6/
6 221/ THERMAL MATRIX
16/ PHOTON PRODUCTION MATRIX    .
```

An example of charged-particle processing for the incident-neutron part of a coupled n-p-γ library follows:

```
3/ CROSS SECTIONS
6/ NEUTRON PRODUCTION MATRIX
16/ PHOTON PRODUCTION MATRIX
21/ PROTON PRODUCTION MATRIX    .
```

The layout of data in a n-p-γ coupled set is shown in Figure 8.

R. Coding Details

GROUPR begins by reading most of the user's input (see RUINB). It then locates the desired material and temperature on the input ENDF, PENDF, and GENDF tapes, and reads in the self-shielded unresolved cross sections (if any) from the PENDF tape using STOUNR. If self-shielding was requested, GENFLX is used to compute the weighting flux as described in Section VIII.D. The next step is to write the header record for this material on the output GENDF tape.

The code is now ready to begin the loop over reactions for this material and temperature. Either an input card is read to get MFD, MTD, and reaction name, or the next reaction in an automatic sequence is selected. First, the default Legendre order, secondary group count, and σ_0 count are selected for the reaction in INIT, and the retrieval routines are initialized. GROUPR then processes the reaction using the PANEL logic described in Section J. If a "shortcut" fission spectrum was requested (MFD=5), for delayed fission, and for the low-energy "constant" spectra, the spectrum is calculated directly using GETFF. As the cross sections for each group are obtained, they are printed out (see DISPLA) and written to the GENDF tape. When the last group has been processed, GROUPR loops back to read a new input card for a new reaction.

This loop over reactions continues until a terminating "0/" card is read. GROUPR then proceeds to the next temperature, if any, and repeats the loop over reactions. After the last temperature has been processed for the first material, an opportunity is provided to change to a new material, keeping all the other input parameters unchanged. A "0/" card at this point causes all the files to be closed, prints out the final messages, and terminates the GROUPR run.

Automatic choice of the next reaction to be processed is done in one of two ways. For a simple range of MT numbers, such as the example 51–66 used above, the negative value is stored in the variable MTDP in the GROUPR subroutine. When MTDP is negative, MTD is incremented after each reaction until it is greater than the absolute value of MTDP. MTDP is then reset to one, and the code proceeds to the

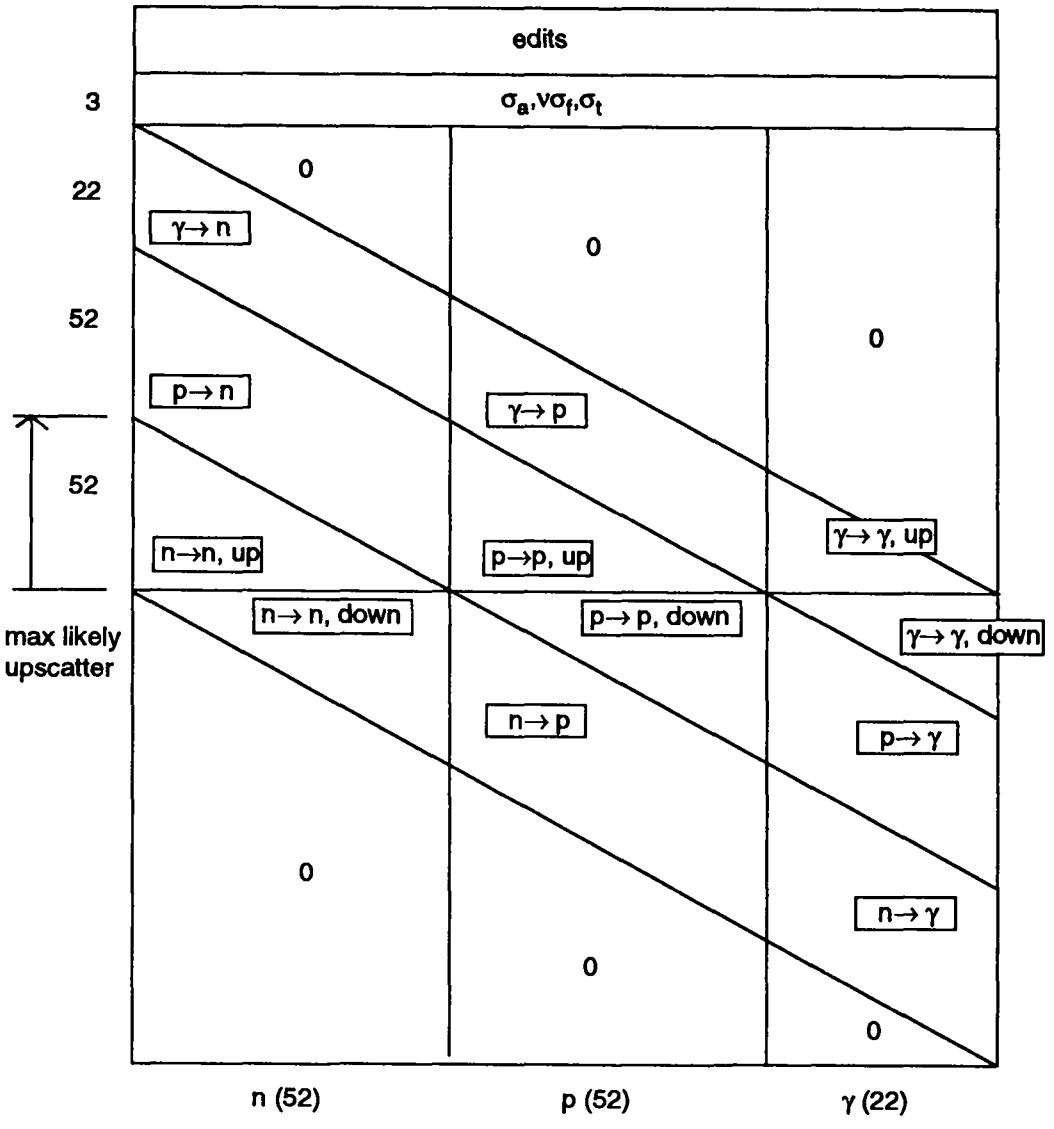


Figure 8: Layout of a coupled table for the simultaneous transport of neutrons, protons, and gamma rays. Normally, only the lower triangle of the "p to n" block would contain values for the upscatter portion of the table (the part above the line in the middle). The "group" index increases from left to right, and the "position" index increases from top to bottom.

next input card. However, MFD=16 will automatically change to 17, if necessary. Similarly, MFD=21, 22, ... will automatically change to MFD=31, 32, ... A more automatic method is triggered by MTD=0. In this case, a subroutine called NEXTR is called to return the next value of MTD to be used, and a subroutine called NAMER is called to generate the reaction name. For MFD=3, NEXTR finds the next reaction in File 3 on the input PENDF tape. MT=251–253 and thermal data (MT=221–250) are excluded. The MT values for the special options (258, 259, etc.) do not appear on the PENDF tape, and they must be requested explicitly if needed. For matrices, GROUPR works with a set of lists loaded into common block /RLIST/ by CONVER. The list MF4 contains all the neutron-scattering MT numbers that appear in the File 4 part of the directory on the ENDF tape, and the list MF6 contains all the MT numbers of sections of File 6 that contain subsections that produce neutrons. Therefore, reading through these two lists returns all the neutron-producing matrix reactions. Similarly, the list MF12 contains all the File 12 entries from the directory, MF13 contains the File 13 entries, and MF18 contains all the MT values for sections in File 6 that contain subsections for photon production. Scanning through these three lists produces all the photon production matrix reactions. Two arrays are used for charged-particle producing reactions; the first index runs through the charged particles in the order p, d, t, ^{3}He, α, recoil. Taking proton production as an example, the list elements MF6P(1,I) contain the MT numbers of sections in File 6 that contain subsections that produce protons. The list elements MF4R(1,I) contain MT numbers from File 4 for two-body reactions that produce protons; namely, MT600–648. NEXTR scans through both of these lists to return indexes to all the reactions that produce protons. The same procedure is used for the other charged particles.

Subroutine NAMER generates name strings with up to 15 Hollerith words with 4 characters each (60 characters). The names depend on the "ZA" of the projectile and the MT number for the reaction. The parameter MFD is used to choose between the suffixes "cross section" and "matrix". Some examples of the names produced follow:

Name	Name	Name
(N,TOTAL)	(N,HEAT)	(P,P02)
(N,ELASTIC)	(N,P02)	(P,N00)
(N,2N)	(P,ELASTIC)	(G,TOTAL)
(N,N01)	(P,2N)	(G,PAIR)

Subroutines MFCHK and MFCHK2 are used with the full input for reaction selection in GROUPR to check whether MFD=17 is needed when MFD=16 was requested,

or whether one of the charged-particle File 4 values MFD=31-36 is needed when MFD=21-26 were requested. The lists in common block /RLIST/ are used, just as for NEXTR.

Subroutine GENGPN generates the group bounds in /GROUPN/ for the neutron group structure from input cards, from data statements, or by calculation. Some of the data statements use energies in eV and some use lethargy. GENGPG generates the photon group structure in /GROUPG/ from input cards or data statements; in this case, all bounds are in eV.

Subroutine GENWTF sets up the weight function option requested with IWT by reading the input cards into /WEIGHTS/, transferring numbers from data statements to /WEIGHTS/, or simply reporting the analytic weight option requested. GETWTF returns the values of the weight function at energy E by calculation or by interpolation in the table established by GENWTF. The current version returns the same value for all Legendre orders. Choosing ENEXT is difficult for GETWTF because the functions have not been explicitly linearized. It is important to generate extra grid points in energy regions where the weight function may vary faster than the cross section (for example, in the fusion peak).

Subroutine GENFLX computes the self-shielded weighting flux using either the Bondarenko model or the flux calculator, and it writes the result on a scratch tape using the LOADA routine from the NJOY level. When fluxes are needed for the generalized group integrals, they are read from this scratch file using FINDA (see GETFLX). GENFLX starts by checking for the weighting option. If the Bondarenko model was selected, it initializes GETY1 and GETWTF to read the total cross section from the PENDF tape and the smooth weighting function $C(E)$ set up by GENWTF. It then steps through the union grid of $\sigma_t(E)$ and $C(E)$ computing the flux vs. σ_0 and Legendre order by means of Eq. (14). In the unresolved energy range, GETUNR is used to retrieve the unresolved cross sections as a function of σ_0, and Eq. (22) is used to compute the weighting flux. In both cases, the data at each energy are stored as the 1+(LORD+1)*NSIGZ components

```
E, PHI(IL,IZ) ,
```

where IL runs from 1 to LORD+1 and IZ runs from 1 to NSIGZ.

If the flux calculator option was requested, GENFLX sets up the parameters for the calculation and requests needed storage space. Next, the cross section retrieval routines GETY1 and GETY2 are set up to return total and elastic cross sections from the PENDF tape. A lower energy limit (ELO) is chosen, and the cross sections are

read into storage until a maximum energy EHI or a maximum number of points NEMAX is reached.

The slowing-down equation, either Eq. (33) or Eq. (37), is then solved from the break energy down to ELO. The scattering source from energies above the break is based on the NR approximation. When the calculation is finished, fluxes from 10^{-5} eV to ELO are written to the scratch file using LOADA in the same format used for the Bondarenko option. From ELO to the energy break point, fluxes are transferred from memory to the LOADA file. Finally, above the break point, fluxes are computed and saved using the Bondarenko model.

Subroutine INIT is used to set up the number of σ_0 values, secondary energy groups, and Legendre components for each combination of MFD and MTD. If MFD=6, a special copy of File 6 is made for use in GETAED. The list of reactions to be self-shielded can be changed if desired. The number of secondary groups NG helps determine the storage required for the accumulating group integrals (see dynamic storage array ANS with pointer IANS in GROUPR). For simple cross section vectors, NG=2. The NZ*NL flux components are stored first, followed by the NZ*NL cross section components. When all the panels for one group have been processed, dividing position 2 by position 1 gives the group-averaged cross section. For ratio quantities like $\overline{\nu}$ and $\overline{\mu}$ (MTD=251–253, 452, 455, 456), NG is 3. Once again, the flux components are stored first, followed by the NL*NZ components of RATIO*SIGMA, followed by the components of the cross section. This arrangement allows for the calculation of group-averaged values of RATIO*SIGMA, RATIO, or cross section by dividing position 2 by 1, position 2 by 3, or position 3 by 1, respectively. For matrices, NG is set to one more than the number of secondary groups (NGN or NGG). The NL*NZ flux components are stored first, followed by the NL*NZ integrals for each secondary-energy group in turn.

Subroutine PANEL performs the generalized group integrals using the logic described in Section J. For most calls to PANEL, the lower point of each "panel" was computed as the upper point of the previous panel. Therefore, PANEL is careful to save these previous values. However, if the bottom of the panel is just above a discontinuity, new values of cross section and flux are retrieved. Once the values at the lower boundary of the panel are in place, new values for the top of the panel are retrieved (see FLUX, SIG). If the top of the panel is at a discontinuity in σ or at a group boundary, the energy used is just below the nominal top of the panel. "Just below" and "just above" are determined by RNDOFF and DELTA. For maximum accuracy, these numbers should be chosen such that RNDOFF>1, DELTA<1, and RNDOFF*DELTA<1 for the precision of the machine being used.

For simple average cross sections, the integrals of $\sigma \times \phi$ and ϕ are computed for the panel using trapezoids. This is justified by the linearization of σ; the value of $\sigma \times \phi$ at the midpoint is too uncertain to justify a more complex treatment. For two-body scattering, the feed function is far from linear over the panel. In fact, it can show oscillations as described in Section L. The integral of the triple product $\mathcal{F} \times \sigma \times \phi$ is obtained by Lobatto quadrature of order 6 or 10 using the quadrature points and weights given in the data statements (see QP6, QW6, QP10, QW10). The cross section and reaction rate are determined at each quadrature point by interpolation, and the feed function is obtained by GETFF. For many reactions, FF will be nonzero for only a certain range of secondary groups. IG1 is the index to the first nonzero result, and NG1 is the number of nonzero values of FF in the range. PANEL maintains the two corresponding values IGLO and NG to specify the nonzero range of values in array ANS. Finally, the flux and cross section at the top of the panel are transferred to FLST and SLST, and control is returned to the panel loop in GROUPR.

Subroutine DISPLA is used to print cross sections and group-to-group matrices on the output listing (NSYSO). Small values are removed for efficiency. Note that different formats are used in different circumstances. Infinitely dilute data are printed without σ_0 labels. Isotropic matrices are printed with several final groups on each line. Delayed neutron spectra are printed using the Legendre order variable for time groups and with the time constants given on a heading line.

Subroutine GETFLX returns NL by NZ components of the weighting flux. If NSIGZ is 1, the flux is computed using GETWTF, and all Legendre orders are taken to be equal. When self-shielding has been requested, the flux components are obtained by interpolating between adjacent values retrieved with FINDA from the scratch tape written by GENFLX. The grid energies found on the scratch tape are used to get ENEXT. The flux is taken to be continuous, so IDIS is always set to zero.

Subroutine GETYLD returns the yield needed by GETFF for fission (MT=452, 455, or 456 from File 1) or radionuclide production yields from File 9 (MFD=91–99). This routine also retrieves the delayed neutron time constants when MTD=455. Tabulated yields are obtained by interpolation using the utility routine TERPA. Polynomial data are expanded by direct computation in this subroutine.

Subroutine GETSIG returns NL*NZ components of the cross section using point data from the PENDF tape and self-shielded unresolved data, if present, from GETUNR. The routine starts by adjusting MF and MT for the special options (MTD=258–259, MFD=91–99, etc.) and locating the desired section on the PENDF tape. GETSIG is then called for each desired energy value E in increasing order.

For MTD=258 or 259, the appropriate velocity or lethargy is computed from E and returned. In the more general cases, GETY1 is used to retrieve the pointwise cross section from the PENDF tape, and GETUNR is called to replace this value with self-shielded unresolved cross sections if necessary.

The unresolved cross sections are handled using STOUNR and GETUNR. STOUNR locates the desired material and temperature in MF=2, MT=152 of the PENDF tape, and then it copies the data into the dynamic storage array with pointer IUNR. The σ_0 grid on the PENDF tape does not have to agree with the list requested for GROUPR, but a diagnostic message will be printed if they are different. However, the NTEMP values of temperature requested by GROUPR must agree with the first NTEMP temperatures on the PENDF tape, or a fatal error will result. GETUNR checks whether E is in the unresolved energy range and whether MT is one of the resonance reactions. If so, it locates the desired interpolation range in the UNR array in dynamic storage, and interpolates for the self-shielded cross sections. If this is an energy range where resolved and unresolved ranges overlap, the resolved part is added to the background σ_0 before interpolation. The subroutine TERPU is used for interpolating in the unresolved cross section tables. MT=261 is used to select the ℓ=1 component of the total cross section. Note that GETSIG also uses this ℓ=1 value for ℓ=2, 3, ...

Subroutine GETFF returns the feed function FF using different portions of the coding for different options. The first section is used for cross sections and ratio quantities. The same yield YLD is returned for every ℓ-component in FF.

The second section of GETFF is for neutron continuum transfer matrices. The yield is either determined from MT [for example, YLD=2 for MT=16, the (n,2n) reaction], or it is obtained using GETYLD for fission. Next, the angular distribution is obtained using GETFLE [see F in Eq. (89)], and the secondary energy distribution is obtained using GETSED [see g in Eq. (89)]. Finally, the product of the three factors is loaded in FF. Note that the range of groups returned extends from IGLO=1 to the highest nonzero result, for a total of NG groups. Since FF for these reactions is a smooth function of incident energy, NQ is set to zero, and no additional quadrature points will be used in PANEL.

The third section of GETFF is for gamma production matrices. The photon yields are obtained using GETGYL. In general, there are NYL different yields, each one corresponding to a different discrete gamma ray, or to the continuum. The angular distributions for these gamma rays are obtained using GETGFL (most ENDF/B photons are given as isotropic). The energy distribution for the continuum (if any) is obtained by GETSED. Now, the code loops through the photon group structure

placing each discrete photon in the appropriate group and adding in the continuum part. During this loop, the range of nonzero values (IGLO, IGHI) is determined. Finally, the nonzero values are packed into FF, and IGLO and NG are returned to describe the distribution. Once again, NQ=0 is used.

The fourth section of GETFF handles two-body scattering, either elastic or discrete-level inelastic, and both neutrons and charged particles. First, subroutine PARTS is called to set up the particle type, A' value, and the scattering law for reactions that use File 4. Then GETDIS is called to finish the processing.

The fifth section handles thermal-neutron scattering. The bulk of the work is done by GETAED, which is discussed below. GETFF takes the output of GETAED and packs it into the final result, FF.

The last two sections of GETFF are used to process energy-angle distributions from ENDF-6 format sections of File 6. The subroutine simply calls GETMF6.

Subroutine PARTS is used to set up particles for reactions that use File 4. Different branches are used for ENDF-6 and earlier ENDF versions because the MT numbers used for charged-particle discrete levels have been changed. For example, proton levels use MT=600–649 in ENDF-6 libraries, but MT=700–719 in earlier libraries. If MFD=3, no action is taken. For other MFD values, the routine determines ZAP, APRIME, and LAW. The result depends on whether File 4 or File 6 data are to be used.

Subroutine GETMF6 is used to compute the feed function for reactions represented in File 6. As is common with NJOY subroutines, it is called with ED=0 to initialize the subroutine. The first step is to locate the desired section of File 6 on the input ENDF tape. It then sets the parameter ZAD (for ZA desired) based on the input value MFD and searches through the section for the desired subsection. If this subsection has LAW=4, it defines a recoil particle; the code backs up to the first subsection in the section, which is assumed to be the subsection describing the emitted particle. The subroutine has now arrived at statement number 140, where it decides whether to branch to special coding for two-body reactions (statement number 500) or phase-space reactions (statement number 150). All the other options have a TAB2 record at this point that defines the incident energy grid. For LAW=1, the data for the first incident energy are read in and converted to the lab frame by CM2LAB. Similarly, for LAW=7, the data are read in and converted to a LAW=1 format by LL2LAB.

For a normal entry, GETMF6 interpolates for the particle yield for the current particle. For all of the laws except 2–5 (discrete two-body laws), it then checks to

see whether the energy ED is in the current panel (or if this is the first time in the first panel). If not, it moves the high data to the low position and reads in new high data. Of course, it also converts the new high data to the correct form with CM2LAB or LL2LAB. Once ED is in the current panel, the code reaches statement number 300, which sets up the loop over secondary energy by initializing F6LAB. This loop uses a grid that consists of the EPNEXT values returned by F6LAB and the group bounds EG(I). The actual integral over E'_L inside each group uses trapezoidal integration. The last step for laws 1, 6, and 7 is to add in the contributions to the feed function from discrete energies in File 6. For the discrete two-body laws (2–5), the routine goes through statement number 500, which simply calls GETDIS. For all laws, the last step is to check the normalization of the feed function. If the error is large, a message is printed out. In any case, the results are adjusted to preserve exact normalization.

Subroutine CM2LAB is used to convert a CM distribution starting at INOW in CNOW into a lab distribution, which will be stored starting at JNOW. The purpose of this routine is to generate a good grid for the lab secondary energy E'_L by adaptive reconstruction. The reconstruction stack is first primed with $p_{L\ell}(E,0)$ as computed by F6CM and $p_{L\ell}(E,E'_{\text{next}})$ (also from F6CM), where the value E'_{next} is the value EPNEXT returned by the first call to F6CM. This panel is then divided in half, and the value returned by F6CM is compared with the linear interpolate. If they agree within 1% (see TOL), this panel is converged. Otherwise, it is subdivided further. When convergence has been achieved for this panel, a new panel is chosen, and it is subdivided to convergence. When the entire range for E'_L has been processed, the final parameters are loaded into the new File 6 record for $p_{L\ell}(E,E'_L)$ (which starts at JNOW in CNOW), and the routine returns to GETMF6.

Subroutine F6CM is used to compute the Legendre coefficients of the double-differential scattering function $p_\ell(E\rightarrow E')$ in the lab system from data given in File 6 in the CM frame. The memory area CNOW contains the data for one particular incident energy E, and the routine must be called first with EP=0 for each new value of E. Thereafter, values of EP can be requested in any order. This is required by the adaptive scheme used to generate the lab E' grid in CM2LAB. The conversion uses Eqs. (122)–(127). On a normal entry with E'_L>0, the code computes the lower limit of the integral μ_{min}. It then sets up an adaptive integration over a panel starting at μ=1. The value for E'_C is computed using

$$E'_C = (1 + c^2 - 2c\mu_L)E'_L \ , \tag{130}$$

which is based on Eq. (123), and either F6DDX or F6PSP is called to compute

$p_C(E, E'_C, \mu_C)$ and EPNEXT. The lower limit of the integration panel is then computed by setting E'_C equal to EPNEXT and solving for the CM cosine using

$$\mu_C = \frac{\mu_L - c}{\sqrt{1 + c^2 - 2c\mu_L}} . \tag{131}$$

Of course, the larger between this result and $\mu_{\min}$ is used as the lower bound of the panel. Therefore, the adaptive integration operates on a nice continuous function. It continues subdividing the μ_C grid in this panel until convergence is achieved within 0.5% for all Legendre orders (see TOL/.005/). The lower bound of this panel becomes the upper bound of a new panel, and a new lower bound is selected as above. The integration is carried out over successive panels until $\mu_{\min}$ is reached. When the routine reaches statement number 310, the calculation of the continuum part of $p_{L\ell}$ is finished for these values of E and E'_L. It checks for possible contributions from delta functions. Warning: this section of the code is not finished. Next, the routine scans through the computed coefficients and zeros out any small ones that may just be noise. Finally, it chooses a value for EPNEXT and returns.

Function F6DDX is used to compute the double-differential scattering function $f(E \rightarrow E', \omega)$, where secondary energy E' and scattering cosine ω are in the CM system. On entry, CNOW contains the File 6 data for a particular value of E, and F6DDX must be called once with EP=0 for each new value of E. Thereafter, EP values can be requested in any order (this is required by the adaptive scheme used to convert to the lab system in F6CM). For a normal entry with EP>0, the code searches for the panel in the data that contain the requested value of E'. If LANG=1 for this subsection of File 6, the data are already given as Legendre coefficients, and the code simply interpolates for the desired results. If LANG=2, the data use the Kalbach-Mann scheme for representing the energy-angle distribution. This routine includes both the original Kalbach-Mann representation[13] and the newer Kalbach representation.[14] It has been set to use the latter by the statement DATA K86/1/. The code interpolates for the model parameters at E' and computes the desired answer with the model's formulas. The ENDF-6 format also allows CM distributions given as values tabulated versus scattering cosine (see LANG=11–15). This capability is not included in F6DDX; a fatal error message is issued if this option is encountered. Note that there is a "stub" at statement number 200 to take special action at low energies. It is currently disabled by the statement EFIRST=0., but it may be used sometime in the future to account for the fact that the low-energy dependence of the scattering function must vary as $f(E') = c\sqrt{E'}$.

Subroutine SEPE computes the neutron separation energy for the target IZAT using the liquid-drop model without pairing and shell terms. This quantity is required for the Kalbach-systematics approach to calculating the energy-angle distributions for particle emission with File 6. The formula used describes the reaction $a + B \rightarrow C + d$:

$$\begin{aligned} E_{sep} &= 15.68(A_C - A_B) \\ &- 28.07\left(\frac{(N_C - Z_C)^2}{A_C} - \frac{(N_B - Z_B)^2}{A_B}\right) \\ &- 18.56(A_C^{2/3} - A_B^{2/3}) \\ &+ 33.22\left(\frac{(N_C - Z_C)^2}{A_C^{4/3}} - \frac{(N_B - Z_B)^2}{A_B^{4/3}}\right) \\ &- 0.717\left(\frac{Z_C^2}{A_C^{1/3}} - \frac{Z_B^2}{A_B^{1/3}}\right) \\ &+ 1.211\left(\frac{Z_C^2}{A_C} - \frac{Z_B^2}{A_B}\right) \\ &- 0.0\,, \end{aligned} \tag{132}$$

where A stands for atomic weight, Z for charge number, and N for neutron number. Note that, even for reactions like (n,2n), C is the residual nucleus resulting from the removal of one particle d; it is not necessarily the real physical residual nucleus for the reaction. If the target IZAT is an element, E_{sep} has to be computed for some dominant isotope in the element. Dominant isotopes are assigned in this subroutine for some materials that often appear as elements in ENDF/B evaluations; if the particular target required does not appear, a fatal error message is issued. The user will have to add a line to the subroutine for the material and reassemble the code.

Subroutine LL2LAB is used to change an energy-angle distribution in File 6 from Law-7 format to Law-1 format using the laboratory Legendre representation. Law 7 represents the double-differential scattering distribution $f(E \rightarrow E', \mu)$ by giving a series of tables of $f(E \rightarrow E')$ for series of μ values. On entry, the entire Law-7 section is stored in C starting at INOW. The code loops through a set of E' values chosen to be the union of all the E' grids for all the different μ values. For each point on this union grid, the code interpolates for all the corresponding $f(\mu)$ values, and it uses them to compute NL Legendre coefficients $f_\ell(E')$. After completing the calculation for a new value of E', it checks the normalization of the result, and then it checks back to see whether the previous value is still needed to represent the curves $f_\ell(E')$ within a tolerance of 0.5% (see DATA TOL/.005/). The results are written in C starting at JNOW. They have the Law-1 Legendre form, namely,

sets of values E', $f_0(E')$, $f_1(E')$, ..., $f_{NL-1}(E')$, given for a series of E' values starting with zero.

Subroutine **F6LAB** is used to return the Legendre coefficients of the double-differential scattering function $f_\ell(E \rightarrow E')$ for particular values of E and E' in the lab system (see **E** and **EP**) from a part of File 6 in Law-1 format. Since Law-7 sections have been converted to Law-1 format using **LL2LAB**, and since CM sections in Law-1 format have been converted to the lab frame using **CM2LAB**, the only two roles left for this subroutine are interpolation to the desired values of E and E' and preparation of Legendre coefficients for sections that use the File-6 variant with **LAW=1** and **LANG**=11–15 (laboratory distributions tabulated) vs. μ. Only the continuum portion of the distribution is processed here; any delta functions given in File 6 must be handled separately. The routine is initialized by calling it with **EP=0**. The data for the two incident energy values that bracket E are already present in **CL0** and **CHI**. At this point, the routine extracts various parameters from the **C** array and prepares the variables used to control interpolation. These variables are complicated, because this routine handles three different interpolation schemes: "cartesian", "unit base", and "corresponding point" (see the ENDF-6 manual[4] for more details). For a normal entry, the code searches the data in **CL0** and **CHI** for the intervals containing E'. If **LANG=1**, it performs a two-dimensional interpolation for the **NL** coefficients at E and E'. For **LANG>1**, it computes the Legendre coefficients from the tabulations in File 6, and then it does the two-dimensional interpolation. Finally, it computes **EPNEXT** and returns.

Subroutine **GETDIS** is used to compute the feed function for elastic or discrete inelastic scattering of neutrons or charged particles using either File 4 or File 6 data. First, the angular distribution is retrieved with **GETFLE**, and an appropriate quadrature order is selected using Eq. (80). Then, a group loop from low energy to high energy is used to compute the ω_1 and ω_2 limits of Eq. (78), and the range (ω_1, ω_2) is subdivided using the appropriate Gauss-Legendre quadrature points (see **QP4**, **QP8**, **QP12**, and **QP20**). The function $f(E, \omega)$ is computed at each of these quadrature points using Eq. (79) and the angular distribution previously returned by **GETFLE**. The laboratory cosine at the quadrature point, $\mu[\omega]$, is computed using Eq. (75). Finally, the integrand of Eq. (78) is multiplied by the appropriate quadrature weight (**QW4**, **QW8**, **QW12**, or **QW20**) and added into the accumulating integral. This process is continued until $\omega_2 = 1$. The nonzero values of **FF** and the parameters **IGL0** and **NG** are then complete for this value of E.

The next step is to determine **ENEXT** based on the next critical point as given by Eqs. (82)–(85). Special cases are used for elastic scattering to avoid numerical

problems. Note that the discontinuity flag IDISC is set for critical points. The NQ variable is also set to force PANEL to subdivide the initial-energy integration.

Subroutine GETFLE retrieves or computes the Legendre coefficients for the angular distribution of a reaction at incident energy E. When called with E=0, GETFLE requests storage for the raw data, reads in the File 4 information for the first two incident energies on the file, and then uses GETCO to retrieve the corresponding coefficients. On subsequent entries with E>0, GETFLE simply interpolates for the desired coefficients. When E exceeds the upper energy in storage, the values for the upper energy are moved to the lower positions, and new upper values are read and converted to coefficients. An isotropic distribution is returned if E is outside the range of the angular data from File 4.

Subroutine GETAED retrieves angle-energy data for thermal scattering reactions. For coherent elastic scattering, the routine reads through the cross section on the PENDF tape using GETY1 and locates the Bragg edges. On each subsequent call to GETAED, the Legendre components of the cross section are computed using Eq. (70). For incoherent elastic scattering, the routine is first initialized by reading in the raw data for the first energy. On subsequent entries, a test is made to see whether E is in the range ELO to EHI. If not, the high data are moved to the low positions, and new high data are read. The Legendre components are then computed using Eq. (72). For incoherent inelastic scattering, GETAED is initialized by reading in the raw data for the first two incident energies. On subsequent entries, the subroutine checks to see whether E is between ELO and EHI. If not, the data for EHI are moved to the low positions, and new raw data are read from File 6 and binned. Once the correct data are in place, the desired energy-angle distribution is computed by using a combination of interpolation along lines of constant energy transfer and unit-base interpolation.

Subroutine GETGFL returns the Legendre coefficients for the angular distributions for all discrete and continuum photons for a reaction simultaneously. When called with ED=0, the routine reads File 14 into scratch storage and finds the starting location for the subsection describing each photon. On subsequent entries with ED>0, GETGFL sets up a loop over the NG photons on this section of File 14. For each photon, it searches for the energy panel that contains ED, uses GETCO to retrieve or compute the Legendre coefficients at the upper and lower File 14 energies, and interpolates for the desired coefficients at ED using TERP1. Since most ENDF/B photons are represented as isotropic, a special short-cut calculation is provided for that case. Isotropic results are also returned if ED is outside the range of the data in File 14.

Subroutine GETCO is used by both GETFLE and GETGFL to retrieve or compute Legendre coefficients from data in File 4 or File 14 format. The user can request output in either the the lab or CM system, and the raw data can be either Legendre coefficients or tabulated probability versus emission cosine. However, if the raw data are in the laboratory system, CM coefficients cannot be produced. If the raw data are already in the form of coefficients in the desired system, GETCO simply checks for the maximum Legendre order needed using a tolerance of TOLER=1E-6 and returns the coefficients in FL and the order in NL. If coordinate conversion is required, or if the raw data are tabulated, GETCO sets up the integral over cosine using Gauss-Legendre quadrature of order 20 (see QP and QW). The scattering probability for the quadrature point is computed from the coefficients or obtained by interpolating in the tabulation. The Legendre polynomials in the desired reference system are then computed. If the raw data are in the CM system (ω) and the result is to be in the lab system (μ), the desired polynomials are $P_\ell(\mu[\omega])$; otherwise, the quadrature angle is used directly to compute the polynomials. Once the coefficients have been computed, they are checked using TOLER to determine the maximum order required, NL, and the final results are returned in FL.

Subroutine GETGYL is used to retrieve the yields for all photons emitted in a specified reaction simultaneously. The raw data are obtained from the ENDF/B tape as either yields (MF=12) or production cross section (MF=13). In the latter case, GETGYL actually returns the fraction of the total yield assigned to each photon. The cross section returned by GETSIG is the total photon production cross section from MF=13 on the PENDF tape, which makes the resulting integral correct. Using the normal GROUPR procedure, GETGYL is initialized by calling it with ED=0. The entire File 12 or File 13 is read into scratch storage, and the starting location for each subsection is determined. On subsequent entries (ED>0), the routine loops over the NYL photons found, and uses TERPA to compute the yield at ED. If this is a primary photon, a discontinuity is set up at the energy where the photon will change groups. For MF=12, the calculation is finished. For MF=13, the numbers calculated above are converted to fractions of the total yield by dividing by the total production subsection from File 13. This routine does not handle ENDF/B transition probability arrays directly, because they will have been converted to File 12 yields by CONVER.

Subroutine CONVER converts ENDF/B evaluations to a standard form. If transition probability arrays were used in File 12, they are converted to yields and written back into File 12. If a section with MF=1 and MT=456 is missing from the evaluation, a copy of MT=452 is added to the tape as MT=456. In addition, a second copy of the modified tape is made on unit NSCR. While CONVER is reading

through the tape, lists of the reactions in File 4, File 12, and File 13 are written to /RLIST/ for use by the automatic reaction selection logic in NEXTR. For ENDF-6 evaluations containing File 6, the routine scans through File 6 looking for sections that produce neutrons, photons, or charged particles. The MT numbers for these sections are stored into MF6, MF18, and MF6P. The routine also checks for sections of File 4 containing charged-particle angular distributions and records their MT numbers in MF4R. Finally, if the section MF6/MT2 contains charged-particle elastic scattering information given using the nuclear-plus-interference format, it is converted into the residual-cross-section format for GETDIS.

Subroutine GETSED returns the secondary-energy distribution for neutrons or continuum photons for all groups simultaneously. Both tabulated and analytic functions are handled. GETSED is initialized for a particular reaction by calling it with ED=0. First, scratch storage is allocated, and all the subsections are read in. Analytic subsections are left in their raw form, but tabulated subsections are averaged over outgoing energy groups for each of the given incident energies. The array LOC contains pointers for each subsection. On subsequent entries (ED>0), GETSED loops over the subsections for this reaction. It first retrieves the fractional probability for the subsection using TERPA. If an analytic law is specified, ANASED is used to compute the group integral for each secondary-energy group. Each integral is multiplied by the fractional probability for the law and accumulated into SED. For tabulated data, the routine simply interpolates between the two values for the group integrals using TERP1, and accumulates them into SED. Note that various restrictions on the ordering of subsections and prohibition of multiple tabulated subsections needed for earlier versions of GROUPR are no longer required. Upscatter is not allowed in secondary-energy distributions except for fission or photon production. If found, it is put into the "in-group" ($g'=g$).

Subroutine ANASED is used to calculate the integral from E1 to E2 for one of the analytic laws [see Eqs. (92)–(121)]. The routine starts with statement functions for the reduced complementary error function, rerfc(x). The parameters are then retrieved from the raw data, and the code branches to the appropriate block of coding. The resulting integral is returned in G.

Subroutine HNAB is used to compute the special functions required for analytic law 11, the energy-dependent Watt spectrum. The method used is described in the BROADR chapter of this manual.

S. Error Messages

The fatal error messages and warning message from GROUPR are listed below, along with recommended actions to recover from the problem.

`ERROR IN GROUPR***UNABLE TO FIND MAT=---, T=---.`

Check for input error or wrong PENDF tape mounted.

`ERROR IN GROUPR***STORAGE EXCEEDED.`

Main container array is not big enough. See the variables `IAMAX=25000` in `GROUPR` and `A(25000)` in `/GSTORE/`.

`ERROR IN GROUPR***PHOTONS NOT ALLOWED WITH IGG=0.`

In order to produce photon data, a photon group structure must be requested. Check the input on card 2.

`ERROR IN GROUPR***ILLEGAL MFD.`

Check input `MFD`; legal values are 3, 5, 6, 8, 16, 17, 18, 21–26, 31–36, 91–99, 101–109.

`MESSAGE FROM GROUPR---AUTO FINDS NO REACTIONS FOR MF=--.`

An automatic reaction selection card of the form "MFD" was given in the input, but the ENDF and PENDF tapes do not contain any sections that would produce the desired cross sections or matrices.

`ERROR IN RUINB***NTEMP=0 NOT ALLOWED.`

There must be at least one temperature requested.

`ERROR IN RUINB***TOO MANY TEMPERATURES.`

The maximum number of temperatures is 10. See the variables `NTMAX` and `TEMP(10)` in `/TEMPR/`.

`ERROR IN RUINB***NSIGZ=0 NOT ALLOWED.`

There must be at least one background cross section. The first input value is automatically changed to 1E10.

`ERROR IN RUINB***TOO MANY SIGMA ZEROES.`

The maximum number of sigma-zero values is 10. See `NZMAX` and `SIGZ(10)` in `/SIGZER/`.

`ERROR IN GENGPN***TOO MANY GROUPS.`

The maximum number of neutron group boundaries is 641. See `NGMAX` and `EG(641)` in `/GROUPN/`.

`ERROR IN GENGPN***READ-IN GROUP STRUCTURE IS OUT OF ORDER.`

Group structures must be given in ascending energy order.

`ERROR IN GENGPN***ILLEGAL GROUP STRUCTURE.`

Check input; current legal values are 1 through 17.

ERROR IN GENGPG***TOO MANY GROUPS.

The maximum number of photon group boundaries is 150. See the variables NGMAX and EGG(150) in /GROUPG/.

ERROR IN GENGPG***ILLEGAL GROUP STRUCTURE.

Check input; current legal values are 1 through 9.

ERROR IN GENWTF***ILLEGAL WEIGHT FUNCTION REQUESTED.

Check input; current legal values are -11 through $+11$.

ERROR IN GENFLX***TOTAL NOT DEFINED OVER ENERGY RANGE.

A complete total cross section is needed for self-shielding. This means that "dosimetry" and "activation" tapes, which normally give only a few key reactions, must be processed using NSIGZ=1 (infinite dilution).

ERROR IN GETFWT***E OUTSIDE RANGE OF DATA IN NINWT.

A premature end-of-file was found on the input flux tape when using the IWT=0 option. Check to be sure the tape was the output from a legal flux calculator run.

ERROR IN GETFWT***REQUESTED E IS OUT OF ORDER.

The cause could be an improper input tape. Check as above.

ERROR IN GETFWT***TEMPERATURE NOT ON NINWT.

The IWT=0 calculation must be consistent with the previous flux calculator run. Check to see if the correct tape was mounted.

ERROR IN PANEL***ELO.GT.EHI.

This indicates some error in the energy grid for PANEL. It usually occurs if RNDOFF and DELTA are incorrect for your machine. Make sure that RNDOFF>1, that DELTA<1, and that the product RNDOFF*DELTA< 1 when evaluated on your machine (for example, 1.00000001 is not greater than unity on a 32-bit machine).

ERROR IN GETYLD***ILLEGAL LND.

The maximum number of time groups is 6. See NDN and DNTC(6) in /DELAYN/.

ERROR IN GETSIG***ILLEGAL MT.

Check input for MTD. Legal values are 1–150, 201–250, 251–253, 258–259, 300–450, 452, 455, and 456.

MESSAGE FROM STOUNR---NO UNRESOLVED SIGMA ZERO DATA....

This message probably means that UNRESR was never run for this material. Infinitely dilute values will be used.

MESSAGE FROM STOUNR---SIGMA ZERO GRIDS DO NOT MATCH....

The unresolved calculations will probably work best if the σ_0 grid in GROUPR matches the one in UNRESR. However, this is not necessary. GETUNR will interpolate to get values on the GROUPR grid from the UNRESR grid. A message is issued in case this isn't what the user really intended.

ERROR IN STOUNR***STORAGE EXCEEDED.

There is not enough storage for unresolved cross section data on PENDF tape. This means that there is no more room in the main container array. See IAMAX=25000 and A(25000) in /GSTORE/.

ERROR IN STOUNR***CANNOT FIND TEMP=---

The list of temperatures requested for the GROUPR run must agree with the first NTEMP temperatures on the PENDF tape. Check your BROADR and UNRESR runs.

ERROR IN GETFF***OPTION NOT AVAILABLE FOR THIS MT.

The GETFF routine branches to different blocks of coding for different combinations of MFD and MTD, but if no appropriate branch is found for the current values, this error message is issued. It probably indicates an error in the evaluation.

ERROR IN GETMF6***DESIRED PARTICLE NOT FOUND.

The outgoing particle for a group-to-group matrix is implied by the value of MFD (for example, protons for MFD=21). This message means that the section of File 6 requested with MFD and MTD does not contain a subsection that produces that particle. Check the user input. This message should not occur with automatic reaction selection.

ERROR IN GETMF6***ILLEGAL LAW.

The value of the LAW parameter is greater than 7. This implies an error in the evaluation.

ERROR IN GETMF6***STORAGE EXCEEDED.

Since the block of storage called SC was reserved with NC=-1, the only recourse is to increase the size of the main container array. See A(25000) and IAMAX=25000 in GROUPR.

MESSAGE FROM GETMF6---BAD GRIDS FOR CORRESPONDING POINT ....

Corresponding-point interpolation won't work correctly unless the two grids above and below the point of interest have the same number of points. This message means that there is an error in the evaluation.

ERROR IN CM2LAB***STORAGE EXCEEDED.

Once again, the only recourse is to increase the size of the main container array. See A(25000) and IAMAX=25000 in GROUPR.

ERROR IN F6DDX***ILLEGAL LANG.

The value of LANG for tabulated angular distributions must be in the range 11–15.

ERROR IN F6DDX***TABULATED ANGULAR DISTRIBUTION NOT CODED.

This option needs to be implemented in a future version.

ERROR IN SEPE***DOMINANT ISOTOPE NOT KNOWN.

The calculation of the neutron separation energy needed for the Kalbach model for particle energy-angle distributions needs a guess for the dominant isotope in an element. It will have to be added to the code. The same problem will occur with the parallel routines in HEATR and ACER.

ERROR IN LL2LAB***STORAGE EXCEEDED.

The size of the main container array will have to be increased. See A(25000) and IAMAX=25000 in GROUPR.

ERROR IN F6LAB***ILLEGAL LANG.

The value of LANG must be in the range 11–15 for tabulated angular distributions.

ERROR IN GETDIS***ILLEGAL NQP GT 20.

The highest quadrature order available is currently 20. See data statements for QP and QW.

ERROR IN GETFLE***DESIRED ENERGY ABOVE HIGHEST ENERGY GIVEN.

Should not occur for well-constructed ENDF files. Check the evaluation to be sure File 3 and File 4 are consistent.

MESSAGE FROM GETFLE---LAB DISTRIBUTION CHANGED TO CM.

Angular distributions for two-body reactions are supposed to be given in the CM frame by ENDF conventions. Some old evaluations for heavy materials violate this rule; changing to the CM frame has little effect on the answers.

ERROR IN GETAED***NOT CODED FOR ENDF-FORMATTED FILE 6.

This subroutine uses a special form of File 6 that is not consistent with the one described in ENDF–102.[3, 4]

ERROR IN GETAED***STORAGE EXCEEDED.

This message also indicates that the main container array is too small. See IAMAX=25000 in GROUPR and A(25000) in /GSTORE/.

MESSAGE FROM BINA---DISCRIMINANT=---//SET TO ABS...

The subroutine is having trouble finding the boundaries of equal-probability bins for the thermal interpolation method. If this negative value is small, the message can be ignored.

ERROR IN GETGFL***DESIRED ENERGY AT HIGHEST GIVEN ENERGY.

This problem should not occur in a well-constructed ENDF file. Check Files 3, 12, 13, and 14 for consistency.

ERROR IN GETCO***LIMITED TO 21 LEGENDRE COEFFICIENTS.

This limit is consistent with ENDF/B procedures. See NLMAX/21/ and P(21). If NLMAX is increased, it might also be advisable to increase the quadrature order (currently 20).

ERROR IN GETCO***LAB TO CM CONVERSION NOT CODED.

The need for this type of conversion rarely occurs on the current ENDF evaluations, because CM is consistently used for two-body reactions, and the laboratory frame is consistently used for continuum reactions. There are a few exceptions for the heavy isotopes, where CM and lab are essentially equivalent, but they were errors when the files were generated.

ERROR IN GETGYL***LO=2 NOT CODED.

This message should not occur, because any transition probability arrays on the ENDF tape should have been converted to yields by CONVER.

ERROR IN GETGYL***TOO MANY GAMMAS.

The current limit is 100 photons. See NYLMAX/100/ and LOCA(100).

ERROR IN GETGYL***STORAGE EXCEEDED.

Main container array is too small. See the variables IAMAX in GROUPR and A(25000) in /GSTORE/.

MESSAGE FROM CONVER---CANNOT DO COMPLETE PARTICLE PRODUCTION...

With the advent of the ENDF-6 format, it will be possible to make evaluations that fully describe all the products of a nuclear reaction. Some carry-over evaluations from earlier ENDF/B versions also have this capability, but many do not. This message is intended to goad evaluators to improve things!

MESSAGE FROM CONVER---GAMMA PRODUCTION PATCH MADE FOR ...

This patch is used to correct the old ENDF/B-III evaluations for MAT=1149 and MAT=1150 (chlorine and potassium).

ERROR IN GETSED***TOO MANY SUBSECTIONS.

The current limit is 20. See NKMAX/20/ and LOC(20).

ERROR IN GETSED***INSUFFICIENT STORAGE FOR RAW ENDF DATA.

The space allocated for raw File 5 or File 15 data is determined by MAXRAW/1200/. It can be increased at will, if there is enough room in the main container array.

MESSAGE FROM GETSED---UPSCATTER CORRECTION....

This reaction should not have upscatter. The error is placed into the in-group element.

ERROR IN ANASED***ILLEGAL LF.

Legal values are 5, 7, 9, 11, and 12.

ERROR IN F6PSP***3, 4, OR 5 PARTICLES ONLY.

Phase-space formulas for 3, 4, or 5 particles are provided in this routine. Check for an error in the evaluation.

T. References

1. I. I. Bondarenko, Ed., *Group Constants for Nuclear Reactor Calculations* (Consultants Bureau, New York, 1964).

2. G. I. Bell and S. Glasstone, *Nuclear Reactor Theory* (Van Nostrand Reinhold, New York, 1970).

3. R. Kinsey, Ed., "ENDF-102, Data Formats and Procedures for the Evaluated Nuclear Data File, ENDF," Brookhaven National Laboratory report BNL-NCS-50496 (ENDF-102) 2nd Edition (ENDF/B-V) (October 1979).

4. P. F. Rose and C. L. Dunford, Eds., "ENDF-102, Data Formats and Procedures for the Evaluated Nuclear Data File, ENDF-6," Brookhaven National Laboratory report BNL-NCS-44945 (July 1990).

5. R. W. Hardie and W. W. Little, Jr., "1DX, A One-Dimensional Diffusion Code for Generating Effective Nuclear Cross Sections," Battelle Northwest Laboratory report BNWL-954 (1969).

6. R. E. MacFarlane, "TRANSX-CTR: A Code for Interfacing MATXS Cross-Section Libraries to Nuclear Transport Codes for Fusion Systems Analysis," Los Alamos National Laboratory report LA-9863-MS (1984).

7. R. E. MacFarlane, "ENDF/B-IV and -V Cross Sections for Thermal Power Reactor Analysis," in Proc. Intl. Conf. of Nuclear Cross Sections for Technology, Knoxville, TN (October 22-26 1979), National Bureau of Standards Publication 594 (September 1980).

8. G. I. Bell, G. E. Hansen, and H. A. Sandmeier, "Multitable Treatments of Anisotropic Scattering in S_N Multigroup Transport Calculations," Nucl. Sci. Eng. 28, 376 (1967).

9. R. D. O'Dell, F. W. Brinkley, Jr., and D. R. Marr, "User's Manual for ONEDANT: A Code Package for One-Dimensional Diffusion-Accelerated, Neutral-Particle Transport," Los Alamos National Laboratory report LA-9184-M (February 1982).

10. C. R. Weisbin, P. D. Soran, R. E. MacFarlane, D. R. Harris, R. J. LaBauve, J. S. Hendricks, J. E. White, and R. B. Kidman, "MINX, A Multigroup Interpretation of Nuclear X-Sections from ENDF/B," Los Alamos Scientific Laboratory report LA-4750-MS (ENDF-237) (September 1976).

11. M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions* (Dover Publications, New York, 1965).

12. D. G. Madland, "New Fission Neutron Spectrum Representation for ENDF," Los Alamos National Laboratory report LA-9285-MS (ENDF-321) (April 1982).

13. C. Kalbach and F. M. Mann, "Phenomenology of Continuum Angular Distributions. I. Systematics and Parameterization," Phys. Rev. C. 23, 112 (1981).

14. C. Kalbach, "Systematics of Continuum Angular Distributions: Extensions to Higher Energies," *Phys. Rev. C.* 37, 2350 (1988).

15. P. G. Young and E. D. Arthur, "GNASH: A Preequilibrium Statistical Nuclear Model Code for Calculation of Cross Sections and Emission Spectra," Los Alamos Scientific Laboratory report LA-6947 (1977).

16. R. D. O'Dell, "Standard Interface Files and Procedures for Reactor Physics Codes, Version IV," Los Alamos Scientific Laboratory report LA-6941-MS (September 1977).

17. R. B. Kidman and R. E. MacFarlane, "LIB-IV, A Library of Group Constants for Nuclear Reactor Calculations," Los Alamos Scientific Laboratory report LA-6260-MS (1976).

IX. GAMINR

The GAMINR module of NJOY is designed to produce complete and accurate multigroup photo-atomic interaction cross sections from ENDF/B-IV, -V, or -VI data.[1, 2] Total, coherent, incoherent, pair production, and photoelectric cross sections can be averaged using a variety of group structures and weighting functions. The Legendre components of the group-to-group coherent and incoherent scattering cross sections are calculated using the form factors[3] now available in ENDF/B. These form factors account for the binding of the electron in its atom. Consequently, the cross sections are accurate for energies as low as 1 keV. GAMINR also computes partial heating cross sections or kinetic energy release in materials (KERMA) factors for each reaction and sums them to obtain the total heating factor. The resulting multigroup constants are written on an intermediate interface file for later conversion to any desired format. Photonuclear reactions such as (γ,n) are not computed by this module.

GAMINR differs from the previously used GAMLEG code[4] in the following ways:

- Coherent form factors are processed thereby allowing higher Legendre components of the coherent scattering cross section to be produced. GAMLEG processed the P_0 cross section only.
- Incoherent structure factors are processed giving accurate results at low energies where the Klein-Nishina formula fails.
- Heat-production cross sections (KERMA factors) are generated.
- Variable dimensioning and dynamic storage allocation allow arbitrarily complex problems to be run.
- GAMINR is much slower than GAMLEG since charge scaling of the incoherent matrix can no longer be used at all energies.

This chapter describes GAMINR in version 91.0.

A. Description of ENDF/B Photon Interaction Files

In the ENDF/B-IV, -V, or -VI photon interaction files, the coherent scattering of photons by electrons is represented by

$$\sigma_C(E, E', \mu)\, dE' d\mu = \frac{3\sigma_T}{8}(1+\mu^2)\,|F(q,Z)|^2 \delta(E-E')\, dE' d\mu\ , \qquad (1)$$

where E is the energy of the incident photon, E' is the energy of the scattered photon, μ is the scattering cosine, σ_T is the classical Thomson cross section (0.66524486

b), Z is the atomic number of the scattering atom, and F is the atomic form factor. The coherent form factor is a function of momentum transfer q given by

$$q = 2k\sqrt{\frac{1-\mu}{2}} , \tag{2}$$

where k is the energy in free-electron units ($k = E/511003.4\,\text{eV}$), but F is actually tabulated versus the quantity $x = 20.60744q$. The coherent form factor is given as MT=502 in File 27.

Incoherent scattering is represented by the expression

$$\sigma_I(E, E', \mu)\, dE' d\mu = S(q, Z)\, \sigma_{KN}(E, E', \mu)\, dE' d\mu , \tag{3}$$

where S is the incoherent scattering function and σ_{KN} is the free-electron Klein-Nishina cross section

$$\sigma_{KN}(E, E', \mu) = \frac{3\sigma_T}{8k^2}\left[\frac{k}{k'} + \frac{k'}{k} + 2\left(\frac{1}{k} - \frac{1}{k'}\right) + \left(\frac{1}{k} - \frac{1}{k'}\right)^2\right] . \tag{4}$$

The scattering angle and momentum transfer for incoherent scattering are given by

$$\mu = 1 + \frac{1}{k} - \frac{1}{k'} , \tag{5}$$

and

$$q = 2k\sqrt{\frac{1-\mu}{2}}\,\frac{\sqrt{1 + (k^2 + 2k)(1-\mu)/2}}{1 + k(1-\mu)} . \tag{6}$$

As was the case for coherent scattering, $S(q, Z)$ is actually tabulated vs. $x = 20.60744q$. It is important to note that S is essentially equal to Z for x greater than Z. The incoherent scattering function is given as MT=504 in File 27.

The ENDF/B-IV, -V, and -VI photon interaction files also contain tabulated cross sections for total, coherent, incoherent, pair production, and photoelectric reactions. They are given in File 23 as MT=501, 502, 504, 516, and 602 respectively (in ENDF/B-VI files, 602 is changed to 522). The coherent and incoherent cross sections were obtained by the evaluator by integrating Eqs.(1) and (3) and are therefore redundant. Due care must be taken to avoid introducing inconsistencies.

The existing photon interaction data libraries contain no photonuclear data.

B. Calculational Method

The multigroup cross sections produced by GAMINR are defined as follows:

$$\sigma_{xg} = \frac{\int_g \sigma_x \phi_0(E)\, dE}{\int_g \phi_0(E)\, dE}, \tag{7}$$

$$\sigma_{T\ell g} = \frac{\int_g \sigma_T(E)\, \phi_\ell(E)\, dE}{\int_g \phi_\ell(E)\, dE}, \text{ and} \tag{8}$$

$$\sigma_{x\ell g\to g'} = \frac{\int_g \mathcal{F}_{x\ell g'}(E)\, \sigma_x(E)\, \phi_\ell(E)\, dE}{\int_g \phi_\ell(E)\, dE}. \tag{9}$$

In these expressions, g represents an energy group for the initial energy E, g' is a group of final energies E', x stands for one of the reaction types, T denotes the total, and ϕ_ℓ is a Legendre component of a guess for the photon flux. In the last equation, $\mathcal{F}$ is the "feed function"; that is, the total normalized probability of scattering from E into group g'. The feed function for coherent scattering is

$$\begin{aligned}\mathcal{F}_{C\ell g'}(E) &= \frac{\int_{-1}^{+1} \sigma_C(E, E', \mu)\, P_\ell(\mu)\, d\mu}{\sigma_C(E)} \\ &= \frac{\int_{-1}^{+1} (1-\mu^2)\, |F(x)|^2 P_\ell(\mu)\, d\mu}{\int_{-1}^{+1} (1-\mu^2)\, |F(x)|^2 d\mu},\end{aligned} \tag{10}$$

for E in g' and zero elsewhere.

Here $P_\ell(\mu)$ is a Legendre polynomial. Note that $\mathcal{F}_{C0g'} = 1$; this form assures that the coherent scattering cross sections are consistent with the values tabulated in File 23.

For incoherent scattering,

$$\mathcal{F}_{I\ell g'} = \frac{\int_{g'} S(q, Z)\, \sigma_{KN}(E, E', \mu)\, P_\ell(\mu)\, dE'}{\int_{g'} S(q, Z)\, \sigma_{KN}(E, E', \mu)\, dE'}. \tag{11}$$

Because S is simply equal to Z for high values of q, it is not necessary to completely recompute the incoherent matrix when processing a series of elements in the order of increasing Z. GAMINR automatically determines that some groups for element Z_{now} can be obtained from element Z_{last} by scaling by Z_{now}/Z_{last}.

Finally, pair production is represented as a $(\gamma, 2\gamma)$ scattering event, where

$$\mathcal{F}_{PP\ell g'}(E) = \begin{cases} 2, & \text{if } g' \text{ includes } 511003.4 \text{ eV}; \\ 0, & \text{otherwise}. \end{cases} \tag{12}$$

In addition, GAMINR computes multigroup heating cross sections or KERMA factors as follows:

$$\sigma_{Hxg} = \frac{\int_g [E - \overline{E}_x(E)]\, \sigma_x(E)\, \phi_0(E)\, dE}{\int_g \phi_o(E)\, dE} , \tag{13}$$

where $\overline{E}_x(E)$ is the average energy of photons scattered from E by reaction type x. The average energies are computed using

$$\overline{E}_C = E , \tag{14}$$

$$\overline{E}_{PP} = 1.022007\,\text{MeV} , \tag{15}$$

$$\overline{E}_{PE} = 0, \text{ and} \tag{16}$$

$$\overline{E}_I(E) = \int_0^\infty E'\, \sigma_I(E, E', \mu)\, dE' / \sigma_I(E) . \tag{17}$$

These separate contributions to the heating cross section are summed to get a quantity that can be combined with a calculated flux to obtain the total heating rate.

C. Integrals Involving Form Factors

The integrals of Eqs.(10) and (11) that involve the form factors are very difficult to perform because of the extreme forward peaking of the scattering at high energies. Figure 1 illustrates the problem for coherent scattering.

For coherent scattering, the integral of Eq.(10) is broken up into panels by the tabulation values of x. Each panel is integrated in the x domain using Lobatto quadrature of order 6 for $\ell = 3$ or less and order 10 for larger Legendre orders. Equation (2) is used to compute the μ value for each x and to obtain the Jacobian required.

Since μ is quadratic in x, the polynomial order of the integrand in the numerator of Eq.(10) is $2\ell + 2$ plus twice the polynomial order of F in the panel. For $\ell = 3$, the theory of Gaussian quadrature implies that the integral will be exact if F can be represented as a quadratic function over the panel.

The incoherent integral of Eq.(11) is also broken up into panels, but here the panels are defined by the union of the tabulation values of x and the momenta corresponding to the secondary-energy group bounds. The relationship between x, μ, and secondary energy is given in Eqs.(5) and (6). This time, the integral is performed vs. E' using Lobatto quadrature of order 6 for ℓ less than or equal to 5 and order 10 for larger ℓ values.

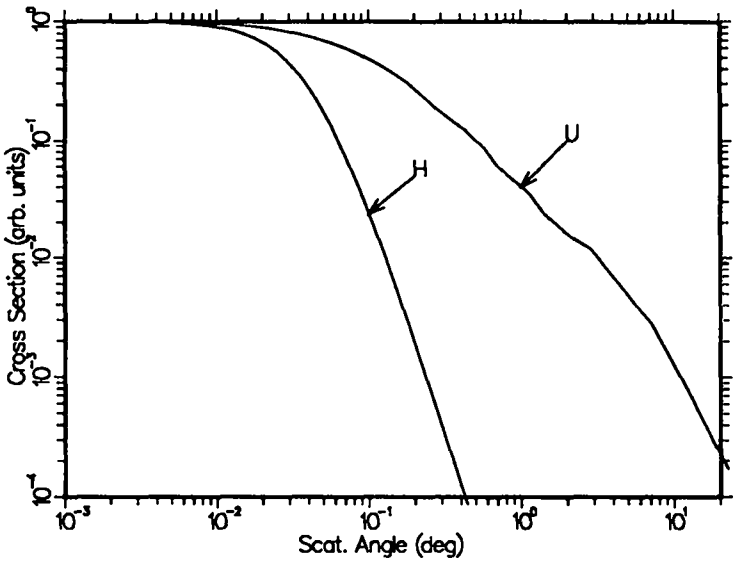


Figure 1: Angular distribution for coherent scattering from hydrogen and uranium at high photon energy.

All form factors and structure factors are interpolated using ENDF/B log-log interpolation as specified by the format. However, the cross sections in the file were evaluated using a special log-log-quadratic scheme. Ignoring this complication may lead to a 5% error in the incoherent cross section at 0.1 keV with a negligible error at the higher energies that are of most practical concern.[3]

D. Coding Details

The code begins by reading the user's input. It then locates the position for the new material on the old GENDF tape (if any) and copies the earlier results to the new output tape. The desired material is also located on the input PENDF tape prepared previously using RECONR. A new material header is then written onto the output tape leaving the code ready to begin the loop over reaction types.

For each of the preset reaction types, GAMINR uses the panel logic of GROUPR to average the cross sections (see subroutine **GPANEL** and the GROUPR write-up). If this is the first material in a series of elements, the incoherent matrix is saved to a scratch area. For subsequent materials, the higher energy matrix elements are obtained by scaling these saved values by the appropriate Z ratio. The resulting cross sections and group-to-group matrix elements are then printed out and written to the output tape. The heat production contribution from each reaction is summed into a storage area. After all reactions have been processed for

this material, a special pass through the output logic is used to create a heating section using MT=621 (or MT=525 for ENDF/B-VI). Finally, the rest of the old output tape is copied to the new output tape. A description of the format of the multigroup output tape will be found in the GROUPR write-up.

As with PANEL in GROUPR, GPANEL integrates the triple product $\mathcal{F}*\sigma*\phi$. The feed into secondary group g' for Legendre order ℓ from initial energy E is computed in gtff as described in Section IX.C above. Cross sections are read from the PENDF tape (see GTSIG). Flux can be read in, constant, or $1/E$ with high and low energy roll-offs (see GNWTF and GTFLX).

E. User Input

The following description of the user input is reproduced from the comment cards at the beginning of the GAMINR module.

```
*---INPUT SPECIFICATIONS (FREE FORMAT)-------------------------*
*                                                              *
* CARD1                                                        *
*    NENDF   UNIT FOR ENDF/B TAPE                              *
*    NPEND   UNIT FOR PENDF TAPE                               *
*    NGAM1   UNIT FOR INPUT NGAM TAPE (DEFAULT=0)              *
*    NGAM2   UNIT FOR OUTPUT NGAM TAPE (DEFAULT=0)             *
* CARD2                                                        *
*    MATB    MATERIAL TO BE PROCESSED                          *
*            INPUT MATERIALS IN ASCENDING ORDER                *
*    IGG     GAMMA GROUP STRUCTURE OPTION                      *
*    IWT     WEIGHT FUNCTION OPTION                            *
*    LORD    LEGENDRE ORDER                                    *
*    IPRINT  PRINT OPTION (0/1=MINIMUM/MAXIMUM) (DEFAULT=1)    *
* CARD3                                                        *
*    TITLE   RUN LABEL UP TO 80 CHARACTERS (DELIMITED BY *,    *
*            ENDED WITH /)                                     *
* CARD4      (IGG=1 ONLY)                                      *
*    NGG     NUMBER OF GROUPS                                  *
*    EGG     NGG+1 GROUP BOUNDS (EV)                           *
* CARD5      (IWT=1 ONLY)                                      *
*    WGHT    WEIGHT FUNCTION AS TAB1 RECORD                    *
* CARD6                                                        *
*    MFD     FILE TO BE PROCESSED                              *
*    MTD     SECTION TO BE PROCESSED                           *
*    MTNAME  DESCRIPTION OF SECTION TO BE PROCESSED            *
*            REPEAT FOR ALL REACTIONS DESIRED                  *
*            MFD=0/ TERMINATES THIS MATERIAL                   *
*            MFD=-1/ IS A FLAG TO PROCESS ALL SECTIONS PRESENT *
*            FOR THIS MATERIAL  (TERMINATION IS AUTOMATIC)     *
* CARD7                                                        *
*    MATD    NEXT MAT NUMBER TO BE PROCESSED                   *
*            TERMINATE GAMINR RUN WITH MATD=0.                 *
```

```
*                                                                    *
*---OPTIONS FOR INPUT VARIABLES---------------------------------------*
*                                                                    *
*        IGG     MEANING                                             *
*        ---     -------                                             *
*         0      NONE                                                *
*         1      ARBITRARY STRUCTURE (READ IN)                       *
*         2      CSEWG 94-GROUP STRUCTURE                            *
*         3      LANL 12-GROUP STRUCTURE                             *
*         4      STEINER 21-GROUP GAMMA-RAY STRUCTURE                *
*         5      STRAKER 22-GROUP STRUCTURE                          *
*         6      LANL 48-GROUP STRUCTURE                             *
*         7      LANL 24-GROUP STRUCTURE                             *
*         8      VITAMIN-C 36-GROUP STRUCTURE                        *
*         9      VITAMIN-E 38-GROUP STRUCTURE                        *
*                                                                    *
*        IWT     MEANING                                             *
*        ---     -------                                             *
*         1      READ IN                                             *
*         2      CONSTANT                                            *
*         3      1/E + ROLLOFFS                                      *
*                                                                    *
**********************************************************************
```

Note that both an ENDF/B tape and a PENDF tape from RECONR are required. [The photon interaction tapes are available from the Radiation Shielding Information Center (RSIC) at the Oak Ridge National Laboratory as DLC7E (for ENDF/B-IV) or DLC-99/HUGO (for ENDF/B-V). A photoatomic library in ENDF-VI format based on DLC-99 is available from the National Nuclear Data Center (NNDC) at the Brookhaven National Laboratory.] Material numbers (MATB) are simply the element Z number for versions IV and V; they are equal to $100*Z$ for ENDF/B-VI. The values of MATD on card 7 should be given in increasing order for maximum economy. The normal mode of operation uses MFD= -1. This automatically processes MT=501, 502, 504, 516, 602, and 621. For ENDF/B-VI, the photoelectric cross section is changed from 602 to 522 and the heating cross section is changed from 621 to 525.

The following sample run prepares a GENDF tape for two elements. The numbers on the left are for reference; they are not part of the input.

```
1.  0
2.  4
3.  RECONR
4.  20 21
5.  *PENDF TAPE FOR 2 ELEMENTS FROM DLC7E*/
6.  1 1 0
7.  .001 0./
8.  *1-HYDROGEN
```

```
 9.   92 1 0
10.   .001 0./
11.   *92-URANIUM*/
12.   0/
13.   GAMINR
14.   20 21 0 22
15.   1 3 3 4 0
16.   *12-GROUPR PHOTON INTERACTION LIBRARY*/
17.   -1/
18.   92
19.   -1/
20.   0/
21.   STOP
```

The input for RECONR is described more completely in RECONR, Chapter III of this manual. On card 4, an ENDF/B tape containing File 23 has been mounted on logical unit 20. The title on line 5 will appear on the PENDF tape. Material 1 is hydrogen (cards 6–8) and material 92 is uranium (cards 9–11). The element names on cards 8 and 11 will appear on the PENDF tape in MF1, MT451. Linearization is accurate to better than 0.1%.

GAMINR uses the same ENDF tape as RECONR (actually only MF=27 is read by GAMINR), but GAMINR also reads the RECONR output tape on unit 21. The GAMINR GENDF tape will be on unit 22. Card 15 specifies hydrogen as the first material, 12 groups, $1/E$ weight with roll-offs, Legendre components through P_3, and the full printed output. Cards 17 and 19 select the default list of reaction types. Card 18 specifies uranium as the second material to be processed, and card 20 terminates the element loop and the GAMINR run.

Figures 2 through 4 give plots of the results of this sample run. These graphs were made using the DTFR module.

The plots of photon transport matrices require some explanation. These curves are given for the "positions" of the transport table; that is, the back curve is for in-group scattering, the next one is the scattering into a group from the next higher-energy group, and so on. The horizontal axes are determined by group index, irrespective of the actual energy width of the group. The vertical scale is logarithmic, and the "max" and "min" values are also logarithms.

F. Error Messages

`ERROR IN RESERV***STORAGE EXCEEDED.`

Increase the size of the `STORAG` container array `/STORE/A(28000)` and increase `IAMAX` to match. Both of these variables will be found in the main routine of GAMINR.

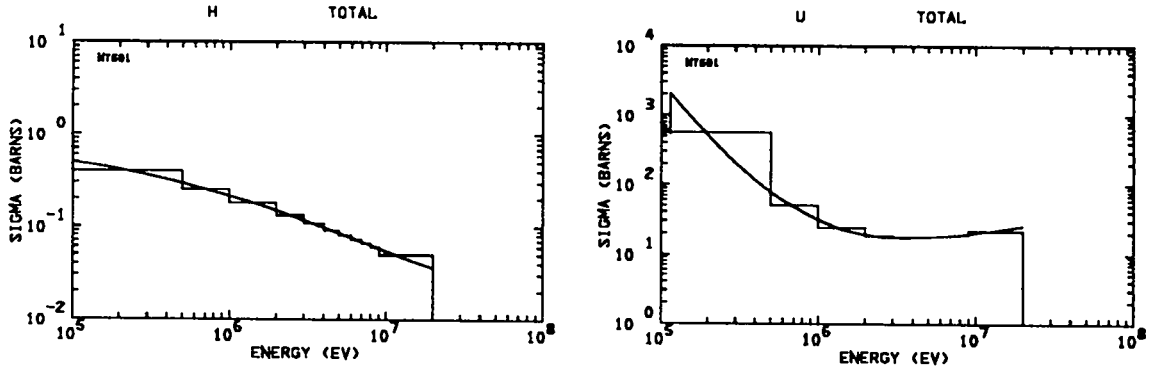


Figure 2: Comparison of pointwise and 12-group total photon interaction cross sections for hydrogen and uranium. Note the photoelectric absorption edge near 100 keV in uranium.

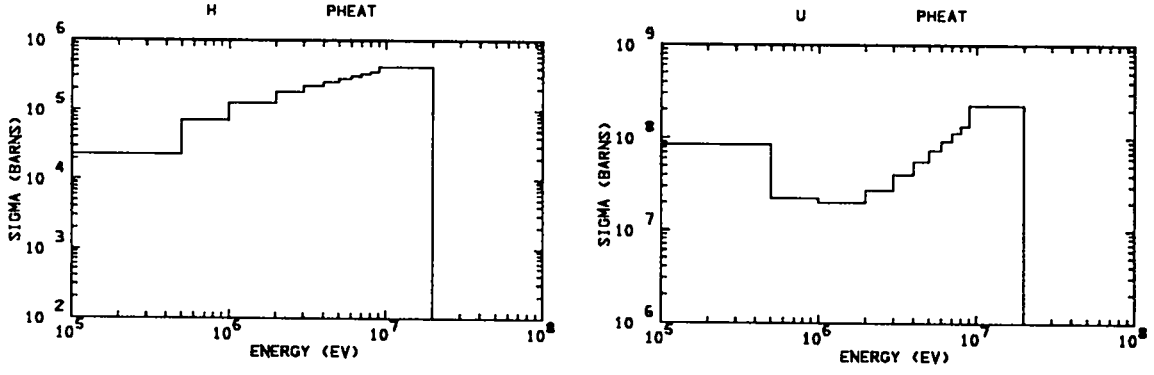


Figure 3: Comparison of 12-group photon heat production cross sections for hydrogen and uranium. The ordinate should actually read "eV-barns".

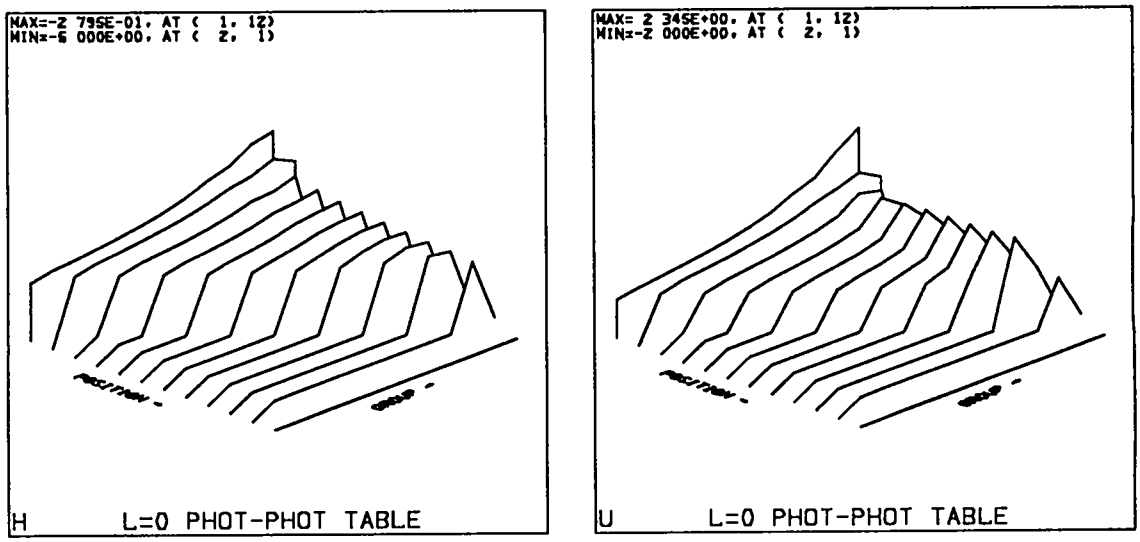


Figure 4: Comparison of photon transport matrices for hydrogen and uranium. These curves are given for the "positions" in the transport table (see text).

`ERROR IN GENGPG***ILLEGAL GROUP STRUCTURE`

Values of IGG must lie between 1 and 9.

`ERROR IN GENGPG***TOO MANY GROUPS.`

Increase /GROUPG/ in GAMINR and NGMAX=150 in subroutine GENGPG.

`ERROR IN GPANEL***ELO GT EHI.`

There is something wrong with the energy grids during integration over incident energy. This usually means there is a problem with the choice of RNDOFF and/or DELTA. Be sure that RNDOFF<1, DELTA>1, and RNDOFF*DELTA<1 as represented on your machine.

`ERROR IN GTFF***ILLEGAL FILE TYPE.`

Only files 23 and 26 can be requested.

`ERROR IN GTFF***ILLEGAL REACTION FOR CROSS SECTION=---`

Only reactions 501, 502, 504, 516, 602, and 621 (heat) can be requested for ENDF/B-IV or -V, or only 501, 502, 504, 516, 522, and 525 for ENDF/B-VI.

`ERROR IN GTFF***INSUFFICIENT STORAGE FOR FORM FACTOR.`

Increase size of container array as above.

G. I/O Units

There are no scratch files used in GAMINR. The only restriction on the files assigned on card 1 of the user input is that NGAM1 and NGAM2 must be in the same mode (that is, both binary or both formatted).

H. References

1. R. Kinsey, "ENDF-102, Data Formats and Procedures for the Evaluated Nuclear Data File, ENDF," Brookhaven National Laboratory report BNL-NCS-50496 (ENDF-102) (October 1979).

2. R. W. Roussin, J. R. Knight, J. H. Hubbell, and R. J. Howerton, "Description of the DLC-99/HUGO Package of Photon Interaction Data in ENDF/B-V Format," Oak Ridge National Laboratory report ORNL/RSIC-46 (ENDF-335) (December 1983).

3. J. H. Hubbell, Wm. J. Viegle, E. A. Briggs, R. T. Brown, D. T. Cromer, and R. J. Howerton, "Atomic Form Factors, Incoherent Scattering Functions, and Photon Scattering Cross Sections," *J. Phys. Chem. Ref. Data* **4**, 471 (1975).

4. K. D. Lathrop, "GAMLEG–A FORTRAN Code to Produce Multigroup Cross Sections for Photon Transport Calculations," Los Alamos Scientific Laboratory report LA-3267 (1965).

X. ERRORR

A. Introduction

After evaluators have completed their review of the available measurements of various nuclear data (having true values $\sigma_1, \sigma_2, \sigma_3, \cdots$) and the theoretical analysis, they will have formed at least a subjective opinion of the joint probability distribution of the data examined; that is, the probability

$$P(\sigma_1, \sigma_2, \cdots)\, d\sigma_1\, d\sigma_2 \cdots$$

that the true value of σ_1 lies in the range $(\sigma_1, \sigma_1+d\sigma_1)$, and that σ_2 lies in the range $(\sigma_2, \sigma_2+d\sigma_2)$, etc. In the early versions of the ENDF format, only the first moments (expectation values) of this probability distribution could be included in the numerical data files. However, beginning with ENDF/B-IV and expanding significantly in ENDF/B-V and ENDF/B-VI, the second moments of the data probability distributions have been included in many of the files. As discussed in Section B, these second moments (or "data covariances") contain information on the uncertainty of individual data, as well as correlations that may exist.

Data covariances have many applications. For example, they can be combined with sensitivity coefficients to obtain the uncertainty, due to the data, in calculated quantities of applied interest.[1] This information can be used in turn to judge the adequacy of the data for that application.

The availability of data covariances also makes it possible to use the generalized method of least squares to improve the data evaluation after new integral or differential measurements have been performed.[2] The least-squares method requires only data covariances (not the full probability distribution), and the improved, or adjusted, data are guaranteed[3] to have the smallest possible uncertainties, regardless of the actual shape of the underlying probability distribution function, $P(\sigma_1, \sigma_2, \cdots)$. Thus, the ENDF-formatted covariance files contain, in about as compact a form as possible, a statement about the quality of the data, as well as sufficient information (in principle) to carry out future improvements on an objective basis.

In many of these applications, it is necessary to begin by converting energy-dependent covariance information in ENDF format[4] into multigroup form. This task can be performed conveniently in the NJOY environment, using the ERRORR module. In particular, ERRORR calculates the uncertainty in infinitely dilute multigroup cross sections (or multigroup $\bar{\nu}$ values), as well as the associated

correlation coefficients. These data are obtained by combining absolute or relative covariances from ENDF Files 31, 32, and 33 with cross-section or $\bar{\nu}$ data from an existing processed cross-section library, either containing resonance-reconstructed "point" data or containing multigroup data from the GROUPR module of NJOY. ERRORR is coded to treat all approved ENDF-4, -5, and -6 covariance formats for these three files, up to and including a new covariance "law," LB=8, except that the processing of File 32 is restricted to Breit-Wigner resolved-resonance parameter uncertainties (LRF=1 and 2) in Version-5 format or in the "Version-5 compatible" option of Version 6 (LCOMP=0). The present version of ERRORR does not process information[5] from Files 30, 34, 35, and 40, nor from the recently proposed[6] File 36. Work is presently underway to remedy these shortcomings, with the focus being placed on Version-6 format features that are being used by current evaluators.

The methodology of ERRORR assumes that the weighting flux used to convert energy-dependent cross sections into multigroup averages is free of uncertainty. In calculating the effect of uncertainty in resonance parameters on the multigroup cross sections, it is approximated that the entire resonance area is contained in the energy group that contains the peak. Finally, in cases in which the cross-section information is obtained from an existing multigroup library, it is usually necessary to make assumptions about the shape of the cross section and the weight function within certain input energy groups, as discussed below in Section X.E. These are the only approximations or assumptions inherent to the program.

B. Definitions of Covariance-Related Quantities

For convenient reference in discussing the methodology and input requirements of the ERRORR module, we next review the basic definitions of covariance-related quantities. Let x_0 and y_0 be the evaluated values of x and y, respectively:

$$x_0 \equiv \mathrm{E}[x] \,, \tag{1}$$

and

$$y_0 \equiv \mathrm{E}[y] \,. \tag{2}$$

Here E is the expectation operator, which performs an average over the joint probability distribution of x and y. The second moment of this distribution is called the covariance of x with y:

$$\mathrm{cov}(x,y) \equiv \mathrm{E}[(x - x_0)\,(y - y_0)] \,. \tag{3}$$

Covariance is a measure of the degree to which x and y are both affected by the same sources of error. The covariance of x with itself is called the variance of x:

$$\mathrm{var}(x) \equiv \mathrm{cov}(x,x) = \mathrm{E}[(x - x_0)^2] \,. \tag{4}$$

The more familiar standard deviation Δx (also called the "uncertainty") is simply

$$\Delta x \equiv [\mathrm{var}(x)]^{1/2} = [\mathrm{cov}(x,x)]^{1/2} \,. \tag{5}$$

The correlation between x and y (also called the correlation coefficient) is defined as

$$\mathrm{corr}(x,y) \equiv \frac{\mathrm{cov}(x,y)}{\Delta x \, \Delta y} \,. \tag{6}$$

The absolute value of a correlation coefficient is guaranteed to be less than or equal to unity. Another useful quantity is the relative covariance of x with y,

$$\mathrm{rcov}(x,y) \equiv \frac{\mathrm{cov}(x,y)}{x_0 \, y_0} \,. \tag{7}$$

Unlike $\mathrm{cov}(x,y)$, the relative covariance $\mathrm{rcov}(x,y)$ is a dimensionless quantity. Closely related to the relative covariance is the relative standard deviation,

$$\frac{\Delta x}{x_0} = \frac{[\mathrm{cov}(x,x)]^{1/2}}{x_0} \,, \tag{8}$$

which, from Eq. (7), can be written as

$$\frac{\Delta x}{x_0} = [\mathrm{rcov}(x,x)]^{1/2} \,. \tag{9}$$

Combining Eqs. (6) and (7), we have another useful result,

$$\mathrm{corr}(x,y) = \frac{\mathrm{rcov}(x,y)}{(\Delta x/x_0)(\Delta y/y_0)} \,. \tag{10}$$

While it is customary to speak of uncertainties and correlations as separate entities, these are actually just two different aspects of the covariance. If one has a set of absolute covariances for various reactions, including the self-covariance, then Eqs. (5) and (6) can be used to calculate Δx and $\mathrm{corr}(x,y)$. Similarly, if one has a set of relative covariances, one can use Eqs. (9) and (10) to calculate $\Delta x/x_0$ and $\mathrm{corr}(x,y)$.

Consider now a set of nuclear data σ_i with uncertainties characterized by the covariances $\mathrm{cov}(\sigma_i, \sigma_j)$. Let A and B be two linear functions of the σ_i,

$$A = \sum_i a_i \, \sigma_i$$

$$B = \sum_j b_j \, \sigma_j \, ,$$

where the a_i and b_j are sets of known constants. The above definitions can be used to calculate the covariances of the functions A and B induced by the covariances of the data. From Eq. (3),

$$\begin{aligned} \mathrm{cov}(A,B) &= \mathrm{E}\left\{ \left(\sum_i a_i \, \sigma_i - \sum_i a_i \mathrm{E}(\sigma_i) \right) \left(\sum_j b_j \, \sigma_j - \sum_j b_j \, \mathrm{E}(\sigma_j) \right) \right\} \\ &= \sum_{i,j} a_i \, b_j \, \mathrm{E}\left\{ \left(\sigma_i - \mathrm{E}(\sigma_i)\right) \left(\sigma_j - \mathrm{E}(\sigma_j)\right) \right\} \, , \end{aligned}$$

so that

$$\mathrm{cov}(A,B) = \sum_{i,j} a_i \, b_j \, \mathrm{cov}(\sigma_i, \sigma_j) \, . \qquad (11)$$

This result, called the "propagation of errors" formula, is fundamental to the subject of multigroup processing of ENDF covariance data and will be referenced frequently in later sections of this chapter.

C. Structure of ENDF Files 31 and 33: Energy-Dependent Data

Data in ENDF format are stored in various numbered "files," where the file number depends on the type of information contained. For example, the covariances of $\bar{\nu}(E)$ (the average number of neutrons per fission, which is a function of the incident neutron energy) are stored in File 31, where the possible "reaction" types are prompt $\bar{\nu}$, delayed $\bar{\nu}$, and total $\bar{\nu}$. File 32 describes the uncertainty in the shape and area of individual resonances. File 33 contains the covariances of energy-dependent cross sections. Within the resolved-resonance energy range, File 33 includes any "long-range" covariances in the reconstructed cross sections, such as covariance components that are highly correlated over many neighboring resonances. The structures of Files 31 and 33 are identical and will be described first.

Files 31 and 33 describe the covariances of energy-dependent data. To expand on this point, we recall that the full energy dependence of a cross section $\sigma(E)$, for example, is described in the ENDF File 3 by specifying the cross-section values at a

relatively small number of energy points and then providing a set of interpolation laws to be used in reconstructing the actual cross section at any intermediate energy. Somewhat the same philosophy is used to describe the two-dimensional energy dependence of data covariances in Files 31 and 33. That is, one specifies a set of numerical data and a set of formulae, which together can be used to compute $\mathrm{cov}(x,y)$ for any desired pair of energies, E_x and E_y. Although the interpolation laws are presently restricted to the simple forms described below, it is not true (as sometimes stated) that ENDF contains multigroup covariances. The expression "multigroup covariance" refers to the covariance of one multigroup-averaged quantity with another averaged quantity, whereas ENDF contains the covariances between point-energy data. It is precisely the task of ERRORR to compute multigroup covariances, starting from point covariances.

Files 31 and 33 of an evaluation for material **MAT** are divided into "sections," indexed by the reaction type **MT**. A section **(MAT,MT)** is further subdivided into "subsections." As described in Ref. 4, a subsection is the repository for all explicit statements of the two-dimensional energy dependence of the covariances of reaction **(MAT,MT)** with another reaction **(MAT1,MT1)**. Because covariances are symmetric, a subsection with **MAT1=MAT** and **MT1<MT** would be redundant with a subsection in an earlier section, and such data are, by convention, omitted from the ENDF files.

Subsections are further divided into "sub-subsections." Two different types of sub-subsections are used in the ENDF-5 and ENDF-6 formats. (Although ERRORR also treats data covariances in the earlier ENDF-4 format, this is of little practical interest since only three covariance evaluations were released in Version-4 format.) "NI-type" sub-subsections are used to express covariances explicitly, while "NC-type" sub-subsections are used to indicate the existence of connections between various data that result in "implicit" covariance contributions for various reaction pairs. We shall return to this point in NC-Type Sub-Subsections, Section X.C.2.

1. NI-Type Sub-Subsections. Multiple NI-type sub-subsections are used to describe multiple, statistically independent sources of uncertainty for a given reaction pair. If $\mathrm{cov}(x,y)_n$ is the covariance computed from the data in one sub-subsection, then, because the uncertainties in different sub-subsections are uncorrelated,

$$\mathrm{cov}(x,y) = \sum_{n=1}^{\mathrm{NI}} \mathrm{cov}(x,y)_n ,$$

where NI is the number of NI-type sub-subsections in the current subsection. The numerical content of one NI-type sub-subsection consists of either one or two en-

ergy grids, a collection of constants, and a parameter LB. The parameter LB governs how the energies and constants are to be used in constructing the covariance in various rectangular regions of $E_x - E_y$ space. For LB=0, 1, 2, and 8, a single table containing pairs (E_i, F_i) is given. For LB=3 and 4, two such tables are given. For LB=5, a single set of energies E_i is given, along with an associated square matrix of constants G_{ij}. Finally, for LB=6, two energy grids are given, along with an associated rectangular matrix of constants G'_{ij}.

The first of these tables (E_i, F_i) defines a function $f(E)$ that is constant except for discrete steps at energies E_i,

$$f(E) = F_i, \text{ if } E_i \leq E < E_{i+1} . \tag{12}$$

Similarly, if there is a second table,

$$f'(E) = F'_i, \text{ if } E'_i \leq E < E'_{i+1} . \tag{13}$$

The LB=5 matrix data G_{ij} also define a function,

$$g(E_x, E_y) = G_{ij}, \text{ if } E_i \leq E_x < E_{i+1} \text{ and } E_j \leq E_y < E_{j+1} , \tag{14}$$

and similarly for LB=6

$$g'(E_x, E_y) = G'_{ij}, \text{ if } E_i \leq E_x < E_{i+1} \text{ and } E'_j \leq E_y < E'_{j+1} . \tag{15}$$

These functions are simply histograms in either one or two dimensions. Using the functions f, f', g, and g' thus defined, we can list the formulae in Eq. (16), which are used to specify energy-dependent covariances for the different allowed values of LB. Thus, if x is the value of the cross section or of the $\bar{\nu}$ value for the reaction (MAT,MT) that determines the ENDF/B section, and if y is the value for the reaction (MAT1,MT1) that determines the subsection, then for

$$\begin{aligned}
\text{LB} &= 0, \quad \text{cov}(x,y)_n &&= f(E_x)\,\delta(E_x, E_y) \\
\text{LB} &= 1, \quad \text{rcov}(x,y)_n &&= f(E_x)\,\delta(E_x, E_y) \\
\text{LB} &= 2, \quad \text{rcov}(x,y)_n &&= f(E_x)\,f(E_y) \\
\text{LB} &= 3, \quad \text{rcov}(x,y)_n &&= f(E_x)\,f'(E_y) \\
\text{LB} &= 4, \quad \text{rcov}(x,y)_n &&= f(E_x)\,\delta(E_x, E_y)\,f'(E_x)\,f'(E_y) \\
\text{LB} &= 5, \quad \text{rcov}(x,y)_n &&= g(E_x, E_y) \\
\text{LB} &= 6, \quad \text{rcov}(x,y)_n &&= g'(E_x, E_y) .
\end{aligned} \tag{16}$$

The symbol $\delta(E_x, E_y)$ has the following meaning: $\delta(E_x, E_y)$=1 if E_x and E_y fall in the same energy interval of the first table (E_i, F_i), and $\delta(E_x, E_y)$=0 otherwise.

The final covariance law, LB=8, is an exceptional case that cannot be expressed in terms of point covariances. LB=8 is used primarily to represent uncertainty effects due to suspected, but unresolved, energy-dependent structure in a given cross section. If ΔE_j is the energy width of the j-th "union" energy group (see the discussion of union groups in Section X.E), and if this group lies within the k-th range (E_k, E_{k+1}) of an ENDF LB=8 energy grid, then the effect of the LB=8 subsection is to trigger the addition of an uncorrelated contribution of $F_k\,(E_{k+1} - E_k)/\Delta E_j$ to the (absolute) variance of the j-th union-group cross section. No contributions to off-diagonal elements of the multigroup covariance matrix are generated by an LB=8 sub-subsection.

2. NC-Type Sub-Subsections. NC-type sub-subsections, which describe covariances indirectly, are used in several evaluation situations, which are flagged by different values of the parameter LTY. The first situation, LTY=0, occurs when the following two conditions are met: (a) the covariances of a given reaction MT, both with itself and with other reactions, can be deduced in an energy range (E1,E2) solely from the application of a cross-section "derivation relation,"

$$x(\mathtt{MAT},\mathtt{MT};E) = \sum_i C_i\, x(\mathtt{MAT},\mathtt{MT}_i;E), \tag{17}$$

and (b) the covariances of all of the reaction cross sections on the right-hand side of Eq. (17) have been given directly (that is, using only NI-type sub-subsections) throughout the range (E1,E2). The energy boundaries E1 and E2, the constants C_i, and the reaction identifiers $\mathtt{MT}_i$ are specified in an NC-type sub-subsection with LTY=0. This format is widely used in the ENDF/B library, and it makes possible the elimination of large volumes of otherwise redundant data. It also introduces considerable complexity in the multigroup processing, as discussed in Section G, and adds to the computer running times. The presence of this one short sub-subsection affects the calculation of the covariances for many different reaction pairs, such as x(MAT,MT) with x(MAT,$\mathtt{MT}_i$). Less widely used are NC-type sub-subsections with LTY=1. These are employed when a reaction MT in material MAT is evaluated in some energy range (E1,E2) as a ratio to a standard reaction MTS in some other material MATS. That is,

$$x(\mathtt{MAT},\mathtt{MT};E) = R(E)\, x(\mathtt{MATS},\mathtt{MTS};E). \tag{18}$$

In practical evaluation situations, the uncertainty of R is almost never correlated with that of x(MATS,MTS). Because of this, the relative uncertainty in R is treated simply as one independent component of the relative uncertainty in x(MAT,MT), and it is described using normal NI-type sub-subsections. On the other hand, the contribution from uncertainty in x(MATS,MTS) is represented with an NC-type sub-subsection with LTY=1, which contains, in ENDF/B-V, only E1, E2, MATS, and MTS. The actual numerical covariance information must be read from the evaluation for the standard material MATS, which usually resides on an entirely different ENDF tape. An NC-type sub-subsection with LTY=1, which occurs in a given subsection (MAT,MT; MAT,MT), affects the calculation of the covariances only for the current reaction pair (reaction MT with itself) and, in this respect, is more like an NI-type sub-subsection than, for example, an NC-type sub-subsection with LTY=0. This similarity is exploited in the processing of ratio covariances, as discussed in Section X.H.

An NC-type sub-subsection with LTY=2 is used, in a similar way, to describe the covariances of x(MAT,MT) with x(MATS,MTS). As in the LTY=1 case, an LTY=2 sub-subsection contains only E1, E2, MATS, and MTS.

An NC-type sub-subsection with LTY=3 is included in material MATS to describe the (redundant) covariances of x(MATS,MTS) with x(MAT,MT). The numerical information contained here is the same as for LTY=1 and 2. As discussed in Section X.H, an important function of LTY=3 data is to help locate reactions other than (MAT,MT) that have been measured relative to the same standard (MATS,MTS).

D. Resonance-Parameter Formats—File 32

File 32 contains covariances of resonance parameters from File 2. ERRORR processes File 32 in the following limited sense: when infinite-dilution cross-section covariances are processed (see previous section) from File 33, the diagonal elements of the resulting (self-reaction) multigroup covariance matrices are augmented by contributions based on the parameter covariances in File 32.

For evaluations produced in ENDF-5 format, all approved covariance format options are treated. For ENDF-6, many new possibilities were created,[5] including the capability of specifying uncertainty in unresolved-resonance parameters (LRU=2) and to represent correlations between the resolved parameters of different resonances (LCOMP$\neq$ 0). At present ERRORR treats the ENDF-6 File-32 format only in the case where resolved parameters are treated in the "Version-5 compatible" format (LCOMP=0, LRU=1, LRF=1 or 2 only). This restriction is not as serious as it may sound, because, as of the issue date of Release 2 of ENDF/B-VI, all

ENDF/B-VI evaluations containing File 32 use the simple "Version-5 compatible" format. The following discussion, therefore, is limited to this option.

For either of the permitted resolved resonance formalisms (LRF=1 or 2), the parameters considered in File 32 are the resonance energy E_r, the neutron width Γ_n, the radiative capture width Γ_γ, the fission width Γ_f, and (in Version 5 only) the total angular momentum J. All cross-parameter relative covariances, such as rcov(Γ_n,Γ_γ), are included, with the exception of the covariances of E_r with the remaining parameters, which are assumed to be negligible. Cross-resonance covariances, such as cov(E_r^i, E_r^j), where i and j refer to different resonances, are omitted in the LCOMP=0 option.

E. Calculation of Multigroup Fluxes, Cross Sections, and Covariances on the Union Grid

As mentioned before, the main function of the ERRORR module is to calculate the uncertainty in group-averaged cross sections at infinite dilution due to uncertainty in the ENDF point data. In this and the following sections, we describe the procedures used in performing this task.

In order to proceed, it is necessary to introduce into the discussion three different energy grids, namely, the user grid, the ENDF grid, and the union grid. The relationship of these three grids is shown in Fig. 1. The user grid is the multigroup structure in which the output multigroup covariances are to be produced. The ENDF grid is the collection of energies obtained (in subroutines GRIDD and GRIST) by forming the union of (a) all energy "lists" appearing in any NI-type sub-subsection of any subsection to be processed in the current ENDF material, and (b) all energy pairs used to define the range of effectiveness of any NC-type sub-subsection of any subsection to be processed. The union grid, on the other hand, is simply the union of the user grid and ENDF grid. The utility of the union grid is that (a) the covariances are particularly simple to calculate in this grid, as discussed below, and (b) the multigroup covariances needed by the user are then easily obtainable by a straightforward collapse from this to the user grid. In fact, the design of the current ENDF covariance format was strongly influenced by the desire to employ this particular procedure for multigroup processing.[7, 8]

After the union grid is formed in SUBROUTINE UNIONG, the cross sections $x(E)$ and weighting flux $\phi(E)$ are integrated to produce x_I and ϕ_I, multigrouped on this grid. If point cross sections are supplied, an exact integration is done in SUBROUTINE GRPAV. If, on the other hand, a multigroup cross-section library is supplied, then SUBROUTINE COLAPS is used. If a library group is subdivided by

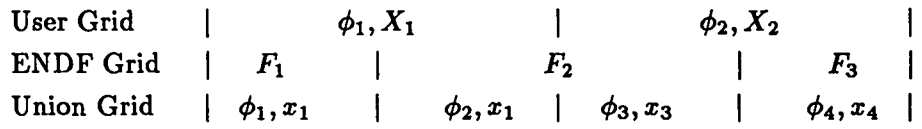


Figure 1: Illustration of energy grid relations.

a union-group boundary, then over the span of that library group, the unknown energy dependencies of $x(E)$ and $\phi(E)$, which are needed to calculate x_I and ϕ_I, are approximated in SUBROUTINE COLAPS by constants. Normally, the effect of this approximation is not large and, in any case, can be reduced or eliminated by increasing the number of groups in the input library.

We next consider the theoretical basis for the calculation of union-grid multigroup covariances, as performed in SUBROUTINE COVCAL. By definition, x_I is just the average of $x(E)$ over union group I,

$$x_I \equiv \frac{\int_I \phi(E)\, x(E)\, dE}{\int_I \phi(E)\, dE}\,, \tag{19}$$

where $\phi(E)$ is the flux "model" assumed for the multigroup calculations. Let y_J denote similarly the average of $y(E)$ over union group J.

Let us imagine that these groups are subdivided into many subintervals of infinitesimal width, so that in the i-th subinterval of group I, for example, $x(E)$ can be well approximated by the constant x_i. By this device, the integrals that define x_I and y_J can be converted to discrete sums:

$$x_I = \frac{\sum_{i\in I} \phi_i\, x_i}{\phi_I} = \sum_{i\in I} \alpha_{Ii}\, x_i\,, \tag{20}$$

where

$$\phi_i = \int_i \phi(E)\, dE\,, \tag{21}$$

$$\phi_I = \sum_{i\in I} \phi_i = \int_I \phi(E)\, dE\,, \tag{22}$$

and

$$\alpha_{Ii} \equiv \frac{\phi_i}{\phi_I} \,. \tag{23}$$

From these definitions, clearly

$$\sum_{i \in I} \alpha_{Ii} \;=\; 1 \,. \tag{24}$$

Similarly for y,

$$y_J = \sum_{j \in J} \alpha_{Jj}\, y_j \,, \tag{25}$$

and

$$\sum_{j \in J} \alpha_{Jj} = 1 \,. \tag{26}$$

The methodology of ERRORR assumes that $\phi(E)$ in Eq. (19) is free of uncertainty. Under this assumption, the terms α_{Ii} and α_{Jj} are simply known constants. The covariance of x_I with y_J can then be calculated using the propagation-of-errors formula, Eq. (11), together with Eqs. (20) and (25).

$$\begin{aligned} \mathrm{cov}(x_I, y_J) \;&=\; \sum_{\substack{i \in I \\ j \in J}} \alpha_{Ii}\, \alpha_{Jj}\, \mathrm{cov}(x_i, y_j) \\ &=\; \sum_{\substack{i \in I \\ j \in J}} \alpha_{Ii}\, \alpha_{Jj} \sum_n \mathrm{cov}(x_i, y_j)_n \,, \end{aligned} \tag{27}$$

where the summation over n results from the different independent contributions to the ENDF point covariances coming from the different NI-type sub-subsections. Changing the order of summation, we obtain

$$\mathrm{cov}(x_I, y_J) \;=\; \sum_n \mathrm{cov}(x_I, y_J)_n \,, \tag{28}$$

where

$$\mathrm{cov}(x_I, y_J)_n \;=\; \sum_{\substack{i \in I \\ j \in J}} \alpha_{Ii}\, \alpha_{Jj}\;\; \mathrm{cov}(x_i, y_j)_n \,. \tag{29}$$

To evaluate the sum in Eq. (29), we make use of the fact that union groups I and J do not cross any ENDF grid boundaries. Recalling the discussion of NI-type sub-subsections in Section X.C (and excluding for the moment the case where sub-subsections with LB=8 are present), there are only two possibilities for the energy dependence of the covariance between ENDF grid points; thus, over the limits of

the sum, either $\mathrm{cov}(x_i, y_j)_n$ is independent of i and j (if LB=0) or $\mathrm{rcov}(x_i, y_j)_n$ is independent of i and j (if $1 \leq$ LB ≤ 6). We consider first the constant-absolute-covariance case, LB=0. Eq. (29) can then be rewritten as follows:

$$\mathrm{cov}(x_I, y_J)_n = \mathrm{cov}(x,y)_n \sum_{\substack{i\in I \\ j\in J}} \alpha_{Ii}\,\alpha_{Jj} = \mathrm{cov}(x,y)_n \left(\sum_{i\in I} \alpha_{Ii}\right)\left(\sum_{j\in J} \alpha_{Jj}\right) . \tag{30}$$

Invoking Eqs. (24) and (26), we obtain

$$\mathrm{cov}(x_I, y_J)_n = \mathrm{cov}(x,y)_n \qquad (\mathrm{LB} = 0) . \tag{31}$$

For LB-values ranging from 1 to 6, the relative covariance is constant over each union group, so we rewrite Eq. (29) in the form

$$\begin{aligned} \mathrm{cov}(x_I, y_J)_n &= \sum_{\substack{i\in I \\ j\in J}} \alpha_{Ii}\,\alpha_{Jj}\,x_i\,x_j\,\mathrm{rcov}(x_i, x_j)_n \\ &= \mathrm{rcov}(x,y)_n \sum_{\substack{i\in I \\ j\in J}} (\alpha_{Ii}\,x_i)\,(\alpha_{Jj}\,y_j) \\ &= \mathrm{rcov}(x,y)_n \left(\sum_{i\in I} \alpha_{Ii}\,x_i\right)\left(\sum_{j\in J} \alpha_{Jj}\,y_j\right) . \end{aligned} \tag{32}$$

Substituting here from Eqs. (20) and (25), we obtain

$$\mathrm{cov}(x_I, y_J)_n = x_I\,y_J\,\mathrm{rcov}(x,y)_n \qquad (1 \leq \mathrm{LB} \leq 6) . \tag{33}$$

The final union-group multigroup covariance is obtained by inserting these results, Eqs. (31) and (33), back into Eq. (28)

$$\mathrm{cov}(x_I, y_J) = \sum_{\mathrm{n(LB=0)}} \mathrm{cov}(x,y)_n + \sum_{\mathrm{n(LB=1,6)}} x_I\,y_J\,\mathrm{rcov}(x,y)_n , \tag{34}$$

where the first sum runs over all sub-subsections with LB=0 and the second runs over all sub-subsections with LB=1 through 6. The quantities $\mathrm{cov}(x,y)_n$ and $\mathrm{rcov}(x,y)_n$ here are simply the point-energy covariances from the ENDF covariance file, as described in Eqs. 12 through 16. Equation (34) then is the basic equation used in SUBROUTINE COVCAL to calculate the desired union-group covariances.

The final step, if sub-subsections with LB=8 are present, is to increment the diagonal elements (variances) as follows:

$$\mathrm{cov}(x_I, x_I) = \mathrm{cov}(x_I, x_I) + F_k\,(E_{k+1} - E_k)/\Delta E_I , \tag{35}$$

where k indexes the range of the LB=8 energy grid that includes union group I.

F. Basic Strategy for Collapse to the User Grid

The union-group fluxes ϕ_I are used, in SUBROUTINE SIGC, to collapse the union-group cross sections to the coarser user grid. Changing notation slightly, let us denote by $x_I(a)$ the cross section in union group I for reaction a, and similarly let $X_K(a)$ be the cross section in user group K for the same reaction. In complete analogy with Eq. (20)

$$X_K(a) = \frac{\sum_{I \in K} \phi_I \, x_I(a)}{\phi_K} = \sum_{I \in K} A_{KI} \, x_I(a) \, , \tag{36}$$

where

$$\phi_K = \sum_{I \in K} \phi_I \, , \tag{37}$$

and

$$A_{KI} = \frac{\phi_I}{\phi_K} \, . \tag{38}$$

Applying the propagation-of-errors formula again gives

$$\mathrm{cov}\,[X_K(a), X_L(b)] = \sum_{\substack{I \in K \\ J \in L}} A_{KI} \, A_{LJ} \, \mathrm{cov}[x_I(a), x_J(b)] \, . \tag{39}$$

An alternative expression, obtained by simply rearranging coefficients, is

$$\mathrm{cov}\,[X_K(a), X_L(b)] = \frac{1}{\phi_K \phi_L} \sum_{\substack{I \in K \\ J \in L}} T_{IJ}(a,b) \, , \tag{40}$$

where

$$T_{IJ}(a,b) = \phi_I \, \phi_J \, \mathrm{cov}[x_I(a), x_J(b)] \, . \tag{41}$$

Eqs. (40) and (41) provide the basic framework in the ERRORR module for the production of multigroup covariances in the coarse user-group structure. Finally, if the user requests it, the absolute covariances from Eq. (40) are converted to relative form,

$$\mathrm{rcov}[X_K(a), X_L(b)] = \frac{\mathrm{cov}[X_K(a), X_L(b)]}{X_K(a) \, X_L(b)} \, . \tag{42}$$

G. Group-Collapse Strategy for Data Derived by Summation

The procedure described in the previous section must be modified if, in some energy range, either reaction a or b is a "derived" quantity in the sense of Eq. (17) or Eq. (18). In such a case, one cannot apply Eq. (34) directly to the calculation of $\mathrm{cov}[x_I(a), x_J(b)]$, because the covariances on the right-hand side of Eq. (34) will be missing from the ENDF file in the affected energy region.

We consider first the case in which some cross sections are evaluated ("derived") by simply summing other evaluated cross sections, as in Eq. (17). Equation (35) can then be rewritten as

$$X_K(a) = \sum_{I \in K} A_{KI} \left\{ \sum_{\text{all c}} C_I(a,c)\, x_I(c) \right\} , \tag{43}$$

where the quantity in curly brackets is the value of the derived cross section "a" in union group I, as reconstructed from the directly evaluated cross sections $x_I(c)$. The derivation coefficients $C_I(a,c)$ depend on the union-group index I, because the evaluator is permitted to employ different derivation strategies in different energy ranges in order to simplify and shorten the covariance files.

To expand on this last point, suppose there are only three reactions, and suppose that the cross sections $x(1)$ and $x(2)$ rigorously sum to $x(3)$ at all energies. Further suppose that within the energy range covered by the first union group, the cross-section evaluator has used this logical connection in order to "derive" $x(3)$, that is to evaluate $x(3)$, by summing the existing evaluations for $x(1)$ and $x(2)$,

$$x_1(3) = x_1(1) + x_1(2) . \tag{44}$$

Further suppose that in the range of union group 2, $x(1)$ is derived by employing the same logical connection, but in a different way,

$$x_2(1) = x_2(3) - x_2(2) , \tag{45}$$

and in union group 3, all three reactions are directly evaluated.

In this example $C_I(a,c)$ has the following values:

$$\begin{array}{lll} C_1(1,1) = 1 & C_1(1,2) = 0 & C_1(1,3) = 0 \\ C_1(2,1) = 0 & C_1(2,2) = 1 & C_1(2,3) = 0 \\ C_1(3,1) = 1 & C_1(3,2) = 1 & C_1(3,3) = 0 \end{array}$$

$$\begin{array}{lll} C_2(1,1) = 0 & C_2(1,2) = -1 & C_2(1,3) = 1 \\ C_2(2,1) = 0 & C_2(2,2) = 1 & C_2(2,3) = 0 \\ C_2(3,1) = 0 & C_2(3,2) = 0 & C_3(3,3) = 1 \end{array}$$

$$\begin{array}{lll} C_3(1,1) = 1 & C_3(1,2) = 0 & C_3(1,3) = 0 \\ C_3(2,1) = 0 & C_3(2,2) = 1 & C_3(2,3) = 0 \\ C_3(3,1) = 0 & C_3(3,2) = 0 & C_3(3,3) = 1 \end{array}$$

Note that in all cases we have formally considered the evaluated cross sections to be derived from themselves, $C_I(a,c)|_{\text{eval}=\delta_{ac}}$. This device allows us to use Eq. (42) for all reactions, regardless of whether they are derived or evaluated. Note that, in every case where reaction c is derived in group I, $C_I(a,c)=0$. (See boxed submatrices.) This null "sensitivity coefficient" is important, because it means that Eq. (42) remains a linear relation involving only dependent quantities with known covariances (the evaluated subset of the union-group cross sections) on the right. This allows us once again to use the propagation-of-errors formula to obtain the desired user-group covariances.

$$\text{cov}[X_K,(a),X_L(b)] = \sum_{\substack{\text{all c} \\ \text{all d}}} \sum_{\substack{I\in K \\ J\in L}} A_{KI}\, C_I(a,c)\, A_{LJ}\, C_J(b,d)\, \text{cov}[x_I(c),x_J(d)]\,, \tag{46}$$

or, alternatively,

$$\text{cov}[X_K,(a),X_L(b)] = \frac{1}{\phi_K \phi_L} \sum_{\substack{\text{all c} \\ \text{all d}}} \sum_{\substack{I\in K \\ J\in L}} C_I(a,c)\, C_J(b,d)\, T_{IJ}(c,d)\,, \tag{47}$$

where

$$T_{IJ}(c,d) = \phi_I\, \phi_J\, \text{cov}[x_I(c),x_J(d)]\,. \tag{48}$$

In SUBROUTINE COVCAL, the flux-covariance product T_{IJ} is calculated for all evaluated reaction pairs and written to a "scratch" binary disk file, unit 11. (For diagnostic purposes, it is possible to change this to a formatted scratch file by manually resetting the variable IMODE to +1 in the main program.) In the subsequent collapse to the user's group structure in SUBROUTINE COVOUT, the user-group

covariances $\mathrm{cov}[X_K(a), X_L(b)]$ for one specific a-b reaction pair, for all "MT1" energy groups L, and for as large a range of "MT" energy groups K as possible, are calculated simultaneously in memory.

In COVOUT, the disk file (unit 11) is rewound and read completely through, once in each range of MT groups for each a-b reaction pair, to access the union-group covariances for the evaluated reactions. As the union-group data pass through memory, contributions to the user-group covariances are accumulated using Eq. (46). To minimize the number of disk reads and the amount of running time, the dimension of the array in COMMON/ESTORE/ (and the value of the associated variable NAMAX) should be set as large as possible.

In the fairly common situation where cross sections $x(a)$ and $x(b)$ are directly evaluated over the whole energy range of the data file,

$$C_I(a,c) = \delta_{ac}, \text{ for all } I, \tag{49}$$

and

$$C_J(b,d) = \delta_{bd}, \text{ for all } J. \tag{50}$$

In this case, Eq. (46) can be greatly simplified.

$$\begin{aligned} \mathrm{cov}[X_K(a), X_L(b)] &= \frac{1}{\phi_K \phi_L} \sum_{\substack{\text{all c} \\ \text{all d}}} \sum_{\substack{I \in K \\ J \in L}} \delta_{ac}\, \delta_{bd}\, T_{IJ}(c,d) \\ &= \frac{1}{\phi_K \phi_L} \sum_{\substack{I \in K \\ J \in L}} T_{IJ}(a,b)\,. \end{aligned} \tag{51}$$

This result, which is identical to Eq. (39), suggests a shortcut calculational path, which is followed in SUBROUTINE COVOUT whenever both reactions are directly evaluated. In preparation for this situation, a second copy of the COVCAL binary output file is generated when COVOUT is first called. The second copy, on unit 12, is read only in the trivial derivation cases just described (flagged in the code by setting ISD=1). Unit 12 is not rewound before processing a given reaction pair a-b, since earlier reaction pairs on the file do not contribute to the sum in Eq. (46) in these cases. Reading of the second copy stops after the union-group covariances for reaction pair a-b are found, because the later reaction pairs do not contribute either.

H. Processing of Data Derived from Ratio Measurements

The treatment of implicit covariances that arise from ratio measurements, as in Eq. (18), is totally different from the treatment of data derived by summation, Eq. (17), discussed in the previous section. Before discussing the processing details, it is helpful first to review the subject of ratio evaluations generally. Returning to the "x-y" notation of Section X.B, let $x(E_x)$ be the value of the cross section for reaction x at energy E_x, and $y(E_y)$ the cross section for reaction y at energy E_y. In some energy region (L_x, H_x), suppose that the best knowledge of $x(E_x)$ is obtained through the application of a measured ratio, $u(E_x)$:

$$x(E_x) = u(E_x)\, z(E_x), \text{ if } L_x \leq E_x \leq H_x\ , \tag{52}$$

where reaction z is some well-known cross section, possibly one of the official ENDF/B standard cross sections. Similarly, suppose y is derived from the same standard, over a possibly different energy range:

$$y(E_y) = v(E_y)\, z(E_y), \text{ if } L_y \leq E_y \leq H_y\ . \tag{53}$$

By performing a first-order Taylor-series expansion and then applying the formula for propagation of errors, one can obtain an expression for the contribution to the relative covariance $\mathrm{rcov}[x(E_x), y(E_y)]$ that is attributable to these ratio measurements. In the usual case, where the ratios u and v are only weakly correlated with the standard cross section z, the result is quite simple:

$$\mathrm{rcov}[x(E_x), y(E_y)]_{\mathrm{ratio}} = \mathrm{rcov}[u(E_x), v(E_y)] + \mathrm{rcov}[z(E_x), z(E_y)]\ , \tag{54}$$

if $L_x \leq E_x \leq H_x$ and $L_y \leq E_y \leq H_y$, and

$$\mathrm{rcov}[x(E_x), y(E_y)]_{\mathrm{ratio}} = 0 \text{ otherwise.} \tag{55}$$

Thus, in this fairly common evaluation situation, the covariance separates naturally into a part involving only the measured ratios and a part involving only the standard. Because the second contribution, cov(MT_z, MT_z), can be read directly from the NI-type sub-subsections in evaluation for the standard, it is not included explicitly in the ENDF subsections for the derived quantities, cov(MT_x, MT_x) or cov(MT_y, MT_y). Instead, the existence of this additional contribution to the covariance is signalled by the presence of an NC-type sub-subsection, as discussed in Section X.C.

The strategy adopted for processing this information in SUBROUTINE STAND is to load the NI-type sub-subsections from the evaluation of the standard into the same storage array that is used to store NI-type covariances from the evaluation for reaction x. From that point on, the data from the standard are handled just as if they had come from the evaluation for x, but with one exception. As indicated by Eq. (55), the covariance contribution $\mathrm{cov}[z(E_x), z(E_y)]$ is not added into the total covariance matrix for the current reaction pair if either E_x or E_y lies outside the corresponding energy "window" (L_x, H_x) or (L_y, H_y), respectively.

As discussed in Section X.C, in addition to identifying the standard reaction, the NC-type sub-subsection contains a control parameter LTY and two energies EL and EH whose significance depends on LTY. LTY is used to identify particular evaluation scenarios: reaction x is the same as reaction y (or at least they are derived from z over the same energy range) (LTY=1); y is identical to the standard (LTY=2); or x is identical to the standard (LTY=3). A fourth possibility, namely, that x, y, and z are entirely distinct, cannot presently be treated with a single NC-type sub-subsection; that is, there is no LTY value defined for this case. However, as discussed later, it is still possible to process covariances for this situation by combining information from sub-subsections in two different evaluations. For convenience, we shall refer to this fourth case as LTY=4.

The interrelationship of LTY, EL, EH, and the windows (L_x, H_x) and (L_y, H_y) used in ERRORR for the "zeroing-out" operation of Eq. (55), is summarized below:

$$
\begin{aligned}
&\mathrm{LTY} = 1 \\
&\qquad (L_x, H_x) = (L_y, H_y) = (\mathrm{EL}, \mathrm{EH}) \\
&\mathrm{LTY} = 2 \\
&\qquad (L_x, H_x) = (\mathrm{EL}, \mathrm{EH}) \\
&\qquad (L_y, H_y) = (10^{-5}\ \mathrm{eV},\ 20\ \mathrm{MeV}) \\
&\mathrm{LTY} = 3 \\
&\qquad (L_x, H_x) = (10^{-5}\ \mathrm{eV},\ 20\ \mathrm{MeV}) \\
&\qquad (L_y, H_y) = (\mathrm{EL}, \mathrm{EH}) \\
&\mathrm{LTY} = 4 \\
&\qquad (L_x, H_x) = (\mathrm{EL}, \mathrm{EH}) \\
&\qquad (L_y, H_y) = (\mathrm{EL}', \mathrm{EH}')\ .
\end{aligned}
\tag{56}
$$

If the user requests covariance data $\mathrm{cov}(x, y)$, where x and y are different and both

Table 1: Covariance Matrices Affected by Ratio Measurements In ENDF/B-V

	^{10}B (n,α)	^{238}U (n,γ)	^{235}U (n,f)	^{239}Pu (n,f)	^{239}Pu (n,γ)	^{241}Am (n,f)	^{242}Pu (n,f)
^{10}B(n,α)	*	3					
^{238}U(n,γ)	2	1					
^{235}U(n,f)			*	3	3	3	3
^{239}Pu(n,f)			2	1	1	4	4
^{239}Pu(n,γ)			2	1	1	4	4
^{241}Am(n,f)			2	4	4	1	4
^{242}Pu(n,f)			2	4	4	4	1

* = standard
Integer = LTY value (see text)

are distinct from the standard (LTY=4), the ERRORR module obtains (EL,EH) from the LTY=2 subsection in the evaluation for x that "points" to z. Then, the covariance file for z is read to obtain both (a) the explicit covariances cov(z,z) and (b) the second energy window (EL', EH'), the latter being found in the LTY=3 sub-subsection that points back to y.

Table 1 lists all ENDF/B-V reactions that contain ratio-to-standard covariance data. The symbols entered in the reaction-by-reaction matrix indicate which reactions are referenced as standards (*), and which reaction pairs have implicit nonzero covariances (LTY). ERRORR will produce multigroup covariances for any of the reaction pairs in Table I that are marked with (*) or (LTY). The cross-material covariances (LTY=2, 3, or 4) must be requested individually using the IREAD=2 option (see Section X.K, especially the discussion of Cards 10 and 11). An attempt to process covariances for any of the cases LTY=1 through 4 without supplying a separate ENDF tape containing the needed standard will result in an error stop in SUBROUTINE GRIDD. The error diagnostic, however, will supply the details of the corrective action required. (See general discussion of diagnostic messages in Section X.M.)

As an example of the ratio-data capabilities of ERRORR, Fig. 2 shows the covariances between the fission cross sections of ^{239}Pu (reaction x) and those of the important actinide ^{241}Am (reaction y). This is an LTY=4 case where (L_x, H_x)= (0.2 MeV, 15 MeV) and (L_y, H_y)=(0.2 MeV, 20 MeV). The effect of using two different windows is apparent in the lower right corner of the correlation matrix. The plot itself was produced with the COVR module, which is described in Chapter

XX of this report. The complete NJOY input needed to compute and plot these data is given below at the end of Section X.K.

I. Multigroup Processing of Resonance-Parameter Uncertainties

In some materials, and in certain energy regions, the cross-section uncertainty is dominated by the uncertainty in resolved resonance parameters. One noteworthy example is ^{63}Cu(n,γ)^{64}Cu (ENDF/B-V dosimetry material, **MAT6435**) in the energy range from 10 eV to 15.9 keV, where the entire cross-section uncertainty is represented by means of resonance-parameter uncertainties. The same is true of ^{237}Np(n,f) (**MAT6337**) from 0 to 10 eV and approximately so for the ENDF/B-VI evaluation of ^{23}Na(n,γ) (**MAT1125**) in the neighborhood of the large resonance at 2.81 keV.

In ERRORR, the resonance-parameter contribution to the uncertainty in infinite-dilution fission and capture cross sections is included automatically when cross-section covariances are processed. (See the discussion of the input option **MFCOV=33** in Section X.K.) Following a procedure suggested in Ref. 8, the contribution is obtained from the Breit-Wigner formulae for the fission and capture areas of a resonance, A_f and A_γ. By differentiating these formulae with respect to the resonance parameters, one obtains a set of sensitivities. With these sensitivities and the covariance matrix of the parameters from MF32, one can apply the propagation-of-errors formula to obtain the covariances cov(A_γ, A_γ), cov(A_γ, A_f), and cov(A_f, A_f). In **SUBROUTINE RESCON**, these results are added directly into the user-group cross-section covariances calculated from MF33.

The resonance contribution is properly weighted with the isotopic abundance and the ratio of the weight function at the resonance to the average weight in the group. It is assumed, however, that the area of a resonance lies entirely within the group that contains the resonance energy E_r. Because of this assumption, and because ENDF-5 and the **LCOMP=0** option of ENDF-6 provide no correlations between parameters of different resonances, the calculated resonance-parameter contribution affects only the diagonal elements of the affected matrices, cov[$X_K(a), X_K(b)$].

With the inclusion of resonance-parameter covariances, the uncertainty in the capture cross section of ^{63}Cu, for a group that contains the large 577-eV resonance, is 3.0%, rather than zero, as it would be if the MF32 contribution were omitted.

With regard to the above-mentioned example of the important reactor coolant ^{23}Na (**MAT1125**), in an energy group extending from 1.235 to 3.350 keV, the addi-

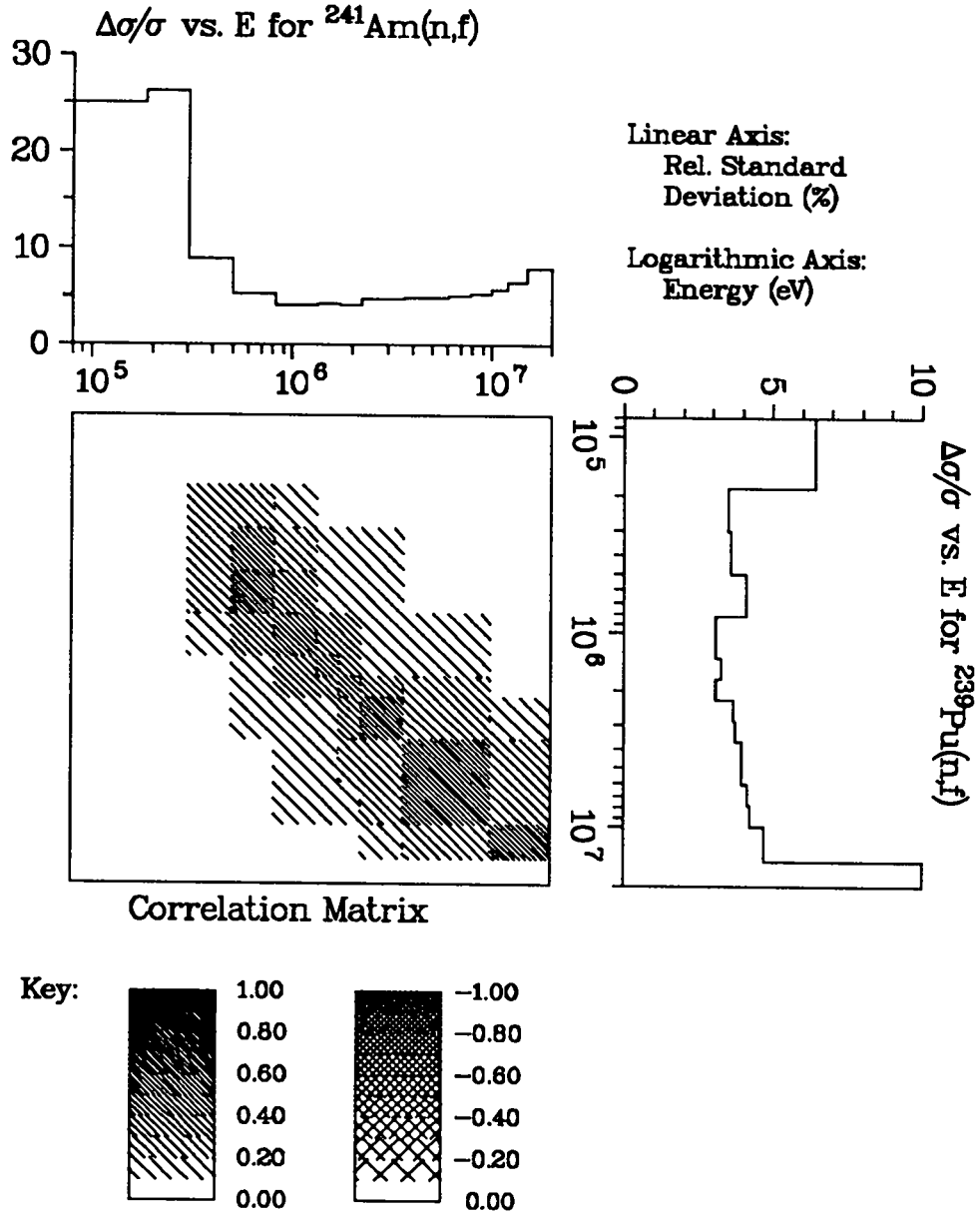


Figure 2: Covariance data for ^{239}Pu(n,f) with ^{241}Am(n,f).

tional uncertainty coming from resonance parameters raises the uncertainty of the capture cross section from 10% to 19%.

J. Processing of Lumped-Partial Covariances

The lumped-partial covariance format, allows the evaluator to specify a group of nuclear reactions and to give the uncertainty only in the sum of the cross sections for that group of reactions. One can, for example, replace 30 or 40 discrete-level inelastic cross sections with 5 or 6 lumped cross sections when constructing the covariance files. Because the volume of the covariance data varies, in general, as the square of the number of reactions, this lumping can greatly reduce the size of the files.

The first ENDF evaluation to employ the lumped-partial format was P. Young's evaluation[9, 10] for ^{7}Li (ENDF/B-V, Rev. 2). All covariance data for this evaluation have been successfully processed into multigroup form using ERRORR. The covariances for MT854 (a single real level with an excitation energy of 4.63 MeV) with MT855 (6 lumped pseudo-levels, with excitation energies ranging from 4.75 MeV to 6.75 MeV) have been plotted in Fig. 3. The large negative correlations along the diagonal result from the fact that, below 10 MeV, these inelastic reactions are the major contributors to the relatively well-known tritium production cross section. An upward variation in one reaction at a given energy must be accompanied by a downward change in the other reaction. As shown in the plot, the magnitude of this negative correlation diminishes at higher energies, as other reactions begin to contribute significantly to the tritium-production cross section. Plots of this type, prepared using ERRORR and COVR, have proved to be useful tools in the validation of the covariance files of new evaluations.[11]

K. Input Instructions and Sample Input for ERRORR

As an aid to discussions of the user input to ERRORR, we list below the input instructions that appear as comment cards at the beginning of the current version of this module. Since the code and, hence, the instructions change from time to time, it is always advisable to consult the comment-card instructions contained in the version of the code actually being used and not to rely on the instructions published in any document, including this one. Note that the word "tape" is used in the instructions to refer to an input or output file, regardless of the actual storage medium.

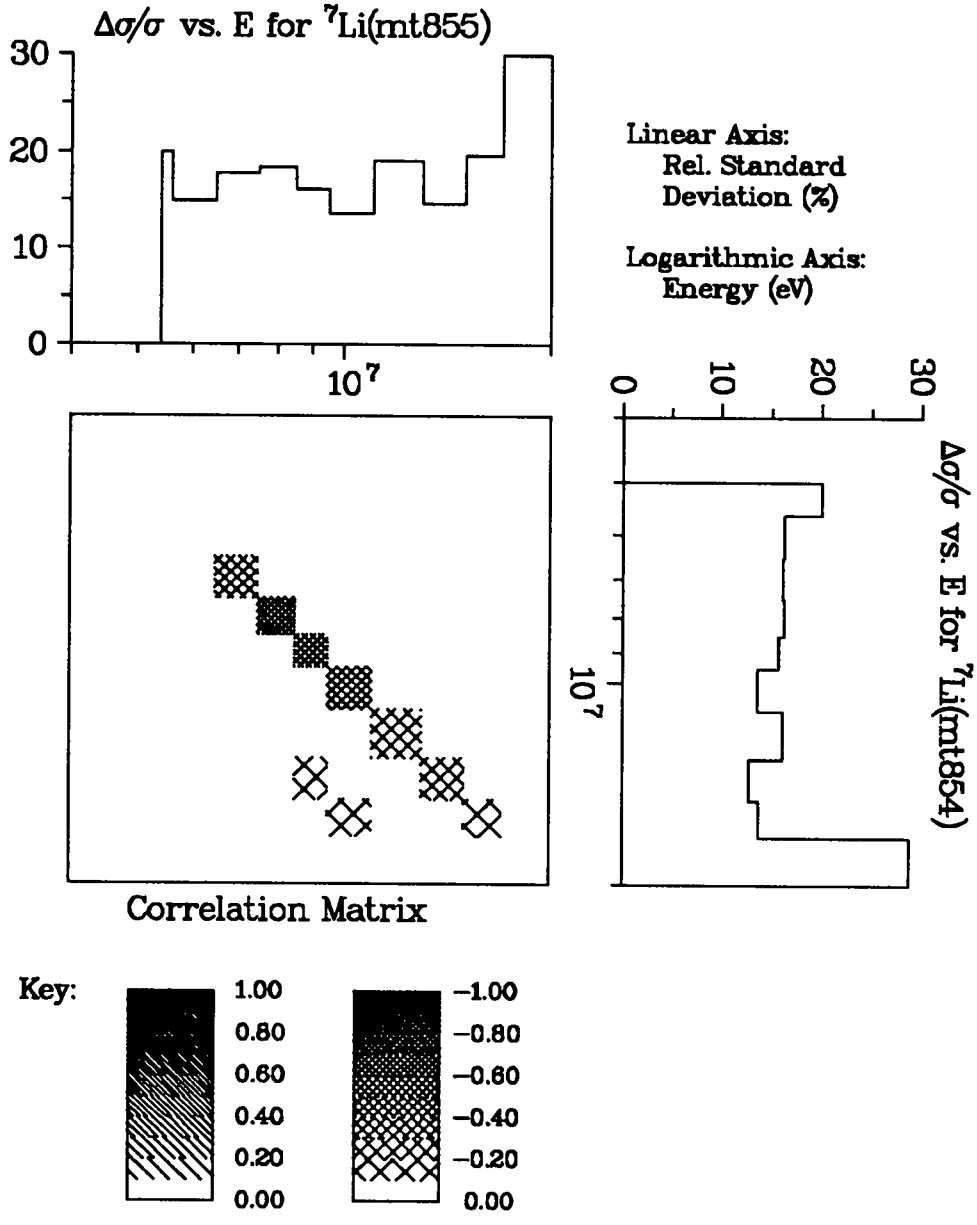


Figure 3: Covariance data for ^{7}Li (MT854) with ^{7}Li (MT855).

```
*---INPUT SPECIFICATIONS (FREE FORMAT)-------------------------*
*                                                              *
*  CARD 1                                                      *
*    NENDF   UNIT FOR ENDF/B TAPE                              *
*    NPEND   UNIT FOR PENDF TAPE                               *
*    NGOUT   UNIT FOR INPUT GROUP XSEC (GENDF) TAPE            *
*            (IF ZERO, GROUP XSECS WILL BE CALCULATED)         *
*            (IF IREAD EQ 2 OR IF MFCOV EQ 31 (SEE CARD 7),    *
*            NGOUT CANNOT BE ZERO)                             *
*            (DEFAULT=0)                                       *
*    NOUT    UNIT FOR OUTPUT COVARIANCE TAPE (DEFAULT=0)       *
*    NIN     UNIT FOR INPUT COVARIANCE TAPE (DEFAULT=0)        *
*            (NIN AND NOUT MUST BE BOTH CODED OR BOTH BINARY)  *
*    NSTAN   UNIT FOR RATIO-TO-STANDARD TAPE (DEFAULT=0)       *
*  CARD 2                                                      *
*    MATD    MATERIAL TO BE PROCESSED                          *
*    IGN     NEUTRON GROUP OPTION                              *
*            (IGN DEFINITION SAME AS GROUPR, EXCEPT IGN=19,    *
*            WHICH MEANS READ IN AN ENERGY GRID, AS IN IGN=1,  *
*            AND SUPPLEMENT THIS WITH THE ENDF COVARIANCE GRID *
*            WITHIN THE RANGE OF THE USER-SPECIFIED ENERGIES)  *
*            (DEFAULT=1)                                       *
*    IPRINT  PRINT OPTION (0/1=MINIMUM/MAXIMUM) (DEFAULT=1)    *
*    IRELCO  COVARIANCE FORM (0/1=ABSOLUTE/RELATIVE) (DEFAULT=1) *
*  CARD 3    (OMIT IF NGOUT.NE.0)                              *
*    IWT     WEIGHT FUNCTION OPTION                            *
*    MPRINT  PRINT OPTION FOR GROUP AVERAGING (0=MIN., 1=MAX.) *
*    TEMPIN  TEMPERATURE (DEFAULT=300)                         *
*                                                              *
*---FOR ENDF/B VERSION 4 (IVERF=4) ONLY-----------------------*
*                                                              *
*  CARD 4                                                      *
*    NEK     NUMBER OF DERIVED XSEC ENERGY RANGES              *
*            (IF ZERO, ALL XSECS ARE INDEPENDENT)              *
*  CARD 5    (OMIT IF NEK=0)                                   *
*    EK      NEK+1 DERIVED XSEC ENERGY BOUNDS                  *
*  CARD 6    (OMIT IF NEK=0)                                   *
*    AKXY    DERIVED CROSS SECTION COEFFICIENTS, ONE ROW/LINE  *
*                                                              *
*---FOR ENDF/B VERSION 5 (IVERF=5) ONLY-----------------------*
*                                                              *
*  CARD 7                                                      *
*    IREAD   0/1/2=PROGRAM CALCULATED MTS/INPUT MTS AND EKS/   *
*            CALCULATED MTS PLUS EXTRA MAT1-MT1 PAIRS FROM INPUT *
*            (DEFAULT=0)                                       *
*    MFCOV   ENDF COVARIANCE FILE (31, 32, OR 33) TO BE        *
*            PROCESSED (DEFAULT=33).                           *
*            NOTE--CONTRIBUTION TO GROUP CROSS SECTION         *
*            COVARIANCES FROM RESONANCE-PARAMETER UNCERTAINTIES *
*            (MF=32) IS INCLUDED WHEN MFCOV=33 IS SPECIFIED.   *
*                                                              *
*  FOLLOWING CARDS ONLY IF IREAD EQ 1                          *
```

```
*  CARD 8                                                          *
*    NMT     NO. MTS TO BE PROCESSED                               *
*    NEK     NO. DERIVED CROSS SECTION ENERGY RANGES               *
*            (IF ZERO, ALL XSECS ARE INDEPENDENT)                  *
*  CARD 8A                                                         *
*    MTS     NMT MTS                                               *
*  CARD 8B   (OMIT IF NEK=0)                                       *
*    EK      NEK+1 DERIVED CROSS SECTION ENERGY BOUNDS             *
*  CARD 9    (OMIT IF NEK=0)                                       *
*    AKXY    DERIVED CROSS SECTION COEFFICIENTS, ONE ROW/LINE      *
*                                                                  *
*  FOLLOWING CARD ONLY IF IREAD EQ 2                               *
*  CARD 10                                                         *
*    MAT1    CROSS-MATERIAL REACTION TO BE ADDED TO                *
*    MT1         COVARIANCE REACTION LIST.                         *
*            REPEAT FOR ALL MAT1-MT1 PAIRS DESIRED                 *
*            TERMINATE WITH MAT1=0.                                *
*                                                                  *
*  FOLLOWING CARD ONLY IF NSTAN NE 0                               *
*  CARD 11                                                         *
*    MATB    STANDARDS REACTION REFERENCED                         *
*    MTB         IN MATD.                                          *
*    MATC    STANDARDS REACTION TO BE                              *
*    MTC         USED INSTEAD.                                     *
*            REPEAT FOR ALL STANDARD REACTIONS TO BE REDEFINED.    *
*            TERMINATE WITH MATB=0.                                *
*  NOTE.  IF MATB(1) AND MTB(1) ARE NEGATIVE, THEN MATC(1) AND     *
*    MTC(1) IDENTIFY A THIRD REACTION, CORRELATED WITH MATD THRU   *
*    THE USE OF THE SAME STANDARD.  COVARIANCES OF ALL REACTIONS   *
*    IN MATD (WHICH REFERENCE THE STANDARD) WITH THE REACTION      *
*    MATC(1)-MTC(1) WILL BE PRODUCED.  THE STANDARD REACTION       *
*    MUST BE IDENTIFIED ON CARD 10 AND REPEATED AS THE NEGATIVE    *
*    ENTRIES ON CARD 11.  THE GROUP XSEC TAPE NGOUT MUST INCLUDE   *
*    ALL COVARIANCE REACTIONS IN MATD, PLUS MATC(1)-MTC(1).        *
*----------------------------------------------------------------*
*                                                                  *
*  CARD 12A (FOR IGN EQ 1 OR IGN EQ 19)                            *
*    NGN     NUMBER OF GROUPS                                      *
*  CARD 12B                                                        *
*    EGN     NGN+1 GROUP BOUNDS (EV)                               *
*  CARD 13 (FOR IWT EQ 1 ONLY)                                     *
*    WGHT    WEIGHT FUNCTION AS A TAB1 RECORD                      *
*                                                                  *
********************************************************************
```

Further details on many of these input parameters are provided in the text below. Actual NJOY input for a few example problems is provided for further clarification. Complete input and output for another example problem are given in Ref. 12.

NPEND, NGOUT ... The user must supply either a PENDF tape on unit NPEND or a GENDF tape on unit NGOUT. If present, the PENDF tape should contain pointwise (resonance-reconstructed, but not multigrouped) data and is normally produced with the RECONR module. If the resonance region is of no interest, a simple copy of the ENDF tape will suffice for this purpose. The GENDF tape, if present, should contain multigroup cross sections and/or $\bar{\nu}$ values, produced by the GROUPR module, for all reactions for which multigroup covariances are needed. Group cross sections need not be in the same group structure as the requested output covariances, although if the input group structure is much coarser than the output structure, rather crude approximations will be made in deriving an effective set of fine-group cross sections and fluxes from the coarse input data. Certain types of covariance calculations related to fission data or the computation of cross-material covariances require the user to supply a GENDF tape on NGOUT (that is, NPEND must be zero).

NIN, NOUT ... Input and output covariance tapes in the user's group structure may either be both formatted or both binary. Although ERRORR lacks an explicit multimaterial loop, the same effect can be achieved by executing a series of ERRORR runs, with the output of one run becoming the input of the next. Only two unit numbers need be employed, with the data going back and forth between them until the multimaterial library is complete. To save time, binary files should be used for this purpose. MODER can be used to convert the final output tape from binary to formatted form, if desired.

NSTAN ... This tape is needed if covariances are requested for a reaction pair that is related by ratio measurements, as discussed in Section X.H. The ENDF subsection for such a pair will contain an NC-type sub-subsection with LTY=1, 2, or 3. The specific pairs in ENDF/B-V that require a standards tape are tabulated in Table 1 (Section X.H). For other evaluations, one can assume NSTAN is not needed. The code will stop with a clear error message if this assumption proves incorrect.

IREAD ... For Versions 5 and 6, the list of reactions for which covariances are produced is constructed in one of several ways, depending on the value of IREAD. If IREAD=0, which is the generally recommended choice, the reaction list is assumed to be identical to the list of sections contained in the ENDF/B covariance file (see MFCOV below). Although IREAD=0 is the most convenient option for constructing the list of covariance reactions, it leads to a problem if some covariance reactions have thresholds above the highest group boundary of the user's structure. These reactions will have zero cross sections in all groups and thus will be omitted from

NGOUT. To avoid this difficulty, ERRORR automatically resets the user's highest group boundary to 20 MeV whenever IREAD=0. By judicious choice of weighting functions, one can minimize the effect of this resetting on the other cross sections and their covariances.

If IREAD=1, only the covariance reactions specifically named by the user will be included in the reaction list. For certain applications, this option can save considerable execution time. However, to take advantage of this option, the user must examine the ENDF evaluation visually to determine which reactions are derived from other reactions in each energy region. As discussed in Section X.C, this information is contained in NC-type sub-subsections with LTY=0. Incorrect results will be obtained if one requests processing for a given reaction without processing all of the reactions from which the given reaction is derived in the energy region spanned by the output group structure. The inclusion of those reactions may require, in turn, additional inclusions as well. In addition, the user must extract from the file (and enter into the input) the appropriate derivation coefficients and the energy ranges over which they apply. An example of the required input is presented below in the discussion of the parameters NMT, NEK, MTS, EK, and AKXY.

If IREAD=0 or 1, a complete set of covariance matrices is written. That is, covariance matrices are produced for every reaction-pair combination that can be formed from the given reaction list. For those reaction pairs where the evaluator has not specified the covariances, the output matrix contains only zeros.

In the evaluation containing cross-material covariances, subsections involving the other materials will be ignored if IREAD=0 or 1. These subsections can be selectively processed by specifying IREAD=2. If IREAD=2, the reaction list is initially constructed in the same way as for IREAD=0. This list is then supplemented with a list of extra reactions (MAT1$_k$, MT1$_k$) (where (MAT1$_k \neq$ MATD), specified by the user. The output in this case contains matrices for all of the (MATD, MT$_i$; MATD,,MT$_j$) combinations, as before, plus matrices for all (MATD, MT$_i$; MAT1$_k$, MT1$_k$) combinations. However, covariances among the extra reactions, for example (MAT1$_k$, MT1$_k$; MAT1$_m$,,MT1$_m$), are not computed. Input for an example problem that illustrates the use of IREAD=2 is given below, in the discussion of MFCOV. That discussion also includes an illustration (Table 2) of the set of MAT/MT combinations that are processed when IREAD=2.

MFCOV ... This parameter is used to specify whether covariances of fission $\overline{\nu}$ (MFCOV=31) or cross sections (MFCOV=33) are needed. If MFCOV=31, it is necessary to supply multigrouped cross sections and $\overline{\nu}$ values on a GENDF tape (that is, NPEND

must be 0); the sample problem below shows the calculational sequence required. MFCOV=32 is reserved for a planned future capability to compute uncertainties in self-shielded cross sections. If MFCOV=33, the contribution of the uncertainty in individual resonance parameters (File 32) is included in the calculated uncertainty in infinite-dilution group cross sections. Given below is the complete NJOY input for a calculation of a multigroup $\overline{\nu}$ covariance library for ^{238}U, including five cross-material reactions.

```
0
5
*MODER*/ MOUNT ENDF TAPES 515, 516, and 555 ON UNITS 20, 21, AND 22.
1 -23
*ENDF/B-V NUBAR COVARIANCE MATERIALS*/
20 1380
20 1381
21 1390
22 1395
22 1398
20 1399
0/
*MODER*/ COPY ENDF FOR USE AS A PENDF.
-23 -24
*GROUPR*/ PREPARE GENDF WITH MULTIGROUPED NUBARS.
-23 -24 0 25
1380 3 0 3 0 1 1 0
*BIG3 + 2 NUBAR*/
0.
1.E10
3 452 *TOTAL NUBAR*/
0/
1381
3 452 *TOTAL NUBAR*/
0/
1390
3 452 *TOTAL NUBAR*/
0/
1395
3 452*TOTAL NUBAR*/
0/
1398
3 452 *TOTAL NUBAR*/
3 455 *DELAYED NUBAR*/
3 456 *PROMPT NUBAR*/
0/
1399
3 452 *TOTAL NUBAR*/
0/
0/
*ERRORR*/ PREPARE MULTIGROUP NUBAR COVARIANCE LIBRARY.
-23 0 25 26/
1398 19 1 1
```

Table 2: Reaction Pairs in ^{238}U Library

	a	b	c	d	e	f	g	h	i
a = (1398,452)	X	X	X	X	X	X	X	X	X
b = (1398,455)		X	X	X	X	X	X	X	X
c = (1398,456)			X	X	X	X	X	X	X
d = (1380,452)				–	–	–	–	–	–
e = (1381,452)					–	–	–	–	–
f = (1390,452)						–	–	–	–
g = (1395,452)							–	–	–
h = (1398,452)								–	–
i = (1399,452)									–

```
2 31
1380 452
1381 452
1390 452
1395 452
1399 452
0/
1
1.E7 1.7E7
*STOP*
```

Note the use of the MODER module to prepare a special ENDF tape containing selected evaluations from other ENDF tapes. Because only high-energy $\overline{\nu}$ data are requested (10–17 MeV), it is unnecessary to use RECONR to prepare a PENDF tape for GROUPR. A copy of the ENDF tape is used instead.

Following the prescription given in the discussion of the IREAD=2 option, with this particular input, covariance matrices will be produced for those reaction pairs marked with an X in Table 2. The remaining non-redundant possibilities (dashes) can be filled in with successive ERRORR runs with MATD=1380, 1381, etc.

NMT, NEK, MTS, EK, AKXY ... For Version 5, if IREAD=1, the user specifies a subset of the evaluator's covariance reactions (that is, a subset of the sections of MFCOV), as the particular set of reactions for which processing is requested. These NMT desired reactions are entered in the array MTS. As mentioned in the discussion of IREAD above, one must also include in the MTS array all reactions from which the desired reactions are derived.

Another requirement of the IREAD=1 option is that the actual derivation coefficients [called **AKXY** in the code, but called $C_I(a,c)$ in the discussion in Section X.G] must be entered in the ERRORR input for **NEK** energy ranges spanning the output group structure. Adjacent ENDF derivation ranges may be merged into a single range if the derivation-coefficient matrix for the **NMT** explicitly requested reactions is the same in each of the adjacent ranges. The ordering of the $C_I(a,c)$ data is as follows: on one line of input the coefficients are specified for a fixed a-value and for all c-values ranging from 1 to **NMT**. One such line is given for each a-value. Finally, there is an outer loop over the **NEK** energy ranges. An example of the input that is required for READ=1 is given below for the case of ENDF/B-V carbon **MAT1306**. It will be necessary to examine File 33 of the evaluation (which is available on the ENDF/B-V standards file, Tape 511), in order to understand the details of this example.

```
0
5
*MODER*
20- 21
*MODER*/COPY ENDF FOR USE AS A PENDF.
-21 -22/
*ERRORR*
-21 -22 0 -23/
1306 3 1/
2 0 0
1 33
7 3
1 2 4 102 103 104 107
1.E-5 2E6 4.812E6 2E7
0 1 0 1 0 0 0
0 1 0 0 0 0 0
0 0 1 0 0 0 0
0 0 0 1 0 0 0
0 0 0 0 1 0 0
0 0 0 0 0 1 0
0 0 0 0 0 0 1
1 0 0 0 0 0 0
0 1 0 0 0 0 0
0 0 1 0 0 0 0
0 0 0 1 0 0 0
0 0 0 0 1 0 0
0 0 0 0 0 1 0
0 0 0 0 0 0 1
1 0 0 0 0 0 0
0 1 0 0 0 0 0
1 -1 0 -1 -1 -1 -1
0 0 0 1 0 0 0
0 0 0 0 1 0 0
0 0 0 0 0 1 0
0 0 0 0 0 0 1
```

```
*STOP*
```

Note that here, even if the user wanted to process only MT2 and MT4, for example, it is nevertheless necessary to include MT1, MT102, MT103, MT104, and MT107, because the IGN=3 group structure extends to 17.0 MeV, and above 4.812 MeV, MT4 is derived from the relation

$$\sigma_4 = \sigma_1 - \sigma_2 - \sigma_{102} - \sigma_{104} - \sigma_{107} \, . \tag{57}$$

See the corresponding $C_I(a,c)$ matrix in the input above (the last 7 lines before STOP).

MATB, MTB, MATC, MTC ... Card 11 provides a capability to remap all references to a given standards reaction (MATB,MTB) appearing in NC-type sub-subsections with LTY=1, 2, or 3 into references to a different reaction (MATC,MTC). (See the discussion of ratio measurements in Section X.H.) This facility is useful if the evaluation (MATB,MTB) is for some reason unavailable. Up to 5 such standards can be redefined. As explained in the note on the input instructions at the beginning of this section, Card 11 is also used [if MATB(1) and MTB(1) are negative] to process covariances between two distinct reactions, both measured relative to a common standards reaction. This was referred to as the LTY=4 case in Section X.H. A separate ERRORR run is required for each requested LTY=4 reaction pair.

This second use of Card 11 is illustrated in the sample NJOY input listed below, which is the input used to generate and plot the data shown in Fig. 2. The input to the COVR module contained in the sample is explained in the following chapter.

```
$ SETUP FOR PLOTTING FIG2 IN ERRORR MANUAL
$ MOUNT T562 ON TAPE30
$ MOUNT T563 ON TAPE40
$ MOUNT T560 ON TAPE50
 0
 5
 MODER
 1 -31
 *U235 FROM T562*/
 30 1395
 0/
 MODER
 1 -21
 *AM241 FROM T560 AND PU239 FROM T563*/
 50 1361/ AM241
 40 1399/ PU239
 0/
```

```
RECONR
-21 -22
*10 PERCENT PENDF FOR AM241 AND PU239*/
1361/
.1/
1399/
.1/
0/
GROUPR
-21 -22 0 -24
1361 3 0 2 0 1 1 0
*30 GROUP XSECS FOR AM241 AND PU239*/
0
1E10
3 1/
3 2/
3 4/
3 16/
3 17/
3 18/
3 102/
0/
1399
3 18/
3 102/
0/
0/
ERRORR
-21 0 -24 -25 0 -31
1361 3 1 1
0 33
0/
ERRORR
-21 0 -24 -28 -25 -31
1399 3 1 1
2 33
1395 18
0/
-1395 -18 1361 18
0/
COVR
-28/
0 0 0 8E4
1 1 0 0 2
1399 18 1361 18
STOP
```

L. ERRORR Output File Specification

The results from ERRORR are written to an output file on unit NOUT, provided that the input parameter NOUT is nonzero. This file contains the user's group

structure, the multigroup cross sections, and either absolute (IRELCO=0) or relative (IRELCO=1) covariances. If NOUT is positive, then a formatted (card-image) file is written, and, if negative, a binary file is written. In either case, the output is written with the standard NJOY I/O utilities, and the result has an ENDF-like structure. As is the case for ENDF, PENDF, and GENDF files, ERRORR output files can be converted from binary to formatted form (and *vice versa*) by using the MODER module. The energy-group ordering of the data written to NOUT follows the ENDF convention. That is, low-energy groups precede high-energy groups.

A small, stand-alone retrieval program, which reads formatted ERRORR output files, reorders the energy groups to high-to-low order, and fills in the implicit zeros to make a full matrix, is listed in Table IV of Ref. 11. A much more powerful retrieval program, which produces libraries for sensitivity codes and publication-quality plots of the data, starting from either binary or formatted ERRORR output, is the COVR module of NJOY. This module is described in the following chapter of this report.

It is occasionally of interest to examine a formatted ERRORR output file visually. For this and other reasons, we describe below the detailed form of the formatted output file. To assist in the description, we give in Table 3 an example of such a file, which was produced during the processing of COVFILS-2 (Ref. 13).

The data in Table 3 are written in the form of standard, 80-column card-images consisting of 10 data fields. As in ENDF, the first 6 fields are 11 characters wide and are used for either floating-point numbers or integers. The seventh, eighth, ninth, and tenth fields are 4, 2, 3, and 5 characters wide, respectively, and are used for integers only. The four digits in Field 7 are the MAT, or material number, of the isotope or element; the two digits in Field 8 are the MF, or file number, which indicates the general type of data (for example, cross-section data *vs.* covariance data); the three digits in Field 9 are the MT, or section number, which indicates the particular nuclear reaction; and finally, the five digits in Field 10 are NSEQ, the card sequence number. As usual, sections are terminated by zeros in the MT field, files by zeros in the MF field, and materials by zeros in the MAT field.

Following an initial tape header record, the second card shown in Table 3 is the first data card for the material MAT1326, which is elemental iron. Note that on this card the number 2.6×10^4 appears in the first field and 55.365 in the second field. These are the "ZA" ($1000 \times Z - A$) and "AWR" (atomic weight relative to the neutron) numbers taken directly from the ENDF file. The fact that ZA is 26 000 (suggesting that $A = 0$) is a convention that means that the data are for the natural element Fe, rather than for a single isotope. The value of -11 appearing

in Field 5 is a flag (read by the MODER module, for example) that identifies the present file format as the ERRORR output format.

Also note that MF=1 and MT=451 on cards NSEQ=1 to 15. This MF-MT combination is used in the ENDF format for descriptive Hollerith information, but it is used here to specify the boundaries of the multigroup structure used for the processed data to follow. On Card 2, note the number 74 in Field 3 and the number 75 in Field 5, which indicate, respectively, the number of energy groups and the number of energy-group boundaries in the multigroup set. The values of the group boundaries, given in eV from low to high energy, follow in Cards 3 through 15. Cards 16 and 17 are section and file terminators, respectively.

Multigroup cross-section data are given in barns on Cards 18 through 631. This file, denoted by MF=3, corresponds to the smooth cross-section file in ENDF. The MT-numbers (Field 9) can be identified with particular nuclear reactions by consulting Appendix B of Ref. 4. Note that cross sections for the lumped-partial reaction MT=851 have been constructed by ERRORR according to the directions given in the ENDF file.

Table 3: Example of ERRORR Output Data File

```
                                                                  1 0  0    0
2.600000+4 5.536500+1          0          0        -11          01326 1451    1
0.000000+0 0.000000+0         74          0         75          01326 1451    2
1.000000-5 7.602200-4 1.239500-2 4.275500-2 8.196800-2 1.523000-11326 1451    3
4.139900-1 8.764200-1 1.125400+0 1.855400+0 3.927900+0 5.043500+01326 1451    4
8.315300+0 1.760300+1 3.726700+1 7.889300+1 1.013000+2 1.670200+21326 1451    5
3.535800+2 7.485200+2 1.584600+3 3.354600+3 7.101700+3 9.118800+31326 1451    6
1.503400+4 2.478800+4 2.605800+4 2.808800+4 3.182800+4 4.086800+41326 1451    7
5.247500+4 6.737900+4 8.651700+4 1.110900+5 1.426400+5 1.831600+51326 1451    8
2.351800+5 3.019700+5 3.877400+5 4.393700+5 4.978700+5 5.641600+51326 1451    9
6.392800+5 7.244000+5 8.208500+5 1.054000+6 1.194300+6 1.353400+61326 1451   10
1.737700+6 1.969100+6 2.231300+6 2.528400+6 2.865000+6 3.246500+61326 1451   11
3.678800+6 4.168600+6 4.723700+6 5.352600+6 6.065300+6 6.872900+61326 1451   12
7.788000+6 8.825000+6 1.000000+7 1.100000+7 1.200000+7 1.300000+71326 1451   13
1.350000+7 1.375000+7 1.394000+7 1.420000+7 1.442000+7 1.464000+71326 1451   14
1.500000+7 1.600000+7 2.000000+7                        1326 1451   15
                                                        1326 1  0   16
                                                        1326 0  0   17
0.000000+0 0.000000+0          0          0         74          01326 3  1   18
3.242538+1 1.638221+1 1.395614+1 1.310400+1 1.266828+1 1.222209+11326 3  1   19
1.192867+1 1.180887+1 1.173992+1 1.164882+1 1.159180+1 1.155850+11326 3  1   20
1.151292+1 1.146973+1 1.143331+1 1.140845+1 1.137811+1 1.103735+11326 3  1   21
1.031032+1 9.276154+0 7.382612+0 7.137478+0 1.602539+1 4.178159+01326 3  1   22
1.482726+0 1.406100+0 3.875233+1 3.832471+1 8.517735+0 5.325881+01326 3  1   23
4.200487+0 6.723474+0 4.481126+0 3.610952+0 3.923129+0 4.868550+01326 3  1   24
2.836503+0 3.245566+0 4.853495+0 3.640536+0 3.205600+0 2.250606+01326 3  1   25
2.732808+0 3.895072+0 2.571093+0 2.501068+0 2.958182+0 2.985432+01326 3  1   26
```

```
2.899251+0 3.213480+0 3.189247+0 3.477117+0 3.389077+0 3.403654+01326 3  1   27
3.625322+0 3.668095+0 3.697471+0 3.674937+0 3.632669+0 3.545578+01326 3  1   28
3.383465+0 3.214864+0 3.053930+0 2.905813+0 2.743816+0 2.674006+01326 3  1   29
2.635890+0 2.613351+0 2.590513+0 2.567519+0 2.547020+0 2.517568+01326 3  1   30
2.457292+0 2.307503+0                                          1326 3  1   31
                                                               1326 3  0   32
0.000000+0 0.000000+0          0          0         74          01326 3  2   33
1.140000+1 1.140000+1 1.139999+1 1.139998+1 1.139996+1 1.139991+11326 3  2   34
1.139978+1 1.139964+1 1.139947+1 1.139900+1 1.139839+1 1.139764+11326 3  2   35
1.139553+1 1.139053+1 1.137995+1 1.136763+1 1.134459+1 1.101228+11326 3  2   36
1.029462+1 9.055728+0 7.376526+0 7.127705+0 1.599799+1 4.168132+01326 3  2   37
1.450895+0 1.368534+0 3.867417+1 3.827977+1 8.473823+0 5.294164+01326 3  2   38
4.169383+0 6.691388+0 4.451224+0 3.586689+0 3.901454+0 4.844768+01326 3  2   39
2.815969+0 3.225118+0 4.830640+0 3.616074+0 3.180014+0 2.223638+01326 3  2   40
2.703083+0 3.863693+0 2.346491+0 2.057345+0 2.535809+0 2.262868+01326 3  2   41
2.127090+0 2.261804+0 2.256791+0 2.533315+0 2.231035+0 2.144008+01326 3  2   42
2.151420+0 2.170908+0 2.095557+0 2.088164+0 2.073116+0 2.006202+01326 3  2   43
1.857245+0 1.711327+0 1.565212+0 1.419054+0 1.283900+0 1.274210+01326 3  2   44
1.266290+0 1.253001+0 1.238749+0 1.224132+0 1.210283+0 1.196914+01326 3  2   45
1.165341+0 1.042323+0                                          1326 3  2   46
                                                               1326 3  0   47
    .
    .
    .
    .
    .
    .
0.000000+0 0.000000+0          0          0         74          01326 3107  603
0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+01326 3107  604
0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+01326 3107  605
0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+01326 3107  606
0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+01326 3107  607
0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+01326 3107  608
0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+01326 3107  609
0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+01326 3107  610
0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+01326 3107  611
0.000000+0 0.000000+0 0.000000+0 0.000000+0 5.054213-6 3.154394-51326 3107  612
6.536392-5 1.636755-4 4.937945-4 1.599742-3 4.467725-3 9.026301-31326 3107  613
1.492042-2 2.174807-2 2.912657-2 3.467999-2 3.849531-2 3.982821-21326 3107  614
3.986411-2 3.988542-2 3.971171-2 3.918684-2 3.866013-2 3.800829-21326 3107  615
3.513214-2 1.922833-2                                          1326 3107  616
                                                               1326 3  0   617
0.000000+0 0.000000+0          0          0         74          01326 3851  618
0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+01326 3851  619
0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+01326 3851  620
0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+01326 3851  621
0.000000+0 0.000000+0 0.000000+0 0.000000+0 0.000000+0 1.713016-41326 3851  622
2.605423-2 2.747897-2 2.715636-2 2.656936-2 2.500170-2 2.240780-21326 3851  623
2.044663-2 1.866612-2 1.650006-2 1.483479-2 1.452751-2 1.445474-21326 3851  624
1.484054-2 1.514701-2 1.823890-2 1.948106-2 2.054081-2 2.151007-21326 3851  625
2.239315-2 2.355651-2 2.568895-2 2.711755-2 2.775385-2 2.831557-21326 3851  626
2.892825-2 2.891470-2 2.891944-2 2.909114-2 2.816820-2 2.648783-21326 3851  627
2.379156-2 2.036489-2 1.658885-2 1.236790-2 8.719756-3 5.515926-31326 3851  628
```

```
2.789373-3 1.967219-3 1.199289-3 5.263406-4 1.809961-4 1.629883-41326 3851  629
1.542762-4 1.491088-4 1.438805-4 1.385182-4 1.332255-4 1.265700-41326 3851  630
1.111753-4 5.198675-5                                             1326 3851  631
                                                                  1326 3   0  632
                                                                  1326 0   0  633
2.600000+4 5.536500+1          0          0          0          41132633  1  634
0.000000+0 0.000000+0          0          1          0          74132633  1  635
0.000000+0 0.000000+0         17          1         17           1132633  1  636
4.771962-4 6.905149-4 7.654629-4 7.983731-4 8.169108-4 8.372624-4132633  1  637
8.514704-4 8.574701-4 8.609741-4 8.656539-4 8.686033-4 8.703185-4132633  1  638
8.707277-4 8.728724-4 8.748390-4 8.757964-4 7.461125-4           132633  1  639
0.000000+0 0.000000+0         17          1         17           2132633  1  640
6.905149-4 1.247608-3 1.443338-3 1.529285-3 1.577697-3 1.630846-3132633  1  641
1.667951-3 1.683619-3 1.692770-3 1.704992-3 1.712694-3 1.717174-3132633  1  642
1.722305-3 1.727680-3 1.731573-3 1.733468-3 1.476784-3           132633  1  643
0.000000+0 0.000000+0         17          1         17           3132633  1  644
7.654629-4 1.443338-3 1.681504-3 1.786085-3 1.844993-3 1.909666-3132633  1  645
1.954815-3 1.973881-3 1.985016-3 1.999887-3 2.009259-3 2.014710-3132633  1  646
2.021500-3 2.028010-3 2.032580-3 2.034804-3 1.733500-3           132633  1  647
0.000000+0 0.000000+0         17          1         17           4132633  1  648
7.983731-4 1.529285-3 1.786085-3 1.898848-3 1.962365-3 2.032098-3132633  1  649
2.080780-3 2.101337-3 2.113343-3 2.129378-3 2.139484-3 2.145360-3132633  1  650
2.152879-3 2.159888-3 2.164754-3 2.167123-3 1.846226-3           132633  1  651
    .
    .
    .
    .
    .
    .
0.000000+0 0.000000+0          0        102          0          74132633  1 1738
0.000000+0 0.000000+0         13          1         13           1132633  1 1739
2.593694-4 2.593694-4 2.593694-4 2.593694-4 2.593694-4 2.593694-4132633  1 1740
2.593694-4 2.593694-4 2.593694-4 2.593694-4 2.593694-4 2.593694-4132633  1 1741
7.341822-5                                                       132633  1 1742
0.000000+0 0.000000+0         13          1         13           2132633  1 1743
1.216493-4 1.216493-4 1.216493-4 1.216493-4 1.216493-4 1.216493-4132633  1 1744
1.216493-4 1.216493-4 1.216493-4 1.216493-4 1.216493-4 1.216493-4132633  1 1745
3.443457-5                                                       132633  1 1746
0.000000+0 0.000000+0         13          1         13           3132633  1 1747
7.326236-5 7.326236-5 7.326236-5 7.326236-5 7.326236-5 7.326236-5132633  1 1748
7.326236-5 7.326236-5 7.326236-5 7.326236-5 7.326236-5 7.326236-5132633  1 1749
2.073796-5                                                       132633  1 1750
0.000000+0 0.000000+0         13          1         13           4132633  1 1751
5.201525-5 5.201525-5 5.201525-5 5.201525-5 5.201525-5 5.201525-5132633  1 1752
5.201525-5 5.201525-5 5.201525-5 5.201525-5 5.201525-5 5.201525-5132633  1 1753
1.472366-5                                                       132633  1 1754
0.000000+0 0.000000+0         13          1         13           5132633  1 1755
4.004717-5 4.004717-5 4.004717-5 4.004717-5 4.004717-5 4.004717-5132633  1 1756
4.004717-5 4.004717-5 4.004717-5 4.004717-5 4.004717-5 4.004717-5132633  1 1757
1.133592-5                                                       132633  1 1758
0.000000+0 0.000000+0         13          1         13           6132633  1 1759
2.690806-5 2.690806-5 2.690806-5 2.690806-5 2.690806-5 2.690806-5132633  1 1760
2.690806-5 2.690806-5 2.690806-5 2.690806-5 2.690806-5 2.690806-5132633  1 1761
```

```
7.616711-6                                                          132633  1 1762
0.000000+0 0.000000+0          13           1          13          7132633  1 1763
1.773523-5 1.773523-5 1.773523-5 1.773523-5 1.773523-5 1.773523-5132633  1 1764
1.773523-5 1.773523-5 1.773523-5 1.773523-5 1.773523-5 1.773523-5132633  1 1765
5.020211-6                                                          132633  1 1766
0.000000+0 0.000000+0          13           1          13          8132633  1 1767
1.386183-5 1.386183-5 1.386183-5 1.386183-5 1.386183-5 1.386183-5132633  1 1768
1.386183-5 1.386183-5 1.386183-5 1.386183-5 1.386183-5 1.386183-5132633  1 1769
3.923789-6                                                          132633  1 1770
0.000000+0 0.000000+0          13           1          13          9132633  1 1771
1.159959-5 1.159959-5 1.159959-5 1.159959-5 1.159959-5 1.159959-5132633  1 1772
1.159959-5 1.159959-5 1.159959-5 1.159959-5 1.159959-5 1.159959-5132633  1 1773
3.283431-6                                                          132633  1 1774
0.000000+0 0.000000+0          13           1          13         10132633  1 1775
8.578299-6 8.578299-6 8.578299-6 8.578299-6 8.578299-6 8.578299-6132633  1 1776
8.578299-6 8.578299-6 8.578299-6 8.578299-6 8.578299-6 8.578299-6132633  1 1777
2.428210-6                                                          132633  1 1778
0.000000+0 0.000000+0          13           1          13         11132633  1 1779
6.674139-6 6.674139-6 6.674139-6 6.674139-6 6.674139-6 6.674139-6132633  1 1780
6.674139-6 6.674139-6 6.674139-6 6.674139-6 6.674139-6 6.674139-6132633  1 1781
1.889210-6                                                          132633  1 1782
0.000000+0 0.000000+0          13           1          13         12132633  1 1783
5.566800-6 5.566800-6 5.566800-6 5.566800-6 5.566800-6 5.566800-6132633  1 1784
5.566800-6 5.566800-6 5.566800-6 5.566800-6 5.566800-6 5.566800-6132633  1 1785
1.575762-6                                                          132633  1 1786
0.000000+0 0.000000+0          17           1          17         13132633  1 1787
1.154480-6 1.154480-6 1.154480-6 1.154480-6 1.154480-6 1.154480-6132633  1 1788
1.154480-6 1.154480-6 1.154480-6 1.154480-6 1.154480-6 1.154480-6132633  1 1789
1.342892-5 1.827518-5 1.827518-5 1.827518-5 1.584766-5          132633  1 1790
0.000000+0 0.000000+0           5          13           5         14132633  1 1791
1.237530-5 1.726138-5 1.726138-5 1.726138-5 1.496853-5          132633  1 1792
0.000000+0 0.000000+0           5          13           5         15132633  1 1793
8.364854-6 1.166751-5 1.166751-5 1.166751-5 1.011770-5          132633  1 1794
0.000000+0 0.000000+0           5          13           5         16132633  1 1795
6.412509-6 8.944329-6 8.944329-6 8.944329-6 7.756243-6          132633  1 1796
0.000000+0 0.000000+0          27          13          27         17132633  1 1797
4.578647-6 6.386412-6 6.386412-6 6.386412-6 6.057870-6 3.913034-6132633  1 1798
3.913034-6 3.913034-6 3.913034-6 3.913034-6 3.913034-6 3.913034-6132633  1 1799
3.913034-6 3.913034-6 3.913034-6 3.913034-6 3.913034-6 3.913034-6132633  1 1800
3.913034-6 3.913034-6 3.913034-6 3.913034-6 3.913034-6 3.913034-6132633  1 1801
3.913034-6 3.913034-6 7.376951-7                                    132633  1 1802
0.000000+0 0.000000+0          23          17          23         18132633  1 1803
3.017049-6 2.271340-5 2.271340-5 2.271340-5 2.271340-5 2.271340-5132633  1 1804
2.271340-5 2.271340-5 2.271340-5 2.271340-5 2.271340-5 2.271340-5132633  1 1805
2.271340-5 2.271340-5 2.271340-5 2.271340-5 2.271340-5 2.271340-5132633  1 1806
2.271340-5 2.271340-5 2.271340-5 2.271340-5 4.281989-6          132633  1 1807
0.000000+0 0.000000+0          23          17          23         19132633  1 1808
2.023187-6 1.523126-5 1.523126-5 1.523126-5 1.523126-5 1.523126-5132633  1 1809
1.523126-5 1.523126-5 1.523126-5 1.523126-5 1.523126-5 1.523126-5132633  1 1810
1.523126-5 1.523126-5 1.523126-5 1.523126-5 1.523126-5 1.523126-5132633  1 1811
1.523126-5 1.523126-5 1.523126-5 1.523126-5 2.871437-6          132633  1 1812
0.000000+0 0.000000+0          23          17          23         20132633  1 1813
3.156422-5 2.376265-4 2.376265-4 2.376265-4 2.376265-4 2.376265-4132633  1 1814
```

```
2.376265-4 2.376265-4 2.376265-4 2.376265-4 2.376265-4 2.376265-4132633  1 1815
2.376265-4 2.376265-4 2.376265-4 2.376265-4 2.376265-4 2.376265-4132633  1 1816
2.376265-4 2.376265-4 2.376265-4 2.376265-4 4.479796-5          132633  1 1817
     .
     .
     .
     .
     .
     .
2.666295-6 2.007280-5 2.007280-5 2.007280-5 2.007280-5 2.007280-5132633  1 1899
2.007280-5 2.007280-5 2.007280-5 2.007280-5 2.007280-5 2.007280-5132633  1 1900
2.007280-5 2.007280-5 2.007280-5 2.007280-5 2.007280-5 2.007280-5132633  1 1901
2.007280-5 2.007280-5 2.007280-5 2.007280-5 3.784176-6          132633  1 1902
0.000000+0 0.000000+0          23          17          23        38132633  1 1903
2.169670-6 1.633404-5 1.633404-5 1.633404-5 1.633404-5 1.633404-5132633  1 1904
1.633404-5 1.633404-5 1.633404-5 1.633404-5 1.633404-5 1.633404-5132633  1 1905
1.633404-5 1.633404-5 1.633404-5 1.633404-5 1.633404-5 1.633404-5132633  1 1906
1.633404-5 1.633404-5 1.633404-5 1.633404-5 3.079335-6          132633  1 1907
0.000000+0 0.000000+0          23          17          23        39132633  1 1908
2.382007-7 1.793258-6 1.793258-6 1.793258-6 1.793258-6 1.793258-6132633  1 1909
1.793258-6 1.793258-6 1.793258-6 1.793258-6 1.793258-6 1.793258-6132633  1 1910
1.793258-6 1.793258-6 1.793258-6 1.793258-6 1.793258-6 1.793258-6132633  1 1911
1.793258-6 1.793258-6 1.793258-6 1.793258-6 3.380696-7          132633  1 1912
0.000000+0 0.000000+0           1          74           1        74132633  1 1913
0.000000+0                                                       132633  1 1914
0.000000+0 0.000000+0           0         103           0        74132633  1 1915
0.000000+0 0.000000+0           1          74           1        74132633  1 1916
0.000000+0                                                       132633  1 1917
0.000000+0 0.000000+0           0         104           0        74132633  1 1918
0.000000+0 0.000000+0           1          74           1        74132633  1 1919
0.000000+0                                                       132633  1 1920
0.000000+0 0.000000+0           0         105           0        74132633  1 1921
0.000000+0 0.000000+0           1          74           1        74132633  1 1922
0.000000+0                                                       132633  1 1923
0.000000+0 0.000000+0           0         106           0        74132633  1 1924
0.000000+0 0.000000+0           1          74           1        74132633  1 1925
0.000000+0                                                       132633  1 1926
0.000000+0 0.000000+0           0         107           0        74132633  1 1927
0.000000+0 0.000000+0           1          74           1        74132633  1 1928
0.000000+0                                                       132633  1 1929
0.000000+0 0.000000+0           0         851           0        74132633  1 1930
0.000000+0 0.000000+0           1          74           1        74132633  1 1931
0.000000+0                                                       132633  1 1932
                                                                 132633  0 1933
2.600000+4 5.536500+1           0           0           0        40132633  2 1934
0.000000+0 0.000000+0           0           2           0        74132633  2 1935
0.000000+0 0.000000+0          17           1          17         1132633  2 1936
2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.500000-3132633  2 1937
2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.500000-3132633  2 1938
2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.128462-3          132633  2 1939
0.000000+0 0.000000+0          17           1          17         2132633  2 1940
2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.500000-3132633  2 1941
2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.500000-3132633  2 1942
```

```
2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.128462-3          132633  2 1943
0.000000+0 0.000000+0         17          1         17         3132633  2 1944
2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.500000-3132633  2 1945
2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.500000-3132633  2 1946
2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.128462-3          132633  2 1947
0.000000+0 0.000000+0         17          1         17         4132633  2 1948
2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.500000-3132633  2 1949
2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.500000-3132633  2 1950
2.500000-3 2.500000-3 2.500000-3 2.500000-3 2.128462-3          132633  2 1951
```

The processed multigroup covariance data for **MAT1326** begin with Card 634. The designation of **MF=33** in Field 8 is the value of **MFCOV**, as discussed in Section X.K. Card 634 repeats the **ZA** and **AWR** numbers in Fields 1 and 2, and the number 41 in Field 6 indicates that covariance matrices follow for 41 reaction pairs. The number 1 in the **MT** field (Field 9) indicates that all 41 reaction-pairs have **MT=1** as the first reaction.

The data for one such reaction pair begin at Card 1738. Field 4 of Card 1738 contains the number 102, so the second reaction here is **MT1=102**. In other words, the data to follow refer to the covariance matrix of the iron total cross section (**MT=1**) with the iron radiative capture reaction **MT1=102**. The occurence of the number 14 in Field 6 of Card 1791 indicates that the block of data starting there refers to Group 14 of **MT=1**, *i.e.*, the fourteenth row of the covariance matrix. The number 5 in Field 3 of Card 1791 indicates that there are 5 consecutive groups of **MT1=102** for which covariances will be given explicitly. The first covariance applies to group 13 of **MT1=102**, as indicated by the number 13 in Field 4. Then, in referring to the entry in the fifth field of Card 1792, for example, one would say that the relative covariance of the iron total cross section in Group 14 (17.603-37.267 eV) with the iron radiative capture cross section in Group 17 (101.3–167.02 eV) is 1.496853×10^{-5}.

Any row of a covariance matrix, or any part of a row, that contains only zeros is omitted from the ERRORR output file. The only exception is that the last row (Group 74 here) is always given explicitly, even if it does contain only zeros. The occurrence of the last row terminates a given covariance matrix and signals that the next card will name a new reaction pair. These two conventions are applied simultaneously in Cards 1915–1917 to compactly represent the covariance matrix for a reaction pair (1, 103) having only zero covariances.

Although the output file in this example contained no cross-material covariances, the only difference in the file structure in the cross-material case is that MAT1 is specified in the integer field just preceding MT1, for example, in Field 3 on Card 1738. The entry MAT1=0 in that position is a flag to indicate that MAT1=MAT=1326.

M. Error Messages

ERRORR***ILLEGAL IREAD=---

Currently defined values are 0, 1, and 2.

ERRORR***ONLY --- EK ENERGIES ALLOWED.

The number of derived-cross-section energy-range boundaries is limited to 50.

ERRORR***TOO MANY REACTION TYPES.

This error occurs in runs with IVERF=4 if the number of requested covariance reactions is more than 60. A possible solution is to use the IREAD=1 option to select only the most needed reactions. This error also occurs in runs where internal group averaging is requested (NGOUT=0) if there are more than 200 sections in MF=33. An alternative is to run GROUP first, then ERRORR with NGOUT.GT.0.

ERRORR***NOT CODED FOR MFCOV=---.

Allowed values are 31 and 33.

ERRORR***TOO MANY STANDARDS REDEFINED.

Limit is 5. See input for NSTAN.NE.0.

ERRORR***ILLEGAL MT GT 870.

There is a fault in the evaluation.

GRIDD***MT --- REFERENCED IN DERIVATION FORMULA FOR ENERGY RANGE....

There is a fault in the evaluation or input.

GRIDD***TOO MANY MT-NUMBERS IN NC-TYPE SUB-SUBSECTIONS WITH LTY=0

Limit is 60.

GRIDD***CANNOT CALCULATE COVARIANCES OF REACTION....

This message is self-explanatory.

GRIDD***COVARIANCES OF REACTION....

There is a fault in the evaluation.

GRIDD***TOO MANY REACTION TYPES.

This error occurs if IVERF=5 and if the requested number of covariance reactions is greater than 60. If cross-material covariances are not needed, then a possible solution is to use the IREAD=1 option to select only the most needed reactions.

GRIDD***ILLEGAL MT1=0.

There is fault in the evaluation. Evaluator may have meant to indicate MT1=MT. If so, change the data value for MT1 on NENDF from 0 to MT and resubmit.

MERGE***STORAGE EXCEEDED.

Either the number of points in the x grid exceeds 150 or the number in the y grid exceeds 1000.

MERGE***Y(---)=--- LT Y(---)=---.

This problem should not occur.

GRIST***STANDARDS TAPE BAD.

There is a fault in the evaluation. Standards reactions must have at least one subsection; that is, they may not be components of a lumped reaction.

GRIST***ILLEGAL LB=0.

There is a fault in the evaluation. Absolute covariances are not permitted in the standards.

GRIST***ILLEGAL NI=0 IN THE STANDARD, MATSTD=---, MTSTD=---.

There is a fault in the evaluation. Standards reactions may not be derived from other reactions.

LMPMT***STORAGE EXCEEDED.

The number of component reactions of a lumped reaction is limited to 50.

COVCAL***STORAGE EXCEEDED IN LOC.

The number of NI-type sub-subsections in any subsection is limited to 30.

COVCAL***LB=--- WHEN LT=---.

There is a fault in the evaluation. LB-LT combination is not defined.

COVCAL***NOT CODED FOR LB=---.

Currently defined values are 0 through 6 and 8.

COVCAL***STORAGE EXCEEDED IN EGT.

At the time of COVCAL processing, the number of groups on NGOUT (equal to the number of union groups) is limited to 1000.

COVCAL***MFCOV MT FOUND NOT EQUAL TO INPUT MT.

There is a conflict between internal list of covariance reactions and the reactions encountered in the ENDF file. This problem should not occur.

COVCAL***STORAGE EXCEEDED IN A.

There is insufficient space in core to store the information from NI-type sub-sub-sections. Increase the size of /ESTORE/ and NAMAX.

COVCAL***ILLEGAL MT1=0.

See similar diagnostic in SUBROUTINE GRIDD above.

COVCAL***DATA IN A(LOCI) ARE ILLEGAL.

There is a fault in the evaluation. The ratio-to-standard data are bad. Either LTY is outside the legal ENDF range of 1–3 or the energies EL, EH are negative or out of order.

COVCAL***MUST REQUEST MAT1=--- AND MT1=--- ON CARD 10.

There is an input error in requesting covariances of X with Y when both are measured relative to Z. The standard Z must be specified on Card 10 and must appear as negative entries on Card 11.

ROGOUT***MAT --- NOT FOUND.

Cannot find MAT on multigroup library.

RDGOUT***MF---, MT--- NOT FOUND.

Cannot find requested group structure or cross sections on multigroup library.

RDGOUT***BAD INDEX FOR B EQUIVALENT TO SIG(IG).

This problem should not occur.

COVOUT***UNABLE TO FIND IY OR IYP FROM MTS ARRAY.

There is a conflict between the internal list of covariance reactions and the reactions encountered on the union-group covariance file produced in COVCAL. This problem should not occur.

COVOUT***STORAGE EXCEEDED IN SUM.

There is insufficient space in core to store the NMT1*NGN user-group cross sections. Increase the size of /ESTORE/ and NAMAX.

COVOUT***UNEXPECTEDLY, IX NE IY OR IXP NE IYP.

This problem should not occur.

SIGC***COVARIANCE REACTION MISSING FROM LUMPING TABLE.

A covariance reaction in the 851–870 range cannot be found in the lumped reaction definition tables. This problem should not occur.

RESPRP***ILLEGAL RESONANCE REPRESENTATION IN MF=32. LRF=---.

There is a fault in the evaluation. Allowed values of LRF are 1 ane 2 (single- and multi-level Breit-Wigner representations).

RESPRP***STORAGE EXCEEDED.

There is insufficient space in array ISCR to store resonance-parameter covariances for the current L-value. Limit is 5006 words.

RESPRP***BAD COVARIANCE DATA FOR RESONANCE PARAMETERS....

There is a fault in the evaluation. Covariance matrix of the parameters has negative variances, infinite correlation coefficients, or correlation coefficients greater than 2.0. Data may be out of order.

GRPAV---MF --- MT --- HAS THRESHOLD GT HIGHEST UNION ENERGY.

Though not a fatal error, this message often precedes an abort in COVCAL caused by the absence of this reaction on NGOUT. If IREAD=1, then delete this reaction from the requested list of covariance reactions.

GRPAV***UNABLE TO FIND TEMP=---.

The program cannot locate requested material and temperature on NPEND.

GRPAV***CANNOT GROUP-AVERAGE MT=---. USE GROUPR FIRST....

The MFCOV=31 and NGOUT=0 options are not compatible.

GRPAV***NOT CODED FOR MULTIMATERIAL GROUP AVERAGING....

The IREAD=2 and NGOUT=0 options are not compatible.

COLAPS***DID NOT FIND EXPECTED MF1 MT451.

Multigroup libraries produced with the GROUPR modules always contain the group structure in MF=1, MT=451.

COLAPS***STORAGE EXCEEDED.

Number of groups in NGOUT library structure is limited to 2*NPAGE+50, which, for the standard value of NPAGE (306), gives a limit of 662 groups.

COLAPS***NGOUT GROUP STRUCTURE DOES NOT SPAN UNION GRID.

The supplied library must cover the entire energy range of the union grid, which is the same as the energy range of the user's grid.

COLAPS***NOT CODED FOR MULTIPLE SIGMA ZEROES OR LEGENDRE ORDERS.

Because of this restriction, existing "general-purpose" GENDF files may not be suitable for use as input to ERRORR, in which case a special GROUPR run, just to prepare the NGOUT file for ERRORR, will be required.

UNIONG***EXCEEDED STORAGE IN MFCOV ENERGY GRID.

Applies only if IVERF=4.

UNION***ENERGIES OUT OF ORDER. --- LT ---.

Applies only if IVERF=4.

UNIONG***EXCEEDED STORAGE IN UNION GRID.

Insufficient space in core to store the union grid. Increase /ESTORE/ and NAMAX.

EGNGPN***TOO MANY GROUPS.

User group structure is limited to 640 groups.

EGNGPN***READ-IN GROUP STRUCTURE IS OUT OF ORDER.

With the IGN=1 option, the supplied group structure must be in increasing order.

EGNGPN***ILLEGAL GROUP STRUCTURE REQUESTED.

Allowed values of IGN are 1–17 and 19.

EGNWTF***ILLEGAL WEIGHT FUNCTION REQUESTED.

Allowed values of IWT are 1–9.

EGTSIG***MT=0

In the NGOUT=0 option, a point cross section has been requested by GRPAV for an illegal MT number. In this context, the allowed values of MT are 1–150, 207, 251–253, and 600–799. MT numbers 600–699 are not defined in ENDF-5, but may be used on ENDF and PENDF files specially prepared as input to ERRORR, to allow processing of nonstandard reaction types.

N. Input/Output Units

The following logical units are used:

10 NGOUT in GRPAV and NTP in COLAPS. Contains the union group cross sections for all reactions with covariances. If the user input value of NGOUT is zero, point data are read from NPEND and written to NGOUT=-10 in GRPAV. If the input value NGOUT is not zero, coarse group multigroup data are read from the user's NGOUT in COLAPS and written to NTP=-10. When COLAPS is finished, NGOUT is reset to -10.

11 NSCR in COVCAL and COVOUT. Contains the union-group multigroup covariances for the directly evaluated reactions.

12 NSCR2 in COVOUT. This is a second copy of NSCR, which is read only for the "trivial" derivation case, ISD=1. Saves execution time by eliminating the calculation of many zero contributions.

13 NSCRG in the COVCAL and RDGOUT. This unit is used to extract a single MAT from a multimaterial NGOUT.

15 NSCR in LUMPTMT and LUMPXS. This unit is used if MT numbers in the 851–870 range (lumped reactions) are found in ENDF covariance file to keep a copy of the covariance file.

20–99 Users can choose any of these numbers for NENDF, NPEND, NGOUT, NOUT, NIN, and NSTAN.

Units 11 and 12 are normally binary, although they can be switched to formatted mode by changing the value of the variable IMODE in the main program to +1. Unit 13 has the same mode as NGOUT. The user can choose the modes for NENDF, NPEND, NGOUT, NOUT, NIN, and NSTAN, except that NOUT and NIN must have the same mode.

O. Acknowledgments

Much of the original programming of the ERRORR and COVR (see following chapter) modules was performed by R. M. Boicourt. We offer thanks to J. W. Davidson for his sustained interest in our development of covariance methods and libraries, and to R. W. Peelle and B. H. Broadhead for helpful advice in several areas. Thanks are also due for the encouragement received from the U.S. Department of Energy, Office of Fusion Energy. Partial financial support for the

preparation of an earlier version of this manual chapter was provided by the Paul Scherrer Institut (PSI), Würenlingen, Switzerland. Support for the preparation of the current version of this chapter has been generously supplied by the OECD Nuclear Energy Agency.

P. References

1. S. A. W. Gerstl, D. J. Dudziak, and D. W. Muir, "Cross Section Sensitivity and Uncertainty Analysis with Application to a Fusion Reactor," *Nucl. Sci. Eng.* **62**, 137 (January 1977).

2. W. A. Reupke, D. W. Muir, and J. N. Davidson, "Consistency of Neutron Cross-Section Data, S_n Calculations, and Measured Tritium Production for a 14-MeV Neutron-Driven Sphere of Natural-Lithium Deuteride," *Nucl. Sci. Eng.* **82**, 416 (December 1982).

3. D. W. Muir, "Evaluation of Correlated Data Using Partitioned Least Squares: A Minimum-Variance Derivation," *Nucl. Sci. and Eng.* **101**, 88-93 (January 1989).

4. P. F. Rose and C. L. Dunford, Eds., "ENDF-102, Data Formats and Procedures for the Evaluated Nuclear Data File, ENDF-6," Brookhaven National Laboratory report BNL-NCS-44945, Rev. 10/91.

5. R. W. Peelle and D. W. Muir, "Extended Covariance Data Formats for the ENDF/B-VI Data Evaluation," Proc. NEACRP Specialists' Meeting on the Application of Critical Experiments and Operating Data to Core Design via Formal Methods of Data Adjustment, Jackson Hole, Wyoming, 23-24 September 1988, report NEACRP-L-307, pp. 305-315 (1988).

6. H. Vonach, "Direct Covariance Data for Coupled Energy-Angle Distributions," Proc. Specialists' Meeting on Evaluation and Processing of Covariance Data (COV 92), 7-9 October 1992, Oak Ridge, Tennessee, sponsored by the Nuclear Energy Agency Nuclear Science Committee of the OECD, Nuclear Energy Agency report NEA/NSC/DOC(93)3, pp. 183-193. See related paper by D. W. Muir, "Description of Covariances of Emission Spectra using ENDF File 30," *ibid*, pp. 195-211.

7. C. R. Weisbin, E. M. Oblow, J. Ching, J. E. White, R. Q. Wright, and J. Drischler, "Cross Section and Method Uncertainties: The Application of Sensitivity Analysis to Study Their Relationship in Radiation Transport Benchmark Problems," Oak Ridge National Laboratory report ONRL-TM-4847 (ENDF-218) (August 1975). (See especially Chapter IV.)

8. J. D. Smith III, "Processing ENDF/B-V Uncertainty Data into Multigroup Covariance Matrices," Oak Ridge National Laboratory report ORNL/TM-7221 (ENDF-295) (June 1980).

9. P. G. Young, "Evaluation of n + ^{7}Li Reactions using Variance-Covariance Techniques," *Trans. Am. Nucl. Soc.* **39**, 272 (1981).

10. P. G. Young, J. W. Davidson, and D. W. Muir, "Evaluation of the ^{7}Li(n,n't)^{4}He Cross Section for ENDF/B-VI and Application to Uncertainty Analysis," *Fusion Tech.* **15**, 440-448 (1989).

11. D. W. Muir and R. J. LaBauve, "COVFILS: A 30-Group Covariance Library Based on ENDF/B-V," Los Alamos National Laboratory report LA-8733-MS (ENDF-306) (March 1981).

12. R. E. MacFarlane, D. W. Muir, and R. M. Boicourt, "The NJOY Nuclear Data Processing System, Vol. I: User's Manual," Los Alamos National Laboratory report LA-9303-M (ENDF 324) (May 1982).

13. D. W. Muir, "COVFILS-2: Neutron Data and Covariances for Sensitivity and Uncertainty Analysis," *Fusion Tech.* **10** (3), Part 2B, 1461 (November 1986).

XI. COVR

A. Introduction

The COVR module of NJOY is an editing module that post-processes the output of ERRORR in a manner analogous to the way MATXSR and DTFR post-process the output of GROUPR. COVR performs two quite separate functions using the multigroup covariance file from ERRORR as input. First, it can prepare a new covariance library in a highly compressed card-image format, which is suitable for use as input to sensitivity analysis programs.[1, 2] (Data in this form can also be copied to the system output file to obtain a compact printed summary of an ERRORR run without using the sometimes bulky long-print option in ERRORR. Such a summary can also provide information on standard deviations and correlation coefficients, neither of which are printed by ERRORR.) The second main function of COVR is to produce publication-quality plots[3] of the multigroup covariance information. We first describe the library option.

B. Production of Boxer-Format Libraries

As discussed in ERRORR, Section X.L of this manual, the output file of that module contains the group structure, cross sections, and either absolute (IRELCO=0) or relative (IRELCO=1) covariances for one or more materials, in either card-image or NJOY blocked-binary form. If the COVR user-input parameter NOUT is greater than zero, COVR reads an ERRORR output file from unit NIN and produces a new multigroup covariance library on unit NOUT. COVR performs only sorting and reformatting operations on the ERRORR data.

In COVR, as well as ERRORR, the ENDF energy ordering is followed. That is, low group indices correspond to low energies, high indices to high energies. Regarding group structures, in the library mode of operation, COVR (like ERRORR) makes extensive use of external (disk) storage, so that even on machines that have a relatively small central memory, very large group structures (up to 640 groups) can be handled successfully. (See, however, the comments at the end of Section XI.C regarding group-number limitations in the plot mode.)

The material and reaction coverage of COVR post-processing is determined by a set of reaction pairs (MAT,MT;MAT1,MT1) supplied by the code user (see input Card 4 in the input instructions and the corresponding discussion that follows in Section XI.D). At the beginning of the output library, the group structure is given. Then, for each specified reaction pair, the output library contains either a covariance matrix or a correlation matrix, depending on the output option (MATYPE)

selected. In the case that covariances are requested (**MATYPE=3**), the type of data on **NOUT** (absolute *vs.* relative) is governed by the covariance type present on **NIN**. In addition, whenever **MAT1=MAT** and **MT1=MT**, the group-cross-section vector and the (absolute/relative) standard-deviation vector for that particular reaction are written to **NOUT** just before the matrix itself. All data (group structure, cross sections, standard deviations, and matrices) are written to **NOUT** using a highly compressed, card-image format.

The design of this format, called the Boxer format, proceeds from a simple fact: as discussed in the ERRORR chapter of this manual, most of the ENDF covariance formats define certain rectangular regions (boxes) in energy "space," over which the relative covariance is constant. (The ENDF format allowing a constant absolute covariance is only rarely used.) The coordinate axes of the two-dimensional energy space in question are E_x and E_y, where x and y indicate the particular reaction pair to which the ENDF covariances apply. Because of this feature of the basic evaluations, one expects that an element of a multigroup relative covariance matrix, derived from the ENDF data for a given reaction pair, frequently will be identical either to the element before it in the same row (E_x constant, E_y varying) or to the element above it in the same column (E_y constant, E_x varying). Thus, the Boxer format allows a combination of horizontal and vertical repeat operations.

Even though the ERRORR output format already suppresses zero covariances, very large data compression factors can be achieved in transforming data from the ERRORR format to the Boxer format. As one example, the ERRORR output file for a particular 137-group reactor-dosimetry library[4] contained 38 000 card images, while the corresponding COVR output file contained fewer than 1000 card images.

In the Boxer format, data are stored as a list of numerical data values (*e.g.*, relative covariances), together with a list of integers that control the loading of these data into the reconstructed array $C(i,j)$. A negative control integer, $-n$, indicates that the next value in the data list is to be loaded into the next n columns (j-locations) of the current row of $C(i,j)$. A positive integer m, on the other hand, means that, for the next m j-values, the value to be loaded is to be carried down from the row above,

$$C(i,j) = C(i-1,j)\,. \tag{1}$$

For the first row ($i = 1$), the row above is defined to be a row containing all zeros.

In constructing the compressed data set in subroutine PRESS, the choice between using the "repeat-new-value" method or the "carry-down" method is made dynamically on the basis of taking the longest possible step. If $m = n$, the carry-down method is chosen, as it does not require an entry in the data list.

As an additional compression feature of the format, one may indicate by a "flag" that the matrix $C(i,j)$ is symmetric; hence, explicit instructions are provided in the compressed data library for the reconstruction of only the upper right triangle of $C(i,j)$. These various aspects of the Boxer format are illustrated by a simple example in Fig. 1. Here a, b, c and d are arbitrary, nonzero, unequal data values.

Before the data values are tested in PRESS to see if they are indeed equal, the NJOY utility SIGFIG is called to round off the trailing digits that would not appear on the formatted output anyway. Relative covariances, for example, are written to NOUT in 1P8E10.3 format, which has only four significant figures. Thus, the three data values 0.036126, 0.036130, and 0.036134 would all be judged to be equal by this logic.

C. Generation of Plots

The COVR plot mode, which is requested by specifying NOUT=0, is used to generate publication-quality plots of multigroup covariance data from a card-image or binary ERRORR output file. Examples of plots produced by COVR can be seen in Figs. 2 and 3 of the ERRORR chapter of this manual. Many more examples can be found in the COVFILS report.[3]

In addition to their usefulness in preparing publications, the plots have proved to be a useful tool for checking the reasonableness and mechanical correctness of new covariance evaluations. One can, for example, execute ERRORR and COVR in tandem, using the evaluator's energy grid as the user group structure (ERRORR input option IGN=19). The output of such a run is a series of plots showing all important features of the covariance evaluation.

As can be seen in the examples mentioned above, each plot contains a shaded contour map of the correlation matrix. Positive-correlation regions are shaded with parallel straight lines, while negative correlations are indicated by cross-hatching. The plots also contain two semilogarithmic curves giving the energy dependence of the associated standard deviation vectors. One of the vector plots is rotated by 90 degrees, so that the logarithmic energy grids for the vector plots can be aligned with the corresponding grids for the matrix plot.

Original Data Set

$j \rightarrow$

$i \downarrow$

a	a	b	b	0	0
a	a	b	b	0	0
b	b	b	b	0	0
b	b	b	b	0	0
0	0	0	0	c	c
0	0	0	0	c	d

Boxer Format, Symmetry Flag Off

a	b	b	0	c	d			
-2	-2	8	-4	8	-4	-2	5	-1

Boxer Format, Symmetry Flag On

a	b	c	d	
-2	-2	14	-2	-1

Figure 1: Illustration of Boxer format.

This type of plot presents the covariance data in a more lucid manner than most alternative plotting packages. However, use of this option does require the availability of the proprietary plotting-software package DISSPLA.[5] Only a few of the DISSPLA subroutines are actually called (curve plotting, shading, Greek lettering, and subplot centering and rotation), so that conversion to other plotting packages is probably quite feasible. Users who do not have access to DISSPLA and who do not wish to convert COVR to another system can install a version of NJOY without DISSPLA coding (and, hence, without the COVR plotting capability) by omitting the `*SET DIS` card in the initial UPD run. (See the introduction of the report for more details.) Omission of the `*SET DIS` card will not interfere in any way with the use of the COVR library option described in Section XI.B.

Just as in the library mode, the material and reaction coverage of the sequence of plots generated in the plot mode is determined by the reaction pairs specified by the user on input Card 4. Other input, specific to the generation of plots, is described in Section XI.D. We next discuss some aspects of the plot-generation coding.

After a covariance matrix is read in subroutine `COVARD`, it is converted to a correlation matrix in subroutine `CORR`. (See the discussion of such conversions in Section XI.B) The correlation matrix is then scanned in subroutine `MATSHD` to find a set of boundary curves that divide $E_x - E_y$ energy space into a small number of connected regions of nearly constant correlation strength. Here the phrase nearly constant refers to the subdivision of the range of possible correlation values (-1.0 to $+1.0$) into a number of equal-width bands, the fineness of the subdivision being controlled by the user-input parameter `NDIV`. Two regions have nearly equal correlations if the correlations fall into the same band. As each boundary curve is located, the enclosed region is shaded (with a line density proportional to the band correlation magnitude) with a single call to the `DISSPLA` area-shading routine `SHADE`.

The algorithm used to find the maximum extent of a region of nearly constant correlation is best described by referring to the simple example in Fig. 2. After locating the upper left corner of a new correlation region, for example at point a in the figure, the search proceeds as far as possible down the first column (E_y held constant) to point b. New columns are scanned in the same way: $c - d$, $e - f$, $g - h$. Having found that the pattern does continue into column $e - f$, for the sake of simplicity, the algorithm ignores the possibility of additional, disjoint, continuations of the pattern elsewhere in the same column (such as at $e' - f'$), and

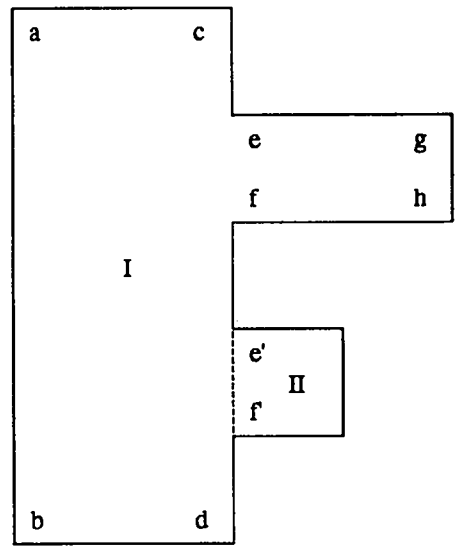


Figure 2: Pattern-search logic in MATSHD.

region I is ended at h. (Region II will be found and correctly shaded at a later stage of the calculation.)

Multiple cells to the SHADE routine usually produce a discernible discontinuity in the shading pattern (along the stepped line in Fig. 2). This is not really a problem, but the effect is occasionally noticeable in applications. See, for example, Fig. 80 of the COVFILS report.[3] Note there that, while c-shaped regions require multiple calls to SHADE, u-shaped regions do not.

As the plot option is presently coded, the entire correlation matrix must reside in memory during the pattern-search operation just described. This fact, coupled with the fact that the availability of central memory is already limited in COVR because of the space occupied by the DISSPLA subroutines, implies that very large group structures cannot be processed on machines with a modest fixed memory.

D. Input Instructions for COVR

As an aid to discussions of the user input to COVR, we list below the input instructions that appear as comment cards at the beginning of the current version of this module. It is always advisable to consult the comment-card instructions embedded in the version of the code actually being used and not to rely on the instructions published in any document, including this one. Following the listing

are further remarks on several of the input items; the remarks supplement the comment cards below. See especially the lengthy discussion of the parameters on Card 4.

```
*---INPUT SPECIFICATIONS (FREE FORMAT)-------------------------*
*                                                              *
*  CARD 1                                                      *
*     NIN             INPUT TAPE UNIT                          *
*     NOUT            OUTPUT TAPE UNIT                         *
*                     (DEFAULT=0=NONE)                         *
*                                                              *
*   ---CARDS 2A AND 3A FOR NOUT=0 (PLOT OPTION) ONLY---        *
*                                                              *
*  CARD 2A                                                     *
*     GAPREF          REFERENCE SHADING GAP WIDTH              *
*                     (DEFAULT=.020)                           *
*     GAP1            FULL-CORRELATION SHADING GAP WIDTH       *
*                     (DEFAULT=.008)                           *
*     XGAP            FACTOR FOR NEGATIVE CORRELATION GAP      *
*                     ADJUSTMENT (DEFAULT=1.75)                *
*     EPMIN           LOWEST ENERGY OF INTEREST (DEFAULT=0.)   *
*     NOTE.  THE ABOVE DEFAULTS ARE ACTIVATED BY ENTERING ZEROES *
*     OR JUST A SLASH.  A SECOND SET OF DEFAULTS FOR GAPREF AND  *
*     GAP1 ARE ACTIVATED BY INPUTTING A VALUE OF GAPREF GREATER  *
*     THAN 0.2.  IN THIS CASE, THE VALUES USED ARE 0.030*FACT    *
*     AND 0.016*FACT, RESPECTIVELY, WHERE FACT IS THE INPUT      *
*     VALUE OF GAPREF.                                         *
*  CARD 3A                                                     *
*     IRELCO          TYPE OF COVARIANCES PRESENT ON NIN       *
*                     0/1=ABSOLUTE/RELATIVE COVARIANCES        *
*                     (DEFAULT=1)                              *
*     NCASE           NO. CASES TO BE RUN (MAXIMUM=40)         *
*                     (DEFAULT=1)                              *
*     NOLEG           PLOT LEGEND OPTION                       *
*                     -1/0/1=LEGEND FOR FIRST SUBCASE ONLY/    *
*                     LEGEND FOR ALL PLOTS/NO LEGENDS          *
*                     (DEFAULT=0)                              *
*     NSTART          SEQUENTIAL FIGURE NUMBER                 *
*                     0/N=NOT NEEDED/FIRST FIGURE IS FIGURE N. *
*                     (DEFAULT=1)                              *
*     NDIV            NO. OF SUBDIVISIONS OF EACH OF THE       *
*                     GRAY SHADES (DEFAULT=1)                  *
*                                                              *
*   ---CARDS 2B, 3B, AND 3C FOR NOUT GT 0 (LIBRARY OPTION) ONLY--*
*                                                              *
*  CARD 2B                                                     *
*     MATYPE          OUTPUT LIBRARY MATRIX OPTION             *
*                     3/4=COVARIANCES/CORRELATIONS             *
*                     (DEFAULT=3)                              *
*     NCASE           NO. CASES TO BE RUN (MAXIMUM=40)         *
*                     (DEFAULT=1)                              *
*  CARD 3B                                                     *
*     HLIBID          UP TO 6 CHARACTERS FOR IDENTIFICATION    *
```

```
*  CARD 3C                                                          *
*     HDESCR          UP TO 21 CHARACTERS OF DESCRIPTIVE            *
*                     INFORMATION                                   *
*                                                                   *
*   ---CARDS 4 FOR BOTH OPTIONS---                                  *
*                                                                   *
*  CARD 4                                                           *
*     MAT             DESIRED MAT NUMBER                            *
*     MT              DESIRED MT NUMBER                             *
*     MAT1            DESIRED MAT1 NUMBER                           *
*     MT1             DESIRED MT1 NUMBER                            *
*                     (DEFAULT FOR MT, MAT1 AND MT1 ARE 0,0,0       *
*                     MEANING PROCESS ALL MTS FOR THIS MAT          *
*                     WITH MAT1=MAT)                                *
*                     (NEG. VALUES FOR MT, MAT1, AND MT1 MEAN       *
*                     PROCESS ALL MTS FOR THIS MAT, EXCEPT FOR      *
*                     THE MT-NUMBERS -MT, -MAT1, AND -MT1.  IN      *
*                     GENERAL, -N WILL STRIP BOTH MT=1 AND MT=N.    *
*                     -4 WILL STRIP MT=1, MT=3, AND MT=4, AND       *
*                     -62, FOR EXAMPLE, WILL STRIP MT=1, MT=62,     *
*                     MT=63, ... UP TO AND INCL. MT=90.)            *
*           REPEAT CARD 4 NCASE TIMES                               *
*                                                                   *
* NOTE---IF MORE THAN ONE MATERIAL APPEARS ON THE INPUT TAPE,       *
* THE MAT NUMBERS MUST BE IN ASCENDING ORDER.                       *
*                                                                   *
*********************************************************************
```

GAPREF, GAP1, XGAP ... These parameters may need to be adjusted away from the default values to achieve good results with some plotting devices.

EPMIN ... This parameter is used to eliminate uninteresting energy regions from the correlation and standard deviation plots, or to display high-energy regions with greater resolution.

IRELCO ... This parameter must match the value used in the ERRORR run that produced the covariances to be plotted.

NCASE ... This is the number of cases, or occurrences of Card 4. See the discussion of input parameters MAT, MT, MAT1, and MT1 below. Presently, NCASE is limited to 40. Problems larger than this can be run as a series of COVR jobs, each processing a batch of up to 40 cases. See also the parameter NSTART below.

NOLEG ... "Legend" here refers to the gray-shading scale (key) as well as the figure caption. NOLEG=1 is used as a rough-draft mode to display plots quickly.

NSTART ... Unless NSTART=0, the plots are assigned a sequential figure number, beginning at NSTART, and a list of figures is drawn on the final plot frame.

NDIV ... One gray-shade step equals 0.20 in correlation magnitude if NDIV=1. Finer gradations are possible with NDIV greater than 1. The plots appearing in Figs. 2 and 3 of the ERRORR chapter of this manual were generated with NDIV=2.

HLIBID ... This is a 6-character string, normally containing the name of the output covariance library. It is written on the header records present at the beginning of each output data block.

HDESCR ... This contains additional information, for example, on where and when the library was produced; it is also written on the data header records.

MAT, MT, MAT1, MT1 ... The information contained in the output library or plot file is controlled by means of these parameters on input Card 4. If MT is positive, then a single covariance matrix for reaction (MAT,MT) with reaction (MAT1,MT1) will be read from NIN and processed. On unit NIN (and, in the library option, on unit NOUT), the rapidly varying, or column, index is the group index of (MAT1,MT1), and the slowly varying, or row, index is the group index of (MAT,MT). If MAT1 is different from MAT, COVR will expect to find separate materials (produced by separate ERRORR runs) for both MAT and MAT1 on NIN. The MAT numbers must occur on NIN in ascending order. In the case of positive MT, the entries MAT1=0 and MT1=0 are shorthand for MAT1=MAT and MT1=MT, respectively. If, on the other hand, MT is zero, negative, or defaulted, Card 4 becomes a kind of macro-instruction that is expanded by SUBROUTINE EXPNDO into a request for many MT-MT1 pairs, all with MAT1=MAT. If, for example, Card 4 contains the entry

```
MAT/
```

or, equivalently,

```
MAT 0 0 0
```

first the cross-section file, MF=3, for material MAT is read from the input covariance tape on unit NIN to obtain the list of reactions present. Then, all possible reaction combinations MT-MT1 are formed in ENDF order. Thus, for example, if the reactions present are MT=1, 2, 3, and 16, then the behavior of the code is the same as if the following input were specified:

```
MAT 1 0 1
MAT 1 0 2
MAT 1 0 3
MAT 1 0 16
MAT 2 0 2
MAT 2 0 3
MAT 2 0 16
```

```
MAT 3 0 3
MAT 3 0 16
MAT 16 0 16  .
```

Because of the clear labor-saving advantage of this feature, especially if there are many reactions, enhancements have been added to permit its use in the additional situation in which many, but not all, combinations are desired. As discussed in some detail in the input instructions, it is possible to strip selected reactions out of the list before the MT-MT1 combinations are formed. For example, if the reactions present are once again 1, 2, 3, and 16, then a Card 4 containing

```
MAT -3/
```

would produce the same output as the following three cards:

```
MAT 2 0 2
MAT 2 0 16
MAT 16 0 16.
```

The stripping of the higher inelastic levels, mentioned in the input instructions, is useful because ERRORR automatically resets the highest user-group energy to 20 MeV whenever IREAD=0, and this can result in the inclusion of unwanted high-threshold reactions on the ERRORR output file.

For one of several reasons, a requested reaction pair (MAT,MT; MAT1,MT1) may be absent from the COVR output. The usual reason that this occurs is that the normal IREAD option for the ERRORR module is IREAD=0, and this results in the generation of an output matrix for every possible reaction-pair combination. For those combinations that do not occur in the ENDF evaluation, the ERRORR output matrix contains only zeroes. These null matrices are omitted at the COVR output stage, in both the library and plot modes.

Additionally, in the plot mode, the correlations may be nonzero, but everywhere less than 0.2/NDIV in absolute magnitude. In this case, the entire correlation plot would consist of a single blank region. These rather uninteresting plots having small correlations are also omitted from the plot file.

A final, similar category is the class of "empty" plots. Because DISSPLA uses parallel lines to achieve a gray effect in the SHADE routine, there is clearly some lower limit, for a given value of the correlation magnitude, on the physical size of a region that can be sensibly shaded. COVR does not attempt to shade regions that are smaller than this limit. If a correlation matrix does contain some plottable data (magnitudes exceeding 0.2/NDIV), but all plottable regions are smaller than

the size limit discussed above, then an empty plot will be generated on the plot file. As a convenience in discarding these empty plots at the time a report is produced, no caption is written on such plots and the figure number is not advanced. Also, these plots are omitted from the list of figures prepared at the end of a plot run. (See the discussion of user-input parameter **NSTART**.)

In the library mode, each omission of a requested, but null, matrix is noted on the OUTPUT file with an informative diagnostic. In the plot mode, a summary table is printed at the end of the run to identify all requested matrices that were omitted because they were null or small, as well as those that were plotted, but empty.

E. COVR Example Problem

In this section we discuss the production of a particular COVR output library, both to illustrate the input and as a supplement to the general discussion of Boxer-format libraries in Section XI.B. In the ERRORR chapter of this report (the second input set in Section X.K), we gave the complete NJOY input for producing a 7-reaction, 30-group covariance library in ERRORR output format for ^{12}C. By appending the following lines to that input (just before the *STOP* card), one can produce, in addition, a 2-reaction COVR output library in Boxer format.

```
*COVR*
-23 24
3 3
*  LIB*
*MAT1306 COVR EXAMPLE*/
1306 2 0 2
1306 2 0 4
1306 4 0 4
```

The resulting library occupies only 49 lines and is listed below.

```
0    LIB-A- 30 MAT1306 COVR EXAMPLE  1306   2 1306   2  31 10  31  3   0  31   1
 1.390E-04 1.520E-01 4.140E-01 1.130E+00 3.060E+00 8.320E+00 2.260E+01 6.140E+01
 1.670E+02 4.540E+02 1.235E+03 3.350E+03 9.120E+03 2.480E+04 6.760E+04 1.840E+05
 3.030E+05 5.000E+05 8.230E+05 1.353E+06 1.738E+06 2.232E+06 2.865E+06 3.680E+06
 6.070E+06 7.790E+06 1.000E+07 1.200E+07 1.350E+07 1.500E+07 1.700E+07
 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
 -1 -1 -1 -1 -1
1    LIB-A- 30 MAT1306 COVR EXAMPLE  1306   2 1306   2  23 10  23  3   0  30   1
 4.739E+00 4.738E+00 4.735E+00 4.729E+00 4.712E+00 4.676E+00 4.579E+00 4.332E+00
 4.002E+00 3.619E+00 3.103E+00 2.477E+00 1.998E+00 1.820E+00 1.710E+00 2.256E+00
 1.447E+00 9.921E-01 8.223E-01 7.627E-01 8.631E-01 8.228E-01 8.845E-01
```

```
 -8 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
2    LIB-A- 30 MAT1306 COVR EXAMPLE  1306   2 1306   2  18 10  18  3   0  30   1
 2.000E-03 1.964E-03 4.583E-03 3.822E-03 4.583E-03 4.288E-03 3.730E-03 4.141E-03
 6.501E-03 1.011E-02 9.558E-03 9.981E-03 1.806E-02 1.854E-02 3.296E-02 3.760E-02
 4.255E-02 7.632E-02
 -9 -1 -5 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
3    LIB-A- 30 MAT1306 COVR EXAMPLE  1306   2 1306   2  43 10  58  4   0  30   0
 4.000E-06 2.797E-06 3.856E-06 6.317E-06 4.613E-06 2.407E-06 1.229E-06 2.100E-05
 1.534E-05 8.000E-06 4.087E-06 1.461E-05 1.366E-05 2.100E-05 1.838E-05 9.004E-06
 1.391E-05 8.934E-06 1.715E-05 4.447E-06 4.227E-05 4.373E-05 2.446E-05 1.980E-05
 1.223E-05 1.022E-04 6.116E-05 4.048E-05 2.500E-05 9.136E-05 4.572E-05 9.962E-05
 6.641E-05 6.308E-05 3.260E-04 1.614E-04 3.437E-04 1.086E-03 4.250E-04 1.413E-03
 7.052E-04 1.811E-03 5.825E-03
  -9  -1 224  -1  -5  -1  -4  -1   9  -5  -1  -4  -1  79  -1  -1  13  -1  13  -1
  -1  11  -1  -1  10  -1  -1   9  -1  -1  -1  -1  -6  -1  -1  -1  -6  -1  -1   6
  -1  -1  -1   4  -1  -1.  4  -1   4  -1  -2   1  -1  -1   1  -1   1  -1
3    LIB-A- 30 MAT1306 COVR EXAMPLE  1306   2 1306   4  38 10  47  4   0  30  30
-1.106E-04-4.897E-05-3.047E-05-2.164E-05-2.630E-05-2.337E-05-2.192E-05-2.261E-04
-1.001E-04-6.228E-05-4.423E-05-5.376E-05-4.777E-05-4.481E-05-1.441E-03-2.659E-04
-1.571E-04-1.210E-03-1.305E-03-4.021E-04-1.131E-03-6.462E-04-8.562E-04-2.261E-04
-1.001E-04-6.228E-05-1.922E-03-9.140E-04-8.121E-04-7.520E-04-3.040E-03-1.348E-03
-1.517E-03-3.460E-03-4.423E-05-5.376E-05-4.777E-05-1.044E-02
 623  -1  -1  -1  -1  -1  -1  -1  23  -1  -1  -1  -1  -1  -1  -1  53  -1  -1  -1
  27  -1  -1  -1  27  -1  -1  -1  27  -1  -1  -1  -1  -1  -1  27  -1  -1  -1  28
  -1  -1  27  -1  -1  -1  -1
1    LIB-A- 30 MAT1306 COVR EXAMPLE  1306   4 1306   4   7 10   8  3   0  30   1
 6.091E-02 2.478E-01 3.301E-01 4.310E-01 4.013E-01 4.306E-01 4.935E-01
 23 -1 -1 -1 -1 -1 -1 -1
2    LIB-A- 30 MAT1306 COVR EXAMPLE  1306   4 1306   4   7 10   8  3   0  30   1
 2.499E-01 1.085E-01 9.724E-02 9.806E-02 1.358E-01 1.263E-01 2.177E-01
 23 -1 -1 -1 -1 -1 -1 -1
3    LIB-A- 30 MAT1306 COVR EXAMPLE  1306   4 1306   4  28 10  29  4   0  30   0
 6.245E-02 7.530E-03 3.823E-03 1.154E-03 1.338E-03 1.211E-03 1.155E-03 1.176E-02
 2.311E-03 5.938E-04 6.451E-04 5.843E-04 5.578E-04 9.455E-03 1.887E-03 4.735E-04
 4.295E-04 4.106E-04 9.617E-03 1.872E-03 1.670E-03 3.033E-04 1.844E-02 5.214E-03
 3.493E-04 1.596E-02 5.253E-03 4.741E-02
 437  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1
  -1  -1  -1  -1  -1  -1  -1  -1  -1
```

Note that the header card at the start of each data block contains an integer ITYPE, specifying the type of data contained in the current block, a 12-character library name ("LIB-A- 30" in this case), 21 characters of user-supplied descriptive information, MAT, MT, MAT1, MT1, and a set of 7 integers. The meaning of the various values of ITYPE is as follows: 0 = group boundaries, 1 = cross sections, 2 = standard deviations, 3 = covariances, and 4 = correlations. The library name is generated within COVR by adding either "-A-" (for a covariance library) or "-B-" (for a correlation library), together with the number of energy groups, to the user-

supplied library name HLIBID. (The ERRORR input option IGN=3 used in this example specifies a built-in 30-group structure). The final seven integers on the header card indicate the number and format of the data values, the number and format of the control integers (called *"m"* and *"-n"* in the discussion in Section XI.B), a data paging flag, and the dimensions of the reconstructed data array, $C(i,j)$.

In order to clarify the process of reconstructing a full matrix from data in Boxer format, we list in Section XI.H a short retrieval program, BOXR. With very few modifications, this program could be incorporated into a sensitivity analysis program, for example, to allow direct access to covariance libraries[6] in this compact format. Alternatively, it could be used to translate COVR libraries into other desired forms. In any case, an examination of the retrieval program should clarify the meaning of the various integer parameters on the header cards in a COVR output library.

F. Error Messages

COVR***REQUESTED TOO MANY CASES.

NCASE is limited to 40. Note that a case refers to one input Card 4, which may request processing of either a single reaction pair or a whole series of reaction pairs. See input instructions for Card 4 and the comments that follow.

EXPNDO***STORAGE EXCEEDED.

Either there is insufficient space in array ISCR to store the cross sections or the total number of reaction pairs in the current case exceeds 300. In the latter instance, it may be necessary to specify the reaction pairs individually (up to 40 per run) in a long series of runs, and then concatenate the numerous output files to form a single COVR output file.

CORR***GROUP STRUCTURES DO NOT AGREE.

This problem should not occur.

COVARD***STORAGE EXCEEDED.

There is insufficient space in array ISCR to store the multigroup data from NIN, the ERRORR output file.

COVARD***DID NOT FIND FILE 33 SUBSECTION FOR MT --- MAT ---

Requested reaction y of the current x-y pair is not found on NIN.

TRUNC***BAD DATA.

Either all cross sections are very small (less than XSLIM = 10^{-4} barn) or EPMIN is larger than the highest group boundary.

MATSHD***IPAT GT 1000.

Maximum number of correlation patterns is 1000. Reduce NDIV to 1 or raise EPMIN in order to reduce the complexity of the correlation plot.

MATSHD***STORAGE EXCEEDED.

NWIG=2*(IXMAX+1)+600 words are available for storing the boundary curve of a constant-correlation region. This will not be exceeded for any practical number of groups IXMAX.

LEVEL***COEFFICIENT = --- OUT OF RANGE.

The absolute magnitude of a computed correlation coefficient is greater than 2. The input covariance file may be faulty.

HISTO***STORAGE EXCEEDED.

This problem should not occur.

ROUND***Z1 (---) NOT FOUND.

This problem should not occur.

FINDS***MAT --- MF --- MT --- NOT ON TAPE.

Requested reaction x of the current x-y pair is not found on NIN.

PRESS***STORAGE EXCEEDED.

This problem should not occur.

PRESS***MATRIX NOT SYMMETRIC. I --- J --- XA(I,J) --- XA (J,I) ---.

Symmetric-matrix format is requested for a matrix that is asymmetric. This problem should not occur.

SETFOR***NVF (= ---) OR NCF (= ---) IS ILLEGAL.

This problem should not occur.

G. Input/Output Units

The following logical units are used:

10 NIN in CORR and COVARD, NSCR in other routines. These units are used to extract either one or two (if MAT1≠MAT) materials from the input covariance tape.

11 NSCR1 in the plot mode. These units are used to document null and small covariance matrices and empty plots.

11/12 NSCR1/NSCR2 in the library mode. In COVARD, the input covariances for the current reaction pair are read from unit 10 and written to NSCR2 (= 12). If the output library is to contain correlation coefficients (MATYPE = 4), then in CORR the covariances are read from NSCR2, and the calculated correlations are written to NSCR1 (= 11). If output covariances are requested (MATYPE = 3), the value of NSCR1 is simply reset to NSCR2 (= 12). In either case, PRESS reads the data from unit NSCR1 and writes the compressed data to NOUT.

20-99 User's choice for NIN and NOUT.

Unit 10 has the same mode as NIN. NOUT, if used, is always formatted. Unit 11 is formatted in the plot option and it is binary in the library option. Unit 12, if used, is always binary.

H. Retrieval Program for COVR Output Libraries

```
      PROGRAM BOXR
C     ********************************************************************
C     FUNCTION OF PROGRAM.  READ DATA FROM UNIT NIN IN THE COMPRESSED
C     *BOXER* FORMAT PRODUCED BY THE COVR MODULE OF NJOY, AND LOAD THE
C     FULL, RECONSTRUCTED MATRIX INTO C(I,J).  THEN WRITE THE RESULT ON
C     UNIT NOUT IN HIGH-TO-LOW ENERGY ORDER.  FAILURE TO FIND A
C     REQUESTED DATA SET RESULTS IN AN ERROR STOP.
C
C       ITYPE   DATA TYPE REQUESTED,
C                 = -1, TO WRITE A TABLE OF CONTENTS OF NIN, IN THE
C                   FORMAT OF BOXR INPUT INSTRUCTIONS, ON UNIT NTAB.
C                 = 0, FOR GROUP BOUNDARIES,
C                 = 1, FOR CROSS SECTIONS,
C                 = 2, FOR STANDARD DEVIATIONS,
C                 = 3, FOR COVARIANCE MATRIX,
C                 = 4, FOR CORRELATION MATRIX, OR (IF MT1 IS ZERO)
C                   TRANSFER MATRIX FROM COVFILS2.
C       ITYPEH  VALUE OF ITYPE ON CURRENT DATA HEADER CARD.
C
C      (MAT,MT,MAT1,MT1)      REQUESTED REACTION PAIR.
C      (MATH,MTH,MAT1H,MT1H) CURRENT REACTION PAIR.
C      (XVAL(IV),IV=1,NVAL)  DATA VALUE ARRAY IN THE *BOXER* FORMAT.
C       NVMAX                MAXIMUM ALLOWABLE VALUE OF NVAL.
C      (ICON(IC),IC=1,NCON)  CONTROL-PARAMETER ARRAY IN *BOXER* FORMAT.
C       NCMAX                MAXIMUM ALLOWABLE VALUE OF NCON.
C
C       I       ROW INDEX OF MATRIX C(I,J), NORMALLY THE ENERGY GROUP
C                 OF THE REACTION (MAT,MT).
C       NROW    NUMBER OF ROWS IN C(I,J).
C       NROWH   VALUE OF NROW ON DATA HEADER CARD.
C       NROWM   CONTINUATION FLAG, = 0 FOR FINAL DATA BLOCK OF CURRENT
C                 REACTION PAIR.
C       J       COLUMN INDEX OF C(I,J), FOR MATRIX DATA THE ENERGY GROUP
C                 OF THE REACTION (MAT1,MT1), = 1 FOR VECTORS.
C       NCOL    NUMBER OF COLUMNS IN C(I,J).
C       NCOLH   VALUE OF NCOL ON DATA HEADER CARD, = 0 IF C(I,J) IS A
C                 SYMMETRIC MATRIX REPRESENTED IN THE *BOXER* FORMAT BY
C                 JUST THE UPPER RIGHT TRIANGLE (J.GE.I).
C       NGMAX   MAXIMUM ALLOWABLE VALUE OF NROW AND NCOL.
C     ********************************************************************
      DIMENSION IVFT(3), ICFT(3), IA(9), IB(9)
      DIMENSION C(200,200), CR(200), XVAL(880), ICON(900)
      DATA NGMAX /200/, NVMAX /880/, NCMAX /900/, IDASH /4H----/
      DATA NINPUT /5/, NIN /20/, NOUT /21/, NTAB /22/, IZERO /0/
C
C     ***READ USER-SUPPLIED REACTION-TYPE AND MAT-MT INFORMATION.
   10 READ (NINPUT,190) ITYPE,MAT,MT,MAT1,MT1
C     INPUT IS TERMINATED BY ENTERING (0,0).
      IF (ITYPE.GE.0.AND.MAT.EQ.0) STOP
C
```

```
C     ***RETRIEVE REQUESTED DATA FROM UNIT NIN.
   20 READ (NIN,210,END=900) ITYPEH,(IA(K),K=1,9),MATH,MTH,MAT1H,MT1H
     1 ,NVAL,NVF,NCON,NCF,NROWM,NROWH,NCOLH
C
C     ***IN COVFILS2, ITYPEH=9 IS USED AS A TERMINATOR.
   30 IF (ITYPEH.EQ.9) GO TO 900
      IF (ITYPE.EQ.-1.AND.MATH.EQ.0) MATH=1
      IF (ITYPE.EQ.-1.AND.IA(2).NE.IDASH)
     1 WRITE (NTAB,190) ITYPEH,MATH,MTH,MAT1H,MT1H
      IF (NVAL.GT.NVMAX) STOP 3
      IF (NCON.GT.NCMAX) STOP 4
C     SET FORMATS, THEN READ BOXER DATA.
      CALL SETFOR (NVF,NCF,NOUT,IVFT,ICFT)
      IF (NVAL.GT.0) READ (NIN,IVFT) (XVAL(K),K=1,NVAL)
      IF (NCON.GT.0) READ (NIN,ICFT) (ICON(K),K=1,NCON)
C     TEST IF THESE ARE THE DESIRED DATA.  FOR THE GROUP BOUNDARIES
C     ONLY, THE VALUES MATH, MTH, MAT1H, AND MT1H ON THE HEADER CARD
C     ARE IGNORED.  FOR ITYPE=1 OR 2, MAT1H AND MT1H ARE IGNORED.
C     FOR MATRIX DATA, ITYPE=3 OR 4, A COMPLETE MATCH IS REQUIRED.
      IF (ITYPEH.NE.ITYPE) GO TO 20
      IF (ITYPE.EQ.0) GO TO 40
      IF (MATH.NE.MAT.OR.MTH.NE.MT) GO TO 20
      IF (ITYPE.EQ.1.OR.ITYPE.EQ.2) GO TO 40
      IF (MAT1H.NE.MAT1.OR.MT1H.NE.MT1) GO TO 20
C     INITIALIZE
   40 NX=0
      I=1
      ISTART=1
      J=0
   50 NROW=NROWH
      NCOL=NCOLH
      IV=0
      NSYM=0
      IF (NCOL.EQ.0) NSYM=1
      IF (NCOL.EQ.0) NCOL=NROW
      IF (NROW.GT.NGMAX.OR.NCOL.GT.NGMAX) GO TO 910
C
C     ***LOAD DATA FROM XVAL INTO C(I,J) AS DIRECTED BY ICON.
      DO 110 IC=1,NCON
      IF (ICON(IC).LT.0) GO TO 60
      IF (ICON(IC).EQ.0) STOP 6
      NLOAD=ICON(IC)
      GO TO 70
   60 NLOAD=-ICON(IC)
      IV=IV+1
      IF (IV.GT.NVAL) STOP 7
      CLOAD=XVAL(IV)
   70 CONTINUE
      DO 100 N=1,NLOAD
      J=J+1
      IF (J.LE.NCOL) GO TO 80
C     START NEW ROW
      I=I+1
```

```
      IF (I.GT.NROW) STOP 10
      J=1
      IF (NSYM.EQ.1) J=I
   80 IF (ICON(IC).LE.0) GO TO 90
      CLOAD=0.
      IF (I.GT.ISTART) CLOAD=C(I-1,J)
   90 C(I,J)=CLOAD
      NX=NX+1
      IF (NSYM.EQ.0.OR.I.EQ.J) GO TO 100
      C(J,I)=CLOAD
      NX=NX+1
  100 CONTINUE
  110 CONTINUE
      IF (NROWM.EQ.0) GO TO 120
C     READ IN NEW PAGE OF DATA FROM NIN
      IF (J.NE.NCOL) STOP 11
      READ (NIN,210) ITYPEH,(IB(K),K=1,9),MATH,MTH,MAT1H,MT1H,NVAL,NVF
     1 ,NCON,NCF,NROWM,NROWH,NCOLH
      CALL SETFOR (NVF,NCF,NOUT,IVFT,ICFT)
      IF (NVAL.GT.0) READ (NIN,IVFT) (XVAL(K),K=1,NVAL)
      IF (NCON.GT.0) READ (NIN,ICFT) (ICON(K),K=1,NCON)
      ISTART=I+1
      GO TO 50
  120 CONTINUE
C     FINISHED LOADING C(I,J)
      IF (NX.NE.NROW*NCOL) STOP 12
C
C     ***WRITE C(I,J) TO NOUT IN HIGH-TO-LOW ENERGY ORDER.
      WRITE (NOUT,220)
      WRITE (NOUT,210) ITYPEH,(IA(K),K=1,9),MATH,MTH,MAT1H,MT1H,NVAL,NVF
     1 ,NCON,NCF,NROWM,NROWH,NCOLH
      IF (NCOL.EQ.1) GO TO 150
      DO 140 I=1,NROW
      IR=NROW+1-I
C     IN COVFILS2, TRANSFER MATRICES ARE ALREADY IN HIGH-TO-LOW ORDER
      IF (ITYPE.EQ.4.AND.MT1.EQ.0) IR=I
      DO 130 J=1,NCOL
      JR=NCOL+1-J
      IF (ITYPE.EQ.4.AND.MT1.EQ.0) JR=J
  130 CR(J)=C(IR,JR)
  140 WRITE (NOUT,200) (CR(L),L=1,NCOL)
      GO TO 10
  150 DO 160 I=1,NROW
      IR=NROW+1-I
  160 CR(I)=C(IR,1)
      WRITE (NOUT,200) (CR(K),K=1,NROW)
      GO TO 10
C
C     ***PRINT ERROR MESSAGES.
  900 IF (ITYPE.NE.-1) WRITE (NOUT,901) ITYPE,MAT,MT,MAT1,MT1
  901 FORMAT (/50H ***ERROR IN BOXR***CANNOT FIND ITYPE,MAT,MT,MAT1,
     1 ,5HMT1 =,5I5)
      IF (ITYPE.EQ.-1) WRITE(NTAB,190) IZERO,IZERO
```

```
      STOP
  910 WRITE (NOUT,911) NGMAX
  911 FORMAT (/45H ***ERROR IN BOXR*** NUMBER OF GROUPS EXCEEDS,I4)
      STOP
C
  190 FORMAT (5I6)
  200 FORMAT (1P8E10.3)
  210 FORMAT (I1,A3,8A4,2(I5,I4),2(I4,I3),3I4)
  220 FORMAT (///)
      END
      SUBROUTINE SETFOR (NVF,NCF,NOUT,IVFT,ICFT)
C     ******************************************************************
C     SET *BOXER* INPUT/OUTPUT FORMATS.
C     ******************************************************************
      DIMENSION IFT(3,14), IVFT(3), ICFT(3)
      DATA IFT /4H(80I,4H1)  ,4H    ,4H(40I,4H2)  ,4H    ,4H(26I,4H3)  ,
     1 4H    ,4H(20I,4H4)  ,4H    ,4H(16I,4H5)  ,4H    ,4H(13I,4H6)  ,4H
     2    ,4H(11F,4H7.4),4H    ,4H(10F,4H8.5),4H    ,4H(1P8,4HE9.2,4H)
     3 ,4H(1P8,4HE10.,4H3)  ,4H(1P7,4HE11.,4H4)  ,4H(1P6,4HE12.,4H5)  ,4
     4 H(1P6,4HE13.,4H6)  ,4H(1P5,4HE14.,4H7)  /
C
      IF (NVF.LT.7.OR.NVF.GT.14) GO TO 900
      IF (NCF.LT.1.OR.NCF.GT.6) GO TO 900
C
C     ***SET FORMATS
      DO 10 I=1,3
      IVFT(I)=IFT(I,NVF)
   10 ICFT(I)=IFT(I,NCF)
      RETURN
C
C     ERROR MESSAGE
  900 WRITE (NOUT,901) NVF,NCF
  901 FORMAT (/28H ***ERROR IN SETFOR***NVF (=,I3,11H) OR NCF (=,I3,13H)
     1 IS ILLEGAL.)
      STOP
      END
```

I. References

1. S. A. W. Gerstl, "SENSIT: A Cross-Section and Design Sensitivity and Uncertainty Analysis Code," Los Alamos Scientific Laboratory report LA-8498-MS (August 1980).

2. Mark J. Embrechts, "SENSIT-2D: A Two-Dimensional Cross-Section Sensitivity and Uncertainty Analysis Code," Los Alamos National Laboratory report LA-9515-MS (October 1982).

3. D. W. Muir and R. J. LaBauve, "COVFILS: A 30-Group Covariance Library Based on ENDF/B-V," Los Alamos National Laboratory report LA-8733-MS (ENDF-306) (March 1981).

4. D. W. Muir, R. E. MacFarlane, and R. M. Boicourt, "Multigroup Processing of ENDF/B Dosimetry Covariances," Proc. 4th ASTM-EURATOM Symp. on Reactor Dosimetry, Gaithersburg, Maryland, March 22-26, 1982, NUREG/CP-0029 (CONF-820321), p. 655 (1982).

5. "DISSPLA (Display Integrated Software System and Plotting Language): User's Manual, Version 9.2," Integrated Software Systems Corporation, 4186 Sorrento Valley Blvd., San Diego, California 92121.

6. D. W. Muir, "COVFILS-2: Neutron Data and Covariances for Sensitivity and Uncertainty Analysis," *Fusion Tech.* **10** (3), Part 2B, 1461 (November 1986).

XII. MODER

The MODER module is used to convert ENDF, PENDF, and GENDF tapes from the NJOY blocked-binary mode to formatted mode (BCD, EBCDIC, ASCII, etc.) and *vice versa*. It can also be used to copy data from one logical unit to another without change of mode, or to make a new tape containing selected materials from one or more ENDF, PENDF, or GENDF tapes. MODER handles all ENDF/B-IV, ENDF/B-V and ENDF/B-VI formats, plus special-purpose formats developed for NJOY, such as the GROUPR and ERRORR output formats and the special thermal File 6 formats. This chapter describes version 91.0.

A. Code Description

At the beginning of execution, MODER rewinds the output tape **NOUT**. Additionally, each time a new input tape **NIN** is specified, that unit is rewound. MODER then processes **NIN** one file at a time, either for all materials on **NIN**, or optionally (see following section) for a single specified material. As each file is identified, the main program calls a subroutine dedicated to that file. Each subroutine makes the series of calls to **CONTIO**, **LISTIO**, etc., that is appropriate to that file.

If **NIN** and **NOUT** are of opposite sign, then mode conversion is performed automatically by the utility I/O subroutines. If **NIN** and **NOUT** have the same sign, then no mode conversion is performed; runs of this type can be used simply to make an extra copy of the input tape or to retrieve selected materials without mode change. Only a little more than one page of scratch storage is needed, so there are no limitations on which tapes can be processed.

B. Input Instructions

As an aid to discussions of the user input to MODER, the input instructions that appear as comment cards at the beginning of the current version of this module are listed below. Since code changes are possible, it is always advisable to consult the comment-card instructions contained in the version of the code actually being used before proceeding with an actual calculation.

```
*---INPUT SPECIFICATIONS (FREE FORMAT)-------------------------*
*                                                              *
* CARD 1       UNIT NUMBERS                                    *
*      NIN     INPUT UNIT                                      *
*      NOUT    OUTPUT UNIT                                     *
*                                                              *
*  A POSITIVE UNIT IS CODED (MODE 3).                          *
*  A NEGATIVE UNIT IS BLOCKED BINARY (NJOY MODE).              *
```

```
*                                                                    *
*  NOTE: ABS(NIN) GE 1 AND LE 19 IS A FLAG TO SELECT VARIOUS         *
*        MATERIALS FROM ONE OR MORE INPUT TAPES, WITH OR             *
*        WITHOUT MODE CONVERSION.  THE KIND OF DATA TO BE            *
*        PROCESSED IS KEYED TO NIN AS FOLLOWS:                       *
*             NIN=1, FOR ENDF OR PENDF INPUT AND OUTPUT,             *
*                 2, FOR GENDF INPUT AND OUTPUT,                     *
*                 3, FOR ERRORR-FORMAT INPUT AND OUTPUT.             *
*                                                                    *
*      CARDS 2 AND 3 FOR ABS (NIN) GE 1 AND LE 19 ONLY.              *
*                                                                    *
* CARD 2                                                             *
*      TPID    TAPEID FOR NOUT.  66 CHARACTERS ALLOWED               *
*              (DELIMITED WITH *, ENDED WITH /)                      *
* CARD 3                                                             *
*      NIN     INPUT UNIT                                            *
*              TERMINATE MODER BY SETTING NIN=0                      *
*      MATD    MATERIAL ON THIS TAPE TO ADD TO NOUT                  *
**********************************************************************
```

The contents of **NIN** and **NOUT** are positive or negative logical unit numbers, with absolute magnitudes normally in the range 20–99, inclusive. Positive unit numbers refer to formatted tapes, and negative unit numbers refer to blocked-binary tapes. No other input is required to copy or convert the entire contents of the data file on unit **NIN**, writing the results to unit **NOUT**.

A positive value of **NIN** in the range 1–3 is used as a trigger to specify that the data to be copied or converted are not the contents of a single tape, but, instead, they selected materials from one or more input tapes. The type of data to be processed (ENDF/PENDF vs. GENDF vs. ERRORR-format) is keyed to the value of **NIN**, as detailed in the instructions above. If **NIN** is in the range 1–3, and only in this case, additional input is supplied to specify (on card 2) the tape-identification information to be written on the first record of the output tape and to specify (on card 3) both the MAT-numbers of the materials to be included and the logical units where each of the desired materials are to be found. Note that the slash terminating the Hollerith information on card 2 is required. In the case of GENDF processing of a material MATD, which is present on the specified input tape at a series of temperatures, a single card 3 causes the retrieval of all temperatures. Card 3 is repeated as many times as needed, and input is terminated with a card containing 0/.

C. Sample Input

It is good practice to convert the mode of the ENDF/B tape before proceeding with any NJOY run. The time spent in MODER is normally much less than the

time saved by the subsequent modules. The required input for this is extremely simple.

```
0
5
MODER / MOUNT FORMATTED ENDF TAPE ON UNIT 20
20 -21
RECONR
-21 -22
...
```

Although it requires 3 additional input cards, the following will result in even greater efficiency, since in this case unit 21 will contain only the single needed material.

```
0
5
MODER / MOUNT FORMATTED ENDF TAPE ON UNIT 20
1 -21
*B-10*/
20 1305
0/
RECONR
-21 -22
...
```

The final example (taken from one of the standard sample problems) shows the use of MODER to prepare a special multimaterial ENDF tape for a covariance calculation involving the 5 primary fissionable isotopes. Since, in this particular example problem, the resonance region is of no interest, a copy of the ENDF serves as the PENDF for later modules.

```
0
5
MODER / MONT ENDF TAPE 515, 516, 555 ON UNITS 20, 21, 22
1 -23
*ENDF/B-V NUBAR COVARIANCE MATERIALS*/
20 1380
20 1381
21 1390
22 1395
22 1398
20 1399
0/
MODER / COPY ENDF FOR USE AS A PENDF
-23 -24
GROUPR / PREPARE GENDF WITH MULTIGROUPED NUBARS
-23 -24 0 25
...
```

D. Error Messages

`ERROR IN MODER***ENDF MATERIALS MUST BE IN ASCENDING ORDER`

This is a problem with the material ordering for the input tape.

`MESSAGE FROM MODER***MAT nnnn NOT FOUND ON GENDF TAPE`

Check the `MATD` value on input card 3 and make sure that the correct input tape was mounted.

`ERROR IN MODER***THIS MATERIAL IS NOT A GENDF MATERIAL`

Input file contains an illegal mixture of data, namely, an initial GENDF material, followed by the indicated non-GENDF MAT.

`ERROR IN MODER***INPUT IS NOT AN ERRORR OUTPUT TAPE`

User has requested ERRORR-format processing, but input data file is not an ERRORR-format tape.

`ERROR IN MODER***INPUT IS NOT AN ENDF OR PENDF TAPE`

User has specified `NIN`=1 on card 1, thereby requesting selective multitape ENDF or PENDF processing, but input data file on the unit `NIN` specified on card 3 is not an ENDF/PENDF file.

`ERROR IN MODER***INPUT IS NOT AN ENDF TAPE`

See comments above.

`ERROR IN MODER***INPUT IS NOT A GENDF TAPE`

User has requested GENDF processing, but input data file is not a GENDF tape.

`ERROR IN MODER***CONVERSION NOT CODED FOR MF=nn`

There is an illegal or unrecognizable MF value on NIN.

`ERROR IN MODER***SHOULD HAVE FOUND SEND CARD`

MODER is lost. The listed data display the contents of the last card read. Input data file may be bad.

`ERROR IN MODER***ILLEGAL COVARIANCE MF=nn`

ERRORR-format file is missing the required MF=3. ENDF and PENDF tapes must be MAT ordered.

`ERROR IN FILE2***ILLEGAL MT`

Only the standard MT151 and the NJOY special value of MT152 are allowed in File 2.

`ERROR IN FILE5***ILLEGAL LF`

The File 5 LF value is outside the legal range 1–12.

`ERROR IN FILE6***ILLEGAL LTT`

This message comes from the branch used for ENDF-5 format or for thermal data generated by the THERMR module. Check the format of the File 6 sections.

ERROR IN FILE6*ILLEGAL ENDF6 LAW**

This message comes from the branch used for ENDF-6 tapes. Check the values for the law parameter in the sections of File 6.

ERROR IN FILE7*ILLEGAL MT=nnn**

Only MT=2 and MT=4 are allowed in File 7 for the ENDF-6 format.

ERROR IN FILE7*ILLEGAL VALUE OF LTHR=n**

Only values of 1 or 2 are allowed for **LTHR** in ENDF-6.

XIII. DTFR

This module is used to prepare libraries for discrete-ordinates transport codes that accept the format designed for the S_N code DTF-IV.[1] Since this code was early, many newer S_N and diffusion codes allow DTF format as an input option. DTFR also contains a simple plotting capability for providing a quick look at its output. The graphs produced are not publication quality, but the plotting system is easy to move to different computer systems. DTFR was the first output module written for the NJOY system, and it has been largely superseded by the MATXS/TRANSX system. The newer PLOTR module certainly produces more attractive graphs. But DTFR is still useful for some purposes because of its simplicity. This chapter describes version 91.0.

A. Transport Tables

Transport tables in DTF format are organized to mirror the structure of the data inside a discrete-ordinates transport code. These codes start with the highest energy group and work downward. (The conventional order for group indices is to increase as energy decreases.) Therefore, for each group g, the data required are the reaction cross sections for group g and the scattering cross sections to g from other groups g'. Each data element is said to occupy a position in the table for group g. The organization is shown in Table 1.

Table 1: Organization of Data for One Group in a Transport Table

Position	Meaning
1	edit cross sections
...	(if any)
IPTOTL-3	last special edit
IPTOTL-2	particle-balance absorption
IPTOTL-1	fission neutron production
IPTOTL	total cross section
IPTOTL+1	upscatter $g \leftarrow g'$ $(g' > g)$
...	(if any)
IPINGP	ingroup scattering $g \leftarrow g$
...	downscatter $g \leftarrow g'$ $(g' < g)$
ITABL	end of table

The basic table consists of the three standard edits, namely, particle balance absorption (σ_a), fission neutron production cross section ($\bar{\nu}\sigma_f$), and total cross section (σ_t). These standard edits are followed by the group-to-group scattering cross sections. If desired, the standard edits can be preceded by IPTOTL-3 special

edits, which can be used in the transport code to calculate various responses of the system (for example, heating, activation, or gas production). If IPTOTL is the position of the total cross section, the positions of the three standard edits will be IPTOTL-2, IPINGP-1, and IPTOTL, respectively. The positions of the special edits (if IPTOTL>3) will be 1, 2, ... IPTOTL-3.

Most transport tables describe only downscatter. In such cases, the position of the ingroup element is IPINGP=IPTOTL+1. Position IPTOTL+1 would contain $\sigma_{g\leftarrow(g-1)}$, and so on. If thermal upscatter is present, the NUP upscatter cross sections are between the total cross section element and the ingroup element. Therefore, position IPTOTL-1 contains $\sigma_{g\leftarrow(g+1)}$, and so on. The position parameters must satisfy the following conditions:

$$\text{IPTOTL} \geq 3 \quad \text{, and}$$
$$\text{IPINGP} = \text{IPTOTL} + \text{NUP} + 1 \ .$$

The number of positions for a group is called the table length. A full table will have the table length given by

$$\text{ITABL} = \text{IPTOTL-3} + 3 + \text{NUP} + \text{NG} \quad ,$$

where NG is the total number of groups in the set. Table lengths can be truncated in some cases. If this is done correctly, the important cross sections will be conserved, and valid results can still be obtained.

An example of a transport table is given in Table 2. This table was generated with DTFR for ENDF/B-VI ^{235}U (MAT=9228). It contains three special edits: the (n,2n) cross section, the fission cross section, and the radiative capture cross section. These are followed by the three standard edits, and then by the ingroup scattering for group 1. Since this is the highest energy group, there is no downscatter to this group from groups above, and the rest of the positions are filled with zeros. Group 2 starts on line 8 with the six edit cross sections. The seventh number is the ingroup cross section, and the eighth is the scattering from group 1 to group 2. Continuing to lines 14 and 15, there are now two downscatter elements: two to three and one to three. Note that there is an entire table for each Legendre order, and that each table has a header card that describes its contents. The ellipses were added to mark removed lines.

The last table in this example describes the photon production matrix. There are 30 neutron groups and 12 photon groups. The photon group index replaces the position index used in the neutron tables. Therefore, the first 12 numbers in

Table 2: Example of a Transport Table with Internal Edits

```
IL= 1 TABLE 30 GP 36 POS, MAT= 9228 IZ= 1 TEMP= 3.00000E+02
 3.2597E-01  2.0823E+00  6.1629E-04  1.3965E+00  9.7755E+00  5.9989E+00
 3.1734E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 4.9950E-01  2.0532E+00  1.1756E-03  1.4681E+00  9.1321E+00  5.8659E+00
 3.0495E+00  1.1523E-01  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 6.8192E-01  1.8830E+00  1.5668E-03  1.1928E+00  8.0521E+00  5.7877E+00
 2.9838E+00  8.2484E-02  3.1214E-02  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
...
IL= 2 TABLE 30 GP 36 POS, MAT= 9228 IZ= 1 TEMP= 3.00000E+02
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 2.8003E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 2.5972E+00  3.8535E-02  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 2.4730E+00  2.3851E-02  7.4855E-03  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
...
IP= 1 TABLE 30 GP 12 POS, MAT= 9228 IZ= 1 TEMP= 3.00000E+02
 0.0000E+00  0.0000E+00  1.1808E-02  2.0992E-02  2.4928E-02  5.7726E-02
 6.0219E-01  1.8328E+00  5.5365E+00  7.5176E+00  1.0018E+01  2.0404E+00
 0.0000E+00  0.0000E+00  1.0938E-02  1.9445E-02  2.4306E-02  5.8998E-02
 5.7158E-01  1.7441E+00  5.2857E+00  7.2063E+00  9.5514E+00  1.9537E+00
 0.0000E+00  0.0000E+00  8.6352E-03  1.5352E-02  2.6314E-02  7.8974E-02
 5.3190E-01  1.6485E+00  5.0937E+00  7.1114E+00  9.1307E+00  1.9158E+00
 0.0000E+00  0.0000E+00  5.0549E-03  8.9865E-03  2.9371E-02  1.0974E-01
 4.6950E-01  1.4973E+00  4.7871E+00  6.9512E+00  8.4624E+00  1.8534E+00
....
```

this table correspond to photon production from neutron group 1. The photon groups are arranged in order of decreasing energy.

The header lines at the start of each table in this example give the Legendre order, number of groups, number of positions, MAT number, σ_0 number, and temperature.

DTFR will also produce a variant of the transport tables that was used for Los Alamos libraries in years past. These are sometimes called "CLAW Libraries" after the CLAW-III and CLAW-IV libraries available from the Radiation Shielding Information Center (RSIC) at the Oak Ridge National Laboratory. Although CLAW-IV uses a version of this format, it was actually generated using MATXS cross sections and the TRANSX code.[2, 3] They key feature of the CLAW tables is that the edits are removed from their normal position at the beginning of each group in the transport table and written out on separate lines. The format specified a particular list of reactions (see Table 3) that is defined using data statements in DTFR. In addition, thermal upscatter is not allowed in this format. An example of this style of transport table is given in Table 4. Note that eye-readable identifiers were added to the right-hand edge of each card by DTFR. The card labels contain the first two letters of the material name, the reaction name or the Legendre order, and a sequence number. The format of the header lines at the start of each table is different from the last example. The quantity in parentheses on the "EDIT XSEC" card is groups by number of edit reactions. For the neutron tables, it is table length by groups. For the gamma table, it is gamma groups by neutron groups.

B. Data Representations

The data stored into the transport tables are obtained from GROUPR. Since all the scattering matrix reactions are kept separate in the multigroup processing, it is necessary to add them up to compute the DTF scattering matrix. They are obtained from the sections with MF=6 on the input GENDF tape. Similarly, all the photon production matrices have to be added up to obtain the final DTF photon production table. These sections have MF=16.

The standard edit called particle-balance absorption is used in discrete-ordinates transport codes to calculate the balance table. The most fundamental definition for this quantity is

$$\sigma_{a,g} = \sigma_{t,g} - \sum_{g'} \sigma_{s,g' \leftarrow g} \ . \tag{1}$$

The total cross section is obtained from MF3,MT1 on the input GENDF tape.

Table 3: Predefined Edits for DTFR

Reaction Name	Position	Reaction	Multiplicity
ELS	1	2	1
INS	2	4	1
N2N	3	16	1
	3	6	1
	3	7	1
	3	8	1
	3	9	1
N3N	4	17	1
NGM	5	102	1
NAL	6	107	1
NP	7	103	1
FDIR	8	19	1
NNF	9	20	1
N2NF	10	21	1
FTOT	11	18	1
ND	12	104	1
NT	13	105	1
NHE3	14	106	1
N2P	15	111	1
NPA	16	112	1
NT2A	17	113	1
ND2A	18	114	1
N2A	19	108	1
N3A	20	109	1
NND	21	32	1
NND2A	22	35	1
NNHE3	23	34	1
NNA	24	22	1
N2NA	25	24	1
NN3A	26	23	1
N3NA	27	25	1
NNP	28	28	1
NN2A	29	29	1
N2N2A	30	30	1
NNT	31	33	1
NNT2A	32	36	1
N4N	33	37	1
N3NF	34	38	1
CHI	35	470	1
CHID	36	471	1
NUD	37	455	1
PHI	38	300	1
THEAT	39	301	1
	39	443	0
KERMA	40	443	1
TDAME	41	444	1
NUSF	42	1	1
TOTL	43	452	1

Table 4: Example of DTFR Transport Tables Using Separate Edits

```
U235       EDIT XSEC ( 30X 48) PROC BY MJOY1  ON 05/03/90
 3.03110E+00 2.85141E+00 2.75464E+00 2.66464E+00 2.87622E+00 3.56158E+00U2  ELS1
 4.43195E+00 4.38571E+00 3.94779E+00 3.53864E+00 3.34261E+00 3.61868E+00U2  ELS2
 4.74566E+00 6.13164E+00 7.65700E+00 9.34710E+00 1.08466E+01 1.16438E+01U2  ELS3
 1.19400E+01 1.28024E+01 1.32190E+01 1.31520E+01 1.29202E+01 1.30545E+01U2  ELS4
 1.27990E+01 1.19229E+01 1.32489E+01 1.41192E+01 1.47783E+01 1.54225E+01U2  ELS5
 3.78514E-01 4.17166E-01 4.61638E-01 5.44816E-01 8.11127E-01 1.37822E+00U2  INS1
 2.20958E+00 2.30405E+00 2.34199E+00 2.28404E+00 2.14540E+00 1.90879E+00U2  INS2
 1.60448E+00 1.29695E+00 9.63151E-01 4.00064E-01 4.54497E-02 3.03043E-03U2  INS3
 1.87467E-06 7.26072E-08 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2  INS4
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2  INS5
 3.25966E-01 4.99502E-01 6.81925E-01 8.22064E-01 6.05949E-01 3.47278E-01U2  N2N1
 8.86188E-03 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2  N2N2
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2  N2N3
 ...
U235   L=0 N-N TABLE ( 33X 30)
 1.39646E+00 9.77549E+00 5.99892E+00 3.17345E+00 0.00000E+00 0.00000E+00U2 0   1
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 0   2
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 0   3
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 0   4
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 0   5
 0.00000E+00 0.00000E+00 0.00000E+00 1.46808E+00 9.13206E+00 5.86594E+00U2 0   6
 3.04950E+00 1.15234E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 0   7
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 0   8
 ...
U235   L=1 N-N TABLE ( 33X 30)
 0.00000E+00 0.00000E+00 0.00000E+00 2.80033E+00 0.00000E+00 0.00000E+00U2 1   1
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 1   2
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 1   3
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 1   4
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 1   5
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 1   6
 2.59724E+00 3.85346E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 1   7
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 1   8
 ...
U235   L=0 N-P TABLE ( 12X 30)
 0.00000E+00 0.00000E+00 1.18077E-02 2.09916E-02 2.49277E-02 5.77264E-02U2 0   1
 6.02189E-01 1.83282E+00 5.53653E+00 7.51757E+00 1.00182E+01 2.04037E+00U2 0   2
 0.00000E+00 0.00000E+00 1.09379E-02 1.94453E-02 2.43062E-02 5.89981E-02U2 0   3
 5.71583E-01 1.74411E+00 5.28569E+00 7.20630E+00 9.55138E+00 1.95373E+00U2 0   4
 0.00000E+00 0.00000E+00 8.63521E-03 1.53515E-02 2.63136E-02 7.89737E-02U2 0   5
 5.31904E-01 1.64847E+00 5.09371E+00 7.11139E+00 9.13067E+00 1.91577E+00U2 0   6
 0.00000E+00 0.00000E+00 5.05490E-03 8.98652E-03 2.93715E-02 1.09745E-01U2 0   7
 ....
```

The scattering matrix is obtained by adding all the matrix reactions found in File 6 on the input tape. The absorption edit is often written in the form

$$\sigma_{a,g} = \sigma_{\gamma,g} + \sigma_{f,g} - \sigma_{n2n,g} \ , \tag{2}$$

which is good up to the threshold for other multiparticle reactions. Because of the presence of the (n,2n) term, the σ_a parameter is not equal to the real neutron absorption for high-energy groups. In fact, it is often negative.

The next standard edit is used to compute the fission neutron production rate when constructing a fission source. It is used together with a fission spectrum, which can be included in the specials edits (see below). The fission contributions from the GENDF tape are complicated. First, the prompt fission matrix may be given in MT18 (total fission), or in the partial fission reactions MT19, MT20, MT21, and MT38, which stand for (n,f), (n,n'f), (n,2nf), and (n,3nf). Second, each of these matrices may take advantage of the fact that the shape of the fission spectrum is constant at low energies. Thus, fission can be represented using single fission χ_g^{LE} vector with this shape at low energies, with an accompanying fission neutron production cross section σ_{Pfg}^{LE} at low energies, and with a rectangular fission matrix $\sigma_{g,g\to g'}^{HE}$ at high energies. The third complication is the presence of delayed fission neutron emission. The delayed neutron yield $\bar{\nu}_g^D$ is retrieved from MF3,MT455, and the delayed neutron spectrum χ_g^D is obtained by adding up the spectra for the six time groups in MF5,MT455. The following equation shows how these separate terms are combined into the steady-state fission neutron production edit:

$$\bar{\nu}_g^{SS}\sigma_{fg} = \sum_{g'} \sigma_{fg\to g'}^{HE} + \sigma_{Pfg}^{LE} + \bar{\nu}_g^D \sigma_{fg} \ . \tag{3}$$

The associated steady-state fission spectrum is given by

$$\chi_{g'}^{SS} = \frac{\sum_g \sigma_{fg\to g'}^{HE}\phi_g + \chi_{g'}^{LE}\sum_g \sigma_{Pfg}^{LE}\phi_g + \chi_{g'}^{D}\sum_g \bar{\nu}_g^D \sigma_{fg}\phi_g}{\text{NORM}} \ , \tag{4}$$

where NORM is the quantity that will normalize the spectrum, namely, the sum of the numerator over all g'.

The total cross section is read directly from MF3,MT1. No transport corrections are made. The normal convention for libraries made with DTFR in the past was to supply P_4 tables and let the application code construct transport-corrected P_3 tables if it wanted them.

A flexible scheme is provided for constructing special edit cross sections. Each position can contain any of the cross sections available in File 3 of the GENDF

tape, or a position can contain a combination of several cross sections weighted by multiplicities. For example, the ENDF/B-IV evaluation for ^{12}C contains the two reactions (n,α) (in MF3,MT107) and (n,n')3α (in MF3,MT91). To obtain the helium production cross section, it is only necessary to request an edit made up as follows:

$$1 \times \text{MT107} + 3 \times \text{MT91} .$$

Most of the reactions for these special edits are requested by giving their ENDF MT numbers. The following special MT values are used to request some special quantities:

Special MT	Meaning
300	weighting flux (from MF3,MT1)
455	delayed neutron yield (MF3,MT455)
470	steady-state spectrum (see Eq. 4)
471	delayed neutron spectrum (MF5,MT455)

See Section XIII.D on user input for more details.

DTFR can also construct transport tables suitable for calculations in the thermal range. These tables allow for upscatter, and the user can specify that bound scattering cross sections be used in the thermal range instead of the normal static elastic scattering reaction. The NJOY thermal capabilities are described in more detail in the THERMR and GROUPR sections of this report. In brief, NJOY can provide thermal scattering cross sections and matrices for any material treated as a free gas in thermal equilibrium (a Maxwell-Boltzmann distribution), or it can provide data for a number of important moderator materials whose scattering laws $S(\alpha,\beta)$ are available in the ENDF libraries. These thermal "materials" look like different reactions for the dominant scattering isotope after being processed by GROUPR. Table 5 lists the different binding states available and the MT numbers used to request them. Materials like "H in H_2O" give the scattering for the principal scattering isotope only; the other isotope should be treated using free-gas scattering. There are two exceptions: benzene and BeO were evaluated as complete molecules, and the results were renormalized to be used with the cross section for the dominant isotope (H and Be, respectively). Therefore, these two materials should be used with the density corresponding to the dominant isotope, and no thermal contribution from the other isotope should be included at all (set both `MTI` and `MTC` to zero).

As described in more detail in the subsection on user input, DTFR has an input parameter called `NTHERM` that specifies the number of incident-energy groups to be

Table 5: Thermal Reactions Available to DTFR When Using the Normal ENDF/B Thermal Evaluations

MTI	MTC	Thermal Binding Condition
221		free gas
222		H in H_2O
223	224	H in polyethylene (CH_2)
225	226	H in ZrH
227		Benzene
228		D in D_2O
229	230	C in graphite
231	232	Be metal
233	234	BeO
235	236	Zr in ZrH

treated using thermal cross sections; this parameter determines the breakpoint between the thermal treatment and the normal static treatment. DTFR subtracts the static elastic cross section (MT2) from both the total and absorption edits in this group range, and it adds the **MTI** and **MTC** cross sections to these positions. Similarly, it omits the contributions from the MT2 matrix for these final-energy groups, and it adds in the contributions from **MTI** and **MTC**. The user is free to use a number of upscatter positions less than **NTHERM**. The code will truncate the table in a way that preserves the total thermal scattering cross section.

C. Plotting

Plots of the output data from a formatting program like DTFR are useful in two ways: first, they provide a nice summary of the library and help its users to understand the trends in the data easily, and second, they are helpful in quality control as an aid to finding errors in processing. The DTFR plotting capability was originally developed before modern graphics systems like DISSPLA became widely available, and the graphs look crude by current standards. However, it is very easy to transport this capability to a variety of different graphics systems. In addition, the current capability still fulfills the two reasons for plotting mentioned above.

DTFR automatically makes two-dimensional log-log graphs for all the special edit positions and the three standard edit positions. If available (see **NPEND**), pointwise cross sections are plotted on the same frame as the group cross sections. If the multigroup cross section is a combination of several reactions, the pointwise cross sections for all of the components are plotted. An example of this will be found below. DTFR also prepares three-dimensional isometric projections of the

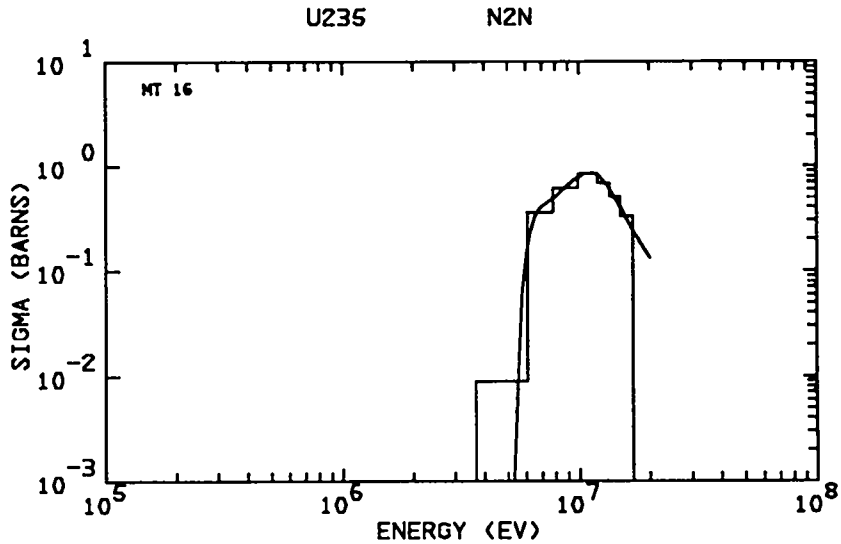


Figure 1: An example of DTFR plotting for the (n,2n) reaction of ENDF/B-VI U-235. This plot was prepared using IFILM=1 and in-table edits. All the threshold reactions are shown using the same energy range to make comparisons easy.

P_0 scattering matrix and the P_0 photon-production table. The user can request that one reaction be plotted per page, or that four reactions be drawn on the same page. Examples of these plots are given in Figures 1–5.

D. User Input

The following user input specifications were copied from the comment cards at the beginning of the DTFR module source code.

```
*---INPUT SPECIFICATIONS (FREE FORMAT)-------------------------*
*                                                             *
* CARD 1       UNITS                                          *
*    NIN       INPUT UNIT WITH DATA FROM GROUPR (BINARY).     *
*    NOUT      OUTPUT UNIT CONTAINING DTF TABLES (CODED).     *
*              (DEFAULT=0=NONE)                               *
*    NPEND     INPUT UNIT WITH PENDF TAPE FOR POINT PLOTS.    *
*              (DEFAULT=0=NONE)                               *
* CARD 2       OPTIONS                                        *
*    IPRINT    PRINT CONTROL (0 MINIMUM, 1 MAXIMUM)           *
*    IFILM     FILM CONTROL (0/1/2=NO/YES WITH 1 PLOT PER FRAME/ *
*              YES WITH 4 PLOTS PER FRAME (DEFAULT=0)         *
*    IEDIT     EDIT CONTROL (0/1=IN TABLE/SEPARATE) (DEFAULT=0) *
*                                                             *
```

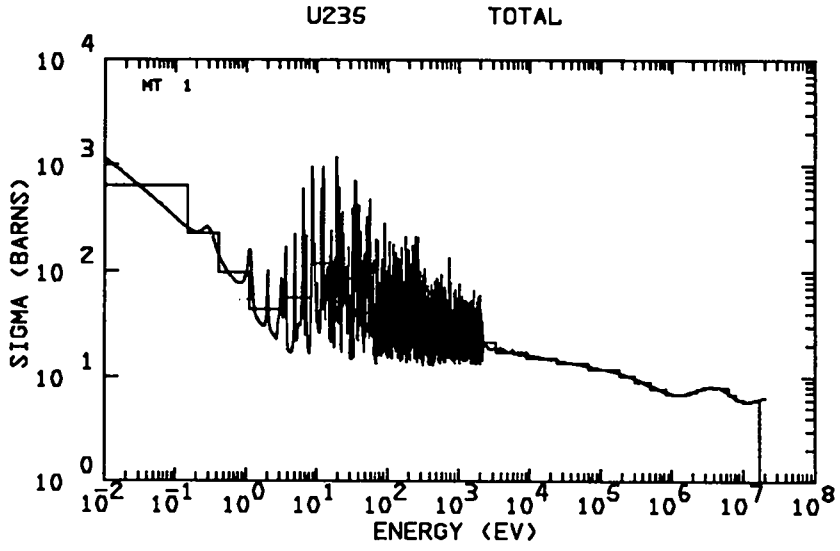


Figure 2: This is the total cross section for ENDF/B-VI U-235 from the same run as the previous plot. The energy range from 1×10^{-5} eV to 1×10^{-2} eV was removed to expand the rest of the energy scale.

```
*       CARDS 3 THROUGH 5 ONLY FOR IEDIT=0                          *
*                                                                   *
* CARD 3        NEUTRON TABLES                                      *
*    NLMAX      NUMBER OF NEUTRON TABLES DESIRED.                   *
*    NG         NUMBER OF NEUTRON GROUPS                            *
*    IPTOTL     POSITION OF TOTAL CROSS SECTION                     *
*    IPINGP     POSITION OF IN-GROUP SCATTERING CROSS SECTION.      *
*    ITABL      NEUTRON TABLE LENGTH DESIRED.                       *
*    NED        NUMBER OF ENTRIES IN EDIT TABLE (DEFAULT=0).        *
*    NTHERM     NUMBER OF THERMAL GROUPS (DEFAULT=0).               *
*  CARD 3A ONLY FOR NTHERM NE 0                                     *
* CARD 3A       THERMAL INCOHERENT AND COHERENT MTS                 *
*    MTI        MT FOR THERMAL INCOHERENT DATA                      *
*    MTC        MT FOR THERMAL COHERENT DATA (DEFAULT=0)            *
*    NLC        NO. COHERENT LEGENDRE ORDERS (DEFAULT=0)            *
* CARD 4        EDIT NAMES                                          *
*       SIX CHARACTER HOLLERITH NAMES FOR EDITS FOR AS MANY         *
*       CARDS AS NEEDED.  THERE WILL BE IPTOTL-3 NAMES READ.        *
*       EACH NAME IS DELIMITED WITH *.                              *
* CARD 5        EDIT SPECIFICATIONS                                 *
*       NED TRIPLETS OF NUMBERS ON AS MANY CARDS AS NEEDED.         *
*       POSITIONS CAN APPEAR MORE THAN ONCE.                        *
*       REACTION TYPES CAN APPEAR MORE THAN ONCE.                   *
*    JPOS       POSITION OF EDIT QUANTITY.                          *
*    MT         ENDF/B REACTION NUMBER.                             *
*    MULT       MULTIPLICITY TO BE USED WHEN ADDING THIS MT.        *
```

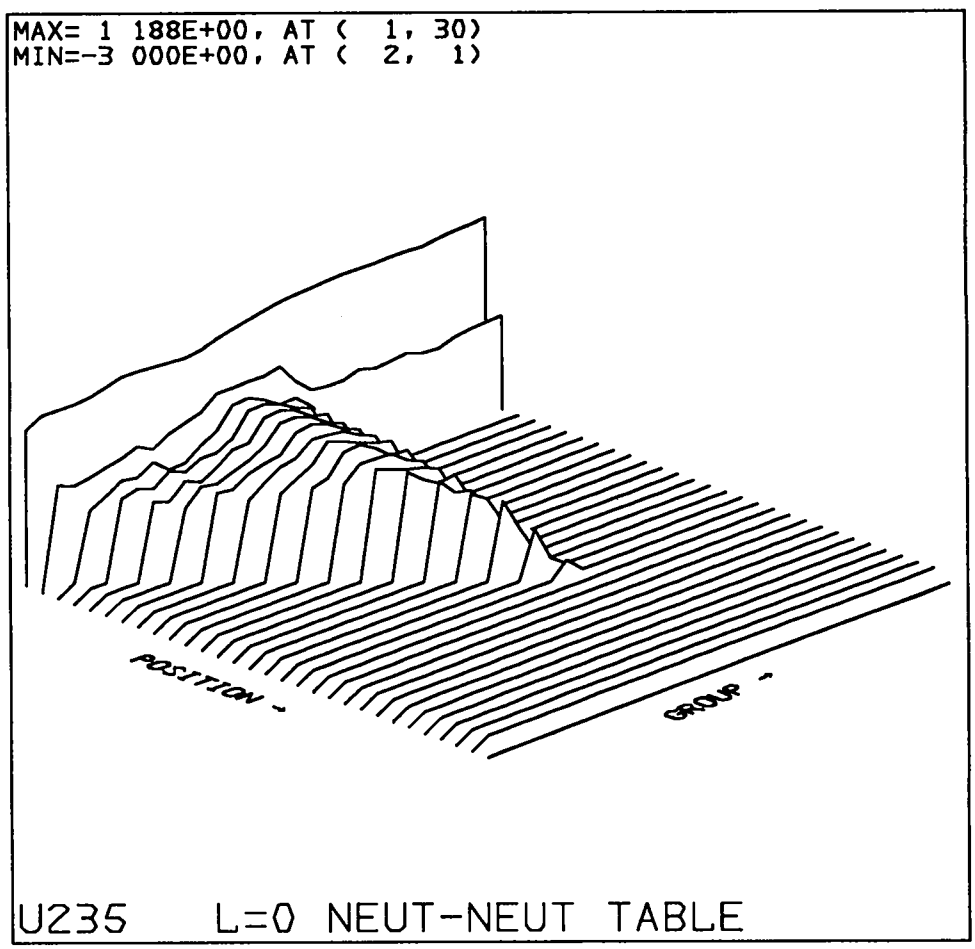


Figure 3: This plot shows the P_0 scattering matrix for ENDF/B-VI U-235 by position and group. The curve to the left rear is the ingroup scattering versus group; the curve in front of it is the source into group g from one group above; and so on. The group scale is plotted versus group index, not particle energy.

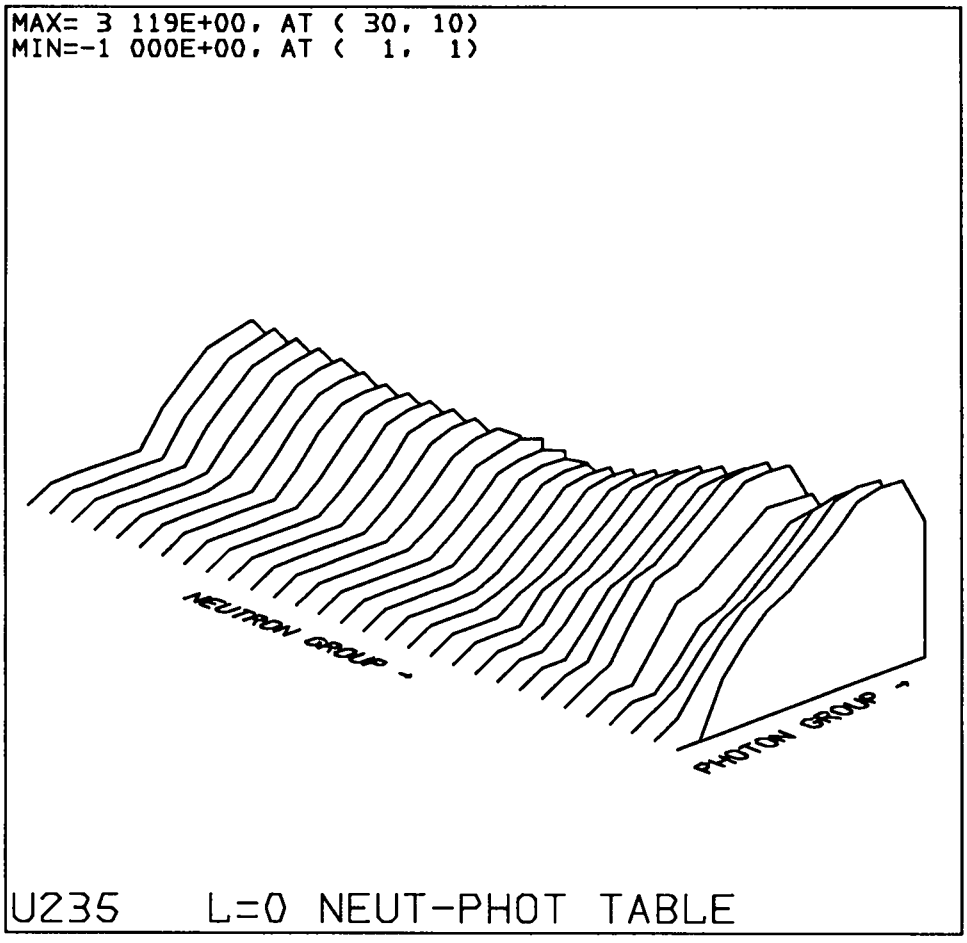


Figure 4: The 30×12 P_0 photon production matrix for U-235 is plotted versus neutron group index and photon group index, not versus neutron and photon energy.

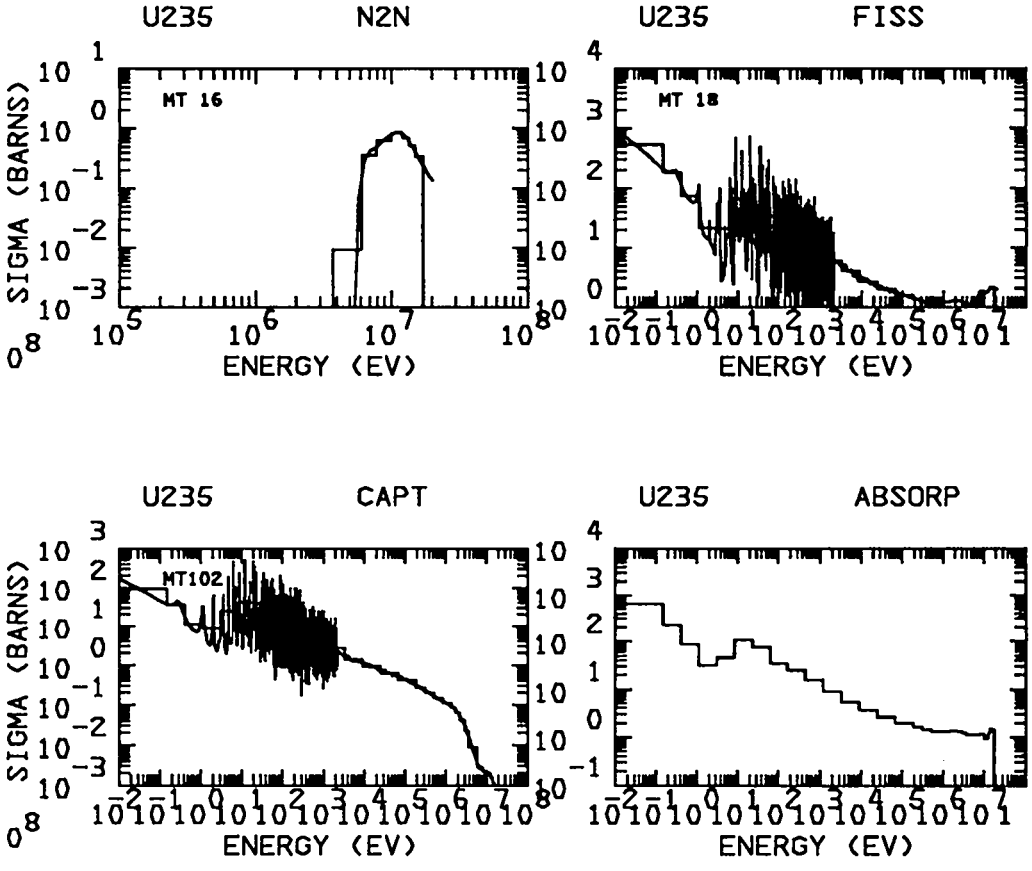


Figure 5: An example of DTFR plotting using IFILM=2 for ENDF/B-VI U-235 with 3 special "in-table" edits. (Note that this option needs a little work!)

```
*                                                                     *
*       CARD 6 FOR IEDIT=1                                            *
*                                                                     *
* CARD 6        CLAW-FORMAT TABLES                                    *
*    NLMAX      NUMBER OF NEUTRON TABLES (DEF=5)                      *
*    NG         NUMBER OF NEUTRON GROUPS (DEF=30)                     *
*               (NUMBER OF THERMAL GROUPS IS ZERO)                    *
*                                                                     *
* CARD 7        GAMMA RAY TABLES                                      *
*    NPTABL     NUMBER OF GAMMA TABLES DESIRED (DEFAULT=0)            *
*    NGP        NUMBER OF GAMMA GROUPS (DEFAULT=0)                    *
* CARD 8        MATERIAL DESCRIPTION                                  *
*       ONE CARD FOR EACH TABLE SET DESIRED.                          *
*       MAT=0/ TERMINATES EXECUTION OF DTFR.                          *
*    HISNAM     HOLLERITH ISOTOPE NAME                                *
*    MAT        MATERIAL NUMBER AS IN ENDF/B (DEFAULT=0)              *
*    JSIGZ      INDEX NUMBER OF SIGMA-ZERO DESIRED (DEFAULT=1)        *
*    DTEMP      TEMPERATURE DESIRED (DEFAULT=300)                     *
*                                                                     *
***********************************************************************
```

As usual, card 1 is used to assign the input and output units for the module. NIN must be an output tape from GROUPR, and it can be in either binary or BCD (coded) mode. The output file NOUT must be in coded mode. It will contain the DTF-format card images. NPEND should be the same PENDF tape that was used in GROUPR when NIN was made. It is only needed if plots are requested. Card 2 starts out with the print flag IPRINT, which is usually set to 1. The parameter IFILM can be used to suppress plotting, or to request plots with either 1 or 4 graphs per frame. Examples of the DTFR plotting capability were given above. IPRINT is used to control the output format. If it is equal to zero, a conventional DTF-type table is produced. If any edits were requested, they are given in the table using the first few positions in the table. The separate-edits option is used to produce cross sections in the "CLAW" format. In this format, the edits are extracted from the table and written out separately with identification information appended to the right side of each card image. The scattering tables have only the standard 3 edits, and they also have standard identification fields added to the right side of each card. Examples of both styles were given above (see Tables 2 and 4).

Cards 3 through 5 are used for IEDIT=0 only. The first parameter is ILMAX. It is the number of Legendre tables desired; that is, it would be 4 for a P_3 set. The number of groups NG must agree with the number on the input tape from GROUPR. The value of IPTOTL is used to determine both the position of the total cross section in the table and the number of special edit positions at the front of the table (IPTOTL-3, which can be zero). Note that NED is not the number of edit positions; it is the number of edit specification triplets to be read. Therefore, NED$\geq$IPTOTL-3.

Also IPINGP=IPTOTL+1 if there is no upscatter, and IPINGP=IPTOTL+NUP+1 when the total number of upscatter groups is NUP. (The parameter NUP is not actually given in the input; it is always equal to IPINGP-IPTOTL-1.) The length of a full table is given by

```
      IPTOTL-3
+     3
+     NUP
+     NG
-------------------
      ITABLE          .
```

However, smaller table lengths can be requested; truncation will be performed in a way that preserves the production cross sections. The parameter NTHERM can be set to zero if no thermal upscatter cross sections are desired. If nonzero, it refers to the number of incident thermal groups, and it is used to define the breakpoint between the thermal and epithermal treatments. It is not necessarily equal to NUP.

Card 3a is only given if NTHERM>0. It gives the MT numbers for the thermal incoherent and coherent cross sections to be extracted from the GENDF tape. Examples might include 221 and zero for thermal scattering, or 229 and 230 for graphite. NLC can be used to truncate the anisotropy of the coherent term, if desired. For thermal cases, DTFR subtracts the static elastic scattering from both the total and absorption positions for the lowest NTHERM groups, and then it adds in the cross sections corresponding to MTI and MTC for these thermal groups. When reading through the matrices, it omits all the contributions from the static elastic matrix for the thermal groups, and adds in the MTI and MTC contributions for these groups instead.

Card 4 gives Hollerith names for the edit cross sections. These NED names will appear on the output listing, but they are not passed on to the output file. Card 5 specifies the contents of each of the special edit fields. Note that each edit can be any linear combination of the cross sections on the input GENDF tape. This feature can be used to produce complex edits like gas production. An example follows:

```
1.   ...
2.   *N.HE4* *KERMA* *FISS*
3.   1 107/
4.   1 91 3/
5.   2 301/
6.   3 444/
7.   ....
```

The line numbers are not part of the input. Line 1 represents all the input cards before card 4, and line 7 represents all the cards after card 5. This input is for

ENDF/B-IV ^{12}C. Lines 3 and 4 construct a helium-production cross section as the sum of (n,α) and three times (n,n')3α. Lines 5 and 6 assign two more edit positions for heating and damage, respectively. The MT numbers used for the **MT** field on card 5 are usually just the ENDF MT numbers for the reaction. However, the following special values are used to request the weighting flux, the steady-state and delayed components of the fission neutron spectrum, or the delayed fission neutron yield:

Special MT	Meaning
300	weighting flux (from MF3,MT1)
455	delayed neutron yield (MF3,MT455)
470	steady-state spectrum (see Eq. 4)
471	delayed neutron spectrum (MF5,MT455)

Remember that the steady-state fission neutron production cross section will be found in position `IPTOTL-1` of the transport table.

When a multigroup edit is a combination of several cross sections as in this example, the plot of the edit includes the pointwise cross sections for all of the component reactions. Figure 6 illustrates this for the ^{12}C helium-production reaction. Another useful trick is to build up a compound edit out of two reactions using a multiplicity of zero for the second reaction. This causes the second reaction to be plotted but not included in the edit cross section. This trick can be used to compare the energy-balance (MT301) and kinematic (MT443) versions of the KERMA factor. The appropriate input lines are

```
...
*HEAT*/
1 301 1 1 443 0/
...
```

Note that this method is used in the predefined edits associated with `IEDIT=1` (see Table 3).

Card 6 is used instead if `IEDIT=1`. As described above, the list of special edit cross sections is fixed for the CLAW format. Therefore, it is only necessary to give **NLMAX** and **NG**. The number of thermal groups is automatically set to zero.

The next card read for either choice of output format is card 6, which controls the generation of a photon production matrix. The number of photon production tables is usually zero (none), or one. Only a few materials in the ENDF/B libraries have anisotropic photon production data. Of course, **NGP** must agree with the photon group structure used in GROUPR.

The input deck ends with a material description card for each material to be processed. These materials must all be on the input GENDF tape. (Multi-MAT

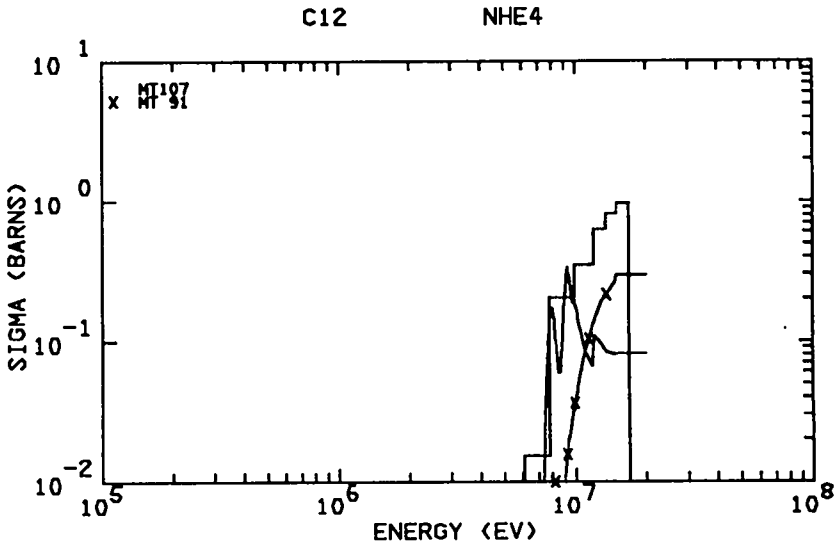


Figure 6: An example of a DTFR edit plot showing multiple pointwise curves that are the components of a compound edit. This graph is for ENDF/B-IV C-12 helium production. The histogram gives the sum of the (n,α) cross section and three times the (n,n')3α cross section.

GENDF tapes can be prepared using MODER from single-material GROUPR output files, but since it is easy to combine materials at the DTF-format stage, DTFR is usually run for one MAT at a time.) The Hollerith material names given will appear in the comment cards before tables and in the special labels added to the right-hand edge of each card. The material numbers **MAT** are the same ENDF MAT numbers used when preparing the multigroup cross sections. DTFR has only a limited capability of handling self-shielded cross sections. The user can specify that a given set of tables be prepared using one particular temperature **DTEMP** and one particular background cross section, the JSIGZth value in the set.

E. Coding Details

The code starts in subroutine DTFR by calling RUIN to read the user's input. Note that the special set of edits used when IEDIT=1 is specified using data statements in RUIN (see KMTED, KJPED, KMULTD, and KMTID). The next steps are to open a scratch file NSCR and to initialize the plotting system (see below for more details on the machine-dependent features of the plotting system).

The main loop over materials, dilution values, and temperatures goes through statement number 105. This is where input card number 8 is read (see input instructions) to identify the material to be processed. The input GENDF tape is then rewound, and NIN is searched for the material and temperature. If a PENDF input tape has been mounted, the corresponding material and temperature are located on NPEND. Note that the materials in the input file do not have to be in the same order as the materials on either NIN or NPEND. If a requested material or temperature is not found, a fatal error message is issued.

When the input files have been properly positioned, DTFR checks to see if there is enough storage for the tables. The limit NWSMAX=20000 is large enough for sets with around 100 groups. If larger sets are desired, modify the common arrays in "COMMON/DSTORE/A(20000),SIG(20000)" and the corresponding limits in "DATA NWAMAX/20000/, NWSMAX/20000/." The loop over reactions and groups in a reaction goes through statement number 150. Once the cross sections have been read off the input tape, the code branches to different sections to process different kinds of data. The first of these processes the File 3 neutron cross sections. Note that the total cross section is stored in both IPTOTL and IPTOTL-2. In addition, the GROUPR weighting flux can be extracted from the section MF3,MT1 and stored into a special edit requested using MT=300. Special edit cross sections are weighted with the specified multiplicities and summed into the specified edit positions in the "DO 250 IED=1,NED" loop. Thermal corrections to static scattering are made at statement number 260. The method used is to subtract MT2 and add MTI and MTC to both the total and absorption positions.

The block of coding starting with statement number 300 is used to accumulate the total scattering matrix. Most numbers are simply added into the correct position in the transport table. The operation specified by Eq.(1) is performed by subtracting all matrix elements from the absorption position. In addition, in the thermal range, the MTI and MTC data are used instead of the MT2 data.

The prompt part of fission is handled starting at statement number 340. Note that a check is made to see if MT18 was already processed if MT19 is found. This results in a fatal error, so the user must be careful to process only one of these representations in GROUPR. The special cases IG=0 and IG2LO=0 flag the presence of the low-energy constant spectrum or production cross section, respectively. Delayed fission contributions are handled by the next block of coding. The method used for combining the fission matrix, the constant-spectrum part, and the delayed parts are defined by Eqs.(3) and (4). The normalization parameter needed to fix up χ is accumulated in the variable CNORM.

The photon-production matrix is accumulated starting at statement number 600. It is a very simple process of adding all the partial matrices with MF=16 on the input GENDF file. When all the reactions and groups for this material and temperature have been processed, DTFR calls the subroutine DTFOUT to prepare the tables and plots, and then it loops back to statement number 105 for the next requested material. After all the materials, temperatures, and background cross sections have been written out, the plotting system is terminated (see below), files are closed, a final message is printed, and control is returned to NJOY.

Subroutine DTFOUT controls the preparation of the output DTF tables on NOUT, the printing of the tables on the user's output device, and the preparation of plots. This is a simple routine with separate paths for in-table edits and separate edits. Plotting is handled using different subroutines for the edits, the neutron table, and the photon-production table.

Subroutine PLOTED is used to prepare plots of all the special and standard edit positions, including overlay plots of PENDF cross sections if NPEND is available. Subroutine HISTOD is used to convert the multigroup values into a pointwise cross section with steps at each of the group boundary energies; that is, into histogram form. Subroutine DPEND prepares the PENDF part of the plot by thinning the original PENDF grid down to a new grid with x spacing something like the resolution of a typical screen. In addition, it "thickens" reactions, if necessary, so that there are enough points to properly follow the curve on the final log-log plot. Both the histogram and PENDF data arrays are drawn onto the graph using PLOPB.

Subroutine PLOTNN is used to plot an isometric view of the neutron table. This is done by removing the edits from the table and passing the scattering part to ISOPLT. A similar process is performed in PLOTNP for the photon-production matrix. Note that the table is transposed to get the desired viewpoint, and very small numbers are removed so that the important features are more visible.

Subroutine PLOPB is based on a routine with the same name from the old Los Alamos film plotter system, and it uses several private routines to draw and label its axes. These routines will not be discussed here. ISOPLT was also based on a routine from the old Los Alamos library. Both of these subroutines need to draw characters for titles and axis labels; this is the role of DLCH. This is a heavily modified version of a character generator from the old Los Alamos library. These modifications include an ASCII interface and skewed letters to be used on isometric plots. The characters are drawn on a fairly coarse 5-by-6 grid, but several fonts are available, including Greek and special characters. The advantage of these fairly

old-fashioned plotting subroutines is that they need only simple vector drawing calls. Therefore, they are easy to transport to different computer systems.

The basic set of plotting primitives in DTFR will now be described. Several variants are available through the *IF capability of UPD.[4] The first step is to initialize the plotting package. This is done by GINIT for the DISSPLA*, CGS (the Los Alamos Common Graphics System), or PostScript†. The initialization procedure for DISSPLA varies from installation to installation, and this first three cards of the "*IF DISS" block will probably have to be changed using a machine-dependent update in UPD. The last step in DTFR is to finalize plotting using GFINL. This step also may change slightly from system to system. PostScript output appears on the file called TAPE77, which is suitable to be sent to any PostScript-compliant laser printer.

Subroutine ADV is used to "advance" to a new page or frame. Once again, options are provided for three different graphics systems. The last of the plotting primitives is DRV, which draws a line from position IX1,IY1 to position IX2,IY2. These positions are given in the coordinate system used on the old SC4020 film plotter; they increase down and to the right and use values from 0 to 1023 on each axis. These coordinates are converted to the native coordinates for the three different graphics options supported, and then the line is drawn. The variables IXLAST and IYLAST are used to save the endpoint of the last line drawn. If IX1 and IY1 do not match the previous endpoint, a move operation is performed before the draw operation.

F. Error Messages

ERROR IN DTFR***NUMBER OF NEUTRON GROUPS DISAGREES WITH...

The value of NG in the input must be consistent with the number of groups on the input GENDF tape. Check the input, and check whether the correct tape was mounted.

ERROR IN DTFR***NUMBER OF GAMMA GROUPS DISAGREES WITH...

Same as above, except check the number of gamma groups NGP.

ERROR IN DTFR***DESIRED TEMPERATURE NOT ON PENDF

Code is unable to find MAT and DTEMP on the input PENDF tape NPEND. Check the input information and check whether the correct PENDF tape was mounted.

ERROR IN DTFR***NOT ENOUGH STORAGE FOR TABLE

*The DISSPLA graphics system is a product of ISSCO Integrated Software Systems Corporation, 10505 Sorrento Valley Road, San Diego, CA 92121.

†PostScript is a trademark of Adobe System Incorporated.

There is not enough space in the SIG array to construct a table with so many positions and groups. See SIG(20000) in common block /DSTORE/ and the data statement containing NWSMAX/20000/.

ERROR IN DTFR***NOT ENOUGH STORAGE FOR RECORD

There is not enough space in the A array to read in the data on the GENDF tape. See A(20000) in common block /DSTORE/ and the data statement containing NWAMAX/20000/.

ERROR IN DTFR***MT18 ALREADY PROCESSED//MT19 NOT ALLOWED

Make sure that GROUPR processes the fission matrix using either MT=18 only, or MT=19, 20, 21, and 38, but not both.

ERROR IN DTFR***DELAYED NUBAR REQUIRED TO ADD DELAYED CHI...

The delayed neutron yield must have been processed in GROUPR (MFD=3, MTD=455), or DTFR will be unable to construct the steady-state fission vectors.

ERROR IN RUIN***IPING.LE.IPTOTL

The ingroup position IPING is normally IPTOTL+1, and it will be larger still if thermal upscatter is to be included. Check input card number 3.

ERROR IN RUIN***NOT ENOUGH STORAGE FOR EDITS

The dimensioned variables in common block /DTF2/ limit the number of edits allowed to 50. See also NEDMAX=50 in DTFR and the dimension statements for Z and IZ in RUIN.

ERROR IN DPEND***NPTS EXCEEDS NDIM

The thinning/thickening process has produced too many points for the arrays X and Y. These arrays are dimensioned in PLOTED, and the limit NDIM=3500 is also set there.

ERROR IN ISOPLT***INPUT ARGUMENT ERROR

There must be more than 3 and fewer than 200 points in the arrays to be plotted using ISOPLT.

G. References

1. K. D. Lathrop,, "DTF-IV, A FORTRAN Program for Solving the Multigroup Transport Equation with Anisotropic Scattering," Los Alamos Scientific Laboratory report LA-3373 (November 1965).

2. R. J. Barrett and R. E. MacFarlane, "Coupled Neutron and Photon Cross Sections for Transport Calculations," Los Alamos Scientific Laboratory report LA-7808-MS (April 1979).

3. R. E. MacFarlane, "TRANSX-CTR: A Code for Interfacing MATXS Cross-Section Libraries to Nuclear Transport Codes for Fusion Systems Analysis," Los Alamos National Laboratory report LA-9863-MS (February 1984).

4. R. E. MacFarlane and D. C. George, "UPD: A Portable Version-Control Program," Los Alamos National Laboratory report LA-12057-MS (April 1991).

XIV. CCCCR

The CCCC interface files (commonly pronounced "four cees") were developed by the Committee for Computer Code Coordination for the US Fast Breeder Reactor Program. When the members of this committee started work in 1970, they noted that because of the large variety of computers and computer systems, computer codes developed at one laboratory were often incompatible with computers at other laboratories. Major rewrites of codes or wasteful duplicate efforts were common. They hoped to create a system that would allow different laboratories to create codes that could be moved to other sites more easily. Moreover, they hoped that the codes developed at different laboratories could easily work together, thereby achieving larger and more capable calculational systems than any one laboratory could hope to develop by itself.

They approached this problem in two ways. First, they tried to establish general programming standards that would make computer codes more portable. And second, they tried to establish standard interface files for reactor physics codes that would make it easier for computer codes to communicate with each other. The results of this work appeared in fullest form as the CCCC-III and CCCC-IV standards.[1, 2]

The MINX code,[3] which was the predecessor of NJOY, was able to produce libraries that used the CCCC-III interface formats.[4] The LINX and BINX library management codes[5] and the CINX group collapse code[6] were also released during this period. Major codes that used data in CCCC-format included SPHINX[7] from Westinghouse, TDOWN[8] from General Electric, DIF3D[9] from Argonne National Laboratory, and ONEDANT[10] from Los Alamos. It was indeed found that codes could be moved more easily than before. Analysts could use ONEDANT and DIF3D on similar problems; they could even use one code to generate utility files (for example, mixture and geometry files) that would work with the other code! When NJOY was developed, it first produced version III formats and was later upgraded to the CCCC-IV standard.

With the demise of the breeder reactor program, development of the CCCC system has stopped. However, many good programs are still available that make use of CCCC files and programming standards. The Los Alamos DANDE system[11] is an example of how the use of standard interface files can be used to couple several reactor physics programs together into an easy-to-use and powerful product. In areas where the CCCC standards were not very successful, such as gamma ray cross sections and cross sections for the fusion energy range, the MATXS format is

available as an alternative. This generalized material cross section library format uses CCCC-type techniques. Thus, the CCCC spirit is not dead.

A. CCCC Procedures and Programming Standards

Although the CCCC programming standards went so far as to give advice on program structure, documentation, and good coding practice, their main purpose was to make it easier to move computer codes from one machine to another. The main problems in those days were the slightly different implementations of input/output on CDC and IBM machines, the different word size on CDC and IBM machines, and the relatively small size of the main memory on the CDC 7600.

The last of these problems was attacked by limiting the maximum memory requirements for CCCC-compliant codes. This problem has essentially disappeared for the current generation of large computers.

The word-size problem has three components. First, it is often necessary to change the statements that allocate space for variables and arrays [for example, "`DIMENSION A(10)`" might have to be changed to "`REAL*8 A(10)`" when moving from a long-word machine (CDC, Cray) to a short-word machine (IBM, VAX, Sun)]. Second, the names of functions that work with double-precision variables normally must be changed (for example, `ALOG10` to `DLOG10`). And third, the word boundaries of double-precision variables must be properly aligned in common blocks and equivalenced arrays. The standard CCCC method for handling name changes has been based on using standard control cards. As an example, a code for a long-word machine might contain the following code fragment:

```
CSW
C     REAL*8 HA(10)
CSW
CLW
      INTEGER HA(10)
CLW
```

The variable `HA` is intended to hold 10 words of Hollerith information using the standard CCCC 6-character word length. Such a variable must be declared as double precision on short-word machines, which typically allow four Hollerith characters per word. To change this code to its short-word version, a special utility code reads through each line removing the comment "C" from lines bracketed by the "SW" comments and inserting a "C" in column 1 for all lines bracketed by "LW" comments. Previous versions of NJOY used this scheme; this version uses UPD[12] conditional statements instead. For example, the source file contains

```
*IF SW
      REAL*8 HA(10)
*ELSE
      INTEGER HA(10)
*ENDIF    ,
```

and the compile file produced by UPD has only one of the two alternatives activated, depending on whether SW has been set or not.

The word-alignment problem requires that some care be used in allocating arrays and common blocks. For example,

```
REAL*8 HA
COMMON/BAD/IA(3),HA(10)
```

should be avoided; it would be OK with IA(4). Most CCCC records contain mixtures of Hollerith, floating-point, and integer variables. The desired data is normally extracted by making use of equivalenced arrays. For example, a code could contain the following declarations:

```
REAL*8 HA(10)
DIMENSION A(20),IA(20)
EQUIVALENCE (HA(1),A(1)),(IA(1),A(1))    .
```

Assume that a record containing 2 Hollerith variables (which require two single-precision variables each), 2 floating-point numbers (at single precision), and 2 integers has been read into array A. How do you extract the first of the integers? The solution depends on defining a CCCC-standard quantity called MULT, which is 1 for long-word machines and 2 for short-word machines. Now, the desired value can be obtained with an expression of the form

```
I1=IA(2*MULT+3)    .
```

The second Hollerith variable would be extracted using the simple expression

```
H2=HA(2)    .
```

Changing the value of MULT when transporting a code to a different machine is easily handled using control-card brackets or UPD conditionals as described above.

The remaining feature of the CCCC programming standards that is used in NJOY is the concept of standardized input/output subroutines. The CCCC interface files are sequential binary files (binary for efficiency and sequential for simplicity). The interface formats are arranged so that the length of any record

can be calculated using parameters already read from previous records. It is convenient to insulate CCCC input/output from system variations by defining two standard routines:

`REED(NREF,IREC,ARRAY,NWDS,MODE)`

Read record `IREC` from unit `NREF` into `ARRAY`. The record has length `NWDS` in single-precision words. The `MODE` parameter is used to control I/O buffering, and it is not used in NJOY. Records can be read out of sequence; the routine does any record skipping (forward or backward) needed to arrive at record number `IREC`.

`RITE(NREF,IREC,ARRAY,NWDS,MODE)`

Write record `IREC` onto unit `NREF` using the data in `ARRAY`. The first `NWDS` single-precision words will be written. The `MODE` parameter is ignored. The records `NREF` must be written in sequence. The unit will be rewound if `IREC=1`.

When transporting a code between different computer systems, it is only necessary to have (or prepare) operational versions of `REED` and `RITE` for the target machine.

B. The Standard Interface Files

The CCCCR module produces data libraries that use three of the CCCC-IV standard interface files, namely:

ISOTXS for nuclide (isotope)-ordered multigroup neutron cross sections including cross section versus energy functions for the principal cross sections, group-to-group scattering matrices, and fission neutron production and spectra tables;

BRKOXS for Bondarenko-type self-shielding factors versus energy group, temperature, and background cross section for the reactions with major resonance contributions; and

DLAYXS for delayed-neutron precursor yields, emission spectra, and decay constants for the major fissionable isotopes.

The format of each of these files, the definition of the types of data included, and the uses and weaknesses of these three standard file formats are discussed in the following three sections.

As mentioned in the preceding section, the normal form of the CCCC files is binary and sequential. CCCCR writes its output in this binary mode. Of course, coded versions (BCD, ASCII, EBCDIC, etc.) are needed to move library files between different machines, and the formats used for the coded versions are given in the file descriptions below. A separate program, BINX,[5] is used to convert back and forth between coded and binary modes. BINX can also be used to prepare

an interpreted listing of a library. CCCCR can prepare an entire multimaterial library in one run if a multimaterial GENDF file is available. It can also be used to prepare an interface file containing only one material. These one-material files can be merged into multimaterial libraries using the LINX code.[5]

C. ISOTXS

The format for the ISOTXS material (isotope)-ordered cross section file is given below. This computer-text format is standard for the CCCC interface files.

```
C*************************************************************************
C                        REVISED 11/30/76                                -
C                                                                        -
CF          ISOTXS-IV                                                    -
CE          MICROSCOPIC GROUP NEUTRON CROSS SECTIONS                     -
C                                                                        -
CN                       THIS FILE PROVIDES A BASIC BROAD GROUP          -
CN                       LIBRARY, ORDERED BY ISOTOPE                     -
CN                       FORMATS GIVEN ARE FOR FILE EXCHANGE PURPOSES    -
CN                       ONLY.                                           -
C                                                                        -
C*************************************************************************

C------------------------------------------------------------------------
CS          FILE STRUCTURE                                               -
CS                                                                       -
CS             RECORD TYPE                          PRESENT IF           -
CS             ==============================       ===============      -
CS             FILE IDENTIFICATION                  ALWAYS               -
CS             FILE CONTROL                         ALWAYS               -
CS             FILE DATA                            ALWAYS               -
CS             FILE-WIDE CHI DATA                   ICHIST.GT.1          -
CS   **************(REPEAT FOR ALL ISOTOPES)                             -
CS   *         ISOTOPE CONTROL AND GROUP                                 -
CS   *                         INDEPENDENT DATA    ALWAYS               -
CS   *         PRINCIPAL CROSS SECTIONS             ALWAYS               -
CS   *         ISOTOPE CHI DATA                     ICHI.GT.1            -
CS   *  ***********(REPEAT TO NSCMAX SCATTERING BLOCKS)                   -
CS   *  *  ********(REPEAT FROM 1 TO NSBLOK)                             -
CS   *  *  *   SCATTERING SUB-BLOCK                LORD(N).GT.0         -
CS   *************                                                        -
C                                                                        -
C------------------------------------------------------------------------

C------------------------------------------------------------------------
CR          FILE IDENTIFICATION                                          -
C                                                                        -
CL    HNAME,(HUSE(I),I=1,2),IVERS                                        -
C                                                                        -
CW    1+3*MULT=NUMBER OF WORDS                                           -
C                                                                        -
```

```
CB    FORMAT(11H OV ISOTXS  ,1H*,2A6,1H*,I6)                              -
C                                                                         -
CD    HNAME       HOLLERITH FILE NAME - ISOTXS -                          -
CD    HUSE(I)     HOLLERITH USER IDENTIFICATION (A6)                      -
CD    IVERS       FILE VERSION NUMBER                                     -
CD    MULT        DOUBLE PRECISION PARAMETER                              -
CD                    1- A6 WORD IS SINGLE WORD                           -
CD                    2- A6 WORD IS DOUBLE PRECISION WORD                 -
C                                                                         -
C-------------------------------------------------------------------------

C-------------------------------------------------------------------------
CR          FILE CONTROL   (1D RECORD)                                    -
C                                                                         -
CL    NGROUP,NISO,MAXUP,MAXDN,MAXORD,ICHIST,NSCMAX,NSBLOK                 -
C                                                                         -
CW    8=NUMBER OF WORDS                                                   -
C                                                                         -
CB    FORMAT(4H 1D ,8I6)                                                  -
C                                                                         -
CD    NGROUP        NUMBER OF ENERGY GROUPS IN FILE                       -
CD    NISO          NUMBER OF ISOTOPES IN FILE                            -
CD    MAXUP         MAXIMUM NUMBER OF UPSCATTER GROUPS                    -
CD    MAXDN         MAXIMUM NUMBER OF DOWNSCATTER GROUPS                  -
CD    MAXORD        MAXIMUM SCATTERING ORDER (MAXIMUM VALUE OF            -
CD                     LEGENDRE EXPANSION INDEX USED IN FILE).            -
CD    ICHIST        FILE-WIDE FISSION SPECTRUM FLAG                       -
CD                     ICHIST.EQ.0,      NO FILE-WIDE SPECTRUM           -
CD                     ICHIST.EQ.1,      FILE-WIDE CHI VECTOR            -
CD                     ICHIST.GT.1,      FILE-WIDE CHI MATRIX            -
CD    NSCMAX        MAXIMUM NUMBER OF BLOCKS OF SCATTERING DATA           -
CD    NSBLOK        SUBBLOCKING CONTROL FOR SCATTER MATRICES. THE         -
CD                     SCATTERING DATA ARE SUBBLOCKED INTO NSBLOK         -
CD                     RECORDS (SUBBLOCKS) PER SCATTERING BLOCK.          -
C                                                                         -
C-------------------------------------------------------------------------

C-------------------------------------------------------------------------
CR          FILE DATA   (2D RECORD)                                       -
C                                                                         -
CL    (HSETID(I),I=1,12),(HISONM(I),I=1,NISO),                            -
CL   1(CHI(J),J=1,NGROUP),(VEL(J),J=1,NGROUP),                            -
CL   2(EMAX(J),J=1,NGROUP),EMIN,(LOCA(I),I=1,NISO)                        -
C                                                                         -
CW    (NISO+12)*MULT+1+NISO                                               -
CW    +NGROUP*(2+ICHIST*(2/ICHIST+1)))=NUMBER OF WORDS                    -
C                                                                         -
CB    FORMAT(4H 2D ,1H*,11A6,1H*/     HSETID,HISONM                       -
CB   11H*,A6,1H*,9(1X,A6)/(10(1X,A6)))                                    -
CB    FORMAT(6E12.5)                  CHI (PRESENT IF ICHIST.EQ.1)        -
CD    FORMAT(6E12.5)                  VEL,EMAX,EMIN                       -
CD    FORMAT(12I6)                    LOCA                                -
C                                                                         -
```

```
CD    HSETID(I)     HOLLERITH IDENTIFICATION OF FILE (A6)                -
CD    HISONM(I)     HOLLERITH ISOTOPE LABEL FOR ISOTOPE I (A6)           -
CD    CHI(J)        FILE-WIDE FISSION SPECTRUM(PRESENT IF ICHIST.EQ.1)   -
CD    VEL(J)        MEAN NEUTRON VELOCITY IN GROUP J (CM/SEC)            -
CD    EMAX(J)       MAXIMUM ENERGY BOUND OF GROUP J (EV)                 -
CD    EMIN          MINIMUM ENERGY BOUND OF SET (EV)                     -
CD    LOCA(I)       NUMBER OF RECORDS TO BE SKIPPED TO READ DATA FOR     -
CD                     ISOTOPE I.  LOCA(1)=0                             -
C                                                                        -
C-------------------------------------------------------------------------

C-------------------------------------------------------------------------
CR          FILE-WIDE CHI DATA   (3D RECORD)                             -
C                                                                        -
CC        PRESENT IF ICHIST.GT.1                                         -
C                                                                        -
CL    ((CHI(K,J),K=1,ICHIST),J=1,NGROUP),(ISSPEC(I),I=1,NGROUP)          -
C                                                                        -
CW    NGROUP*(ICHIST+1)=NUMBER OF WORDS                                  -
C                                                                        -
CB    FORMAT(4H 3D ,5E12.5/(6E12.5))   CHI                               -
CB    FORMAT(12I6)                     ISSPEC                            -
C                                                                        -
CD    CHI(K,J)      FRACTION OF NEUTRONS EMITTED INTO GROUP J AS A       -
CD                     RESULT OF FISSION IN ANY GROUP,USING SPECTRUM K   -
CD    ISSPEC(I)     ISSPEC(I)=K IMPLIES THAT SPECTRUM K IS USED          -
CD                     TO CALCULATE EMISSION SPECTRUM FROM FISSION       -
CD                     IN GROUP I                                        -
C                                                                        -
C-------------------------------------------------------------------------

C-------------------------------------------------------------------------
CR          ISOTOPE CONTROL AND GROUP INDEPENDENT DATA   (4D RECORD)     -
C                                                                        -
CL    HABSID,HIDENT,HMAT,AMASS,EFISS,ECAPT,TEMP,SIGPOT,ADENS,KBR,ICHI,   -
CL   1IFIS,IALF,INP,IN2N,IND,INT,LTOT,LTRN,ISTRPD,                       -
CL   2(IDSCT(N),N=1,NSCMAX),(LORD(N),N=1,NSCMAX),                        -
CL   3((JBAND(J,N),J=1,NGROUP),N=1,NSCMAX),                              -
CL   4((IJJ(J,N),J=1,NGROUP),N=1,NSCMAX)                                 -
C                                                                        -
CW    3*MULT+17+NSCMAX*(2*NGROUP+2)=NUMBER OF WORDS                      -
C                                                                        -
CB    FORMAT(4H 4D ,3(1X,A6)/6E12.5/                                     -
CB   1(12I6))                                                            -
C                                                                        -
CD    HABSID        HOLLERITH ABSOLUTE ISOTOPE LABEL - SAME FOR ALL      -
CD                            VERSIONS OF THE SAME ISOTOPE IN FILE (A6)-
CD    HIDENT        IDENTIFIER OF LIBRARY FROM WHICH BASIC DATA          -
CD                            CAME (E.G. ENDF/B) (A6)                   -
CD    HMAT          ISOTOPE IDENTIFICATION (E.G. ENDF/B MAT NO.) (A6)    -
CD    AMASS         GRAM ATOMIC WEIGHT                                   -
CD    EFISS         TOTAL THERMAL ENERGY YIELD/FISSION (W.SEC/FISS)      -
CD    ECAPT         TOTAL THERMAL ENERGY YIELD/CAPTURE (W.SEC/CAPT)      -
```

```
CD    TEMP         ISOTOPE TEMPERATURE (DEGREES KELVIN)                -
CD    SIGPOT       AVERAGE EFFECTIVE POTENTIAL SCATTERING IN           -
CD                           RESONANCE RANGE (BARNS/ATOM)             -
CD    ADENS        DENSITY OF ISOTOPE IN MIXTURE IN WHICH ISOTOPE      -
CD                           CROSS SECTIONS WERE GENERATED (A/BARN-CM)-
CD    KBR          ISOTOPE CLASSIFICATION                              -
CD                    0=UNDEFINED                                      -
CD                    1=FISSILE                                        -
CD                    2=FERTILE                                        -
CD                    3=OTHER ACTINIDE                                 -
CD                    4=FISSION PRODUCT                                -
CD                    5=STRUCTURE                                      -
CD                    6=COOLANT                                        -
CD                    7=CONTROL                                        -
CD    ICHI         ISOTOPE FISSION SPECTRUM FLAG                       -
CD                     ICHI.EQ.0,     USE FILE-WIDE CHI                -
CD                     ICHI.EQ.1,     ISOTOPE CHI VECTOR               -
CD                     ICHI.GT.1,     ISOTOPE CHI MATRIX               -
CD    IFIS         (N,F) CROSS SECTION FLAG                            -
CD                    IFIS=0, NO FISSION DATA IN PRINCIPAL CROSS       -
CD                                  SECTION RECORD                     -
CD                        =1, FISSION DATA PRESENT IN PRINCIPAL        -
CD                                  CROSS SECTION RECORD               -
CD    IALF         (N,ALPHA) CROSS SECTION FLAG                        -
CD                    SAME OPTIONS AS IFIS                             -
CD    INP          (N,P) CROSS SECTION FLAG                            -
CD                    SAME OPTIONS AS IFIS                             -
CD    IN2N         (N,2N) CROSS SECTION FLAG                           -
CD                    SAME OPTIONS AS IFIS                             -
CD    IND          (N,D) CROSS SECTION FLAG                            -
CD                    SAME OPTIONS AS IFIS                             -
CD    INT          (N,T) CROSS SECTION FLAG                            -
CD                    SAME OPTIONS AS IFIS                             -
CD    LTOT         NUMBER OF MOMENTS OF TOTAL CROSS SECTION PROVIDED   -
CD                    IN PRINCIPAL CROSS SECTIONS RECORD               -
CD    LTRN         NUMBER OF MOMENTS OF TRANSPORT CROSS SECTION        -
CD                    PROVIDED IN PRINCIPAL CROSS SECTION RECORD       -
CD    ISTRPD       NUMBER OF COORDINATE DIRECTIONS FOR WHICH           -
CD                    COORDINATE DEPENDENT TRANSPORT CROSS SECTIONS    -
CD                    ARE GIVEN, IS ISTRPD=0, NO COORDINATE DEPENDENT -
CD                    TRANSPORT CROSS SECTIONS ARE GIVEN.              -
CD    IDSCT(N)     SCATTERING MATRIX TYPE IDENTIFICATION FOR           -
CD                    SCATTERING BLOCK N, SIGNIFICANT ONLY IF          -
CD                    LORD(N).GT.0                                     -
CD                    IDSCT(N)=000 + NN, TOTAL SCATTERING, (SUM OF     -
CD                        ELASTIC, INELASTIC, AND N2N SCATTERING       -
CD                        MATRIX TERMS),                               -
CD                            =100 + NN, ELASTIC SCATTERING            -
CD                            =200 + NN, INELASTIC SCATTERING          -
CD                            =300 + NN, (N,2N) SCATTERING,----SEE     -
CD                             NOTE BELOW----                          -
CD                    WHERE NN IS THE LEGENDRE EXPANSION INDEX OF THE  -
CD                    FIRST MATRIX IN BLOCK N                          -
```

```
CD    LORD(N)       NUMBER OF SCATTERING ORDERS IN BLOCK N.  IF        -
CD                     LORD(N)=0, THIS BLOCK IS NOT PRESENT FOR THIS   -
CD                     ISOTOPE.  IF NN IS THE VALUE TAKEN FROM         -
CD                     IDSCT(N), THEN THE MATRICES IN THIS BLOCK       -
CD                     HAVE LEGENDRE EXPANSION INDICES OF NN,NN+1,     -
CD                     NN+2,...,NN+LORD(N)-1                           -
CD    JBAND(J,N)    NUMBER OF GROUPS THAT SCATTER INTO GROUP J,         -
CD                     INCLUDING SELF-SCATTER, IN SCATTERING BLOCK N.  -
CD                     IF JBAND(J,N)=0, NO SCATTER DATA IS PRESENT IN  -
CD                     BLOCK N                                         -
CD    IJJ(J,N)      POSITION OF IN-GROUP SCATTERING CROSS SECTION IN   -
CD                     SCATTERING DATA FOR GROUP J, SCATTERING BLOCK   -
CD                     N, COUNTED FROM THE FIRST WORD OF GROUP J DATA. -
CD                     IF JBAND(J,N).NE.0 THEN IJJ(J,N) MUST SATISFY   -
CD                     THE RELATION 1.LE.IJJ(J,N).LE.JBAND(J,N)        -
C                                                                      -
CD                  NOTE- FOR N,2N SCATTER, THE MATRIX CONTAINS TERMS  -
CD                     SCAT(J TO G), WHICH ARE EMISSION (PRODUCTION)-  -
CD                     BASED, I.E., ARE DEFINED SUCH THAT MACROSCOPIC  -
CD                     SCAT(J TO G) TIMES THE FLUX IN GROUP J GIVES    -
CD                     THE RATE OF EMISSION (PRODUCTION) OF NEUTRONS   -
CD                     INTO GROUP G.                                   -
C                                                                      -
C----------------------------------------------------------------------

C----------------------------------------------------------------------
CR          PRINCIPAL CROSS SECTIONS   (5D RECORD)                     -
C                                                                      -
CL     ((STRPL(J,L),J=1,NGROUP),L=1,LTRN),                             -
CL    1((STOTPL(J,L),J=1,NGROUP),L=1,LTOT),(SNGAM(J),J=1,NGROUP).      -
CL    2(SFIS(J),J=1,NGROUP),(SNUTOT(J),J=1,NGROUP),                    -
CL    3(CHISO(J),J=1,NGROUP),(SNALF(J),J=1,NGROUP),                    -
CL    4(SNP(J),J=1,NGROUP),(SN2N(J),J=1,NGROUP),                       -
CL    5(SND(J),J=1,NGROUP),(SNT(J),J=1,NGROUP),                        -
CL    6((STRPD(J,I),J=1,NGROUP),I=1,ISTRPD)                            -
C                                                                      -
CW    (1+LTRN+LTOT+IALF+INP+IN2N+IND+ISTRPD+2*IFIS+                    -
CW    ICHI*(2/(ICHI+1)))*NGROUP=NUMBER OF WORDS                        -
C                                                                      -
CB    FORMAT(4H 5D ,5E12.5/(6E12.5)) LENGTH OF LIST AS ABOVE           -
C                                                                      -
CD     STRPL(J,L)   PL WEIGHTED TRANSPORT CROSS SECTION                -
CD                     THE FIRST ELEMENT OF ARRAY STRPL IS THE         -
CD                     CURRENT (P1) WEIGHTED TRANSPORT CROSS SECTION   -
CD                     THE LEGENDRE EXPANSION COEFFICIENT FACTOR (2L+1) -
CD                     IS NOT INCLUDED IN STRPL(J,L).                  -
CD     STOTPL(J,L)  PL WEIGHTED TOTAL CROSS SECTION                    -
CD                     THE FIRST ELEMENT OF ARRAY STOTPL IS THE        -
CD                     FLUX (P0) WEIGHTED TOTAL CROSS SECTION          -
CD                     THE LEGENDRE EXPANSION COEFFICIENT FACTOR (2L+1) -
CD                     IS NOT INCLUDED IN STOTPL(J,L).                 -
CD     SNGAM(J)     (N,GAMMA)                                         -
CD     SFIS(J)      (N,F)        (PRESENT IF IFIS.GT.0)               -
```

```
CD    SNUTOT(J)    TOTAL NEUTRON YIELD/FISSION (PRESENT IF IFIS.GT.0)  -
CD    CHISO(J)I    ISOTOPE CHI  (PRESENT IF ICHI.EQ.1)                -
CD    SNALF(J)     (N,ALPHA)    (PRESENT IF IALF.GT.0)                -
CD    SNP(J)       (N,P)        (PRESENT IF INP.GT.0)                 -
CD    SN2N(J)      (N,2N)       (PRESENT IF IN2N.GT.0)  ----SEE       -
CD                    NOTE----                                        -
CD    SND(J)       (N,D)        (PRESENT IF IND.GT.0)                 -
CD    SNT(J)       (N,T)        (PRESENT IF INT.GT.0)                 -
CD    STRPD(J,I)   COORDINATE DIRECTION I TRANSPORT CROSS SECTION     -
CD                              (PRESENT IF ISTRPD.GT.0)              -
C                                                                     -
CN                 NOTE - THE PRINCIPAL N,2N CROSS SECTION SN2N(J)    -
CN                    IS DEFINED AS THE N,2N REACTION CROSS SECTION,  -
CN                    I.E., SUCH THAT MACROSCOPIC SN2N(J) TIMES THE   -
CN                    FLUX IN GROUP J GIVES THE RATE AT WHICH N,2N    -
CN                    REACTIONS OCCUR IN GROUP J.  THUS, FOR N,2N     -
CN                    SCATTERING, SN2N(J) = 0.5*(SUM OF SCAT(J TO G)  -
CN                    SUMMED OVER ALL G).                             -
C                                                                     -
C---------------------------------------------------------------------

C---------------------------------------------------------------------
CR          ISOTOPE CHI DATA   (6D RECORD)                            -
C                                                                     -
CC          PRESENT IF ICHI.GT.1                                      -
C                                                                     -
CL    ((CHIISO(K,J),K=1,ICHI),J=1,NGROUP),(ISOPEC(I),I=1,NGROUP)      -
C                                                                     -
CW    NGROUP*(ICHI+1)=NUMBER OF WORDS                                 -
C                                                                     -
CB    FORMAT(4H 6D ,5E12.5/(6E12.5))   CHIISO                         -
CB    FORMAT(12I6)                     ISOPEC                         -
C                                                                     -
CD    CHIISO(K,J)   FRACTION OF NEUTRONS EMITTED INTO GROUP J AS      -
CD                     RESULT OF FISSION IN ANY GROUP,USING SPECTRUM K -
CD    ISOPEC(I)     ISOPEC(I)=K IMPLIES THAT SPECTRUM K IS USED       -
CD                     TO CALCULATE EMISSION SPECTRUM FROM FISSION    -
CD                     IN GROUP I                                     -
C                                                                     -
C---------------------------------------------------------------------

C---------------------------------------------------------------------
CR          SCATTERING SUB-BLOCK   (7D RECORD)                        -
C                                                                     -
CC          PRESENT IF LORD(N).GT.0                                   -
C                                                                     -
CL    ((SCAT(K,L),K=1,KMAX),L=1,LORDN)                                -
C                                                                     -
CC    KMAX=SUM OVER J OF JBAND(J,N) WITHIN THE J-GROUP RANGE OF THIS  -
CC       SUB-BLOCK.  IF M IS THE INDEX OF THE SUB-BLOCK, THE J-GROUP  -
CC       RANGE CONTAINED WITHIN THIS SUB-BLOCK IS                     -
CC       JL=(M-1)*((NGROUP-1)/NSBLOK+1 TO JU=MIN0(NGROUP,JUP),        -
CC       WHERE JUP=M*((NGROUP-1)/NSBLOK+1).                           -
```

```
C                                                                      -
CC    LORDN=LORD(N)                                                    -
CC    N IS THE INDEX FOR THE LOOP OVER NSCMAX (SEE FILE STRUCTURE)     -
C                                                                      -
CW    KMAX*LORDN=NUMBER OF WORDS                                       -
C                                                                      -
CB    FORMAT(4H 7D ,5E12.5/(6E12.5))                                   -
C                                                                      -
CD    SCAT(K,L)     SCATTERING MATRIX OF SCATTERING ORDER L, FOR       -
CD                     REACTION TYPE IDENTIFIED BY IDSCT(N) FOR THIS   -
CD                     BLOCK, JBAND(J,N) VALUES FOR SCATTERING INTO    -
CD                     GROUP J ARE STORED AT LOCATIONS K=SUM FROM 1    -
CD                     TO (J-1) OF JBAND(J,N) PLUS 1 TO K-1+JBAND(J,N).-
CD                     THE SUM IS ZERO WHEN J=1, J-TO-J SCATTER IS     -
CD                     THE IJJ(J,N)-TH ENTRY IN THE RANGE JBAND(J,N),  -
CD                     VALUES ARE STORED IN THE ORDER (J+JUP),         -
CD                     (J+JUP-1),...,(J+1),J,(J-1),...,(J-JDN),        -
CD                     WHERE JUP=IJJ(J,N)-1 AND JDN=JBAND(J,N)-IJJ(J,N)-
C                                                                      -
C----------------------------------------------------------------------
```

Most of the variables in the "File Identification and File Control" record are taken from the user's input. Note that MAXUP is always set to zero. CCCCR does not process the NJOY thermal data at the present time. The ICHIST parameter will always be zero. CCCCR does not produce a file-wide fission spectrum or matrix. The old practice of using a single fission spectrum for all calculations is inaccurate and obsolete. Actually, the effective fission spectrum depends on the mixture of isotopes and the flux. Any file-wide spectrum would have to be at least problem dependent, and it should also be region dependent. The parameters NSCMAX and NSBLOK in the "File Control" record will be discussed in connection with the scattering matrix format.

In the "File Data" record, the Hollerith set identification and the isotope names are taken from the user's input. As mentioned above, the file-wide fission spectrum CHI never appears. The mean neutron velocities by group (VEL) are obtained from the inverse velocities computed by GROUPR:

$$\left\langle \frac{1}{v} \right\rangle_g = \frac{\int_g \frac{1}{v}\phi(E)\,dE}{\int_g \phi(E)\,dE}, \tag{1}$$

where g is the group index, $\phi(E)$ is the GROUPR weighting spectrum, and v is the neutron velocity, which is computed from the neutron mass and energy using $v=\sqrt{2E/m}$. The units of these quantities are s/m; they are converted to cm/s for ISOTXS by inverting and multiplying by 100. The group structure [see

EMAX(J) and EMIN] is obtained directly from MF1,MT451 on the GENDF tape. Note that GROUPR energy groups are given in order of increasing energy and ISOTXS energy groups are given in order of decreasing energy. CCCCR handles the conversion.

The "File-Wide Chi Data" record never appears; see the discussion above for the reasons.

In the "Isotope Control and Group Independent Data" record, the first ten parameters are taken from the user's input. The gram atomic weight for the material (AMASS) can be computed from the ENDF AWR parameter available on the GENDF file using the gram atomic weight of the neutron as a multiplier. The energy-release parameters EFISS and ECAPT must also be computed by the user. The ECAPT values are normally based on the ENDF Q values given in File 3, but, in some cases, it is also necessary to add additional decay energy coming from short-lived activation steps. For example, the ECAPT value for ^{238}U should include the energy for the ^{239}U β-decay step, and perhaps even the energy from the ^{239}Np β decay. The values for EFISS should be based on the total non-neutrino energy release, which can be obtained from MF3,MT18 or MF1,MT458 on the ENDF tape. The TEMP parameter is normally set to 300 K. The value of SIGPOT can be computed from the scattering-radius parameter AP in File 2 of the ENDF tape using $\sigma_p = 4\pi a^2$. The parameter ADENS is usually set to zero to imply infinite dilution. KBR can be chosen based on the normal use of the material by the community for which the library is being produced.

The ICHI parameter is related to the ICHIX parameter in the user's input. As discussed above, the option ICHI=0 is never used by CCCCR. Beyond that, the NJOY user has the option of producing a fission χ vector using the default GROUPR flux (which is available on the GENDF tape) or a user-supplied weighting flux SPEC. This enables the user to produce an ISOTXS library appropriate to a class of problems with a flux similar to SPEC. In general, the incident-energy dependence of the fission spectrum is weak, so the choice of this weighting spectrum is not critical. Noticeable differences might be expected between a thermal spectrum on the one hand and a fast-reactor or fusion-blanket spectrum on the other. The χ vector is defined as follows:

$$\chi_{g'} = \frac{\sum_g \sigma_{fg\to g'} \phi_g}{\sum_{g'} \sum_g \sigma_{fg\to g'} \phi_g} , \qquad (2)$$

where σ_f is the fission group-to-group matrix from GROUPR, ϕ_g is either the model flux or SPEC, and the denominator assures that $\chi_{g'}$ will be normalized. Actually, the calculation is more complicated than that because of the necessity to include delayed-neutron production. A "steady-state" value for the fission spectrum can be obtained as follows:

$$\chi_{g'}^{SS} = \frac{\sum_g \sigma_{fg\to g'}\phi_g + \chi_{g'}^D \sum_g \bar{\nu}_g^D \sigma_{fg}\phi_g}{\sum_{g'}\sum_g \sigma_{fg\to g'}\phi_g + \sum_g \bar{\nu}_g^D \sigma_{fg}\phi_g} , \tag{3}$$

where $\bar{\nu}_g^D$ is the delayed-neutron yield obtained from MF3,MT455 on the GENDF tape, χ_g^D is the total delayed-neutron spectrum obtained by summing over the 6 time groups in MF5,MT455, and σ_{fg} is the fission cross section for group g obtained from MF3,MT18.

This is still not the end of the complications of fission. If the partial fission reactions MT19, 20, 21, and 38 are present, the fission matrix term in the above equations is obtained by adding the contributions from all the partial reactions found. In these cases, a matrix for MT18 will normally not be present on the GENDF tape. If it is, it will be ignored. Beginning with NJOY 91.0, a new and more efficient representation is used for the fission matrix computed in GROUPR. It is well known that the shape of the fission spectrum is independent of energy up to energies of several hundred keV. GROUPR takes advantage of this by computing this low-energy spectrum only once. It then computes a fission neutron production cross section for all the groups up to the energy at which significant energy dependence starts. At higher energies, the full group-to-group fission matrix is computed as in earlier versions of NJOY. Therefore, it is now necessary to compute the values of $\sigma_{fg\to g'}$ as used in the above equations using

$$\sigma_{fg\to g'} = \chi_{g'}^{LE}(\bar{\nu}\sigma_f)_g^{LE} + \sigma_{fg\to g'}^{HE} , \tag{4}$$

where LE stands for low energy, HE stands for high energy; the low-energy production cross sections written as $\nu\sigma_f$ will be found on the GENDF tape using the special flag `IG2LO=0`, and the low-energy χ will be found on the GENDF tape with `IG=0`.

In order to obtain still better accuracy, CCCCR can produce a fission χ matrix instead of the vector. Using the above notation, the full χ matrix becomes

$$\chi_{g\to g'}^{SS} = \frac{\chi_{g'}^{LE}(\bar{\nu}\sigma_f)_g^{LE} + \sigma_{fg\to g'}^{HE} + \chi_{g'}^D \bar{\nu}_g^D \sigma_{fg}}{\text{NORM}} , \tag{5}$$

where NORM is just the value that normalizes the χ matrix; that is, the sum of the numerator over all g'. Note that the "Isotope Chi Data" record allows for a rectangular fission matrix similar to the one produced by GROUPR. It is obtained by using the input SPEC array to define the range of groups that will be averaged into each of the final ICHIX spectra. For example, to collapse a ten-group χ matrix into a five-group matrix, SPEC might contain the ten values 1, 2, 3, 4, 5, 5, 5, 5, 5, 5. More formally,

$$\chi^{SS}_{k\to g'} = \frac{\sum_{s(g)=k} (\chi^{LE}_{g'}(\bar{\nu}\sigma_f)^{LE}_g + \sigma^{HE}_{fg\to g'} + \chi^{D}_{g'}\bar{\nu}^{D}_g\sigma_{fg})\,\phi_g}{\text{NORM}}, \qquad (6)$$

where ϕ_g is the default weighting function from the GENDF tape, $s(g)$ is the SPEC array provided by the user, and the summation is over all groups g satisfying the condition that $s(g)=k$. Future versions of CCCCR could construct the SPEC array automatically using the information in the new GENDF format.

Continuing with the description of the "Isotope Control and Group Independent Data" record, the next 9 parameters are flags that tell what reactions will be described in the "Principal Cross Sections" record. They will be described below. Similarly, the parameters IDSCT, LORDN, JBAND, and IJJ will be described later in connection with the "Scattering Sub-Block" records.

The ISOTXS format allows for a fixed set of principal cross sections that was chosen based on the needs of fission reactor calculations. This is one of its main defects; the list does not allow for other reactions that become important above 6–10 MeV, and it does not allow for other quantities of interest, such as gas production, KERMA factors, and radiation damage production cross sections. Most of the reactions are simply copied from MF3 on the GENDF tape with the group order inverted—this is true for SNGAM, the (n,γ) cross section, which is taken from MT102; for SFIS, the (n,f) cross section, which is taken from MT18; for SNALF, the (n,α) cross section, which is taken from MT107; for SNP, the (n,p) cross section, which is taken from MT103; for SND, the (n,d) cross section, which is taken from MT104; and for SNT, the (n,t) cross section, which is taken from MT105. The (n,2n) cross section, SN2N, is normally taken from MT16. However, earlier versions of ENDF represented the sequential (n,2n) reaction in ^{9}Be using MT6, 7, 8, and 9. If present, these partial (n,2n) reactions are added into SN2N. The flags IFIS, IALF, INP, IN2N, IND, and INT in the "Isotope Control and Group Independent Data" record are set to indicate which of these reactions have been found for this material.

In going from version III of the CCCC specifications to version IV, there was some controversy over the appropriate definition for the (n,2n) cross section and matrix. It was decided that the quantity in SN2N would be the (n,2n) reaction cross section; that is, it would define the probability that an (n,2n) reaction takes place. The (n,2n) matrix would be defined such that the sum over all secondary groups would produce the (n,2n) production cross section, which is two times larger than the reaction cross section.

The CHISO vector, which contains the fission spectrum vector (if any), was discussed above. A complete calculation of the fission source also requires the fission yield, SNUTOT, which can be used together with the fission cross section to calculated the fission neutron production cross section, $\bar{\nu}\sigma_f$. The fission yield can be calculated from the GROUPR fission matrix using

$$\bar{\nu}_g = \frac{\sum_{g'} \sigma_{fg\to g'}}{\sigma_{fg}} \ . \tag{7}$$

Adding delayed neutron contributions and accounting for the partition of the fission matrix into low-energy and high-energy parts (see the discussion of χ above) gives the equation actually used by CCCCR:

$$\bar{\nu}_g^{SS} = \frac{\sum_{g'} \sigma_{fg\to g'}^{HE} + (\bar{\nu}\sigma_f)_g^{LE} + \bar{\nu}_g^D \sigma_{fg}}{\sigma_{fg}} \ . \tag{8}$$

The total cross section produced by GROUPR contains two components: the flux-weighted or P_0 total cross section, and the current-weighted or P_1 total cross section. The P_0 part is stored into STOTPL, and the LTOT flag is set to 1. STRPL contains the transport cross section used by diffusion codes; this is,

$$\sigma_{tr,g} = \sigma_{t1,g} - \sum_{g'} \sigma_{e1,g\to g'} \ , \tag{9}$$

where $\sigma_{t1,g}$ is the P_1 total cross section and $\sigma_{e1,g\to g'}$ is the P_1 component of the elastic scattering matrix, which is obtained from MF6,MT2 on the GENDF tape from GROUPR. The flag LTRN is set to 1; that is, no higher-order transport corrections are provided. Direction-dependent transport cross sections are not computed by CCCCR; therefore, ISTRPD is always zero, and the STRPD vectors are missing.

As discussed above, the "Isotope Chi Data" record may be present if the user set ICHIX>1 and supplied a SPEC vector to define how the full χ matrix is to be collapsed into a rectangular χ matrix.

The treatment of scattering matrices in the ISOTXS format is complex and has lots of possible variations. Only the variations supported by CCCCR will be described here. First of all, the scattering data is divided into blocks and subblocks. A block is either one of the designated scattering reactions [that is, total, elastic, inelastic, or (n,2n)] and contains all the group-to-group elements and Legendre orders for that reaction (IFOPT=1), or it is one particular Legendre order for one of the designated reactions and contains all the group-to-group elements for that order and reaction (IFOPT=2). Its actual content is determined by IDSCT and LORDN. If IFOPT=1 has been selected, IDSCT(1)=100 and LORDN(1)=4 would designate a block for the elastic scattering matrix of order P_3 that contains all 4 Legendre orders and all group-to-group elements. If IFOPT=2 has been selected, IDSCT(1)=100 with LORDN(1)=1 would designate a block containing all group-to-group elements for the P_0 elastic matrix, IDSCT(2)=101 with LORDN(2)=1 would designate a block containing the P_1 elastic matrix, and so on. In CCCCR, LORDN is always equal to 1 for IFOPT=2.

The ISOTXS format attempts to pack scattering matrices efficiently. First, all the scattering matrices treated here are triangular because only downscatter is present. And second, because of the limited range of elastic downscatter, only a limited range of groups above the inscatter group will contribute to the scattering into a given secondary-energy group. Therefore, ISOTXS removes zero cross sections by defining bands of incident energy groups that contribute to each final energy group. The bands are defined by JBAND, the number of initial energy groups in the band, and IJJ, an index to identify the position of the ingroup element in the band. The following table illustrates banding for a hypothetical elastic scattering reaction:

Band	Element	JBAND	IJJ
1	1→1	1	1
2	2→2	2	1
	1→2		
3	3→3	2	1
	2→3		
4	4→4	2	1
	3→4		

Note that IJJ is always 1 in the absence of upscatter. This scheme is efficient for elastic scattering, but it is not efficient for threshold reactions because the ingroup element must always be included in the band. This means that lots of zeros must

be given for final energy groups below the threshold group. An improved and simplified scheme is used in the MATXS format.

For IFOPT=2, the elements in the table above would be stored in the sequence shown, top to bottom. Each Legendre order would have its own block arranged in the same order. For IFOPT=1, the Legendre order data are intermixed with the group-to-group data. In each band, the elements for all initial groups for P_0 are given, then all initial groups for P_1, and so on through LORDN Legendre orders.

Scattering matrices can be very large. For example, an 80-group elastic matrix can have as many as $80 \times 79/2 = 3160$ elements per Legendre order. That would be 12 640 words for a P_3 block using IFOPT=1, or four blocks of 3160 words each for IFOPT=2. The latter is practical; the former is not. The corresponding numbers for 200 groups would be 79 600 and 19 900. Both of these numbers are clearly impractical as record sizes. This is where subblocking comes in. If each block is divided up so that there is one subblock for each energy group, the maximum record size is reduced substantially. For IFOPT=1, the maximum record size is equal to the number of groups times the number of Legendre orders, or 800 for the 200-group P_3 case. For IFOPT=2, the maximum record size is just equal to the number of groups. Although the ISOTXS formats allows subblocks to contain several groups, CCCCR does not. The possible values of NSBLOK are limited to 1 and NGROUP. In summary, the CCCCR user has four matrix blocking options:

1. *IFOPT=1 and NSBLOK=1.* This produces a single block and a single subblock for each reaction. It is probably the best choice for small group structures (up to about 30 groups). The maximum record size is $n_\ell \times n_g(n_g-1)/2$.

2. *IFOPT=1 and NSBLOK=NGROUP.* This produces a block for each reaction, and each block contains n_g subblocks. The maximum record size is $n_\ell \times n_g$. This is a good choice for larger group structures because it keeps the record size up as compared with option 4 below.

3. *IFOPT=2 and NSBLOK=1.* This produces n_ℓ blocks and subblocks for each reaction. The maximum record size is $n_g(n_g-1)/2$. It has only a modest advantage in the maximum number of groups over option 1. Unless the application that uses the library finds it convenient to read one Legendre order at a time, the user might just as well choose option 2 if option 1 produces records that are too large.

4. *IFOPT=2 and NSBLOK=NGROUP.* This produces n_ℓ blocks for each reaction, each with n_g subblocks. The maximum record size is n_g. The number of groups would have to be on the order of 1000 before this option would be preferred to option 2.

If CCCCR does not have enough memory to process option 1 or 3, the code automatically sets NSBLOK to NGROUP, thereby activating option 2 or 4, respectively.

Note that the ISOTXS format specifies that the total scattering matrix is the sum of the elastic, inelastic, and (n,2n) matrices [see the definition of IDSCT(N)]. This implies that the inelastic matrix must contain the normal (n,n′) reactions from MT51–91, and also any other neutron-producing reactions that might be present. Examples are (n,n′α), (n,n′p), and (n,3n).

D. BRKOXS

The format for the BRKOXS Bondarenko-type self-shielding factor file is given below in the standard format.

```
C*************************************************************************
C                        REVISED 11/30/76                                  -
C                                                                          -
CF          BRKOXS-IV                                                      -
CE          MICROSCOPIC GROUP DELAYED NEUTRON PRECURSOR DATA               -
C                                                                          -
CN                       THIS FILE PROVIDES DATA NECESSARY FOR             -
CN                       BONDARENKO TREATMENT IN ADDITION TO               -
CN                       THOSE DATA IN FILE ISOTXS                         -
CN                       FORMATS GIVEN ARE FOR FILE EXCHANGE PURPOSES      -
CN                       ONLY.                                             -
C                                                                          -
C*************************************************************************

C--------------------------------------------------------------------------
CS          FILE STRUCTURE                                                 -
CS                                                                         -
CS             RECORD TYPE                          PRESENT IF             -
CS             ================================     ===============       -
CS             FILE IDENTIFICATION                  ALWAYS                -
CS             FILE CONTROL                         ALWAYS                -
CS             FILE DATA                            ALWAYS                -
CS   **************(REPEAT FROM 1 TO NISOSH)                              -
CS   *         SELF-SHIELDING FACTORS               ALWAYS                -
CS   *         CROSS SECTIONS                       ALWAYS                -
CS   **************                                                        -
C                                                                          -
C--------------------------------------------------------------------------

C--------------------------------------------------------------------------
CR          FILE IDENTIFICATION                                            -
C                                                                          -
CL    HNAME,(HUSE(I),I=1,2),IVERS                                          -
C                                                                          -
CW    1+3*MULT=NUMBER OF WORDS                                             -
C                                                                          -
CB    FORMAT(11H OV BRKOXS  ,1H*,2A6,1H*,I6)                               -
C                                                                          -
CD    HNAME       HOLLERITH FILE NAME - BRKOXS -                           -
```

```
CD    HUSE(I)     HOLLERITH USER IDENTIFICATION (A6)                    -
CD    IVERS       FILE VERSION NUMBER                                   -
CD    MULT        DOUBLE PRECISION PARAMETER                            -
CD                    1- A6 WORD IS SINGLE WORD                         -
CD                    2- A6 WORD IS DOUBLE PRECISION WORD               -
C                                                                       -
C-----------------------------------------------------------------------

C-----------------------------------------------------------------------
CR          FILE CONTROL   (1D RECORD)                                  -
C                                                                       -
CL    NGROUP,NISOSH,NSIGPT,NTEMPT,NREACT,IBLK                           -
C                                                                       -
CW    6 = NUMBER OF WORDS                                               -
C                                                                       -
CB    FORMAT(4H 1D ,6I6)                                                -
C                                                                       -
CD    NGROUP        NUMBER OF ENERGY GROUPS IN SET                      -
CD    NISOSH        NUMBER OF ISOTOPES WITH SELF-SHIELDING FACTORS      -
CD    NSIGPT        TOTAL NUMBER OF VALUES OF VARIABLE X (SEE FILE DATA-
CD                     RECORD) WHICH ARE GIVEN.  NSIGPT IS EQUAL TO     -
CD                     THE SUM FROM 1 TO NISOSH OF NTABP(I)             -
CD    NTEMPT        TOTAL NUMBER OF VALUES OF VARIABLE TB (SEE FILE     -
CD                     DATA RECORD) WHICH ARE GIVEN.  NTEMPT IS EQUAL   -
CD                     TO THE SUM FROM 1 TO NISOSH OF NTABT(I)          -
CD    NREACT        NUMBER OF REACTION TYPES FOR WHICH SELF-SHIELDING   -
CD                     FACTORS ARE GIVEN (IN PREVIOUS VERSIONS OF THIS -
CD                     FILES NREACT HAS BEEN IMPLICITLY SET TO 5).      -
CD    IBLK          BLOCKING OPTION FLAG FOR SELF-SHIELDING FACTORS,    -
CD                     IBLK=0, FACTORS NOT BLOCKED BY REACTION TYPE,    -
CD                     IBLK=1, FACTORS ARE BLOCKED BY REACTION TYPE.    -
C                                                                       -
C-----------------------------------------------------------------------

C-----------------------------------------------------------------------
CR          FILE DATA   (2D RECORD)                                     -
C                                                                       -
CL    (HISONM(I),I=NISOSH),(X(K),K=1,NSIGPT),(TB(K),K=1,NTEMPT),        -
CL   1(EMAX(J),J=1,NGROUP),EMIN,(JBFL(I),I=1,NISOSH),                   -
CL   2(JBFH(I),I=NISOSH),(NTABP(I),I=1,NISOSH),(NTABT(I),I=1,NISOSH)    -
C                                                                       -
CW    (4+MULT)*NISOSH+NSIGPT+NTEMPT+NGROUP+1=NUMBER OF WORDS            -
C                                                                       -
CB    FORMAT(4H 2D ,9(1X,A6)/          HISONM                           -
CB   1(10(1X,A6)))                                                      -
CB    FORMAT(6E12.5)                   X,TB,EMAX,EMIN                   -
CB    FORMAT(12I6)                     JBFL,JBFH,NTABP,NTABT            -
C                                                                       -
CD    HISONM(I)     HOLLERITH ISOTOPE LABEL FOR ISOTOPE I (A6).  THESE -
CD                     LABELS MUST BE A SUBSET OF THOSE IN FILE ISOTXS -
CD                     OR GRUPXS, IN THE CORRESPONDING ARRAY.           -
CD    X(K)          ARRAY OF LN(SIGPO)/LN(10) VALUES FOR ALL ISOTOPES,  -
CD                     WHERE SIGPO IS THE TOTAL CROSS SECTION OF THE    -
```

```
CD                    OTHER ISOTOPES IN THE MIXTURE IN BARNS PER ATOM -
CD                    OF THIS ISOTOPE.  FOR ISOTOPE I, THE NTABP(I)   -
CD                    VALUES OF X FOR WHICH SELF-SHIELDING FACTORS    -
CD                    ARE GIVEN ARE STORED STARTING AT LOCATION L=1+  -
CD                    SUM FROM 1 TO I-1 OF NTABP(K).                  -
CD    TB(K)          ARRAY OF TEMPERATURES (DEGREES C) FOR ALL ISOTOPES.-
CD                    FOR ISOTOPE I, THE NTBT(I) VALUES OF TB FOR     -
CD                    WHICH SELF-SHIELDING FACTORS ARE GIVEN ARE      -
CD                    STORED AT LOCATION L=1+SUM FROM 1 TO I-1 OF     -
CD                    NTABT(K).                                       -
CD    EMAX(J)        MAXIMUM ENERGY BOUND OF GROUP J (EV)            -
CD    EMIN           MINIMUM ENERGY BOUND OF SET (EV)                -
CD    JBFL(I)        LOWEST NUMBERED OF HIGHEST ENERGY GROUP FOR WHICH -
CD                    SELF-SHIELDING FACTORS ARE GIVEN                -
CD    JBFH(I)        HIGHEST NUMBERED OR LOWEST ENERGY GROUP FOR WHICH -
CD                    SELF-SHIELDING FACTORS ARE GIVEN                -
CD    NTABP(I)       NUMBER OF SIGPO VALUES FOR WHICH SELF-SHIELDING  -
CD                    FACTORS ARE GIVEN FOR ISOTOPE I.                -
CD    NTABT(I)       NUMBER OF TEMPERATURE VALUES FOR WHICH SELF-     -
CD                    SHIELDING FACTORS ARE GIVEN FOR ISOTOPE I.      -
C                                                                     -
C---------------------------------------------------------------------

C---------------------------------------------------------------------
CR          SELF-SHIELDING FACTORS   (3D RECORD) .                    -
C                                                                     -
CL    ((((FFACT(N,K,J,M),N=1,NBINT),K=1,NBTEM),J=JBFLI,JBFHI),M=ML,MU) -
CL                    ------ SEE DESCRIPTION BELOW ------             -
C                                                                     -
CC    NBINT=NTABP(I)                                                  -
CC    NBTEM=NTABT(I)                                                  -
CC    JBFLI=JBFL(I)                                                   -
CC    JBFHI=JBFH(I)                                                   -
CC    FOR ML, MU SEE STRUCTURE BELOW                                  -
C                                                                     -
CW    NBINT*NBTEM*(JBFHI-JBFLI+1)*(MU-ML+1) = NUMBER OF WORDS         -
C                                                                     -
CB    FORMAT(4H 3D ,5E12.5/(6E12.5))                                  -
C                                                                     -
CC    DO 1 L=1,NBLOK                                                  -
CC  1 READ(N) *LIST AS ABOVE*                                         -
C                                                                     -
CC          IF IBLK=0, NBLOK=1, ML=1, MU=NREACT                       -
CC          IF IBLK=1, NBLOK=NREACT, ML=MU=L, WHERE L IS THE BLOCK    -
C                                                                     -
CD    FFACT(N,K,J,M)    SELF-SHIELDING FACTOR EVALUATED AT X(N) AND   -
CD                         TB(K) FOR ENERGY GROUP J, THE M INDEX IS   -
CD                         A DUMMY INDEX TO DENOTE THE REACTION TYPE, -
CD                         THE FIRST FIVE REACTION TYPES ARE, IN      -
CD                         ORDER, TOTAL, CAPTURE, FISSION, TRANSPORT, -
CD                         AND ELASTIC.                               -
C                                                                     -
CB          NOTE THAT IS IBLK=1, EACH REACTION TYPE WILL CONSTITUTE   -
```

```
CB              A SEPARATE DATA BLOCK.                                  -
C                                                                       -
C-----------------------------------------------------------------------

C-----------------------------------------------------------------------
CR          CROSS SECTIONS   (4D RECORD)                                -
C                                                                       -
CL    (XSPO(J),J=1,NGROUP),(XSIN(J),J=1,NGROUP),(XSE(J),J=1,NGROUP),    -
CL   1(XSMU(J),J=1,NGROUP),(XSED(J),J=1,NGROUP),(XSX(J),J=1,NGROUP)     -
C                                                                       -
CW    6*NGROUP=NUMBER OF WORDS                                          -
C                                                                       -
CB    FORMAT(4H 4D ,5E12.5/(6E12.5))                                    -
C                                                                       -
CD    XSPO(J)       POTENTIAL SCATTERING CROSS SECTION (BARNS)          -
CD    XSIN(J)       INELASTIC CROSS SECTION (BARNS)                     -
CD    XSE(J)        ELASTIC CROSS SECTION (BARNS)                       -
CD    XSMU(J)       AVERAGE COSINE OF ELASTIC SCATTERING ANGLE          -
CD    XSED(J)       ELASTIC DOWN-SCATTERING TO ADJACENT GROUP           -
CD    XSXI(J)       AVERAGE ELASTIC SCATTERING LETHARGY INCREMENT       -
C                                                                       -
C-----------------------------------------------------------------------
```

The BRKOXS file is used to supply self-shielding factors for use with the Bondarenko method[13] for calculating effective macroscopic cross sections for the components of nuclear systems. As discussed in detail in the GROUPR chapter of this manual, this system is based on using a model flux for isotope i of the form

$$\phi_\ell^i(E,T) = \frac{C(E)}{[\sigma_0^i + \sigma_t^i(E,T)]^{\ell+1}} , \tag{10}$$

where $C(E)$ is a smooth weighting flux, $\sigma_t^i(E,T)$ is the total cross section for material i at temperature T, ℓ is the Legendre order, and σ_0^i is a parameter that can be used to account for the presence of other materials and the possibility of escape from the absorbing region (heterogeneity). GROUPR uses this model flux to calculate effective multigroup cross sections for the resonance-region reactions (total, elastic, fission, capture) for several values of σ_0 and several values of T.

When σ_0 is large with respect to the highest peaks in σ_t, the flux is essentially proportional to $C(E)$. This is called infinite dilution, and the corresponding cross sections are appropriate for an absorber in a dilute mixture or for a very thin sample of the absorber. As σ_0 decreases, the flux $\phi(E)$ develops dips where σ_t has peaks. These dips cancel out part of the effect of the corresponding peaks in the resonance cross sections, thereby reducing, or self-shielding the reaction rate. It is convenient to represent this effect as a self-shielding factor; that is,

$$\sigma_{xg}(T,\sigma_0) = f_{xg}(T,\sigma_0) \times \sigma_{xg}(300K,\infty)\ . \qquad (11)$$

In the CCCC system, the f-factors are stored in the BRKOXS file and the infinitely dilute cross sections are stored in the ISOTXS file. A code that prepares effective cross sections, such as SPHINX[7] or 1DX,[14] determines the appropriate T and σ_0 values for each region and group in a reactor problem, using equivalence theory together with the user's specifications for composition and geometry. It then reads in the cross sections and f-factors from the ISOTXS and BRKOXS files, interpolates in the T and σ_0 tables to obtain the desired f-factors, and multiplies to obtain the effective cross sections.

In the BRKOXS file specification given above, the components of the "File Identification" record are the same as for ISOTXS. The parameters in the "File Control" record are used to calculate the sizes and locations of data in the records to follow. The "File Data" record contains all the isotope names, all the σ_0 values for every isotope, all the T values for every isotope, the group structure, and several arrays used for unpacking the other data. The Hollerith material names in `HISONM` are the same as those used in ISOTXS, and they are obtained from the user's input. The σ_0 values are packed into `X(K)` using `NTABP(I)`. Note that the base-10 logarithm of σ_0 is stored. Therefore, a typical library might contain the following:

material I	NTABP(I)	X(K)
1	3	3.0 2.0 1.0
2	7	4.0 3.0 2.0 1.477 1. 0. -1.
3	5	4.0 3.0 2.0 1.0 0.0
...	...	...

Similarly, the T values are stored in `TB(K)` using `NTABT(I)`. The absolute temperatures used by GROUPR must be changed to degrees C before being stored. The energy bounds for the group structure found on the GENDF tape are stored in `EMAX(J)` and `EMIN`. As discussed in connection with the ISOTXS format, group boundaries are stored in order of decreasing energy in the BRKOXS file.

The self-shielding factor approach is designed to account for resonance self-shielding. It is not necessarily appropriate for low energies if only broad resonance features are apparent, or for high energies where only small residual fluctuations in the cross section are seen. For this reason, the BRKOXS format provides the `JBFL(I)` and `JBFH(I)` arrays in the "File Data" record. They are used to limit the range of the numbers given in the "Self-Shielding Factors" record. The reactions

that are active in the resonance energy range usually include only the total, elastic, fission, and capture channels. The total cross section is usually computed for the two Legendre orders ℓ=0 and ℓ=1. This second value is often called the current-weighted total cross section, and it is needed to compute the self-shielded diffusion coefficient. GROUPR also computes a self-shielded elastic scattering matrix. It can be used to provide two quantities for the BRKOXS file. First, the diffusion coefficient requires the calculation of a transport cross section for diffusion. The relationships are as follows:

$$D_g = \frac{1}{3\sigma_{tr,g}} , \tag{12}$$

and

$$\sigma_{tr,g} = \sigma_{t1,g} - \sum_{g'} \sigma_{e1,g\to g'} . \tag{13}$$

Therefore, the current-weighted P_1 elastic cross section contributes to the transport self-shielding factor. The second use for the self-shielded elastic scattering matrix is to compute a self-shielding factor for elastic removal. For heavy isotopes, the energy lost in elastic scattering is small, and all the removal is normally to one group. The format requires that at least the five standard reactions be given in a specified order. NJOY is able to add one more as follows:

1. total (P_0 weighted),
2. capture (P_0),
3. fission (P_0),
4. transport (P_1),
5. elastic (P_0), and
6. elastic removal (P_0).

The normal pattern for the BRKOXS file expects that there will be one record of self-shielding factors for each material. Such a record could get quite large. For example, with 6 reactions, 6 temperatures, 6 σ_0 values, and 100 resonance groups, a "Self-Shielding Factors" record could have over 20 000 words. This number can be made more manageable by setting the `IBLK` flag in the "File Control" record to 1. Then there will be a separate record of self-shielding factors for each reaction; this would reduce the record size for the example to a more reasonable 3600 words.

The actual self-shielding factors are computed from the cross sections given on the GENDF tape and stored into the `FFACT(N,K,J,M)` array of the "Self-Shielding Factors" record with group order converted to the standard decreasing-energy convention. Following the FORTRAN convention, `N` is the fastest varying index in

this array, K is the next fastest varying index, and so on. The identities of these indices are

N σ_0,

K temperature T,

J group, and

M reaction.

The "Cross Sections" record contains some additional cross sections and special parameters that are often used in self-shielding codes and are not included in the ISOTXS file. The XSPO cross section is taken to be constant and equal to the CCCCR input parameter XSPO, which is computed as $4\pi a^2$. The XSIN cross section is obtained by summing over the final group index for every group-to-group matrix in File 6 except elastic, (n,2n), and fission. Therefore, it may contain effects of multiplicities greater than 1 if reactions like (n,3n) or (n,2nα) are active for the material. It may be slightly larger in high-energy groups than the cross section that would be obtained using the same sum over reactions in File 3. The XSE cross section is obtained from MF6,MT2 on the GENDF tape by summing over final groups. The parameters for continuous slowing down theory, XSMU and XSXI, are obtained from MT251 and MT252, respectively. The methods used for calculating these quantities are described in the GROUPR section of this report. Finally, the elastic downscattering cross section is obtained from the elastic matrix on the GENDF tape (MF6,MT2).

E. DLAYXS

The format for the DLAYXS delayed-neutron data file is given below in the standard format.

```
C*************************************************************************
C                        REVISED 11/30/76                                -
C                                                                        -
CF          DLAYXS-IV                                                    -
CE          MICROSCOPIC GROUP DELAYED NEUTRON PRECURSOR DATA             -
C                                                                        -
CN                       THIS FILE PROVIDES PRECURSOR YIELDS,            -
CN                       EMISSION SPECTRA, AND DECAY CONSTANTS           -
CN                       ORDERED BY ISOTOPE.  ISOTOPES ARE IDENTIFIED    -
CN                       BY ABSOLUTE ISOTOPE LABELS FOR RELATION TO      -
CN                       ISOTOPES IN EITHER FILE ISOTXS OR GRUPXS.       -
CN                       FORMATS GIVEN ARE FOR FILE EXCHANGE PURPOSES    -
CN                       ONLY.                                           -
C                                                                        -
```

```
C*************************************************************************

C-------------------------------------------------------------------------
CS          FILE STRUCTURE                                               -
CS                                                                       -
CS             RECORD TYPE                          PRESENT IF           -
CS             ================================     ===============      -
CS             FILE IDENTIFICATION                  ALWAYS               -
CS             FILE CONTROL                         ALWAYS               -
CS             FILE DATA, DECAY CONSTANTS, AND                           -
CS                EMISSION SPECTRA                  ALWAYS               -
CS   *************(REPEAT TO NISOD)                                      -
CS   *          DELAYED NEUTRON PRECURSOR                                 -
CS   *             YIELD DATA                        ALWAYS               -
CS   *************                                                        -
C                                                                        -
C-------------------------------------------------------------------------

C-------------------------------------------------------------------------
CR          FILE IDENTIFICATION                                          -
C                                                                        -
CL    HNAME,(HUSE(I),I=1,2),IVERS                                        -
C                                                                        -
CW    1+3*MULT=NUMBER OF WORDS                                           -
C                                                                        -
CB    FORMAT(11H OV DLAYXS  ,1H*,2A6,1H*,I6)                             -
C                                                                        -
CD    HNAME       HOLLERITH FILE NAME - DLAYXS -                         -
CD    HUSE(I)     HOLLERITH USER IDENTIFICATION (A6)                     -
CD    IVERS       FILE VERSION NUMBER                                    -
CD    MULT        DOUBLE PRECISION PARAMETER                             -
CD                    1- A6 WORD IS SINGLE WORD                          -
CD                    2- A6 WORD IS DOUBLE PRECISION WORD                -
C                                                                        -
C-------------------------------------------------------------------------

C-------------------------------------------------------------------------
CR          FILE CONTROL   (1D RECORD)                                   -
C                                                                        -
CL    NGROUP,NISOD,NFAM,IDUM                                             -
C                                                                        -
CW    4=NUMBER OF WORDS                                                  -
C                                                                        -
CB    FORMAT(4H 1D ,4I6)                                                 -
C                                                                        -
CD    NGROUP        NUMBER OF NEUTRON ENERGY GROUPS IN SET               -
CD    NISOD         NUMBER OF ISOTOPES IN DELAYED NEUTRON SET            -
CD    NFAM          NUMBER OF DELAYED NEUTRON FAMILIES IN SET            -
CD    IDUM          DUMMY TO MAKE UP FOUR-WORD RECORD                    -
C                                                                        -
C-------------------------------------------------------------------------

C-------------------------------------------------------------------------
```

```
CR          FILE DATA, DECAY CONSTANTS, AND EMISSION SPECTRA          -
C                          (2D RECORD)                                -
C                                                                     -
CL    (HABSID(I),I=1,NISOD),(FLAM(N),N=1,FAM),((CHID(J,N),J=1,NGROUP), -
CL    N=1,NFAM),(EMAX(J),J=1,NGROUP),EMIN,(NKFAM(I),I=1,NISOD),      -
CL    (LOCA(I),I=1,NISOD)                                            -
C                                                                     -
CW    (2+MULT)*NISOD+(NGROUP+1)*(NFAM+1)=NUMBER OF WORDS             -
C                                                                     -
CB    FORMAT(4H 2D ,9(1X,A6))          HABSID                       -
CB   1(10(1X,A6)))                                                   -
CB    FORMAT(6E12.5)                   FLAM,CHID,EMAX,EMIN          -
CB    FORMAT(12I6)                     NKFAM,LOCA                   -
C                                                                     -
CD    HABSID(I)     HOLLERITH ABSOLUTE ISOTOPE LABEL FOR ISOTOPE I (A6)-
CD    FLAM(N)       DELAYED NEUTRON PRECURSOR DECAY CONSTANT         -
CD                     FOR FAMILY N                                  -
CD    CHID(J,N)     FRACTION OF DELAYED NEUTRONS EMITTED INTO NEUTRON -
CD                     ENERGY GROUP J FROM PRECURSOR FAMILY N        -
CD    EMAX(J)       MAXIMUM ENERGY BOUND OF GROUP J (EV)             -
CD    EMIN          MINIMUM ENERGY BOUND OF SET (EV)                 -
CD    NKFAM(I)      NUMBER OF FAMILIES TO WHICH FISSION IN ISOTOPE I  -
CD                     CONTRIBUTES DELAYED NEUTRON PRECURSORS        -
CD    LOCA(I)       NUMBER OF RECORDS TO BE SKIPPED TO READ DATA FOR  -
CD                     ISOTOPE I, LOCA(1)=0                          -
C                                                                     -
C---------------------------------------------------------------------

C---------------------------------------------------------------------
CR          DELAYED NEUTRON PRECURSOR YIELD DATA   (3D RECORD)        -
C                                                                     -
CL    (SNUDEL(J,K),J=1,NGROUP),K=1,NKFAMI),(NUMFAM(K),K=1,NKFAMI)    -
C                                                                     -
CC    NKFAMI=NKFAM(I)                                                -
C                                                                     -
CW    (NGROUP+1)*NKFAMI=NUMBER OF WORDS                              -
C                                                                     -
CB    FORMAT(4H 3D ,5E12.5/(6E12.5))  SNUDEL                         -
CB    FORMAT(12I6)                    NUMFAM                         -
C                                                                     -
CD    SNUDEL(J,K)   NUMBER OF DELAYED NEUTRON PRECURSORS PRODUCED IN  -
CD                     FAMILY NUMBER NUMFAM(K) PER FISSION IN        -
CD                     GROUP J                                       -
CD    NUMFAM(K)     FAMILY NUMBER OF THE K-TH YIELD VECTOR IN        -
CD                     ARRAY SNUDEL(J,K)                             -
C                                                                     -
C---------------------------------------------------------------------
```

This file is used to communicate delayed-neutron data to reactor kinetics codes. The ENDF files give a total delayed-neutron yield $\bar{\nu}_d$ in the section labeled MF1,MT455.

GROUPR averages this yield for each neutron group g using

$$\bar{\nu}_g^D = \frac{\int_g \nu_d(E)\sigma_f(E)\phi(E)\,dE}{\int_g \sigma_f(E)\phi(E)\,dE} . \tag{14}$$

This same section of the ENDF tape contains the decay constants for six delayed-neutron time groups. These numbers are passed on to the GROUPR routine that averages the delayed-neutron spectra, and they end up in the MF5 record on the GENDF tape. The ENDF evaluations give the delayed-neutron spectra for the six time groups in MF5,MT455. These spectra are not separately normalized. Rather, the sum over all six time groups is normalized, but the integral of any one of the spectra gives the "delayed fraction" for that time group. GROUPR simply averages these six spectra using the specified neutron group structure. The resulting group spectra (and the decay constants from File 1) are written onto the GENDF tape.

Returning to the DLAYXS format description, the parameters in the "File Identification" record are obtained from the user's input, just as for ISOTXS and BRKOXS. The parameter NGROUP is also a user input quantity. NISOD is determined after the entire GENDF tape has been searched for isotopes that are on the user's list of NISO materials and that have delayed-neutron data. The NFAM parameter needs some additional explanation. A "family" for the DLAYXS file is actually an index that selects one particular spectrum from the CHID array. It could correspond to an actual delayed-neutron precursor isotope left after a fission event. In such a case, there would be many "families" corresponding to the many possible fission fragment isotopes. The SNUDEL yields would be analogous to fission product yields. The ENDF evaluations take a more macroscopic approach. Spectra are chosen to include all the emissions from a group of delayed-neutron precursors for a particular fissioning nucleus with similar decay constants. In this representation, the DLAYXS "family" corresponds to a particular decay constant and spectrum for a particular target isotope. Therefore, the number of families is simply six times the number of isotopes, or NFAM=6*NISOD.

In the "File Data, Decay Constants, and Emission Spectra" record, the Hollerith isotope names HABSID are obtained from the names in the user's input. The FLAM values come directly from the decay constant values originally extracted from File 1 of the ENDF tape. The family index for isotope ISO and time group I is simply computed as 6*(ISO-1)+I. The CHID array is loaded from the MF5,MT455 section on the GENDF tape using the family index and the group index. As

usual, the order of groups has to be changed from the GROUPR convention with increasing-energy order to the CCCC convention with decreasing-energy order. The group structure itself is obtained from the GENDF header record and stored into EMAX and EMIN in the conventional order. The value of NKFAM is simply 6 for every isotope. The LOCA values are also easy to compute; they are just ISO minus 1.

The delayed-neutron yields versus incident neutron energy group and family index are given in the "Delayed Neutron Precursor Yield Data" record. As mentioned above, the total yield is given in MF3 of the GENDF tape, and the delayed fractions can be computed by summing the spectrum for each time group over the energy group index. The array SNUDEL in this DLAYXS record contains the product of these values. Note that there is one of these yield records for each delayed-neutron isotope, and each record contains six families. The NKFAM vector is used to establish the correspondence between these families and the entire list of families used in the file data record. As an example, NUMFAM(1) would contain 1, 2, 3, 4, 5, 6; NUMFAM(2) would contain 7, 8, 9, 10, 11, 12, and so on.

F. Coding Details

The main subroutine of CCCCR starts by initializing some variables and the storage system. Note that the CCCC standards specify that Hollerith variables contain 6 characters. This means that they must be declared to be double-precision variables on short-word machines, which normally allow up to 4 characters per word. Long-word machines usually allow from 8 to 10 characters per word. This requirement is handled by using the letter "h" to start every Hollerith word and by activating the statement "IMPLICIT REAL*8 (H)" on short-word machines. For use in calculating data pointers, the value of MULT is set to 2 on short-word machines. In order to make sure that dynamic arrays allocated by RESERV will have correct word alignment for short-word machines, a 2-word array called DUM is allocated. If its pointer LZERO is even, one of the words is released. This assures that the next array assigned by RESERV will have an odd pointer.

Next, the unit numbers for input and output are read. The signs given for CCCC units are ignored; they are opened as binary files. The subroutine continues by reading input cards 2 through 5. All integers are converted from the Z array of FREE using the FORTRAN nearest-integer function NINT to avoid round-off problems that occur in some machines. All Hollerith variables are moved into variables whose names start with "h" by making use of the equivalence between

the arrays Z and HZ. The MULT parameter is used to calculate the addresses in the equivalenced array.

At this point, CCCCR branches to different subroutines for each of the three interface file types that have been requested (see CISOTX, CBRKXS, and CDLYXS). Each of these routines is paired with a print routine that is called if the IPRINT flag has been set to 1 (see PISOTX, PBRKXS, and PDLYXS). When the last of these routines returns, CCCCR closes its active I/O units, writes a report, and terminates.

1. ISOTXS File Preparation. The CISOTX routine starts by calling RUINIS to read in the portion of the user's input specific to the ISOTXS file. It then calls ISXDAT to read through the input GENDF file and extract the ISOTXS data.

Subroutine ISXDAT reads the GENDF header record for the first material and extracts the group boundary energies. If the group structure found does not match the one requested in the user's input, an error message is issued. Otherwise, the order of the groups is changed from the GROUPR/ENDF convention of increasing energy to the CCCC convention of decreasing energy (which is the normal ordering for application programs). The routine then begins a loop over all the materials (isotopes) requested (see DO 300 I=1,NISO). It searches through the input GENDF tape for the first material with the requested MAT number (that is, the first temperature for the MAT) and copies it to a scratch file. Note that materials are written in the order requested by the user's input material list, not in the order that they are found on the GENDF tape.

This scratch tape is scanned for the ISOTXS principal cross section data in PRINXS. This process is fairly simple for most of the ENDF File 3 cross sections. Each MT number in File 3 is compared to the list of desired principal cross sections in NSTX; if a match is found, the corresponding element of IFLAG is set, and the cross section table is copied into memory using a pointer from the IPTR array. The (n,2n) reaction is a special case. In addition to MT16, the routine watches for the partial (n,2n) representation used in the ENDF/B-IV and -V evaluations for ^{9}Be (namely, MT6, 7, 8, and 9), and adds them into the appropriate memory locations. Fission is obtained from MT18, but if the partial fission reactions are present (MT19, 20, 21, 38), the flag MT19 is set. Average neutron velocities are obtained from the special GROUPR-produced section labeled by MT259. These are inverse velocities in s/m, and PRINXS inverts them and multiplies by 100 to get velocities in cm/s. In addition to the normal flux-weighted total cross section, the routine also stores the current-weighted, or P_1, cross section for use in calculating the transport cross section (see below). Finally, File 3 may also contain MT455,

the delayed-neutron $\bar{\nu}$ parameter. The GENDF record for this parameter also contains σ_f and the flux for each group. The delayed-neutron production rate $\bar{\nu}_d\sigma_f$ is stored for use in calculating the total $\bar{\nu}$ parameter, and $\bar{\nu}_d\sigma_f\phi$ is added into the DNORM array for later use in normalizing the total fission spectrum, χ. Note that there are two options for the flux used in this normalization: it can be the default library flux found on each record of the GENDF file, or it can be a flux spectrum SPEC provided by the user.

While reading through the scratch tape, PRINXS also watches for the elastic scattering matrix (MF6, MT2). The total P_1 elastic cross section is calculated by summing over all secondary neutron groups, and the result is subtracted from the current-weighted total cross section to obtain the transport cross section.

The processing of the fission matrix (MF6, MT18) and the delayed-fission spectra (MF5, MT455) depend on the ICHIX option. The $\bar{\nu}$ parameter SNUTOT is always calculated for fissionable isotopes. CHISO is only calculated if the vector representation was requested; in addition, it can be calculated using the flux from GENDF or using the input flux in SPECT. Starting with NJOY 91.0, the fission matrix is represented by a real group-to-group matrix at high energies, and by a single spectrum (with IG=0) and an associated production cross section (with IG2LO=0) at low energies. This representation can lead to great savings for libraries with many low-energy groups. The contributions to SNUTOT from the low-energy range are easily obtained by adding in the production cross section. At high energies, the sum of the group-to-group matrix over all secondary-energy groups is added in. After MF3,MT455 and MF6,MT18 have been processed, it is only necessary to divide the total production rate in memory by the fission cross section to obtain the required total $\bar{\nu}$ values (see statement number 242).

The prompt part of the fission χ vector or matrix is obtained from MF6,MT18, or if the MT19 flag was set while reading File 3, from MT19, 20, 21, and 38. The code starts at statement number 207, and it is fairly complicated because it has to cope with the following options:

1. χ vector versus χ matrix,
2. default weighting for the vector versus input weighting,
3. square matrix versus collapsed rectangular matrix, and
4. constant and matrix parts of MF6 GENDF record.

Note that separate normalization sums are accumulated in CNORM(ISPEC) for the chi vector or for each column of the matrix while the χ elements are being stored. The delayed contributions to the fission χ vector or matrix are taken from

MF5,MT455 (see statement number 230). The spectra for all of the six time groups are added and multiplied by the delayed neutron production rate DNORM (which can be incident energy dependent through the ISPEC index if a matrix is being constructed). The delayed contributions to the normalization sum for the χ vector or for each column of the χ matrix are added into CNORM during this step. After all the components of the fission spectrum have been read from the GENDF file, the χ vector or the columns of the χ matrix are normalized in the DO 249 loop.

When all sections of the scratch tape have been processed, PRINXS goes through the principal cross section block in memory removing any parts that have zero cross sections, and it writes the resulting block on scratch file NSCRT2. Finally, it writes the chi matrix data, if any, to the same scratch file and returns to ISXDAT.

Subroutine ISOMTX is then called to read through the scratch tape and process the group-to-group matrices into CCCC format. In order to allow large matrices to be processed on machines with limited memory, one or more passes can be made through the scratch tape (see DO 400 NJ=1,NPASS). The number of passes used depends upon the amount of space available in scratch array B and on the subblocking option requested by the user. The four options supported by CCCCR were discussed in connection with the ISOTXS format description. Only one pass is required for the first option. In the other cases, the length of one subblock is divided into the length of the scratch array to determine the number of subblocks that can be accumulated on one pass (NREC) and the number of passes required (NPASS).

Four matrix reactions are extracted from the scratch tape (see DO 500 I=1,4). The total matrix is obtained by adding all matrices found on the GENDF tape except the fission matrices. The elastic matrix is obtained from MT2 only. The (n,2n) matrix is normally obtained from MT16, but the sum of MT6, 7, 8, 9, 46, 47, 48, and 49 will be used for the old ^{9}Be representation, if found. The inelastic matrix is the sum of everything else; that is, it includes the normal inelastic reactions MT51–91, and it also includes other neutron-producing reactions like (n,3n), (n,n′p), and (n,n′α). Each GENDF section found is passed to SHUFFL, which rearranges the input data into the CCCC order in scratch array B, and then to WRTMTX, which repacks the data into banded form and writes the results to NSCRT2.

Subroutine SHUFFL reads each of the group-to-group cross sections for a given reaction (MT) and calculates a location for each cross section in the scratch array (NOLOC). Different formulas are used for the location of the two allowed values of IFOPT. For IFOPT=1, the data are stored with incident group index JG1 changing fastest, then Legendre index IL, and with final group index JG2 changing slowest.

Note that the indexing scheme only stores the triangle of the matrix corresponding to ingroup scattering and downscatter (that is, JG2≥JG1). In addition, the first group in a subblocking range NG2Z is used. This means that values of JG2 less than NG2Z will result in storage locations less than IRSIZE, and they will not be stored (see statement 200). Groups with JG2 above the subblock might be stored in memory, or they might end up above the upper limit of the memory area and be suppressed by statement number 200. Similarly, the data for IFOPT=2 are stored with JG1 as the fastest varying index, then JG2, and finally with IL as the slowest varying index. Here also, the storage pointer NOLOC is calculated so that only elements with JG2≥JG1 are stored, and elements that are outside of a given subblock are removed if they fall outside the bounds of the memory array.

Subroutine WRTMTX searches through the memory area loaded in SHUFFL to find the bands of group-to-group elements that will be written on the final ISOTXS file. The calculation of locations in the memory array depends on the blocking and subblocking options selected. In general, the routine calculates locations NOLOCA and tests the cross section found there against the value EPS=1.E-10. The highest index that violates this test for a given final energy group determines the bandwidth for that group. (Remember that the inscattering group is always kept, even if it has zero cross section.) Once the band limits have been found, the subroutine loops back through the memory area and squeezes out the locations that are not included in the bands. At this point, all the elements of the "Scattering Sub-Block" are in their final locations, and the record is written out to scratch tape NSCRT2.

When ISOMTX returns to ISXDAT, the "Isotope Control and Group Independent Data" record is completed by filling in the values of IDSCT and IJJ, and the record is written to scratch tape NSCRT3. The LOCA values for the "File Date" record are also calculated at this point. They remain in memory at L19. Once the "DO 300" loop over the requested isotopes has finished, the ISXDAT routine returns.

Back in CISOTX, all the data needed for the ISOTXS file are now in memory or on one of the CCCC-style scratch files NSCRT2 or NSCRT3. The routine simply steps through the ISOTXS records and either constructs them or copies them from a scratch file. When the file has been written, the routine returns to CCCCR, which checks the print flag, and calls \cwordPISOTX—, if requested.

PISOTX is a fairly simple routine. It reads through the ISOTXS file produced by CCCCR and prepares an interpreted listing of the data.

2. BRKOXS File Preparation. Subroutine CBRKXS is used to prepare the BRKOXS file, if requested. Following the same pattern as CISOTX, it calls RUINBR to read the user's input, then it calls BRKDAT to extract the desired data from the GENDF file, and finally it writes the output BRKOXS file using data stored in memory and on a scratch tape by BRKDAT.

The storage locations used by BRKDAT are outlined in the comment cards at the beginning of the subroutine. Note that L1 is the pointer to dynamic array FLID reserved in CBRKXS. Pointers down through L8 are not really independent; they are defined relative to L1 and are used to load data from the general CCCCR input plus the σ_0 and T counts from the BRKOXS input section. Next, dynamic arrays are reserved using pointers L9 through L17 for data arrays of known length. The dynamic array IRINP is reserved using all of the remaining memory (NWD=-1). The subroutine now opens the input GENDF file and reads in the header record. After checking for possibly incompatible group structures, it reverses the order of the group bounds and loads them into the EMAX and EMIN arrays. The number of words required for reading the rest of the GENDF records is computed as N19, and the rest of the IRINP array is released. The dynamic array INFX with pointer L19 is reserved to hold 13 infinitely dilute cross section vectors that will either be needed during the processing (vectors 1 through 7) or will be written into the "Cross Sections" record. These 13 quantities are

1. P_0 total cross section,
2. capture cross section,
3. fission cross section,
4. P_1 total - P_1 elastic,
5. elastic cross section,
6. P_1 total cross section,
7. P_1 flux,
8. XSPO,
9. inelastic cross section,
10. elastic cross section,
11. average elastic scattering cosine $\bar{\mu}$,
12. elastic downscatter to adjacent group, and
13. average elastic lethargy increment ξ.

Now the entire remaining memory is reserved for the array SCRT with pointer L20.

The next step is to determine whether it is necessary to subblock the self-shielding factor record. The amount of space available in dynamic array SCRT is compared with the maximum amount of memory required for the f-factors. If there is not enough space, NSBLK is increased to NREACT; otherwise, it is left at 1. The value of NSBLK is stored into the IBLK field of the BRKOXS "File Control" record.

The isotope loop is complicated because CCCCR allows the materials on the GENDF tape to be in any order, but it arranges things so that the materials on the BRKOXS files are in the order that the materials are named in the user's input. The main loop is controlled by DO 370 I=1,NISO. For each pass, the input GENDF tape is rewound and searched for isotope I (see DO 170 J=1,NISO). Once a desired material has been found, the σ_0 list in the header record is examined. Either the first abs(NZI) values are extracted (for NZI negative), or the particular values that occur on the input ASIG list are extracted (for NZI positive). In either case, NZJ is the number of σ_0 values found, ISIG contains the pointers to the values selected, and CSIG contains the actual values of σ_0.

GENDF tapes contain one of more temperatures for each material recorded as consecutive MATs. Starting at statement number 190, BRKDAT reads through all the materials with the current MAT number selecting the desired temperatures and copying them to a scratch file NSCRT1. The procedure used to select temperatures is similar to the one used to select σ_0 values. If NTI is negative, the first abs(NTI) temperatures are extracted. If NTI is positive, only temperatures on the list in ATEM are extracted. In either case, NTJ is the number of temperatures found for this material, ITEM contains indices to those temperatures, and CTEM contains the actual temperature values. When all the desired temperatures have been copied (see statement number 260), the space used for GENDF input is released, and a new block is reserved with a size appropriate for reading in the desired data. In addition, INFX and SCRT are reserved to hold the BRKOXS cross sections and f-factors, respectively.

In the loop beginning at DO 340 NSB=1,NSBLK, a pass through the scratch tape is made for each subblock (1 or NREACT). Each section of the GENDF format is located, and either XSPROC (for MF3) or MXPROC (for MF6) is called to process the data in the section.

Subroutine XSPROC is called once for each reaction in File 3 of the GENDF tape. It reads in all the data, checks which reaction is present, and stores the data in one of the 13 cross section locations (see L19), or in one of the f-factor locations

(see L20). If the denominator for the f-factor calculation is zero, the division is skipped, and a warning message is issued.

Subroutine MXPROC is called once for each reaction in File 6 of the GENDF tape. It loops through statement number 120 to read all of the incident groups, and then it uses different sections of coding to fill in or fix up the rest of the 13 elements in the infinitely dilute cross section block at L19. These quantities include the elastic and inelastic cross sections, the transport cross section, and the removal cross section. This routine also computes the elastic removal self-shielding factors from MF6,MT2, and stores the results in the f-factor area at L20.

Back in BRKDAT, the self-shielded transport cross sections are converted into f-factors. A check is then made to see if the transport values were properly computed. This requires that a self-shielded elastic matrix was available for all the higher temperatures. A warning message is issued if the required data were not present. The last 6 of the 13 vectors stored starting at L19 are the data needed for the "Cross Sections" record; that block is written to scratch file NSCRT2. Next, the accumulated f-factor data at L20 are written to NSCRT2, and the NTABP and NTABT arrays are stored at L14 and L15, respectively.

The last step inside the isotope loop is to call THNWRT to thin out the f-factor data and write the results onto scratch file NSCRT3. It starts by reading the cross section data from NSCRT2 into memory. Then it reads the f-factor data into memory and repacks it to take account of the group range for interesting self-shielding factors, namely, JBFL to JBFH. When this is finished, it writes the f-factor array out to NSRT3, and then it writes the unchanged cross section block out to NSCRT3. Note that these two records are now in the correct order to be copied to the BRKOXS file.

Subroutine THNWRT now returns control to BRKDAT. When the "DO 370" loop has finished, the routine cleans up the "File Data" information in memory and returns to the main BRKOXS routine. At this point, all the information required to construct the output file is present in memory or on scratch tape NSCRT3. The routine steps through the records of the BRKOXS format constructing them or copying them from the CCCC-style scratch file. It then returns to CCCCR, which checks to see whether PBRKXS should be called.

Subroutine PBRKXS is a fairly simple routine. It reads through the BRKOXS file produced by CCCCR and prepares an interpreted listing of the data.

3. DLAYXS File Preparation. Subroutine CDLYXS is used to prepare the DLAYXS file, if requested. It starts by calling DLDATA to extract the delayed-neutron data from the input GENDF file. Since all the data are stored into memory, CDLYXS continues by simply writing out the required "File Identification," "File Control," "File Data, Decay Constants, and Emission Spectra," and "Delayed Neutron Precursor Yield Data" records. If no delayed-neutron data are found, CDLYXS issues a fatal error message.

Subroutine DLDATA starts by reserving space for the dynamic arrays used to store the delayed-neutron data; the purpose for each of these arrays is summarized in the comment block at the beginning of the subroutine. ENDF delayed-neutron files are based on the traditional 6 time groups; therefore, CCCCR makes each time group for each isotope correspond to one DLAYXS "delayed-neutron family" (see NFAM=6*NISOD). Next, DLDATA starts reading through the materials on the GENDF tape and looking at each material requested in the user's input (the material loop goes through statement 110). For the first material, it reads the header record, checks that the group structure is compatible with the user's input value NGROUP, and stores the structure in EMAX and EMIN in the conventional decreasing-energy order. For all materials, it watches for sections with MF3,MT455 or MF5,MT455.

When MF3/MT455 is found (see statement 310), DLDATA stores the total delayed-neutron yield from GROUPR in the SNUDEL array (pointer L8) using an offset computed from the time group index (which varies from 1 to 6), the group index, and the isotope index. (The group index goes through statement 130). For the present, the same value is stored for every time group.

When MF5/MT455 is found (see statement 410), the rest of the data for this isotope are stored into memory. The HABSID field is obtained from the isotope name in the user's input. The LOCA field is easy to calculate from the isotope index. The decay constants for each of the time groups are copied into FLAM from the GENDF record. The number of families for this isotope is simply 6; this number is loaded into the location corresponding to NKFAM. Finally, the actual delayed-neutron spectra for each time group are loaded into the CHID area using energy group and family number as indexes. At this point, the spectrum for each time group is summed over group index to determine the delayed fraction for that time group [see FRACT(I)]. Then this fraction is used to change the total delayed-neutron yield for each time group in the SNUDEL area into the fractional delayed-neutron yield for that time group (family).

When all the isotopes containing delayed-neutron data have been processed, DLDATA returns to CDLYXS. When CDLYXS returns to the main subroutine of CCCCR

after writing the DLAYXS file, the print routine PDLYXS may be called. This is a fairly simple routine that reads through a file in DLAYXS format and prepares an interpreted listing.

G. User Input

The user input instructions copied from the comment cards at the beginning of the CCCCR source code are given below. It is always a good idea to check the comments cards in the current version of the code in case there have been any changes.

```
*---INPUT SPECIFICATIONS (FREE FORMAT)-------------------------*
*                                                              *
*-CCCCR-                                                       *
* CARD 1 UNITS                                                 *
*    NIN       INPUT UNIT FOR DATA FROM GROUPR                 *
*    NISOT     OUTPUT UNIT FOR ISOTXS (0 IF ISOTXS NOT WANTED) *
*    NBRKS     OUTPUT UNIT FOR BRKOXS (0 IF BRKOXS NOT WANTED) *
*    NDLAY     OUTPUT UNIT FOR DLAYXS (0 IF DLAYXS NOT WANTED) *
* CARD 2 IDENTIFICATION                                        *
*    LPRINT    PRINT FLAG (0/1=NOT PRINT/PRINTED)              *
*    IVERS     FILE VERSION NUMBER (DEFAULT=0)                 *
*    HUSE      USER IDENTIFICATION (12 CHARACTERS)             *
*              DELIMITED BY *, ENDED BY /.                     *
*              (DEFAULT=BLANK)                                 *
* CARD 3                                                       *
*    HSETID    HOLLERITH IDENTIFICATION OF SET (12 CHARACTERS) *
*              DELIMITED BY *, ENDED BY /.                     *
*              (DEFAULT=BLANK)                                 *
* CARD 4 FILE CONTROL                                          *
*    NGROUP    NUMBER OF NEUTRON ENERGY GROUPS                 *
*    NGGRUP    NUMBER OF GAMMA ENERGY GROUPS                   *
*    NISO      NUMBER OF ISOTOPES DESIRE                       *
*    MAXORD    MAXIMUM LEGENDRE ORDER                          *
*    IFOPT     MATRIX BLOCKING OPTION (1/2=BLOCKING BY         *
*                                 REACTION/LEGENDRE ORDER)     *
* CARD 5 ISOTOPE PARAMETERS (ONE CARD PER ISOTOPE)             *
*       (FIRST FOUR WORDS ARE HOLLERITH, UP TO SIX CHARACTERS  *
*       EACH, DELIMITED BY *)                                  *
*    HISNM     HOLLERITH ISOTOPE LABEL                         *
*    HABSID    HOLLERITH ABSOLUTE ISOTOPE LABEL                *
*    HIDENT    IDENTIFIER OF DATA SOURCE LIBRARY (ENDF/B)      *
*    HMAT      ISOTOPE IDENTIFICATION                          *
*    IMAT      NUMERICAL ISOTOPE IDENTIFIER (ENDF/B MAT NUMBER)*
*    XSPO      AVERAGE POTENTIAL SCATTERING CROSS SECT. (BRKOXS)*
*                                                              *
*-CISOTX- (ONLY IF NISOT.GT.0)                                 *
* CARD 1 FILE CONTROL                                          *
*    NSBLOK    SUBBLOCKING OPTION FOR SCATTERING MATRIX        *
*              (1 OR NGRUP SUB-BLOCKS ALLOWED)                 *
```

```
*    MAXUP     MAXIMUM NUMBER OF UPSCATTER GROUPS (ALWAYS ZERO)  *
*    MAXDN     MAXIMUM NUMBER OF DOWNSCATTER GROUPS              *
*    ICHIX     FISSION CHI REPRESENTATION                        *
*                   -1   VECTOR (USING GROUPR FLUX)              *
*                    0   NONE                                    *
*                   +1   VECTOR (USING INPUT FLUX)               *
*                 .GT.1  MATRIX                                  *
* CARD 2 CHI VECTOR CONTROL (ICHIX=1 ONLY)                       *
*    SPEC      NGROUP FLUX VALUES USED TO COLLAPSE THE GROUPR    *
*              FISSION MATRIX INTO A CHI VECTOR                  *
* CARD 3 CHI MATRIX CONTROL (ICHIX.GT.1 ONLY)                    *
*    SPEC      NGROUP VALUES OF SPEC(I)=K DEFINE THE RANGE OF    *
*              GROUPS I TO BE AVERAGED TO OBTAIN SPECTRUM K.     *
*              INDEX K RANGES FROM 1 TO ICHI.                    *
*              THE MODEL FLUX IS USED TO WEIGHT EACH GROUP I.    *
* CARD 4 ISOTOPE CONTROL (ONE CARD PER ISOTOPE)                  *
*    KBR       ISOTOPE CLASSIFICATION                            *
*    AMASS     GRAM ATOMIC WEIGHT                                *
*    EFISS     TOTAL THERMAL ENERGY/FISSION                      *
*    ECAPT     TOTAL THERMAL ENERGY/CAPTURE                      *
*    TEMP      ISOTOPE TEMPERATURE                               *
*    SIGPOT     AVERAGE EFFECTIVE POTENTIAL SCATTERING           *
*    ADENS     DENSITY OF ISOTOPE IN MIXTURE                     *
*                                                                *
*-CBRKXS- (ONLY IF NBRKS.GT.0)                                  *
* CARD 1 (2I6) FILE DATA                                         *
*    NTI       NUMBER OF TEMPERATURES DESIRED                    *
*              (-N MEANS ACCEPT FIRST N TEMPERATURES)            *
*    NZI       NUMBER OF SIGPO VALUES DESIRE                     *
*              (-N MEANS ACCEPT FIRST N DILUTION FACTORS)        *
* CARD 2 (NOT NEEDED IF NTI.LT.0)                                *
*    ATEM(NTI) VALUES OF DESIRED TEMPERATURES                    *
* CARD 3 (NOT NEEDED IF NZI.LT.0)                                *
*    ASIG(NZI) VALUES OF DESIRED SIGPO                           *
*                                                                *
*-CDLAYX-- NO INPUT REQUIRED                                    *
*                                                                *
******************************************************************
```

These instructions are divided into four parts. First, there is a general section that applies to all three interface files following -CCCCR-. Second, there is a section of special parameters for ISOTXS following -CISOTX-. Third, there is a section of special parameters for BRKOXS following -CBRKXS-. And fourth, there is a comment following -CDLYXS- noting that no special input is required for the DLAYXS file.

In the -CCCCR- section, Card 1 is used to read in the unit numbers for input and output. NIN must be a GENDF tape prepared using GROUPR, and it can have either binary (NIN>0) or BCD (NIN<0) mode. The units for the output ISOTXS, BRKOXS, and DLAYXS files are all binary, but either sign can be used on the unit

numbers. If any unit number is given as zero, the corresponding CCCC interface file will not be generated. On Card 2, the LPRINT=1 is used to request a full printout of all the CCCC files generated. The file version number IVERS can be used to distinguish between different libraries generated using NJOY. The user identification field HUSE can contain any desired 12-character string. An example of Card 2 might be

```
0  4  *T2 LANL NJOY*/  .
```

Card 3 contains a description of the library using up to 72 characters (12 standard CCCC words of 6 characters each). For example,

```
*LIB-IV 50-GROUP LMFBR LIBRARY FROM ENDF/B-IV (1976)*/  .
```

On Card 4, the values given for NGROUP must agree with the number of groups on the input GENDF tape or a fatal error message will be issued. NGGROUP is not used. NISO is the total number of materials or isotopes to be searched for on the GENDF tape. The value of MAXORD should be less than or equal to the maximum Legendre order used in the GROUPR run. The use of the matrix blocking parameter IFOPT was discussed in detail in connection with the description of the ISOTXS format above. A value of 1 has been normally used for Los Alamos libraries.

A line using the Card 5 format is given for each of the NISO isotopes or materials to be processed. The Hollerith isotope label and the Hollerith absolute isotope label have normally been set to the conventional isotope name at Los Alamos; for example, U235 or FENAT. The library name in HABSID can vary quite a lot now that many other libraries are available in ENDF format. The values of HMAT and IMAT will normally be derived from the MAT number characteristic of all libraries in ENDF format. Unfortunately, XSPO, the average potential scattering cross section, must be entered by hand. It can be obtained from MF2,MT151 on the ENDF file for the material by determining the scattering radius a from the AP field and computing $\sigma_p = 4\pi a^2$. The following fragment of an ENDF/B evaluation shows the vicinity of AP:

```
  ...
                                                          9228 0  0  572
 9.223500+4 2.330250+2          0          0          1  09228 2151  573
 9.223500+4 1.000000+0          0          1         11  09228 2151  574
 1.000000-5 1.100000+2          1          3          0  09228 2151  575
 3.500000+0 1.001760+0          0          0          1  29228 2151  576
 2.330250+2 0.000000+0          0          0       1416 2369228 2151  577
-9.997600+1 3.000000+0 3.771300-3 3.980000-2 1.861800-1 4.700200-19228 2151  578
  ...
```

The scattering radius is the second number on the third card in the section MF2,MT151. Using it as a gives $4\pi(1.00176)^2 = 12.611$ barns for the value of XSPO. An example of Card 5 follows:

```
U235 U235 ENDF6 *9228* 9228 12.611
```

Note that the Hollerith string "9228" had to be delimited by stars because it does not begin with a letter. The delimiters are optional for the other Hollerith variables.

The next block of input cards is specific to ISOTXS and only appears if ISOTXS processing was requested with a nonzero value for NISOT. The first parameter on Card 1 is NSBLOK, which was discussed in connection with matrix blocking and subblocking. NSBLOK and IFOPT work together to control how a large scattering matrix is broken up into smaller records on the ISOTXS interface file. The important factor is the maximum size of the binary records. They should be small enough to fit into a reasonable amount of memory in any application codes that use ISOTXS files, but they should be large enough to keep the number of I/O operations to a minimum. The maximum record size for each option is repeated below for the convenience of the reader:

1. *IFOPT=1 and NSBLOK=1*: $n_\ell \times n_g(n_g-1)/2$,
2. *IFOPT=1 and NSBLOK=NGROUP*: $n_\ell \times n_g$,
3. *IFOPT=2 and NSBLOK=1*: $n_g(n_g-1)/2.$,
4. *IFOPT=2 and NSBLOK=NGROUP*: n_g,

where n_ℓ is the number of Legendre orders and n_g is the number of groups. If the user specifies NSBLOK=1 and the resulting output record is too large for the available memory, NSBLOK will be changed to NGROUP automatically.

Card 1 of the ISOTXS section continues with MAXUP. This parameter is always zero for CCCCR; thermal upscatter matrices are not processed. The normal value of MAXDN is NGROUP, but it can be made smaller to reduce the size of the matrices. The cross section for any removed downscatter groups will be lumped into the last group in order to preserve the production cross section. The ICHIX is used to control the generation of fission χ vectors and matrices. The representations allowed were discussed above in connection with the description of the ISOTXS format (see Section XIV.C). The most commonly used option has been ICHIX=-1 because most application codes cannot handle fission χ matrices. The GROUPR flux is normally chosen to be characteristic of the class of problems a given library

is intended to treat; therefore, it is rarely necessary to supply an input weighting spectrum (see ICHIX=+1 and SPEC). The following scenario illustrates a case where this option might be useful. Assume that an 80-group library is made using the GROUPR fast reactor weight function (IWT=8), which contains a shape in the fission range typical of both fast reactors and fusion blankets plus a fusion peak at 14 MeV. This GENDF library could be used to generate two different ISOTXS libraries, one using the default flux and useful for fusion problems, and one using a spectrum SPEC from which the fusion peak has been removed. The latter would be better for fast reactor analysis because the χ vectors would not contain the component of high-energy fission from the 14 MeV range. Card 2 is used to input the user's choice for SPEC.

CCCCR can also produce fission χ matrices for codes that can use them. These matrices can be rectangular to take advantage of the fact that the $\chi_{g\to g'}$ function is basically independent of g at low energies (large values of g). Taking the GROUPR 30-group structure as an example, if the energy at which significant incident-energy dependence begins is taken to be about 100 keV, then groups 16 through 30 will have identical χ vectors. The value of ICHIX should be set to 16, and the SPEC vector of Card 3 should be set to

```
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 15R16/  .
```

The resulting χ matrix will be rectangular with 16×30 elements.

Card 4 completes the input specific to ISOTXS. The value of KBR can be set to reflect the normal use of this material in the applications that this library is intended to treat. The AMASS parameter has units of gram atomic weight. It can be calculated from the normal ENDF AWR parameter (the atomic weight ratio to the neutron) by multiplying by the gram atomic weight of the neutron. TEMP is normally 300 K for NJOY libraries. The same value can be used for SIGPOT and XSPO (see above). The ADENS parameter has no meaning for CCCCR; it can be set to zero to imply infinite dilution.

The choice of values for EFISS and ECAPT is more complicated. As discussed in Section XIV.C, EFISS is basically the total non-neutrino energy released by a fission reaction. It is available in eV as the pseudo Q value in MF3,MT18 (the energy release from fission is given in more detail in MF1,MT458). The following fragment of the ENDF/B-VI evaluation for ^{235}U shows how to find the Q value:

```
 ...
                                                                  9228 3  0 4413
9.223500+4 2.330250+2          0          0          0          09228 3 18 4414
1.937200+8 1.937200+8          0          0          1        3339228 3 18 4415
       333          2                                           9228 3 18 4416
1.000000-5 0.000000+0 7.712960+1 0.000000+0 2.250000+3 0.000000+09228 3 18 4417
2.250000+3 5.362770+0 2.300000+3 5.396710+0 2.500000+3 5.957280+09228 3 18 4418
 ...
```

The Q value is the second number on the second card of the section MF3,MT18. Converting to CCCC units gives

193.72×10^{6} eV $\times$ 1.602×10^{-13} watt-s/eV = $.31034\times10^{-10}$ watt-s/fission

The value for ECAPT is determined from the Q value for MF3,MT102, the radiative capture reaction. However, if the isotope remaining after capture has a relatively short half-life, the energy of the decays leading to the final stable daughter should be added onto the capture Q value. (The meaning of "stable" may vary from application to application). As an example, consider aluminum. The ^{28}Al capture product decays with a half-life of 2.24 minutes producing 9.31 MeV of β^- energy and a 1.779 MeV photon. Therefore, the actual value of ECAPT should be calculated as follows:

7.724 MeV	MT102 prompt Q value
9.310 MeV	β^- energy
1.779 MeV	delayed-γ energy
18.813 MeV	
$\times 1.602 \times 10^{-13}$	
$.3014 \times 10^{-11}$	in watt-s/capture

The next section of the input file is specific to BRKOXS. Card 1 enables the user to just accept all or part of the temperatures and sigma-zero values on the input GENDF tape. If the value of NTI is negative, the first abs(NTI) T values for each material will be used. If fewer values are available, only those will be used. If NTI is positive, input Card 2 will be read for a list of T values, and only data with temperatures on that list will be extracted from the GENDF tape. The parameter NZI and the list of σ_0 values on Card 3 work in the same way.

No additional input is required for DLAYXS files.

H. Error Messages

`ERROR IN ISXDAT***INCOMPATIBLE GROUP STRUCTURES`

The number of groups on the input GENDF tape must match the number of groups specified in the CCCCR input. Check whether the correct input file was mounted.

`ERROR IN ISOMTX***INPUT RECORD TOO LARGE`

There is not enough space in the scratch array `B` to read in the input records from the GENDF tape. The only solution is to increase the size of the main container array. See `COMMON/CSTORE/A(25000)` and `ISIZA=25000` in subroutine CCCCR.

`ERROR IN ISOMTX***OUTPUT RECORD TOO LARGE`

There is not enough space in the scratch array `B` for the output subblock record, even with `NSBLOK` changed to `NGROUP`. The only solution is to increase the size of the main container array. See `COMMON/CSTORE/A(25000)` and `ISIZA=25000` in subroutine CCCCR.

`ERROR IN SHUFFL***INPUT RECORD OVERRAN OUTPUT`

This shouldn't happen because of the checks in `ISOMTX`. It may reflect a problem in the input data record.

`ERROR IN PISOTX***INPUT RECORD TOO LARGE`

One of the binary records on the ISOTXS file is too large for the memory available to `PRINXS`. This should not occur because there was enough memory to create the record in the first place.

`ERROR IN BRKDAT***INCOMPATIBLE GROUP STRUCTURES`

The number of groups on the input GENDF tape must match the number of groups specified in the CCCCR input. Check whether the correct input file was mounted.

`MESSAGE FROM BRKDAT---ALL AVAILABLE MATS HAVE BEEN PROCESSED`

This message is issued when all the materials on the GENDF file have been processed, but one or more of the materials requested in the user's input were not found. Check the input material list, and check which materials were written onto the input GENDF file.

`MESSAGE FROM BRKDAT---NO TEMPERATURES FOR MAT=nnnn`

This means that none of the requested temperatures were found for this MAT. This makes it impossible to include the material in the BRKOXS file. The warning message is issued, and all references to this material are thinned out of the BRKOXS records.

`MESSAGE FROM BRKDAT---NEED ELASTIC MATRICES AT HIGHER TEMPERATURES`

The self-shielded transport cross section requires self-shielded P_1 elastic scattering matrices for accurate results. This means that MF6,MT2 should be available on the GENDF tape for all temperatures. If this scattering matrix is missing for the higher temperatures, this warning message is issued.

MESSAGE FROM XSPROC---INFINITE F-FACTOR mt jg jz temp

The calculation of an f-factor requires division by the infinitely dilute cross section. This message means that the divisor was zero for reaction *mt*, group *jg*, background cross section *jz*, and temperature *temp*. The division is skipped, and the numerator is used unchanged.

ERROR IN MXPROC***NOT ENOUGH STORAGE FOR INPUT

There is not enough space in the dynamic array RINP with pointer L18 for the data for one incident group of a scattering matrix. The only solution is to increase the size of the main container array. See COMMON/CSTORE/A(25000) and ISIZA=25000 in subroutine CCCCR.

ERROR IN PBRKXS***INPUT RECORD TOO LARGE

One of the binary records on the BRKOXS file is too large for the memory available to PBRKXS. This should not occur because there was enough memory to create the record in the first place.

MESSAGE FROM CDLYXS---NO DELAYED NEUTRON DATA FOUND

There was no delayed-neutron data found by DLDATA. Make sure that MF3,MT455 and MF5,MT455 were requested during the GROUPR run and that the delayed-neutron isotopes were included in the material list given in the CCCCR input.

ERROR IN DLDATA***INCOMPATIBLE GROUP STRUCTURES

The number of groups on the input GENDF tape must match the number of groups specified in the CCCCR input. Check whether the correct input file was mounted.

ERROR IN PDLYXS***INPUT RECORD TOO LARGE

One of the binary records on the DLAYXS file is too large for the memory available to PDLYXS. This should not occur because there was enough memory to create the record in the first place.

I. References

1. B. M. Carmichael, "Standard Interface Files and Procedures for Reactor Physics Codes, Version III," Los Alamos Scientific Laboratory report LA-5486-MS (February 1974).

2. R. Douglas O'Dell, "Standard Interface Files and Procedures for Reactor Physics Codes, Version IV," Los Alamos Scientific Laboratory report LA-6941-MS (September 1977).

3. C. R. Weisbin, P. D. Soran, R. E. MacFarlane, D. R. Harris, R. J. LaBauve, J. S. Hendricks, J. E. White, and R. B. Kidman, "MINX: A Multigroup Interpretation of Nuclear X-Sections from ENDF/B," Los Alamos Scientific Laboratory report LA-6486-MS (ENDF-237) (September 1976).

4. R. B. Kidman and R. E. MacFarlane, "LIB-IV: A Library of Group Constants for Nuclear Reactor Calculations," Los Alamos Scientific Laboratory report LA-6260-MS (March 1976).

5. R. E. MacFarlane and R. B. Kidman, "LINX and BINX: CCCC Utility Codes for the MINX Multigroup Processing Code," Los Alamos Scientific Laboratory report LA-6219-MS (February 1976).

6. "CINX: Collapsed Interpretation of Nuclear X Sections," Los Alamos Scientific Laboratory report LA-6287-MS (April 1976).

7. W. J. Davis, M. B. Yarbrough, and A. B. Bortz, "SPHINX: A One Dimensional Diffusion and Transport Nuclear Cross Section Processing Code," Westinghouse Advanced Reactors Division report WARD-XS-3045-17 (August 1977).

8. C. L. Cowan, B. A. Hutchins, and J. E. Turner, "TDOWN – A Code to Generate Composition and Spatially Dependent Cross Sections," General Electric report GEAP-13740 (August 1971).

9. K. L. Derstine, "DIF3D: A Code to Solve One-, Two-, and Three-Dimensional Finite-Difference Diffusion Theory Problems," Argonne National Laboratory report ANL-82-64 (April 1984).

10. R. Douglas O'Dell, Forrest W. Brinkley, Jr., Duane R. Marr, and Raymond E. Alcouffe, "Revised User's Manual for ONEDANT: A Code Package for One-Dimensional, Diffusion-Accelerated, Neutral-Particle Transport," Los Alamos National Laboratory report LA-9184-M, Rev. (December 1989).

11. R. J. LaBauve, T. R. England, D. C. George, R. E. MacFarlane, and W. B. Wilson, "DANDE—A Linked Code System for Core Neutronics/Depletion Analysis," Los Alamos National Laboratory report LA-10412-MS (June 1985).

12. R. E. MacFarlane and D. C. George, "UPD: A Portable Version-Control Program," Los Alamos National Laboratory report LA-12057-MS (April 1991).

13. I. I. Bondarenko, Ed., *Group Constants for Nuclear Reactor Calculation* (Consultants Bureau, New York, 1964).

14. R. W. Hardie and W. W. Little, Jr., "1DX, A One-Dimensional Diffusion Code for Generating Effective Nuclear Cross Sections," Battelle Northwest Laboratories report BNWL-954 (1969).

XV. MATXSR

The MATXS material cross section format is a generalized CCCC-type interface format for neutron, photon, and charged-particle data, including cross sections, group-to-group matrices, temperature variations, self-shielding, and time-dependence. The CCCC standards are discussed in more detail in the CCCCR chapter of this manual and in the CCCC-III and CCCC-IV reports.[1, 2] MATXS libraries can be used with the TRANSX code[3, 4] to produce effective cross sections for a wide variety of application codes. This chapter describes version 91.0 of the MATXSR module. Caution: the MATXS format described here is significantly different from the previous version. Existing codes that use the MATXS format will not be compatible with libraries produced using NJOY 91.0.

A. Background

Even the very best nuclear cross section processing code would be useless if it were unable to deliver its products to users. This is the role of the interface file. There have been interface files since the beginning of calculational neutronics; examples include the DTF format (see the DTFR section of this manual) that was devised for the early discrete-ordinates transport code DTF-IV,[5] and the CCCC ISOTXS format[2] (see the CCCCR section of this report). Both of these interface formats are still in use today, but both of them have problems and show their age. Some of these problems result from the increase in the capabilities of computer systems (capabilities that allow us to consider much more complex problems), some arise from the many new kinds of nuclear systems that are being studied today, and some come from 20/20 hindsight, which makes it easy to see the design flaws in earlier formats.

Based on the problems seen with existing interface files, an ideal interface file should be

extendable, so that new cross section types, new incident or secondary particle types, or new energy ranges are easy to add without changing the basic format;

comprehensive, in order to be able to handle as many of the kinds of data produced by the processing code as possible (results should not be lost just because there are no places for them);

generalizable, to allow common methods to be used for similar kinds of data (for example, nn matrices and $\gamma\gamma$ matrices) in order to transfer the experience gained in one field to another, and in order to simplify coding by allowing components to be reused;

self-contained, because it should not be necessary to provide additional information that is not in the file in order to use or interpret the file;

compact, because nuclear data often have many zeros or very small numbers in tables (for example, threshold reactions, scattering matrices), and these zeros must be removed effectively for economic storage and fast transfer of libraries; and

efficient, thus implying that binary mode should be used, that the records have a well-defined maximum size, and that there is a minimum number of records to reduce the number of I/O operations.

Comparing the DTF format to these principles gives the following results: it is fairly *extendable* because it has no fixed particles, energy limits, or reaction types; it is not very *comprehensive* because it can only transmit the total scattering matrix; it is fairly *generalizable* because of the lack of fixed types; it is not at all *self-contained* in that it requires outside definitions like table length, position of the total, group structure, and identity of edit cross sections; it is not very *compact* because most zeros must be given explicitly in the tables; and it is not very *efficient* because it uses coded card-image records.

Similarly, studying the ISOTXS format gives the following results: it is not *extendable* because it works for neutrons only and allows only very limited types of reactions to be included; it is not *comprehensive* because it works for neutrons only and allows only very limited types of reactions to be included; it is not *generalizable* because of its specialization to fast-reactor problems (as proof, note that the CCCC files for γ cross sections use completely different formats); it is reasonably *self-contained* because all the parameters are internal, the group structure is given, and all names needed for labeling an interpreted listing are well determined; it is fairly *compact* because many zeros are removed (but too many still remain); it is fairly *efficient* because it uses binary mode, but record sizes are poorly predictable and very nonuniform, thereby needlessly increasing the size of application codes and the number of I/O operations.

B. The MATXS Format

Following these general principles, the MATXS material cross section file was designed to extend and generalize the existing interface formats while still using the CCCC approach for efficiency and familiarity (see the CCCCR section of this report for more details). The first design principle was that all information would be identified using lists of Hollerith names. As an example, if the list of reactions included in the file contains entries such as `NF`, `NG`, and `N2N`, it is trivial to add additional reactions such as `KERMA` or `DPA`. This approach is much more extendable than the fixed set of reaction flags used in ISOTXS. The second design principle was that the file would be designed to hold sets of vectors and rectangular matrices

Table 1: Standard Particle Names

Name	Particle
N	neutron
G	gamma
P	proton
D	deuteron
T	triton
H	^{3}He nucleus
A	alpha
B	beta
R	residual or recoil (heavier than α)

and that the same format would be used regardless of the contents of the vectors and matrices. As a consequence, once a code can handle $n \rightarrow n$, it can also handle $\gamma \rightarrow \gamma$; once a code can handle $n \rightarrow \gamma$, it can also handle $\gamma \rightarrow n$, $n \rightarrow \beta$, or even $d \rightarrow p$. This approach is an example of generalization. Each material is now divided into data types identified by input and output particle. As an example, $n \rightarrow \gamma$ is a data type characterized by input particle equals neutron, and output particle equals photon. The matrices for this data type contain cross sections for producing photons in photon group γ due to reactions of neutrons in group n. The vectors, if any, would contain photon production cross sections versus neutron group. The use of completely general data types helps make the format comprehensive.

The names for materials are written in the forms `U235`, `FE56`, `TINAT`, or `H2A`. Note that "`NAT`" is used explicitly for elements; names like `BE` or `C` should be avoided. Suffixes "`A`", "`B`", or "`C` are used to label different versions of a material in a library. In order to keep names to six characters, isomers should be identified by incrementing the "thousands" digit in the atomic number field; for example, `NB193A` would be the second version of the first isomer of ^{93}Nb. The standard names for MATXS particles are given in Table 1.

The standard names for the data types (`HTYPE`) are mostly based on these particle names; the use of the terms `SCAT`, `DK`, `THERM` are exceptions. Table 2 illustrates the scheme used.

Reactions names are constructed in MATXSR from the ENDF MT number, the LR flag (if present), and the incident particle name. Examples of the standard names are given in Tables 3–11. Note that the first `N` is omitted from the last three reactions in Table 3. It is implicit in the data-type name. This convention saves space in the name for possible breakup products (see Table 4). The first `N` is also implicit in the reactions of Table 4. No multiplicity is used in the breakup

Table 2: Standard Data-Type Names

Name	Data Type
NSCAT	neutron scattering
NG	neutron-induced gamma production
NP	neutron-induced proton production
NR	neutron-to-recoil matrix
GSCAT	gamma scattering
PSCAT	proton scattering
PN	proton-induced neutron production
...	...
NTHERM	thermal scattering data
DKN	delayed-neutron data
DKHG	decay heat and gamma data
DKB	decay beta data

Table 3: Simple Neutron-Emitting Reactions

Name	MT	Description
NELAS	2	neutron elastic scattering
NNONEL	3	neutron nonelastic (MT1–MT2)
NINEL	4	neutron inelastic sum (MT51–91)
N2N	16	(n,2n)
N3N	17	(n,3n)
NNA	22	(n,n'α)
NNP	28	(n,n'p)
NO1	51	(n,n_1)
NO2	52	(n,n_2)
NCN	91	(n,n') to continuum

product strings in order to avoid possible confusions with the discrete-level number; just count the like letters to get the multiplicity. The names used for the most common neutron absorption reactions are given in Table 5, and the names used for the fission reactions are given in Table 6. MATXS libraries typically give the total fission cross section and all the partial cross sections (when available) in the vector blocks, but they do not give the total fission matrix when the partial matrices are available.

Table 7 gives some additional reaction names, some of which are special NJOY names. As discussed in the GROUPR section of this report, the total cross section can be averaged with the ℓth order of the flux to obtain the multigroup total cross section components $\sigma_{t\ell,g}$. These total cross section components and the corresponding Legendre fluxes are given names like the first four shown in Table 7. Related names with first letters G, P, D, etc., will also be found in MATXS libraries.

Table 4: Breakup Reactions (LR flags)

Name	MT	LR	Description
NO7A	57	22	(n,n_7)α
N15P	65	28	(n,n_{15})p
NO2AA	52	29	(n,n_2)2α
NCNAAA	91	23	(n,n')3α
NO6NA	56	24	(n,n_5)nα
NO1EE	51	40	(n,n_1)ee

Table 5: Neutron-Absorption Reactions

Name	MT	Description
NABS	101	total absorption
NG	102	radiative capture
NP	103	(n,p)
NA	107	(n,α)

Table 6: Fission Reactions

Name	MT	Description
NFTOT	18	total fission
NF	19	(n,f) first-chance fission
NNF	20	(n,n'f) second-chance fission
N2NF	21	(n,n2f) third-chance fission
N3NF	38	(n,n3f) fourth-chance fission
NUDEL	455	delayed-neutron yield (MF3)
CHID	455	delayed-neutron spectrum (MF5)

The average inverse velocities are defined to preserve the time term of the time-dependent Boltzmann equation:

$$\left\langle \frac{1}{v} \right\rangle = \frac{\int_g (1/v)\phi(E)\,dE}{\int_g \phi(E)\,dE} . \qquad (1)$$

The meaning of the terms energy-balance heat production and kinematic KERMA factor are discussed in more detail in the HEATR chapter of this manual. Briefly, the energy-balance heating (MT301) is computed by subtracting the energy carried away by neutrons and photons from the energy available for a reaction. The result should be the energy deposited by charged particles and the recoil nucleus, that is, the local heating. Unfortunately, problems with the energy-balance consistency of evaluations, the difficulty of determining the available energy for elements, and the inaccuracy in the difference between relatively large numbers sometimes cause this value to have unphysical values (for example, negative heating). These values do have the property of always conserving energy for large systems. The kinematic value (MT443) is computed from reaction kinematics alone. It is very accurate at low energies, but when three or more particles are involved in the reaction, it begins to fail. The results in **KERMA** are always larger than the correct heating value. Comparing the two estimates for the local heating can reveal problems in the evaluations.[6, 7] The MATXS user is free to choose whichever number is more appropriate for the problem. The reaction **DAME** is also generated using data from HEATR. As discussed in the HEATR section of this report, this "damage-energy production" cross section can be used to obtain the DPA (displacements per atom) parameter used in radiation damage studies.

The gas-production reaction names given in Table 8 can also appear with other particle names before the decimal point. The names for reactions induced by incident charged particles follow the neutron names in most cases, except that the first letter is changed to indicate the incident particle type. Discrete-level scattering reactions are exceptions; **N01** is used for both (n,n_1) and (p,n_1). Also note the **N00** cannot be used for incident neutrons; the name **NELAS** is used instead. Similarly, **P00** is not used for incident protons.

As discussed in more detail below, scattering from thermal moderators is treated like *materials* in ENDF/B libraries, but it is treated like *reactions* on the GENDF files. The free-gas scattering reaction can appear in any material, but the other thermal MT numbers can only appear in the material corresponding to the dominant scattering isotope. For example, **H2O** only appears in H1. There are

Table 7: Special NJOY Names

Name	MT	Description
NTOTO	1	P_0 total cross section
NTOT1	1	P_1 total cross section
NWTO	1	P_0 weight function (flux)
NWT1	1	P_1 weight function (flux)
MUBAR	251	scattering $\bar{\mu}$
XI	252	scattering ξ
INVEL	259	inverse velocity (sec/m)
HEAT	301	energy-balance heat production
KERMA	443	kinematic KERMA factor
DAME	444	damage-energy production

two versions of ZRHYD; one appears in H1 and the other in Zr. The coherent and incoherent terms in the thermal cross section are kept separate for the convenience of applications; all the coherent names end with $. Note that MATXS files contain two different representations for the scattering cross sections at low energies:

static, where the cross section and group-to-group matrix are obtained from NSCAT, which is derived from MT2 on the ENDF evaluation (This is scattering for "static" nuclei; energy loss from recoil is included.); and

thermal, where the cross section and group-to-group matrix are obtained from one of the thermal reactions in the NTHERM data type. (The scattering nuclei are in motion with a distribution described by the Maxwell-Boltzmann law; both energy loss and energy gain events are possible.)

The TRANSX code gives the user the choice of static or thermal scattering, and it also allows the user to choose which binding state is desired for a particular moderator material.

The ENDF representation of photoatomic reactions is described in the GAMINR section of this report. The GHEAT reaction is constructed in GAMINR to represent the local heating due to atomic recoil and the photo-electric production of electrons. Flourescence photons from photoelectric interactions are assumed to deposit their energy locally.

The CCCC standards have always used 6-character Hollerith strings for names. These kinds of names are represented as "REAL*8" double precision variables on 32-bit machines (IBM, VAX, Sun, etc.) and as single-precision variables on 60- to 64-bit machines (CDC, Cray). However, a double-precision variable on a short-word machine can hold 8 characters. So can single-precision variables on CDC and Cray machines. There do not seem to be any computer systems currently in use that require 6-character words. Therefore, the latest versions of the MATXS

Table 8: Gas-Production Reactions

Name	MT	Description
N.NEUT	201	total neutron production
N.GAM	202	total γ production
N.H1	203	hydrogen production
N.H3	205	tritium production
N.HE4	207	helium production

Table 9: Incident-Proton Reactions

Name	MT	Description
PELAS	2	proton elastic scattering
P01	601	discrete-level (p,p_1) scattering
N00	50	discrete-level (p,n_0)
N01	51	discrete-level (p,n_1)
P2N	16	(p,2n)
PG	102	(p,γ)
PT	104	(p,t)

Table 10: Thermal Cross Sections

Name	MT	Description
FREE	221	free-gas scattering
H2O	222	H in H_2O
POLY	223	H in polyethylene (CH_2) incoherent
POLY$	224	H in polyethylene (CH_2) coherent
ZRHYD	225	H in ZrH incoherent
ZRHYD$	226	H in ZrH coherent
BENZ	227	Benzene incoherent
D2O	228	D in D_2O
GRAPH	229	C in graphite incoherent
GRAPH$	230	C in graphite coherent
BE	231	Be metal incoherent
BE$	232	Be metal coherent
BEO	233	BeO incoherent
BEO$	234	BeO coherent
ZRHYD	235	Zr in ZrH incoherent
ZRHYD$	236	Zr in ZrH coherent

Table 11: Photoatomic Cross Sections

Name	MT	Description
GTOTO	501	P_0 total
GWTO	501	P_0 weight function (flux)
GCOH	502	coherent scattering
GINCH	504	incoherent scattering
GPAIR	516	pair production $(\gamma, 2\gamma)$
GABS	522	photoelectric absorption
GHEAT	525	heating

format and the MATXSR module have been written to handle 8-character names.

The formal format specification for the MATXS material cross section file follows, using the standard CCCC presentation:

```
C*************************************************************************
C               PROPOSED 09/09/77                                        -
C                      (MODIFIED 09/80)                                  -
C                      (NOMENCLATURE CHANGED 06/88)                      -
C                      (MODIFIED FOR CONST SUB-BLOCKS 06/90)             -
C                      (ORDERING CHANGED 10/90)                          -
C                                                                        -
CF          MATXS                                                        -
CE          MATERIAL CROSS SECTION FILE                                  -
C                                                                        -
CN                     THIS FILE CONTAINS CROSS SECTION                  -
CN                     VECTORS AND MATRICES FOR ALL                      -
CN                     PARTICLES, MATERIALS, AND REACTIONS;              -
CN                     DELAYED NEUTRON SPECTRA BY TIME GROUP;            -
CN                     AND DECAY HEAT AND PHOTON SPECTRA.                -
C                                                                        -
CN          FORMATS GIVEN ARE FOR FILE EXCHANGE ONLY                     -
C                                                                        -
C*************************************************************************
C
C
C-------------------------------------------------------------------------
CS          FILE STRUCTURE                                               -
CS                                                                       -
CS              RECORD TYPE                          PRESENT IF          -
CS              ==============================       ===============     -
CS              FILE IDENTIFICATION                    ALWAYS            -
CS              FILE CONTROL                           ALWAYS            -
CS              SET HOLLERITH IDENTIFICATION           ALWAYS            -
CS              FILE DATA                              ALWAYS            -
CS                                                                       -
CS   *************(REPEAT FOR ALL PARTICLES)                            -
CS   *          GROUP STRUCTURES                       ALWAYS            -
CS   *************                                                       -
CS                                                                       -
```

```
CS   *************(REPEAT FOR ALL MATERIALS)                              -
CS   *          MATERIAL CONTROL                    ALWAYS               -
CS   *                                                                   -
CS   * ************(REPEAT FOR ALL SUBMATERIALS)                        -
CS   * *        VECTOR CONTROL                      N1DB.GT.0            -
CS   * *                                                                 -
CS   * * **********(REPEAT FOR ALL VECTOR BLOCKS)                       -
CS   * * *      VECTOR BLOCK                        N1DB.GT.0            -
CS   * * **********                                                      -
CS   * *                                                                -
CS   * * **********(REPEAT FOR ALL MATRIX BLOCKS)                       -
CS   * * *      MATRIX CONTROL                      N2D.GT.0             -
CS   * * *                                                              -
CS   * * * ********(REPEAT FOR ALL SUB-BLOCKS)                          -
CS   * * * *    MATRIX SUB-BLOCK                    N2D.GT.0             -
CS   * * * ********                                                     -
CS   * * *                                                              -
CS   * * *      CONSTANT SUB-BLOCK                  JCONST.GT.0          -
CS   * * *                                                              -
CS   *************                                                      -
C                                                                        -
C-------------------------------------------------------------------------
C
C
C-------------------------------------------------------------------------
CR          FILE IDENTIFICATION                                          -
C                                                                        -
CL    HNAME,(HUSE(I),I=1,2),IVERS                                        -
C                                                                        -
CW    1+3*MULT                                                           -
C                                                                        -
CB    FORMAT(4H OV ,A8,1H*,2A8,1H*,I6)                                   -
C                                                                        -
CD    HNAME          HOLLERITH FILE NAME  - MATXS -  (A8)                -
CD    HUSE           HOLLERITH USER IDENTIFICATION   (A8)                -
CD    IVERS          FILE VERSION NUMBER                                 -
CD    MULT           DOUBLE PRECISION PARAMETER                          -
CD                        1- A8 WORD IS SINGLE WORD                      -
CD                        2- A8 WORD IS DOUBLE PRECISION WORD            -
C                                                                        -
C-------------------------------------------------------------------------
C
C
C-------------------------------------------------------------------------
CR          FILE CONTROL                                                 -
C                                                                        -
CL    NPART,NTYPE,NHOLL,NMAT,MAXW,LENGTH                                 -
C                                                                        -
CW    6                                                                  -
C                                                                        -
CB    FORMAT(4H 1D ,4I6)                                                 -
C                                                                        -
CD    NPART       NUMBER OF PARTICLES FOR WHICH GROUP                    -
```

```
CD                    STRUCTURES ARE GIVEN                                  -
CD    NTYPE       NUMBER OF DATA TYPES PRESENT IN SET                        -
CD    NHOLL       NUMBER OF WORDS IN SET HOLLERITH                           -
CD                    IDENTIFICATION RECORD                                  -
CD    NMAT        NUMBER OF MATERIALS ON FILE                                -
CD    MAXW        MAXIMUM RECORD SIZE FOR SUB-BLOCKING                       -
CD    LENGTH      LENGTH OF FILE                                             -
C                                                                            -
C-------------------------------------------------------------------------
C
C
C-------------------------------------------------------------------------
CR           SET HOLLERITH IDENTIFICATION                                    -
C                                                                            -
CL    (HSETID(I),I=1,NHOLL)                                                  -
C                                                                            -
CW    NHOLL*MULT                                                             -
C                                                                            -
CB    FORMAT(4H 2D ,8A8/(9A8))                                               -
C                                                                            -
CD    HSETID      HOLLERITH IDENTIFICATION OF SET (A8)                       -
CD                 (TO BE EDITED OUT 72 CHARACTERS PER LINE)                 -
C                                                                            -
C-------------------------------------------------------------------------
C
C
C-------------------------------------------------------------------------
CR          FILE DATA                                                        -
C                                                                            -
CL    (HPRT(J),J=1,NPART),(HTYPE(K),K=1,NTYPE),(HMATN(I),I=1,NMAT),          -
CL   1(NGRP(J),J=1,NPART),(JINP(K),K=1,NTYPE),(JOUTP(K),K=1,NTYPE),          -
CL   2(NSUBM(I)I=1,NMAT),(LOCM(I),I=1,NMAT)                                  -
C                                                                            -
CW    (NPART+NTYPE+NMAT)*MULT+2*NTYPE+NPART+2*NMAT                           -
C                                                                            -
CB    FORMAT(4H 3D ,8A8/(9A8))      HPRT,HTYPE,HMATN                         -
CB    FORMAT(12I6)                  NGRP,JINP,JOUTP,NSUBM,LOCM               -
C                                                                            -
CD    HPRT(J)     HOLLERITH IDENTIFICATION FOR PARTICLE J                    -
CD                     N          NEUTRON                                    -
CD                     G          GAMMA                                      -
CD                     P          PROTON                                     -
CD                     D          DEUTERON                                   -
CD                     T          TRITON                                     -
CD                     H          HE-3 NUCLEUS                               -
CD                     A          ALPHA (HE-4 NUCLEUS)                       -
CD                     B          BETA                                       -
CD                     R          RESIDUAL OR RECOIL                         -
CD                                (HEAVIER THAN ALPHA)                       -
CD    HTYPE(K)     HOLLERITH IDENTIFICATION FOR DATA TYPE K                   -
CD                     NSCAT      NEUTRON SCATTERING                         -
CD                     NG         NEUTRON INDUCED GAMMA PRODUCTION           -
CD                     GSCAT      GAMMA SCATTERING                           -
```

```
CD                    PN        PROTON INDUCED NEUTRON PRODUCTION     -
CD                      .          .                                   -
CD                      .          .                                   -
CD                      .          .                                   -
CD                    DKN       DELAYED NEUTRON DATA                  -
CD                    DKHG      DECAY HEAT AND GAMMA DATA             -
CD                    DKB       DECAY BETA DATA                       -
CD    HMATN(I)    HOLLERITH IDENTIFICATION FOR MATERIAL I              -
CD    NGRP(J)      NUMBER OF ENERGY GROUPS FOR PARTICLE J              -
CD    JINP(K)     TYPE OF INCIDENT PARTICLE ASSOCIATED WITH            -
CD                   DATA TYPE K.  FOR DK DATA TYPES, JINP IS 0.      -
CD    JOUTP(K)    TYPE OF OUTGOING PARTICLE ASSOCIATED WITH            -
CD                   DATA TYPE K                                      -
CD    NSUBM(I)    NUMBER OF SUBMATERIALS FOR MATERIAL I                -
CD    LOCM(I)     LOCATION OF MATERIAL I                               -
C                                                                      -
C----------------------------------------------------------------------
C
C
C----------------------------------------------------------------------
CR          GROUP STRUCTURE                                            -
C                                                                      -
CL    (GPB(I),I=1,NGR),EMIN                                            -
C                                                                      -
CC    NGR=NGRP(J)                                                      -
C                                                                      -
CW    NGRP(J)+1                                                        -
C                                                                      -
CB    FORMAT(4H 4D ,1P5E12.5/(6E12.5))                                 -
C                                                                      -
CD    GPB(I)      MAXIMUM ENERGY BOUND FOR GROUP I FOR PARTICLE J      -
CD    EMIN        MINIMUM ENERGY BOUND FOR PARTICLE J                  -
C                                                                      -
C----------------------------------------------------------------------
C
C
C----------------------------------------------------------------------
CR          MATERIAL CONTROL                                           -
C                                                                      -
CL    HMAT,AMASS,(TEMP(I),SIGZ(I),ITYPE(I),N1D(I),N2D(I),              -
CL   1LOCS(I),I=1,NSUBM)                                               -
C                                                                      -
CW    MULT+1+6*NSUBM                                                   -
C                                                                      -
CB    FORMAT(4H 6D ,A8,1H*,1P2E12.5/(2E12.5,5I6))                      -
C                                                                      -
CD    HMAT        HOLLERITH MATERIAL IDENTIFIER                        -
CD    AMASS       ATOMIC WEIGHT RATIO                                  -
CD    TEMP        AMBIENT TEMPERATURE OR OTHER PARAMETERS FOR          -
CD                    SUBMATERIAL I                                    -
CD    SIGZ        DILUTION FACTOR OR OTHER PARAMETERS FOR              -
CD                    SUBMATERIAL I                                    -
CD    ITYPE       DATA TYPE FOR SUBMATERIAL I                          -
```

```
CD    N1D          NUMBER OF VECTORS FOR SUBMATERIAL I                -
CD    N2D          NUMBER OF MATRIX BLOCKS FOR SUBMATERIAL I          -
CD    LOCS         LOCATION OF SUBMATERIAL I                          -
C                                                                      -
C---------------------------------------------------------------------
C
C
C---------------------------------------------------------------------
CR          VECTOR CONTROL                                             -
C                                                                      -
CL    (HVPS(I),I=1,N1D),(NFG(I),I=1,N1D),(NLG(I),I=1,N1D)              -
C                                                                      -
CW    (MULT+2)*N1D                                                     -
C                                                                      -
CB    FORMAT(4H 7D ,8A8/(9A8))          HVPS                           -
CB    FORMAT(12I6)                      IBLK,NFG,NLG                   -
C                                                                      -
CD    HVPS(I)      HOLLERITH IDENTIFIER OF VECTOR                      -
CD                       NELAS     NEUTRON ELASTIC SCATTERING          -
CD                       N2N       (N,2N)                              -
CD                       NNF       SECOND CHANCE FISSION               -
CD                       GABS      GAMMA ABSORPTION                    -
CD                       P2N       PROTONS IN, 2 NEUTRONS OUT          -
CD                          .           .                              -
CD                          .           .                              -
CD                          .           .                              -
CD    NFG(I)       NUMBER OF FIRST GROUP IN BAND FOR VECTOR I          -
CD    NLG(I)       NUMBER OF LAST GROUP IN BAND FOR VECTOR I           -
C                                                                      -
C---------------------------------------------------------------------
C
C
C---------------------------------------------------------------------
CR          VECTOR BLOCK                                               -
C                                                                      -
CL    (VPS(I),I=1,KMAX)                                                -
C                                                                      -
CC    KMAX=SUM OVER GROUP BAND FOR EACH VECTOR IN BLOCK J              -
C                                                                      -
CW    KMAX                                                             -
C                                                                      -
CB    FORMAT(4H 8D ,1P5E12.5/(6E12.5))                                 -
C                                                                      -
CD    VPS(I)       DATA FOR GROUP BANDS FOR VECTORS IN BLOCK J.        -
CD                 BLOCK SIZE IS DETERMINED BY TAKING ALL THE GROUP    -
CD                 BANDS THAT HAVE A TOTAL LENGTH LESS THAN OR EQUAL   -
CD                 TO MAXW.                                            -
C                                                                      -
C---------------------------------------------------------------------
C
C
C---------------------------------------------------------------------
CR        SCATTERING MATRIX CONTROL                                    -
```

```
C                                                                       -
CL    HMTX,LONE,LORD,JCONST,                                            -
CL   1(JBAND(L),L=1,NOUTG(K)),(IJJ(L),L=1,NOUTG(K))                     -
C                                                                       -
CW    MULT+3+2*NOUTG(K)                                                 -
C                                                                       -
CB    FORMAT(4H 9D ,A8/(12I6))          HMTX,LONE,LORD,JCONST,          -
CB                                       JBAND,IJJ                      -
C                                                                       -
CD    HMTX         HOLLERITH IDENTIFICATION OF BLOCK                    -
CD    LONE         LOWEST ORDER PRESENT                                 -
CD    LORD         NUMBER OF ORDERS PRESENT                             -
CD    JCONST       NUMBER OF GROUPS WITH CONSTANT SPECTRUM              -
CD    JBAND(L)     BANDWIDTH FOR GROUP L                                -
CD    IJJ(L)       LOWEST GROUP IN BAND FOR GROUP L                     -
C                                                                       -
C-----------------------------------------------------------------------
C
C
C-----------------------------------------------------------------------
CR          SCATTERING SUB-BLOCK                                        -
C                                                                       -
CL    (SCAT(K),K=1,KMAX)                                                -
C                                                                       -
CC    KMAX=LORD TIMES THE SUM OVER ALL JBAND IN THE GROUP RANGE OF      -
CC            THIS SUB-BLOCK                                            -
C                                                                       -
CB    FORMAT(5H 10D ,1P5E12.5/(6E12.5))                                 -
C                                                                       -
CW    KMAX                                                              -
C                                                                       -
CD    SCAT(K)      MATRIX DATA GIVEN AS BANDS OF ELEMENTS FOR INITIAL   -
CD                 GROUPS THAT LEAD TO EACH FINAL GROUP.  THE ORDER     -
CD                 OF THE ELEMENTS IS AS FOLLOWS:  BAND FOR P0 OF       -
CD                 GROUP I, BAND FOR P1 OF GROUP I, ... , BAND FOR P0   -
CD                 OF GROUP I+1, BAND FOR P1 OF GROUP I+1, ETC.  THE    -
CD                 GROUPS IN EACH BAND ARE GIVEN IN DESCENDING ORDER.   -
CD                 THE SIZE OF EACH SUB-BLOCK IS DETERMINED BY THE      -
CD                 TOTAL LENGTH OF A GROUP OF BANDS THAT IS LESS THAN   -
CD                 OR EQUAL TO MAXW.                                    -
CD                                                                      -
CD                 IF JCONST.GT.0, THE CONTRIBUTIONS FROM THE JCONST    -
CD                 LOW-ENERGY GROUPS ARE GIVEN SEPARATELY.              -
C                                                                       -
C-----------------------------------------------------------------------
C                                                                       -
C                                                                       -
C-----------------------------------------------------------------------
CR          CONSTANT SUB-BLOCK                                          -
C                                                                       -
CL    (SPEC(L),L=1,NOUTG(K)),(PROD(L),L=L1,NING(K))                     -
C                                                                       -
CC    L1=NING(K)-JCONST+1                                               -
```

```
C                                                                       -
CW    NOUTG(K)+JCONST                                                   -
C                                                                       -
CB    FORMAT(4H11D ,1P5E12.5/(6E12.5))                                  -
C                                                                       -
CD    SPEC        NORMALIZED SPECTRUM OF FINAL PARTICLES FOR INITIAL    -
CD                PARTICLES IN GROUPS L1 TO NING(K)                     -
CD    PROD        PRODUCTION CROSS SECTION (E.G., NU*SIGF) FOR          -
CD                INITIAL GROUPS L1 THROUGH NING(K)                     -
CD                                                                      -
CD          THIS OPTION IS NORMALLY USED FOR THE ENERGY-INDEPENDENT      -
CD          NEUTRON AND PHOTON SPECTRA FROM FISSION AND RADIATIVE        -
CD          CAPTURE USUALLY SEEN AT LOW ENERGIES.                        -
C                                                                       -
C-----------------------------------------------------------------------
```

The MATXS format is intended to communicate multigroup cross sections and matrices for all reaction types, incident particles, and outgoing particles from a nuclear data processing code to applications. It also includes temperature and self-shielding effects, delayed-neutron data, and a limited format for decay heat and delayed photon or particle emission. As shown in the "File Structure" presentation above, the main loop is over material. Materials are subdivided into submaterials, which usually correspond to different data types, temperatures, and background cross section (σ_0) values. Each submaterial can contain a series of "Vector Blocks" giving cross section versus energy for one of the allowed group structures and incident particles, and it can contain a series of matrix blocks and subblocks giving the cross sections for group-to-group transfers.

The "File Identification" record is the same for all CCCC files. It gives the Hollerith name for the file (which is always **MATXS**), a version number **IVERS**, which can be used to distinguish between different libraries in this format, and a Hollerith identification string **HUSE**, which can be used for entries like "**T2 LANL NJOY**".

The "File Control" record contains parameters that are needed to compute the lengths of the following records. The meaning of the various names is well explained in the format specification, and the values of the parameters are obtained from the user's input. The purpose of **MAXW** is to tell application codes how much memory they will need to read through the records on these MATXS files. It is used for both vector blocks and matrix subblocks when deciding how to break them up. The MATXSR value is 5000 words. The code tries to make as many records as possible that have nearly this size in order to minimize the number of I/O operations. The parameter **LENGTH** is used to help find the end of the file when appending a new material to an existing file. Its units are left unspecified in the format. It is usually the length in records, in which case record skipping can be used to find

the end. Or it could be the length in words on computer systems that allow direct word-addressed I/O operations (for example, CTSS on Cray computers).

The "Set Hollerith Identification" record comes next. It contains an arbitrary amount of Hollerith text to describe the contents of the library. The description comes from the user's input.

The "File Data" record contains a number of important arrays that define the structure of data types and the location of materials. The parameter HPRT contains the standard names for the NPART particles. The standard names for particles were discussed above. The standard names for the data types and materials (HTYPE and HMATN) were also discussed above. The next three parameters are used to complete the specification of the data types included in this MATXS library. NGRP just gives the number of groups used for each particle type; for example, the traditional Los Alamos 30×12 library (with 30 neutron groups and 12 photon groups) would have NGRP(1)=30 and NGRP(2)=12. Using the same example, the NSCAT data type would have JINP(1)=1 and JOUTP(1)=1; the NG data type would have JINP(2)=1 and JOUTP(2)=2; and the GSCAT data type would have JINP(3)=2 and JOUTP(3)=2. The information for these 6 arrays is given in the user's input.

The final two parameters in the "File Data" record are NSUBM and LOCM. The value of NSUBM depends on the number of data types, the number of temperatures, and the number of background cross sections found on the input GENDF tapes. The value for LOCM(I) is usually the record index for material I. A code can then jump directly to a desired material using record skipping (forward or backward). However, the units for LOCM have been left unspecified to allow direct random access for systems that use word-addressable random-access I/O operations (such as CTSS on Cray computers).

The "Group Structure" records give the energy bounds for NPART group structures. Following the normal convention for application codes, the energy bounds are given in the order of decreasing energy. The numbers are obtained from MF1,MT451 on the GENDF tape. The GROUPR module currently uses one group structure for all particles (n, p, α, etc.), and another for photons (γ).

Inside the material loop, there is a "Material Control" record for each material. The choice of names for HMAT was discussed above. The AMASS parameter is the same as the ENDF AWR parameter; that is, it is the ratio of the target mass to the neutron mass. Temperatures TEMP are given in degrees Kelvin, and background cross sections for self-shielding codes (SIGZ, or σ_0) are given in barns. The parameter ITYPE tells which data-type each submaterial belongs to using the data type codes defined by HTYPE. Although it is not specified in the format de-

scription, the order that MATXSR loops through submaterials is as follows: the outer loop is over data type, the next loop is over temperature, and the innermost loop is over background cross section. The number of cross section vector reactions and matrix reactions for each submaterial are given in N1D and N2D. Finally, LOCS(I) normally gives the record index for submaterial I. A code can search the arrays TEMP, SIGZ, and ITYPE for a desired submaterial, and then jump right to the desired submaterial using record skipping. Alternatively, a version that uses word addresses in LOCS could use random-access methods to jump to the desired submaterial.

Each submaterial that contains vectors (N1D>0) starts with a "Vector Control" record. This record gives a list of reaction names in HVPS. MATXSR constructs these reaction names automatically based on the MT number, ENDF "LR flag" (if any), and the incident particle type. Examples of these names were given above. The parameters NFG and NLG are used to remove unnecessary leading or trailing zeros in reaction cross section vectors. The zeros are usually due to thresholds. For example, an (n,2n) reaction might only have nonzero cross sections for groups 1 through 5 out of an 80-group structure. Storing only the 5-element band will save 75 words on the MATXS file. Even with the zeros removed, the number of words of vector cross section data for a submaterial can be quite large. Therefore, the MATXS format provides a way to break the vector data into a number of vector blocks. The idea is to sum up the bandwidths for each reaction (that is, NLG(I)-NFG(I)+1) in order to find the largest number of reactions that will fit within a block of length less than or equal to MAXW. The data for this block are written to the output file, and then the next group of reactions are found. This continues until all the vector data have been written out as a series of "Vector Block" records with lengths less than MAXW (usually 5000 words in MATXSR). This method minimizes the number of records on the file while allowing the application codes that read MATXS libraries to allocate space for reading the records economically.

Most submaterials will contain N2B "Scattering Matrix Control" records. The convention for the reaction names used in HMTX were discussed above. LORD gives the *number* of Legendre orders present for this reaction; that is, LORD=4 for a P_3 matrix. Matrices are compacted for efficient storage and data transfer using two techniques. First, unnecessary leading and trailing zeros for group-to-group transfers into a particular final group are removed by banding. JBAND(I) gives the number of incident groups in the band for final group I, and IJJ(I) gives the group index for the lowest-energy group (highest group index) in the band for group I. For example, consider an isotropic (n,2n) reaction with a threshold in

group 3 of a 30-group structure. The values for JBAND and IJJ for group 20 might well be 3 and 3, respectively. That is, groups 1 through 3 scatter into group 20. The order of storage for this matrix would be as follows:

Band	Element	JBAND	IJJ
1	1→1	1	1
2	2→2	2	2
	1→2		
3	3→3	3	3
	2→3		
	1→3		
4	3→4	3	3
	2→4		
	1→4		
...	...		

Note that this scheme is more efficient than the similar one used for the CCCC ISOTXS file, because that method required that the ingroup element had to be included in the band. For anisotropic matrices, the LORD Legendre components are stored with the source-group elements. Thus, there are LORD(I)*JBAND(I) elements in each band, and they are stored in the following order (assuming a P_1 matrix):

Band	Element	Order	JBAND	IJJ
1	1→1	0	1	1
	1→1	1		
2	2→2	0	2	2
	1→2	0		
	2→2	1		
	1→2	1		
3	3→3	0	3	3
	2→3	0		
	1→3	0		
	3→3	1		
...	...			

This kind of matrix block can be subdivided into subblocks using a method similar to the one described above for vector cross sections. The code starts summing the product of JBAND and LORD for the final energy groups until the data for the next band will cause the sum to exceed MAXW (normally 5000 words in MATXSR). These data are then written out as a matrix subblock. The code then repeats the process for the rest of the group ranges. The result is a minimal number of "Scattering Sub-Blocks," none of which has a size larger than the file limit.

The second method used to compact scattering matrices is based on the observation that the shape of the outgoing neutron or photon spectrum from fission and

radiative capture reactions tends to be independent of energy at low neutron energies. GROUPR determines the group where significant energy dependence begins. Below this point, it computes a single spectrum to describe the outgoing neutron or photon distribution and a production cross section to go with it. At high energies, it produces a group-to-group matrix. This method can lead to appreciable reductions in storage requirements. For example, consider the 187-group structure, which has many low-energy groups. The fission spectrum for ^{235}U doesn't begin to show significant energy dependence until an energy of about 9 keV. This means that there are 118 constant groups. A 187×187 matrix is thereby reduced to a 187×69 matrix, and 187-group vector, and a 118-group vector—a 62 % reduction in storage requirements. Even larger reductions in size are obtained in more favorable cases. The parameter JCONST gives the number of low-energy groups having a constant spectrum. If JCONST>0, a single "Constant Sub-Block" record will be given after the regular "Scattering Sub-Block" records. This record will contain the spectrum SPEC and the production cross section PROD needed to reconstruct the low-energy part of the full matrix. In mathematical form,

$$\sigma_{x,g\to g'} = \chi^{LE}_{g'}\,\sigma^{LE}_{Px,g} + \sigma^{HE}_{x,g\to g'} \ , \tag{2}$$

where LE stands for low energy, HE stands for high energy, χ^{LE} is the constant spectrum, σ^{LE}_{Px} is the production cross section for reaction x, and σ^{HE} is the normal group-to-group matrix.

C. Historical Notes

As noted at the top of the MATXS format description, the original version of the file was constructed in September of 1977. The "Matrix Control" record in this original version contained the names and banding parameters for every group of every reaction. As a result, the record could become very large for libraries with many groups. The MATXS format was modified in September of 1980 to have a different "Matrix Control" record for each reaction. For several years following this date, NJOY contained both MATXSR and NMATXS modules, and two different versions of TRANSX were in use. All traces of the original version of the format have now disappeared.

The last few changes are all fairly recent. The changes of June 1988 were introduced to make MATXSR able to handle data types with either incoming or outgoing charged particles. Actually, the format wasn't changed. The particle, data type, and reaction name conventions described in Tables 1–11 were chosen, and these new names required some corresponding changes in the code. The

TRANSX code also had to be modified to recognize the new names and to work with multiparticle coupled sets.

The concept of using constant subblocks to reduce the size of GENDF and MATXS files is actually quite old. Various versions of updates to install the scheme were written over about a 5-year period. It was finally decided to permanently install the scheme in June of 1990. The changes required to GROUPR were the easier part of the job; corresponding changes were required in DTFR, CCCCR, MATXSR, POWR, and WIMSR. In addition, TRANSX had to be changed to accept the new constant subblocks. Unfortunately, this change is large enough to impact all MATXS users.

If many users were to be impacted, other nagging problems with the code could also be corrected at this time. With the original format, it was always very difficult to add, replace, or extract a material. The data type loop was outside the material loop, and the pieces for a given material were scattered throughout the file. Therefore, the ordering of the file was changed to put the data type loop inside the material loop along with the other submaterials. With this change, it was easy to rewrite the BBC code to be able to insert a new material at any point in the library. It was also easy to give BBC the capability of extracting a short library containing only selected materials from a large MATXS library. Using the short library can significantly speed up TRANSX runs to prepare data for reactor design problems. The only disadvantage of the new ordering is that identical photoatomic data blocks have to be given for each isotope of an element. However, this data is not too bulky, and the additional overhead is manageable. This arrangement might make it easier to add photonuclear data in the future.

Finally, the current scheme of dividing the vector and matrix data into subblocks was added. The goal was to keep the record size below some fairly large maximum value (5000 words is currently being used). This kind of limit makes it easier to design application codes that make efficient use of memory. In addition, using as few records as possible reduces the number of I/O operations needed for a trip through the library, thereby improving execution time. The new subblocking scheme is also very simple.

D. MATXS Libraries

The normal process for preparing a MATXS library using NJOY starts with a series of runs to prepare PENDF tapes for each of the materials of interest. (For incident neutrons, these PENDF runs normally involve running the modules RECONR, BROADR, UNRESR, HEATR, and THERMR. For incident photons and

charged particles, only RECONR is needed.) The next step is to run GROUPR for each material and incident particle. Each GROUPR run can produce data for all outgoing particles and for photons. As an example, consider the problem of producing data for a coupled neutron-photon-proton library. GROUPR would be run using the ENDF-6 incident-neutron data and requesting MFD values of 3 (cross sections), 6 (neutron matrices, nn), 16 (photon production matrices, $n\gamma$), and 21 (proton production matrices, np). GAMINR would be run to produce the photoatomic cross sections and $\gamma\gamma$ matrices (no photonuclear data are currently available, so γn and γp matrices cannot be generated). GROUPR would be run again for the ENDF-6 proton sublibrary, requesting MFD values of 3 (cross sections), 6 (neutron production, pn), 16 (photon production, $p\gamma$), and 21 (proton scattering and production, pp).

Once all the GENDF fragments from all these GROUPR and GAMINR runs are available, they can be merged into multimaterial GENDF tapes using MODER. For the example above, three multimaterial GENDF tapes should be prepared: one for incident neutrons, one for incident photons, and one for incident protons. The input description in the following section shows how the unit numbers for these three GENDF files are given to MATXSR.

When the MATXSR run is made, the code scans through all the input GENDF tapes searching for the specified materials. For each material, it extracts all the data types requested in the MATXSR input and appearing on the input tapes. In addition, it extracts every temperature, σ_0 value, reaction, and Legendre order found on the input tapes. The following paragraphs describe various special features of the formatting process.

1. Normal Neutron and Photon Data. Normal, infinitely dilute cross sections and group-to-group matrices for neutron reactions, photon production, and photonuclear reactions are read from the input tapes and stored in the MATXS file using the formats described above. Almost all partial reactions are kept. Reaction names are constructed automatically; a number of examples were given above in Tables 3–11. The total cross section and weighting flux from GROUPR are given in the section MF3,MT1 with both P_0 and P_1 components. All four vectors are written to the MATXS file. A similar treatment is used for MT501 in the photoatomic case, except that only P_0 terms are saved.

The treatment of fission is also special. ENDF libraries sometimes use only MT18, the total fission reaction, for both cross section and emission spectrum data; sometimes the partial fission cross sections (n,f), (n,n'f), (n,2nf), and (n,3nf)

are also given in File 3 (MT19, 20, 21, and 38); and sometimes the partial reactions are also used to describe the neutron emission matrix. If the last case is found, the MT18 matrix may not be complete above the threshold for second-chance fission, and it should be ignored. Normally, it would not be processed in GROUPR, but just in case, MATXSR will ignore it. Delayed-neutron spectra processed in GROUPR are written into the section labeled MF5,MT455 on the GENDF tape, and this section includes data for all six time groups. MATXSR adds up these separate time-group spectra to produce a single delayed-neutron spectrum for the MATXS vector data blocks. Application codes can use the fission data in the MATXS library to construct steady-state fission vectors as follows:

$$\bar{\nu}_g^{SS} = \frac{\sum_{g'} \sigma_{fg\to g'}^{HE} + \sigma_{Pfg}^{LE} + \bar{\nu}_g^D \sigma_{fg}}{\sigma_{fg}} , \tag{3}$$

where σ^{HE} is the matrix part of the prompt fission reaction, σ_{Pfg}^{LE} is the low-energy production cross section for fission neutrons, $\bar{\nu}^D$ is the total delayed-neutron yield, and σ_{fg} is the fission cross section. If partial fission matrices are available, the first two terms in the denominator would also have to be summed over the reactions present. Continuing,

$$\chi_{g'}^{SS} = \frac{\sum_g \sigma_{fg\to g'}^{HE} \phi_g + \chi_{g'}^{LE} \sum_g \sigma_{Pfg}^{LE} \phi_g + \chi_{g'}^D \sum_g \bar{\nu}_g^D \sigma_{fg} \phi_g}{\text{NORM}} , \tag{4}$$

where χ^{LE} is the constant-spectrum part of the fission reaction, χ^D is the total delayed-neutron spectrum, and NORM is the quantity that normalizes χ^{SS}, namely, the sum of the numerator over all g'. The weighting flux ϕ_g would normally be problem- and region-dependent in the application code.

2. Self-Shielding Data. If higher temperatures and σ_0 values are found on the input GENDF file, MATXSR automatically prepares submaterials for them. This self-shielding information can be used by application codes to prepare effective cross sections for mixtures of materials in various geometrical arrangements using equivalence theory. This approach is often called the Bondarenko method.[8]

As discussed in more detail in the GROUPR section of this report, this system is based on using a model flux for isotope i of the form

$$\phi_\ell^i(E,T) = \frac{C(E)}{[\sigma_0^i + \sigma_t^i(E,T)]^{\ell+1}} , \tag{5}$$

where $C(E)$ is a smooth weighting flux, $\sigma_t^i(E,T)$ is the total cross section for material i at temperature T, ℓ is the Legendre order, and σ_0^i is a parameter that can be used to account for the presence of other materials and the possibility of escape from the absorbing region (heterogeneity). GROUPR uses this model flux to calculate effective multigroup cross sections for the resonance-region reactions (total, elastic, fission, capture) for several values of σ_0 and several values of T.

When σ_0 is large with respect to the highest peaks in σ_t, the flux is essentially proportional to $C(E)$. This is called infinite dilution, and the corresponding cross sections are appropriate for an absorber in a dilute mixture or for a very thin sample of the absorber. As σ_0 decreases, the flux $\phi(E)$ develops dips where σ_t has peaks. These dips cancel out part of the effect of the corresponding peaks in the resonance cross sections, thereby reducing, or self-shielding the reaction rate. In the MATXS format, these effects are represented by differences instead of the f-factors used in earlier formats (see the description of the BRKOXS file in the CCCCR section of this report). That is, the submaterial corresponding to temperature T and background cross section σ_0 contains the differences between the cross sections found on the GENDF tape for those parameters and the infinitely dilute values found in the first submaterial (normally, T=300 K and σ_0=1×10^{10}barns).

This approach has two advantages. First, data for groups with no self-shielding are automatically removed from MATXS vectors and matrices by the banding process without having to violate the principle of *generalization* and define a special format for neglecting values of "1.0" instead of values of "0.0." Second, effective cross sections can be accumulated by simply adding the self-shielding effects multiplied by appropriate interpolation weights to an accumulating sum that starts out equal to the infinitely dilute cross section. With f-factors, it is necessary to save the infinitely dilute cross sections during each step so that they can be multiplied by the f-factors. Thus, using differences is more economical in coding and in storage space requirements. The TRANSX code makes good use of this feature.

3. Thermal Data. ENDF-6 thermal data comes from a thermal sublibrary. In this sublibrary, the various material configurations are handled as separate materials. However, after the ENDF data have been processed by the THERMR module, the different thermal materials are handled as reactions. Table 10 gives the correspondence between the special NJOY thermal MT numbers and the actual thermal materials.

Each of these thermal reactions corresponds to a particular dominant material. For example, MT222 gives the thermal cross section for hydrogen bound in water.

It will only appear in the material H1 in a MATXS library. Similarly, MT223–227 will only appear in H1, MT228 will only appear in H2, MT229–230 will only appear in CNAT, MT231–234 will only appear in BE9, and MT235–236 will only appear in ZRNAT. Free-gas scattering (MT221) will appear in all materials.

Many of these materials describe the scattering from one atom of a compound bound in that compound; for example, H in H_2O, D in D_2O, or Zr in ZrH. The application code using this data is expected to add on the effects of scattering from the other atoms of the compound. For water, the scattering from free oxygen is added to the "H in H_2O" scattering. For zirconium, the scattering for "H in ZrH" is added to the scattering from "Zr in ZrH". There are two exceptions in the existing ENDF/B evaluations. The benzene data set contains the entire scattering from the C_6H_6 molecule normalized to the hydrogen cross section. Therefore, if an application code specifies H1 with the benzene scattering option and the correct density for H1 in the system, the result will contain all of the benzene scattering effect. No additional scattering is to be added for the carbon atoms. Similarly, BeO contains all the scattering from the compound normalized to the beryllium cross section. Check the THERMR section of this report for more information.

Four of these materials have names ending with $. These reactions represent coherent elastic scattering from crystalline powdered materials (C, Be, BeO) or incoherent elastic scattering from solids containing hydrogen (polyethylene). These reactions contain ingroup elements only in the scattering matrices; that is, they cause angular redistribution without energy loss in scattering. Each $ reaction should be added to the corresponding inelastic reaction by the application code. This part of the scattering can lead to difficulties with transport corrections in discrete-ordinates transport codes.

The final unique aspects of the thermal data are that the matrices show upscatter, and they are only defined below some maximum energy. The energy range aspect is easily handled by the banding method used to reduce the size of the vector blocks. The upscatter aspect is handled by `JBAND` and `IJJ`. The order of storage for a simple thermal case with only 2 upscatter groups would be as follows:

Band	Element	JBAND	IJJ
...	...		
27	27→27	2	27
	26→27		
28	29→28	3	29
	28→28		
	27→28		
29	30→29	3	30
	29→29		
	28→29		
30	30→30	2	30
	29→30		

4. Charged-Particle Data. The treatment of data for incident charged particles is very similar to that used for neutron data, except for charged-particle elastic scattering. The elastic channel has contributions from Coulomb scattering that become infinite as the scattering angle goes to zero. In nature, this singularity is removed by electronic shielding, but some other approach is needed for a data set that concentrates on isolated nuclear reactions. The approach selected is used in some existing applications. The elastic scattering distribution is broken up into two parts: (1) a normal angular distribution for angles from some low cutoff, say 20°, back to 180°, and (2) a straight-ahead continuous slowing-down contribution to represent the effects of angles below the cutoff. The continuous slowing-down part is closely related to the normal "stopping power" for charged particles. The large-angle part can be converted into a normal scattering matrix (see the discussion of charged-particle elastic scattering in the GROUPR chapter of this manual for more details).

Discrete-ordinates transport codes can often be modified to handle charged-particle data in this form, but they require an effective total cross section. The ENDF-6 format for charged-particle data does not define a total cross section because of the singularity in the elastic contribution. However, for application purposes, a reasonable definition of an effective total cross section is that it is the sum of all the partial reaction cross sections including the cross section obtained for the truncated elastic scattering reaction. This is the method that MATXSR uses to compute the quantities PTOTO, DTOTO, TTOTO, etc.

5. Delayed-Neutron Data. Delayed-neutron yields, spectra, and decay constants by time group are required by reactor kinetics codes. The previous CCCC format for these data was DLAYXS (see the CCCCR chapter of this manual). Because of the generalization inherent in the MATXS format, these data can be added without any changes in the structure of the file. This option has not yet

been added to MATXSR, and the procedures used will be described in a future version of this report.

6. Decay-Photon and Decay-Heat Data. Many nuclear reactions leave radioactive products. These products may be simple daughter isotopes that decay in a few steps to a stable final state with the emission of a few photons and electrons (or heat), or they may be a complex array of fission product isotopes that emit a complex spectrum of photons and electrons (or heat) showing many time constants. Procedures to store these data in MATXS libraries will be included in a future version of this report.

E. User Input

The user input specifications below were copied from the comment cards at the beginning of the MATXSR module. It is always a good idea to check the comment cards in the current version of the code for possible changes.

```
*---INPUT SPECIFICATIONS (FREE FORMAT)-------------------------*
*                                                             *
* CARD 1 UNITS                                                *
*   NGEN1     INPUT UNIT FOR DATA FROM GROUPR                 *
*   NGEN2     INPUT UNIT FOR DATA FROM GAMINR                 *
*   NMATX     OUTPUT UNIT FOR MATXS                           *
*             (OUTPUT UNIT IS ALWAYS BINARY)                  *
*   NGEN3     INCIDENT PROTON DATA FROM GROUPR (DEFAULT=0)    *
*   NGEN4     INCIDENT DEUTERON DATA FROM GROUPR (DEFAULT=0)  *
*   NGEN5     INCIDENT TRITON DATA FROM GROUPR (DEFAULT=0)    *
*   NGEN6     INCIDENT HE3 DATA FROM GROUPR (DEFAULT=0)       *
*   NGEN7     INCIDENT ALPHA DATA FROM GROUPR (DEFAULT=0)     *
* CARD 2 USER IDENTIFICATION                                  *
*   LPRINT    0/1 MEANS NO PRINT/PRINT                        *
*   IVERS     FILE VERSION NUMBER (DEFAULT=0)                 *
*   HUSE      USER ID (UP TO 16 CHARACTERS, DELIMITED BY *,   *
*             ENDED BY /) (DEFAULT=BLANK)                     *
* CARD 3 FILE CONTROL                                         *
*   NPART     NUMBER OF PARTICLES FOR WHICH GROUP             *
*                STRUCTURES ARE GIVEN                         *
*   NTYPE     NUMBER OF DATA TYPES IN SET                     *
*   NHOLL     NUMBER OF CARDS TO BE READ FOR HOLLERITH        *
*             IDENTIFICATION RECORD.                          *
*   NMAT      NUMBER OF MATERIALS DESIRED                     *
* CARD 4 SET HOLLERITH IDENTIFICATION                         *
*   HSETID    HOLLERITH IDENTIFICATION OF SET                 *
*             (EACH LINE CAN BE UP TO 72 CHARACTERS,          *
*             DELIMITED WITH *, ENDED BY /)                   *
* CARD 5 PARTICLE IDENTIFIERS                                 *
*   HPART     HOLLERITH IDENTIFIERS FOR PARTICLES             *
*             (UP TO 8 CHARACTERS EACH)                       *
```

```
* CARD 6 ENERGY GROUPS                                              *
*   NGRP       NUMBER OF GROUPS FOR EACH PARTICLE                   *
* CARD 7 DATA TYPE IDENTIFIERS                                       *
*   HTYPE      HOLLERITH IDENTIFIERS FOR DATA TYPES                 *
*              (UP TO 8 CHARACTERS EACH)                            *
* CARD 8 INPUT PARTICLE IDS                                          *
*   JINP      INPUT PARTICLE ID FOR EACH DATA TYPE                  *
* CARD 9 OUTPUT PARTICLE IDS                                         *
*   JOUTP     OUTPUT PARTICLE ID FOR EACH DATA TYPE                 *
* CARD 10 MATERIAL DATA (ONE CARD PER MATERIAL)                      *
*   HMAT       HOLLERITH MATERIAL IDENTIFIER                        *
*              (UP TO 8 CHARACTERS EACH)                            *
*   MATNO      INTEGER MATERIAL IDENTIFIER                          *
*              (ENDF/B MAT NUMBER)                                  *
*   MATGG      MAT NUMBER FOR PHOTOATOMIC DATA                      *
*              (DEFAULT=100*(MATNO/100) AS IN ENDF-6)               *
*                                                                   *
*********************************************************************
```

Card 1 is used to specify the units for the input GENDF tapes and the output MATXS library. As usual, the sign of a GENDF unit number is used to determine its mode; negative numbers mean binary, and positive numbers mean coded (that is, ASCII, EBCDIC, etc.). The output file is always binary, and its sign is ignored. The most common MATXSR runs are for neutron and photon data only. In these cases, card 1 can be truncated after the **NMATX**. If incident charged-particle GENDF files are available, any of the units **NGEN3** through **NGEN7** can be assigned.

Card 2 is used to control the MATXSR print option **IPRINT** and to provide the information for the MATXS "User Identification" record. An example for Card 2 follows:

```
0  17  *T2 LANL NJOY*/  .
```

The delimiting * characters and the terminating / are required (see the description of free-form input in the NJOY section of this report). Card 3 is used to input the counts for the "File Control" record and to tell the user input routine how many quantities to read in subsequent input cards. Card 4 is repeated **NHOLL** times to give the Hollerith description for the library. Each line must be delimited by * characters and terminated by /. For example,

```
* MATXS17                                  27 FEB 91 */
* 69-GROUP THERMAL LIBRARY FROM ENDF/B-VI            */
*   THIS LIBRARY INCLUDES NEUTRON, PHOTON, AND       */
*   THERMAL SCATTERING DATA FOR 133 MATERIALS.       */
```

The lines are written out on the MATXS file as given, except that the maximum length for each line is 72 characters.

Card 5 is used to read in the Hollerith names for the **NPART** particles for this library. The standard names for the particles were given above in Table 1. The number of groups desired for each particle are given on card 6. The data-type names (see Table 2) are given on Card 7. Cards 8 and 9 are used to specify the input and output particles for each data type. The user should take care that the number of groups given for each particle is consistent with the input data, and that the particle assignments to data type names (**JINP** and **JOUTP**) are consistent with the names. The code does not check.

Card 10 is repeated **NMAT** times to specify the materials to be written out on the library. The rules for constructing the material names **HMAT** were given in Section B. The **MATNO** parameter is the ENDF MAT number for this material as used on the input GENDF files. For ENDF/B-VI, the MAT number is the same for all sublibraries (that is, for incident neutrons, photons, protons, etc.), and only one value is needed to specify the desired material. However, photoatomic data are atomic in character, and the MAT numbers always refer to the element. For example, MAT=2600 for the photoatomic data of iron. MATXSR reads a second MAT number field, **MATGG**, for the photoatomic data. Its default value is given by

```
100*(MATNO/100)  ,
```

and it is, therefore, not usually needed. The normal form for card 10 would be as follows:

```
FE56  2631/  .
```

However, the photoatomic libraries for earlier versions of the ENDF/B files used MAT numbers like 26 for elements. The **MATGG** parameter can be used to process data from these older libraries. Note that delimiting * characters are not required for Hollerith names that are single words and start with a letter.

MATXSR always processes all submaterials found on all the nonzero GENDF units **NGEN1** through **NGEN7**. It processes every temperature and σ_0 value found. It processes (almost) every reaction cross section and matrix found, and it processes every Legendre order given. The only way to control the contents of the MATXS library is through the input to GROUPR or GAMINR and by the materials included when building the input GENDF files. A convenient way to handle this task is to assemble the results of a number of single-material GROUPR runs into composite GENDF tapes using MODER.

F. Coding Details

The main container array for dynamic storage allocation is defined by the statement "COMMON/MSTORE/A(35000)," and its length is specified in the statement "ISIZA=35000." It may be necessary to increase this allocation if very large group structures are used. The code will issue error messages when more storage is needed. The dynamic array DUM is allocated with two words of storage. If the resulting pointer LO is even, one of these two words is released. This assures that the first real array reserved will have an odd pointer, as required for proper word alignment on short-word machines.

The values of MULT and NCW change between short-word machines (32 bits) and long-word machines (60–64 bits). Also note that all Hollerith variables (which have "h" as their initial letter) are implicitly REAL*8 double-precision variables on short-word machines. The role of MULT is discussed in the CCCCR section of this report. It is used for calculating pointers and the lengths of arrays for dynamic arrays and common blocks that contain mixtures of Hollerith, real, and integer variables. The parameter NCW contains the number of Hollerith characters per word for the target machine (either 4 or 8). It is used by FREE (see the NJOY section of this report).

The next step in subroutine MATXSR is to read in the unit numbers for the input and output files and to open the files. Note that the sign for NMATX is ignored. It will always be binary. Several binary scratch files are also opened. Next, the user input for the "File Identification" record is read (see card 2 in the user input description), and the CMATXS routine is called to read the input GENDF files and construct the output MATXS file. When CMATXS returns, the main routine checks whether a listing of the new library is desired (LPRINT>1). Then, it calls INDEX to prepare an index listing summarizing all the materials, submaterials, data types, and reactions given in the new library. Finally, MATXSR closes the files, writes a summary, and returns to the main loop over modules in the NJOY program.

Subroutine CMATXS starts by defining three constants:

NSUBMX=100 is the maximum number of submaterials allowed for a material, including data types, temperatures, and background cross sections;

MAXORD=5 is the maximum Legendre order (that is, P_5 allowed for group-to-group matrices; and

MAXW=5000 is the maximum size for vector block and matrix subblock records on the MATXS file.

It continues by calling RUINM to read the user's input (see the "User Input" section

above), and then it calls MTXDAT to read the input GENDF data and write it to the scratch tapes used later in this subroutine.

The first step in MTXDAT is to reserve dynamic storage space for the NPART group structures. Note that the number of words reserved is forced to be even in order to avoid possible word-alignment problems on short-word machines. With the current version of GROUPR, all particles are assumed to use the same group structure. It is only necessary to look through the unit numbers from input card 1 and find the first one that contains data. The group structures are then read in from MF1,MT451, reversed to have the conventional decreasing-energy order, and stored into the GRUP array. The subroutine now reserves space for the "Material Control" record (see MATC and IMATC) and begins the material loop (see DO 700 IM=1,NMAT). Note how the Hollerith material name is loaded into the "Material Control" record storage area by making use of the MULT parameter and the equivalence between the arrays A and HA.

The first loop inside the material loop is the data-type loop (see DO 600 I=1,NTYPE). The same MAT number is used for all particles, but a different one is used for photoatomic data. The input unit to be used depends on the identity of the incident particle. Since these comparisons use constants like HPROT with values like "6HP ," it is important to use the standard names in the user input (see Table 1). The identity of the outgoing particle for the data type determines the MF number for the matrices on the GENDF tape, MFM. As described in the GROUPR section of this report, the MF assignments are as follows:

MF value	Outgoing Particle
6	neutrons
16	photons
21	protons
22	deuterons
23	tritons
24	^{3}He particles
25	alphas
26	photoatomic data

The MF number for the vector data (MFV) is 3, except for the photoatomic case, where it is 23.

The next step is to search for the desired material IMAT on the current input tape. When the first occurrence of IMAT has been found, CMATXS proceeds to copy the data for the first temperature to the two scratch files NSCR and IREF. File IREF

will be used for the higher temperature and σ_0 values. At the same time, MTXDAT counts the number of one-dimensional reactions, NONED, and the number of two-dimensional reactions, NTWOD, for each value of the background cross section, σ_0. When the scratch tapes are complete, the subroutine starts a loop over σ_0 for this material and temperature (this loop goes through statement number 440), and it calls VECIN, VECOUT, and MATRIX to process the vector and matrix data for this σ_0. As each σ_0 is finished, it loads the parameters for the current submaterial into the "Material Control" record and jumps back to statement number 440 to continue the loop. When all the σ_0 values for this temperature have been processed, it jumps back to statement number 300 to get the next temperature. Finally, it reaches statement number 600 and goes back through the entire process again for the next data type. When the "DO 600" loop exits, the entire "Material Control' block has been filled in, and the subroutine writes the record to scratch file NSCRT6.

At this point, the processing for material IMAT is complete. The "DO 700" loop continues until all the requested materials have been found and processed.

Subroutine FINDG is used to search GENDF file NTAPE for a specified section (MAT, MF, MT). If it does not find the requested section, a fatal error message is issued. Materials (MAT) do not have to be in order on a GENDF tape. Therefore, this routine simply reads forward until it comes to the first record matching the requested MAT, MF, and MT values. If the section is not found, it rewinds once and searches again. When the section is found, it backspaces by one record so that the next record read will be the desired record.

Subroutine HNAME is used to construct a Hollerith reaction name HREACT from an ENDF MT number (MT), an ENDF "LR flag" (LR), and an incident particle name (HP). The definitions of the names were discussed above in connection with Tables 3–11, and the actual MT numbers and strings used are given in data statements in this subroutine. If no preset name is found for an MT number, a default name of the form MTnnn is constructed.

Subroutine VECIN reads the cross section vectors for one submaterial (that is, one data type, one temperature, and one σ_0). It starts by clearing some flags that are used to detect the occurrence of particular reactions. For example, K107 is set if the (n,α) reaction (MT107) is found. Dynamic storage arrays are reserved for the vector control and vector data arrays (see VCON, IVCON, VDAT, and IVDAT). Note that there must be enough container memory available to hold all the vector data (NING*N1D words). The routine now starts up a loop over sections on the input scratch tape (the loop goes through statement number 115). The sections that it processes are determined by the value of MFD, which will be 3, except for

photoatomic data, when it will be 23. Delayed fission χ is a special case; it will be found in the section with an MT value of MFD+2. As each interesting section is found, the Hollerith name for the reaction is constructed, and the location of the reaction in VDAT is computed. When the total cross section is found (MT=1), names and locations are set up for the P_ℓ components of the weight function (for example, NWT0 and NWT1) and for the P_ℓ components of the total cross section (NTOT0 and NTOT1). If appropriate, the first letter of these names might be P, D, etc., depending on the incident particle. When the total photoatomic cross section is found (MT=501), names and locations are defined for the γ weight and the P_0 cross section (GWT0 and GTOT0). When the charged-particle elastic cross section is found for particle x (MT2), names and locations are set up for the weight function xWT0 and the effective charged-particle total cross section xTOT0. Once the name and location have been selected, the code reads the data for all groups from file NSCR and stores them at the selected location. If this submaterial corresponds to a higher temperature or σ_0, the corresponding data is also read from file IREF and subtracted from the NSCR data. As a result, self-shielding information in the MATXS file is given as differences rather than f-factors as in the previous CCCC self-shielding file, BRKOXS. The delayed-neutron spectra are read from MF5,MT455; the spectra from the six time groups on the GENDF tape are added to obtain a single spectrum χ_d.

There is a special feature in VECIN for incident charged-particle reactions. Although discrete-ordinates transport codes require a total cross section, it is not actually possible to compute a total cross section for charged-particle reactions because Coulomb scattering is singular for straight-ahead scattering (at least in the absence of electronic screening). The solution to this problem used by GROUPR is to exclude a range of forward angles from the charged-particle elastic matrix (this angular range is handled by continuous-slowing-down theory in the application codes). Therefore, MATXSR can construct an effective total cross section by adding all the reaction cross sections found to this truncated elastic cross section. However, it must be careful to watch for possibly redundant cross sections that must be omitted from the sum; for example, if the total (x,p) cross section (MT103) is present for incident particle x, the discrete-level (x,p_n) cross sections and the (x,p) continuum reactions (MT600–649) must be omitted from the sum. This is the function of flags like K103.

Subroutine VECOUT thins the vector data loaded by VECIN by removing unneeded leading or trailing zero from each reaction. These zeros can arise from thresholds and upper limits for thermal-range reactions. In addition, since data for the higher temperatures and σ_0 values are stored as differences, zero (or very

small) elements can imply that there is no significant temperature or σ_0 dependence. The subroutine simply steps through the VDAT data block to find the band limits NFG and NLG for each reaction, to squeeze out the zeros, and to record the band limits in the vector control block VCON. When all of the N1D reactions have been processed, the resulting vector data is written out as a series of "Vector Block" records on scratch tape NSCRT3 (see the block of coding starting at statement number 290). The subblocking is controlled by MAXW (normally 500 words). The code just steps through the cross section bands until the next reaction will cause the sum to exceed MAXW. It then writes these reactions to the scratch file, and it repeats the process for the next group of reactions. Each record will have fewer than MAXW words, and the minimum number of records will be produced within the restriction that no reaction is broken up between records. The last step in the subroutine is to write the "Vector Control" record for this submaterial to scratch file NSRT2.

Subroutine MATRIX reads the group-to-group matrix data for this submaterial, converts it to MATXS format, and writes the results to a scratch file. The dynamic storage arrays used include MCON to accumulate matrix control information, CDAT for the low-energy constant spectrum data (if any), JGLL for banding information, and MDAT for the actual matrix data. The main loop is over matrix reaction types (see "DO 280 I=1,N2D"). The head card for the reaction is read in from NSCR, and for higher temperatures and σ_0 values, from IREF also. Subroutine HNAME is called to construct the Hollerith reaction name, subroutine BAND is called to read through the data and determine the banding parameters used to remove excess zero elements, and subroutine SHUFL is called to read through the input scratch file again and load the data into memory in its final compact form.

Subroutine BAND is fairly simple. It loops over all the groups for this reaction on NSCR and/or IREF (the loop goes through statement number 110). When the loop is finished, JGLLO(I) and JGLHI(I) contain lowest and highest initial-group indices found for each final group I. Note that the actual cross section for a matrix element (or the cross section difference when IREF is being used) is not checked; it is assumed that GROUPR has done a good job removing excess zeros. This should be improved in a future version. Before returning, these limiting group indices are used to compute the MATXS parameters JBAND and IJJ, and to insert them into the accumulating matrix control block (see IMCON).

Subroutine SHUFL starts by initializing a loop through statement number 100 that will be used to produce "Matrix Sub-Block" records. Inside this loop, the code sums the bandwidths found by BAND to find out how many final-energy group

bands can fit in a subblock with length less than or equal to MAXW words (see the DO 101 loop). The result is a pair of group indices IO and IMAX that define the range of groups to be included in the subblock. SHUFL now makes a pass through scratch file NSCR, and also through scratch file IREF for higher temperatures and σ_0 values. It stores away the constant subblock data (IG=0 for the constant spectrum and IG2LO=0 for the production cross section), if found. For each group-to-group element, it computes the final group index JG2. If the group is in the allowed range for this subblock, it computes the output location NOLOC and stores the element in memory. For higher temperatures and σ_0 values, it subtracts the base value found on file IREF. (This step is normally needed for neutron elastic scattering only.) When the group loops for NSCR and IREF are complete, the subroutine writes out the "Matrix Sub-Block" record on a scratch file, resets the NG2Z parameter to the top group of the subblock, and continues the loop through statement number 100 to produce the rest of the subblocks for this reaction. The resulting series of subblock records on the scratch file are all less than MAXW words in length, and the number of records is minimal within the restriction that bands are not split between records.

Returning to MATRIX, data already in memory are used to complete the "Matrix Control" record, which is written out onto NSCRT2. If no matrix data were found by BAND and SHUFL, a dummy matrix control record is written on NSCRT2 and a dummy matrix data record is written on NSCRT3. Finally, the constant subblock data saved in SHUFL (if any) is written to NSCRT3.

Subroutine PMATXS is a simple routine that reads through an input file NIN in MATXS format and prepares an interpreted listing on NOUT. This subroutine is also available in the auxiliary code BBC.

Subroutine INDEX reads an input file NIN in MATXS format and prepares an index listing that gives the materials, data types, temperatures, background cross sections, vector reaction types, and matrix reaction types found on the library. This subroutine is also available in BBC.

G. Error Messages

The fatal-error and warning messages generated by MATXSR are given below, along with suggested actions to alleviate the problem.

ERROR IN MTXDAT***INPUT ERROR (NIN=0)

No input GENDF tape has been given. Check Card 1 of the user input.

MESSAGE FROM MTXDAT---NO DATA FOR ITYPE=nn MAT=nnnn ITEMP n

The input GENDF tapes do not have to contain data for all the materials requested by the user with the HMAT values. This diagnostic message warns the user that some potential materials were not found. In some cases, the user will want to modify the input data if the warning is a surprise!

ERROR IN MTXDAT***TOO MANY SUBMATERIALS

The code is currently limited to 100 submaterials per material. See NSUBMX=100 in CMATXS.

ERROR IN FINDG***MAT OR MF OR MT LE 0 NOT ALLOWED

Subroutine FINDG has been asked to search for an illegal section on the GENDF tape.

ERROR IN FINDG***MAT=nnnn MF=nn MT=nnn NOT ON TAPE

The requested section was not found on the input GENDF tape. Check that the correct tape was mounted.

ERROR IN BAND***INPUT TOO LARGE

There is not enough room in memory to read the matrix record for one group from the GENDF tape. The only solution is to increase the size of the main container array. See COMMON/MSTORE/A(35000) and ISIZA=35000 at the beginning of the MATXSR routine.

ERROR IN SHUFL***INPUT TOO LARGE

See comments above.

ERROR IN INDEX***STORAGE EXCEEDED

This error should not occur for the version of INDEX that is incorporated into MATXSR because the code was able to write the records in the first place!

ERROR IN LST1IO***STORAGE EXCEEDED

See comments above.

H. References

1. B. M. Carmichael, "Standard Interface Files and Procedures for Reactor Physics Codes, Version III," Los Alamos Scientific Laboratory report LA-5486-MS (February 1974).

2. R. Douglas O'Dell, "Standard Interface Files and Procedures for Reactor Physics Codes, Version IV," Los Alamos Scientific Laboratory report LA-6941-MS (September 1977).

3. R. E. MacFarlane, "TRANSX-CTR: A Code for Interfacing MATXS Cross-Section Libraries to Nuclear Transport Codes for Fusion Systems Analysis," Los Alamos National Laboratory report LA-9863-MS (February 1984).

4. R. E. MacFarlane, "TRANSX 2: A Code for Interfacing MATXS Cross-Section Libraries to Nuclear Transport Codes," Los Alamos National Laboratory report LA-12312-MS (July 1992).

5. K. D. Lathrop, "DTF-IV, A FORTRAN Program for Solving the Multigroup Transport Equation with Anisotropic Scattering," Los Alamos Scientific Laboratory report LA-3373 (November 1965).

6. R. E. MacFarlane, "Energy Balance of ENDF/B-V," *Trans. Am. Nucl. Soc.* 33, 681 (1979). See also R. E. MacFarlane, "Energy Balance of ENDF/B-V.2," Minutes of the Cross Section Evaluation Working Group (available from the National Nuclear Data Center, Brookhaven National Laboratory, Upton, NY) (May 1984).

7. Y. Farawila, Y. Gohar, and C. Maynard, "KAOS/LIB-V: A Library of Nuclear Response Functions Generated by KAOS-V Code From ENDF/B-V and Other Data Files," Argonne National Laboratory report ANL/FPP/TM-241 (April 1989).

8. I. I. Bondarenko, Ed., *Group Constants for Nuclear Reactor Calculations* (Consultants Bureau, New York, 1964).

XVI. RESXSR

In thermal reactor systems, resonance self-shielding in the near epithermal range (*e.g.,* 4–200 eV) is not usually represented well by the simple Bondarenko model.[1] The normal form of this model as used by the TRANSX code[2] assumes that all the absorber resonances are narrow with respect to the energy lost in neutron scattering. In this near epithermal range, many resonances have widths that violate this assumption. A number of ways exist to try to compensate for this problem, one example being intermediate-resonance theory as discussed in "WIMSR," Chapter XIX of this manual. The GROUPR flux calculator can also be used to compensate for wide and intermediate resonances, but only for homogeneous systems or idealized reactor cells.

One way to solve this problem accurately is to move the flux calculation into TRANSX, where full information on material mixtures and geometry can be made available. To test this concept, an experimental version of TRANSX was created that could do a full pointwise flux solution using collision probabilities for cylindrical systems with an arbitrary number and arrangement of shells. It could then take the computed flux by region and generate new, accurate, self-shielded cross sections for the groups in the near epithermal range.

Of course, this experimental code required pointwise cross sections in the epithermal range to do these calculations. Since TRANSX uses CCCC standard interface files[3] for all other communication to the outside world, it was decided to define an interface format for resonance-region pointwise cross sections, or RESXS. This RESXSR module was written to produce the RESXS file from PENDF tapes produced using the other modules of NJOY.

This module and the RESXS format may be of use for other applications besides TRANSX.

A. Method

For each material, the data for elastic (MT=2), fission (MT=18), if present, and capture (MT=102) for the desired energy range and for all the desired temperatures are read in and interpolated onto a single union grid. This large set of cross sections is then thinned down using an input tolerance EPS. Using a union grid for all reactions and temperatures makes temperature interpolation and reconstruction of the total cross section easy.

These data are then written out in the specially defined RESXS format. The key to this format is the "Cross Section Block." For each incident energy, it is

necessary to give two or three cross sections for each temperature. If all the many energy points were given in one record, the record could be many thousands of words in length. Therefore, the cross section data are broken up into a set of blocks, where each block contains NBLOK records, except the last block can be shorter.

B. RESXS Format Specification

The specification for the RESXS format follows in the standard CCCC style.[3] These card images were copied from the comment cards at the beginning of the RESXSR source.

```
C*************************************************************************
C               PROPOSED 09/24/90                                        -
C                                                                        -
CF          RESXS                                                        -
CE          RESONANCE CROSS SECTION FILE                                 -
C                                                                        -
CM                      THIS FILE CONTAINS POINTWISE CROSS               -
CM                      SECTIONS FOR THE EPITHERMAL RESONANCE            -
CM                      RANGE TO BE USED FOR HYPER-FINE FLUX             -
CM                      CALCULATIONS.  ELASTIC, FISSION, AND             -
CM                      CAPTURE CROSS SECTIONS ARE GIVEN VS              -
CM                      TEMPERATURE.  LINEAR INTERPOLATION IS            -
CM                      ASSUMED.                                         -
C                                                                        -
CM          FORMATS GIVEN ARE FOR FILE EXCHANGE ONLY                     -
C                                                                        -
C*************************************************************************
C
C
C-------------------------------------------------------------------------
CS          FILE STRUCTURE                                               -
CS                                                                       -
CS              RECORD TYPE                          PRESENT IF          -
CS              ==============================       ==========          -
CS              FILE IDENTIFICATION                    ALWAYS            -
CS              FILE CONTROL                           ALWAYS            -
CS              SET HOLLERITH IDENTIFICATION           ALWAYS            -
CS              FILE DATA                              ALWAYS            -
CS                                                                       -
CS   *************(REPEAT FOR ALL MATERIALS)                            -
CS   *          MATERIAL CONTROL                       ALWAYS            -
CS   *                                                                   -
CS   * ************(REPEAT FOR ALL CROSS SECTION BLOCKS)                -
CS   * *         CROSS SECTION BLOCK                   ALWAYS            -
CS   * ************                                                      -
CS   **************                                                      -
C                                                                        -
C-------------------------------------------------------------------------
```

```
C
C
C-----------------------------------------------------------------------
CR           FILE IDENTIFICATION                                       -
C                                                                      -
CL    HNAME,(HUSE(I),I=1,2),IVERS                                      -
C                                                                      -
CW    1+3*MULT                                                         -
C                                                                      -
CB    FORMAT(4H OV ,A8,1H*,2A8,1H*,I6)                                 -
C                                                                      -
CD    HNAME          HOLLERITH FILE NAME  - RESXS -  (A8)              -
CD    HUSE           HOLLERITH USER IDENTIFICATION    (A8)            -
CD    IVERS          FILE VERSION NUMBER                              -
CD    MULT           DOUBLE PRECISION PARAMETER                       -
CD                        1- A8 WORD IS SINGLE WORD                   -
CD                        2- A8 WORD IS DOUBLE PRECISION WORD         -
C                                                                      -
C-----------------------------------------------------------------------
C
C
C-----------------------------------------------------------------------
CR           FILE CONTROL                                              -
C                                                                      -
CL    EFIRST,ELAST,NHOLL,NMAT,NBLOK
C                                                                      -
CW    5                                                                -
C                                                                      -
CB    FORMAT(4H 1D ,2I6)                                               -
C                                                                      -
CD    EFIRST       LOWEST ENERGY ON FILE (EV)                          -
CD    ELAST        HIGHEST ENERGY ON FILE (EV)                         -
CD    NHOLL        NUMBER OF WORDS IN SET HOLLERITH                    -
CD                     IDENTIFICATION RECORD                          -
CD    NMAT         NUMBER OF MATERIALS ON FILE                         -
CD    NBLOK        ENERGY BLOCKING FACTOR                              -
C                                                                      -
C-----------------------------------------------------------------------
C
C
C-----------------------------------------------------------------------
CR           SET HOLLERITH IDENTIFICATION                              -
C                                                                      -
CL    (HSETID(I),I=1,NHOLL)                                            -
C                                                                      -
CW    NHOLL*MULT                                                       -
C                                                                      -
CB    FORMAT(4H 2D ,8A8/(9A8))                                         -
C                                                                      -
CD    HSETID       HOLLERITH IDENTIFICATION OF SET (A8)                -
CD                  (TO BE EDITED OUT 72 CHARACTERS PER LINE)         -
C                                                                      -
C-----------------------------------------------------------------------
```

```
C
C
C-----------------------------------------------------------------------
CR          FILE DATA                                                  -
C                                                                      -
CL    (HMATN(I),I=1,NMAT),(NTEMP(I),I=1,NMAT),(LOCM(I),I=1,NMAT)       -
C                                                                      -
CW    (MULT+2)*NMAT                                                    -
C                                                                      -
CB    FORMAT(4H 3D ,8A8/(9A8))       HMATN                             -
CB    FORMAT(12I6)                   NTEMP,LOCM                        -
C                                                                      -
CD    HMATN(I)    HOLLERITH IDENTIFICATION FOR MATERIAL I              -
CD    NTEMP(I)    NUMBER OF TEMPERATURES FOR MATERIAL I                -
CD    LOCM(I)     LOCATION OF MATERIAL I                               -
C                                                                      -
C-----------------------------------------------------------------------
C
C
C-----------------------------------------------------------------------
CR          MATERIAL CONTROL                                           -
C                                                                      -
CL    HMAT,AMASS,(TEMP(I),I=1,NTEMP),NREAC,NENER                       -
C                                                                      -
CW    MULT+3+NTEMP                                                     -
C                                                                      -
CB    FORMAT(4H 6D ,A8,1H*,1P1E12.5/(6E12.5))     HMAT,TEMP            -
CB    FORMAT(2I6)                                 NENER,BLOK           -
C                                                                      -
CD    HMAT        HOLLERITH MATERIAL IDENTIFIER                        -
CD    AMASS       ATOMIC WEIGHT RATIO                                  -
CD    TEMP        TEMPERATURE VALUES FOR THIS MATERIAL                 -
CD    NREAC       NUMBER OF REACTIONS FOR THIS MATERIAL                -
CD                  (3 FOR FISSIONABLE, 2 FOR NONFISSIONABLE)          -
CD    NENER       NUMBER OF ENERGIES FOR THIS MATERIAL                 -
C                                                                      -
C-----------------------------------------------------------------------
C
C
C-----------------------------------------------------------------------
CR          CROSS SECTION BLOCK                                        -
C                                                                      -
CL    (XSB(I),I=1,IMAX)                                                -
C                                                                      -
CC    IMAX=3*NTEMP*(NUMBER OF ENERGIES IN THE BLOCK)                   -
C                                                                      -
CW    IMAX                                                             -
C                                                                      -
CB    FORMAT(4H 8D ,1P5E12.5/(6E12.5))                                 -
C                                                                      -
CD    XSB(I)      DATA FOR A BLOCK OF NBLOK OR FEWER POINT ENERGY      -
CD                VALUES.  THE DATA VALUES GIVEN FOR EACH ENERGY       -
CD                ARE NELAS, NFIS, AND NG AT TEMP(1), FOLLOWED BY      -
```

```
CD              NELAS, NFIS, AND NG AT TEMP(2), AND SO ON.          -
C                                                                   -
C-------------------------------------------------------------------
```

C. User Input

The following input specifications were taken from the comment cards at the beginning of the RESXSR source. It is always a good idea to check the comment cards in the current version in case there have been changes.

```
C     *  USER INPUT                                                 *
C     *                                                             *
C     *   CARD 1   UNITS                                           *
C     *      NOUT     OUTPUT UNIT                                  *
C     *                                                             *
C     *   CARD 2   CONTROL                                         *
C     *      NMAT     NUMBER OF MATERIALS                          *
C     *      MAXT     MAX. NUMBER OF TEMPERATURES                  *
C     *      NHOLL    NUMBER OF LINES OF DESCRIPTIVE COMMENTS      *
C     *      EFIRST   LOWER ENERGY LIMIT (EV)                      *
C     *      ELAST    UPPER ENERGY LIMIT                           *
C     *      EPS      THINNING TOLERANCE                           *
C     *                                                             *
C     *   CARD 3   USER ID                                         *
C     *      HUSE     HOLLERITH USER IDENTIFICATION (UP TO 16 CHARS)*
C     *      IVERS    FILE VERSION NUMBER                          *
C     *                                                             *
C     *   CARD 4   DESCRIPTIVE DATA (REPEAT NHOLL TIMES)           *
C     *      HOLL     LINE OF HOLLERITH DATA (72 CHARS MAX)        *
C     *                                                             *
C     *   CARD 5   MATERIAL SPECIFICATIONS (REPEAT NMAT TIMES)     *
C     *      HMAT     HOLLERITH NAME FOR MATERIAL (UP TO 8 CHARS)  *
C     *      MAT      ENDF MAT NUMBER FOR MATERIAL                 *
C     *      UNIT     NJOY UNIT NUMBER FOR PENDF DATA              *
```

The only difficulty in constructing the input file for RESXSR is in choosing EFIRST, ELAST, and EPS. The goal is to get a set that can be used to generate reasonably good fluxes and cross sections without being too expensive. Also, the energy limits should be consistent with the group bounds that will be used for the multigroup part of the calculation. However, the upper energy limit should be enough above the highest group to be treated with this method that "Placek" oscillations from the discontinuity in the source from higher energies have some chance to die out. It is possible that the values of EPS could be somewhat larger than the value used in RECONR and BROADR to attain additional economy. In practice, it may be necessary to iterate a few times to get a good compromise.

D. References

1. I. I. Bondarenko, Ed., *Group Constants for Nuclear Reactor Calculations* (Consultants Bureau, New York, 1964).

2. Robert E. MacFarlane, "TRANSX2: A Code for Interfacing MATXS Cross-Section Libraries to Nuclear Transport Codes," Los Alamos National Laboratory report LA-12312-MS (July 1992).

3. R. Douglas O'Dell, "Standard Interface Files and Procedures for Reactor Physics Codes, Version IV," Los Alamos Scientific Laboratory report LA-6941-MS (September 1977).

XVII. ACER

The ACER module prepares libraries in ACE format (A Compact ENDF) for the MCNP continuous-energy neutron-photon Monte Carlo code.[1] One of the design goals for MCNP has been to use the most detailed representation of the physics of a problem that is practical. Therefore, the ACE format has evolved to include all the details of the ENDF[2] representations for neutron and photon data. However, for sake of efficiency, the representation of data in ACE is quite different from that in ENDF. The fundamental difference is the use of random access with pointers to the various parts of the data. Other key differences include the use of union energy grids, equal-probability bins, and cumulative probability distributions. This chapter will first describe the data representations used in the ACE format, then continue with instructions on how to run the module, and finish with a more detailed description of the details of the code. This chapter is for version 91.76.

A. Energy Grids and Cross Sections

MCNP requires that all the cross sections be given on a single union energy grid suitable for linear interpolation. This was also true of its predecessor MCN,[3] and this is one of the reasons that the RECONR and BROADR modules of NJOY are also organized around union grids and linear interpolation.

Although the energy grid and cross section data on an NJOY PENDF tape are basically consistent with the requirements of MCNP, there is still one problem. Many ENDF evaluations (especially ENDF/B-VI evaluations) produce energy grids with very large numbers of points. A few examples from ENDF/B-VI are shown in Table 1. Although modern computers have very large memories, it is still often useful to reduce the size of these data sets by reducing the number of energy points in the union grid.

Table 1: Union Energy Grid Sizes for Some Evaluations from ENDF/B-V and ENDF/B-6

Evaluation	Union Grid
U-235 vers. V at 300 K and .5%	7 200
U-235 vers. VI at 300 K and .5%	49 100
U-238 vers. V at 300 K and .5%	30 900
U-238 vers. VI at 300 K and .5%	58 300

Of course, any thinning of the energy grid will result in a loss of accuracy. The goal is to control the accuracy loss and balance it against the memory requirements. This balance will vary from application to application. For example, a user doing fusion calculations may be able to drastically reduce the number of resonance points at low energies without affecting the results significantly. Similarly, a thermal-reactor designer may be able to reduce the number of energy points used above 100 to 200 eV with minimal impact on the answers.

The ACER module will ultimately provide three different thinning algorithms. First, the code can do a very crude removal of points; for example, it can remove 2 out of every 3 points for all energies between E1 and E2. This is called the "energy skip" option. It is now obsolete, and it is not recommended.

The second thinning option is "integral fraction" thinning. The idea here is to attempt to preserve the resonance integral. Two weighting functions are provided: 1/E and flat. The former is best for thermal-type problems, and the latter preserves more points in the high-energy range. The user specifies a target number of points for the final energy grid. The code uses this target number to estimate an initial thinning tolerance, and it starts moving through the energy grid and calculating the contributions to the total and capture resonance integrals from each energy panel. Panels whose contributions to both integrals are small with respect to the current tolerance are candidates for rejection. The code has additional tolerances designed to preserve major features and to preserve a reasonable minimum lethargy step; these features keep some of the points from being rejected. When the entire energy range has been scanned, the code checks the resulting number of points against the user's target. If the goal has not been reached, it doubles the tolerance and repeats the entire process. When the target has been reached, it prints out the new and original values for the resonance integrals for several subranges of the total energy range. If the errors introduced by thinning are too large, the user will have to repeat the ACER run using a larger target for the final number of energy points. An example of the printout provided with integral thinning is given below.

```
ORIGINAL GRID= 19585 WITH INTEGRALS   5.9781E+07  2.1716E+04

NEW GRID= 18809 WITH INTEGRALS   5.9782E+07  2.1720E+04

NEW GRID= 17842 WITH INTEGRALS   5.9782E+07  2.1724E+04

NEW GRID= 16227 WITH INTEGRALS   5.9782E+07  2.1735E+04

NEW GRID= 13786 WITH INTEGRALS   5.9783E+07  2.1727E+04

NEW GRID= 11033 WITH INTEGRALS   5.9785E+07  2.1762E+04
```

```
TOTAL
  8.0942E+04  5.4213E+05   0.0   0.0   0.0   0.0   0.0
  1.7327E+05  4.1284E+05   0.0   0.0   0.0   0.0   0.1
  2.5257E+05  3.0142E+05   0.0   0.0   0.0   0.1   0.1
  3.2191E+05  1.8647E+05   0.0   0.0   0.0   0.1   0.3
  4.4842E+05  5.0962E+05   0.0   0.0   0.0   0.0   0.1
  5.5888E+05  3.5549E+05   0.0   0.0   0.0   0.0   0.1
  6.5861E+05  2.0999E+05   0.0   0.0   0.0   0.1   0.3
  7.8043E+05  4.1624E+05   0.0   0.0   0.0   0.0   0.1
  1.1206E+06  9.0192E+05   0.1   0.1   0.1   0.1   0.1
  2.0000E+07  5.5945E+07   0.0   0.0   0.0   0.0   0.0

CAPTURE
  8.0942E+04  1.0141E+03   0.3   0.4   0.8  -0.5   1.5
  1.7327E+05  7.3599E+02   0.0   0.2   0.5   0.0   0.2
  2.5257E+05  5.4686E+02   0.0   0.0   0.3   0.5   1.3
  3.2191E+05  4.2149E+02   0.0   0.1   0.3   1.0   1.4
  4.4842E+05  5.9879E+02   0.0   0.1   0.2   0.4   0.7
  5.5888E+05  5.7626E+02   0.0   0.0   0.1   0.2   0.4
  6.5861E+05  5.5110E+02   0.0   0.1   0.3   0.5   0.8
  7.8043E+05  8.4225E+02   0.0   0.0   0.1   0.2   0.4
  1.1206E+06  1.3748E+03   0.0   0.0   0.0   0.0   0.1
  2.0000E+07  1.5054E+04   0.0   0.0   0.0   0.0   0.0

   861   827   825   739   983  1137   867  1262  1575  1956
```

The numbers at the ends of the first few lines of this listing are the total and capture resonance integrals computed by ACER. The sections starting with the words TOTAL and CAPTURE give the resonance integrals for a few energy ranges, and they also show the percentage change caused by thinning for each step of the process. This sample shows that the capture integral increases by as much as 1.5% after thinning to 11 000 points. If this seems too large, the user can repeat the run using a target of 15 000 points; the maximum capture error will be reduced to 1%. The last line of the listing shows the number of points remaining in each energy interval with the intervals listed horizontally. In this case, the original number of points was about 1958 for each interval. The high energy band has not been thinned much at all, but the low energy band has lost 56% of its points.

The third thinning option does not exist yet, but it will use the probability tables that we plan to introduce for the unresolved energy range (see Subsection XVII.F). It will be especially useful for the new ENDF/B-VI evaluations that have extended the unresolved energy range to much higher energies. Table 2 summarizes the old and new values for the resonance ranges for these materials. This extension has resulted in huge energy tabulations, and it is likely that "integral fraction" thinning would do too much damage in these high-energy resolved ranges for some

Table 2: Upper Energy Limit of Resolved-Resonance Energy Range for Some Evaluations from ENDF/B-V and ENDF/B-VI

Material	ENDF/B-V	ENDF/B-VI
U-235	82 eV	2.25 keV
U-238	4 keV	10 keV
Pu-239	301 eV	2 keV

type of problems. An alternative is to convert the $\sigma_t(E)$ function into a probability table $C(\sigma_t, E)$ that gives the probability of seeing a total cross section less than σ_t for a neutron of energy E. This table would be accompanied by conditional probability tables that give the values of the elastic, fission, and capture cross sections expected when a particular value for the total cross section is selected. As discussed in more detail in Section XVII.F, using these tables would preserve the values for the self-shielded cross sections when the resonances are narrow with respect to the average energy loss in scattering for the system.

The formats for storing energy grid and cross section data in an ACE library are completely described in Appendix F of the MCNP manual, but they will also be reviewed briefly here for the reader's convenience. The principal cross sections are given in the ESZ block. First, the NES energy values of the union grid are given, then the NES values of the total cross section. These are followed by the absorption cross section, elastic cross section, and average heating numbers. The cross sections for the other NTR reaction types are controlled by a set of blocks called MTR, LQR, TYR, and LSIG that contain the reaction ENDF MT numbers, the Q values, the reaction types, and pointers to the cross section data for each reaction, respectively.

The energy and cross section values from the input PENDF tape are copied onto the grid of the total cross section in subroutine `UNIONX`. This routine also handles the thinning as described above. The results are written onto a scratch tape and passed on to subroutine `ACELOD`, which reads in the cross sections and stores them into the ACE-format blocks. Note that all energy values in the ACE libraries are given in MeV. The ACE heating numbers are computed by dividing the heat production cross sections from MT301 on the PENDF tape by the corresponding total cross sections to obtain heating in MeV per reaction.

B. Two-Body Scattering Distributions

Reactions like elastic and discrete-level inelastic scattering are completely described by their reaction cross sections, Q values, and angular distributions in the center-of-mass (CM) system. The ACE locations for the cross sections and Q values were noted above. The angular distributions are stored in the AND block using a set of pointers stored in the LAND block. For easy sampling, the angular distributions are represented by 32 equally probable cosine bins (except for isotropic cases). The methods for doing this calculation in ACER were borrowed from ETOPL.[4] The calculation is driven by `TOPFIL`, which uses `PTLEG` for distributions represented using Legendre coefficients and `PTTAB` for distributions given as tabulations of scattering probability versus scattering cosine $P(\mu)$. The ENDF angular distributions are obtained from File 4 on the input ENDF tape.

The `TOPFIL` routine can also remove angular distributions for particular incident energies if they can be obtained within a given fractional tolerance `ERR` by linear interpolation using neighboring energies. This incident-energy thinning for distributions is entirely independent from the energy-grid thinning described above.

C. Secondary-Energy Distributions

In earlier versions of MCNP, and in the original MCN code, tabulated energy distributions for secondary neutrons from multibody reactions like $(n, 2n)$ or composite reactions like (n, n'_c) were represented using equally probable bins (see LAW=1 in the DLW block). This representation turned out to be poor because it didn't sample low-probability important events like those in the high-energy tails of energy distributions. The current standard representation for tabulated energy distributions is LAW=4, the "Continuous Tabular Distribution." This scheme is based on sampling from a cumulative density distribution $C(E')$, which gives the probability that the energy of the emitted particle will be less than E'. Since this probability runs from 0 to 1, it is easy to select a random number in this range, and interpolate for the corresponding value of E'. The differential density distribution $P(E')$ is also given for use in MCNP's interpolation scheme. These quantities are computed in subroutine `ACELOD` and stored into the ACE DLW block using pointers stored in the LDLW block.

As for angular distributions, subroutine `TOPFIL` thins out tabulated energy distributions for particular incident energies that can be obtained by linear interpolation within the tolerance `ERR`.

Analytic energy distribution laws, such as the LAW=7 simple Maxwellian fission spectrum, the LAW=9 evaporation spectrum, or the LAW=11 energy-dependent Watt spectrum, are also stored into the DLW and LDLW blocks. The ACE representation is a faithful image of the ENDF representation, so ACELOD simply stores the various fields into the correct location in memory.

D. Energy-Angle Distributions

A new feature of the ENDF-6 format is coupled energy-angle distributions in File 6. (There was a File 6 format available in earlier versions of the ENDF format, but it was never used. The new ENDF-6 MF6 format is different.) For neutrons, there are four different representations to be considered:

- The Kalbach law for $\sigma(E \rightarrow E')$ angular distributions as used in the ENDF/B-VI evaluations from Los Alamos;
- Legendre coefficients for $\sigma(E \rightarrow E')$ in the laboratory system as used in the ENDF/B-VI evaluations from Oak Ridge;
- Secondary-energy distributions versus laboratory scattering cosine as used in the Livermore evaluation of 4-Be-9 in ENDF/B-VI; and
- The phase-space distribution as used in the Los Alamos evaluation of the $(n, 2n)$ reaction for 1-H-2 in ENDF/B-VI.

1. Kalbach Systematics. Kalbach and Mann[5] examined a large number of experimental angular distributions for neutrons and charged particles. They noticed that each distribution could be divided into two parts: an equilibrium part symmetric in μ, and a forward-peaked preequilibrium part. The relative amount of the two parts depended on a parameter r, the preequilibrium fraction, that varied from zero for low E' to 1.0 for large E'. The shapes of the two parts of the distributions depended most directly on E'. This representation is very useful for preequilibrium statistical-model codes like GNASH,[6] because they can compute the parameter r, and all the rest of the angular information comes from simple universal functions. More specifically, Kalbach's latest work[7] says that

$$f(\mu) = \frac{a}{2\sinh(a)}\Big[\cosh(a\mu) + r\sinh(a\mu)\Big], \tag{1}$$

where a is a simple function of E, E', and B_b, the separation energy of the emitted particle from the liquid-drop model without pairing and shell terms. The

separation energies are computed by function SEP. This routine has a problem for elemental evaluations, because it needs an A value for the calculation, and it is difficult to guess which A value is most characteristic of the element. A short table is included in the function, and an "error in sep" will result if the function is called with an element that doesn't appear in the table. Similar routines appear in HEATR and GROUPR (SEPH and SEPE, respectively). A better long-range solution would be desirable.

A special sampling scheme has been developed for this case. The MCNP code already had logic to select a secondary energy E' from a distribution. The problem was to select an emission cosine μ for this E'. First, the Kalbach distribution is written in the form

$$f(\mu) = \frac{a}{2\sinh(a)}\left[(1-r)\cosh(a\mu) + re^{a\mu}\right] . \tag{2}$$

Now select a random number R_1. If $R_1 < r$, use the first distribution in Eq. (2). Select a second random number R_2, where

$$R_2 = \int_{-1}^{\mu} \frac{a\cosh(ax)}{2\sinh(a)}\,dx = \frac{\sinh(a\mu)}{2\sinh(a)} + \frac{1}{2} . \tag{3}$$

Therefore, the emission cosine is

$$\mu = \frac{1}{a}\sinh^{-1}\left[(2R_2 - 1)\sinh(a)\right] . \tag{4}$$

If $R_1 \leq r$, use the second distribution in Eq. (2). Select a random number R_2, where

$$R_2 = \int_{-1}^{\mu} \frac{ae^{ax}}{2\sinh(a)}\,dx = \frac{e^{a\mu} - e^{-a}}{e^{a} - e^{-a}} , \tag{5}$$

and emit a particle with cosine

$$\mu = \frac{1}{a}\ln\left[R_2 e^{a} + (1-R_2)e^{-a}\right] . \tag{6}$$

The ACE format for the File 6 data is similar to the LAW=4 format used for other continuous energy distributions, namely, cumulative distribution functions. To this are added tables for the preequilibrium ratio r and the Kalbach slope parameter a. The result is the LAW=44 format.

2. Laboratory Legendre or Tabulated Distributions for E to E'. This option is used in many of the new Oak Ridge evaluations, such as the isotopes of

chromium, ^{55}Mn, the isotopes of iron, the isotopes of nickel, the isotopes of copper, and the isotopes of lead. The energy-angle distribution for outgoing neutrons is given as a set of normalized emission spectra $g(E, E')$ for various incident energies E. In addition, an angular distribution is given for each $E \rightarrow E'$ as a Legendre expansion. Emission energy and angle are given in the laboratory frame.

It is difficult to sample directly from a polynomial in a Monte Carlo code. Normally, the angular distribution $f(\mu)$ is reconstructed from the parameters and converted into equal probability bins or a cumulative distribution function. Both these options are fairly bulky. However, if Kalbach systematics is a reasonable representation of reality, it is clear that the Legendre expansion is just an approximation to an underlying Kalbach shape. Therefore, ACER transforms the function $f(\mu)$ into the CM frame for $\mu = -1$ and $\mu = +1$, determines the Kalbach r and a parameters that fit these two values, and writes out the results using LAW=44 format.

The resulting representation is compact, physically realistic, and it can be sampled from easily without making any additional changes to the Monte Carlo code. However, it is an approximation to the evaluation.

For E' small with respect to E, many of the current evaluations contain unreasonable laboratory distributions that are given as symmetric functions of μ. Transformation to the CM gives a backward-peaked distribution that cannot be accepted. In these cases, r is set to zero and a is chosen to fit the average anisotropy at the limits of the μ range. ACER prints out a diagnostic message whenever this happens.

3. Laboratory Angle-Energy Distributions. The ENDF/B-VI evaluation for ^{9}Be prepared at the Lawrence Livermore National Laboratory (LLNL) uses the angle-energy option. That is, the outer loop is on incident energy E, the next loop is on laboratory scattering cosine μ, and the inner loop is on secondary energy E'. In order to sample from data in this form, the first step is to integrate over E' for each μ in order to obtain the differential angular distribution $f(E, \mu)$. This angular distribution is converted into 32 equally probable bins and stored into the ACE file using the same format used for two-body angular distributions like elastic scattering. The emission spectra for the individual μ values are normalized and stored into the file using a format called Law 67 (for ENDF File 6, Law 7). MCNP can sample from this representation as follows: for each emission, first sample from $f(\mu)$ to get an emission angle, then find the corresponding spectrum and sample from its cumulative probability distribution to get the value of E'.

4. N-Body Phase-Space Distributions. The phase-space distribution for particle i in the CM system is given by

$$P_i^{\mathrm{CM}}(\mu, E, E') = C_n \sqrt{E'} (E_i^{\max} - E')^{3n/2-4}, \tag{7}$$

where $E_i^{\max}$ is the maximum possible CM energy for particle i, μ and E' are in the CM system, and the C_n are normalization constants. The value of $E_i^{\max}$ is a fraction of the energy available in the CM:

$$E_i^{\max} = \frac{M - m_i}{M} E_a, \tag{8}$$

where M is the total mass of the n particles being treated by this law, and

$$E_a = \frac{m_T}{m_p + m_T} E + Q. \tag{9}$$

Here, m_T is the target mass, and m_p is the projectile mass. In summary, the data items required for the phase-space law are

Symbol	ENDF	Location
n	NPSX	N2 field of the MF6 CONT for LAW=6
m_i	AWI	C1 field of third card in MF1
m_p	AWP	C2 field of LAW=6 TAB1 record
m_T	AWR	C2 field of section HEAD record
M	APSX	C1 field of LAW=6 CONT record
Q	Q	C1 field of the MF3 TAB1 record

These equations are sampled with a compact numerical scheme similar to Law 4. Note that all the spectra scale with the maximum possible outgoing energy. Therefore, it is easy to construct a single normalized distribution with $E_i^{\max}$=1 with a reasonable number of $x = E'/E_i^{\max}$ points and then to construct a cumulative distribution function for it. The grid uses uniform spacing above $x = 0.10$ and log spacing below. The x grid, the probability density values $P(x)$, the cumulative densities $C(x)$, NPXS, and APSX are stored in the Law=66 format. For any given E, the cumulative distribution function is sampled with a random number between 0 and 1. The resulting x value is then multiplied by $E_i^{\max}$ to get the emitted E' value. The corresponding CM cosine value is obtained by sampling uniformly in the interval $[-1, 1]$.

The CM to lab transformation is carried out by adding the CM velocity of the initial collision to the emitted particle velocity. The appropriate equations are

$$E'_{\rm LAB} = E_{\rm CM} + E'_{\rm CM} + 2\mu_{\rm CM}\sqrt{E_{\rm CM}E'_{\rm CM}}\,, \tag{10}$$

and

$$\mu_{\rm LAB} = \frac{\sqrt{E'_{\rm CM}}\,\mu_{\rm CM} + \sqrt{E_{\rm CM}}}{\sqrt{E'_{\rm LAB}}}\,, \tag{11}$$

where the CM energy is

$$E_{\rm CM} = \frac{A}{A+1}E\,. \tag{12}$$

E. Photon Production

Earlier versions of MCNP used a very simple representation for photon production from neutron reactions. There was a single total photon production cross section on the same union grid as the neutron data, and there were 600 words of data describing the spectrum of outgoing neutrons. This table contained 20 equally likely outgoing photon energies for each of 30 incident neutron groups. This representation did not achieve the MCNP goal of providing the best possible representation of the physics of the problem. It was inadequate in representing discrete photons because their real energies were often lost, and it was inadequate in representing low-probability events from the tails of distributions. This was especially noticeable in capture events because of the high photon energies possible. It is still possible to use this representation, but it is no longer recommended. The newer "Expanded Photon Production Data" option is preferred.

1. Photon Production Cross Section. In the earlier versions of the ENDF format, photon production cross section information was given in File 13 (photon production cross sections), or as a combination of File 3 (reaction cross sections) and File 12 (photon production yields). With the ENDF-6 format, photon production can also be computed using a combination of File 3 and File 6 (product yields and energy-angle distributions).

The first step in photon production processing takes place in subroutine `CONVER`. MF12 on the ENDF tape is examined for transition probability arrays (LO=2). If they are found, they are converted into the photon yield format (LO=1). The final photon yield data is written onto a scratch tape. Next, the MF13 data is copied, and MF14 (photon angular distributions) is updated to reflect the changes made in MF12. Finally, if File 6 is present, any photon production subsections

found are converted into a special MF16 format on the scratch tape. The next step is performed in GAMSUM. The scratch tape from CONVER is used together with the input PENDF tape to calculate the sum of MF13, MF12×MF3, and MF16×MF3 for all the photon reactions on the normal union energy grid. Later, this total photon production cross section is written into the ACE GPD block in ACELOD.

2. Photon Production Matrix. The 30-by-20 photon production matrix is computed from input multigroup data. Therefore, it is necessary to run GROUPR first. This run should use the 30-group option for neutrons and a photon group structure with many groups (the CSEWG 94-group structure is normally used). The GAMOUT routine reads the multigroup data and adds up all the reactions. It then integrates through the photon groups for each neutron group and finds the equal-probability boundaries. For each of these equally probable bins, it selects a single photon energy that preserves the average energy for the bin. The results are written on a scratch tape in a special ENDF-type format and passed to ACELOD to be inserted into the GPD block.

3. Expanded Photon Production Data. This newer representation allows each discrete photon to be treated with its proper energy, and it allows for a much better representation of the spectrum of continuum photons. In the ACE representation, the MTRP block lists all the photon reactions included by ENDF MT number. Since some reactions may describe more than one photon (for example, radiative capture reactions usually describe many discrete photons), the identifier numbers are given as 1000×MT plus a photon index. Thus 102002 would stand for the second photon described under radiative capture (MT=102). Each of the NMTR photons listed in the MTRP block can have its own cross section or yield as described in the SIGP and LSIGP blocks, its own angular distribution as described in the ANDP and LANDP blocks, and its own energy distribution as described in the DLWP and LDLWP blocks. In addition, the YP block contains a list of reaction MT numbers that are needed as photon production yield multipliers.

These expanded photon production data are stored into the ACE-format blocks in ACELOD using the information written on a scratch tape by CONVER.

F. Probability Tables for the Unresolved Region

This subsection will be added in a future version of this report when the unresolved-resonance probability tables are installed in a standard release of MCNP.

G. Dosimetry Cross Sections

The ACE dosimetry library provides cross sections to be used for response functions in MCNP; the data cannot be used for actual neutron transport. The information on a dosimetry file is limited to an MTR block, which describes the reactions included in the set, an LSIG block containing pointers to the cross section data for the reactions, and the SIGD block, which contains the actual data. The format for dosimetry cross section storage is different from the format for neutron cross sections. The union grid for linear interpolation is not used; instead, the cross sections are stored with their ENDF interpolation laws. If the file mounted as the input PENDF tape is a real PENDF tape (that is, if it has been through RECONR), all the reactions are already linearized, and the interpolation information stored in the SIGD block will indicate linear interpolation for a single interpolation range. However, if the file mounted as NPEND is actually an ENDF file, the SIGD block may indicate multiple interpolation ranges with nonlinear interpolation laws.

H. Thermal Cross Sections

For energies below about 4 eV, the thermal motions of nuclei can lead to significant energy gains in neutron scattering. In addition, the binding of atoms into liquids and solids begins to affect the scattering cross section and the distribution of scattered neutrons in angle and energy. MCNP can handle thermal neutron scattering from the atoms of a free gas using internal kinematic formulas that assume a Boltzmann distribution. The bound-atom effects are treated using thermal data from ENDF evaluations stored in a special MCNP thermal library.

The ENDF format allows for several thermal processes. Thermal inelastic scattering is represented using the scattering law $S(\alpha,\beta)$, where α and β are dimensionless momentum and energy transfer parameters, respectively:

$$\sigma(E \rightarrow E',\mu) = \frac{\sigma_b}{4\pi T}\sqrt{\frac{E'}{E}}e^{-\beta/2}S(\alpha,\beta)\,, \tag{13}$$

where

$$\alpha = \frac{E' + E - 2\mu\sqrt{EE'}}{AkT}\,, \tag{14}$$

$$\beta = \frac{E' - E}{kT}\,, \tag{15}$$

and where E and E' are the incident and outgoing neutron energies, μ is the scattering cosine, T is the absolute temperature, A is the mass ratio to the neu-

tron of the scatterer, and k is Boltzmann's constant. This process occurs in all the ENDF thermal materials, such as water, heavy water, graphite, beryllium, beryllium oxide, polyethylene, benzine, and zirconium hydride.

The THERMR module of NJOY uses this equation and evaluated $S(\alpha,\beta)$ data from an ENDF-format evaluation to compute $\sigma(E \rightarrow E',\mu)$. The E' dependence of the integral over μ is computed adaptively so as to represent the function using linear interpolation within a specified tolerance. The angular distribution at each of these E' values is then calculated in a similar way, but the curve of σ vs. μ is then converted into equally probable bins (typically 8), and a discrete angle is selected for each bin that preserves the average scattering cosine for that bin. The data are written onto the PENDF tape using special MF6-like formats.

The ACESIX subroutine in ACER reads this thermal section on the input PENDF tape and converts the σ versus E' function for each of the discrete angles into equally probable bins (typically 16) and then chooses a discrete energy in each bin that preserves the average energy in that bin. The result of this process is a set of equally probable events (typically $8 \times 16 = 128$ events) in E',μ space for each incident energy. It is very easy to sample from this representation, and it is fairly compact.

However, it must be recognized that this scheme is only reasonable if each neutron undergoes several scattering events before being detected. The artificial discrete lines must be averaged out. Be careful when using this method to analyze experimental arrangements using optically thin elements and small-angle detectors. In addition, as in all equal-probability bin schemes, the wings of functions (which may be unlikely but important) are not well sampled. ACER includes a variation to partially relieve this problem: instead of equal bin weights, the pattern 1, 4, 10, 10,..., 10, 4, 1 is used (see "variable weighting" in the input instructions). This approach produces some samples fairly far out on the wings of the energy distribution. Angles are still equally weighted.

The thermal inelastic data prepared by ACESIX is loaded into the ACE blocks ITIE and ITXE by THRLOD.

The second ENDF process to consider is "coherent elastic" scattering. This process occurs in powdered crystalline materials, such as graphite, beryllium, and beryllium oxide. Bragg scattering from the crystal planes leads to jumps in the cross section vs. energy curve as scattering from each new set of planes becomes

possible. The formula for this process can be written in the following form:

$$\sigma(E,\mu) = \sigma_c \frac{\pi\hbar^2}{4MEV} \sum_{\tau\neq 0}^{\tau<\tau_{max}} f(\tau)\delta(\mu - \mu_0[\tau]) \ , \tag{16}$$

where

$$\tau_{max} = \sqrt{\frac{8ME}{\hbar^2}} \ , \tag{17}$$

and

$$\mu_0 = 1 - \frac{\hbar^2\tau^2}{4ME} \ , \tag{18}$$

and where E is the incident neutron energy, E' is the outgoing neutron energy, μ is the scattering cosine, σ_c is the characteristic coherent scattering cross section for the material, M is the target mass, V is the volume of the unit cell, τ is the radius of one of the reciprocal lattice shells, and $f(\tau)$ is the effective structure factor for that shell.

Examination of these equations shows that the angle-integrated cross section will go through a jump proportional to $f(\mu)$ when E gets large enough so that $\mu_0 = -1$ for a given value of τ. At this energy, a backward directed component of discrete-angle scattering will appear. As the energy increases, this discrete-angle line will shift toward the forward direction. It is clear that the only information that MCNP needs to represent this process in complete detail is a histogram $P(E)$ tabulated at the values of E where the cross section jumps. The cross section will then be given by $P(E)/E$. The intensity and angle of each of the discrete lines can be deduced from the sizes of the steps in $P(E)$ and the E values where the steps take place.

The $P(E)$ function is computed from $\sigma(E)$ in `ACESIX`. Subroutine `THRLOD` stores the number of Bragg edges, the Bragg energies, and the P values at the Bragg edges into the ACE-format ITCE block.

The third ENDF thermal process is incoherent elastic scattering. It occurs for hydrogeneous solids like polyethylene and zirconium hydride by virtue of the large incoherent scattering length and small coherent scattering length of hydrogen. The equation describing this process is

$$\sigma(E,\mu) = \frac{\sigma_b}{2} e^{-2EW(1-\mu)/A} \ , \tag{19}$$

where σ_b is the characteristic bound cross section, and W is the Debye-Waller integral. The THERMR module of NJOY computes the integrated cross section $\sigma(E)$ for this process and a set of equally probable cosines for each incident energy

E; it writes them onto the PENDF tape using a special format. These quantities are copied into the ITCE and ITCA blocks of the ACE format in subroutine THRLOD.

I. Photoatomic Data

Photons from direct sources and photons produced by neutron reactions are scattered and absorbed by atomic processes, producing heat at the same time. The existing MCNP "photon interaction" libraries were based on fairly old cross section data and assembled by hand.[8, 9] This version of ACER contains the beginnings of an automated capability to produce these libraries from the latest ENDF/B photoatomic data.

The cross sections for the basic photoatomic process incoherent scattering, coherent scattering, pair production, and photoelectric absorption are given on a union energy grid. Actually, the energies and the cross sections are stored as logarithms, and MCNP uses linear interpolation on them; therefore, the effective interpolation law is log-log. MCNP determines the mean-free-path to a reaction using the sum of these partial cross sections instead of a total cross section.

If the reaction is incoherent (Compton) scattering, the scattering is assumed to be given by the product of the free-electron Klein-Nishina cross section and the incoherent scattering function, $I(v)$. MCNP assumes that this function is tabulated on a given 21-point grid of v values, where v is the momentum of the recoil electron given in inverse angstroms. It is easy to extract the scattering function from the section MF26,MT504 of the photoatomic ENDF library and to interpolate the function onto the required v grid.

If the reaction is coherent (Thomson) scattering, the photon will be scattered without energy loss, and the scattering distribution is given by the Thomson cross section times the coherent form factor. The sampling scheme used in MCNP requires the coherent form factor tabulated on a predefined set of 55 v values, and it also requires integrated form factors tabulated at 55 values of v^2, where the v values are the same set used for the form factors. These values are extracted from the section MF26,MT502, and the integrals are done using the standard NJOY routine INTEGA.

Photoelectric absorption results in the emission of a complex pattern of discrete "flourescence"photons and electrons (which lead to heating) due to the cascades through the atomic levels as the atom de-excites.

The flourescence part is not coded yet. This section will be completed when the new flourescence methods have been developed and installed in MCNP. In the meantime, ambitious users may be able to meld the new cross sections computed by the methods discussed in this section with the flourescence data from the current MCNP library.

J. Type 1, Type 2, and Type 3

ACE library files come in three different types in order to allow for efficiency, portability, and history.

- Type 1 is a simple formatted file suitable for exchanging ACE libraries between different computers.
- Type 2 is a FORTRAN-77 direct-access binary file for efficient use during actual MCNP runs.
- Type 3 is a word-addressable direct-access binary file. It uses nonstandard read and write calls, and it is normally used only at Los Alamos. Type 3 is only used for the fast library—not for dosimetry, thermal, or photoatomic data.

The problem here is that Type 2 files are written with all data as real numbers (except for some of the fields in the header record). Some of these numbers represent integers, and the Type 1 format requires that these numbers be written into their fields in integer format, that is, right justified and without decimal points. For the Type 3 format, the real quantities are stored in memory and written on files as real numbers, and the integer quantities are stored and written as integers. This changing back-and-forth between integer and real versions of the same field caused major portability problems in the previous versions of ACER. While some machines (like Crays) can deduce whether a number is real or an integer by looking at the bit pattern, others (like VAX) cannot. Therefore, the new version of ACER written for NJOY 91 was designed so that all ACE-format values are stored in memory as real numbers. This means that the contents of memory can be written out in Type 2 format with no further processing. Furthermore, it is easy to read either Type 1 or Type 2 data into memory as real numbers.

In order to write the fast data in memory out in Type 1 or Type 3 format, subroutine `CHANGE` is used; it knows the type (real or integer) of every word in the ACE fast format. When writing a Type 1 file from memory, it uses subroutine `TYPEN` to write each number directly to `NOUT` using the appropriate format (I20 or

1PE20.12). When writing a Type 3 file, it goes through the ACE data stored in memory and uses TYPEN to change each field that is supposed to be an integer into a real integer. The entire memory array can then be written out as a Type 3 file using the nonstandard RDISK routine. Conversely, to read a Type 3 fast library into memory, the nonstandard subroutine RDISK is used to read in the mixed real and integer data, and then CHANGE reads through the data in memory using TYPEN to convert the integer fields into real numbers.

Since Type 3 is not used for dosimetry, thermal, or photoatomic libraries, there is no difficulty reading them into memory, and no analogs to CHANGE are required. Type 1 output for these libraries is obtained by calling TYPEN directly from the corresponding output routines DOSOUT, THROUT, and PHOOUT.

The ACER user would normally prepare libraries using either Type 1 or Type 2 output. The advantage of Type 1 is that the files can be easily moved to other machines or laboratories. The advantage of Type 2 is that it is more compact and can be used directly by MCNP with no performance penalties. At any time, ACER can be used to convert one format to another, or to make a listing of the data from any of the formats (see IOPT values from 7 through 9 in the input instructions).

K. Running ACER

The following input instructions are copied from the comment cards at the beginning of the ACER module, version 91.0. Users should always check the actual comment cards for the current version to see if there have been any changes.

```
C     *                                                              *
C     *---INPUT SPECIFICATIONS (FREE FORMAT)-------------------------*
C     *                                                              *
C     * CARD 1                                                       *
C     *    NENDF    UNIT FOR INPUT ENDF/B TAPE                       *
C     *    NPEND    UNIT FOR INPUT PENDF TAPE                        *
C     *    NGEND    UNIT FOR INPUT MULTIGROUP PHOTON DATA            *
C     *    NACE     UNIT FOR OUTPUT ACE TAPE                         *
C     *    NDIR     UNIT FOR OUTPUT MCNP DIRECTORY                   *
C     * CARD 2                                                       *
C     *    IOPT     TYPE OF ACER RUN OPTION                          *
C     *               1   FAST DATA                                  *
C     *               2   THERMAL DATA                               *
C     *               3   DOSIMETRY DATA                             *
C     *               4   PHOTO-ATOMIC DATA                          *
C     *               7   READ TYPE 1 ACE FILES TO PRINT OR EDIT     *
C     *               8   READ TYPE 2 ACE FILES TO PRINT OR EDIT     *
C     *               9   READ TYPE 3 ACE FILES TO PRINT OR EDIT     *
C     *    IPRINT   PRINT CONTROL (0 MIN, 1 MAX, DEFAULT=1)          *
C     *    NTYPE    ACE OUTPUT TYPE (1, 2, OR 3, DEFAULT=1)          *
C     *    SUFF     ID SUFFIX FOR ZAID (DEFAULT=.00)                 *
```

```
C     *    NXTRA    NUMBER OF IZ,AW PAIRS TO READ IN (DEFAULT=0)       *
C     * CARD 3                                                         *
C     *    HK       DESCRIPTIVE CHARACTER STRING (70 CHAR MAX)         *
C     * CARD 4 (NXTRA.GT.0 ONLY)                                       *
C     *    IZ,AW    NXTRA PAIRS OF IZ AND AW                          *
C     *                                                                *
C     *    --- FAST DATA (IOPT=1 ONLY) ---                            *
C     *                                                                *
C     * CARD 5                                                         *
C     *    MATD     MATERIAL TO BE PROCESSED                          *
C     *    TEMPD    TEMPERATURE DESIRED (KELVIN) (DEFAULT=300)        *
C     * CARD 6                                                         *
C     *    ERR      TOLERANCE FOR THINNING DISTRIBUTIONS              *
C     *    IOPP     DETAILED PHOTONS (0=NO, 1=YES, DEFAULT=1)         *
C     * CARD 7                                                         *
C     *  TYPE OF THINNING IS DETERMINED BY SIGN OF THIN(1)             *
C     *  (POS. OR ZERO/NEG.=ENERGY SKIP/INTEGRAL FRACTION)             *
C     *  (ALL ENTRIES DEFAULTED=NO THINNING)                           *
C     *    THIN(1)  E1 ENERGY BELOW WHICH TO USE ALL ENERGIES (EV)     *
C     *             OR IWTT WEIGHTING OPTION (1=FLAT,2=1/E)            *
C     *             (1/E ACTUALLY HAS WEIGHT=10 WHEN E LT .1)          *
C     *    THIN(2)  E2 ENERGY ABOVE WHICH TO USE ALL ENERGIES          *
C     *             OR TARGET NUMBER OF POINTS                         *
C     *    THIN(3)  ISKF SKIP FACTOR--USE EVERY ISKF-TH ENERGY         *
C     *             BETWEEN E1 AND E2, OR RSIGZ REFERENCE SIGMA ZERO   *
C     *                                                                *
C     *   --- THERMAL DATA (IOPT=2 ONLY) ---                          *
C     *                                                                *
C     * CARD 8                                                         *
C     *    MATD     MATERIAL TO BE PROCESSED                          *
C     *    TEMPD    TEMPERATURE DESIRED (KELVIN) (DEFAULT=300)        *
C     *    TNAME    THERMAL ZAID NAME (6 CHAR MAX, DEF=ZA)            *
C     * CARD 8A                                                        *
C     *    IZA01    MODERATOR COMPONENT ZA VALUE                      *
C     *    IZA02    MODERATOR COMPONENT ZA VALUE (DEF=0)              *
C     *    IZA03    MODERATOR COMPONENT ZA VALUE (DEF=0)              *
C     * CARD 9                                                         *
C     *    MTI      MT FOR THERMAL INCOHERENT DATA                    *
C     *    NBINT    NUMBER OF BINS FOR INCOHERENT SCATTERING          *
C     *    MTE      MT FOR THERMAL ELASTIC DATA                       *
C     *    IELAS    0/1=COHERENT/INCOHERENT ELASTIC                   *
C     *    NMIX     NUMBER OF ATOM TYPES IN MIXED MODERATOR           *
C     *             (DEFAULT=1, NOT MIXED)                            *
C     *             (EXAMPLE, 2 FOR BEO OR C6H6)                      *
C     *    EMAX     MAXIMUM ENERGY FOR THERMAL TREATMENT (EV)         *
C     *             (DEFAULT=1000.=DETERMINED FROM MF3, MTI)          *
C     *    IWT      WEIGHTING OPTION                                  *
C     *             0/1=VARIABLE/CONSTANT (DEFAULT=VARIABLE)          *
C     *                                                               *
C     *   --- DOSIMETRY DATA (IOPT=3 ONLY) ---                        *
C     *                                                               *
C     * CARD 10                                                       *
C     *    MATD     MATERIAL TO BE PROCESSED                         *
```

```
C     *    TEMPD    TEMPERATURE DESIRED (KELVIN) (DEFAULT=300)          *
C     *                                                                 *
C     *   --- PHOTO-ATOMIC DATA (IOPT=4 ONLY) ---                       *
C     *                                                                 *
C     * CARD 11                                                         *
C     *    MATD     MATERIAL TO BE PROCESSED                           *
C     *                                                                 *
C     *   --- PRINT OR EDIT EXISTING FILES (IOPT=7-9) ---               *
C     *                                                                 *
C     *    NO ADDITIONAL INPUT CARDS ARE REQUIRED.  MOUNT THE OLD       *
C     *    ACE TAPE ON "NPEND".  THE CODE CAN MODIFY ZAID, HK,          *
C     *    THE (IZ,AW) LIST, AND THE TYPE OF THE FILE.  USE SUFF<0      *
C     *    TO LEAVE THE OLD ZAID UNCHANGED.  USE JUST "/" ON            *
C     *    CARD 3 TO LEAVE THE COMMENT FIELD HK UNCHANGED.  USE         *
C     *    NXTRA=0 TO LEAVE THE OLD IZ,AW LIST UNCHANGED.               *
C     *    THE CODE CAN MODIFY ZAID, HK, AND TYPE OF FILE.              *
C     *                                                                 *
```

1. Card 1. ACER uses the information from the ENDF tape mounted on unit **NENDF** for angle, energy, energy-angle, and photon emission distributions, and it uses the data on **NPEND** for the unionized and linearized cross sections. The latter file should have been processed through RECONR, and maybe BROADR. If it wasn't, ACER will still work, but the energy grids may not be quite right. The **NGEND** unit is only needed if the old 30×20 photon production matrix is to be constructed; otherwise, set it to zero. Unit **NACE** is the main output tape for the ACE-format library, and unit **NDIR** will contain a single line of text intended to be edited and incorporated into a directory for a big multimaterial ACE library.

2. Card 2. The value of **IOPT** specifies the kind of ACE data being produced as indicated in the instructions. The long print option, **IPRINT=1**, produces a complete, interpreted listing of the ACE data. The shorter print options just put out progress information from the ACER job and a brief listing of the header information for the library that was generated. The **NTYPE** parameter specifies whether the output library will be in ASCII or binary form. Type 3 files are used at Los Alamos, and most other users can ignore them. In MCNP jobs, materials are identified by their "zaid" numbers (rhymes with "staid"); they are constructed by using the value 1000×Z+A, appending the value of **SUFF** (the suffix), and then adding a letter that indicates the library type ("c" for continuous, "d" for dosimetry, "t" for thermal, and "p" for photoatomic). For example, 92235.50c denotes an ENDF/B-V continuous (fast data) library entry for ^{235}U. Thermal zaids are actually alphanumeric before the dot; see Card 8. Finally, **NXTRA** is the number of extra **IZ, AW** pairs of values to be read in on Card 4.

3. Card 3. This card contains a descriptive character string up to 70 characters long. It must be delimited by * characters and terminated by the / character.

4. Card 4. Read in NXTRA pairs of numbers IZ and AW for photoatomic data. Use as many cards as necessary.

5. Card 5. This is the first card for a fast data library run. It specifies the ENDF MAT number and the absolute temperature for the material to be processed. ENDF MAT numbers are 4-digit numbers. For the earlier versions of ENDF/B, they were assigned in an arbitrary way; for example, 1276 was 8-O-16 for ENDF/B-IV and 1395 represented 92-U-235 for ENDF/B-V. For ENDF/B-VI, a systematic scheme has been selected that allows the same MAT number to be used for all the various sublibraries (for example, neutron data, thermal data, incident proton data, etc.). This scheme is based on using Z to get the first two digits. The second two digits are chosen to be zero for elements, and for normal isotopes, they step in units of 3 up and down around 25, the value for the lowest isotope of the normal stable group of isotopes. This leaves room for two isomers for each isotope in between. Examples include 9225 for ^{234}U, 9228 for ^{235}U, and 2200 for natural Ti. If the temperature is greater than zero, the input PENDF tape must have been run through BROADR.

6. Card 6. The value ERR is the fractional tolerance used in thinning out incident energy values from the angular distribution and energy distribution sections; that is, ERR=.01 means that linear interpolation should describe the incident-energy dependence of the distributions within 1%. The value of IOPP determines whether the newer detailed photon data option is used (preferred), or whether the older 30×20 photon production matrices are to be generated. Remember that if the latter option is selected, GROUPR must be run before ACER to generate multigroup versions of the photon production matrices for all reactions, and the resulting GENDF tape must be coupled to ACER using the NGEND input parameter.

7. Card 7. Thinning of the union energy grid can be performed using several options as described in detail in Section A above. Defaulting the entire card by entering only / results in no thinning.

8. Card 8. This card starts the input of parameters for the thermal library option. The material MAT number and absolute temperature are given. The default temperature is 300 K and the default **TNAME** is generated from ZA. Therefore, the simplest version of this card would consist of a MAT number followed by /. Check the discussion above for Card 5. In almost all cases, an entry for **TNAME** is desirable. Examples are "LWTR" for H in light water (or "HH2O" for hydrogen in water), "GRAPH" for graphite, "ZRZRH" for zirconium in zirconium hydride, and so on.

9. Card 8a. MCNP needs to know the zaid values to get the fast data needed to go with a particular set of thermal data. For a thermal set like HH2O, only **IZA01=1001** would be needed. For a mixed moderator like BeO or benzine, values for both **IZA01** and **IZA02** must be given (e.g., 4009 and 8016). See **NMIX** below. The third input parameter allows for the aliases 6012 and 6000, if needed.

10. Card 9. The value **MTI** must correspond to one of the values used for this material in the THERMR runs that generated the input PENDF tape (see Table 3). The number of bins to use for the equally probable bins for the outgoing neutron spectra is defined by **NBINT**; the default value is 16. The parameter **MTE** defaults to zero; a nonzero value is only needed for materials that show elastic scattering (see Table 3). The value **IELAS** indicates whether this elastic scattering is coherent or incoherent. In the ENDF/B thermal evaluations, some isotopes for mixtures are represented like water, which describes the scattering from hydrogen bound in water, and some are given like benzine, which describes the scattering from the benzine molecule normalized to the hydrogen cross section. The parameter **NMIX** is used to tell ACER about this effect. For the existing ENDF/B evaluations, the user should use **NMIX**=2 for BeO and benzine; the user should use **NMIX**=1 for all other materials. The parameter **EMAX** defines the maximum energy to be used for the thermal scattering treatment. This value should be coordinated with the value of **EMAX** in THERMR. A number around 4 eV is reasonable for most problems. The default for this parameter is 1000., which means that the code will determine the upper limit from the data in MF3 on the PENDF tape. Thus, the value used in THERMR is passed into ACER without the user having to check it. The last parameter on Card 9 is **IWT**. As described in Section XVII.H, the weighting pattern for probability bins for emitted thermal neutrons can be flat (that is, equally probable bins), or it can be variable in the pattern 1, 4, 10, 10,..., 10, 4, 1 in order to better sample the outlying wings of the energy distribution. The default is variable.

Table 3: Conventional Values for the Thermal MT Numbers (MTI and MTE) Used in ACER and THERMR

Thermal Material	MTI Value	MTE Value
H in H_2O	222	
D in D_2O	228	
Be metal	231	232
Graphite	229	230
Methane	227	
Zr in ZrH	235	236
H in ZrH	225	226
BeO	233	234
H in Polyethylene	223	224

The simplest version of Card 9 would contain only MTI followed by /. This works for many materials, including water, heavy water, and polyethylene.

11. Card 10. This card is used for dosimetry libraries only. It specifies the material MAT number and absolute temperatures. See the discussion of MAT and TEMPD for the fast (continuous) libraries.

12. Card 11. Photoatomic libraries require only the single parameter MATD. For ENDF/B-VI input, this will be a number like 100 for hydrogen or 9200 for uranium.

13. Editing Runs. No special input cards are required for editing runs. The data in the library is automatically determined from the ZAID suffix. The type of output library is determined by NTYPE on Card 2. Changes can be made in the fields ZAID, HK, and in the IZ,AW list. One common use for these editing parameters would be as follows. A user runs several isotopes, one at a time, using the default ZAID of ".00c", and 1/E integral thinning. Later, the user decides that all these materials should go into a particular library with the suffix ".67c" and the comment field "ENDF/B-VI library 7 for thermal reactor applications." ACER can handle these changes.

L. Coding Details

ACER begins in the ACER program by reading the user's input. The input varies according to whether "fast", "thermal", "dosimetry", or "photoatomic" data are wanted, or if editing or type conversion is desired. See IOPT. Input ENDF and PENDF tapes are required. The input GENDF tape is only needed if a photon production matrix is to be constructed. The code then branches to a different subroutine for each different value of IOPT.

Processing of fast data is controlled by ACETOP. It begins by opening the ENDF, PENDF, and GENDF units, and by opening a scratch file MSCR used to accumulate the input for the ACELOD procedure. Subroutine FIRST is then called to read the MF1 and MF2 data from the input tapes and to prepare the corresponding data on MSCR. It copies the TAPEID record and Hollerith descriptive data on the PENDF tape to MSCR, processes the directory on the ENDF tape in order to set flags that depend on which reactions and data types are present, converts the fission yield information to a standard form (see TABIZE), writes a dummy File 2 on MSCR, and prepares an ordered list of all the threshold energies in File 3.

If necessary, CONVER is called next. It converts MF12 photon transition probability arrays (LO=2), if present, into the photon yield format (LO=1) by tracing all the cascades through the photon level structure. The results are written on NSCR2. Sections of File 13 are simply copied to the scratch tape. If MF12 was converted, corresponding isotropic photon angular distribution sections are constructed and written into File 14 on the scratch tape. Other sections of File 14 are copied. If File 6 is present (ENDF/B-VI evaluations only), CONVER checks to see whether photon production subsections are present. If so, it converts them into specially defined MF16 sections on the scratch tape. The resulting scratch tape contains sections for MF12, MF13, MF14, and MF16.

Returning to ACETOP, subroutine UNIONX is called to construct the union energy grid. It starts by reading in the energy grid of the total cross section on the PENDF tape. If RECONR and BROADR were used to produce the PENDF tape, this is the desired union grid. If another file is used for PENDF input, ACER will still work, but the grid might not be really correct. Subroutine UNIONX now reads forward on the PENDF tape to find the capture cross section, MT102, and it carries out the integral thinning procedure described above. When an energy grid has been found that satisfies the input target, UNIONX prints out a table of the final resonance integrals by energy bands. It then returns to the PENDF tape and writes out all the partial cross sections on the new, thinned, union energy grid on a temporary scratch tape. At the same time, UNIONX accumulates new values for

the total cross section by adding up the parts. Finally, the new total cross section is written on MSCR, and the partial cross sections are copied from the temporary scratch tape to MSCR. MSCR now has a complete File 3.

Next, ACETOP calls TOPFIL to prepare new versions of MF4, MF5, and MF6 on MSCR. First, PTLEG1 is called to precalculate some of the constants needed for converting angular distributions to equally probable bins. Note PTLEG1 and PTLEG are different entries to the same subroutine. After the constants have been calculated, TOPFIL starts a loop over all the sections for Files 4, 5, and 6 on the ENDF tape. Inside this loop, there is a loop over incident energy, which removes distributions that can be obtained by interpolation from neighboring distributions within the fractional tolerance ERR. This thinning process uses subroutines TEST4, TEST5, and TEST6 to check whether interpolation will work. The File 5 test only checks tabulated distributions (LF=1). The File 6 test is currently only a dummy routine. No File 6 incident-energy thinning takes place.

Distributions that are not removed are checked for type. Angular distributions from MF4, or from MF6 with LAW=2 or 3, are converted to equal-probability bins using PTLEG or PTTAB for distributions given in Legendre form or tabulated form, respectively, and written out to MSCR. Data from MF6 with LAW=7 are used to compute a differential scattering probability $f(E,\mu)$ by integrating over E' for each μ; the results are stored by converting the TAB2 record for each E into a TAB1 record that looks just like a piece of an angular distribution from MF4. They are then converted to equal-probability bins using PTTAB and stored on MSCR. The corresponding energy distributions are normalized and copied to MSCR. Sections of File 5 with LF=5 (delayed neutron data) are converted to equal-probability form using PTTAB. Other sections of File 5 and the neutron parts of File 6 with LAW=1 are copied to MSCR.

When TOPFIL is complete, ACETOP calls GAMSUM, which computes the total photon production cross section on the union grid. It does this by adding the contributions from the MF12 photon yields times the corresponding MF3 cross sections, the MF13 photon production cross sections, and MF16 yields times the corresponding MF3 cross sections. The results are written out using MF13, MT1.

Next, subroutine GAMOUT is called to add the photon distribution information to the main scratch tape. If no GENDF tape is available, GAMOUT simply copies MF14, MF15, and MF16 from the scratch tape prepared by CONVER. However, if the GENDF tape is present, it prepares the 600-word photon production matrix. The first step is to read through the input tape extracting the photon group boundaries and adding up all the photon production reactions into one matrix.

The code then loops over neutron groups converting the outgoing photon groups into equal-probability bins and computing the single discrete photon in each bin that conserves energy. Finally, the resulting 30-by-20 matrix is written onto the output tape using the identification MF15, MT1.

Upon returning to ACETOP, the tape MSCR is completed by adding a material-end, or MEND, record. Subroutine ACELOD is called with MSCR as its input file. It reads through the file in order and stores the numbers into memory in ACE format. The first step is to extract the fission $\bar{\nu}$ data from MF1 and store it away for future use. Next, the energy grid of the total cross section from MF3 is read in starting at pointer ESZ. This determines the number of energy points in the union grid, NES, which can then be used to compute the pointers to the other cross sections in the main cross section block (namely, IT for total, IC for absorption, IE for elastic, and IH for heating). The blocks for supplemental cross sections can also be assigned; for example, LQR for reaction Q values, or LSIG for reaction cross section pointers. With all these pointers computed, ACELOD can simply read through File 3 and store all the cross sections, Q values, and cross section locators in their assigned slots. The total and absorption cross sections are summed up from their parts during this process. After MF3 has been read, the fission $\bar{\nu}$ data can be stored in the main memory block.

The next step is to assign the LAND and AND blocks after the cross section data, and to read in the angular distribution data, store them in the AND block, and save the pointers in the LAND block. Then, the LDLW and DLW blocks are assigned after the angular data, and the energy distribution data are read and stored.

Each section of energy distribution data is examined for its "LAW" and stored accordingly. The analytic laws from ENDF File 5 are simple to store; there is basically a one-to-one correspondence between the ENDF quantities and their ACE equivalents (except, eV are converted to MeV). Tabulated sections in File 5 (LF=1) are converted into the LAW=4 cumulative density function format by computing the cumulative probability function and storing tables of E', $P(E')$, and $C(E')$ for each E. The processing of sections from File 6 is a little bit more complicated.

The particle yield for data from MF5 is determined by the MT number; for example, the yield is 2 for MT=16, the (n,2n) reaction. On the other hand, subsections of File 6 contain explicit values for the particle yield, and the yield may vary with E. In addition, there may be more than one subsection describing emission for a particle. An example of this is ^{19}F from ENDF/B-VI, which has two

subsections for emitted neutrons (first neutron and second neutron). To handle all these complications, ACER reads through the entire section of MF6 data for a given reaction, looks at all the yield tabulations, and computes the total yield for the reaction. It types out messages if multiple subsections are found for one particle, if energy-dependent yields are found, or if noninteger yields are found.

The case of constant integer yields is simple; the value is stored in the TYR array just as for MF5 reactions. The sign of the yield in TYR is positive for laboratory data and negative for CM data.

Generalized yield data are stored as a table of E and $Y(E)$ starting at location NTYR with respect to the start of the DLW block. The value of NTYR is stored in the TYR array as 100+NTYR. The sign of this value is left positive for laboratory data and made negative for CM data. The code then repositions the input file to the start of the section for the current reaction in order to read in the distributions.

As each subsection is read, the yield tabulation is converted into a fractional probability for this subsection by dividing by the generalized yield. There are four different types of secondary particle distributions that must be processed: Kalbach data (LAW=1, LANG=2), Legendre data (LAW=1, LANG=1), angle-energy data (LAW=7), and phase-space data (LAW=6).

The first two share the same loop over incident energy E. For each secondary energy E', the probability $P(E')$ and the angular representation are read from the input file, and the cumulative probability density $C(E')$ is computed. For Kalbach data, the only angular parameter is the preequilibrium fraction r. The corresponding slope parameter a is computed using the function BACHAA, and both r and a are stored in the table using the ACE Law 44 format.

Legendre data are converted into equivalent Kalbach data and stored using Law 44. There are several possible paths. Isotropic data are stored using $r = 0$ and $a = .00001$. Nonisotropic Legendre distributions may be in the laboratory or the CM frame. In either case, the values of the scattering probability are computed for $\mu = -1$ and $\mu = +1$ in the CM frame. These two values are used to compute FBARCM and DELFCM, the average value and the forward peaking, respectively. If the forward-peaking value is negative, it means that the laboratory distribution was physically unreasonable. This typically occurs for E' much less than E when the evaluator didn't take the trouble to provide the small amount of forward peaking needed for an accurate representation. In such cases, DELFCM is set to zero and a message is printed out. The next step is to use subroutine FNDAR1 to find the values of r and a that correspond to FBARCM and DELFCM. A separate but similar path is provided for tabulated data. The only additional complication occurs for

angular data that do not span the entire angle range. In this case, subroutine **FNDAR2** is used to compute r and a.

LAW=7 data are handled in a separate incident-energy loop, which stores data using the ACE Law 67 format. The individual energy distributions on the input file have already been normalized and are ready to be stored in the **XSS** array. Note that the values of **INTMU** and **NMU** were passed to this part of the code using two nonstandard locations in the TAB1 record that was originally the TAB2 record controlling the loop over μ but now contains the angular distribution in 32 equal-probability bins.

The phase-space distribution doesn't need an incident-energy loop. It is only necessary to store the values of **APSX** and **NPXS** into the **XSS** array and to compute a single set of E', $P(E')$, and $C(E')$ values for $E'_{\max} = 1$. The normalization factor C_n is obtained from the integration over E' in order to guarantee that $C(E'_{\max} = 1)$.

Once the secondary-particle energy distributions and angle-energy distributions have all been stored, the GPD pointer is computed to point to the start of the photon production data. The total photon production cross section itself is simply read from the section MF13,MT1 on the input tape and stored starting at GPD. The next step depends on whether photon production matrices were requested by giving ACER a nonzero value for the input GENDF tape. If so, the matrix is read from the section MF15,MT1 on the input tape and stored in memory just after the photon production cross section. The ACE fast library is finished. If not, a dummy matrix of 600 zeros is stored instead, and **ACELOD** starts to work storing the detailed photon data.

Now that all the fast ACE data have been stored into memory, **ACETOP** calls **ACEPRT** to print the data on the output listing file. The amount of information printed depends on the value of the input parameter **IPRINT**. Finally, **ACETOP** calls **ACEOUT** to write out the ACE fast library.

The output library file can be written in Type 1, Type 2, or Type 3 format, depending on the value of **ITYPE**. As described above, Type 1 is a simple formatted file suitable for exchanging ACE libraries between different computers, Type 2 is a FORTRAN-77 direct-access binary file, and Type 3 is a word-addressed binary file. (Type 3 files require nonstandard read and write calls **RDISK** and **WDISK**. They are currently used only at Los Alamos). The problem here is that Type 2 files are written with all data as real numbers (except for some of the fields in the header record). Some of these numbers represent integers, and the Type 1 format requires that these numbers be written into their fields in integer format,

that is, right justified and without decimal points. For the Type 3 format, the real quantities are written as real numbers and the integer quantities are written as integers. In order to handle all these file types in a portable way, subroutine ACELOD first stores all values into memory as real numbers in the array XSS. Therefore, the contents of memory can be written out in Type 2 format with no further processing. In order to convert to Type 1 or Type 3 formats, subroutine CHANGE is used. CHANGE knows the type (real or integer) of every word in the ACE format. For changing the contents of XSS into Type 3 data, it uses TYPEN to change each field representing an integer into an actual integer by means of the equivalenced arrays XSS and ISS. When converting from internal Type 2 data to Type 1 output, it uses TYPEN to write the number directly to NOUT using the appropriate format (I20 or 1PE20.12).

Processing of thermal data is controlled by subroutine ACESIX. It starts by finding the desired temperature on the input PENDF tape. The inelastic and elastic cross sections are copied to a scratch file NSCR. The scratch file is then rewound, and the inelastic cross section is read again to determine the maximum thermal energy EMAX.

If the elastic component is coherent, the input cross sections from NIN are divided by E to get a stair-step function, which is written to the output file.

If the elastic component is incoherent, the incident energies and equally probable emission cosines are read from MF6 on NIN and the corresponding cross sections are read from MF3 on NSCR. All data are stored into memory in ACE format and then written onto NOUT.

The processing of inelastic scattering is more complex. After the proper section on File 6 is located, a uniform or variable pattern of weights is constructed in WT(I). The energy grid is obtained from File 6 on NIN, and the corresponding cross sections are read from MF3 on NSCR. The secondary-energy spectrum for each incident energy is converted into bins using the weight pattern in WT(I), and the single E' that conserves the average energy for the bin is computed. The NANG equally probable cosines for this new E' are obtained by interpolation. Once all of the NBIN*NANG events have been computed and stored in memory, they are copied out to NOUT.

At this point, all the thermal data have been computed, and NOUT is passed to subroutine THRLOD for storing into memory in ACE format. This memory image is then printed out using subroutine THRPRT, if desired, and written to the final Type 1 or Type 2 output file using THROUT.

Processing of dosimetry data is controlled by subroutine ACEDOS. It starts by reserving a scratch storage area for reading in cross sections, and it locates the desired material and temperature on the input tape NPEND. Once the first reaction in File 3 has been located successfully, it defines locators for the MTR, LSIG, and SIG blocks by assuming that there are no more than NMAX=100 reactions present, and it begins a loop over all the reactions in File 3.

For each reaction, its MT identifier is stored in the MTR block, the current pointer value is stored in the LSIG block, and the interpolation table and cross sections are stored in the SIGD block starting at the current pointer value. The pointer is then increased by the number of words stored, and the reaction loop continues. Note that if the input tape is real PENDF tape (that is, if it has been through RECONR), the cross sections will have been linearized onto a union grid. There will only be a single interpolation range for each reaction. However, if the input file was an ENDF tape, there may be several interpolation ranges specifying nonlinear interpolations laws for a given reaction, and the energy grids for different reactions may be different.

When the reaction loop has been completed, the excess space in the MTR and LSIG blocks is squeezed out, the input file is closed, and the scratch storage array is released. The final steps are to call DOSPRT to print the results, and to call DOSOUT to write the ACE dosimetry library file. There is no Type 3 output format for dosimetry. The DOSOUT routine calls TYPEN directly to cause Type 1 fields to be written, if requested.

Processing of photoatomic data is controlled by subroutine ACEPHO. It starts by allocating an area for scratch storage and opening the input file. It then reads in the energy grid for the total cross section, MT=501, which will be used as the union grid for all the photoatomic reactions, starting at pointer ESZ. The number of energy points in the union grid is NES, and that value can now be used to compute the pointers IINC, ICOH, IABS, and IPAIR, representing incoherent scattering, coherent scattering, photoelectric absorption, and pair production, respectively. The ACEPHO routine then reads through File 23 on the input tape, extracts the cross sections for each of the reactions using the energy points of the union grid, and stores the cross sections at the appropriate pointer values. Note that the cross sections and energies have not been converted to log form at this point, but the energies are converted to MeV.

The pointers to the JINC block for incoherent scattering functions, the JCOH block for coherent form factors, the JFLO block for flourescence data, and the LHNM

block for heating numbers are now computed in the storage area just above the cross sections.

The JCOH block uses fixed grid of 55 values for the momentum transfer of the recoil electron (in inverse angstroms) specified in DATA VC/.../. The code first reads through MF27,MT502 and interpolates for the values of the coherent form factor at these 55 recoil values. They are stored in the JCOH block as the second block of 55 numbers. The code then loops through the 55 recoil values again, computing the cumulative integral of the coherent form factor for each recoil value and storing them as the first group of 55 words in JCOH.

The incoherent scattering function is tabulated on a fixed grid of 21 values for the momentum of the recoil electron (in inverse angstroms) that is given in DATA VI/.../. The values are obtained by interpolating in the section MF27,MT504 from the input tape. At the same time, the contribution to the heating from incoherent photon scattering is computed on the union grid with IHEAT.

The calculation of flourescence data for photoelectric absorption is not complete in this version.

Finally, PHOPRT is called to prepare the output listing for the photoatomic data, and PHOOUT is called to prepare the output library file. There is no Type 3 output format for the photoatomic library. Note that TYPEN is called for each field in order to write it out in Type 1 format, if requested.

Conversion of ACE files between the three types, printing of ACE files in any of the types, and editing of the parameters of an ACE file is controlled by ACEFIX. It first determines the file type ITYPE and branches to the appropriate part of the routine to read the data into memory. Note that CHANGE is called for Type 3 files to convert the integers in the binary data into equivalent real numbers in memory. Next, ACEFIX changes the ZAID, comment field, and/or the IZ,AW list. Finally, the subroutine prints the ACE data using the appropriate subroutine (ACEPRT for fast data, THRPRT for thermal data, DOSPRT for dosimetry data, or PHOPRT for photoatomic data) and/or writes a new output file using one of the output subroutines (ACEOUT, THROUT, DOSOUT, or PHOOUT). The output file type can be different from the input file type.

M. Error Messages

ERROR IN ACER***ILLEGAL IOPT

IOPT must be between 1 and 9.

ERROR IN FIRST***DESIRED TEMPERATURE NOT FOUND

Desired temperature was not found on the input PENDF tape. Check for an input error or whether the wrong tape was mounted.

ERROR IN FIRST***STORAGE EXCEEDED

This can result if there are more than 10 sections in File 12 on the input ENDF tape or if there are more than NGMTMX=500 different gamma rays described in the evaluation (NGMTMX is defined in a data statement in FIRST).

ERROR IN TABIZE***STACK EXCEEDED

The stack used for linearizing the fission yield has been exceeded. There must be a major problem with the input data from File 1.

ERROR IN ISLAN1***....

These various errors are self-explanatory. They reflect problems in the data being linearized.

ERROR IN ISLIN2***XLOW GT XHIGH

The upper limit of this panel is lower than its lower limit. Check the data being linearized.

ERROR IN TOPFIL***NXC.GT.NXCMAX

More than 300 reactions have been found on the input ENDF tape. The limit NXCMAX=300 is set at the beginning of ACER. It controls the size of the directory of reactions [see MFs(300) and MTS(300) in COMMON/ACE2/].

ERROR IN TOPFIL***STORAGE EXCEEDED IN A

The input data for this routine is stored in a scratch array with a length of 5000 words. See DATA NWMAX4/5000/.

ERROR IN PTLEG***MT NOT EQUAL TO ZERO, CARD IS ....

Expected a SEND card, but found ....

ERROR IN CPTAB***TOO MANY TERP RANGES

The code can only handle records with a single interpolation range; however, some sections with NR=2 can be converted to NR=1.

ERROR IN CPTAB***ILLEGAL TERP SCHEME

The first interpolation scheme must be linear-linear.

ERROR IN CPTAB***EXPECTED SEND CARD

The expected SEND card was not found.

ERROR IN PTTAB***STORAGE EXCEEDED

This routine can process up to 300 secondary angles. See DATA NPMAX/300/ and the local arrays AMU(300), P(300), TBMU(300), and PMU(300).

ERROR IN PTTAB***TAB ANG DIS HAS MORE THAN ONE TERP RANGE

Only one interpolation range is allowed when processing tabulated angular distributions in PTTAB. In some cases, two ranges are allowed; see CHEKIT.

ERROR IN PTTAB***TAB ANG DIST NOT ALLOWED FOR

Interpolation schemes that use logs for the scattering cosine (INT=3 or 5) are not allowed because μ can take on negative values.

ERROR IN PTTAB***L+1.LT.NPT

The code is having problems computing equally probable bins for histogram interpolation. Check input data for normalization.

ERROR IN PTTAB***DID NOT FIND EXPECTED SEND CARD

This card should have been a section-end card. There must be something wrong with the input data.

ERROR IN GAMSUM***EXCEEDED STORAGE IN DICTIONARY

Only NXCMAX=300 reactions are allowed. See MFS, MTS, and NCS in common ACE2. The parameter NXCMAX is set at the start of subroutine ACER.

ERROR IN CONVR***STORAGE EXCEEDED FOR PHOTON DATA

There is not enough room in the dynamic array TOT for the total photon yield array from MF12 or MF13. Since this array was allocated using NWTOT=-1, the only way to fix the problem is to increase the size of the container array A, which normally contains 11 100 words. See DATA NAMAX/11100/ and COMMON ASTORE(11100) in ACER.

ERROR IN CONVR***STORAGE EXCEEDED FOR EDIS

The list of discontinuity energies is limited to NNED=30 elements. The limit is set in ACER and passed in common block ACE2.

ERROR IN CONVR***ONLY LAW=1 ALLOWED FOR ENDF6 FILE6 PHOTONS

Photon sections in File 6 should use the tabulated representation only.

ERROR IN GAMOUT***MAT NOT FOUND

The desired material was not found on the input GENDF tape. Make sure that the correct file was mounted.

ERROR IN GAMOUT***STORAGE IN A EXCEEDED

Storage exceeded in the dynamic array SCR. Check the value for NWAMAX in this subroutine.

ERROR IN GAMOUT***NO GAMMA GROUPS ON NGEND

The input GENDF tape does not contain a photon group structure. Remember that using the 30×20 matrix option for photon emission requires that a GROUPR run be made to produce multigroup cross sections for all of the photon production reactions.

ERROR IN GAMOUT***STORAGE IN SIG EXCEEDED

Storage limit for the dynamic array SIG has been exceeded. Since this array was allocated using NSMAX=-1, the only recourse is to increase the size of the container array A, which normally contains 11 100 words. See DATA NAMAX/11100/ and COMMON ASTORE(11100) in ACER.

ERROR IN ACELOD***INSUFFICIENT STORAGE FOR ESZ BLOCK

The ESZ block contains 5×NES words, and this value must be less than the limit of MAX3=120 000 words, which is set in ACER and passed in the common block LSIZE. This limit is also the size of the common array XSS in block XSST.

ERROR IN ACELOD***INSUFFICIENT SPACE FOR CROSS SECTIONS

There is not enough space for the SIG block in the container array XSS. See the discussion for the ESZ block above.

ERROR IN ACELOD***INSUFFICIENT SPACE FOR ANGULAR DISTRIBUTIONS

There is not enough space for the ANG block in XSS. See the discussion for the ESZ block above.

ERROR IN ACELOD***INSUFFICIENT SPACE FOR ENERGY DISTRIBUTIONS

There is not enough space for the DLW block in XSS. See the discussion for the ESZ block above.

ERROR IN ACELOD***ILLEGAL LF=...

The ACELOD routine can process LF=1, 5, 7, 9, and 11 from ENDF File 5.

ERROR IN ACELOD***ONLY LAW=1 ALLOWED FOR ENDF6 FILE6 NEUTRONS

The current coding only handles the LAW=1 tabulated format for neutron energy-angle distributions. Future versions of ACER also need to include LAW=6, the "N-Body Phase Space" distributions, and LAW=7, the "Laboratory Angle-Energy" law. The other laws aren't applicable to this part of the code.

ERROR IN ACELOD***INSUFFICIENT SPACE FOR MF6 NEUTRON YIELD

There is not enough space in XSS. See the discussion for the ESZ block above.

ERROR IN ACELOD***ONLY LANG=2 ALLOWED FOR ENDF6 FILE6 NEUTRONS

The current version of ACER can only handle the Kalbach systematics representation. Future versions need to add the LANG=1 Legendre coefficients representation and the LANG=11–15 representations for tabulated angular distributions.

ERROR IN ACELOD***INSUFFICIENT SPACE FOR MF6 TAB2

There is not enough space in XSS. See the discussion for the ESZ block above.

ERROR IN ACELOD***INSUFFICIENT SPACE FOR PHOTON PRODUCTION

There is not enough space in XSS. See the discussion for the ESZ block above.

ERROR IN ACELOD***INSUFFICIENT SPACE FOR PHOTON SPECTRA

There is not enough space in XSS. See the discussion for the ESZ block above.

ERROR IN ACELOD***NO. OF GAMMA ENERGIES NOT COMPLETE

There is something wrong with the data for this reaction on the main ACER scratch file.

ERROR IN ACELOD***INSUFFICIENT STORAGE FOR INPUT PHOTON DATA

There is not enough storage in the dynamic array SCR. This size is controlled by the statement NWSCR=10000 in subroutine ACELOD.

ERROR IN SEP***DOMINANT ISOTOPE NOT KNOWN FOR ...

The separation energy calculation only works for isotopes. This routine contains a small table of the dominant isotope in elements that sometimes appear in evaluations. This message means that the dominant isotope is not known for this element, and that the table must be extended.

ERROR IN ACESIX***STORAGE EXCEEDED FOR COHERENT REACTIONS

There is insufficient space in the A array at pointer ISIX for the coherent or incoherent data, respectively. This is a space of 2000 words. This value is set by the statement NINMAX=2000 at the beginning of the ACESIX routine.

ERROR IN ACESIX***EXCEEDED STORAGE FOR INCOHERENT REACTIONS

See the explanations above.

ERROR IN ACESIX***CODED FOR EQUIPROBABLE ANGLES ONLY

The input thermal File 6 (a nonstandard format) must use the equiprobable angle format. Since this is currently the only format produced by THERMR, this error should not occur.

ERROR IN ACESIX***SOLUTION OUT OF RANGE

The routine is not able to find a legal solution while trying to find the equiprobable bins for inelastic scattering.

ERROR IN ACEDOS***DESIRED MAT AND TEMP NOT FOUND

The requested material and temperature were not found on the input photoatomic PENDF file. Check for an input error, and make sure that the correct file has been mounted as NPEND. Remember that this is normally the output of a RECONR run to assure correct unionization and linearization.

N. References

1. Judith F. Briesmeister, Ed., "MCNP—A General Monte Carlo Code for Neutron and Photon Transport, Version 3A," Los Alamos National Laboratory report LA-7396-M, Rev. 2 (September 1986).

2. P. F. Rose and C. L. Dunford, Eds., "ENDF-102, Data Formats and Procedures for the Evaluated Nuclear Data File, ENDF-6," Brookhaven National Laboratory report BNL-NCS-44945 (July 1990).

3. E. D. Cashwell, J. R. Neergaard, W. M. Taylor, and G. D. Turner, "MCN: A Neutron Monte Carlo Code," Los Alamos Scientific Laboratory report LA-4751 (January 1972).

4. R. J. LaBauve, C. R. Weisbin, R. E. Seamon, M. E. Battat, D. R. Harris, P. G. Young, and M. M. Klein, "PENDF: A Library of Nuclear Data for Monte Carlo Calculations Derived from Data in the ENDF/B Format," Los Alamos Scientific Laboratory report LA-5687 (October 1974).

5. C. Kalbach and F. M. Mann, "Phenomenology of Continuum Angular Distributions. I. Systematics and Parameterization," *Phys. Rev. C.* 23, 112 (1981).

6. P. G. Young and E. D. Arthur, "GNASH: A Preequilibrium Statistical Nuclear Model Code for Calculation of Cross Sections and Emission Spectra," Los Alamos Scientific Laboratoy report LA-6947 (1977).

7. C. Kalbach, "Systematics of Continuum Angular Distributions: Extensions to Higher Energies," *Phys. Rev. C.* 37, 2350 (1988).

8. E. D. Cashwell, J. R. Neergaard, C. J. Everett, R. G. Schrandt, W. M. Taylor, and G. D. Turner, "Monte Carlo Photon Codes MCG and MCP," Los Alamos Scientific Laboratory report LA-5157-MS (March 1973).

9. C. J. Everett and E. D. Cashwell, "MCP Code Flourescence-Routine Revision," Los Alamos Scientific Laboratory report LA-5240-MS (May 1973).

XVIII. POWR

The POWR module is used to prepare libraries for the Electric Power Research Institute (EPRI) reactor analysis codes EPRI-CELL and EPRI-CPM.* These codes were originally developed to provide an alternative to the reactor manufacturers' computer codes for calculating reactor core performance as required for operating and reloading power reactors run by US electric utility companies. Because these codes require up-to-date accurate cross sections that were not controlled by the reactor manufacturers, EPRI contracted with the Los Alamos National Laboratory to generate new libraries based on the US-standard Evaluated Nuclear Data Files (ENDF). With this funding, we were able to develop the THERMR module and the associated thermal multigroup methods in GROUPR. We also developed this POWR module to format cross section data in GENDF format for use in CELL or CPM. In addition, it was necessary to make a number of modifications to EPRI-CELL to make it perform well with unadjusted cross sections. The results of all this work were reported in 1984.[1]

This module has not been used at Los Alamos since 1984, although we have had scattered reports of use elsewhere. A more complete description of the module should be provided in the future.

A. Input Instructions

As an aid to discussions of the user input to POWR, the input instructions that appear as comment cards at the beginning of the current version of this module are listed below. Since code changes are possible, it is always advisable to consult the comment-card instructions contained in the version of the code actually being used before proceeding with an actual calculation.

```
*---INPUT SPECIFICATIONS (FREE FORMAT)-------------------------*
*                                                              *
* CARD 1                                                       *
*    NGENDF  UNIT FOR INPUT GOUT TAPE                          *
*    NOUT    UNIT FOR OUTPUT TAPE                              *
* CARD 2                                                       *
*    LIB     LIBRARY OPTION (1=FAST, 2=THERMAL, 3=CPM)         *
*    IPRINT  PRINT OPTION (0=MINIMUM, 1=MAXIMUM)               *
*            (DEFAULT=0)                                       *
*    ICLAPS  GROUP COLLAPSING OPTION (0=COLLAPSE FROM 185 GROUP *
*            TO DESIRED GROUP STRUCTURE, 1=NO COLLAPSE)        *
*            (DEFAULT=0)                                       *
```

*EPRI-CELL and EPRI-CPM are proprietary products of the Electric Power Research Institute, 3412 Hillview Avenue, Palo Alto, CA 94304. For more information, please contact the owners.

```
*                                                                    *
*---FOR LIB=1---------------------------------------------------------*
*                                                                    *
* CARD 3                                                             *
*    MATD     MATERIAL TO BE PROCESSED                              *
*             IF MATD LT 0, READ-IN ABSORPTION DATA ONLY FOR        *
*             THIS MATERIAL WITH MAT=ABS(MATD) DIRECTLY FROM        *
*             INPUT DECK (SEE CARD 6)                               *
*   FOLLOWING THREE PARAMETERS IRRELEVANT FOR MATD LT 0             *
*    RTEMP    REFERENCE TEMPERATURE (DEGREES KELVIN)                *
*             (DEFAULT=300  K)                                      *
*    IFF      F-FACTOR OPTION                                       *
*             (0/1=DO NOT CALCULATE F-FACTORS/CALCULATE IF FOUND)   *
*             (DEFAULT=1)                                           *
*    NSGZ     NO. OF SIGMA ZEROES TO PROCESS FOR THIS MATERIAL      *
*             (DEFAULT=0=ALL FOUND ON INPUT TAPE)                   *
*    IZREF    REF. SIGZERO FOR ELASTIC MATRIX (DEFAULT=1)           *
* CARDS 4 AND 5 FOR NORMAL RUN ONLY (MATD GT 0)                      *
* CARD 4                                                             *
*    WORD     DESCRIPTION OF NUCLIDE (UP TO 16 CHARACTERS,          *
*             DELIMITED WITH *, ENDED WITH /)  (DEFAULT=BLANK)      *
* CARD 5                                                             *
*    FSN      TITLE OF FISSION SPECTRUM (UP TO 40 CHARACTERS,       *
*             DELIMITED WITH *, ENDED WITH /0  (DEFAULT=BLANK)      *
*              DELIMITED WITH *, ENDED WITH /)  (DEFAULT=BLANK)     *
* CARD 6 FOR READING IN ABSORPTION DATA ONLY                         *
*    ABS      NGND ABSORPTION VALUES (DEFAULT VALUES=0)             *
* REPEAT CARDS 3 THROUGH 6 FOR EACH MATERIAL DESIRED.                *
* TERMINATE WITH MATD=0/ (I.E., A 0/ CARD).                          *
*                                                                    *
*---FOR LIB=2---------------------------------------------------------*
*                                                                    *
* CARD 3                                                             *
*    MATD     MATERIAL TO BE PROCESSED                              *
*    IDTEMP   TEMPERATURE ID (DEFAULT=300  K)                       *
*    NAME     HOLLERITH NAME OF ISOTOPE (UP TO 10 CHARACTERS,       *
*             DELIMITED WITH *, ENDED WITH /)  (DEFAULT=BLANK)      *
* CARD 4      DEFAULT FOR ALL VALUES=0.                              *
*    ITRC     TRANSPORT CORRECTION OPTION (0 NO, 1 YES)             *
*    MTI      THERMAL INELASTIC MT                                  *
*    MTC      THERMAL ELASTIC MT                                    *
* CARD 5      DEFAULT FOR ALL VALUES=0.                              *
*    XI                                                             *
*    ALPHA                                                          *
*    MUBAR                                                          *
*    NU                                                             *
*    KAPPA FISSION                                                  *
*    KAPPA CAPTURE                                                  *
*    LAMBDA                                                         *
*    SIGMA S   IF 0, SET TO SCATTERING CROSS SECTION AT GROUP 35    *
* REPEAT CARDS 3 THRU 5 FOR EACH MATERIAL AND TEMPERATURE DESIRED    *
* (MAXIMUM NUMBER OF TEMPERATURES ALLOWED IS 7.)                     *
* TERMINATE WITH MATD=0/ (I.E., A 0/ CARD).                          *
```

```
*                                                                     *
*---FOR LIB=3----------------------------------------------------------*
*                                                                     *
* CARD 3                                                              *
*    NLIB     NUMBER OF LIBRARY.                                      *
*    IDAT     DATE LIBRARY IS WRITTEN (I FORMAT).                     *
*    NEWMAT   NUMBER OF MATERIALS TO BE ADDED.                        *
*    IOPT     ADD OPTION (0=MATS WILL BE READ IN,                     *
*                         1=USE ALL MATS FOUND ON NGENDF).            *
*    MODE     0/1/2=REPLACE ISOTOPE(2) IN CPMLIB/                     *
*                    ADD/CREATE A NEW LIBRARY (DEFAULT=0)             *
*    IF5      FILE5 (BURNUP DATA) OPTION                              *
*             0/1/2=DO NOT PROCESS FILE5 BURNUP DATA/                 *
*                   PROCESS BURNUP DATA ALONG WITH REST OF DATA/      *
*                    PROCESS BURNUP DATA ONLY (DEFAULT=0)             *
*             (DEFAULT=0)                                             *
*    IF4      FILE4 (CROSS SECTION DATA) OPTION                       *
*             0/1=DO NOT PROCESS/PROCESS                              *
*             (DEFAULT=1)                                             *
* CARD 4 FOR IOPT=0 ONLY                                              *
*    MAT      ENDF MAT NUMBER OF ALL DESIRED MATERIALS.               *
*             FOR MATERIALS NOT ON GENDF TAPE, USE IDENT FOR MAT.     *
*             IF MAT LT 0, ADD 100 TO OUTPUT IDENT                    *
*             (FOR SECOND ISOMER OF AN ISOTOPE)                       *
* CARD 5                                                              *
*    NINA     NINA INDICATOR.                                         *
*             0/1/2/3=NORMAL/                                         *
*                   NO FILE2 DATA, CALCULATE ABSORPTION IN FILE4/     *
*                   NO FILE2 DATA, READ IN ABSORPTION IN FILE4/       *
*                   READ IN ALL FILE2 AND FILE4 DATA.                 *
*    NTEMP    NO. OF TEMPERATURES TO PROCESS FOR THIS MATERIAL        *
*             (DEFAULT=0=ALL FOUND ON INPUT TAPE)                     *
*    NSIGZ    NO. OF SIGMA ZEROES TO PROCESS FOR THIS MATERIAL        *
*             (DEFAULT=0=ALL FOUND ON INPUT TAPE)                     *
*    SGREF    REFERENCE SIGMA ZERO                                    *
*    FOLLOWING 2 PARAMETERS ARE FOR NINA=0 OR NINA=3.                 *
*    IRES     RESONANCE ABSORBER INDICATOR (0/1=NO/YES)               *
*    SIGP     POTENTIAL CROSS SECTION FROM ENDF/B.                    *
*    FOLLOWING 5 PARAMETERS ARE FOR NTAPEA=0 ONLY                     *
*    MTI      THERMAL INELASTIC MT                                    *
*    MTC      THERMAL ELASTIC MT                                      *
*    IP1OPT   0/1=CALCULATE P1 MATRICES/                              *
*                 CORRECT P0 SCATTERING MATRIX INGROUPS.              *
*  ******IF A P1 MATRIX IS CALCULATED FOR ONE OF THE ISOTOPES        *
*         HAVING A P1 MATRIX ON THE OLD LIBRARY, FILE 6 ON THE        *
*         NEW LIBRARY WILL BE COMPLETELY REPLACED.******              *
*    INORF    0/1=INCLUDE RESONANCE FISSION IF FOUND/                 *
*                 DO NOT INCLUDE                                      *
*    FOLLOWING TWO PARAMETERS FOR MODE=0 ONLY                         *
*    POS      POSITION OF THIS ISOTOPE IN CPMLIB                      *
*    POSR     (FOR IRES=1) POSITION OF THIS ISOTOPE IN RESONANCE      *
*             TABULATION IN CPMLIB                                    *
*  REPEAT CARD 5 FOR EACH NUCLIDE.                                    *
```

```
* FOLLOWING THREE CARDS ARE FOR IF5 GT 0 ONLY                          *
* CARD 6                                                               *
*    NTIS     NO. TIME-DEPENDENT ISOTOPES                             *
*    NFIS     NO. FISSIONABLE BURNUP ISOTOPES                         *
* CARD 7                                                               *
*    IDENTB   IDENT OF EACH OF THE NFIS ISOTOPES                      *
* CARD 8                                                               *
*    IDENTA   IDENT OF TIME-DEPENDENT ISOTOPE                         *
*    DECAY    DECAY CONSTANT (DEFAULT=0.)                             *
*    YIELD    NFIS YIELDS (DEFAULT=0.)                                *
*  REPEAT CARD 8 FOR EACH OF THE NTIS ISOTOPES.                       *
* CARD 9 FOR IF5=2 ONLY                                                *
*    AW       ATOMIC WEIGHT                                           *
*    INDFIS   FISSION INDICATOR                                       *
*    NTEMP    NO. TEMPERATURES ON OLD LIBRARY                         *
*  REPEAT CARD 9 FOR EACH OF THE NTIS ISOTOPES.                       *
* CARD 10                                                              *
*    LAMBDA   RESONANCE GROUP GOLDSTEIN LAMBDAS                       *
*  ******REMEMBER THAT THE 69-GROUP STRUCTURE HAS 13 RESONANCE        *
*         GROUPS WHILE THE COLLAPSED 185-GROUP STRUCTURE HAS 15.      *
*         USE A SLASH AT END OF EACH LINE OF CARD 10 INPUT.******     *
*  REPEAT CARD 10 FOR EACH NUCLIDE HAVING NINA=0, NINA=3, OR          *
*                    IRES=1.                                          *
* CARDS 11 AND 11A FOR NUCLIDES HAVING NINA=3 ONLY.                    *
* CARD 11                                                              *
*    RESNU     NRG NUS VALUES TO GO WITH THE LAMBDA VALUES            *
* CARD 11A                                                             *
*    TOT       NRG TOTAL XSEC VALUES TO GO WITH THE LAMBDA VALUES     *
*  READ CARDS 11 AND 11A FOR EACH NUCLIDE HAVING NINA=3.              *
* CARDS 12 FOR NINA GT 2 ONLY                                          *
*    AW       ATOMIC WEIGHT                                           *
*    TEMP     TEMPERATURE                                             *
*    FPA      NGND ABSORPTION VALUES (DEFAULT=0.)                     *
* CARDS 12A, 12B, 12C FOR NUCLIDES HAVING NINA=3 ONLY.                 *
* CARD 12A                                                             *
*    NUS       NGND NUS VALUES                                        *
*    FIS       NGND FISSION VALUES                                    *
*    XTR       NGND TRANSPORT VALUES                                  *
* CARD 12B                                                             *
*    IA        GROUP.  0 MEANS NO SCATTERING FROM THIS GROUP          *
*    L1        LOWEST GROUP TO WHICH SCATTERING OCCURS                *
*    L2        HIGHEST GROUP TO WHICH SCATTERING OCCURS               *
* CARD 12C  FOR IA GT 0 ONLY                                           *
*    SCAT      L2-L1+1 SCATTERING VALUES                              *
*  REPEAT CARD 12B AND 12C FOR EACH GROUP                             *
* REPEAT CARDS 12 FOR EACH OF THE NINA GT 2 NUCLIDES                   *
*                                                                      *
************************************************************************
```

B. References

1. R. E. MacFarlane, "ENDF/B-V Cross-Section Library for Reactor Cell Analysis," Electric Power Research Institute report NP-3418 (February 1984).

XIX. WIMSR

This module is used to prepare libraries for the reactor-physics code WIMS.[1] WIMS stands for "Winfrith improved multigroup scheme;" it has been developed through its various versions at the UK laboratory AEE/Winfrith. WIMS-E is the current Winfrith version, and it is distributed commercially. WIMS-D is an older version that is freely available through various distribution centers; therefore, it is very popular all around the world.

WIMS uses collision-probability methods for computing fluxes in reactor pin cells and more complicated geometrical arrangements. Therefore, it requires transport, fission, and capture cross sections, a transfer matrix for epithermal neutrons, fission-source information ($\overline{\nu}$ and χ), and a bound-atom scattering matrix for thermal neutrons. Self-shielded cross sections are obtained using equivalence theory from tabulated resonance integrals with intermediate resonance corrections. The resonance integrals can be obtained from the self-shielded cross sections produced by GROUPR, and the intermediate-resonance λ values by group can be computed using the NJOY flux calculator. WIMS libraries normally use a standard 69-group structure with 14 fast groups, 13 resonance groups, and 42 thermal groups.

A. Resonance Integrals

WIMS computes the self-shielded cross sections for a wide range of mixtures and fuel geometries using equivalence theory. The GROUPR chapter of this report describes the narrow-resonance (NR) version of equivalence theory; that is, all systems with the same value for the "sigma-zero,"

$$\sigma_{0i} = \frac{1}{n_i} \left\{ \sum_{j \neq i} n_j \sigma_{tj} + \sigma_e \right\} , \tag{1}$$

in a group have the same self-shielded cross section in that group. Here, n_j is the number density for material j with cross section σ_{tj}, and σ_e is the escape cross section (which takes care of the geometry of the fuel).

However, in the near epithermal range (*e.g.*, 4–100 eV), some resonances are too wide for the NR approximation to apply well. For these resonances, the effect of a moderator material is reduced, because collisions with the moderator do not always result in enough energy loss to remove the neutron from the resonance. For this reason, WIMS uses an intermediate-resonance (IR) extension to equivalence

theory in which the background cross section is taken to be the following:

$$\sigma_{Pi} = \frac{1}{n_i}\left\{\sum_j n_j \lambda_j \sigma_{pj} + \sigma_e\right\} , \tag{2}$$

where the λ factors are numbers between zero and one. Note that σ_p, the potential scattering cross section, is used here, and that the sum is now over all materials. The basic concept is the same: all systems with the same value of the IR "sigma-P" for a group will have the same self-shielded cross sections for that group.

WIMS takes the additional step of expressing the self-shielding data in terms of "resonance integrals," instead of using the self-shielded cross sections produced by GROUPR. That is,

$$\sigma_x(\sigma_0) = \frac{\sigma_P I_x(\sigma_P)}{\sigma_P - I_a(\sigma_P)} , \tag{3}$$

and

$$I_x(\sigma_P) = \frac{\sigma_P \sigma_x(\sigma_0)}{\sigma_P + \sigma_a(\sigma_0)} , \tag{4}$$

where x denotes the reaction, either "a" for absorption or "nf" for nu*fission, and I_x is the corresponding resonance integral.

In order to clarify the meaning of this pair of equations, consider a homogeneous mixture of ^{238}U and hydrogen with concentrations such that there are 50 barns of hydrogen scattering per atom of uranium. The GROUPR flux calculator can be used to solve for the flux in this mixture, and GROUPR can then calculate the corresponding absorption cross section for ^{238}U. Assuming that $\lambda = 0.1$ and $\sigma_p = 10$ for the uranium, the numbers being appropriate for WIMS group 25, we get $\sigma_P = 51$. This value of σ_p goes into the SIGZ array in the resonance-integral block on the WIMS library, and the corresponding I_a goes into the RESA array.

At some later time, a WIMS user runs a problem for a homogeneous mixture of ^{238}U and hydrogen that matches these specifications. WIMS will compute a value of σ_P of 51.0, interpolate in the table of resonance integrals, and compute a new absorption cross section that is exactly equal to the accurate computed result from the original GROUPR flux-calculator run.

This argument can be extended to more complex systems. For example, the assembly calculated using the flux calculator could represent an enriched uranium-oxide fuel pin of a size typical of a user's reactor system with a water moderator. The computed absorption cross section is converted to a resonance integral and stored with the computed value of σ_P. In any later calculation that happens to

mimic the same composition and geometry, WIMS will return the accurate calculated absorption cross section. Equivalence theory, with all its approximations, is only used to interpolate and extrapolate around these calculated values. This is a powerful approach, because it allows a user to optimize the library in order to obtain very accurate results for a limited range of systems without having to modify the methods used in the lattice-physics code. Unfortunately, the present version of WIMSR does not allow you to enter σ_P directly; it computes it from input data that only consider one material at a time. A future version may include the more general capabilities described in this paragraph.

Let us call the homogeneous uranium-hydrogen case discussed above "case 0." Now, consider a homogeneous mixture of ^{238}U, oxygen, and hydrogen. Ratio the number densities to the uranium density such that there is 1 barn/atom of oxygen scattering and 50 barn/atom of hydrogen scattering. Carry out an accurate flux calculation for the mixture, and call the result "case 1." Also do an accurate flux calculation with only hydrogen, but at a density corresponding to 51 barns/atom. Call this result "case 2." The IR lambda value for oxygen is then given by

$$\lambda = \frac{\sigma_a(1) - \sigma_a(0)}{\sigma_a(2) - \sigma_a(0)} . \tag{5}$$

Note that λ will be 1 if the oxygen and hydrogen have exactly the same effect on the absorption cross section. In practice, $\lambda = .91$ for WIMS group 27 (which contains the large 6.7 eV resonance of ^{238}U), and $\lambda = 1$ for all the other resonance groups. That is, all the resonances above the 6.7 eV resonance are effectively narrow with respect to oxygen scattering.

Similarly, do a flux-calculator solution for a homogeneous mixture of ^{238}U combined with 1 barn/atom of ^{235}U and 50 barn/atom of hydrogen. Call the result "case 3." Now, the lambda value for ^{235}U is given by

$$\lambda = \frac{\sigma_a(3) - \sigma_a(0)}{\sigma_a(2) - \sigma_a(0)} . \tag{6}$$

The actual value obtained for WIMS group 27 is .035. Group 26 gives 0.50, and group 25 gives 0.09. An examination of the flux-calculator equations in the GROUPR chapter of this manual shows that the effect of the "admixed" moderator term depends only on its atomic mass (through the α value); therefore, the IR λ values will be the same for all uranium isotopes (and the same values should work for all the actinides). This conclusion neglects the small effects of absorption in the admixed isotope on the intraresonance flux for one resonance.

Table 1: IR λ Values for Several Resonance Groups and Two Different Reactor Systems

WIMS group	λ(U) U238@50b	λ(O) U238@50b	λ(U) U235@200b	λ(O) U235@200b
27	.035	.91	0.20	1.00
26	.50	1.00	.38	1.00
25	.092	1.00	.44	1.00
24	.090	1.00	.55	1.00
23	.29	1.00	.46	1.00

This process can be continued for additional admixed materials from each important range of atomic mass. The result is the table of λ_{gi} values needed as WIMSR input.

What are the implications of this discussion? Foremost is the observation that the lambda values for the isotopes are a function of the composition of the mixture that was used for the base calculation. To make the effect of this clear, let us consider two different types of cells:

1. a homogeneous mixture of U238 and hydrogen, and
2. a homogeneous mixture of U235 and hydrogen.

A look at the pointwise cross sections in group 27 shows very different pictures for the two uranium isotopes. The U-238 cross section has one large, fairly wide resonance at 6.7 eV, and the U-235 cross section has several narrower resonances scattered across the group. If the lambda values are computed for these two different situations, the results in Table 1 are obtained.

It is clear that the energy dependence of the two lambda sets is quite different. This is because of the difference in the resonance structure between U-238 and U-235. Clearly, the one resonance in group 27 in U-238 is effectively wider than the group of resonances in group 27 for U-235. Group 26 has essentially no resonance character for ^{238}U, which reverses the sense of the difference. In groups 24 and 25, the ^{235}U resonances become more narrow, while the ^{238}U resonances stay fairly wide. Finally, in group 23, the ^{238}U resonances begin to get narrower.

These results imply that completely different sets of lambda values should be used for different fuel/moderator systems, such as U-238/water, U-235/water, or U-238/graphite. In practice, this is rarely done.

The remaining question is, "How should the self-shielded cross sections for the minor isotopes be calculated?" Formally, the best approach using NJOY would be to first do an accurate flux calculation for pure ^{238}U mixed with hydrogen (or to be really accurate, a typical reactor cell containing pure ^{238}U oxide), and to save the resulting flux on a scratch file. This flux would then be used as input for the ^{235}U calculation. (See the GROUPR chapter for details.) This approach takes care of all the complexities of resonance-resonance interference, the drop in the average flux across the group caused by accumulated ^{238}U absorptions, and so on. In practice, this is rarely done. Since the self-shielding effects in the minor actinides are much smaller than those in the main absorber, it is usually sufficient to do a simple NR calculation for the minor actinides and to convert them into WIMS resonance integrals with the normal lambda values for heavy isotopes.

B. Cross Sections

The first part of the WIMS cross section data contains σ_p for the resonance groups (15–27 in the normal 69-group structure), the scattering power per unit lethargy for the resonance groups, the transport cross section for the fast groups (1–27 normally), the absorption cross section for the fast groups, an obsolete quantity for the resonance groups (set to zero), and the intermediate-resonance λ values for the resonance groups. For WIMS-E, the (n,2n) cross section is added between the slowing-down power and the transport cross section.

The σ_p value is assumed to be constant (see **SIGP** in the user input instructions). It must be obtained by finding the scattering length a in the ENDF file and computing $\sigma_p = 4\pi a^2$. The scattering power per unit lethargy is $\xi\sigma_s/\tau$, where ξ is the log energy loss parameter given as MT=252 on the GENDF file, σ_s is the elastic scattering cross section (MT=2), and τ is the lethargy width for the group, which can be calculated from the group structure given in MF=1/MT=451 on the GENDF file. The (n,2n) cross section is obtained from MF=3/MT=16 on the GENDF file. The absorption cross section is computed by adding up the fission cross section (MT=18) and all the cross sections given with MT=102–150. The (n,2n) cross section (MT=16) is then subtracted from the sum. Finally, the λ values are obtained from the user's input. See Section A for more details on these intermediate-resonance corrections.

The next part of the WIMS data file contains the fission neutron production cross section $\overline{\nu}\sigma_f$ and the fission cross section σ_f for the fast groups (1–27 normally). The cross section is always obtained from MT=18 on the GENDF file, but there are several complications involved in getting $\overline{\nu}$.

A shortcut for obtaining the fission data is to run MFD=3/MTD=452 and MFD=5/MTD=452 in GROUPR. This approach ignores the energy dependence of fission neutron emission at high energies and the effects of delayed neutrons on the fission spectrum. If these options are used in GROUPR, it is important not to use the other options described below at the same time. When WIMSR finds a section on the GENDF file with MF=3/MT=452, it can read in $\overline{\nu}\sigma_f$ directly.

A better approach to fission in GROUPR is to prepare a full fission matrix for MT18, or to prepare matrices for all the partial fission reactions, MT=19, 20, 21, and 38. The latter is the recommended approach for evaluations with both MT18 and MT19 given in File 5. See the GROUPR chapter for more details. WIMSR reads in the data given in MF=6/MT=18, or in MF=6/MT19,20,..., and sums over all secondary-energy groups to obtain the prompt part of $\overline{\nu}\sigma_f$. It adds in the delayed part of $\overline{\nu}\sigma_f$ from MF=3/MT=455. If the input GENDF file contains both MT=18 and partial fission matrices, a diagnostic message will be printed, and the partial-fission representation will be used.

The next section of the WIMS data file contains the nonthermal P_0 scattering matrix for incident-energy groups in the fast range (1–27 normally). This matrix is loaded by summing over all of the reactions found on the GENDF tape except the thermal reactions **MTI** and **MTC**. If requested, this matrix is transport corrected by subtracting the sum over secondary-energy groups of the P_1 matrix for each primary group. When the individual reactions are read, they are loaded into "full" matrix (typically 69×69). At the same time, a record is kept of the lowest and highest secondary groups found for each primary group. These limits are then used to pack the scattering matrix into a more compact form.

The scattering matrix is not actively self-shielded in WIMS, but WIMSR allows the user to request that the elastic component be evaluated at some reference σ_0 value different from infinity. This option can be useful for the major fertile component of reactor fuel, that is, for ^{238}U in pins of a uranium system, or for ^{232}Th for fuel in a Thorium/U233 system.

Because the thermal scattering matrix depends on temperature, the next component of the WIMSR data contains the **NTEMP** versions of the basic cross sections and the P_0 scattering matrix for the thermal groups (28–69 normally). The cross sections included are transport, absorption, nu*fission, and fission. The transport positions contain the sum of the thermal inelastic cross section obtained by summing up the P_0 matrix (MF=6/MT=MTI), the thermal elastic cross section from the diagonal elements of the P_0 matrix (MF=6/MT=MTC), if present, and the absorption cross section. If separate P_1 matrices are not given for this material, the

P_1 cross section obtained by summing the P_1 matrices over secondary groups for each primary group is subtracted. The matrix data are read from sections on the GENDF file with MF=6/MT=MTI and MF=6/MT=MTC (if present). As for the temperature-independent matrices, they can be transport-corrected by subtracting the sum over secondary groups of the P_1 matrix for each primary group from the self-scatter position. Also, minimum and maximum limits on the secondary group are determined for each primary group, and the matrix is compacted for efficiency.

The next part of the WIMS data file contains the resonance data, which were discussed in Section XIX.A.

In some cases, these resonance data are followed by a fission spectrum block. The complications of obtaining the fission spectrum are the same as those described above for obtaining the fission neutron production cross section, $\bar{\nu}\sigma_f$. If the shortcut option was used in GROUPR, the fission spectrum can be read directly from MF=5/MT=452 on the GENDF file. The sum over groups is also accumulated in CNORM in order to allow the final spectrum to be normalized accurately. The shortcut approach neglects the effects of delayed neutron emission on the fission spectrum.

If fission matrices are available (either MT18, or MT19+20+21...), the prompt part of the fission spectrum is obtained by summing $\sigma_{gg'}\phi_g$ over all primary groups, g. These numbers are also summed into CNORM for use later in normalizing χ. The delayed part is obtained as

$$\left\{\sum_g \nu_d \sigma_{fg} \phi_g\right\} \chi_{dg'} \ , \tag{7}$$

which also contributes to the normalization. Note that the energy dependence of the fission matrix is factored into the final χ in proportion to the weighting flux used in GROUPR to prepare the WIMSR input file. For thermal reactor problems, it is easy to provide a good estimate for this weighting flux.

The final block on the WIMS data tape is optional. If present, it contains P_1 scattering matrices for each temperature. These matrices are defined over the entire group range (normally 1–69), and they contain both the temperature-independent and temperature-dependent reactions in each matrix. The methods used to build up these matrices are parallel to those discussed above for the P_0 matrices. Note that if P_1 matrices are given, the transport corrections are <u>not</u> included in the transport cross sections or P_0 matrices.

C. Burn Data

WIMS uses a simplified burn model for tracking the production and depletion of actinides and fission products, and the chains used are hard-wired into the code. WIMSR provides a method to enter new fission-yield data into the WIMS library format, but it has not been used or tested very much so far.

D. User Input

The following user input specifications were copied from the comment cards at the beginning of the WIMSR source. It is always a good idea to check the comment cards in the current version to see if there have been any changes.

```
C     *---INPUT SPECIFICATIONS (FREE FORMAT)-------------------------*
C     *                                                               *
C     * CARD 1                                                        *
C     *    NGENDF  UNIT FOR INPUT GENDF TAPE                          *
C     *    NOUT    UNIT FOR OUTPUT WIMS TAPE                          *
C     *                                                               *
C     * CARD 2                                                        *
C     *    IPRINT  PRINT OPTION                                       *
C     *               0=MINIMUM (DEFAULT)                             *
C     *               1=REGULAR                                       *
C     *               2=1+INTERMEDIATE RESULTS                        *
C     *    IVERW   WIMS VERSION                                       *
C     *               4=WIMS-D (DEFAULT)                              *
C     *               5=WIMS-E                                        *
C     *    IGROUP  GROUP OPTION                                       *
C     *               0=69 GROUPS (DEFAULT)                           *
C     *               9=USER'S CHOICE                                 *
C     *                                                               *
C     * CARD 2A  (IGROUP.EQ.9 ONLY)                                   *
C     *    NGND    NUMBER OF GROUPS                                   *
C     *    NFG     NUMBER OF FAST GROUPS                              *
C     *    NRG     NUMBER OF RESONANCE GROUPS                         *
C     *    IGREF   REFERENCE GROUP (DEFAULT IS LAST FAST GROUP)       *
C     *                                                               *
C     * CARD 3                                                        *
C     *    MAT     ENDF MAT NUMBER OF THE MATERIAL TO BE PROCESSED    *
C     *    NFID    IDENTIFICATION OF MATERIAL FOR THE WIMS LIBRARY    *
C     *    RDFID   IDENTIFICATION NUMBER FOR THE RESONANCE DATA       *
C     *    IBURN   BURNUP DATA OPTION                                 *
C     *               0=NO BURNUP DATA IS PROVIDED (DEFAULT)          *
C     *               1=BURNUP DATA IS PROVIDED IN CARDS 5 AND 6      *
C     *                                                               *
C     * CARD  4                                                       *
C     *    NTEMP   NO. OF TEMPERATURES TO PROCESS FOR THIS MATERIAL   *
C     *               (0=ALL FOUND ON INPUT TAPE)                     *
C     *    NSIGZ   NO. OF SIGMA ZEROES TO PROCESS FOR THIS MATERIAL   *
C     *               (0=ALL FOUND ON INPUT TAPE)                     *
```

```
C     *    SGREF   REFERENCE SIGMA ZERO                                *
C     *    IRES    RESONANCE ABSORBER INDICATOR (0=NO,1=YES)           *
C     *    SIGP    POTENTIAL CROSS SECTION FROM ENDF/B.                *
C     *    MTI     THERMAL INELASTIC MT (DEFAULT=0=NONE)               *
C     *    MTC     THERMAL ELASTIC MT (DEFAULT=0=NONE)                 *
C     *    IP1OPT  INCLUDE P1 MATRICES                                 *
C     *               0=YES                                            *
C     *               1=NO, CORRECT P0 INGROUPS (DEFAULT)              *
C     *    INORF   RESONANCE FISSION (IF FOUND)                        *
C     *               0=INCLUDE RESONANCE FISSION (DEFAULT)            *
C     *               1=DO NOT INCLUDE                                 *
C     *    ISOF    FISSION SPECTRUM                                    *
C     *               0=DO NOT INCLUDE FISSION SPECTRUM (DEFAULT)      *
C     *               1=INCLUDE FISSION SPECTRUM                       *
C     *    IFPROD  FISSION PRODUCT FLAG                                *
C     *               0=NOT A FISSION PRODUCT (DEFAULT)                *
C     *               1=FISSION PRODUCT, NO RESONANCE TABLES           *
C     *               2=FISSION PRODUCT, RESONANCE TABLES              *
C     *                                                                *
C     *   FOLLOWING TWO CARDS ARE FOR IBURN GT 0 ONLY                  *
C     * CARD  5                                                        *
C     *    NTIS    NO. OF TIME-DEPENDENT ISOTOPES                      *
C     *    EFISS   ENERGY RELEASED PER FISSION                         *
C     *                                                                *
C     * CARD  6  (REPEAT THIS CARD NTIS TIMES)                         *
C     *    IDENTA  IDENT OF FISSION PRODUCT ISOTOPE                    *
C     *    YIELD   FISSION YIELD OF IDENTA FROM BURNUP OF MAT          *
C     *                                                                *
C     * CARD  7                                                        *
C     *    LAMBDA  RESONANCE-GROUP GOLDSTEIN LAMBDAS (13 FOR           *
C     *            DEFAULT 69-GROUP STRUCTURE, NRG OTHERWISE).         *
C     *                                                                *
C     ******************************************************************
```

The first card specifies the input and output unit numbers, as is normal for NJOY modules. NGENDF comes from a previous GROUPR run, and it can be in either binary or ASCII mode. NOUT is always in ASCII mode.

The options card allows the user to select how much detail will be printed on the output listing (IPRINT), whether the output is intended for WIMS-D or WIMS-E (IVERW), and how many groups are desired. Currently, the only difference between WIMS-D and WIMS-E output is that some additional reaction cross sections are included for the latter. If the user selects some group structure different from the standard 69-group structure, an additional input card is required to give the number of groups (NGND), the number of fast groups (NFG, 14 for 69 groups), the number of resonance groups (NRG, 13 for 69 groups), and the reference group used for normalizing the flux (IGREF, normally the low-energy group of the fast groups).

Card three is always required. It gives the ENDF MAT number for the materials to be processed. If this MAT doesn't appear on the GENDF tape, a fatal error message will be issued. NFID will be the identification number for this material used on the output WIMS library, and RDFID will be the identification number for the resonance data. Formally, WIMS libraries allow for data sets with more than one version of the resonance-integral tabulation. The last parameter on this card is IBURN to flag whether burn data are included in the input stream.

The next card starts out with NTEMP and NSIGZ, which define the size of the resonance-integral tables. They are normally both set to zero; the code then uses all of the values computed by GROUPR. The reference sigma-zero value, SGREF, is used for the elastic cross section and matrix, because these quantities are not normally self-shielded by WIMS. Normally, 1E10 is appropriate, but for the major fissionable material in the reactor (*i.e.*, U-238 or Th-232), it may be better to use a realistic number like SGREF=50. WIMSR assumes that the potential scattering cross section for the material is constant, but this constant value is not available from the GROUPR output. The WIMSR user will have to look in the ENDF-format evaluation for the scattering length a, compute $\sigma_p = 4\pi a^2$, and enter the value as SIGP. The following fragment shows where to find the scattering length (9.56630- 1 in this case):

```
...
9.22350+ 4 2.33025+ 2          0          0          1          01395 2151
9.22350+ 4 1.00000+ 0          0          1          2          01395 2151
1.00000+ 0 8.20000+ 1          1          1          0          01395 2151
3.50000+ 0 9.56630- 1          0          0          1          01395 2151
2.33025+ 2 0.00000+ 0          0          0        780        1301395 2151
...
```

The parameters MTI and MTC select the thermal inelastic and elastic data from the sections that might be available on the GENDF tape. Most materials have only free-gas scattering available, and the appropriate values would be MTI=221 and MTC=0. The conventional values to use for reactor moderator materials are given in Table 2.

WIMSR allows P_1 scattering to be treated in two ways. If IP1OPT=1, the P_1 matrix for the material is written to the WIMS output file explicitely. This option is normally used only for major moderator materials, such as the components of water. The other option, IP1OPT=1, instructs the code to use the P_1 data to transport-correct the P_0 elastic scattering matrix; that is, the ingroup elements of the P_0 matrix are reduced by the sum over all outgoing groups of the P_1 matrix for that ingoing group.

Table 2: Conventional Values for the Thermal MT Numbers (MTI and MTE) Used in WIMSR, GROUPR, and THERMR

Thermal Material	MTI Value	MTC Value
H in H_2O	222	
D in D_2O	228	
Be metal	231	232
Graphite	229	230
Methane	227	
Zr in ZrH	235	236
H in ZrH	225	226
BeO	233	234
H in Polyethylene	223	224

The INORF parameter can be set to 1 to eliminate the resonance-integral table for nu*fission from the WIMS output. Some of the higher actinides are treated this way for some WIMS libraries. The ISOF flag is set to 1 to tell WIMSR to produce a fission spectrum. This is usually done for main fissile materials, such as ^{235}U and ^{239}Pu. The IFPROD flag is used to control whether resonance tables are included for fission products.

WIMSR has some capability to format burn data for incorporation into a WIMS library (see cards 5 and 6). This part of the code has not been used or tested very much.

The final card gives the intermediate-resonance λ values for each of the resonance groups. Methods for obtaining these quantities with NJOY are outlined in Section XIX.A.

E. Coding Details

WIMSR starts by allocating dynamic storage for NAMAX=100000 words. Note that the first 7500 words of this area are reserved for a scratch area called SCR. In some cases, error messages may appear that suggest that this number be increased. The scratch files needed by WIMSR use the unit numbers from 10 through 14.

The next step is to read and echo the user's input. Subroutine WMINIT is then called. It looks at the record MF=1/MT=451 for the desired material on the input GENDF tape to obtain the group structure. It then reads through the entire GENDF tape for this material to set the fission flags I318 and I618 and to

count the number of temperatures that are available. The fission flag is used to handle cases where both MT18 and the partial fission representation (MT19, 20, ...) appear on the GENDF tape.

Subroutine **RESINT** is called next to compute the WIMS resonance integrals from the GROUPR self-shielded cross sections. It reads through the entire GENDF tape and extracts the flux, absorption cross section, fission cross section, and elastic cross section versus temperature and sigma-zero for all of the resonance groups. It also extracts the reference-group flux versus temperature and sigma-zero and the fission $\overline{\nu}$ value. The latter is computed from File 6 with delayed contributions from MT=455 in File 3. Once all the data are in place, **RESINT** uses the $\overline{\nu}$ values to convert the self-shielded fission cross sections into self-shielded $\overline{\nu}\sigma_f$ values.

If the user has asked for a value of **NTEMP** that is larger than the number of temperatures on the GENDF tape for this material, **RESINT** will duplicate the values from the last temperature given into the higher temperature positions for the flux and the absorption cross section.

The final step in **WMINIT** is to convert the self-shielded cross sections into resonance integrals using the method described in Section XIX.A. These resonance integrals are written out to a scratch file and displayed on the output listing using subroutine **RSIOUT**. The elastic resonance integrals are only written for WIMS-E.

Next, the main program calls **XSECS** to process the cross sections. The outermost loop is over temperature. A distinction has to be made between the temperature-independent matrix data, such as (n,2n) and (n,n′) reactions, and the temperature-dependent matrix data, such as thermal scattering. While reading through the GENDF tape, the following quantities are extracted and stored using the pointer indicated:

IABS1	radiative capture (MT102);
IABS2	other absorption reactions (MT=103–150);
LOCSFO	the fission cross section (MT18);
LOCABO	also the fission cross section (MT18);
IN2N	the (n,2n) cross section;
ISCAT	the elastic scattering cross section (MT2), possibly using the value corresponding to the reference sigma-zero value instead of infinite dilution;
IXI	the log slowing down ξ (MT252);
LOCNUS	the fission yield $\overline{\nu}\sigma_f$ computed from either 3/452 or File 6 plus the delayed part from 3/455;

LOCCHI	the fission spectrum χ computed from either 5/455 or File 6 plus the delayed contributions from MT455;
XS	the temperature-dependent scattering matrix, containing MTI and MTC, the thermal inelastic and elastic reactions, respectively;
L1	the smallest group number for a nonzero element of the nonelastic part of the matrix stored in XS;
L1E	the smallest group number for a nonzero element of the elastic part of the matrix stored in XS;
L2	the largest group number for a nonzero element of the nonelastic part of the matrix stored in XS;
L2E	the largest group number for a nonzero element of the elastic part of the matrix stored in XS; and
CSIGP1	the P_1 cross section for the thermal matrix obtained by summing over the P_1 matrix elements for scattering from each group.

Once all the data for a temperature have been stored in memory, several additional operations are performed on them. The initial absorption cross section located at LOCABO contains the fission cross section. The final value is formed by adding the data at IABS1 and IABS2, and then subtracting the (n,2n) cross section at IN2N. The final transport cross section located at LOCXTR is formed by adding the absorption and subtracting the transport correction (CSIGP1). The slowing-down power per unit lethargy is computed from ξ at IXI, σ_{scat} at ISCAT, and the group boundary energies at IEGB.

As usual, the treatment of fission is more complex. For some combinations of options, $\overline{\nu}$ is computed by dividing the fission neutron production cross section by the fission cross section, and for others, the value of $\overline{\nu}\sigma_f$ has to be computed from $\overline{\nu}$ and σ_f. In addition, the fission spectrum, if requested, is normalized.

Subroutine XSECO starts with a section that writes and prints the temperature-independent part of the WIMS data. This section is skipped when XSECO is called with ITEMP>1. It first processes the temperature-independent vectors: potential scattering, slowing-down power, transport, absorption, IR lambda, and sometimes (n,2n). Next, it processes the fission vectors nu*sigf and sigf. The temperature-independent part of the scattering matrix includes the fast groups and the resonance groups (this normally totals 27 groups). It is packed by retrieving the low (LONE) and high (LTWO) limits for each band of nonzero elements from L1 and L2, which were loaded in XSECS. They are used to compute the number of elements in the band and the location of the self-scatter element in the band (always 1 for

this part of the matrix, because there is no upscatter). They are also used to direct how the numbers in XS are moved into ISCR with the zeros outside of the band removed. Note that the data are stored as follows: location of self-scatter for group 1, number of elements in band for group1, the band of elements for group 1, location of self-scatter for group 2, number of elements in group 2, the band of elements for group 2, etc. If the number of elements in a band is zero, the two counts are there, but no band data are given. After being printed, the temperature-independent part of the matrix is written out on a scratch file NSCR2.

The next part of XSECO is executed for ITEMP=1 and all the higher temperatures. It prints and writes the temperature-dependent transport, and absorption cross sections (they are defined in the thermal range only, normally groups 28–69). Note that the absorption cross section is also written out on scratch file NSCR3. If available, the temperature-dependent fission neutron production cross section and fission cross section are also printed and written. The temperature-dependent scattering matrix is processed as described above, except that the incident-neutron group range is limited to the thermal range (normally 28–69). Note that the location of the self-scatter element will no longer be 1 for these data, because of the presence of upscatter. The packed matrix length and the packed data are written out to scratch file NSCR3.

If the user has requested that P_1 scattering matrices be constructed for the material, subroutine P1SCAT is called. It uses methods similar to those described above. The results are printed and written onto NSCR4 by P1SOUT.

The last step in WIMSR is to call WIMOUT to prepare the final WIMS data library. The first card on NOUT is slightly different for WIMS-D and WIMS-E. It is followed by lines for the burnup data using numbers obtained from common storage.

The data needed for the material identifier card are available in common storage. The temperature-independent data are read from NSCR2 and written to NOUT. Similarly, the temperature-dependent data are read from NSCR3 (although the TEMPR array is obtained from common storage). For WIMS-D, a record mark is written at this point.

The resonance data, if needed, are read from NSCR1 and written to NOUT. The format is slightly different for WIMS-D and WIMS-E. The WIMS-D version has extra lines containing NTNP, the product of the number of temperatures and the number of sigma-zero values, and it has a record mark after the resonance data block. The WIMS-E version has an extra section of resonance-integral data for computing the self-shielded elastic scattering cross section.

If a fission spectrum was requested, it is written out next by using data from the dynamic storage pointer IUFF. If a P_1 matrix was requested, it is read in from NSCR4 and written out onto NOUT.

This completes the entire WIMS library. The final step takes place in the main WIMSR program, where the normal timing and storage usage messages are printed.

F. WIMS Data File Format

The following section describes the WIMS data output provided by WIMSR. It consists of a number of logical blocks of information written out in coded form. The output is intended to be used by a library maintenance code to prepare a binary library for use by the WIMS code.

Library Header (2I5)

NFID	material identifier
NPOS	position to insert material on a large multimaterial library (given for WIMS-E only)

Burnup Data (3(1PE15.8,I6))

(YIELD(I),IFISP(I),I=1,JCC/2 fission yields and fission product flag

Material Identifier Data (I6,1PE15.8,5I6)

IDENT	material identifier
AWR	atomic weight ratio to neutron
IZNUM	atomic charge number Z for this material
IFIS	fission and resonance flag: 0=non-fissile with no resonance tabulation, 1=non-fissile with resonance absorption only, 2=fissile with resonance absorption only (*e.g.*, Pu-240), 3=fissile with resonance absorption and fission, 4=fissile with no resonance tabulations.
NTEMP	number of temperatures
NRESTB	number of resonance tables included (0 or 1)
ISOF	fission spectrum flag (0=no, 1=yes)

Temperature-Independent Vectors (1P5E15.8)

(SIGP(I),I=1,N2)	σ_p for resonance groups
(XX(I),I=1,N2)	$\xi\sigma_s/\tau$ for resonance groups
(XTR(I),I=1,N1+N2)	transport cross section for fast and resonance groups
(ABS(I),I=N1+N2)	absorption cross section for fast and resonance groups
(DUM,I=1,N2)	unused dummy for resonance groups
(ALAM(I),I=1,N2)	IR λ values for resonance groups

Temperature-Independent Fission (IFIS>1 only) (1P5E15.8)

(NSIGF(I),I=1,N1)	$\overline{\nu}\sigma_f$ for fast and resonance groups
(SIGF(I),I=1,N1)	σ_f for fast and resonance groups

P_0 Matrix Length (I15)

NDAT	length of P_0 scattering block to follow

Temperature-Independent P_0 Matrix (1PE15.8)

(XS(I),I=1,NDAT)	packed scattering data: IS for group 1, NS for group 1, NS scattering elements for group 1, IS for group 2, NS for group 2, etc., through all of the fast and resonance groups (normally through group 27). IS is the position of self-scatter in the band of scattering elements (always 1 here), and NS is the number of elements in the band.

Temperature Values (1PE15.8)

(TEMP(I),I=1,NTEMP)	temperatures in Kelvin

Repeat the following 4 blocks for each of the NTEMP *temperatures.*

Temperature-Dependent Transport and Absorption (1PE15.8)

(XTR(I),I=1,N3)	transport cross section for thermal groups
(ABS(I),I=1,N3)	absorption cross section for thermal groups

Temperature-Dependent Fission Vectors (1PE15.8)

(NSIGF(I),I=1,N3)	fission neutron production cross section $\overline{\nu}\sigma_f$ for thermal groups
(SIGF(I),I=1,N3)	fission cross section σ_f for thermal groups

Temperature-Dependent P_0 Matrix Length (I15)

NDAT	length of P_0 scattering block

Temperature-Dependent P_0 Matrix (1PE15.8)

XS(I),I=1,NDAT)	packed scattering data: IS for group N, NS for group N, NS scattering elements for group N, IS for group N+1, NS for group N+1, etc., through the last group (normally group 69). IS is the position of self-scatter in the band of scattering elements, and NS is the number of elements in the band.

Record Mark (' 999999999999'), WIMS-D only

Resonance Control Data

RID	resonance set identifier
NTEMP	number of temperatures for this resonance set
NSIGZ	number of background cross section values

Absorption Resonance Data

(TEMP(I),I=1,NTEMP)	temperatures
(SIGZ(I),I=1,NSIGZ)	background cross section values
(RESA(I),I=1,NTEMP*NSIGZ	absorption resonance integrals

Nu*fission Resonance Data

(TEMP(I),I=1,NTEMP)	temperatures
(SIGZ(I),I=1,NSIGZ)	background cross section values
(RESNF(I),I=1,NTEMP*NSIGZ	nu*fission resonance integrals

Scattering Resonance Data, WIMS-E only

(TEMP(I),I=1,NTEMP)	temperatures
(SIGZ(I),I=1,NSIGZ)	background cross section values
(RESS(I),I=1,NTEMP*NSIGZ	scattering resonance integrals

Record Mark (' 999999999999'), WIMS-D only

Fission Spectrum

(FSPECT(I),I=1,NGND)	fission spectrum χ

Repeat the following two blocks
for each of NTEMP *temperatures.*

P_1 Matrix Length (I15)

NDAT	length of P_1 scattering block

P_1 Matrix (1PE15.8)

XS(I),I=1,NDAT)	packed scattering data: IS for group 1, NS for group 1, NS scattering elements for group 1, IS for group 2, NS for group 2, etc., through the last group (normally group 69). IS is the position of self-scatter in the band of scattering elements, and NS is the number of elements in the band.

G. WIMSR Auxiliary Codes

The WIMSR output as described above is not directly usable by WIMS. Two library-maintenance codes are used at Los Alamos. FIXER is used to modify (fix up) an existing WIMS-D library, or to create a new one, using WIMSR output. It processes burnup data, main data, resonance data, and P_1 matrices. In its fix-up mode, it can replace, delete, or add a material. WRITER is a code to read a WIMS-D library in coded form, convert it to binary form, and list it in a user readable form.

H. Error Messages

ERROR IN WMINIT***DESIRED MATERIAL IS NOT ON GENDF TAPE

Check whether the right input GENDF input tape was mounted.

ERROR IN WMINIT***INCORRECT GROUP STRUCTURE

The group structure on the input GENDF file does not agree with the one specified in the WIMSR input deck.

MESSAGE FROM WMINIT***MAT xxxx MF xx HAS BOTH MT18 AND ...

If both MT18 and MT19 are present in File 3 or File 6, WIMSR must make a choice of which to use. For materials with partial fission reactions, GROUPR normally does not process the fission matrix from MT18. It is more accurate to use the sum of the partial fission matrices.

ERROR IN RESINT***STORAGE EXCEEDED

It will be necessary to increase NWSCR, which is currently set to 7500 by a statement in the WIMSR routine.

ERROR IN XSECS***STORAGE EXCEEDED

It will be necessary to increase NWSCR, which is currently set to 7500 by a statement in the WIMSR routine.

MESSAGE FROM XSECS***USE ONLY xx TEMPS FOR MAT xxxx

The thermal inelastic sections are missing from some of the higher temperatures on the input GENDF tape. This message tells you how many temperatures can be used correctly. Check the THERMR and GROUPR runs if more temperatures are needed.

ERROR IN XSECO***SCRATCH STORAGE EXCEEDED

It will be necessary to increase NWSCR, which is currently set to 7500 by a statement in the WIMSR routine.

ERROR IN P1SCAT***STORAGE EXCEEDED

It will be necessary to increase NWSCR, which is currently set to 7500 by a statement in the WIMSR routine.

ERROR IN P1SCAT***NO P1 MATRICES FOUND FOR MAT xxxx

Check the GROUPR run to make sure that P_1 matrices have been requested for the desired materials.

ERROR IN P1SCAT***NO TEMPERATURE-DEPENDENT REACTIONS ...

Check the GROUPR run to make sure that the elastic scattering matrix was requested.

I. Acknowledgments

Rosemary Boicourt did a lot of the original coding on WIMSR (based on the CPM option in POWR). Since then, WIMSR has been reworked and improved several times by MacFarlane and Fortunato Aguilar of the Instituto Nacional de Investigaciones Nucleares (ININ) working in both Mexico and Los Alamos. Fred Mann of Westinghouse-Hanford, Corneliu Costescu of the University of Illinois, and Andrej Trkov of the Institute Jozef Stefan, Ljubljana, Slovenia have also made important contributions. The original impetus for including WIMSR in NJOY

came from the WIMS users at Atomic Energy of Canada, Ltd. (AECL), and their help was very useful in the early phases of the work.

J. References

1. J. R. Askew, F. J. Fayers, and P. B. Kemshell, "A General Description of the Lattice Code WIMS," *J. Brit. Nucl. Energy Soc.* 5, p. 564 (1966).

XX. PLOTR

The PLOTR module provides a general-purpose plotting capability for ENDF, PENDF, and GENDF files by making use of the DISSPLA graphics system.* PLOTR is useful for getting a quick look at the results of other NJOY modules, for focusing down to look at details of evaluations or derived data, and for preparing publication-quality figures of ENDF evaluations and NJOY results. Some of the types of graphs that PLOTR can produce include

- conventional 2-D plots (for example, cross section vs. energy) of ENDF, PENDF, or GENDF data with the normal combinations of linear and log scales, automatic or user-specified ranges and labels, an optional alternate right-hand axis, and with one or two title lines;
- a set of experimental data by itself or superimposed on ENDF, PENDF, or GENDF curves;
- curves of various patterns, labeled with tags and arrows or described in a legend block;
- data points given with a variety of symbols with error bars (they can be identified in a legend block);
- dramatic 3-D perspective plots of File 4 or File 6 angular distributions with a choice of a linear or a log axes for incident energy and a choice of energy range and viewpoint;
- selected 2-D plots of File 5 and File 15 emission spectra for specified incident energies, and selected 2-D emission spectra for given energies and particle types for File 6 data;
- detailed 3-D perspective plots of File 5, 6, or 15 energy distributions with a choice of log or linear axes and viewpoint (both $EE'\theta$ and $E\theta E'$ laws are supported);
- 3-D plots of GENDF data; and
- various 2-D plots for File 7 data, including both symmetric and asymmetric $S(\alpha,\beta)$ vs. either α or β.

This chapter describes PLOTR 91.91.

Methods for generating these types of plots will be given in the following subsections. An attempt has been made to keep the input as simple as possible by moving the less common options to the right-hand side of each input line so that they can be easily defaulted. A complete copy of the input instructions will be found in Section XX.H. It may be useful to refer to it occasionally while reading the following sections.

*DISSPLA (Display Integrated Software and Plotting Language) is a product of ISSCO Graphics (Integrated Software Systems Corporation), San Diego, CA.

A. Simple 2-D Plots

The simplest kind of 2-D plot is for a single reaction from an ENDF, PENDF, or GENDF file using automatic scales and default labels. For example, to plot the total cross section of carbon from ENDF/B-V Tape 511, use the following input (don't type the line numbers; they are inserted here for reference):

```
 1.  0
 2.  5
 3.  PLOTR
 4.  /
 5.  1/
 6.  *ENDF/B-V CARBON*/
 7.  *T<OTAL> C<ROSS> S<ECTION>*/
 8.  4/
 9.  /
10.  /
11.  /
12.  /
13.  5 20 1306 3 1/ TAPE20 is T511
14.  /
15.  99/
16.  STOP
```

The data to be plotted are selected in line 13 using the normal MAT, MF, MT notation of ENDF. The "slash" at the end of the line hides several defaults, the first of which is the temperature, which defaults to 0 K. The "4" in line 8 selects log-log axes (a number of other options are defaulted here also). Lines 9 through 12 are blank, resulting in the choice of automatically defined ranges and default labels. Two title lines are given on lines 6 and 7. Note the use of special shift characters to change between uppercase and lowercase. The result is shown in Fig. 1.

In many cases, the default scales will give reasonable plots. However, in this case, the low-energy portion of the plot is approximately constant. It makes sense to change the lower limit of the x axis in order to expand the amount of detail shown at higher energies. A slight change in the lower limit of the y axis would also be beneficial. It is only necessary to change two lines as follows:

```
 9.  1E4 2E7/
11.  .5 10/
```

Note that the third parameter on these axes cards should always be defaulted for log scales. The results are shown in Fig. 2. This is a better balanced plot.

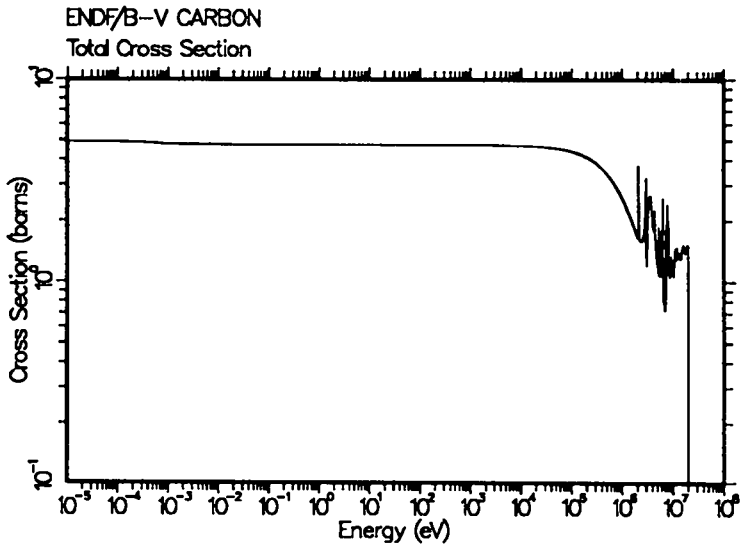


Figure 1: Simple 2-D plot of the total cross section of ENDF/B-V carbon using automatic log-log axes.

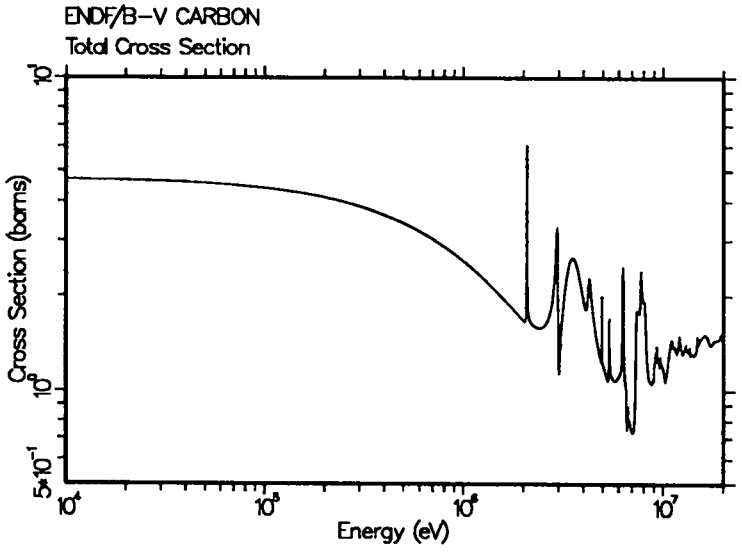


Figure 2: Simple 2-D plot of the total cross section of ENDF/B-V carbon using log-log axes with user-selected ranges.

If the user needs to emphasize the high-energy region, linear scales are more appropriate. In addition, the single-stroke fonts might be too simple for some people's preferences. The following input gives the results shown in Fig. 3. Note how the new font is specified in line 4. The linear-linear axes option is selected in line 5.

```
 1.  0
 2.  5
 3.  PLOTR
 4.  10 8 3/
 5.  1/
 6.  *ENDF/B-V CARBON*/
 7.  *T<OTAL> C<ROSS> S<ECTION>*/
 8.  1/
 9.  /
10.  /
11.  /
12.  /
13.  5 20 1306 3 1/ TAPE20 is T511
14.  /
15.  99/
16.  STOP
```

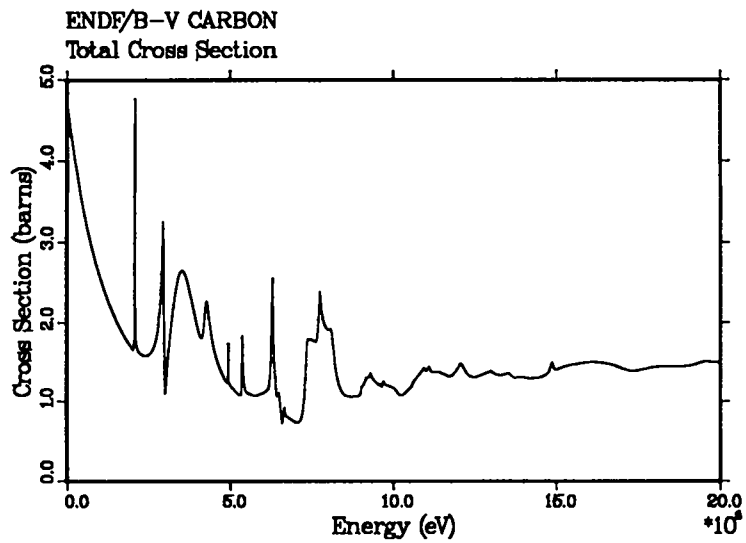


Figure 3: Simple 2-D plot of the total cross section of ENDF/B-V carbon using linear axes to emphasize the high-energy region and a more elaborate font.

The limits for the linear x axis in this example could have been specified by the user with a card of the form

```
9.  0 1E7 2E6/  .
```

The general rule for linear axes is either give all three parameters explicitly, or default all three parameters. For log axes, either give the first two parameters and default the third, or default all three parameters.

Most ENDF or PENDF reactions will have many more energy points than can be shown on a graph like those in these figures. Therefore, PLOTR "thins" the grid down until there are fewer than **MAX** points* on the plot (**MAX** is currently 2000). On the other hand, at some energies some ENDF reactions are described on fairly coarse energy grids using interpolation laws like "lin-lin" or "log-log". These representations will look as the evaluator intended if they are plotted using corresponding scales (for example, log-log interpolation on log-log scales, or log-lin interpolation on log-lin scales), but if they were to be plotted on a different set of axes, the cross sections between the grid points would be different from those intended. Therefore, PLOTR "thickens" the energy grid by adding additional energy points between the grid points of the evaluation and computing the cross section at each of these points from the given interpolation law. The resulting curves will be faithful to the evaluation, but they may exhibit unphysical bumps and cusps in certain modes of presentation. An example is shown in Fig. 4. Most evaluators try to use a fine enough grid to minimize these ugly artifacts.

B. Multicurve and Multigroup Plots

Several curves can be drawn on each set of axes, and each curve can be taken from a different data source. The following input deck demonstrates how GENDF data can be compared with PENDF data by overplotting:

```
1.  0
2.  5
3.  PLOTR
4.  /
5.  1/
6.  *ENDF/B-V +EH.5>235+EXHX>U*/
7.  /
8.  4 0 2 1 5E3 500/
9.  .1 2E7/
```

*A note about typography: the convention for Cray FORTRAN programs and input decks at Los Alamos is to use all lowercase characters. This convention has been followed in this report. In addition, all input lines, FORTRAN variable names, and computer code sequences are set in a "typewriter" font so that they can be distinguished from normal text.

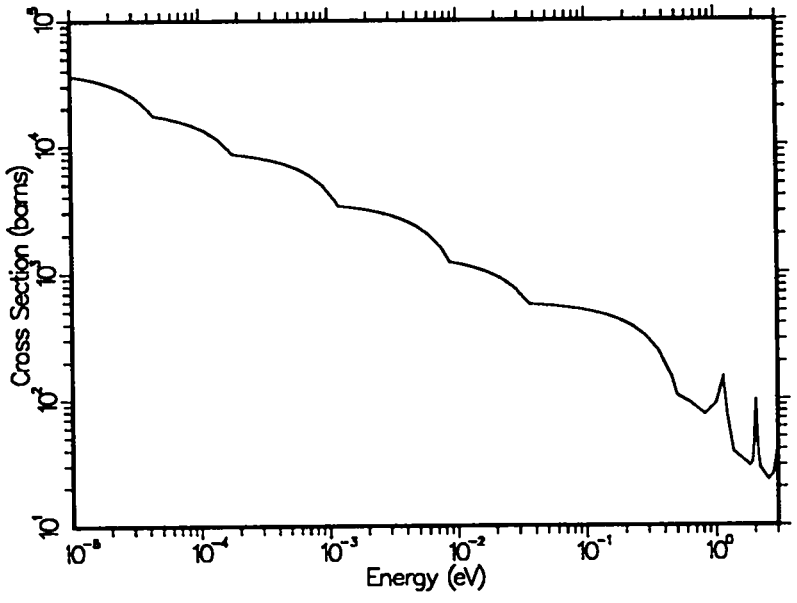


Figure 4: An example of the bumps and cusps seen when coarsely interpolated data is displayed using axes different from those appropriate for the particular interpolation law used. This example is for linearly interpolated data shown on log-log axes. The curve is a PENDF representation of the total cross section of ENDF/B-V ^{235}U at 300 K generated with tolerances of 50%.

```
10.  /
11.  1 1000/
12.  /
13.  5 23 1395 3 18 300/
14.  /
15.  *<POINTWISE FISSION>*/
16.  2/
17.  1 24 1395 3 18 300 1 1 0/
18.  0 0 1/
19.  *<MULTIGROUP FISSION>*/
20.  99/
21.  STOP
```

The result is shown in Fig. 5. The PENDF data are requested on Card 13, and the GENDF data are requested on Card 17. Note the use of **IVERS**=1 to denote GENDF data, and also note the settings for **NTH**, **NTP**, and **NKH** necessary to select the P_0 infinitely dilute cross section for plotting. The GENDF-format data are automatically converted into histogram form for plotting. This example also demonstrates using a "legend" block to identify the two curves. The position for the legend is given on Card 8. These values are normally determined by trial and error. Note also the presence of a superscript in the title. The superscript depends on the DISSPLA "instruction" mode, which is described in the DISSPLA documentation.

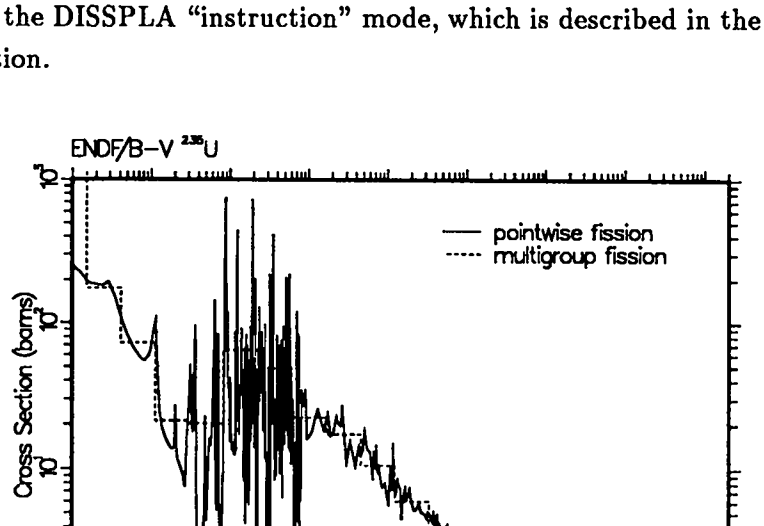


Figure 5: Comparison of the multigroup fission cross section of ENDF/B-V ^{235}U (dashed curve) with the corresponding pointwise cross section from the PENDF tape (solid curve).

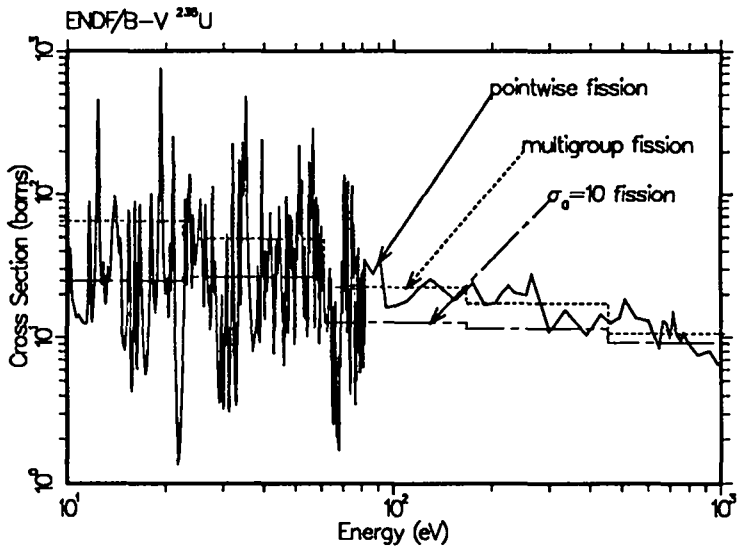


Figure 6: Multigroup fission cross sections of ENDF/B-V ^{235}U for infinite dilution and a 10-barn background compared with the corresponding pointwise cross section.

As another example of a plot of multigroup data, Fig. 6 shows both infinitely dilute and self-shielded cross sections, and the plot also compares them with the pointwise cross section. In addition, this plot uses "tags" and arrows to identify the different curves. The position of the tags and the x location for the arrowhead usually must be determined by trial and error. See lines 16, 21 and 26. The input for this example follows:

```
 1.  0
 2.  5
 3.  PLOTR
 4.  /
 5.  1/
 6.  *ENDF/B-V +EH.5>235+EXHX>U*/
 7.  /
 8.  4 0 2 2/
 9.  10 1000/
10.  /
11.  1 1000/
12.  /
13.  5 23 1395 3 18 300/
14.  /
15.  *<POINTWISE FISSION>*/
16.  200 500 90/
17.  2/
18.  1 24 1395 3 18 300 1 1 0/
19.  0 0 1/
```

```
20.  *<MULTIGROUP FISSION>*/
21.  250 200 110/
22.  3/
23.  1 24 1395 3 18 300 1 3 0/
24.  0 0 2/
25.  *#S+LH.5>0+LXHX>=10 <FISSION>*/
26.  300 90 130/
27.  99/
28.  STOP
```

These two examples require 300 K PENDF data for ^{235}U on the input file TAPE23 and 300 K multigroup data for $\sigma_0 = \infty$, $\sigma_0 = 500$ barns, and $\sigma_0 = 10$ barns on TAPE24. These tapes can be generated with the following NJOY input deck:

```
 1.  0
 2.  5
 3.  RECONR
 4.  20 21/ TAPE20 is T511
 5.  /
 6.  1395/
 7.  .1/
 8.  0/
 9.  BROADR
10.  21 22
11.  1395 1/
12.  .1/
13.  300
14.  0/
15.  UNRESR
16.  20 22 23
17.  1395 1 3 1
18.  300
19.  1E10 500 10
20.  0/
21.  GROUPR
22.  20 23 0 24
23.  1395 3 0 3 1 1 3 1
24.  /
25.  300
26.  1E10 500 10
27.  3 1/
28.  3 18/
29.  3 102/
30.  0/
31.  0/
32.  STOP
```

C. Plotting Input Data

PLOTR allows the user to insert data directly into the input deck. The main use of this is to superimpose experimental data over curves obtained from ENDF, PENDF, or GENDF tapes, but reading data directly from the input deck can also be used to add precalculated curves or eye guides to plots, or to add special features such as vertical lines to separate regions on plots. Experimental data points can be plotted with a variety of symbols, and x and/or y error bars can be included if desired. The error bars can be either symmetric or asymmetric. For experimental data, the curves and various sets of data points are normally identified using a legend block. The following input produces a typical example of this type of plot:

```
 1.  0
 2.  5
 3.  PLOTR
 4.  10 8 2/
 5.  1/
 6.  *ENDF/B-V C<ARBON (N,#A<) WITH FAKE DATA*/
 7.  /
 8.  1 0 2 1 1.4E7 .32/
 9.  /
10.  /
11.  /
12.  /
13.  5 20 1306 3 107/
14.  0 0 0 2/
15.  *ENDF/B-V MAT+K.86<1306*/
16.  2/
17.  0/
18.  -1 0/
19.  *S<MITH? & W<ESSON> +K.86<1914*/
20.  0/
21.  1.1E7 .08 .05 .05/
22.  1.2E7 .10 .05 .05/
23.  1.3E7 .09 .04 .04/
24.  1.4E7 .08 .03 .03/
25.  -1/
26.  3/
27.  0/
28.  -1 2/
29.  *B<LACK> & B<LUE> +K.86<2001*/
30.  0/
31.  1.15E7 .07 .02 0. .2E6 0./
32.  1.25E7 .11 .02 0. .2E6 0./
33.  1.35E7 .08 .015 0. .2E6 0./
34.  1.45E7 .075 .01 0. .2E6 0./
35.  -1/
36.  99/
37.  STOP
```

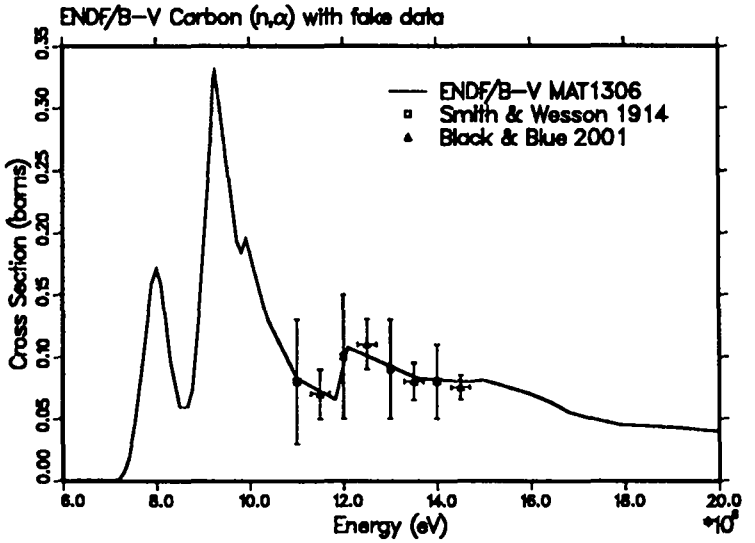


Figure 7: Carbon (n,α) cross section compared with two sets of simulated experimental data represented with two types of error bars.

The results are shown in Fig. 7. Note that the font has been changed to "duplex" for this figure. In addition, the DISSPLA "instruction" option to equally space characters has been used for the dates in the legend (see lines 19 and 29). The error bars for both of these simulated data sets are symmetric, as indicated explicitly in Cards 21 through 24, or by the zeroes in Cards 31 through 34. They can also be asymmetric if the lower and upper (or right and left) values are nonzero and different.

D. Three-D Plots of Angular Distributions

ENDF angular distribution data, whether given in File 4 or File 6, can be very bulky. Therefore, it is useful to present them in the form of a perspective plot showing a family of angular distribution curves for each value of incident particle energy. An input deck to make such a perspective plot follows:

```
 1.  0
 2.  5
 3.  PLOTR
 4.  /
 5.  1/
 6.  *ENDF/B-V CARBON*/
 7.  *E<LASTIC> MF4*/
 8.  -4/
 9.  /
10.  /
```

```
11.  /
12.  /
13.  5 20 1306 4 2/
14.  /
15.  99/
16.  STOP
```

Figure 8 shows the result using a log axis for incident energy. This axis is the x axis, and its range can be limited to expand a particular part of the distribution if desired. A linear scale can be used if the user wants to emphasize the high-energy region.

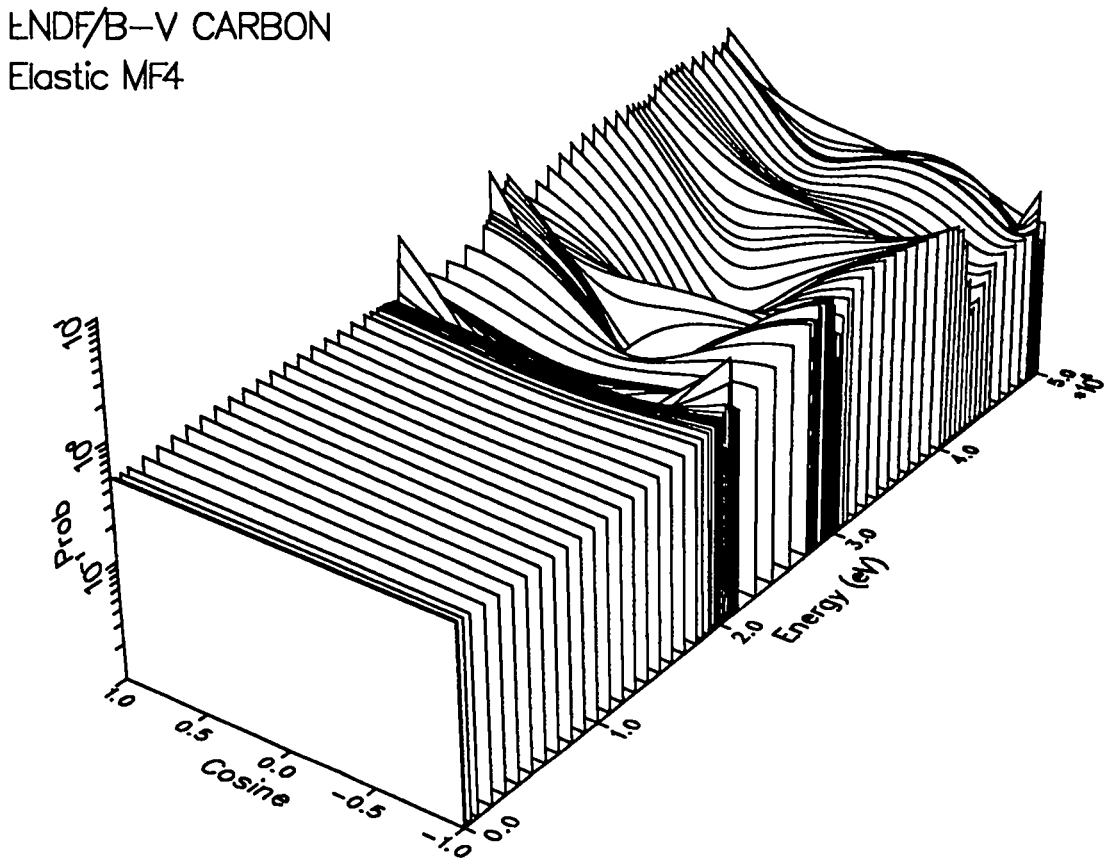


Figure 8: Perspective view of carbon elastic scattering angular distribution.

E. Two-D Angular Distributions Selected from File 4 or File 6

Future versions of PLOTR need a capability to plot individual angular distributions from File 4 or File 6 vs. either μ or θ so that evaluations can be compared with experimental data.

F. Three-D Plots of Energy Distributions

Three-D perspective plots are also useful for energy distributions. For the ENDF-5 and earlier formats, neutron secondary-energy distributions are given in File 5. The following input deck shows how to request a 3-D plot for File 5:

```
 1.  0
 2.  5
 3.  PLOTR
 4.  /
 5.  1/
 6.  *ENDF/B-V L<I>-6*/
 7.  *<(N,2N)#A <NEUTRON DISTRIBUTION*/
 8.  -1/
 9.  /
10.  /
11.  0 12E6 2E6/
12.  /
13.  5 20 1303 5 24/
14.  /
15.  99/
16.  STOP
```

The result is shown in Fig. 9. Similar methods can be used to plot photon emission distributions from File 15.

The new ENDF-6 format also provides for giving distributions for other emitted particles, such a protons, alphas, photons, and even recoil nuclei. This complicates the task of selecting which distribution is to be extracted from File 6. The user must specify the index for the particular outgoing particle to be considered (see NKH). Line 13 might become, for example,

```
13.  6 20 2437 103 0. 0 0 1/  ,
```

which would request a plot of the proton distribution for the (n,p) reaction of ^{54}Cr from ENDF/B-VI. Future versions of PLOTR should allow the particle to be selected by giving a ZA identifier.

For some evaluations, File 6 uses Law 7, where the energy-angle distribution is represented as $E \rightarrow E'$ distributions for several emission cosines μ. In these cases,

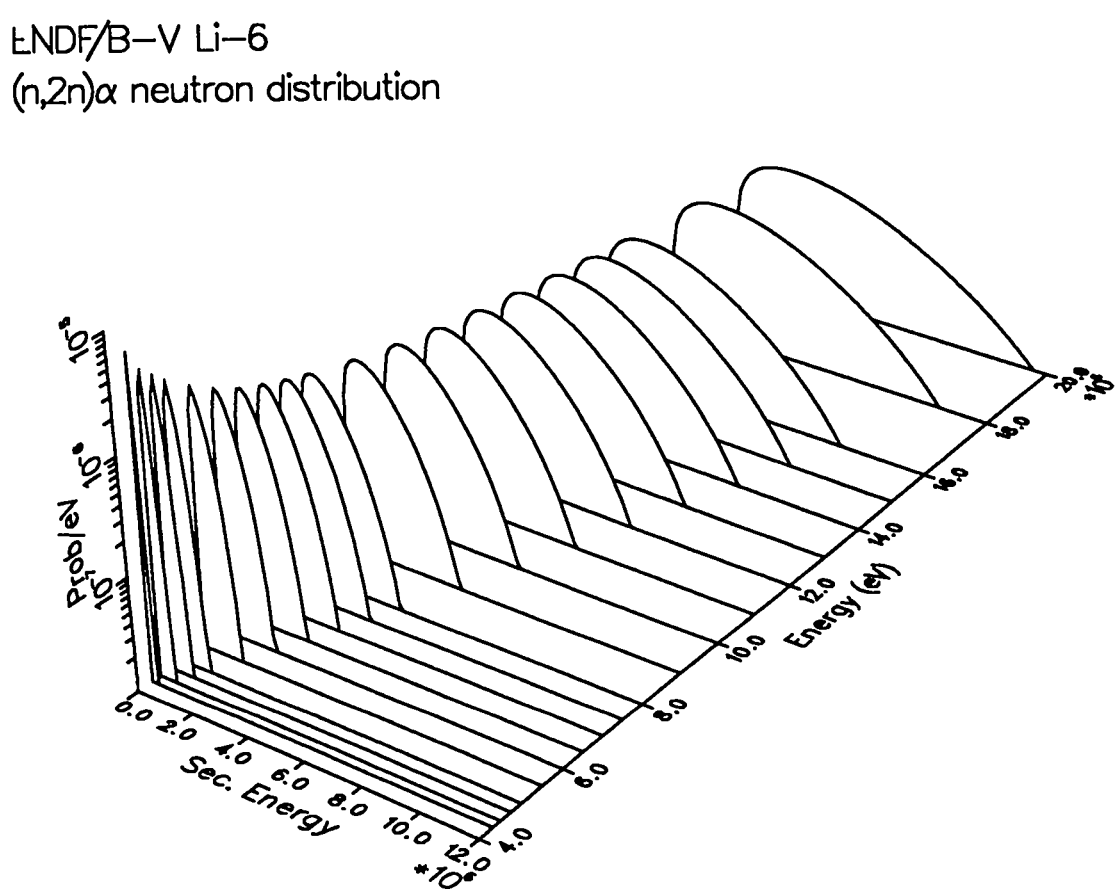


Figure 9: Perspective plot of Li-6 (n,2n)α neutron secondary-energy distributions.

the ntp parameter may be used to select one of the emission angles, and the 3-D plot shows the distribution for that angle.

G. Two-D Spectra Plots from Files 5 and 6

It is difficult to see real detail on 3-D plots, and the emission spectra cannot be compared with measurements. Therefore, PLOTR has the capability to extract the spectrum for a particular particle and incident energy. The following input deck shows how several such spectra can be plotted on one graph.

```
 1.  0
 2.  5
 3.  PLOTR
 4.  /
 5.  1/
 6.  *ENDF/B-V L<I>-6*/
 7.  *<(N,2N)#A <NEUTRON SPECTRA VS >E*/
 8.  4 0 2 2/
 9.  10. 2.E7/
10.  /
11.  1E-11 1E-6/
12.  *C<ROSS> S<ECTION> (<BARNS/E>V)*/
13.  5 20 1303 5 24 0. 12/
14.  /
15.  *10 M<E>V*/
16.  1E3 2E-11 1e2/
17.  2/
18.  5 20 1303 5 24 0. 16/
19.  /
20.  *14 M<E>V*/
21.  1E4 2E-10 2E3/
22.  3/
23.  5 20 1303 5 24 0. 20/
24.  /
25.  *20 M<E>V*/
26.  1E5 2E-9 4E4/
27.  99/
28.  STOP
```

The results are shown in Fig. 10. "Tags" are used to distinguish between the different incident energy values.

The method used for selecting which curve is to be plotted is awkward in the current version of PLOTR. The user must give the index number of the incident energy desired (like the value 12 in line 13 of this example). Future versions of PLOTR should allow actual energy values to be entered, and they should interpolate for intermediate values if necessary.

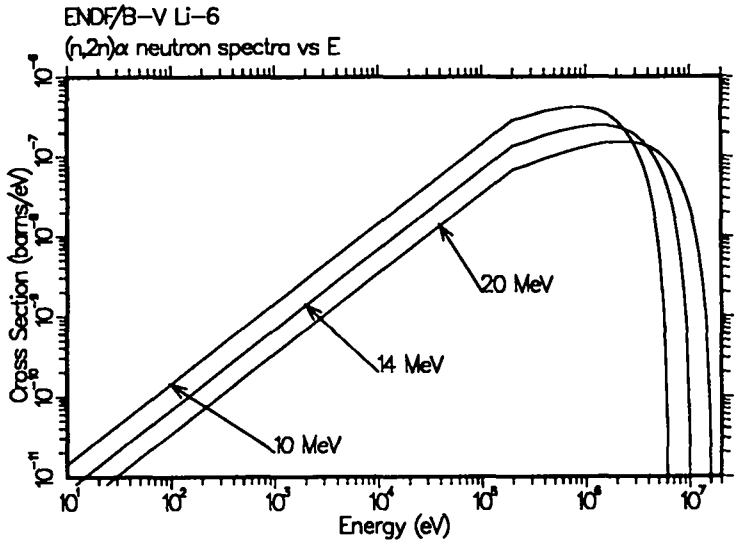


Figure 10: Two-D plot of selected secondary neutron spectra for the Li-6 (n,2n)α reaction.

When an ENDF-6 format File 6 is available, emission spectra will normally be given for several emitted particles, photons, and recoil nuclei. In addition, angular data may be given for various $E \rightarrow E'$ transfers using several different representations. This complicates the task of selecting which curve is to be extracted from File 6. For Law 1 data, the user must specify the particular outgoing particle to be considered (see NKH), the index for the incident energy desired (see NTH), and the dependent variable to plot (see NTP). The result depends on the representation used. In every case, NTP=1 gives the cross section vs E', but for Legendre polynomials, NTP=2 gives the P_1 component vs. E', and for Kalbach-Mann, NTP=2 gives the preequilibrium ratio vs. E'. For Law 7 data, NTP specifies the emission angle. The graph will show a spectrum vs. E' for the specified angle, specified incident energy E, and specified particle.

Future versions of PLOTR should also allow for the construction and plotting of explicit angular distributions for any specified transfer and particle using File 6 Law 1 data.

H. Input Instructions

```
*---INPUT-----------------------------------------------------------*
*                                                                   *
*          ---------------------------------------                  *
*          STYLE TRIGGERS FOR TITLES AND LABELS ARE                 *
*                    > = UPPER-CASE STANDARD                        *
*                    < = LOWER-CASE STANDARD                        *
*                    ? = UPPER-CASE GREEK                           *
*                    # = LOWER-CASE GREEK                           *
*                    + = INSTRUCTIONS                               *
*          ---------------------------------------                  *
*                                                                   *
*  CARD 1                                                           *
*     XPAGE,YPAGE   PAGE SIZE FOR DISPLAY (DEFAULT=10X8)            *
*     ISTYLE        CHARACTER SYLE (DEF=1)                          *
*                     1 = SIMPLEX                                   *
*                     2 = DUPLEX                                    *
*                     3 = COMPLEX                                   *
*                     4 = SWISSM                                    *
*     SIZE          CHARACTER SIZE OPTION                           *
*                     POS = HEIGHT IN PAGE UNITS                    *
*                     NEG = HEIGHT AS FRACTION OF SUBPLOT SIZE      *
*                                                                   *
* -----REPEAT CARDS 2 THROUGH 13 FOR EACH CURVE-----                *
*                                                                   *
*  CARD 2                                                           *
*     IPLOT         PLOT INDEX                                      *
*                     99 = TERMINATE PLOTTING JOB                   *
*                      1 = NEW AXES, NEW PAGE                       *
*                     -1 = NEW AXES, EXISTING PAGE                  *
*                      N = NTH ADDITIONAL PLOT ON EXISTING AXES     *
*                     -N = START A NEW SET OF CURVES USING          *
*                          THE ALTERNATE Y AXIS                     *
*                     DEFAULT = 1                                   *
*     FACTX         FACTOR FOR ENERGIES (DEFAULT=1.)                *
*     FACTY         FACTOR FOR CROSS-SECTIONS (DEFAULT=1.)          *
*     XLL,YLL       LOWER-LEFT CORNER OF PLOT AREA                  *
*     XUR,YUR       UPPER-RIGHT CORNER OF PLOT AREA                 *
*                   (PLOT AREA DEFAULTS TO ONE PLOT PER PAGE)       *
*                                                                   *
* -----CARDS 3 THRU 7 FOR IPLOT = 1 OR -1 ONLY-----                 *
*                                                                   *
*  CARD 3                                                           *
*     T1            FIRST LINE OF TITLE                             *
*                   60 CHARACTERS ALLOWED.                          *
*                   DEFAULT=BLANK                                   *
*                                                                   *
*  CARD 3A                                                          *
*     T2            SECOND LINE OF TITLE                            *
*                   60 CHARACTERS ALLOWED.                          *
*                   DEFAULT=BLANK                                   *
*                                                                   *
```

```
*  CARD 4                                                          *
*     ITYPE          TYPE FOR PRIMARY AXES                        *
*                      1 = LINEAR X - LINEAR Y                    *
*                      2 = LINEAR X - LOG Y                       *
*                      3 = LOG X - LINEAR Y                       *
*                      4 = LOG X - LOG Y                          *
*                      SET NEGATIVE FOR 3D AXES                   *
*                      DEFAULT=4                                  *
*     JTYPE          TYPE FOR ALTERNATE Y AXIS OR Z AXIS          *
*                      0 = NONE                                   *
*                      1 = LINEAR                                 *
*                      2 = LOG                                    *
*                      DEFAULT=0                                  *
*     IGRID          GRID AND TIC MARK CONTROL                    *
*                      0 = NO GRID LINES OR TIC MARKS             *
*                      1 = GRID LINES                             *
*                      2 = TIC MARKS ON OUTSIDE                   *
*                      3 = TIC MARKS ON INSIDE                    *
*                      DEFAULT=2                                  *
*     ILEG           OPTION TO WRITE A LEGEND.                    *
*                      0 = NONE                                   *
*                      1 = WRITE A LEGEND BLOCK WITH UPPER LEFT   *
*                          CORNER AT XTAG,YTAG (SEE BELOW)        *
*                      2 = USE TAG LABELS ON EACH CURVE WITH      *
*                          A VECTOR FROM THE TAG TO THE CURVE     *
*                      DEFAULT=0                                  *
*     XTAG           X COORDINATE OF UPPER LEFT CORNER            *
*                    OF LEGEND BLOCK                              *
*     YTAG           Y COORD OF UPPER LEFT CORNER                 *
*                    DEFAULT=UPPER LEFT CORNER OF PLOT            *
*                                                                 *
*  CARD 5                                                         *
*     EL             LOWEST ENERGY TO BE PLOTTED                  *
*     EH             HIGHEST ENERGY TO BE PLOTTED                 *
*     XSTEP          X AXIS STEP                                  *
*                    DEFAULT = AUTOMATIC SCALES                   *
*                    (DEFAULT ALL 3, OR NONE)                     *
*                    (THE ACTUAL VALUE OF XSTEP IS                *
*                        IGNORED FOR LOG SCALES)                  *
*                                                                 *
*  CARD 5A                                                        *
*     XLABL          LABEL FOR X AXIS                             *
*                    60 CHARACTERS ALLOWED.                       *
*                     DEFAULT="ENERGY (EV)"                       *
*                                                                 *
*  CARD 6                                                         *
*     YL             LOWEST VALUE OF Y AXIS.                      *
*     YH             HIGHEST VALUE OF Y AXIS.                     *
*     YSTEP          STEP FOR Y AXIS (LINEAR SCALES ONLY)         *
*                    DEFAULT = AUTOMATIC SCALES                   *
*                    (DEFAULT ALL 3, OR NONE)                     *
*                    (THE ACTUAL VALUE OF YSTEP IS                *
*                        IGNORED FOR LOG SCALES)                  *
```

```
*                                                                     *
*  CARD 6A                                                            *
*     YLABL          LABEL FOR Y AXIS                                 *
*                    60 CHARACTERS ALLOWED.                           *
*                     DEFAULT="CROSS SECTION (BARNS)"                 *
*                                                                     *
*  CARD 7   (JTYPE.GT.0 ONLY)                                         *
*     RBOT           LOWEST VALUE OF SECONDARY Y AXIS OR Z AXIS       *
*     RTOP           HIGHEST VALUE OF SECONDARY Y AXIS OR Z AXIS      *
*     RSTEP          STEP FOR SECONDARY Y AXIS OR Z AXIS              *
*                    DEFAULT FOR LAST THREE = AUTOMATIC               *
*                                                                     *
*  CARD 7A  (JTYPE.GT.0 ONLY)                                         *
*     RL             LABEL FOR ALTERNATE Y AXIS OR Z AXIS             *
*                    60 CHARACTERS ALLOWED.                           *
*                    DEFAULT=BLANK                                    *
*                                                                     *
*  -----CARDS 8 THRU 9 ARE ALWAYS GIVEN-----                          *
*                                                                     *
*  CARD 8                                                             *
*     IVERF          VERSION OF ENDF TAPE                             *
*                      SET TO ZERO FOR DATA ON INPUT FILE             *
*                        AND IGNORE REST OF PARAMETERS ON CARD        *
*                      SET TO 1 FOR GENDF DATA                        *
*     NIN            INPUT TAPE                                       *
*                    CAN CHANGE FOR EVERY CURVE IF DESIRED.           *
*     MATD           DESIRED MATERIAL                                 *
*     MFD            DESIRED FILE                                     *
*     MTD            DESIRED SECTION                                  *
*     TEMPER         TEMPERATURE FOR ENDF/B DATA (K)                  *
*                    DEFAULT=0.                                       *
*     NTH,NTP,NKH    SEE BELOW (DEFAULTS=1)                           *
*                                                                     *
*       SPECIAL MEANINGS FOR NTH,NTP,NKH FOR FILE 3 OR 5 DATA         *
*            NTH    NUMBER OF SUBSECTION TO PLOT                      *
*              (WORKS FOR ISOMER PROD, DELAYED N, ETC.)               *
*            NTP    NOT USED                                          *
*            NKH    NOT USED                                          *
*                                                                     *
*       SPECIAL MEANINGS FOR NTH,NTP,NKH FOR FILE 6 DATA              *
*            NTH    INDEX FOR  INCIDENT ENERGY                        *
*            NTP    NUMBER OF DEP. VARIABLE IN CYCLE TO PLOT          *
*                   (OR ANGLE NUMBER FOR LAW 7)                       *
*            NKH    NUMBER OF OUTGOING PARTICLE TO PLOT               *
*                                                                     *
*       SPECIAL MEANINGS FOR NTH,NTP,NKH FOR GENDF MF=3 DATA          *
*            NTH=0 FOR FLUX PER UNIT LETHARGY                         *
*            NTH=1 FOR CROSS SECTION (DEFAULT)                        *
*            NTP=1 FOR INFINITE DILUTION (DEFAULT)                    *
*            NTP=2 FOR NEXT LOWEST SIGMA-ZERO VALUES, ETC.            *
*            NKH=1 FOR P0 WEIGHTING (DEFAULT)                         *
*            NKH=2 FOR P1 WEIGHTING (TOTAL ONLY)                      *
*                                                                     *
```

```
*       SPECIAL MEANING FOR NTH FOR GENDF MF=6 DATA                *
*            NTH=1 PLOT 2-D SPECTRUM FOR GROUP 1                   *
*            NTH=2 PLOT 2-D SPECTRUM FOR GROUP 2                   *
*              ETC.                                                *
*        NO SPECIAL FLAGS ARE NEEDED FOR MF=6 3D PLOTS             *
*                                                                  *
*      SPECIAL MEANINGS FOR NTH AND NTP FOR MF7 PLOTS              *
*           NTH IS INDEX FOR INDEP. VARIABLE (ALPHA OR BETA)       *
*           NTP=1 SELECTS ALPHA AS INDEP. VARIABLE (DEFAULT)       *
*           NTP=2 SELECTS BETA AS INDEP. VARIABLE                  *
*           NKH=1 SELECTS NORMAL S(ALPHA,BETA)                     *
*           NKH=2 SELECTS SCRIPT S(ALPHA,-BETA)                    *
*           NKH=3 SELECTS SCRIPT S(ALPHA,BETA)                     *
*                                                                  *
* -----CARDS 9 AND 10 FOR 2D PLOTS ONLY-----                       *
*                                                                  *
*  CARD 9                                                          *
*     ICON          SYMBOL AND CONNECTION OPTION                   *
*                     0 = POINTS CONNECTED, NO SYMBOLS             *
*                    -I = POINTS NOT CONNECTED, SYMBOL AT EVERY    *
*                         ITH POINT                                *
*                     I = POINTS CONNECTED, SYMBOL AT EVERY ITH    *
*                         POINTS                                   *
*                    DEFAULT=0                                     *
*     ISYM          NO. OF SYMBOL TO BE USED                       *
*                     0 = SQUARE                                   *
*                     1 = OCTAGON                                  *
*                     2 = TRIANGLE                                 *
*                     3 = CROSS                                    *
*                     4 = EX                                       *
*                     5 = DIAMOND                                  *
*                     6 = INVERTED TRIANGLE                        *
*                     7 = EXED SQUARE                              *
*                     8 = CROSSED EX                               *
*                     9 = CROSSED DIAMOND                          *
*                     10 = CROSSED OCTAGON                         *
*                     11 = DOUBLE TRIANGLE                         *
*                     12 = CROSSED SQUARE                          *
*                     13 = EXED OCTAGON                            *
*                     14 = TRIANGLE AND SQUARE                     *
*                     15 = FILLED CIRCLE                           *
*                     16 = OPEN CIRCLE                             *
*                     17 = OPEN SQUARE                             *
*                     18 = FILLED SQUARE                           *
*                     DEFAULT=0                                    *
*     IDASH         TYPE OF LINE TO PLOT                           *
*                     0 = SOLID                                    *
*                     1 = DASHED                                   *
*                     2 = CHAIN DASH                               *
*                     3 = CHAIN DOT                                *
*                     4 = DOT                                      *
*                     DEFAULT=0                                    *
*     ITHICK        NO. OF TIMES TO REDRAW CURVE                   *
```

```
*                  DEFAULT=0.                                          *
*                                                                      *
*  CARD 10  ---ILEG.NE.0 ONLY---                                       *
*     ALEG          TITLE FOR CURVE TAG OR LEGEND BLOCK               *
*                   60 CHARACTERS ALLOWED.                            *
*                   DEFAULT=BLANK                                     *
*                                                                      *
*  CARD 10A  ---ILEG.EQ.2 ONLY---                                      *
*     XTAG          X POSITION OF TAG TITLE                           *
*     YTAG          Y POSITION OF TAG TITLE                           *
*     XPOINT        X COORDINATE OF VECTOR POINT                      *
*                    (.LE.0 TO OMIT VECTOR)                           *
*                                                                      *
* -----CARD 11 FOR 3D PLOTS ONLY-----                                  *
*                                                                      *
*  CARD 11                                                             *
*     XV,YV,ZV      ABS. COORDS OF VIEW POINT                          *
*                   DEFAULTS=5.,2.,2.                                  *
*     X3,Y3,Z3      ABS. SIDES OF WORK BOX VOLUME                      *
*                   DEFAULTS=5.,2.,2.                                  *
*                                                                      *
*          SET Y3 NEGATIVE TO FLIP THE ORDER OF THE AXIS ON            *
*          THAT SIDE OF THE BOX (SECONDARY ENERGY).                    *
*                                                                      *
*  -----CARDS 12 THRU 13 FOR IVERF = 0 ONLY-----                       *
*                                                                      *
*  CARD 12                                                             *
*     NFORM          FORMAT CODE FOR INPUT DATA                       *
*                    0 = FREE FORMAT INPUT WITH                       *
*                    OPTIONAL X AND Y ERROR BARS                      *
*                                                                      *
*  CARD 13   ---NFORM = 0 ONLY---                                      *
*     XDATA          DEPENDENT VALUE                                  *
*                    TERMINATE WITH XDATA NEGATIVE                    *
*     YDATA          INDEPENDENT VALUE                                *
*     YERR1          LOWER Y ERROR LIMIT                              *
*                    NO Y ERROR BAR IF ZERO                           *
*     YERR2          UPPER Y ERROR LIMIT                              *
*                    IF ZERO, EQUALS YERR1                            *
*     XERR1          X LEFT ERROR LIMIT                               *
*                    NO X ERROR BAR IF ZERO                           *
*     XERR2          X RIGHT ERROR LIMIT                              *
*                    IF ZERO, EQUALS XERR1                            *
*                                                                      *
*                                                                      *
* ALL CURVES CONTAIN AT LEAST 10 POINTS PER DECADE (SEE DELTA).        *
* CODE CAN PLOT CURVES CONTAINING FEWER THAN 2000 POINTS (SEE          *
* MAX) WITHOUT THINNING.  CURVES WITH MORE POINTS ARE THINNED          *
* BASED ON A MINIMUM SPACING DETERMINED FROM MAX AND THE               *
* LENGTH OF THE X AXIS.                                                *
*                                                                      *
************************************************************************
```

I. Coding Details

PLOTR uses a limited number of Hollerith strings, and the first step is to set up variables MAXLAB and NCW for the maximum number of characters in such a string and the number of characters in a computer word for the target machine. Values for 32–bit (VAX, IBM), 60–bit (CDC), and 64–bit (Cray) machines are given near the start of PLOTR. Note that labels can have at least 60 characters on all machines.

The next step is to initialize the DISSPLA graphics package. This step differs from system to system. At Los Alamos, we call GPLOT and LIBDISP. We also call UCCHAR to tell the system that we will use explicit style triggers to change back and forth between uppercase and lowercase. PLOTR does not expect to see mixed case actually typed in as input.

The next step is to start reading the user's input. For each input line, the defaults are loaded into the z array, and subroutine FREE is used to read the input. Once all the input parameters have been read, the code branches to different regions for different data types.

For ENDF and PENDF data, PLOTR reads the first part of File 1 and searches for the desired temperature. It then uses FINDF to locate the desired material and reaction. For 2-D plots, it is necessary to extract the desired x and y values out of a tab1 or a list record. It is fairly tricky finding this record because of the large number of possible formats in Files 3, 5, 6, 15, and so on. Some improvements are needed here to allow PLOTR to construct angular distributions from File 4 and to allow access by actual values rather than the indexes NTH, NTP, and NKH. Methods to access File 7 are also needed.

Once the appropriate record has been located, PLOTR sets up a process to extract the x and y values from the record either thinning or thickening the given x grid as necessary to get a good plot. After all the values have been computed, the x and y arrays are passed to subroutine SETUP to be plotted.

For 3-D ENDF or PENDF plots, the code first skips to the desired subsection of File 4, 5, 6, 15, etc. The structure of a subsection varies with the law used to describe the data. The format for File 4 and File 6 two-body angular distribution data is similar, so a common procedure can be used for these two Files. At each incident energy, the angular distribution is either obtained by interpolation in the given tabulation, or it is computed from the given Legendre coefficients. Once all the numbers have been loaded into the AA array, they are passed to subroutine SET3D for plotting.

For Files 5, 6, and 15 3-D plots, the data from the desired subsection is converted to log form and loaded into the AA array. During this process, the maximum and minimum values on the z axis are determined. The code then determines an appropriate vertical axis range of about 3 decades and loops through the AA array again removing all z values smaller than the new lower limit. The modified AA array is passed to SET3D for plotting.

For GENDF 2-D data, PLOTR searches the GENDF tape for the requested material and temperature. It then reads in the entire MT451 header record and sets pointers to the energy boundaries for the particle and photon group structures. The next step is to loop through the records on the file looking for the desired reaction. The cross section values are read from the list records, and energy values for the histogram break points are computed from the group structure information. The resulting x and y arrays are passed to SETUP for plotting.

Experimental data is read directly into the x and y arrays from the input deck. If error bars are present, they are read into the arrays DXM, DXP, DYM , and DYP. The x and y arrays are passed to SETUP for plotting. If error bars are present, the error bar arrays are passed to ERBAR, which adds the error bars to the plot made by SETUP.

The code now loops back to do the next plot. When plot number 99 is seen, PLOTR is finished. The graphics system is terminated using the GDONE call at Los Alamos. The termination process may be different for other systems.

Subroutine SETUP is a simplified interface to DISSPLA for preparing 2-D plots. The basic input consists of a plot number IPLOT, the X and Y arrays, and a data count N. However, there are many additional parameters that are passed to SETUP in labeled common arrays with names like SETUP1, SETUP2, and so on. SETUP starts by initializing the font styles to be used and their trigger characters. SETUP then sets up the subplot area and begins working on the axes. Scales can be passed directly to SETUP through common, or determined automatically. Automatic scaling does not use the DISSPLA procedure, which sometimes is inappropriate for scientific applications; instead, it uses a procedure ASCALE based on earlier Los Alamos software. Next, the actual axes, tick marks, grid lines, labels, and frame are drawn. The optional titles are added with calls to the DISSPLA MESSAG routine. SETUP is now ready to add the curves to the plot. Finally, the tags or legend lines are drawn.

SETUP has companion routines ERBAR to add error bars to a previously drawn curve, LEGND to write the legend block, and TAGIT to write the curve tags and arrows.

Three-D plots are done using subroutine SET3D. It is parallel to SETUP. It starts by assigning font styles and trigger characters, and it then defines the subplot area. Next the optional titles are drawn using calls to MESSAG. The 3-D coordinates are set up by the GRAF3D call of DISSPLA. The families of curves plotted by SET3D are not evenly spaced along the incident-energy axis. The SURTRN call is used to specify that subroutine Y3DMAT will be used to assign the y axis location for each curve in the family. The actual curves are drawn by SURMAT. The most difficult part of SET3D is drawing appropriately scaled and labeled axes. Some additional work is need here to change the positions of some of the axes as the viewpoint changes.

The remaining routines in PLOTR are GETY6, which is used to page in data from an ENDF-6 format File 6, and FIXL7, which is used to transform File 6 Law 7 into Law 1 form for plotting. This requires putting the spectra for all angles onto a common secondary-energy grid, and then integrating over angle for each energy of this common grid. The result is a P_0 energy distribution ready for plotting using SET3D.

J. Storage Allocation

The array A(1200) with length NWAMAX=1200 words is used for reading in ENDF-format records, and with the paging strategy used, it should always be sufficient. The main container array is AA(30000) with maximum length given in DATA MAXAA/30000/. The maximum number of x, y pairs in any plot is set by data MAX/2000/. There are several arrays of this length equivalenced to various parts of the container array AA. See X(2000), Y(2000), B(2000), DXM(2000), DXP(2000), DYM(2000), and DYP(2000). These assignments could be changed, if necessary, for a very high resolution device. The arrays used to map coordinate values to curves for 3-D plots are limited to 100. This could be a problem if an evaluation had more than 100 incident energies in File 4, 5, or 6. The only other storage allocation limitation is the choice of 15 words for strings containing Hollerith characters. This allows for 60 characters on machines with 4 characters per word.

K. Input and Output Units

PLOTR doesn't use any internal scratch units. The only units used are those mentioned on type 8 cards, and they can be either ASCII (NIN positive) or blocked binary (NIN negative) as desired. ASCII and blocked binary units can be mixed on a single plot if necessary.

L. Error Messages

`ERROR IN PLOTR***ERROR IN AXIS INPUT`

Check your input Cards 5, 6, and/or 7. Remember that the easiest thing to do with XSTEP, YSTEP, or RSTEP for log axes is to take the default value. For linear scales, either give all three parameters, or default all three. Usually this message means some previous card is missing.

`ERROR IN PLOTR***DESIRED MAT AND TEMP NOT FOUND a`

Check the input cards against the input tape mounted.

`ERROR IN PLOTR***LF=1 ONLY FOR MF5 OR 15.`

The analytic secondary distribution laws are not supported by PLOTR.

`ERROR IN PLOTR***LF=1 OR 7 ONLY FOR FILE 6.`

Two-D plots from File 6 are currently limited to laws 1 and 7.

`ERROR IN PLOTR***STORAGE EXCEEDED.`

Either there is an undiscovered error in the thinning/thickening logic, or an attempt has been made to read a GENDF reaction with more than NWAMAX=1200 words.

`ERROR IN PLOTR***ILLEGAL MF6 LAW.`

Only laws 0, 1, 2, 3, 4, and 7 are currently supported.

`MESSAGE FROM PLOTR---NO DISTRIBUTION, NO PLOT`

No distribution was found on the input tape.

M. Future Improvements

An important improvement would be the ability to construct and plot an angular distribution for a particular incident energy E for File 4 or File 6 two-body angular data. The value of E needs to be given, not an index value, and interpolation should be provided for E values between the grid values of the evaluation. Either angle θ or cosine μ should be allowed for the abscissa. It would also be convenient to be able to select the incident energy for extracting spectra out of File 5 or File 6 with an actual energy value instead of an index number. Interpolation to E values between the grid values should be available. Similarly, the code needs the abilities to select the outgoing particle to be plotted for File 6 using a ZA identifier instead of an index number, and to construct and plot angular distributions for particular $E \rightarrow E'$ transfers for data in File 6 Law 1 format.

It would be convenient if the position of the 3-D axes would change with viewpoint to give better displays from unusual directions.

Finally, 3-D plotting for File 7 data would be useful.

XXI. MIXR

This module will construct a new PENDF tape with a specified set of reactions that are linear combinations of the cross sections on the input tapes. MIXR can also be used for ENDF tapes, but the input interpolation laws are ignored. The output file contains ENDF File 1 and File 3 sections only, and linear-linear interpolation is assumed.

MIXR can be used for several purposes. When combined with PLOTR, it can be used to construct mixed cross sections for plotting. One example would be to combine the isotopic cross sections given for ENDF/B-VI evaluations into elemental cross sections for comparison with ENDF/B-V data (see the sample input deck below). Another would be to compute the difference between two versions of a particular reaction; plots that simultaneously show one version of a function and the difference between that version and another are sometimes very revealing. In both cases, the mixed cross section is computed on the union grid of its components, and all details are preserved.

Another use for MIXR would be to prepare a new ENDF material to represent a particular "enriched uranium," having a particular amount of ^{235}U mixed with ^{238}U. This file could then be run through the GROUP flux calculator to generate an accurate pointwise flux for enriched uranium mixed with water, or for pins of enriched uranium in a typical reactor lattice. This computed flux could then be used as the weighting flux for computing the self-shielded cross sections of both ^{235}U and ^{238}U in two subsequent GROUPR runs. This approach handles the interference between resonances in ^{235}U and ^{238}U accurately, including proper treatment of all wide and intermediate resonance effects.

A. User Input

The following input instructions were copied from the comment cards at the beginning of the MIXR source. It is always a good idea to check the cards in the current version of the program for possible changes.

```
C     *  USER INPUT --                                                *
C     *                                                               *
C     *  CARD 1  --  UNITS                                            *
C     *     NOUT      OUTPUT UNIT FOR MIXED CROSS SECTIONS            *
C     *     NIN1      FIRST INPUT UNIT (ENDF OR PENDF)                *
C     *     NIN2      SECOND INPUT UNIT                               *
C     *       ...      CONTINUE FOR NNIN<=10 INPUT UNITS              *
C     *                                                               *
C     *  CARD 2  --  REACTION LIST                                    *
C     *      MTN       LIST OF NMT<=20 MT NUMBERS FOR                 *
```

```
C     *                    THE OUTPUT REACTIONS                        *
C     *                                                                *
C     *  CARD 3  --  MATERIAL LIST                                     *
C     *     MATN,     LIST OF NMAT<=10 PAIRS (MATN,WTN) OF MAT         *
C     *      WTN       NUMBERS AND ASSOCIATED WEIGHT FACTORS          *
C     *                                                                *
C     *  CARD 4  --  TEMPERATURE                                       *
C     *     TEMP      TEMPERATURE (USE ZERO EXCEPT FOR PENDF TAPES)    *
C     *                                                                *
C     *  CARD 5  --  OUTPUT MATERIAL                                   *
C     *     MATD      MATERIAL NUMBER                                  *
C     *     ZA        ZA VALUE                                         *
C     *     AWR       AWR VALUE                                        *
C     *                                                                *
C     *  CARD 6  --  FILE 1 COMMENT CARD                               *
C     *     DES       DESCRIPTION (60 CHAR MAX)                        *
C     *                                                                *
C     ******************************************************************
```

CARD 1. This card gives the unit numbers for the output that will contain the mixed cross sections and the input files that contain the materials to be mixed together. Note that there is a limit of 10 input materials.

CARD 2. The code currently allows up to 20 mixed reactions to be generated. This card contains a list of the MT numbers for these reactions. The number of reactions is counted automatically.

CARD 3. This card (it may continue onto several lines) contains the MAT numbers and weights for the materials to be mixed. The first pair **MATN**, **WTN** refers to the data on input file **NIN1**; the next pair is for **NIN2**, etc.

CARD 4. The same temperature is used for all of the materials, and its value is given on this card in Kelvin. The appropriate value for ENDF tapes is zero. The temperatures used for PENDF cases must be in the set included on the PENDF tape.

CARD 5. This card gives some values that are needed to construct the ENDF or PENDF file for the output material. **MATD** is the conventional material number. **ZA** is $1000 * Z + A$, where Z is the charge number and A is the atomic mass number for the mixture. Finally, **AWR** is the mass ratio to the neutron for the mixture.

CARD 6. This card contains one line of text to be used for the Hollerith descriptive information in MF=1/MT=451 on the new ENDF tape for the fixture. The final tape on **NOUT** will contain the section MF=1/MT=451 and sections MF=3/MT=MT_n only.

The following sample input deck combines the four isotopes of iron to generate plots for the total cross section (MT=1), the energy-balance heating cross section (MT=301), and the radiation damage cross section (MT=444). See the PLOTR chapter for more information on the part of the input deck that produces the actual graphs.

```
CFS GET XNJOY:/NJOY/91/XNJOYNR
CFS GET TAPE21:/PENDF/5/FE/MATR2
CFS GET TAPE22:/PENDF/6/FE/54
CFS GET TAPE23:/PENDF/6/FE/56
CFS GET TAPE24:/PENDF/6/FE/57
CFS GET TAPE25:/PENDF/6/FE/58
CAT>INPUT <<EOF
 0
 6
 MIXR
 20 -22 -23 -24 -25/
 1 301 444/
 2625 .0590
 2631 .9172
 2634 .0210
 2637 .0028/
 300/
 2600 26000 56.
 *FE-MAT FOR ENDF/B-VI FROM MIXR*/
 PLOTR
 /
 1/
 *ENDF/B-VI IRON*/
 *TOTAL*/
 1 0 2 1 13E6 9./
 0 20E6 2E6/
 /
 0 10 2/
 /
 6 20 2600 3 1 300/
 /
 *ENDF/B-VI*/
 2/
 5 -21 1326 3 1 300/
 0 0 4/
 *ENDF/B-V*/
 1/
 *ENDF/B-VI IRON*/
 *TOTAL*/
 4 0 2 1/
 1E2 1E5/
 /
 /
 /
 6 20 2600 3 1 300/
 /
```

```
 *ENDF/B-VI*/
 2/
 5 -21 1326 3 1 300/
 0 0 4/
 *ENDF/B-V*/
 1/
 *ENDF/B-VI IRON*/
 *HEATING CHECK*/
 1 0 2 1 1E6 2.7E6/
 0 20E6 2E6/
 /
 0 3E6 .5E6/
 *H<EATING (E>V<-BARNS>)*/
 6 20 2600 3 301 300/
 /
 *ENDF/B-VI*/
 3/
 5 -21 1326 3 301 300/
 0 0 4/
 *ENDF/B-V*/
 1/
 *ENDF/B-VI IRON*/
 *DAMAGE CHECK*/
 1 0 2 1 13E6 7E4/
 0 20E6 2E6/
 /
 0 35E4 5E4/
 *D<AMAGE (E>V<-BARNS>)*/
 6 20 2600 3 444 300/
 /
 *ENDF/B-VI*/
 3/
 5 -21 1326 3 444 300/
 0 0 4/
 *ENDF/B-V*/
 99/
 STOP
EOF
XNJOY<INPUT
CFS STORE PLOT:/PENDF/6/FE/NATPLOT
```

The first few lines retrieve the NJOY executable, the ENDF/B-V version of Fe-nat, and the four iron isotopes for ENDF/B-VI. A c-shell "here file" is then created with the input for MIXR and PLOTR. The lines after the EOF line take care of executing NJOY and saving the plot file for future use.

Returning to the actual NJOY input cards, note that the MIXR output will be on TAPE20 in ASCII mode, and the four isotopic input files (in binary mode) will be on TAPE22 through TAPE25. The next input card lists the three reaction types to be written onto the output file; it is followed by four cards giving the isotope

MAT numbers and the atomic fractions that they are to have in the mixture. These mix-specification cards are then followed by a card asking for a temperature of 300 K.

The last two cards in the MIXR section specify the ENDF parameters for the mixed material that will be written on TAPE20. The MAT number is to be 2600 (the standard ENDF-6 value for natural iron), the ZA value is to be 26000, and the AWR value is to be 56.0000 (a rather crude choice). The text line shown will appear in MF=1/MT=451 on the output tape.

At this point, the MIXR module will run and produce its output file. The rest of the input deck reads in this output file and makes a graphical comparison of the mixed elemental cross section with the corresponding elemental values from ENDF/B-V.

B. Coding Details

The heart of MIXR is subroutine GETY. It is similar to GETY1 in NJOY, except that it works with several TAB1 records in parallel. The routine is initialized for up to 10 input TAB1 records on up to 10 input tapes by calling it with X=0 for each tape. It returns the value of XNEXT for each tabulation. The calling program can compare these XNEXT values to determine the lowest energy point for the mixed reaction (the threshold).

On subsequent calls with X>0, MIXR returns the value of Y corresponding to X for the specified tabulation, and it also computes XNEXT for this tabulation. The calling program can loop through all the input tapes, compute the mixed value of Y as a linear combination of the separate GETY results, and compute the next energy grid point as the lowest of the values of XNEXT returned by the separate GETY calls. In this way, the mixed results are obtained on a unionized grid, and no features are lost.

To do its job, GETY has to keep tables of the current location and related parameters on each of the input tapes. These tables will be found in common /GETYC/ with dimension 10.

C. Error Messages

ERROR IN MIXR***MAT AND TEMP NOT FOUND

There is an inconsistency between the requested MATN and TEMP values and the materials found on the input files NIN1, NIN2, ...

MESSAGE FROM MIXR***MT=xxx NOT PRESENT FOR MAT=xxxx

This message is just information. If the user asked for construction of, for example, an (n,2n) reaction for an element, and if one of the isotopes did not have an (n,2n) reaction, this message would be issued.

ERROR IN GETY***NOT PROPERLY INITIALIZED

The subroutine GETY has to be called with X=0 for each material/section combination to be introduced into the mixed reaction before it is called with X>0.

XXII. PURR

The unresolved self-shielding data generated by UNRESR is suitable for use in multigroup methods after processing by GROUPR, but the so-called Bondarenko method[1] is not very useful for continuous-energy Monte Carlo codes like MCNP.[2] As pointed out by Levitt,[3] the natural approach for treating unresolved-resonance self-shielding for Monte Carlo codes is the "Probability Table" method. The trick is to develop tables of the probability that the total cross section will be less than some value σ_t for a number of incident energies. Then, when the Monte Carlo code needs the cross section at E, it selects a random number between 0 and 1 and looks up the corresponding σ_t in the appropriate probability table. The corresponding capture and fission cross sections are obtained from conditional probability tables that give σ_γ and σ_f versus σ_t. This approach allows geometry and mix effects on self-shielding to arise naturally during the Monte Carlo calculation, and it supplies reasonable variances for the tallies. The Probability Table method has been used successfully in a number of applications, notably the VIM code,[4] which was developed to solve fast-reactor problems where unresolved effects become very important.

The PURR module produces probability tables that have been used in experimental versions of MCNP to treat unresolved-resonance self-shielding. It works by generating a "ladder" of resonances that obey the chi-square distributions for widths and the Wigner distribution for spacing as specified in the ENDF evaluation. Cross sections are sampled from the ladder at randomly selected energies and accumulated into probability tables and Bondarenko moments. The whole process is repeated for additional ladders until reasonable statistics are obtained. The probability tables are written to a file for use by MCNP, and the Bondarenko integrals are printed out for comparison with the results of other methods, such as UNRESR.

PURR is scheduled for a major update with the next version of NJOY. This update includes changes to track temperature correlations and new ways of coupling PURR results to ACER. Some of the details are contingent on related changes in MCNP that have not yet been made. This chapter will be expanded into a full description for this future version. In the meantime, the current code is suitable for experimentation with unresolved-resonance methods.

A. User Input

The following user input specification was copied from the comment cards at the beginning of the PURR source. It is always a good idea to check these comments in the current version in case there have been changes.

```
C     *---INPUT DATA CARDS-----------------------------------------------*
C     *                                                                  *
C     * CARD 1                                                          *
C     *   NENDF   UNIT FOR ENDF/B TAPE                                  *
C     *   NIN     UNIT FOR INPUT PENDF TAPE                             *
C     *   NOUT    UNIT FOR OUTPUT PENDF TAPE                            *
C     *   NACE    UNIT FOR OUTPUT ACE TAPE                              *
C     * CARD 2                                                          *
C     *   MATD    MATERIAL TO BE PROCESSED                              *
C     *           MATD=0 TERMINATES PURR                                *
C     *   NTEMP   NO. OF TEMPERATURES                                   *
C     *   NSIGZ   NO. OF SIGMA ZEROES                                   *
C     *   NBIN    NO. OF PROBABILITY BINS                               *
C     *   NLADR   NO. OF RESONANCE LADDERS                              *
C     *   IPRINT  PRINT OPTION (0=MIN, 1=MAX, DEF=1)                    *
C     *   NUNX    NO. OF ENERGY POINTS DESIRED (DEF=0=ALL)              *
C     * CARD 3                                                          *
C     *   TEMP    TEMPERATURES IN KELVIN (INCLUDING ZERO)               *
C     * CARD 4                                                          *
C     *   SIGZ    SIGMA ZERO VALUES (INCLUDING INFINITY)                *
C     *                                                                  *
C     ********************************************************************
```

B. Error Messages

ERROR IN PURR***MODE CONVERSION BETWEEN NIN AND NOUT NOT ALLOWED

NIN and NOUT must both be binary (negative) or ASCII (positive).

ERROR IN PURR***TOO MANY TEMPERATURES

The maximum number of temperatures is NTMAX=9. The value is set in the main program of PURR. The temperature and Z arrays are also dimensioned by 9.

ERROR IN PURR***TOO MANY SIGMA ZEROES

The maximum number of sigma-zero values is NZMAX=9. The value is set in the main program of PURR. The sigma-zero and Z arrays are also dimensioned by 9.

ERROR IN PURR***STORAGE EXCEEDED

The amount of space in dynamic storage array B has been exceeded. It will be necessary to increase the size of the main container array. The array is A(20000) in common /PUSTOR/. The size is defined by NAMAX in a data statement.

ERROR IN RDF2UN***STORAGE EXCEEDED

The amount of space in dynamic storage array ARRY has been exceeded. It will be necessary to increase the size of the main container array as described above.

ERROR IN UNRESX***STORAGE EXCEEDED

The limit of 30 spin sequences allowed in subroutine UNRESX has been exceeded. Change the dimension of ABNS in this routine.

ERROR IN LADR3***TOO MANY RESONANCES IN LADDER

There is a limit of NERMAX=1000 resonances in a ladder. This quantity is carried in common /CONTR/, and it is set in the PURR main routine.

ERROR IN LADR2***TOO MANY RESONANCES IN LADDER

There is a limit of NERMAX=1000 resonances in a ladder. This quantity is carried in common /CONTR/, and it is set in the PURR main routine.

ERROR IN FSORT***NOT INSTALLED

The current version of PURR uses a local library routine for this function. Other sites will have to install some equivalent routine.

ERROR IN FSRCH***NOT INSTALLED

The current version of PURR uses a local library routine for this function. Other sites will have to install some equivalent routine.

ERROR IN RAND***NOT INSTALLED

The current version of PURR uses a local library routine for this function. Other sites will have to install some equivalent routine.

C. References

1. I. I. Bondarenko, Ed., *Group Constants for Nuclear Reactor Calculations* (Consultants Bureau, New York, 1964).

2. Judith F. Briesmeister, Ed., "MCNP—A General Monte Carlo Code for Neutron and Photon Transport, Version 3A," Los Alamos National Laboratory report LA-7396-M, Rev. 2 (September 1986).

3. L. B. Levitt, "The Probability Table Method for Treating Unresolved Resonances in Monte Carlo Criticality Calculations," *Trans. Am. Nucl. Soc.* 14, 648 (1971).

4. R. E. Prael and L. J. Milton, "A User's Manual for the Monte Carlo Code VIM," Argonne National Laboratory report FRA-TM-84 (1976).

Appendix A. GROUP STRUCTURES

1. IGN=2...CSEWG 240-Group Structure

This set of group bounds was proposed[1] as a "supergroup" structure to be used for fine-group libraries that could be collapsed to several other group structures in general use at the time. It was felt that such a collapse would be more economical than producing a set of group constants directly in the target group structure. A large 171-group subset of these group bounds was used for the VITAMIN-C library[2] generated a the Oak Ridge National Laboratory for shielding and fusion applications.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0000E-05	4.1399E-01	5.3158E-01	6.2506E-01	6.8256E-01
6	8.3368E-01	8.7642E-01	1.1254E+00	1.4450E+00	1.8554E+00
11	2.3824E+00	3.0590E+00	3.9279E+00	5.0435E+00	6.4760E+00
16	8.3153E+00	1.0677E+01	1.3710E+01	1.7603E+01	2.2603E+01
21	2.9023E+01	3.7267E+01	4.7851E+01	6.1442E+01	7.8893E+01
26	1.0130E+02	1.3007E+02	1.6702E+02	2.1445E+02	2.7536E+02
31	3.5358E+02	4.5400E+02	5.8295E+02	7.4852E+02	9.6112E+02
36	1.2341E+03	1.3639E+03	1.5073E+03	1.5846E+03	1.6659E+03
41	1.8411E+03	2.0347E+03	2.2487E+03	2.4852E+03	2.6126E+03
46	2.7465E+03	2.8635E+03	3.0354E+03	3.3546E+03	3.7074E+03
51	4.0973E+03	4.3074E+03	4.5283E+03	5.0045E+03	5.5308E+03
56	6.2673E+03	7.1017E+03	8.0473E+03	9.1188E+03	1.0333E+04
61	1.1709E+04	1.3268E+04	1.5034E+04	1.7036E+04	1.9305E+04
66	2.1335E+04	2.1875E+04	2.3579E+04	2.4176E+04	2.4788E+04
71	2.6058E+04	2.8088E+04	3.1828E+04	3.4307E+04	3.5175E+04
76	3.6066E+04	4.0868E+04	4.6309E+04	5.2475E+04	5.6562E+04
81	5.9462E+04	6.2511E+04	6.7379E+04	7.6351E+04	8.6517E+04
86	9.8037E+04	1.1109E+05	1.1679E+05	1.2277E+05	1.2907E+05
91	1.3569E+05	1.4264E+05	1.4996E+05	1.5764E+05	1.6163E+05
96	1.6573E+05	1.6992E+05	1.7422E+05	1.8316E+05	1.9255E+05
101	2.0242E+05	2.1280E+05	2.2371E+05	2.3518E+05	2.4724E+05
106	2.5991E+05	2.7324E+05	2.8015E+05	2.8725E+05	2.9452E+05
111	3.0197E+05	3.0962E+05	3.1746E+05	3.3373E+05	3.5084E+05
116	3.6883E+05	3.8774E+05	4.0762E+05	4.2852E+05	4.5049E+05
121	4.7359E+05	4.9787E+05	5.1047E+05	5.2340E+05	5.3665E+05
126	5.5023E+05	5.7844E+05	6.0810E+05	6.3928E+05	6.7206E+05
131	7.0651E+05	7.4274E+05	7.8082E+05	8.2085E+05	8.6294E+05
136	9.0718E+05	9.5369E+05	9.6164E+05	9.7783E+05	1.0026E+06
141	1.0540E+06	1.1080E+06	1.1648E+06	1.1943E+06	1.2246E+06
146	1.2873E+06	1.3534E+06	1.4227E+06	1.4957E+06	1.5335E+06
151	1.5724E+06	1.6122E+06	1.6530E+06	1.7377E+06	1.8268E+06
156	1.8731E+06	1.9205E+06	1.9691E+06	2.0190E+06	2.1225E+06
161	2.2313E+06	2.2689E+06	2.3069E+06	2.3457E+06	2.3653E+06

166	2.3852E+06	2.4251E+06	2.4660E+06	2.5924E+06	2.7253E+06
171	2.8650E+06	3.0119E+06	3.0882E+06	3.1664E+06	3.2465E+06
176	3.3287E+06	3.4994E+06	3.6788E+06	3.8674E+06	4.0657E+06
181	4.2741E+06	4.4933E+06	4.6070E+06	4.7237E+06	4.8432E+06
186	4.9659E+06	5.2205E+06	5.4881E+06	5.7695E+06	5.9156E+06
191	6.0653E+06	6.2189E+06	6.3763E+06	6.5377E+06	6.5924E+06
196	6.6476E+06	6.7032E+06	6.8729E+06	7.0469E+06	7.2253E+06
201	7.4082E+06	7.5957E+06	7.7880E+06	7.9852E+06	8.1873E+06
206	8.3946E+06	8.6071E+06	8.8250E+06	9.0484E+06	9.2774E+06
211	9.5123E+06	9.7531E+06	1.0000E+07	1.0253E+07	1.0513E+07
216	1.0779E+07	1.1052E+07	1.1331E+07	1.1618E+07	1.1912E+07
221	1.2214E+07	1.2523E+07	1.2840E+07	1.3165E+07	1.3499E+07
226	1.3840E+07	1.4191E+07	1.4550E+07	1.4918E+07	1.5296E+07
231	1.5683E+07	1.6080E+07	1.6487E+07	1.6905E+07	1.7333E+07
236	1.7771E+07	1.8221E+07	1.8682E+07	1.9155E+07	1.9640E+07
241	1.9970E+07				

2. IGN=3...LANL 30-Group Structure

This is a small structure with complete energy coverage, and it is very economical to use. This makes it useful for survey calculations for fusion systems, fast fission systems, and radiation shields. The group size at low energies is "one lethargy". At higher energies, some one-half and one-quarter lethargy groups are used. Finally, the fusion energy range is represented by bounds close to even energy values. This structure was used in the CLAW-3 and CLAW-4 libraries[3] that are available from the Radiation Shielding Information Center at the Oak Ridge National Laboratory, and in the MATXS1 and MATXS5 libraries used with the TRANSX code[4].

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.3900E-04	1.5200E-01	4.1400E-01	1.1300E+00	3.0600E+00
6	8.3200E+00	2.2600E+01	6.1400E+01	1.6700E+02	4.5400E+02
11	1.2350E+03	3.3500E+03	9.1200E+03	2.4800E+04	6.7600E+04
16	1.8400E+05	3.0300E+05	5.0000E+05	8.2300E+05	1.3530E+06
21	1.7380E+06	2.2320E+06	2.8650E+06	3.6800E+06	6.0700E+06
26	7.7900E+06	1.0000E+07	1.2000E+07	1.3500E+07	1.5000E+07
31	1.7000E+07				

3. IGN=4...ANL 27-Group Structure

This small group structure with even one-half lethargy spacings has been used for fast breeder reactor studies. It is most appropriate for data that has been collapsed from a finer structure for use in two- or three-dimensional calculations; however, it has also been used for processing code comparison studies.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	5.0435E+00	2.2603E+01	3.7267E+01	6.1442E+01	1.0130E+02
6	1.6702E+02	2.7536E+02	4.5400E+02	7.4852E+02	1.2341E+03
11	2.0347E+03	3.3546E+03	5.5308E+03	9.1188E+03	1.5034E+04
16	2.4788E+04	4.0868E+04	6.7379E+04	1.1109E+05	1.8316E+05
21	3.0197E+05	4.9787E+05	8.2085E+05	1.3534E+06	2.2313E+06
26	3.6788E+06	6.0653E+06	1.0000E+07		

4. IGN=5...RRD 50-Group Structure

This structure was specified for a standardized library[5] of group constants based on ENDF/B-IV to be used in the U.S. Fast Breeder Reactor Program (base technology was then under the Office of Reactor Research and Development, RRD). It is still useful for this application, but comparisons with more detailed structures suggested that more groups were needed in the region where U-238 epithermal capture is important and in the region of the 27 keV iron resonance (see the 70- and 80-group structures below). The RRD structure uses quarter lethargy spacing in the middle energy range, with one-half lethargy groups above 500 keV and below 275 eV.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0000E-05	6.8256E-01	1.1254E+00	1.8554E+00	3.0590E+00
6	5.0435E+00	8.3153E+00	1.3710E+01	2.2603E+01	3.7267E+01
11	6.1442E+01	1.0130E+02	1.6702E+02	2.7536E+02	3.5358E+02
16	4.5400E+02	5.8295E+02	7.4852E+02	9.6112E+02	1.2341E+03
21	1.5846E+03	2.0347E+03	2.6126E+03	3.3546E+03	4.3074E+03
26	5.5308E+03	7.1017E+03	9.1188E+03	1.1709E+04	1.5034E+04
31	1.9305E+04	2.4788E+04	3.1828E+04	4.0868E+04	5.2475E+04
36	6.7379E+04	8.6517E+04	1.1109E+05	1.4264E+05	1.8316E+05
41	2.3518E+05	3.0197E+05	3.8774E+05	4.9787E+05	8.2085E+05
46	1.3534E+06	2.2313E+06	3.6788E+06	6.0653E+06	1.0000E+07
51	1.9971E+07				

5. IGN=6...GAM-I 68-Group Structure

This group structure uses a simple uniform quarter-lethargy spacing. It is accurate for the epithermal portion of thermal reactor calculations, and it has been used extensively for both water-moderated and graphite-moderated designs. In addition to its use in the GAM-I code at General Atomic, this structure is used

in libraries[6] produced for the state-of-the-art EPRI-CELL code developed by the Electric Power Research Institute for the U.S. electric utilities.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	4.1399E-01	5.3158E-01	6.8256E-01	8.7642E-01	1.1254E+00
6	1.4450E+00	1.8554E+00	2.3824E+00	3.0590E+00	3.9279E+00
11	5.0435E+00	6.4760E+00	8.3153E+00	1.0677E+01	1.3710E+01
16	1.7603E+01	2.2603E+01	2.9023E+01	3.7267E+01	4.7851E+01
21	6.1442E+01	7.8893E+01	1.0130E+02	1.3007E+02	1.6702E+02
26	2.1445E+02	2.7536E+02	3.5358E+02	4.5400E+02	5.8295E+02
31	7.4852E+02	9.6112E+02	1.2341E+03	1.5846E+03	2.0347E+03
36	2.6126E+03	3.3546E+03	4.3074E+03	5.5308E+03	7.1017E+03
41	9.1188E+03	1.1709E+04	1.5034E+04	1.9305E+04	2.4788E+04
46	3.1828E+04	4.0868E+04	5.2475E+04	6.7379E+04	8.6517E+04
51	1.1109E+05	1.4264E+05	1.8316E+05	2.3518E+05	3.0197E+05
56	3.8774E+05	4.9787E+05	6.3928E+05	8.2085E+05	1.0540E+06
61	1.3534E+06	1.7377E+06	2.2313E+06	2.8650E+06	3.6788E+06
66	4.7237E+06	6.0653E+06	7.7880E+06	1.0000E+07	

6. IGN=7...GAM-II 100-Group Structure

This structure was a successor to the GAM-I set described above. It is the same as GAM-I below 100 keV, but for groups above that point it shifts to one-tenth lethargy. It also includes several groups above 10 MeV to represent the D-T fusion source and the high-energy tail of the fission spectrum. The upper limit of the highest group was changed at Los Alamos (from 14.918 to 17.0 MeV) to permit more accurate averaging over the fusion peak.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0000E-05	4.1399E-01	5.3158E-01	6.8256E-01	8.7642E-01
6	1.1254E+00	1.4450E+00	1.8554E+00	2.3824E+00	3.0590E+00
11	3.9279E+00	5.0435E+00	6.4760E+00	8.3153E+00	1.0677E+01
16	1.3710E+01	1.7603E+01	2.2603E+01	2.9023E+01	3.7267E+01
21	4.7851E+01	6.1442E+01	7.8893E+01	1.0130E+02	1.3007E+02
26	1.6702E+02	2.1445E+02	2.7536E+02	3.5358E+02	4.5400E+02
31	5.8295E+02	7.4852E+02	9.6112E+02	1.2341E+03	1.5846E+03
36	2.0347E+03	2.6126E+03	3.3546E+03	4.3074E+03	5.5308E+03
41	7.1017E+03	9.1188E+03	1.1709E+04	1.5034E+04	1.9305E+04
46	2.4788E+04	3.1828E+04	4.0868E+04	5.2475E+04	6.7379E+04
51	8.6517E+04	1.1109E+05	1.2277E+05	1.3569E+05	1.4996E+05
56	1.6573E+05	1.8316E+05	2.0242E+05	2.2371E+05	2.4724E+05
61	2.7324E+05	3.0197E+05	3.3373E+05	3.6883E+05	4.0762E+05

66	4.5049E+05	4.9787E+05	5.5023E+05	6.0810E+05	6.7206E+05
71	7.4274E+05	8.2085E+05	9.0718E+05	1.0026E+06	1.1080E+06
76	1.2246E+06	1.3534E+06	1.4957E+06	1.6530E+06	1.8268E+06
81	2.0190E+06	2.2313E+06	2.4660E+06	2.7253E+06	3.0119E+06
86	3.3287E+06	3.6788E+06	4.0657E+06	4.4933E+06	4.9659E+06
91	5.4881E+06	6.0653E+06	6.7032E+06	7.4082E+06	8.1873E+06
96	9.0484E+06	1.0000E+07	1.1052E+07	1.2214E+07	1.3499E+07
101	1.7000E+07				

7. IGN=8...LASER-THERMOS 35-Group Structure

This is a thermal group structure. It is based on the THERMOS equal-velocity structure with additional groups added in the vicinity of the Pu-240 resonance. It is used for the thermal calculation in EPRI-CELL.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	2.5300E-04	2.2770E-03	6.3250E-03	1.2397E-02	2.0493E-02
6	3.0613E-02	4.2757E-02	5.6925E-02	8.1972E-02	1.1159E-01
11	1.4573E-01	1.8444E-01	2.2770E-01	2.5104E-01	2.7053E-01
16	2.9075E-01	3.0113E-01	3.2064E-01	3.5768E-01	4.1704E-01
21	5.0326E-01	6.2493E-01	7.8211E-01	9.5070E-01	1.0137E+00
26	1.0428E+00	1.0525E+00	1.0624E+00	1.0722E+00	1.0987E+00
31	1.1664E+00	1.3079E+00	1.4575E+00	1.5950E+00	1.7262E+00
36	1.8550E+00				

8. IGN=9...EPRI-CPM 69-Group Structure

This group structure is designed for water-moderated thermal reactor data sets. It is the same as the WIMS structure, and it was used at Los Alamos to prepare libraries[6] for the EPRI-CPM code. The portion above 4 eV is basically one-half lethargy, except that the boundaries of several groups have been shifted to move important resonances close to the center of their groups. The portion below 4 eV is a detailed thermal structure including extra detail for the Pu-240 resonance.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0000E-05	5.0000E-03	1.0000E-02	1.5000E-02	2.0000E-02
6	2.5000E-02	3.0000E-02	3.5000E-02	4.2000E-02	5.0000E-02
11	5.8000E-02	6.7000E-02	8.0000E-02	1.0000E-01	1.4000E-01
16	1.8000E-01	2.2000E-01	2.5000E-01	2.8000E-01	3.0000E-01
21	3.2000E-01	3.5000E-01	4.0000E-01	5.0000E-01	6.2500E-01

26	7.8000E-01	8.5000E-01	9.1000E-01	9.5000E-01	9.7200E-01
31	9.9600E-01	1.0200E+00	1.0450E+00	1.0710E+00	1.0970E+00
36	1.1230E+00	1.1500E+00	1.3000E+00	1.5000E+00	2.1000E+00
41	2.6000E+00	3.3000E+00	4.0000E+00	9.8770E+00	1.5968E+01
46	2.7700E+01	4.8052E+01	7.5501E+01	1.4873E+02	3.6726E+02
51	9.0690E+02	1.4251E+03	2.2395E+03	3.5191E+03	5.5300E+03
56	9.1180E+03	1.5030E+04	2.4780E+04	4.0850E+04	6.7340E+04
61	1.1100E+05	1.8300E+05	3.0250E+05	5.0000E+05	8.2100E+05
66	1.3530E+06	2.2310E+06	3.6790E+06	6.0655E+06	1.0000E+07

9. IGN=10...LANL 187-Group Structure

This structure is intended to be a single set of fine-lethargy bounds that can be used for thermal reactors, fast breeder reactors, fusion systems, and shielding calculations. As such, it has a thermal structure below 4 eV that is somewhat finer than either the 69-group or the 35-group structures. It is superior to all the other structures in the epithermal range (say, 4 eV to 100 keV). Finally, it has a high-energy structure that is finer than the 80-group set, but not as detailed as the 100- or 240-group structures.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0000E-05	2.5399E-04	7.6022E-04	2.2769E-03	6.3247E-03
6	1.2396E-02	2.0492E-02	2.5500E-02	3.0612E-02	3.5500E-02
11	4.2755E-02	5.0000E-02	5.6922E-02	6.7000E-02	8.1968E-02
16	1.1157E-01	1.4572E-01	1.5230E-01	1.8443E-01	2.2769E-01
21	2.5103E-01	2.7052E-01	2.9074E-01	3.0112E-01	3.2063E-01
26	3.5767E-01	4.1499E-01	5.0323E-01	6.2506E-01	7.8208E-01
31	8.3368E-01	8.7642E-01	9.1000E-01	9.5065E-01	9.7100E-01
36	9.9200E-01	1.0137E+00	1.0427E+00	1.0525E+00	1.0623E+00
41	1.0722E+00	1.0987E+00	1.1254E+00	1.1664E+00	1.3079E+00
46	1.4574E+00	1.5949E+00	1.7261E+00	1.8554E+00	2.1024E+00
51	2.3824E+00	2.6996E+00	3.0590E+00	3.4663E+00	3.9279E+00
56	4.4509E+00	5.0435E+00	5.7150E+00	6.4760E+00	6.8680E+00
61	7.3382E+00	8.3153E+00	9.4225E+00	1.0677E+01	1.2099E+01
66	1.3710E+01	1.5535E+01	1.7603E+01	1.9947E+01	2.2603E+01
71	2.5613E+01	2.9023E+01	3.2888E+01	3.7267E+01	4.2229E+01
76	4.7851E+01	5.4222E+01	6.1442E+01	6.9623E+01	7.8893E+01
81	8.9398E+01	1.0130E+02	1.1479E+02	1.3007E+02	1.4739E+02
86	1.6702E+02	1.8926E+02	2.1445E+02	2.4301E+02	2.7536E+02
91	3.1203E+02	3.5358E+02	4.0065E+02	4.5400E+02	5.1445E+02
96	5.8295E+02	6.6057E+02	7.4852E+02	8.4818E+02	9.6112E+02
101	1.0891E+03	1.2341E+03	1.3984E+03	1.5846E+03	1.7956E+03
106	2.0347E+03	2.3056E+03	2.6126E+03	2.9604E+03	3.3546E+03
111	3.8013E+03	4.3074E+03	4.8810E+03	5.5308E+03	6.2673E+03
116	7.1017E+03	8.0473E+03	9.1188E+03	1.0333E+04	1.1709E+04
121	1.3268E+04	1.5034E+04	1.7036E+04	1.9305E+04	2.1875E+04

126	2.4788E+04	2.6058E+04	2.8088E+04	3.1828E+04	3.6066E+04
131	4.0868E+04	4.6309E+04	5.2475E+04	5.9462E+04	6.7379E+04
136	7.6351E+04	8.6517E+04	9.8037E+04	1.1109E+05	1.2588E+05
141	1.4264E+05	1.6163E+05	1.8316E+05	2.0754E+05	2.3518E+05
146	2.6649E+05	3.0197E+05	3.4218E+05	3.8774E+05	4.3937E+05
151	4.9787E+05	5.6416E+05	6.3928E+05	7.2440E+05	8.2085E+05
156	9.3014E+05	1.0540E+06	1.1943E+06	1.3534E+06	1.5335E+06
161	1.7377E+06	1.9691E+06	2.2313E+06	2.5284E+06	2.8650E+06
166	3.2465E+06	3.6788E+06	4.1686E+06	4.7237E+06	5.3526E+06
171	6.0653E+06	6.8729E+06	7.7880E+06	8.8250E+06	1.0000E+07
176	1.1000E+07	1.2000E+07	1.3000E+07	1.3500E+07	1.3750E+07
181	1.3940E+07	1.4200E+07	1.4420E+07	1.4640E+07	1.5000E+07
186	1.6000E+07	1.7000E+07	2.0000E+07		

10. IGN=11...LANL 70-Group Structure

This structure was designed for the ENDF/B-V follow-on to the 50-group library. More groups were added in the region between 450 eV and 3 keV where U-238 capture is important in fast breeder reactors, and a few more groups were added for important resonances in iron and oxygen. The low limit was left at a relatively high 10 eV to discourage the use of the corresponding library (which was produced with a specific fast-reactor weight function) at inappropriate energies. Later consideration showed this to be a mistake, because the library doesn't work well in the outlying blanket and shield regions of some benchmark assemblies and reactor systems. The 80-group structure is preferred.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0677E+01	6.1442E+01	1.0130E+02	1.3007E+02	1.6702E+02
6	2.1445E+02	2.7537E+02	3.5358E+02	4.5400E+02	5.8295E+02
11	7.4852E+02	9.6112E+02	1.0891E+03	1.2341E+03	1.3984E+03
16	1.5846E+03	1.7956E+03	2.0347E+03	2.3056E+03	2.6126E+03
21	2.9605E+03	3.3546E+03	3.8013E+03	4.3074E+03	4.8810E+03
26	5.5308E+03	6.2673E+03	7.1017E+03	8.0473E+03	9.1188E+03
31	1.0333E+04	1.1709E+04	1.3268E+04	1.5034E+04	1.7036E+04
36	1.9305E+04	2.1875E+04	2.4788E+04	2.8088E+04	3.1828E+04
41	4.0868E+04	5.2475E+04	6.7380E+04	8.6517E+04	1.1109E+05
46	1.4264E+05	1.8316E+05	2.3518E+05	3.0197E+05	3.8774E+05
51	4.3937E+05	4.9787E+05	5.6416E+05	6.3928E+05	7.2440E+05
56	8.2085E+05	9.3015E+05	1.0540E+06	1.1943E+06	1.3534E+06
61	1.7377E+06	2.2313E+06	2.8651E+06	3.6788E+06	4.7237E+06
66	6.0653E+06	7.7880E+06	1.0000E+07	1.2840E+07	1.6487E+07
71	2.0000E+07				

11. IGN=12...SAND-II 620-Group Structure

This very fine group structure is used in dosimetry applications for flux unfolding. It has also been used for processing code comparisons[7], where it is almost as accurate as pointwise cross sections without the complication of different grid structures for different codes. Many users now use the 640-group version of this structure (see IGN=15 below) that has 20 additional groups above 18 MeV for high-energy dosimetry.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0000E-04	1.0500E-04	1.1000E-04	1.1500E-04	1.2000E-04
6	1.2750E-04	1.3500E-04	1.4250E-04	1.5000E-04	1.6000E-04
11	1.7000E-04	1.8000E-04	1.9000E-04	2.0000E-04	2.1000E-04
16	2.2000E-04	2.3000E-04	2.4000E-04	2.5500E-04	2.7000E-04
21	2.8000E-04	3.0000E-04	3.2000E-04	3.4000E-04	3.6000E-04
26	3.8000E-04	4.0000E-04	4.2500E-04	4.5000E-04	4.7500E-04
31	5.0000E-04	5.2500E-04	5.5000E-04	5.7500E-04	6.0000E-04
36	6.3000E-04	6.6000E-04	6.9000E-04	7.2000E-04	7.6000E-04
41	8.0000E-04	8.4000E-04	8.8000E-04	9.2000E-04	9.6000E-04
46	1.0000E-03	1.0500E-03	1.1000E-03	1.1500E-03	1.2000E-03
51	1.2750E-03	1.3500E-03	1.4250E-03	1.5000E-03	1.6000E-03
56	1.7000E-03	1.8000E-03	1.9000E-03	2.0000E-03	2.1000E-03
61	2.2000E-03	2.3000E-03	2.4000E-03	2.5500E-03	2.7000E-03
66	2.8000E-03	3.0000E-03	3.2000E-03	3.4000E-03	3.6000E-03
71	3.8000E-03	4.0000E-03	4.2500E-03	4.5000E-03	4.7500E-03
76	5.0000E-03	5.2500E-03	5.5000E-03	5.7500E-03	6.0000E-03
81	6.3000E-03	6.6000E-03	6.9000E-03	7.2000E-03	7.6000E-03
86	8.0000E-03	8.4000E-03	8.8000E-03	9.2000E-03	9.6000E-03
91	1.0000E-02	1.0500E-02	1.1000E-02	1.1500E-02	1.2000E-02
96	1.2750E-02	1.3500E-02	1.4250E-02	1.5000E-02	1.6000E-02
101	1.7000E-02	1.8000E-02	1.9000E-02	2.0000E-02	2.1000E-02
106	2.2000E-02	2.3000E-02	2.4000E-02	2.5500E-02	2.7000E-02
111	2.8000E-02	3.0000E-02	3.2000E-02	3.4000E-02	3.6000E-02
116	3.8000E-02	4.0000E-02	4.2500E-02	4.5000E-02	4.7500E-02
121	5.0000E-02	5.2500E-02	5.5000E-02	5.7500E-02	6.0000E-02
126	6.3000E-02	6.6000E-02	6.9000E-02	7.2000E-02	7.6000E-02
131	8.0000E-02	8.4000E-02	8.8000E-02	9.2000E-02	9.6000E-02
136	1.0000E-01	1.0500E-01	1.1000E-01	1.1500E-01	1.2000E-01
141	1.2750E-01	1.3500E-01	1.4250E-01	1.5000E-01	1.6000E-01
146	1.7000E-01	1.8000E-01	1.9000E-01	2.0000E-01	2.1000E-01
151	2.2000E-01	2.3000E-01	2.4000E-01	2.5500E-01	2.7000E-01
156	2.8000E-01	3.0000E-01	3.2000E-01	3.4000E-01	3.6000E-01
161	3.8000E-01	4.0000E-01	4.2500E-01	4.5000E-01	4.7500E-01
166	5.0000E-01	5.2500E-01	5.5000E-01	5.7500E-01	6.0000E-01
171	6.3000E-01	6.6000E-01	6.9000E-01	7.2000E-01	7.6000E-01
176	8.0000E-01	8.4000E-01	8.8000E-01	9.2000E-01	9.6000E-01
181	1.0000E+00	1.0500E+00	1.1000E+00	1.1500E+00	1.2000E+00

186	1.2750E+00	1.3500E+00	1.4250E+00	1.5000E+00	1.6000E+00
191	1.7000E+00	1.8000E+00	1.9000E+00	2.0000E+00	2.1000E+00
196	2.2000E+00	2.3000E+00	2.4000E+00	2.5500E+00	2.7000E+00
201	2.8000E+00	3.0000E+00	3.2000E+00	3.4000E+00	3.6000E+00
206	3.8000E+00	4.0000E+00	4.2500E+00	4.5000E+00	4.7500E+00
211	5.0000E+00	5.2500E+00	5.5000E+00	5.7500E+00	6.0000E+00
216	6.3000E+00	6.6000E+00	6.9000E+00	7.2000E+00	7.6000E+00
221	8.0000E+00	8.4000E+00	8.8000E+00	9.2000E+00	9.6000E+00
226	1.0000E+01	1.0500E+01	1.1000E+01	1.1500E+01	1.2000E+01
231	1.2750E+01	1.3500E+01	1.4250E+01	1.5000E+01	1.6000E+01
236	1.7000E+01	1.8000E+01	1.9000E+01	2.0000E+01	2.1000E+01
241	2.2000E+01	2.3000E+01	2.4000E+01	2.5500E+01	2.7000E+01
246	2.8000E+01	3.0000E+01	3.2000E+01	3.4000E+01	3.6000E+01
251	3.8000E+01	4.0000E+01	4.2500E+01	4.5000E+01	4.7500E+01
256	5.0000E+01	5.2500E+01	5.5000E+01	5.7500E+01	6.0000E+01
261	6.3000E+01	6.6000E+01	6.9000E+01	7.2000E+01	7.6000E+01
266	8.0000E+01	8.4000E+01	8.8000E+01	9.2000E+01	9.6000E+01
271	1.0000E+02	1.0500E+02	1.1000E+02	1.1500E+02	1.2000E+02
276	1.2750E+02	1.3500E+02	1.4250E+02	1.5000E+02	1.6000E+02
281	1.7000E+02	1.8000E+02	1.9000E+02	2.0000E+02	2.1000E+02
286	2.2000E+02	2.3000E+02	2.4000E+02	2.5500E+02	2.7000E+02
291	2.8000E+02	3.0000E+02	3.2000E+02	3.4000E+02	3.6000E+02
296	3.8000E+02	4.0000E+02	4.2500E+02	4.5000E+02	4.7500E+02
301	5.0000E+02	5.2500E+02	5.5000E+02	5.7500E+02	6.0000E+02
306	6.3000E+02	6.6000E+02	6.9000E+02	7.2000E+02	7.6000E+02
311	8.0000E+02	8.4000E+02	8.8000E+02	9.2000E+02	9.6000E+02
316	1.0000E+03	1.0500E+03	1.1000E+03	1.1500E+03	1.2000E+03
321	1.2750E+03	1.3500E+03	1.4250E+03	1.5000E+03	1.6000E+03
326	1.7000E+03	1.8000E+03	1.9000E+03	2.0000E+03	2.1000E+03
331	2.2000E+03	2.3000E+03	2.4000E+03	2.5500E+03	2.7000E+03
336	2.8000E+03	3.0000E+03	3.2000E+03	3.4000E+03	3.6000E+03
341	3.8000E+03	4.0000E+03	4.2500E+03	4.5000E+03	4.7500E+03
346	5.0000E+03	5.2500E+03	5.5000E+03	5.7500E+03	6.0000E+03
351	6.3000E+03	6.6000E+03	6.9000E+03	7.2000E+03	7.6000E+03
356	8.0000E+03	8.4000E+03	8.8000E+03	9.2000E+03	9.6000E+03
361	1.0000E+04	1.0500E+04	1.1000E+04	1.1500E+04	1.2000E+04
366	1.2750E+04	1.3500E+04	1.4250E+04	1.5000E+04	1.6000E+04
371	1.7000E+04	1.8000E+04	1.9000E+04	2.0000E+04	2.1000E+04
376	2.2000E+04	2.3000E+04	2.4000E+04	2.5500E+04	2.7000E+04
381	2.8000E+04	3.0000E+04	3.2000E+04	3.4000E+04	3.6000E+04
386	3.8000E+04	4.0000E+04	4.2500E+04	4.5000E+04	4.7500E+04
391	5.0000E+04	5.2500E+04	5.5000E+04	5.7500E+04	6.0000E+04
396	6.3000E+04	6.6000E+04	6.9000E+04	7.2000E+04	7.6000E+04
401	8.0000E+04	8.4000E+04	8.8000E+04	9.2000E+04	9.6000E+04
406	1.0000E+05	1.0500E+05	1.1000E+05	1.1500E+05	1.2000E+05
411	1.2750E+05	1.3500E+05	1.4250E+05	1.5000E+05	1.6000E+05
416	1.7000E+05	1.8000E+05	1.9000E+05	2.0000E+05	2.1000E+05
421	2.2000E+05	2.3000E+05	2.4000E+05	2.5500E+05	2.7000E+05
426	2.8000E+05	3.0000E+05	3.2000E+05	3.4000E+05	3.6000E+05
431	3.8000E+05	4.0000E+05	4.2500E+05	4.5000E+05	4.7500E+05
436	5.0000E+05	5.2500E+05	5.5000E+05	5.7500E+05	6.0000E+05
441	6.3000E+05	6.6000E+05	6.9000E+05	7.2000E+05	7.6000E+05
446	8.0000E+05	8.4000E+05	8.8000E+05	9.2000E+05	9.6000E+05

```
451     1.0000E+06  1.1000E+06  1.2000E+06  1.3000E+06  1.4000E+06
456     1.5000E+06  1.6000E+06  1.7000E+06  1.8000E+06  1.9000E+06
461     2.0000E+06  2.1000E+06  2.2000E+06  2.3000E+06  2.4000E+06
466     2.5000E+06  2.6000E+06  2.7000E+06  2.8000E+06  2.9000E+06
471     3.0000E+06  3.1000E+06  3.2000E+06  3.3000E+06  3.4000E+06
476     3.5000E+06  3.6000E+06  3.7000E+06  3.8000E+06  3.9000E+06
481     4.0000E+06  4.1000E+06  4.2000E+06  4.3000E+06  4.4000E+06
486     4.5000E+06  4.6000E+06  4.7000E+06  4.8000E+06  4.9000E+06
491     5.0000E+06  5.1000E+06  5.2000E+06  5.3000E+06  5.4000E+06
496     5.5000E+06  5.6000E+06  5.7000E+06  5.8000E+06  5.9000E+06
501     6.0000E+06  6.1000E+06  6.2000E+06  6.3000E+06  6.4000E+06
506     6.5000E+06  6.6000E+06  6.7000E+06  6.8000E+06  6.9000E+06
511     7.0000E+06  7.1000E+06  7.2000E+06  7.3000E+06  7.4000E+06
516     7.5000E+06  7.6000E+06  7.7000E+06  7.8000E+06  7.9000E+06
521     8.0000E+06  8.1000E+06  8.2000E+06  8.3000E+06  8.4000E+06
526     8.5000E+06  8.6000E+06  8.7000E+06  8.8000E+06  8.9000E+06
531     9.0000E+06  9.1000E+06  9.2000E+06  9.3000E+06  9.4000E+06
536     9.5000E+06  9.6000E+06  9.7000E+06  9.8000E+06  9.9000E+06
541     1.0000E+07  1.0100E+07  1.0200E+07  1.0300E+07  1.0400E+07
546     1.0500E+07  1.0600E+07  1.0700E+07  1.0800E+07  1.0900E+07
551     1.1000E+07  1.1100E+07  1.1200E+07  1.1300E+07  1.1400E+07
556     1.1500E+07  1.1600E+07  1.1700E+07  1.1800E+07  1.1900E+07
561     1.2000E+07  1.2100E+07  1.2200E+07  1.2300E+07  1.2400E+07
566     1.2500E+07  1.2600E+07  1.2700E+07  1.2800E+07  1.2900E+07
571     1.3000E+07  1.3100E+07  1.3200E+07  1.3300E+07  1.3400E+07
576     1.3500E+07  1.3600E+07  1.3700E+07  1.3800E+07  1.3900E+07
581     1.4000E+07  1.4100E+07  1.4200E+07  1.4300E+07  1.4400E+07
586     1.4500E+07  1.4600E+07  1.4700E+07  1.4800E+07  1.4900E+07
591     1.5000E+07  1.5100E+07  1.5200E+07  1.5300E+07  1.5400E+07
596     1.5500E+07  1.5600E+07  1.5700E+07  1.5800E+07  1.5900E+07
601     1.6000E+07  1.6100E+07  1.6200E+07  1.6300E+07  1.6400E+07
606     1.6500E+07  1.6600E+07  1.6700E+07  1.6800E+07  1.6900E+07
611     1.7000E+07  1.7100E+07  1.7200E+07  1.7300E+07  1.7400E+07
616     1.7500E+07  1.7600E+07  1.7700E+07  1.7800E+07  1.7900E+07
621     1.8000E+07
```

12. IGN=13...LANL 80-Group Structure

This structure has been optimized for fast breeder reactors and fusion systems. The resonance-region treatment uses numerous one-eighth lethargy groups to give an accurate treatment of U-238 capture, and extra group bounds are provided around the important sodium resonance near 3 keV and the iron resonance near 27 keV. A fairly high level of resolution is used in the fission spectrum range to accommodate the differences between different reactor designs and fast critical assemblies such a GODIVA and JEZEBEL, and to represent the resonances and "window" in oxygen. The structure above 10 MeV is similar to that of the 30-group set, which is adequate for practical fusion blanket calculations of tritium production and radiation damage. A finer structure should only be needed for careful

high-threshold dosimetry work. Finally, a few low-energy groups were provided to represent the soft flux in outlying blanket and shield regions of reactor systems. The 80-group structure is recommended over the 50- or 70-group structures for these kinds of problems.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.3888E-04	1.5230E-01	4.1399E-01	1.1254E+00	3.0590E+00
6	5.0435E+00	8.3153E+00	1.3710E+01	2.2603E+01	3.7267E+01
11	6.1442E+01	1.0130E+02	1.6702E+02	2.7536E+02	3.5358E+02
16	4.5400E+02	5.8295E+02	7.4852E+02	9.6112E+02	1.0891E+03
21	1.2341E+03	1.3984E+03	1.5846E+03	1.7956E+03	2.0347E+03
26	2.3056E+03	2.6126E+03	2.9604E+03	3.3546E+03	3.8013E+03
31	4.3074E+03	4.8810E+03	5.5308E+03	6.2673E+03	7.1017E+03
36	8.0473E+03	9.1188E+03	1.0333E+04	1.1709E+04	1.3268E+04
41	1.5034E+04	1.7036E+04	1.9305E+04	2.1875E+04	2.4788E+04
46	2.6058E+04	2.8088E+04	3.1828E+04	4.0868E+04	5.2475E+04
51	6.7379E+04	8.6517E+04	1.1109E+05	1.4264E+05	1.8316E+05
56	2.3518E+05	3.0197E+05	3.8774E+05	4.3937E+05	4.9787E+05
61	5.6416E+05	6.3928E+05	7.2440E+05	8.2085E+05	9.3014E+05
66	1.0540E+06	1.1943E+06	1.3534E+06	1.7377E+06	2.2313E+06
71	2.8650E+06	3.6788E+06	4.7237E+06	6.0653E+06	7.7880E+06
76	1.0000E+07	1.1912E+07	1.3499E+07	1.4918E+07	1.6905E+07
81	2.0000E+07				

13. IGN=14...EURLIB 100-Group Structure

This structure has been used extensively in Western Europe in the analysis of integral experiments and in NEA shielding benchmark exercises. It is very similar to the constant-lethargy-step GAM-II structure (see IGN=7 above). The only changes are the removal of 7 of the 16 group boundaries below 17 eV, the insertion of 4 new boundaries at high energy (7.0469, 6.3763, 4.7237, and 2.3457 MeV), and the addition of 3 new boundaries near the iron minimum (26.058, 23.579, and 21.875 keV).

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0000E-05	4.1399E-01	6.2506E-01	1.1254E+00	1.8554E+00
6	3.0590E+00	5.0435E+00	8.3153E+00	1.0677E+01	1.7603E+01
11	2.2603E+01	2.9023E+01	3.7267E+01	4.7851E+01	6.1442E+01
16	7.8893E+01	1.0130E+02	1.3007E+02	1.6702E+02	2.1445E+02
21	2.7536E+02	3.5358E+02	4.5400E+02	5.8295E+02	7.4852E+02
26	9.6112E+02	1.2341E+03	1.5846E+03	2.0347E+03	2.6126E+03

31	3.3546E+03	4.3074E+03	5.5308E+03	7.1017E+03	9.1188E+03
36	1.1709E+04	1.5034E+04	1.9305E+04	2.1875E+04	2.3579E+04
41	2.4788E+04	2.6058E+04	3.1828E+04	4.0868E+04	5.2475E+04
46	6.7379E+04	8.6517E+04	1.1109E+05	1.2277E+05	1.3569E+05
51	1.4996E+05	1.6573E+05	1.8316E+05	2.0242E+05	2.2371E+05
56	2.4724E+05	2.7324E+05	3.0197E+05	3.3373E+05	3.6883E+05
61	4.0762E+05	4.5049E+05	4.9787E+05	5.5023E+05	6.0810E+05
66	6.7206E+05	7.4274E+05	8.2085E+05	9.0718E+05	1.0026E+06
71	1.1080E+06	1.2246E+06	1.3534E+06	1.4957E+06	1.6530E+06
76	1.8268E+06	2.0190E+06	2.2313E+06	2.3457E+06	2.4660E+06
81	2.7253E+06	3.0119E+06	3.3287E+06	3.6788E+06	4.0657E+06
86	4.4933E+06	4.7237E+06	4.9659E+06	5.4881E+06	6.0653E+06
91	6.3763E+06	6.7032E+06	7.0469E+06	7.4082E+06	8.1873E+06
96	9.0484E+06	1.0000E+07	1.1052E+07	1.2214E+07	1.3499E+07
101	1.4918E+07				

14. IGN=15...SAND-IIA 640-Group Structure

This fine-group structure is used for dosimetry work with flux unfolding codes. It is an extension of the 620-group structure (see above) that includes 20 extra groups above 18 MeV.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0000E-04	1.0500E-04	1.1000E-04	1.1500E-04	1.2000E-04
6	1.2750E-04	1.3500E-04	1.4250E-04	1.5000E-04	1.6000E-04
11	1.7000E-04	1.8000E-04	1.9000E-04	2.0000E-04	2.1000E-04
16	2.2000E-04	2.3000E-04	2.4000E-04	2.5500E-04	2.7000E-04
21	2.8000E-04	3.0000E-04	3.2000E-04	3.4000E-04	3.6000E-04
26	3.8000E-04	4.0000E-04	4.2500E-04	4.5000E-04	4.7500E-04
31	5.0000E-04	5.2500E-04	5.5000E-04	5.7500E-04	6.0000E-04
36	6.3000E-04	6.6000E-04	6.9000E-04	7.2000E-04	7.6000E-04
41	8.0000E-04	8.4000E-04	8.8000E-04	9.2000E-04	9.6000E-04
46	1.0000E-03	1.0500E-03	1.1000E-03	1.1500E-03	1.2000E-03
51	1.2750E-03	1.3500E-03	1.4250E-03	1.5000E-03	1.6000E-03
56	1.7000E-03	1.8000E-03	1.9000E-03	2.0000E-03	2.1000E-03
61	2.2000E-03	2.3000E-03	2.4000E-03	2.5500E-03	2.7000E-03
66	2.8000E-03	3.0000E-03	3.2000E-03	3.4000E-03	3.6000E-03
71	3.8000E-03	4.0000E-03	4.2500E-03	4.5000E-03	4.7500E-03
76	5.0000E-03	5.2500E-03	5.5000E-03	5.7500E-03	6.0000E-03
81	6.3000E-03	6.6000E-03	6.9000E-03	7.2000E-03	7.6000E-03
86	8.0000E-03	8.4000E-03	8.8000E-03	9.2000E-03	9.6000E-03
91	1.0000E-02	1.0500E-02	1.1000E-02	1.1500E-02	1.2000E-02
96	1.2750E-02	1.3500E-02	1.4250E-02	1.5000E-02	1.6000E-02
101	1.7000E-02	1.8000E-02	1.9000E-02	2.0000E-02	2.1000E-02
106	2.2000E-02	2.3000E-02	2.4000E-02	2.5500E-02	2.7000E-02
111	2.8000E-02	3.0000E-02	3.2000E-02	3.4000E-02	3.6000E-02
116	3.8000E-02	4.0000E-02	4.2500E-02	4.5000E-02	4.7500E-02

121	5.0000E-02	5.2500E-02	5.5000E-02	5.7500E-02	6.0000E-02
126	6.3000E-02	6.6000E-02	6.9000E-02	7.2000E-02	7.6000E-02
131	8.0000E-02	8.4000E-02	8.8000E-02	9.2000E-02	9.6000E-02
136	1.0000E-01	1.0500E-01	1.1000E-01	1.1500E-01	1.2000E-01
141	1.2750E-01	1.3500E-01	1.4250E-01	1.5000E-01	1.6000E-01
146	1.7000E-01	1.8000E-01	1.9000E-01	2.0000E-01	2.1000E-01
151	2.2000E-01	2.3000E-01	2.4000E-01	2.5500E-01	2.7000E-01
156	2.8000E-01	3.0000E-01	3.2000E-01	3.4000E-01	3.6000E-01
161	3.8000E-01	4.0000E-01	4.2500E-01	4.5000E-01	4.7500E-01
166	5.0000E-01	5.2500E-01	5.5000E-01	5.7500E-01	6.0000E-01
171	6.3000E-01	6.6000E-01	6.9000E-01	7.2000E-01	7.6000E-01
176	8.0000E-01	8.4000E-01	8.8000E-01	9.2000E-01	9.6000E-01
181	1.0000E+00	1.0500E+00	1.1000E+00	1.1500E+00	1.2000E+00
186	1.2750E+00	1.3500E+00	1.4250E+00	1.5000E+00	1.6000E+00
191	1.7000E+00	1.8000E+00	1.9000E+00	2.0000E+00	2.1000E+00
196	2.2000E+00	2.3000E+00	2.4000E+00	2.5500E+00	2.7000E+00
201	2.8000E+00	3.0000E+00	3.2000E+00	3.4000E+00	3.6000E+00
206	3.8000E+00	4.0000E+00	4.2500E+00	4.5000E+00	4.7500E+00
211	5.0000E+00	5.2500E+00	5.5000E+00	5.7500E+00	6.0000E+00
216	6.3000E+00	6.6000E+00	6.9000E+00	7.2000E+00	7.6000E+00
221	8.0000E+00	8.4000E+00	8.8000E+00	9.2000E+00	9.6000E+00
226	1.0000E+01	1.0500E+01	1.1000E+01	1.1500E+01	1.2000E+01
231	1.2750E+01	1.3500E+01	1.4250E+01	1.5000E+01	1.6000E+01
236	1.7000E+01	1.8000E+01	1.9000E+01	2.0000E+01	2.1000E+01
241	2.2000E+01	2.3000E+01	2.4000E+01	2.5500E+01	2.7000E+01
246	2.8000E+01	3.0000E+01	3.2000E+01	3.4000E+01	3.6000E+01
251	3.8000E+01	4.0000E+01	4.2500E+01	4.5000E+01	4.7500E+01
256	5.0000E+01	5.2500E+01	5.5000E+01	5.7500E+01	6.0000E+01
261	6.3000E+01	6.6000E+01	6.9000E+01	7.2000E+01	7.6000E+01
266	8.0000E+01	8.4000E+01	8.8000E+01	9.2000E+01	9.6000E+01
271	1.0000E+02	1.0500E+02	1.1000E+02	1.1500E+02	1.2000E+02
276	1.2750E+02	1.3500E+02	1.4250E+02	1.5000E+02	1.6000E+02
281	1.7000E+02	1.8000E+02	1.9000E+02	2.0000E+02	2.1000E+02
286	2.2000E+02	2.3000E+02	2.4000E+02	2.5500E+02	2.7000E+02
291	2.8000E+02	3.0000E+02	3.2000E+02	3.4000E+02	3.6000E+02
296	3.8000E+02	4.0000E+02	4.2500E+02	4.5000E+02	4.7500E+02
301	5.0000E+02	5.2500E+02	5.5000E+02	5.7500E+02	6.0000E+02
306	6.3000E+02	6.6000E+02	6.9000E+02	7.2000E+02	7.6000E+02
311	8.0000E+02	8.4000E+02	8.8000E+02	9.2000E+02	9.6000E+02
316	1.0000E+03	1.0500E+03	1.1000E+03	1.1500E+03	1.2000E+03
321	1.2750E+03	1.3500E+03	1.4250E+03	1.5000E+03	1.6000E+03
326	1.7000E+03	1.8000E+03	1.9000E+03	2.0000E+03	2.1000E+03
331	2.2000E+03	2.3000E+03	2.4000E+03	2.5500E+03	2.7000E+03
336	2.8000E+03	3.0000E+03	3.2000E+03	3.4000E+03	3.6000E+03
341	3.8000E+03	4.0000E+03	4.2500E+03	4.5000E+03	4.7500E+03
346	5.0000E+03	5.2500E+03	5.5000E+03	5.7500E+03	6.0000E+03
351	6.3000E+03	6.6000E+03	6.9000E+03	7.2000E+03	7.6000E+03
356	8.0000E+03	8.4000E+03	8.8000E+03	9.2000E+03	9.6000E+03
361	1.0000E+04	1.0500E+04	1.1000E+04	1.1500E+04	1.2000E+04
366	1.2750E+04	1.3500E+04	1.4250E+04	1.5000E+04	1.6000E+04
371	1.7000E+04	1.8000E+04	1.9000E+04	2.0000E+04	2.1000E+04
376	2.2000E+04	2.3000E+04	2.4000E+04	2.5500E+04	2.7000E+04
381	2.8000E+04	3.0000E+04	3.2000E+04	3.4000E+04	3.6000E+04

386	3.8000E+04	4.0000E+04	4.2500E+04	4.5000E+04	4.7500E+04
391	5.0000E+04	5.2500E+04	5.5000E+04	5.7500E+04	6.0000E+04
396	6.3000E+04	6.6000E+04	6.9000E+04	7.2000E+04	7.6000E+04
401	8.0000E+04	8.4000E+04	8.8000E+04	9.2000E+04	9.6000E+04
406	1.0000E+05	1.0500E+05	1.1000E+05	1.1500E+05	1.2000E+05
411	1.2750E+05	1.3500E+05	1.4250E+05	1.5000E+05	1.6000E+05
416	1.7000E+05	1.8000E+05	1.9000E+05	2.0000E+05	2.1000E+05
421	2.2000E+05	2.3000E+05	2.4000E+05	2.5500E+05	2.7000E+05
426	2.8000E+05	3.0000E+05	3.2000E+05	3.4000E+05	3.6000E+05
431	3.8000E+05	4.0000E+05	4.2500E+05	4.5000E+05	4.7500E+05
436	5.0000E+05	5.2500E+05	5.5000E+05	5.7500E+05	6.0000E+05
441	6.3000E+05	6.6000E+05	6.9000E+05	7.2000E+05	7.6000E+05
446	8.0000E+05	8.4000E+05	8.8000E+05	9.2000E+05	9.6000E+05
451	1.0000E+06	1.1000E+06	1.2000E+06	1.3000E+06	1.4000E+06
456	1.5000E+06	1.6000E+06	1.7000E+06	1.8000E+06	1.9000E+06
461	2.0000E+06	2.1000E+06	2.2000E+06	2.3000E+06	2.4000E+06
466	2.5000E+06	2.6000E+06	2.7000E+06	2.8000E+06	2.9000E+06
471	3.0000E+06	3.1000E+06	3.2000E+06	3.3000E+06	3.4000E+06
476	3.5000E+06	3.6000E+06	3.7000E+06	3.8000E+06	3.9000E+06
481	4.0000E+06	4.1000E+06	4.2000E+06	4.3000E+06	4.4000E+06
486	4.5000E+06	4.6000E+06	4.7000E+06	4.8000E+06	4.9000E+06
491	5.0000E+06	5.1000E+06	5.2000E+06	5.3000E+06	5.4000E+06
496	5.5000E+06	5.6000E+06	5.7000E+06	5.8000E+06	5.9000E+06
501	6.0000E+06	6.1000E+06	6.2000E+06	6.3000E+06	6.4000E+06
506	6.5000E+06	6.6000E+06	6.7000E+06	6.8000E+06	6.9000E+06
511	7.0000E+06	7.1000E+06	7.2000E+06	7.3000E+06	7.4000E+06
516	7.5000E+06	7.6000E+06	7.7000E+06	7.8000E+06	7.9000E+06
521	8.0000E+06	8.1000E+06	8.2000E+06	8.3000E+06	8.4000E+06
526	8.5000E+06	8.6000E+06	8.7000E+06	8.8000E+06	8.9000E+06
531	9.0000E+06	9.1000E+06	9.2000E+06	9.3000E+06	9.4000E+06
536	9.5000E+06	9.6000E+06	9.7000E+06	9.8000E+06	9.9000E+06
541	1.0000E+07	1.0100E+07	1.0200E+07	1.0300E+07	1.0400E+07
546	1.0500E+07	1.0600E+07	1.0700E+07	1.0800E+07	1.0900E+07
551	1.1000E+07	1.1100E+07	1.1200E+07	1.1300E+07	1.1400E+07
556	1.1500E+07	1.1600E+07	1.1700E+07	1.1800E+07	1.1900E+07
561	1.2000E+07	1.2100E+07	1.2200E+07	1.2300E+07	1.2400E+07
566	1.2500E+07	1.2600E+07	1.2700E+07	1.2800E+07	1.2900E+07
571	1.3000E+07	1.3100E+07	1.3200E+07	1.3300E+07	1.3400E+07
576	1.3500E+07	1.3600E+07	1.3700E+07	1.3800E+07	1.3900E+07
581	1.4000E+07	1.4100E+07	1.4200E+07	1.4300E+07	1.4400E+07
586	1.4500E+07	1.4600E+07	1.4700E+07	1.4800E+07	1.4900E+07
591	1.5000E+07	1.5100E+07	1.5200E+07	1.5300E+07	1.5400E+07
596	1.5500E+07	1.5600E+07	1.5700E+07	1.5800E+07	1.5900E+07
601	1.6000E+07	1.6100E+07	1.6200E+07	1.6300E+07	1.6400E+07
606	1.6500E+07	1.6600E+07	1.6700E+07	1.6800E+07	1.6900E+07
611	1.7000E+07	1.7100E+07	1.7200E+07	1.7300E+07	1.7400E+07
616	1.7500E+07	1.7600E+07	1.7700E+07	1.7800E+07	1.7900E+07
621	1.8000E+07	1.8100E+07	1.8200E+07	1.8300E+07	1.8400E+07
626	1.8500E+07	1.8600E+07	1.8700E+07	1.8800E+07	1.8900E+07
631	1.9000E+07	1.9100E+07	1.9200E+07	1.9300E+07	1.9400E+07
636	1.9500E+07	1.9600E+07	1.9700E+07	1.9800E+07	1.9900E+07
641	2.0000E+07				

15. IGN=16...VITAMIN-E 174-Group Structure

This group structure was used for the VITAMIN-E library[8] generated at the Oak Ridge National Laboratory using ENDF/B-V cross sections. It is an evolution of the VITAMIN-C structure, and it shares many group bounds with the CSEWG 240-group structure (see IGG=2 above).

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0000E-05	1.0000E-01	4.1399E-01	5.3158E-01	6.8256E-01
6	8.7642E-01	1.1254E+00	1.4450E+00	1.8554E+00	2.3824E+00
11	3.0590E+00	3.9279E+00	5.0435E+00	6.4760E+00	8.3153E+00
16	1.0677E+01	1.3710E+01	1.7603E+01	2.2603E+01	2.9023E+01
21	3.7267E+01	4.7851E+01	6.1442E+01	7.8893E+01	1.0130E+02
26	1.3007E+02	1.6702E+02	2.1445E+02	2.7536E+02	3.5358E+02
31	4.5400E+02	5.8295E+02	7.4852E+02	9.6112E+02	1.2341E+03
36	1.5846E+03	2.0347E+03	2.2487E+03	2.4852E+03	2.6126E+03
41	2.7465E+03	3.0354E+03	3.3546E+03	3.7074E+03	4.3074E+03
46	5.5308E+03	7.1017E+03	9.1188E+03	1.0595E+04	1.1709E+04
51	1.5034E+04	1.9305E+04	2.1875E+04	2.3579E+04	2.4176E+04
56	2.4788E+04	2.6058E+04	2.7000E+04	2.8500E+04	3.1828E+04
61	3.4307E+04	4.0868E+04	4.6309E+04	5.2475E+04	5.6562E+04
66	6.7379E+04	7.2000E+04	7.9500E+04	8.2500E+04	8.6517E+04
71	9.8037E+04	1.1109E+05	1.1679E+05	1.2277E+05	1.2907E+05
76	1.3569E+05	1.4264E+05	1.4996E+05	1.5764E+05	1.6573E+05
81	1.7422E+05	1.8316E+05	1.9255E+05	2.0242E+05	2.1280E+05
86	2.2371E+05	2.3518E+05	2.4724E+05	2.7324E+05	2.8725E+05
91	2.9452E+05	2.9720E+05	2.9850E+05	3.0197E+05	3.3373E+05
96	3.6883E+05	3.8774E+05	4.0762E+05	4.5049E+05	4.9787E+05
101	5.2340E+05	5.5023E+05	5.7844E+05	6.0810E+05	6.3928E+05
106	6.7206E+05	7.0651E+05	7.4274E+05	7.8082E+05	8.2085E+05
111	8.6294E+05	9.0718E+05	9.6164E+05	1.0026E+06	1.1080E+06
116	1.1648E+06	1.2246E+06	1.2873E+06	1.3534E+06	1.4227E+06
121	1.4957E+06	1.5724E+06	1.6530E+06	1.7377E+06	1.8268E+06
126	1.9205E+06	2.0190E+06	2.1225E+06	2.2313E+06	2.3069E+06
131	2.3457E+06	2.3653E+06	2.3852E+06	2.4660E+06	2.5924E+06
136	2.7253E+06	2.8650E+06	3.0119E+06	3.1664E+06	3.3287E+06
141	3.6788E+06	4.0657E+06	4.4933E+06	4.7237E+06	4.9659E+06
146	5.2205E+06	5.4881E+06	5.7695E+06	6.0653E+06	6.3763E+06
151	6.5924E+06	6.7032E+06	7.0469E+06	7.4082E+06	7.7880E+06
156	8.1873E+06	8.6071E+06	9.0484E+06	9.5123E+06	1.0000E+07
161	1.0513E+07	1.1052E+07	1.1618E+07	1.2214E+07	1.2523E+07
166	1.3499E+07	1.3840E+07	1.4191E+07	1.4550E+07	1.4918E+07
171	1.5683E+07	1.6487E+07	1.6905E+07	1.7333E+07	1.9640E+07

16. IGN=17...VITAMIN-J 175-Group Structure

The VITAMIN-J structure is being used in Europe for processed cross section libraries based on the JEF-I and JEF-II evaluated cross section libraries (which are in ENDF format). It is identical to the VITAMIN-E structure shown above, except one additional group bound at 1.2840E+07 is inserted between entries 165 and 166. The total number of groups is thus increased by one.

17. IGG=2...CSEWG 94-Group Structure

This is a supergroup structure for photons that was developed at the same time as the 240-group neutron structure given above. As such, it was designed to contain a number of the photon group structures in use at that time as subsets. This has made it much more detailed than needed for practical calculations of the comparatively smooth photon spectra found in actual systems. It is also lacking in the detail above 10 MeV needed to represent the emission of "primary" photons from capture reactions in systems with a large flux of 14-MeV neutrons.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	5.0000E+03	1.0000E+04	1.5000E+04	2.0000E+04	3.0000E+04
6	3.5000E+04	4.0000E+04	4.5000E+04	5.5000E+04	6.0000E+04
11	6.5000E+04	7.5000E+04	8.0000E+04	9.0000E+04	1.0000E+05
16	1.2000E+05	1.4000E+05	1.5000E+05	1.6000E+05	1.9000E+05
21	2.2000E+05	2.6000E+05	3.0000E+05	3.2500E+05	3.5000E+05
26	3.7500E+05	4.0000E+05	4.2500E+05	4.5000E+05	5.0000E+05
31	5.2500E+05	5.5000E+05	5.7500E+05	6.0000E+05	6.2500E+05
36	6.5000E+05	6.7500E+05	7.0000E+05	7.5000E+05	8.0000E+05
41	8.2500E+05	8.6500E+05	9.0000E+05	1.0000E+06	1.1250E+06
46	1.2000E+06	1.2500E+06	1.3300E+06	1.4200E+06	1.5000E+06
51	1.6000E+06	1.6600E+06	1.7500E+06	1.8750E+06	2.0000E+06
56	2.1660E+06	2.3330E+06	2.5000E+06	2.6660E+06	2.8330E+06
61	3.0000E+06	3.1660E+06	3.3330E+06	3.5000E+06	3.6500E+06
66	3.8000E+06	3.9000E+06	4.0000E+06	4.2000E+06	4.4000E+06
71	4.5000E+06	4.7000E+06	5.0000E+06	5.2000E+06	5.4000E+06
76	5.5000E+06	5.7500E+06	6.0000E+06	6.2500E+06	6.5000E+06
81	6.7500E+06	7.0000E+06	7.2500E+06	7.5000E+06	7.7500E+06
86	8.0000E+06	8.5000E+06	9.0000E+06	9.5000E+06	1.0000E+07
91	1.0600E+07	1.1000E+07	1.2000E+07	1.4000E+07	2.0000E+07

18. IGG=3...LANL 12-Group Structure

This is a comparatively coarse photon structure that has been used extensively at Los Alamos. It is part of the 42-group structure used in the CLAW-3 and

CLAW-4 coupled libraries[3] available from RSIC, and also in the MATXS1 and MATXS5 libraries used with TRANSX[4]. It is most suitable for systems containing a reasonable concentration of heavy atoms; strongly hydrogenous systems have a very soft photon flux that is not well represented by the single group below 100 keV. The 24-group option (see below) is better in this low range, and it also does a better job of calculating photon energy deposition because of more detail above 9 MeV.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0000E+04	1.0000E+05	5.0000E+05	1.0000E+06	2.0000E+06
6	3.0000E+06	4.0000E+06	5.0000E+06	6.0000E+06	7.0000E+06
11	8.0000E+06	9.0000E+06	2.0000E+07		

19. IGG=4...Steiner 21-Group Structure

This structure was used extensively in early fusion-reactor systems studies at the Oak Ridge National Laboratory and elsewhere.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0000E+04	1.0000E+05	2.0000E+05	4.0000E+05	1.0000E+06
6	1.5000E+06	2.0000E+06	2.5000E+06	3.0000E+06	3.5000E+06
11	4.0000E+06	4.5000E+06	5.0000E+06	5.5000E+06	6.0000E+06
16	6.5000E+06	7.0000E+06	7.5000E+06	8.0000E+06	1.0000E+07
21	1.2000E+07	1.4000E+07			

20. IGG=5...Straker 22-Group Structure

This structure has been used in air-transport and weapons effects calculations.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0000E+04	3.0000E+04	6.0000E+04	1.0000E+05	1.5000E+05
6	3.0000E+05	4.5000E+05	6.0000E+05	8.0000E+05	1.0000E+06
11	1.3300E+06	1.6600E+06	2.0000E+06	2.5000E+06	3.0000E+06
16	3.5000E+06	4.0000E+06	5.0000E+06	6.0000E+06	7.0000E+06
21	8.0000E+06	1.0000E+07	1.4000E+07		

21. IGG=6...LANL 48-Group Structure

This is intended to be a comprehensive photon structure that includes all the detail needed for practical photon-transport calculations. It includes enough detail at low energies to represent systems dominated by light elements, and enough detail at high energies to give accurate energy-deposition results, even in systems with many primary photons from 14 MeV capture events. In addition, several groups are provided for "signatures" of fissionable materials needed for nondestructive assays in safeguards work. The only applications that require more groups would be those that depend on the observation of particular discrete gamma rays, such as quantitative analyses. The 12- and 24-group structures are both subsets of this 48-group set.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0000E+03	1.0000E+04	2.0000E+04	3.0000E+04	4.5000E+04
6	6.0000E+04	8.0000E+04	1.0000E+05	1.5000E+05	2.0000E+05
11	3.0000E+05	4.0000E+05	4.5000E+05	5.0000E+05	5.2500E+05
16	6.0000E+05	7.0000E+05	8.0000E+05	9.0000E+05	1.0000E+06
21	1.1250E+06	1.2000E+06	1.3300E+06	1.5000E+06	1.6600E+06
26	1.8750E+06	2.0000E+06	2.3330E+06	2.5000E+06	2.6660E+06
31	3.0000E+06	3.5000E+06	4.0000E+06	4.5000E+06	5.0000E+06
36	5.5000E+06	6.0000E+06	6.5000E+06	7.0000E+06	7.5000E+06
41	8.0000E+06	9.0000E+06	1.0000E+07	1.2000E+07	1.4000E+07
46	1.7000E+07	2.0000E+07	3.0000E+07	5.0000E+07	

22. IGG=7...LANL 24-Group Structure

This structure is intended for practical calculations of photon transport in fission and fusion systems. It includes enough detail at low energies for normal shield compositions, and it gives reasonably good results for photon energy deposition in fusion blanket calculations. It has been used with the 80-group neutron structure in the MATXS6 library used with the TRANSX[4] code. It should be used in preference to the 12-group structure when practical.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0000E+04	3.0000E+04	6.0000E+04	1.0000E+05	2.0000E+05
6	3.0000E+05	5.0000E+05	5.2500E+05	7.5000E+05	1.0000E+06
11	1.3300E+06	1.6600E+06	2.0000E+06	2.5000E+06	3.0000E+06
16	4.0000E+06	5.0000E+06	6.0000E+06	7.0000E+06	8.0000E+06
21	9.0000E+06	1.0000E+07	1.2000E+07	1.7000E+07	3.0000E+07

23. IGG=8...VITAMIN-C 36-Group Structure

This structure was designed for use with the VITAMIN-C library[2], which was intended for fusion and fast-reactor shielding applications. It is a combination of the 21-group and 22-group structures described above, with an extra group added to transport the annihilation radiation at 511 keV, and an extra bound added at 70 keV. The main problem with this structure is the upper limit of 14 MeV (see IGG=9 below).

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0000E+04	2.0000E+04	3.0000E+04	4.5000E+04	6.0000E+04
6	7.0000E+04	1.0000E+05	1.5000E+05	2.0000E+05	3.0000E+05
11	4.0000E+05	4.5000E+05	5.1000E+05	5.1200E+05	6.0000E+05
16	7.0000E+05	8.0000E+05	1.0000E+06	1.3300E+06	1.5000E+06
21	1.6600E+06	2.0000E+06	2.5000E+06	3.0000E+06	3.5000E+06
26	4.0000E+06	4.5000E+06	5.0000E+06	5.5000E+06	6.0000E+06
31	6.5000E+06	7.0000E+06	7.5000E+06	8.0000E+06	1.0000E+07
36	1.2000E+07	1.4000E+07			

24. IGG=9...VITAMIN-E 38-Group Structure

This is a slightly extended version of the previous group structure that was selected for use with the VITAMIN-E library[8]. There were only two changes made: a new group was added above 14 MeV, and a new bound was inserted at 75 keV. The structure given in the reference is not exactly correct; contact the Radiation Shielding Information Center (RSIC) at Oak Ridge for a full description of the VITAMIN-E library as issued.

GROUP INDEX	GROUP LOWER BOUNDARIES (EV)				
N	EL(N+0)	EL(N+1)	EL(N+2)	EL(N+3)	EL(N+4)
1	1.0000E+04	2.0000E+04	3.0000E+04	4.5000E+04	6.0000E+04
6	7.0000E+04	7.5000E+04	1.0000E+05	1.5000E+05	2.0000E+05
11	3.0000E+05	4.0000E+05	4.5000E+05	5.1000E+05	5.1200E+05
16	6.0000E+05	7.0000E+05	8.0000E+05	1.0000E+06	1.3300E+06
21	1.5000E+06	1.6600E+06	2.0000E+06	2.5000E+06	3.0000E+06
26	3.5000E+06	4.0000E+06	4.5000E+06	5.0000E+06	5.5000E+06
31	6.0000E+06	6.5000E+06	7.0000E+06	7.5000E+06	8.0000E+06
36	1.0000E+07	1.2000E+07	1.4000E+07	2.0000E+07	

25. References

1. C. R. Weisbin and R. J. LaBauve, "Specifications of a Generally Useful Multigroup Structure for Neutron Transport," Los Alamos Scientfic Laboratory report LA-5277-MS (May 1973).

2. R. W. Roussin, C. R. Weisbin, J. E. White, N. M. Greene, R. Q. Wright, and J. B. Wright, "The CTR Processed Multigroup Cross Section Library for Neutronics Studies," Oak Ridge National Laboratory report ORNL/RSIC-37 (1978).

3. R. J. Barrett and R. E. MacFarlane, "Coupled Neutron and Photon Cross Sections of Transport Calculations," Los Alamos Scientfic Laboratory report LA-7808-MS (April 1979).

4. R. E. MacFarlane, "TRANSX-CTR: A Code for Interfacing MATXS Cross-Section Libraries to Nuclear Transport Codes for Fusion Systems Analysis," Los Alamos National Laboratory report LA-9863-MS (February 1984).

5. R. B. Kidman and R. E. MacFarlane, "LIB-IV, A Library of Group Constants for Nuclear Reactor Calculations," Los Alamos Scientific Laboratory report LA-6260-MS (March 1976).

6. R. E. MacFarlane, "ENDF/B-V Cross-Section Library for Reactor Cell Analysis," Electric Power Research Institute report EPRI NP-3418 (February 1984).

7. D. E. Cullen, "Report on the IAEA Cross Section Processing Code Verification Project," International Atomic Energy Agency Nuclear Data Section report INDC(NDS)-170/NI (May 1985).

8. C. R. Weisbin, R. W. Roussin, J. J. Wagschal, J. E. White, and R. Q. Wright, "VITAMIN-E: An ENDF/B-V Multigroup Cross-Section Library for LMFBR Core and Shield, LWR Shield, Dosimetry, and Fusion Blanket Technology," Oak Ridge National Laboratory report ORNL-5505 (ENDF-274) (February 1979).

Appendix B. WEIGHT FUNCTIONS

1. Neutron Weight Functions in GROUPR

Most of the weight functions available in GROUPR are shown in Figs. 1 and 2. As discussed in the GROUPR chapter of this manual, the accuracy of a set of multigroup constants is determined by the group structure and the weight function. If only a few groups are used, it is necessary to have a weight function that is similar to the flux in the kinds of nuclear systems to be calculated with that particular group structure. If the flux is not well enough known, or if it varies strongly between different regions of the device, it is necessary to use more groups. The weight functions provided in the code are representative of a few typical nuclear systems, as described below. For systems with unusual spectra, or for very high accuracy calculations, the user should consider entering custom weight functions.

IWT=2 The weight function is constant (not shown). This option is usually chosen for very fine group structures such as the 620-group or 640-group dosimetry structures.

IWT=3 The weight function is proportional to $1/E$ (not shown). The slowing-down of neutrons in water gives a $1/E$ flux from about 1 eV up to 100 keV, or so. This weight function is traditionally used for calculating resonance integrals, but it is not useful at the lower and higher energies needed for a full set of transport constants. The graph of this function is a flat line on a flux-per-unit-lethargy plot, such as the one in Fig. 1.

IWT=4 This weight function combines a thermal Maxwellian at low energies, a $1/E$ function at intermediate energies, and a fission spectrum at high energies to obtain a function appropriate for several different applications. The temperatures of the Maxwellian and fission parts and the energies where the components join can be chosen by the user. Therefore, option 4 can be used to produce typical thermal reactor weight functions like those shown in Figs. 1 and 2, a pure fission spectrum for calculating some kinds of dosimetry cross sections, or a pure thermal spectrum for getting effective thermal average cross sections. The function for IWT=4 shown in the figure was produced using a thermal temperature of 0.025 eV joined to $1/E$ at 0.1 eV, and a fission temperature of 1.40 MeV joined to $1/E$ at 820.3 keV.

IWT=5 This is a mid-life PWR (pressurized water reactor) flux spectrum with a fusion peak added (see below for a discussion of the fusion peak used). Note the peaks and dips resulting from oxygen resonances and windows at high

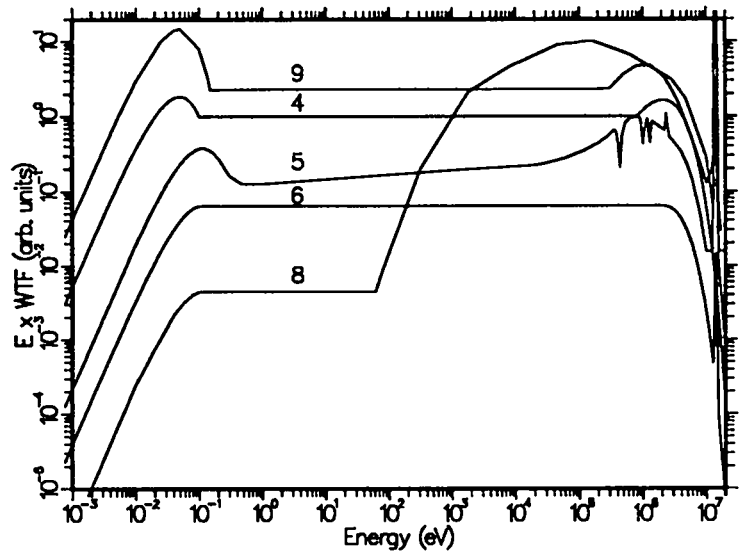


Figure 1: Built-in neutron weight functions of GROUPR on a logarithmic flux-per-unit-lethargy plot that emphasizes the low energy range.

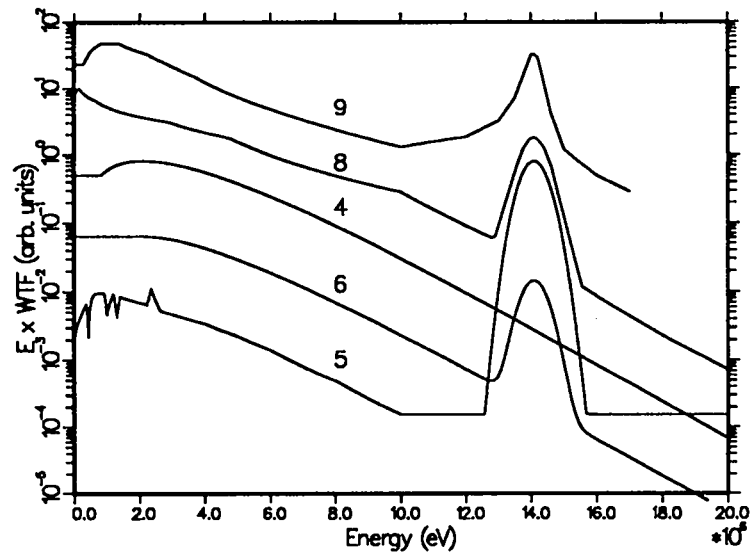


Figure 2: Built-in neutron weight functions on a linear plot that emphasizes the high-energy range.

energies, and the hardening apparent in the epithermal region. The thermal part of the function is also hardened with respect to a simple Maxwellian shape. The dips that should show up in the eV range due to resonances and U-238 have been removed to allow the self-shielding method to work without risk of counting the shielding effects twice. This weight function has been used for several libraries[1] prepared for the Electric Power Research Institute (EPRI), and for the MATXS7 library used with the TRANSX code[2].

IWT=6 This function is similar to option 4, but the breakpoints were chosen to keep the curves continuous. The thermal Maxwellian is calculated for 300 K. In this case, the fusion peak (see below) was added to the high-energy tail of the fission spectrum for smoothness.

IWT=7 This option is reserved for future use.

IWT=8 This function is intended for cross sections libraries to be used for fast reactor analysis (typically, fast breeder designs), but is also useful for fusion-blanket problems. It has a fusion peak at high energies, followed by a fission spectrum, a slowing-down spectrum typical of a fast reactor, and a thermal tail. The tail is provided to help give reasonable results in shields far from the core; its characteristic temperature is 300 K. Note the sharp drop in the flux as energy decreases from 19 keV. This region is important for U-238 absorption, and this drop-off helps to give good group constants for fast reactors. Of course, it would be entirely wrong for a thermal reactor.

IWT=9 This is another typical thermal+$1/E$+fission+fusion function that has been used for many libraries at the Los Alamos National Laboratory in the 30-group structure. The CLAW-3 and CLAW-4 libraries are available from the Radiation Shielding Information Center at the Oak Ridge National Laboratory.

IWT=10 This is the same as the CLAW weight function (IWT=9), but the shape of the thermal part is automatically recalculated to follow a Maxwellian law for temperature T.

IWT=11 This is the weight function used in the VITAMIN-E library. It has the following segments: a .0253-eV thermal Maxwellian below 0.414 eV, a $1/E$ law from .414 eV to 2.12 MeV, a 1.415-MeV fission spectrum from 2.12 to 10 MeV, another $1/E$ section from 10 to 12.52 MeV, a fusion peak (25 keV width) between 12.52 and 15.68 MeV, and a final $1/E$ section for all higher energies. The shape of the fusion peak is almost identical to IWT=5 (see Fig. B-2). The low-energy part of this weight function is not shown.

2. Shape of the Fusion Peak

Because of the finite width of the distribution of ion energies in a D-T fusion plasma, the emitted 14-MeV neutrons clearly will not have a delta-function energy spectrum. In fact, owing to the presence of a cross-product term in the kinematic relations, the typical ion-energy spread of the few tens of kilovolts is magnified into a neutron-energy spread of around 1 MeV.

For an assumed isotropic Maxwellian plasma, the neutron peak shape (for example, see the review article by Lehner[3]) is given by

$$S(E) = C \int_0^\infty \exp\{-b[v^2 + v_0(g)^2] - cg^2\} \sinh(2bvv_0) \frac{g^3\sigma(g)}{v_0(g)} dg \ . \qquad (1)$$

Here $S(E)$ is the number of neutrons emitted with laboratory energy between E and $E+dE$, C is a normalization factor, v is the laboratory velocity corresponding to energy E, and v_0 is the velocity of the neutron in the CM system. Both v_0 and the fusion cross section σ are determined by the relative velocity g of the reacting ions, the integration variable in Eq. (1). The coefficient b is equal to $M/2kT$, where M is the total mass of the reacting ions and kT is the plasma temperature. Similarly, c is $\mu/2kT$, where μ is the reduced mass of the ion system.

The only approximation involved in Lehner's derivation of Eq. (1) is that all particles may be treated nonrelativistically. At 14 MeV, the relativistic factor

$$\gamma = \frac{1}{\sqrt{1 - v^2/c^2}} \ , \qquad (2)$$

is very close to unity (1.015), and it varies negligibly over the range of interest (say, 13 to 17 MeV). It is sufficient then to invoke relativistic mechanics in defining the location of the 14-MeV peak but not in discussing the shape. This effect moves the peak toward lower neutron energies, but only by about 20 keV.

Although the expression for $S(E)$ in Eq. (1) is accurate, it has the disadvantage of requiring a numerical integration at each point E in the energy spectrum. For this reason, we consider what simplifications can be made without serious loss of accuracy. In the energy range around the 14-MeV peak, the product $2bvv_0$ in Eq. (1) has a numerical value of about 6000. Thus, the hyperbolic sine can obviously be replaced by just the positive exponential term. If we make this change in Eq. (1), we can write the following, still nearly exact, expression for the neutron spectrum:

$$S_1(E) = C_1 \int_0^\infty \exp[-b(v - v_0)^2] \, P(v_0) \, dv_0 \ . \qquad (3)$$

Here we also have inverted the function $v_0(g)$ and changed the integration variable. The spectrum then is a linear superposition of velocity exponentials with slightly different peak locations. For normal plasma temperatures, the velocity distribution $P(v_0)$ is very narrow, since the expression for v_0 is dominated by the nuclear Q-value (17.586 MeV) rather than the contribution from the ion kinetic energy (typically around 50 keV). Thus, it seems reasonable to approximate $P(v_0)$ as a delta function

$$P(v_0) \approx \delta(v_0 - v_p) \; . \tag{4}$$

This gives a second approximate form,

$$S_2(E) = C_2 \exp[-b(v - v_p)^2] \; , \tag{5}$$

where v_p has the obvious meaning of the laboratory neutron velocity at the center of the peak. We shall refer to this as the velocity exponential form of the neutron energy spectrum. An expression essentially identical to Eq. (5) was given in an early paper by Nagle and coworkers[4].

In order to examine the accuracy of the velocity exponential form, we have calculated $S(E)$ from Eq. (1) and $S_2(E)$ from Eq. (5) at 20 keV, a typical plasma temperature in current fusion-reactor concepts. In performing the numerical integration over g in Eq. (1), we used numerical values for the D-T fusion cross section taken from the compilation by Jarmie and Seagrave[5]. In evaluating $S_2(E)$ using Eq. (5), a value of v_p was chosen so as to force agreement between $S_2(E)$ and $S(E)$ at 17 MeV. As discussed by Muir[6], the overall agreement is remarkably good, the maximum error over the range from 13.5 to 17 MeV being about 2%. The value of v_p thus derived corresponds to a peak-center energy E_p of 14.07 MeV. This value includes the small ($\sim$ 20 keV) relativistic correction mentioned above.

If we approximate the mass of the D+T system as 5 times the neutron mass, then we obtain the recommended peak shape

$$S_{\rm rec}(E) = \exp[-\frac{5}{kT}(\sqrt{E} - \sqrt{E_p})^2] \; , \tag{6}$$

where E_p=14.07 MeV. The functional form in Eq. (6) was used to calculate the fusion peak shapes appearing in two of the data statements in `GENWTF` (namely, those utilized for IWT=5 and 8) and is also used explicitly in `GETWTF` to calculate analytically the weight function for IWT=6. In all three cases, kT=25 keV is used as an average or typical fusion-reactor plasma temperature. See Fig. 2 for a graphical display of the resulting weight functions in the 14-MeV region.

3. Photon Weight Functions in GAMINR

The main features of a typical photon spectrum are a $1/E$ region at intermediate energies, a drop-off at low energies caused by photoelectric absorption, and a shoulder at high energies corresponding to the maximum Q-value for capture. The relative size and importance of these features depends on the composition of the system. With light materials like water and concrete, the $1/E$ portion is very obvious and extends to low energies. As heavy materials are added, the low-energy range is suppressed. The highest-energy photons in a system are normally primary photons from capture, with maximum energies of about $Q + (A/A+1)E$, where E is the neutron energy and A is the atomic weight ratio to the neutron for the absorber. Thus, the shape of the high-energy part of the photon flux may be different for fusion and fission systems, with energies above 10 MeV being important for fusion.

IWT=2 The weight function is constant (not shown). This option is most useful for fine-group libraries, or for benchmark comparisons of different codes where simplicity is useful.

IWT=3 The weight function is basically $1/E$, however low- and high-energy roll-offs are included to keep the first and last group from being unrealistically biased by unimportant regions of the cross-section energy range. See Fig. 3.

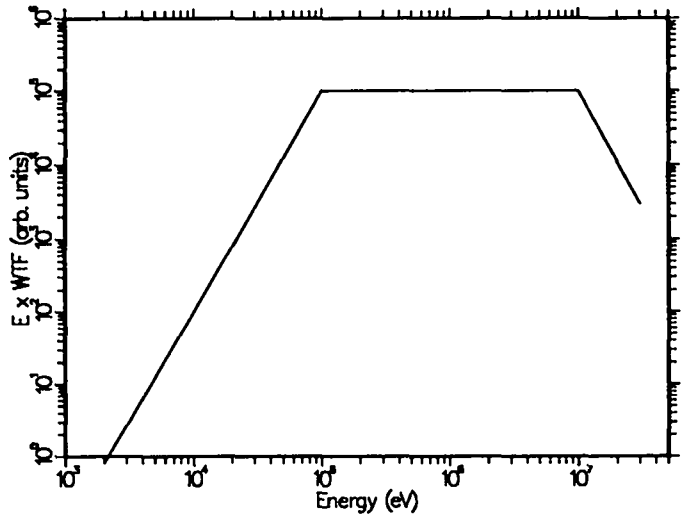


Figure 3: Built-in photon weight function of GAMINR showing the high- and low-energy roll-offs.

4. References

1. R. E. MacFarlane, "ENDF/B-V Cross-Section Library for Reactor Cell Analysis," Electric Power Research Institute report EPRI NP-3418 (February 1984).

2. R. E. MacFarlane, "TRANSX-CTR: A Code for Interfacing MATXS Cross-Section Libraries to Nuclear Transport Codes for Fusion Systems Analysis," Los Alamos National Laboratory report LA-9863-MS (February 1984).

3. G. Lehner, "Reaction Rates and Energy Spectra for Nuclear Reactions in High Energy Plasmas," *Z. Physik* **232**, 174 (1970).

4. D. E. Nagle, W. E. Quinn, W. B. Riesenfeld, and W. Leland, "Ion Temperature in Scylla, as Determined from the Reaction D(d,p)T," *Phys. Rev. Letters* **3**, 318 (1959).

5. Nelson Jarmie and John D. Seagrave (Eds.), "Charged Particle Cross Sections," Los Alamos Scientific Laboratory report LA-2014 (1956).

6. D. W. Muir, "Sensitivity of Neutron Multigroup Cross Sections to Thermal Broadening of the Fusion Peak," Proc. 1st Topical Meeting on the Technology of Controlled Nuclear Fusion, San Diego, Calif., April 1974, CONF-740402-P2, p.166 (1975).

Appendix C. TEST PROBLEMS

The NJOY test problems have two purposes: first, they provide readily available examples for the user on how to run the code, and second, they provide Quality Assurance (QA) verification that the code has been installed correctly on the user's computer system. The current test problem suite is inadequate for the second purpose, and we hope to expand it significantly.

1. Test Problem 1

This problem demonstrates how to prepare data for natural carbon as given on the ENDF/B-V "Standards Tape." This evaluation is freely available without restrictions (which is not true of all of ENDF/B-V).

The input cards for NJOY are listed below in the form of a UNIX shell script. We normally run this script in a subdirectory called TEST, and the first few cards copy the ENDF general-purpose and thermal data from their normal locations in the next higher directory into the test subdirectory. Note that the data files are assigned the local names TAPE20 and TAPE26. The CAT line starts a "here" file, which continues down to the EOF line near the end of the input. The NJOY code is then run using this new input file, and the output file and PENDF file are saved in the names OUT1 and PEND1 for later comparisons with previous runs.

The following input file is given using uppercase characters. On many systems, lowercase will be used instead. The user will have to determine if an uppercase or lowercase method was used when NJOY was installed.

```
ECHO 'GETTING ENDF TAPE 511'
CP ../T511 TAPE20
ECHO 'GETTING THERMAL TAPE 322'
CP ../T322 TAPE26
ECHO 'RUNNING NJOY'
CAT>INPUT <<EOF
 0
 5
 *MODER*
 20 -21
 *RECONR*
 -21 -22
 *PENDF TAPE FOR C-NAT FROM ENDF/B TAPE 511*/
 1306 3/
 .005 0. 7/
 *6-C-NAT FROM TAPE 511*/
 *PROCESSED BY THE NJOY NUCLEAR DATA PROCESSING SYSTEM*/
 *SEE ORIGINAL ENDF/B-V TAPE FOR DETAILS OF EVALUATION*/
 0/
```

```
*BROADR*
-22 -23
1306 1/
.005/
300.
0/
*HEATR*
-21 -23 -22
1306 1/
444
*THERMR*
26 -22 -24
1065 1306 8 1 1 0 1 221 0
300.
.05 1.2
*THERMR*
26 -24 -23
1065 1306 8 1 4 1 1 229 0
300.
.05 1.2
*GROUPR*
-21 -23 0 -24
1306 3 3 3 3 1 1 1
*CARBON IN GRAPHITE*/
300
1.E10
3 1 *TOTAL*/
3 2 *ELASTIC*/
3 4 *INELASTIC*/
3 51 *DISCRETE INELASTIC*/
3 -68 *CONTINUED*/
3 91 *CONTINUUM INELASTIC*/
3 102 *N,G*/
3 103 *(N,P)*/
3 104 *(N,D)*/
3 107 *(N,A)*/
3 203 *TOTAL H PRODUCTION*/
3 204 *TOTAL H2 PRODUCTION*/
3 207 *TOTAL HE4 PRODUCTION*/
3 221 *FREE THERMAL SCATTERING*/
3 229 *GRAPHITE INELASTIC THERMAL SCATTERING*/
3 230 *GRAPHITE ELASTIC THERMAL SCATTERING*/
3 251 *MUBAR*/
3 252 *XI*/
3 253 *GAMMA*/
3 301 *TOTAL HEAT PRODUCTION*/
3 444 *TOTAL DAMAGE ENERGY PRODUCTION*/
6 2 *ELASTIC*/
6 51 *DISCRETE INELASTIC*/
6 -68 *CONTINUED*/
6 91 *CONTINUUM INELASTIC*/
6 221 *FREE THERMAL SCATTERING*/
6 229 *GRAPHITE INELASTIC THERMAL SCATTERING*/
```

```
 6 230 *GRAPHITE ELASTIC THERMAL SCATTERING*/
 17 51 *INELASTIC GAMMA PRODUCTION*/
 16 102 *CAPTURE GAMMA PRODUCTION*/
 0/
 0/
 *MODER*
 -23 25
 *STOP*
EOF
../XNJOY<INPUT
ECHO 'SAVING OUTPUT FILES'
CP OUTPUT OUT1
CP TAPE25 PEND1
```

The first step is to run the MODER module to convert the ASCII ENDF file to binary mode. Using binary mode will often cut the cost of running NJOY jobs by a factor of two. RECONR is then used to linearize and unionize the cross sections (no resonance reconstruction is needed for carbon). A tolerance of 0.5% was requested for this linearization, and 7 significant figures will be used for all energy grid values. The BROADR module is used to Doppler-broaden the carbon cross sections to 300 K. It is recommended that the same thinning and reconstruction tolerances be used in BROADR as are used in RECONR.

This NJOY run supplements the original ENDF data with computed values for heating, radiation damage, and thermal scattering. The call to the HEATR module requests MT444 to get the damage cross section; heating (MT301) is automatically provided. The first of the two THERMR runs generates thermal scattering data for a carbon free gas at 300 K (MT221). The second THERMR run generates data for graphite (MT229). The "4" on the input card directs the module to use the $S(\alpha, \beta)$ data from MT1065 on TAPE26 in order to compute the inelastic scattering cross section and scattering matrix, and the following "1" directs it to compute the coherent elastic scattering cross section using built-in parameters for graphite.

When the second THERMR run has finished, TAPE23 contains the complete PENDF tape needed by GROUPR. Multigroup neutron reaction and photon production cross sections are computed using the Los Alamos 30-group structure for neutrons, the Los Alamos 12-group structure for photons, and the CLAW weight function. The scattering order is P_3. Note that the long-input form is used to specify the list of reactions to be processed. Most users now prefer the automatic input option. The user has to carefully look at the reactions available on the ENDF tape and to consider the additional reactions added by HEATR and THERMR. Note especially the inclusion of several gas production reactions from the ENDF

tape, the thermal reactions MT221 and MT229 from the PENDF tape, and MT301 and MT444 as generated by HEATR from the PENDF tape. The output is easy to read, but remember that groups are given in the order of increasing energy.

The thermal scattering reactions must also be requested in the scattering matrix section (MFD=6). MT230 was automatically generated by THERMR at MT229+1 when coherent elastic scattering for graphite was requested.

Photon production matrices are requested in the lines with **MFD=17** and **MFD=16**. Actually, the use of "17" to denote data given in MF13 on the ENDF tape is no longer required by GROUPR.

As a last step, this problem runs MODER once more to convert the binary PENDF tape to ASCII mode.

2. Test Problem 2

The second problem supplements the first by adding resonance reconstruction and output formatting. The ENDF/B-IV material Pu-238 (MAT1050) was originally chosen for this problem because it was freely available and its execution time was fairly small.

```
ECHO 'GETTING ENDF TAPE 404'
CP ../T404 TAPE20
ECHO 'RUNNING NJOY'
CAT>INPUT <<EOF
 0
 4
 *MODER*
 20 -21
 *RECONR*
 -21 -22
 *PENDF TAPE FOR PU-238 FROM ENDF/B-IV TAPE 404*/
 1050 3/
 .005 0. 7/
 *94-PU-238 FROM ENDF/B TAPE T404*/
 *PROCESSED BY THE NJOY NUCLEAR DATA PROCESSING SYSTEM*/
 *SEE ORIGINAL ENDF/B-IV TAPE FOR DETAILS OF EVALUATION*/
 0/
 *BROADR*
 -22 -23
 1050 3 0 1 0 /
 .005/
 300. 900. 2100.
 0/
 *UNRESR*
 -21 -23 -22
 1050 3 7 1
 300 900 2100
```

```
1.E10 1.E5 1.E4 1000. 100. 10. 1
0/
*GROUPR*
-21 -22 0 -25
1050 5 0 4 3 3 7 1
*94-PU-238*/
300. 900. 2100.
1.E10 1.E5 1.E4 1000. 100. 10. 1
.1 0.025 0.8208E06 1.4E06
3 1 *TOTAL*/
3 2 *ELASTIC*/
3 16 *N2N*/
3 17 *N3N*/
3 18 *FISSION*/
3 102 *CAPTURE*/
3 251 *MUBAR*/
3 252 *XI*/
3 253 *GAMMA*/
3 259 *1/V*/
6 2 *ELASTIC*/
6 16 *N2N*/
6 17 *N,3N*/
6 18 *FISSION*/
6 51 *DISCRETE INELASTIC*/
6 -59 *CONTINUED*/
6 91 *CONTINUUM INELASTIC*/
0/
3 1 *TOTAL*/
3 2 *ELASTIC*/
3 18 *FISSION*/
3 102 *CAPTURE*/
6 2 *ELASTIC*/
0/
3 1 *TOTAL*/
3 2 *ELASTIC*/
3 18 *FISSION*/
3 102 *CAPTURE*/
6 2 *ELASTIC*/
0/
0/
*CCCCR*
-25 21 22 0
1 1 *T2LANL NJOY*/
*CCCCR TESTS FOR NJOY87*/
50 0 1 4 1
*PU238* *PU238* *ENDFB4* * 1050 * 1050 10.89
1 0 50 -1
0 2.3821E02 3.3003E-11 1.7461E-12 0. 1.E10 0.0
3 6
300 900 2100
1.E5 1.E4 1000. 100. 10. 1
*STOP*
EOF
```

```
../XNJOY<INPUT
ECHO 'SAVING OUTPUT FILES'
CP OUTPUT OUT2
```

The RECONR input is identical to the one in Problem 1, but reading the listing file will demonstrate that RECONR added about 2700 points to the original grid in the resonance region. After a little thinning, it ended up with about 2800 resonance energy points at 0 K for .5 % reconstruction. The data from RECONR was then passed to BROADR for preparation of cross sections at 300, 900, and 2100 K. An examination of the output listing shows that some thinning was possible because of the smoothing effect of Doppler broadening; the zero degree grid of 3241 points changed to 2341 points at 2100 degrees.

One of the new features of this run is the call to UNRESR. The user will normally notice that unresolved-resonance parameters are available for a material by reading the introductory information on the ENDF tape, but UNRESR will return gracefully if no unresolved data are present in the evaluation. The long output from UNRESR gives a table of the self-shielded cross sections by reaction (vertical) and sigma-zero value (horizontal) for each point of the unresolved-range energy grid. The reactions are flux-weighted total, elastic, fission, capture, and current-weighted total reading from the top down. Heating, damage, and thermal cross sections were omitted for this particular problem. Therefore, the final PENDF tape is on TAPE22, the output from UNRESR.

The GROUPR run shown here adds several new features over the one given for problem 1. First, multiple temperatures and sigma-zero values are specified in order to get tables of self-shielded cross sections for the total, elastic, fission, and capture reactions. It is desirable to use the same sigma-zero grid in GROUPR that was used in UNRESR, although GROUPR will attempt to interpolate if the grids are different. Note that a section of GROUPR input is given for 300 K with complete coverage of all the reactions, and additional sections are given for the two higher temperatures with only the self-shielded reactions included. An examination of the output listing will show that self-shielded cross sections are given for the low-energy reactions (total, elastic, fission, capture), with group index reading down and sigma-zero value reading across.

Another new feature of this input is the computation of the fission matrix (MFD=6, MTD=18). More complicated input may be necessary for other materials. Examination of the output listing will show that GROUPR discovered that the fission spectrum shape was constant over the entire energy range; therefore, it only put out a spectrum and a fission-neutron production cross section.

When the GROUPR run is complete, the final GENDF tape will be found on TAPE25. This file is used as input to the CCCCR module to produce ISOTXS and BRKOXS files for Pu-238.

3. Test Problem 3

This problem demonstrates the use of GAMINR to prepare photon interaction (or photoatomic) cross sections. It also demonstrates DTFR, including plotting, and MATXSR. For the sake of continuity, this test problem uses a rather old version of the photon interaction files called DLC7E. The MF23 and MF27 parts of these data are written on two separate files. Later libraries have everything on a single tape.

```
ECHO 'GETTING PHOTOATOMIC TAPE GAM23'
CP ../GAM23 TAPE30
ECHO 'GETTING PHOTOATOMIC TAPE GAM27'
CP ../GAM27 TAPE32
ECHO 'RUNNING NJOY'
CAT>INPUT <<EOF
 0
 4
 *RECONR*
 30 31
 *PENDF TAPE FOR PHOTON INTERACTION CROSS SECTIONS FROM DLC7E*/
 1 1 0
 .001/
 *1-HYDROGEN*/
 92 1 0
 .001/
 *92-URANIUM*/
 0/
 *GAMINR*
 32 31 0 33
 1 3 3 4 1
 *12 GROUP PHOTON INTERACTION LIBRARY*/
 -1 0/
 92
 -1 0/
 0/
 *DTFR*
 33 34 31
 1 1 0
 5 12 4 5 16 1 0
 *PHEAT*
 1 621 1
 0/
 *H* 1 1 0./
 *U* 92 1 0./
 0/
 *MATXSR*
```

```
 0 33 35/
 1 *T2LANL NJOY*/
 1 1 1 2
 *12-GROUP PHOTON INTERACTION LIBRARY*/
 *G*
 12
 *GSCAT*/
 1
 1
 *H* 1 1
 *U* 92 92
 *STOP*
EOF
../XNJOY<INPUT
ECHO 'SAVING OUTPUT AND PLOT FILES'
CP OUTPUT OUT3
CP PLOT PLOT3
```

This run starts with an application of the RECONR module to linearize and unionize the File 23 cross sections. Of course, there is no resonance reconstruction required here. This RECONR run demonstrates the use of a material loop; both hydrogen and uranium are processed in one run. GAMINR is then used to prepare the multigroup photon interaction cross sections, including heating; this run also uses a multimaterial loop. The Los Alamos 12-group photon structure is used. The GENDF tape (TAPE33) is processed into two completely different library formats. First, DTFR is called. This is a rather old output module, but it is still useful for some purposes. For one thing, it automatically produces plots of the multigroup data overlayed with the PENDF data. Examples of the plots are given in the DTFR chapter of this manual. The MATXS output is more useful, because it can be used in a much more flexible way by the TRANSX code.

4. Test Problem 4

This problem illustrates several aspects of the calculation of covariances (uncertainties) of multigroup data using ERRORR. The first ERRORR problem produces, on unit 23, a file of multigroup cross section covariances (MFCOV=23) for all reactions present (IREAD=0) in File 33 for ^{235}U. The group structure employed is identical to the energy grid selected by the evaluator (IGN=19); however, no covariances are produced for cross sections below 1 eV or above 1 keV. The second ERRORR run adds to the above results a second data set containing multigroup $\overline{\nu}$ covariances. Note that the use of MFCOV=31 dictates that a GENDF file be produced (NGOUT=24) prior to the start of ERRORR. In the $\overline{\nu}$ run, a user-defined group structure is employed (IGN=1).

```
ECHO 'GETTING ENDF TAPE 511'
```

```
CP ../T511 TAPE20
ECHO 'RUNNING NJOY...'
CAT>INPUT <<EOF
 0
 5
 *MODER*
 20 -21
 *RECONR*
 -21 -22
 *U-235 10% PENDF FOR ERRORR TEST PROBLEM FROM T511*/
 1395 0 0
 .10/
 0/
 *ERRORR*
 -21 -22 0 23 0/
 1395 19 1 1
 3 0 0
 0 33
 1
 1.E0 1.E3
 *GROUPR*
 -21 -22 0 24
 1395 3 0 3 0 1 1 1
 *U-235 MULTIGROUP NUBAR CALCULATION*/
 0.
 1.E10
 3 452 *TOTAL NUBAR*/
 0/
 0/
 *ERRORR*
 -21 0 24 25 23/
 1395 1 1 1
 0 31
 7
 1.E0 1.E1 1.E2 1.E3 1.E4 1.E5 1.E6 1.E7
 *STOP*
EOF
../XNJOY<INPUT
ECHO 'SAVING OUTPUT'
CP OUTPUT OUT4
```

5. Test Problem 5

This short problem produces multigroup covariances for ^{12}C, again using IGN=19, but here all cross sections from 1E-5 to 2E7 are treated. The COVR module reads the binary output file from ERRORR and produces publication-quality plots of all reactions for which covariance data exist. This simple problem setup could be used with only a few simple changes for any nuclide, in order to take a quick look at the contents of the covariance files. For applications where the resonance region is of interest, it is necessary to replace the second MODER step with a resonance

reconstruction step using RECONR, as in the previous example.

```
ECHO 'GETTING ENDF TAPE 511'
CP ../T511 TAPE30
ECHO 'RUNNING NJOY...'
CAT>INPUT <<EOF
 0
 5
 MODER
 30 -31
 MODER
 -31 -32
 ERRORR
 -31 -32 0 -33/
 1306 19 1/
 2 0 0
 0 33
 1
 1E-5 2E7/
 COVR
 -33/
 /
 /
 1306/
 STOP
EOF
../XNJOY<INPUT
ECHO 'SAVING OUTPUT AND PLOT'
CP OUTPUT OUT5
CP PLOT PLOT5
```

6. Test Problem 6

This test problem demonstrates and tests a number of different kinds of plots using data from ENDF/B-V Tape 511 (the "Standards Tape"). The graphs produced and a detailed discussion of the input cards will be found in the PLOTR chapter of this manual.

```
ECHO 'GETTING ENDF TAPE 511'
CP ../T511 TAPE30
ECHO 'RUNNING NJOY'
CAT>INPUT <<EOF
 0
 5
 PLOTR
 /
 1/
 *ENDF/B-V CARBON*/
 *T<OTAL> C<ROSS> S<ECTION>*/
 4/
 1E3 2E7/
```

```
/
.5 10/
/
5 30 1306 3 1/
/
1/
*ENDF/B-V CARBON*/
*<(N,#A<) WITH FAKE DATA*/
1 0 2 1 1.4E7 .32/
/
/
/
/
5 30 1306 3 107/
/
*ENDF/B-V MAT1306*/
2/
0/
-1 0/
*S<MITH> & S<MITH> 1914*/
0/
1.1E7 .08 .05 .05/
1.2E7 .10 .05 .05/
1.3E7 .09 .04 .04/
1.4E7 .08 .03 .03/
-1/
3/
0/
-1 2/
*B<LACK> & B<LUE> 2008*/
0/
1.15E7 .07 .02 0. .2E6 0./
1.25E7 .11 .02 0. .2E6 0./
1.35E7 .08 .015 0. .2E6 0./
1.45E7 .075 .01 0. .2E6 0./
-1/
1/
*ENDF/B-V CARBON*/
*E<LASTIC> MF4*/
-4/
/
/
/
/
5 30 1306 4 2/
/
1/
*ENDF/B-V L<I>-6*/
*<(N,2N)#A <NEUTRON DISTRIBUTION*/
-1/
/
/
0 12E6 2E6/
```

```
 /
 5 30 1303 5 24/
 /
 1/
 *ENDF/B-V L<I>-6*/
 *<(N,2N)#A <NEUTRON SPECTRA VS >E*/
 4 0 2 2/
 10. 2.E7/
 /
 1E-11 1E-6/
 *C<ROSS> S<ECTION> (<BARNS/E>V)*/
 5 30 1303 5 24 0. 12/
 /
 *10 M<E>V*/
 1E3 2E-11 1E2/
 2/
 5 30 1303 5 24 0. 16/
 /
 *14 M<E>V*/
 1E4 2E-10 2E3/
 3/
 5 30 1303 5 24 0. 20/
 /
 *20 M<E>V*/
 1E5 2E-9 4E4/
 99/
 STOP
EOF
../XNJOY<INPUT
ECHO 'SAVING PLOT FILE'
CP PLOT PLOT6
```

7. Test Problem 7

This test problem demonstrates and tests the preparation of ACE-format libraries for the MCNP continuous-energy Monte Carlo code. The material selected for processing was U-235 from ENDF/B-V. Since this material is given on the "Standards Tape," there are no restrictions on its distribution.

```
ECHO 'GETTING ENDF TAPE 511'
CP ../T511 TAPE20
ECHO 'RUNNING NJOY...'
CAT>INPUT <<EOF
 0
 5
 *MODER*
 20 -21
 *RECONR*
 -21 -22
 *PENDF TAPE FOR U-235 FROM ENDF/B-V TAPE 511* /
 1395 3 /
```

```
.005 /
*92-U-235 FROM ENDF/B-V TAPE 511 * /
*PROCESSED BY THE NJOY NUCLEAR DATA PROCESSING SYSTEM* /
*SEE ORIGINAL ENDF/B-V TAPE FOR DETAILS OF EVALUATION* /
0 /
*BROADR*
-22 -23
1395 1 0 1 0 /
.005 /
300.
0 /
*HEATR*
-21 -23 -24
1395 /
*GROUPR*
-21 -24 0 -25
1395 3 2 9 0 1 1 1 /
*U-235 FROM TAPE 511* /
300.
1.0E10
16 /
0 /
0 /
*ACER*
-21 -24 -25 26 27 /
1/
*NJOY TEST PROBLEM 7*/
1395 300. /
0.05 /
/
*STOP*
EOF
../XNJOY<INPUT
ECHO 'SAVING OUTPUT AND ACE FILES'
CP OUTPUT OUT7
CP TAPE26 ACE7
```

Index

*U.S. GOVERNMENT PRINTING OFFICE 1995-0-673-029/7062

Los Alamos
NATIONAL LABORATORY
Los Alamos, New Mexico 87545