LA-UR-25-26179

Approved for public release; distribution is unlimited.

Title:	MCNP Utilities: 2025 Update
Author(s):	Jones, Fred Burke
Intended for:	MCNP Symposium, 2025-07-07/2025-07-10 (Los Alamos, New Mexico, UNITED STATES)
Issued:	2025-06-27

Los Alamos
NATIONAL LABORATORY

MCNP Utilities: 2025 Update

Introducing a new version of “pstudy”

Fred Jones

2025 MCNP User Symposium
July 7-10, 2025

Agenda

- Background and motivation
- Status of the “Appendix E” tools
- A closer look at the new “pstudy”

Background and Motivation

Background and motivation

- Appendix “E” of the MCNP manual lists pre- and post-processing utilities that come with the MCNP code
  - Some are rarely used
  - Some depend upon deprecated functionality within the MCNP code
  - Others are still useful but may lack developer support
- Several of the Perl-based utilities stood out as good candidates for modernization

Status of the “Appendix E” Tools

Tool Overview

Tool	Use	Tech
dbrc_make_lib	Doppler broadening resonance correction lib. gen.	Fortran
ela	Event Log Analyzer: interrogate event logs	Perl
fit_otf	On-the-fly Doppler broadened data fitting	Fortran
gridconv	Convert tally data to graphics files	Fortran
makxsf	ACE library manipulation	Fortran
merge_mctal	Statistically merge MCTAL files	Perl
merge_meshtal	Statistically merge MESHTAL files	Perl
mcnp_pstudy	Parametric studies	Perl
simple_ace	Simple ACE file generation	Perl
um_convert	Abaqus to MCNPUM	Fortran
um_post_op	Operations on EEOUT files	Fortran
um_pre_op	Skeleton MCNP input file and checks	Fortran

The Case for Modernization

Tool	Update?	Note
`dbrc_make_lib`	No	Infrequent usage
`ela`	Maybe	Useful, but overlaps PTRAC capability
`fit_otf`	No	OTFDB has numerical stability issues
`gridconv`	No	Use the built-in plotter and/or HDF5 output
`makxsf`	No	Adequate for Type 1/2 ACE conversion
`merge_mctal`	Yes	Closely connected with `pstudy`
`merge_meshtal`	Yes	Current tooling supports deprecated formats
`mcnp_pstudy`	Yes	My main focus
`simple_ace`	No	Adequate for current benchmarking
`um_convert`	No	MCNPUM is deprecated
`um_post_op`	Yes	Capability moving to Python or MCNP itself
`um_pre_op`	Yes	Capability moving to Python or MCNP itself

`pstudy`'s Strengths and Weaknesses

- Strengths
  - Parametric studies are inherently useful
  - "Poor man's parallelism" is still useful for serial-only features
  - `pstudy` supports a modest amount of in-line math
    - Useful for unit conversions, trigonometry, repetition
    - Supports a "literate programming" style
    - Reduces the needs for secondary tooling or files
- Weaknesses
  - Perl-based
    - Introduces a dependency on the Perl interpreter
    - Not a favorite language among the development team
    - Slow for large input and output files
  - `pstudy` input templates are not themselves valid MCNP input files
  - Current implementation has poor Windows support

Serial Features of MCNP6.3.1

- Some MCNP features cannot run in parallel using threading
- MPI is an option

- `DBCN` event logging,
- CINDER for delayed neutrons and/or photons using `ACT`,
- LLNL photofission multiplicity,
- `FMULT` CGMF, FREYA, or LLNL fission multiplicity models,
- `SSR` surface source write,
- legacy (i.e., non-HDF5) `PTRAC`,
- `TMESH` tallies,
- `HISTP` file generation,
- model physics for missing cross section data, and
- any particle other than neutrons, photons, or electrons.

Ref: LA-UR-24-24602, §3.3.2.3

A Closer Look at the New “pstudy”

Requirements of a new `pstudy`

- Does what the old tool did, plus...
- Performs identically on all three major OSes
- Zero-hassle installation
- Performant on large files
- Seamless interaction with MCNP
- Remembers often-repeated command-line arguments

My proposed solution: A command-line tool, a “driver” for MCNP

Example Command-Line Usage: Perl Script vs. New CLI

Perl	`mcnp_pstudy` (‘`mp`‘)
`perl /path/to/mcnp_pstudy.pl -i file.inp -setup`	`mp setup`
`perl /path/to/mcnp_pstudy.pl -i file.inp -setup -run`	`mp run`
`perl /path/to/mcnp_pstudy.pl -i file.inp -collect`	`mp collect`

Note: The new tool will automatically detect when it needs to write new input files.

Example Command-Line Usage: MCNP Interaction

MCNP	`mcnp_pstudy ('mp')`	Operation
`mcnp6 i=file.inp i`	`mp check`	Input checks
`mcnp6 i=file.inp ip`	`mp plotg`	Plot geometry
`mcnp6 i=file.inp ixr`	`mp run`	Do particle transport
`mcnp6 r=runtpe.h5 z`	`mp plot`	Plot results

Note 1: The `mp` version of the commands runs on **all** cases.
Note 2: The "first" instantiated input file is selected for geometry plotting, but the user can opt for a different one.
Note 3: The "last" output file is automatically found for consecutive runs.

New Features

- Accept LANL “WORM” syntax
- Target k-effective search
- Automate consecutive runs of MCNP weight window generator
- Automate running NJOY to generate at-temperature libraries
- Generate input examples
- Whisper integration
- Standalone merge capability

Demonstration

Thank you for your attention.
Any questions?

Backup Slides

pstudy Directives

- c @@@ symbol = value
- c @@@ symbol = value_1 value_2 ...
- c @@@ symbol = normal n mean standard_deviation
- c @@@ symbol = lognormal n mean standard_deviation
- c @@@ symbol = uniform n lower_bound upper_bound
- c @@@ symbol = beta n alpha beta
- c @@@ symbol = repeat n
- c @@@ symbol = (expression)
- c @@@ constraint = (logical-expression)
- c @@@ tied = list-of-symbols