LA-UR-25-24192

Approved for public release; distribution is unlimited.

Title:	MCNP® Class Criticality Calculations with MCNP6 C-P-21: Case Study: BEAVRS Criticality Benchmark
Author(s):	Josey, Colin James
Intended for:	Report
Issued:	2025-04-30

Los Alamos NATIONAL LABORATORY

MCNP® MONTE CARLO N-PARTICLE

MCNP® Class

Criticality Calculations with MCNP6

C-P-21: Case Study: BEAVRS Criticality Benchmark

Colin Josey

NNSA Managed by Triad National Security, LLC, for the U.S. Department of Energy's NNSA.

MCNP® Trademark

MCNP® and Monte Carlo N-Particle® are registered trademarks owned by Triad National Security, LLC, manager and operator of Los Alamos National Laboratory. Any third party use of such registered marks should be properly attributed to Triad National Security, LLC, including the use of the ® designation as appropriate.

- Please note that trademarks are adjectives and should not be pluralized or used as a noun or a verb in any context for any reason.
- Any questions regarding licensing, proper use, and/or proper attribution of Triad National Security, LLC marks should be directed to trademarks@lanl.gov.

Outline

The BEAVRS Core

MCNP
MONTE CARLO N-PARTICLE

BEAVRS [1] is an open MIT model of a 4-loop Westinghouse pressurized water reactor. This reactor model includes:

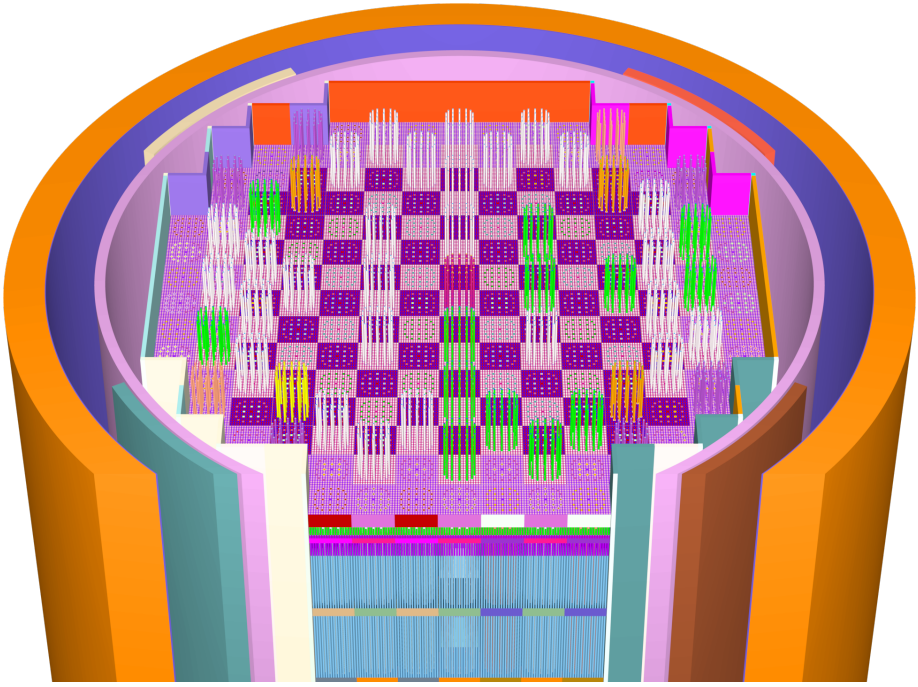
- 3 levels of enrichment (first cycle)
- Guide tubes, instrument tubes, burnable absorbers, and control rods
- Detail below, above, and around the core
- Grid spacers and gaps between assemblies

For this class, we will focus on its hot-zero-power initial configuration:

- Cycle 1 fresh fuel
- Uniform 566.5 K temperature
- All but control rod bank D fully out

The BEAVRS Core

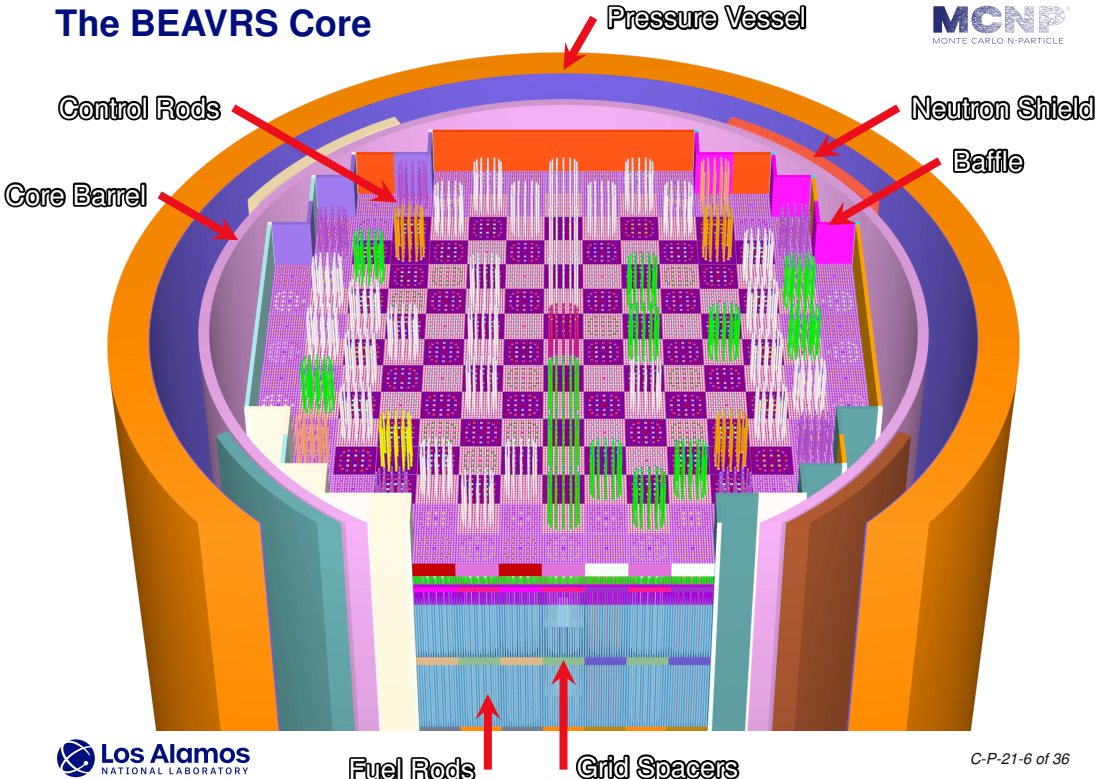
MCNP
MONTE CARLO N-PARTICLE



Los Alamos
NATIONAL LABORATORY

The BEAVRS Core

MCNP
MONTE CARLO N-PARTICLE



Los Alamos
NATIONAL LABORATORY

Case Study Goals

MCNP
MONTE CARLO N-PARTICLE

Your simulation goals will always be dependent on your needs and should be evaluated on a case-by-case basis. For this example, we will introduce some rather arbitrary goals so that we can go through the process.

Let's say we want to know:

- k-eigenvalue to within 10 pcm.
- Flux at the assembly and fuel rod level to 1% error (95% confidence interval).
- An estimate on how difficult it would be to get pellet-level results to within 1%.

These constraints are pretty strict, and this geometry is massive, so it is unlikely that our first pass will meet all of these goals.

Note that these constraints are so tight that we're likely well below the error in the data!

Model Setup

MCNP
MONTE CARLO N-PARTICLE

We first start by putting together the geometry. As this model is complex, we used the MCNP Python API to generate it. The input files are included with this class.

For the source, it is simple to start particles in a cylinder that overlaps all the fuel rods in the model. We do not need to refine this initial guess any further.

We now need to decide the following:

- Shannon Entropy mesh:
  - Automatic or specified?
  - How fine should it be?
- Physics options:
  - Do we need $S(\alpha, \beta)$?
  - Do we need `DBRC`?
- Material options:
  - Do we need `TMP`?
  - Do we need to specify at-temperature ACE data?
- KCODE options:
  - How many particles per batch?
  - How many batches?

Notation

MCNP
MONTE CARLO N-PARTICLE

Rules of Thumb

Some of the analyses we will show in these slides are not practical to repeat due to their extreme computational cost. As such, we will provide rules of thumb in boxes like these as we go.

It took around 100 thousand CPU hours to generate the results in these slides!

Parameters Used for BEAVRS

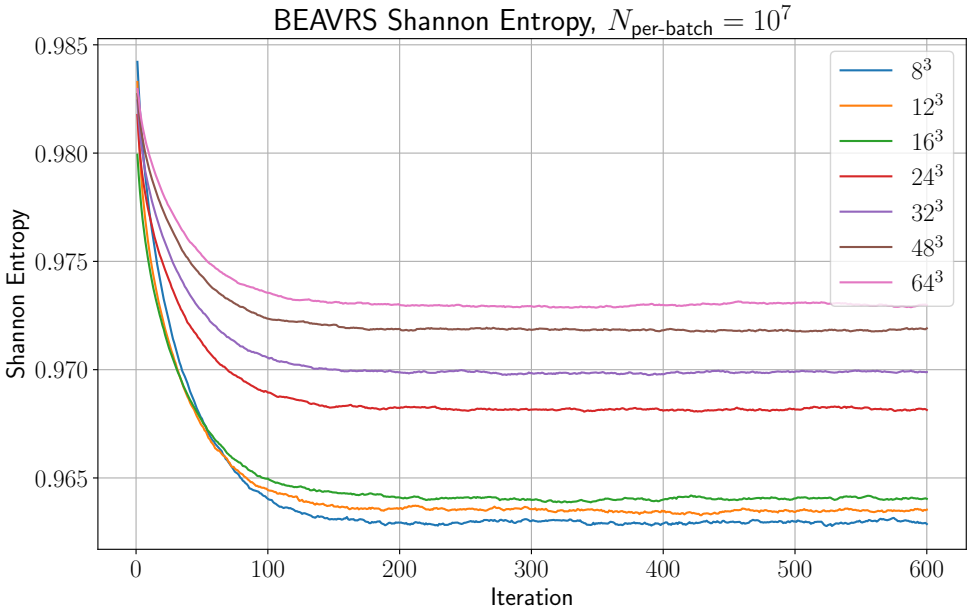
Bold values are the parameters selected based on nearby discussion.

Exploring Shannon Entropy

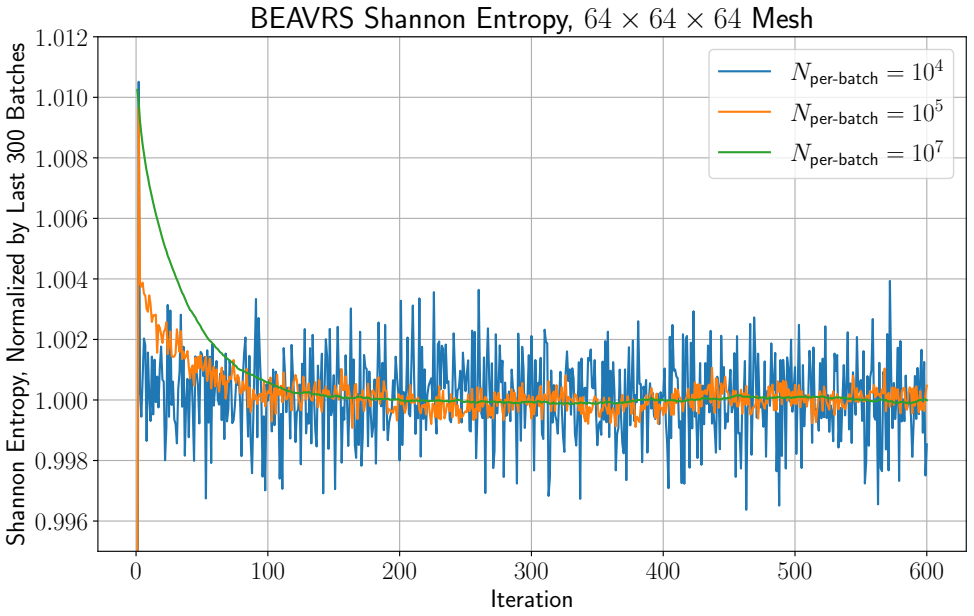
MCNP
MONTE CARLO N-PARTICLE

- Before we set `KCODE`, we need to configure the entropy mesh
- We can use an automatically generated mesh by not setting `HSRC`
- We can use our own by setting `HSRC`
  - Coarse or fine?
  - Where is the dividing line?
- But what about the fission matrix? It uses the same grid.

Selecting a Shannon Entropy Grid

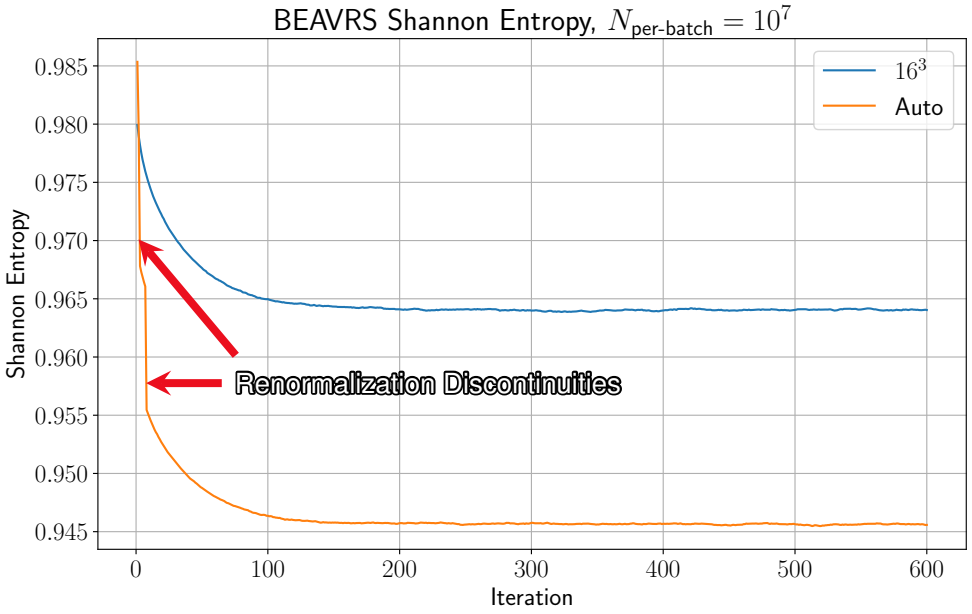


Selecting a Shannon Entropy Grid



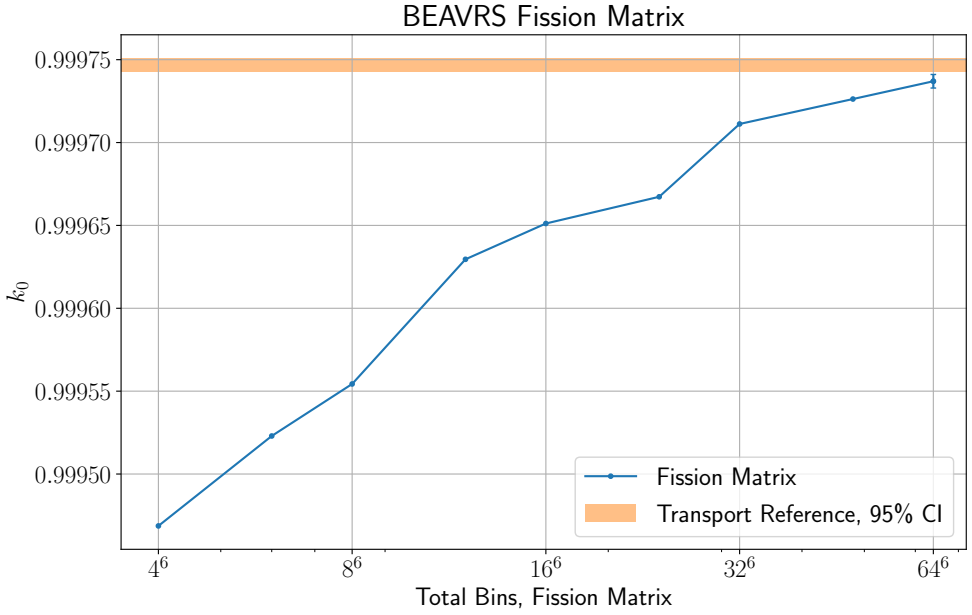
If the Shannon entropy mesh is too fine, noise begins to dominate and it becomes harder to distinguish convergence.

What About Automatic Entropy?



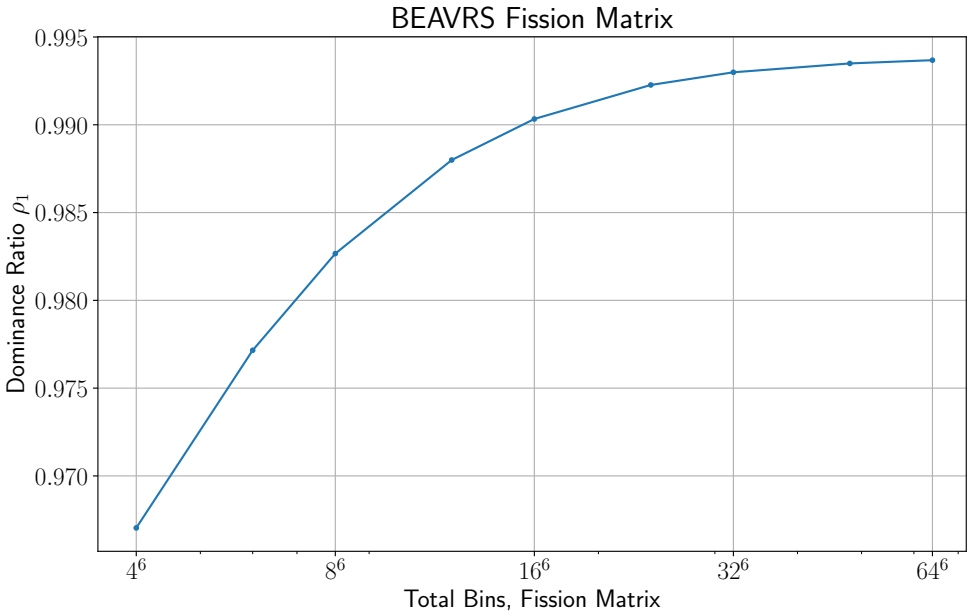
Auto (no HSRC) works well, but resizing can introduce discontinuities.

Fission Matrix Eigenvalue Convergence



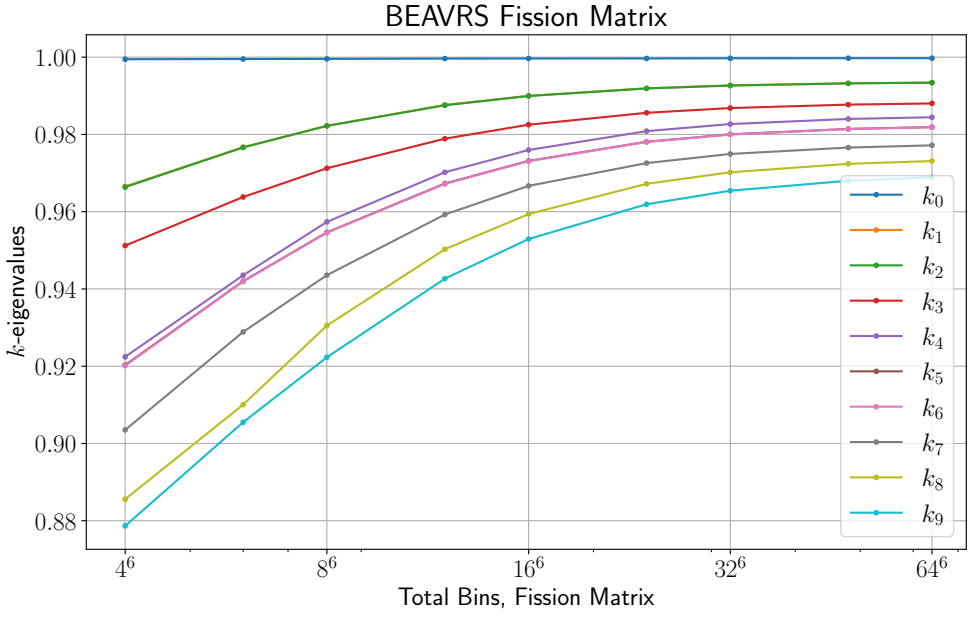
KCODE and fission matrix will converge if the mesh is fine enough.

Fission Matrix Dominance Ratio



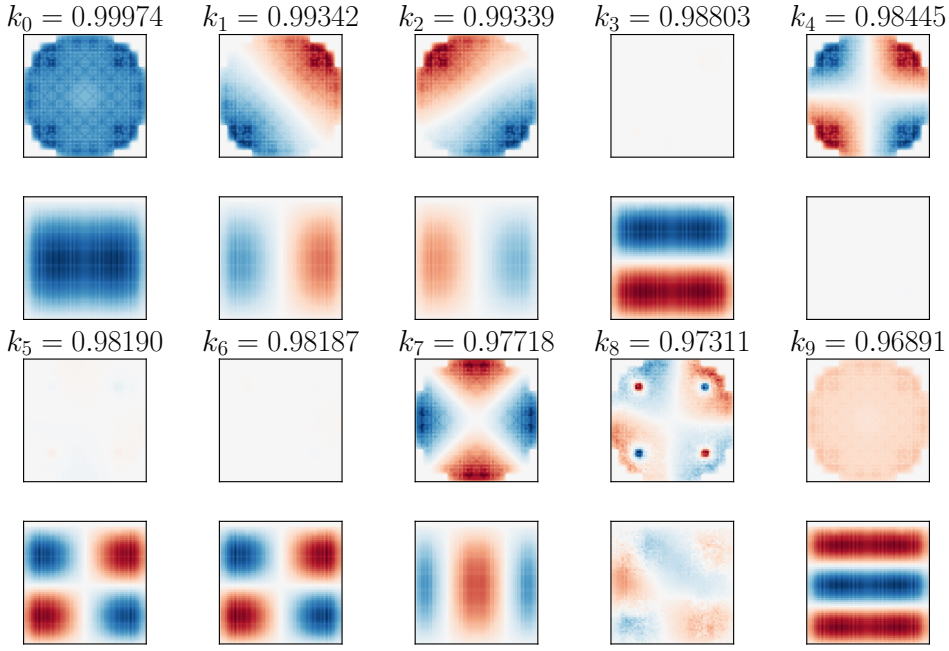
Higher modes converge more slowly than the fundamental.

Fission Matrix Higher Modes



Fission Matrix Higher Modes

MCNP
MONTE CARLO N-PARTICLE



Entropy Summary

MCNP
MONTE CARLO N-PARTICLE

- Mesh fineness depends on application
- For k-eigenvalue, a coarse mesh is ideal
- For higher mode calculations, a fine mesh is ideal
- The finer the mesh, the more particles you need for “distinguishing power”

Rule of Thumb

If you don’t care about fission matrix analysis, use a mesh with roughly 1000 to 10000 cells, with discretization in x, y, and z, or use automatic mesh generation.

Parameters Used for BEAVRS

```
hsrc 16 -161.2773 161.2773 16 -161.2773 161.2773 16 36.7480 419.704
kcode n-per-batch 1.0 200 n-batches
```

Physics

MCNP MONTE CARLO N-PARTICLE

What physics should we enable?

Simulation	k-Eigenvalue	δk, pcm	Performance (M/hr)
Reference	0.999747 ± 0.000002	—	596 ± 7
No `h-h2o` $S(\alpha, \beta)$	0.999815 ± 0.000010	6.8	597
No `x-uo2` $S(\alpha, \beta)$	0.999830 ± 0.000010	8.3	600
No `DBRC`	1.000264 ± 0.000010	51.7	603

Surprisingly, `DBRC` has a more significant contribution to k than $S(\alpha, \beta)$.

Rule of Thumb

It's better to enable physics than disable physics and find out you needed it later. Most options do not take much additional time. So we recommend adding all relevant $S(\alpha, \beta)$ and enabling `DBRC`.

Temperature Effects

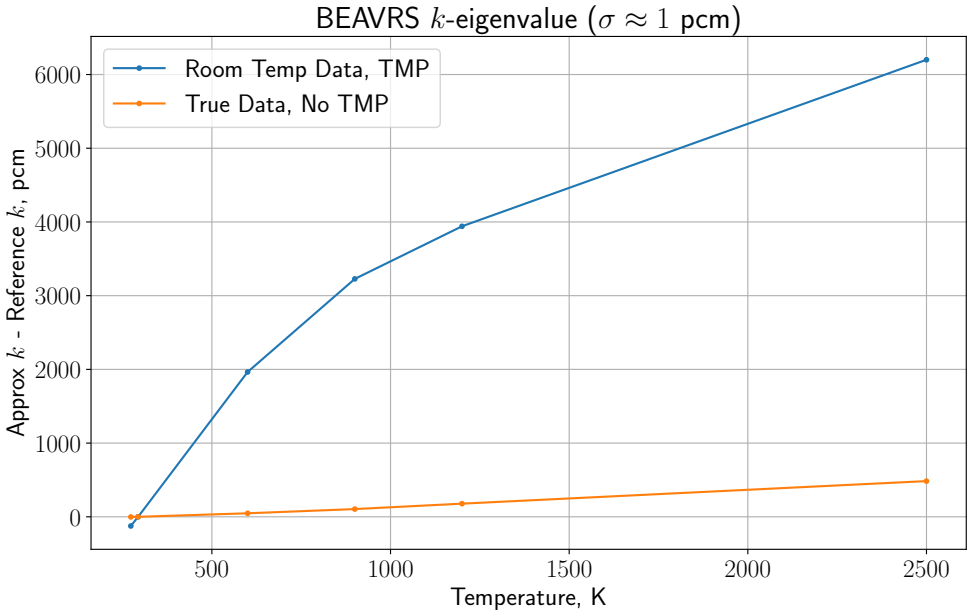
MCNP
MONTE CARLO N-PARTICLE

What about temperature? Should we:

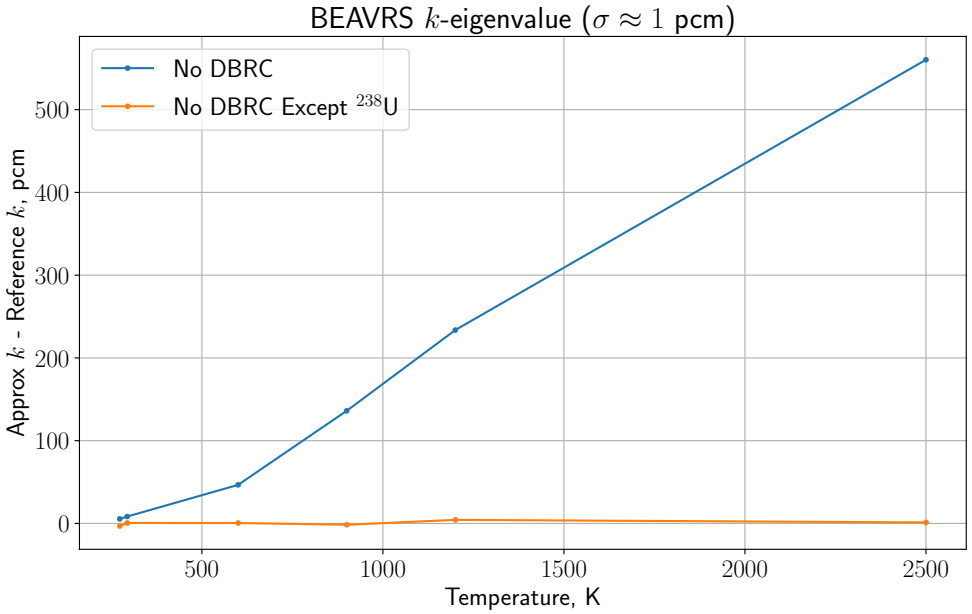
- Use room-temperature ACE files and the default `TMP`?
- Use room-temperature ACE files but set the `TMP`?
- Use at-temperature ACE files and the default `TMP`?
- Use both at-temperature ACE and set the `TMP`?

What about DBRC? It's also temperature-sensitive. Should we specify all or some nuclides?

The Effect of the TMP Card and ACE Data



The Effect of DBRC



Physics Summary

MCNP
MONTE CARLO N-PARTICLE

Rules of Thumb

When not at room temperature:

- Use either the nearest-temperature ACE file (less accurate) or use NJOY to process a custom ACE set (most accurate)
- Always set the `TMP` card on all cells
- Enable DBRC with at least ^{238}U if present

Parameters Used for BEAVRS

`nlib=`**`11c`** (closest available data, 600 K)
`tmp=`**`4.882e-08`**
`dbrc endf=81 isos=`**`{Every nuclide}`**
`mtX` **`h-h2o.84t`** for all water
`mtX` **`o-uo2.73t u-uo2.73t`** for all fuel

Selecting Batch Size and Active Cycle Count

MCNP MONTE CARLO N-PARTICLE

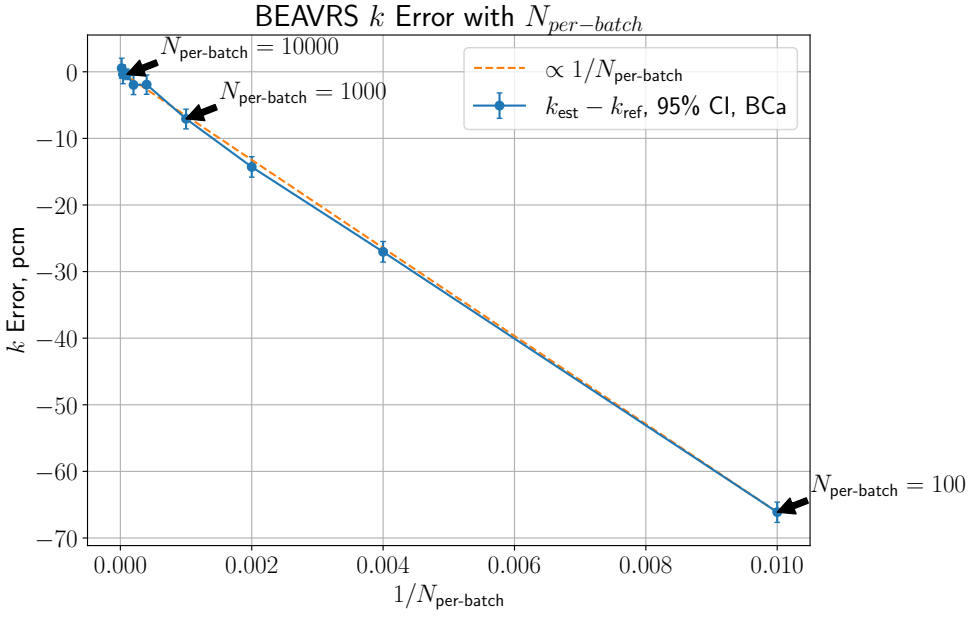
We have two last unknowns: $N_{\text{per-batch}}$ and N_{active}.

General procedure:

1. Determine $N_{\text{per-batch}}$ that will be unbiased.
2. Run simulation with that $N_{\text{per-batch}}$ and a reasonable N_{active}.
3. Using computed variance, extrapolate needed multiplier for N.
4. Re-run the simulation with this new value and verify.

In particular, directly computing the batch size that would be unbiased in step #1 is extremely expensive, so the next few slides will help show some rules of thumb.

Effect of Batch Size on k



Introducing Tallies

4 FMESH flux tallies of different scales were added to examine bias in small spatial regions:

Tally	δx and δy, cm	δz, cm	N_{energies}
Assembly Flux	21.503	382.956	1
Assembly Flux w/ z and E	21.503	21.275	3
Fuel Rod Flux	1.260	382.956	1
Fuel Rod Flux w/ z and E	1.260	1.264	3

The second two tallies were only placed over the central assembly to reduce memory usage.

The last tally is roughly pellet-scale.

Effect of Batch Size on Tallies

MCNP MONTE CARLO N-PARTICLE

Tally	Max Rel. Bias $N_{\text{per-batch}} = 100$, 95% CI	$N_{\text{per-batch}}$ Required for 0.1%
Assembly Flux	$[0.0247, 0.0352]$	≥ 3500
Assembly Flux w/ z and E	$[0.0685, 0.1034]$	≥ 10000
Fuel Rod Flux	$[0.0333, 0.0387]$	≥ 4000
Fuel Rod Flux w/ z and E	$[-0.002, 0.337]$	≥ 35000

Tallies with smaller phase-space extents require more $N_{\text{per-batch}}$ to be unbiased.

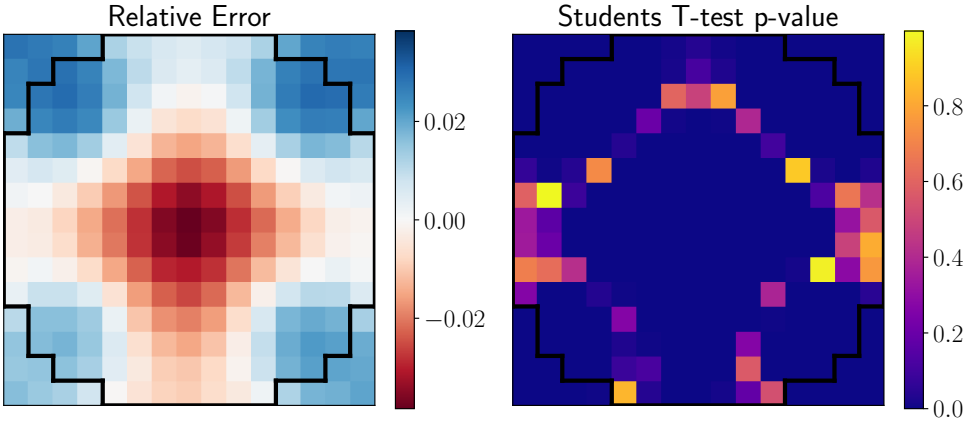
Rule of Thumb

You want your bias to be much smaller than your target error. As you cannot predict bias ahead of time, err on the side of large batch sizes.

Effect of Batch Size on Tallies

MCNP
MONTE CARLO N-PARTICLE

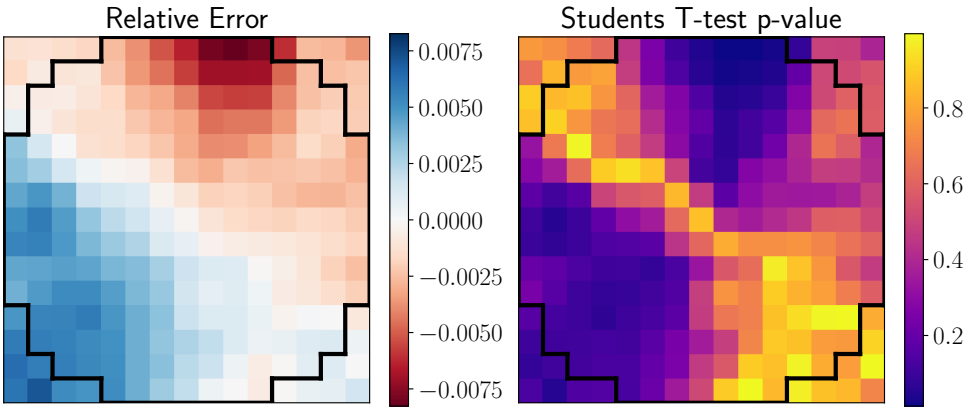
Assembly-level Flux, $N_{\text{per-batch}} = 100$ vs. 10^7



Effect of Batch Size on Tallies

MCNP MONTE CARLO N-PARTICLE

Assembly-level Flux, $N_{\text{per-batch}} = 2500$ vs. 10^7



$N_{\text{per-batch}}$ Conclusions

MCNP MONTE CARLO N-PARTICLE

Directly computing bias took 500 simulations and even then the error bars were still too large for the pellet-level tallies.

So, we don't expect you to repeat them. We take the previous numbers and add a factor of roughly 10 to be conservative.

Depending on your application, use a larger factor. You likely will need more particles to converge statistics anyways.

Rules of Thumb

- k-eigenvalue-only analysis: $N_{\text{per-batch}} \geq 10000$
- Assembly-level analysis: $N_{\text{per-batch}} \geq 50000$
- Pellet-level analysis: $N_{\text{per-batch}} \geq 500000$
- Even finer analysis: $N_{\text{per-batch}} \gg 500000$

Consider these to be minimum values.

Getting N_{total}

MCNP MONTE CARLO N-PARTICLE

Let's re-run the simulation with $N_{\text{per-batch}} = 10^7$ and $N_{\text{active}} = 400$, for $N_{\text{total}} = 4 \times 10^9$. Why these numbers?

- Calculations with $N_{\text{per-batch}} = 10^7$ were already done before knowing 500000 would be a good batch size.
- 400 is 2 times the number of inactive, giving a good ratio of active to inactive.

Tally	Max Rel. σ	N_{total} for 95% CI < 1%
Assembly Flux	0.006	2.24×10^{10}
Assembly Flux w/ z and E	0.056	1.98×10^{12}
Fuel Rod Flux	0.003	3.53×10^{9}
Fuel Rod Flux w/ z and E	0.223	3.05×10^{13}

95% CI range $\approx 3.92\sigma$

We're already converged for fuel rods, and close for assemblies.

The fine tallies can require prodigious numbers of particles to converge, however.

Allocating N_{total}

MCNP MONTE CARLO N-PARTICLE

From all that we've seen, to achieve our goals we need N_{total} to be $\geq 2.24 \times 10^{10}$, and $N_{\text{per-batch}} > 500000$. How do we split them up?

Advantages of increasing $N_{\text{per-batch}}$:

- Ensures bias is constrained
- Parallel performance is higher with more particles

Advantages of increasing N_{active}:

- If you need better convergence, you can restart your run
- MCNP uses more memory with increasing $N_{\text{per-batch}}$ than increasing N_{active}
- More time is spent computing tallies than converging, improving figure of merit

Recommendations for N_{total}

MCNP MONTE CARLO N-PARTICLE

Rules of Thumb

- Ensure $N_{\text{active}} \geq 100$ (central limit theorem)
- Ensure $N_{\text{active}} \geq 2N_{\text{inactive}}$ (spend majority of time tallying)
- Ensure $N_{\text{per-batch}}$ is large enough based on earlier rules of thumb
- Maximize $N_{\text{per-batch}}$ based on resources
- If N_{total} is still too low, increase N_{active}
- Run the simulation, and use continue runs if variance is still too high

So after all of this, to get the 95% CI below 1% on all the tallies we are interested in:

Parameters Used for BEAVRS

```
kcode 1e7 1.0 200 2500
```

For pellet-level precision, it would probably be more efficient to run 1000 simulations with independent random number seeds.

Underestimating Standard Deviation

MCNP
MONTE CARLO N-PARTICLE

For fixed source problems, the standard deviation MCNP provides is a good estimate of the true standard deviation of the underlying distribution.

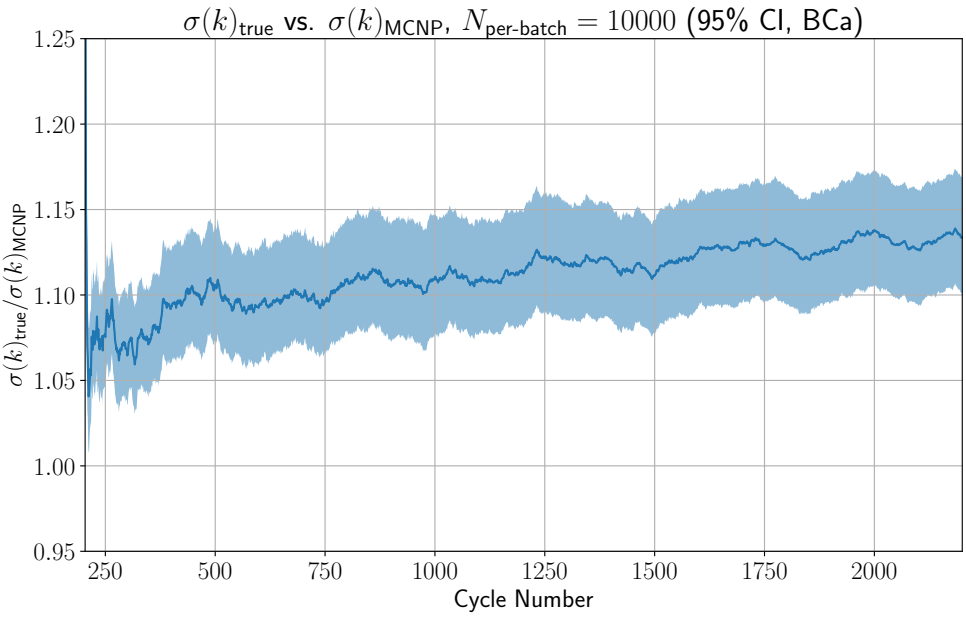
For k-eigenvalue, however, there's a bit of a problem:

- Since each fission site came from the previous generation, each generation is loosely autocorrelated
- This correlation increases with dominance ratio
- Our estimators for k do not include any compensation for autocorrelation

Rules of Thumb

MCNP's standard deviation is a lower bound for k-eigenvalue calculations. If your application depends on the exact value of your variance, run MCNP multiple times with different seeds and compute it afterwards.

Underestimating Standard Deviation



Questions?

MCNP
MONTE CARLO N-PARTICLE

The BEAVRS Core

Exploring Shannon Entropy

Toggling Physics

Temperature Effects

Selecting Batch Size and Active Cycle Count

Underestimating Standard Deviation

Backup Slides

Outline

MCNP
MONTE CARLO N-PARTICLE

References

References

[1] N. Horelik, B. Herman, M. Ellis, S. Kumar, J. Liang, B. Forget, and K. Smith, "Benchmark for Evaluation and Validation of Reactor Simulations (BEAVRS), v2.0.2," Tech. Rep., 2018.
URL: https://crpg.mit.edu/wp-content/uploads/2017/01/BEAVRS_2.0.2_spec.pdf

[2] J. A. Kulesza, T. R. Adams, J. C. Armstrong, S. R. Bolding, F. B. Brown, J. S. Bull, T. P. Burke, A. R. Clark, R. A. Forster, III, J. F. Giron, T. S. Grieve, C. J. Josey, R. L. Martz, G. W. McKinney, E. J. Pearson, M. E. Rising, C. J. Solomon, Jr., S. Swaminarayan, T. J. Trahan, S. C. Wilson, and A. J. Zukaitis, "MCNP® Code Version 6.3.0 Theory & User Manual," Los Alamos National Laboratory, Los Alamos, NM, USA, Tech. Rep. LA-UR-22-30006, Rev. 1, Sep. 2022.
DOI: 10.2172/1889957