

LA-UR-13-20454

Approved for public release; distribution is unlimited.

Title: Fission Matrix Capability for MCNP, Part II - Applications

Author(s): Carney, Sean E.
Brown, Forrest B.
Kiedrowski, Brian C.
Martin, William R.

Intended for: Mathematics & Computation 2013, 2013-05-05/2013-05-09 (Sun Valley, Idaho, United States)
MCNP documentation
Web



Disclaimer:

Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by the Los Alamos National Security, LLC for the National Nuclear Security Administration of the U.S. Department of Energy under contract DE-AC52-06NA25396. By approving this article, the publisher recognizes that the U.S. Government retains nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

FISSION MATRIX CAPABILITY FOR MCNP, PART II - APPLICATIONS

S.E. Carney¹, F.B. Brown², B.C. Kiedrowski², W.R. Martin¹

¹ University of Michigan, NERS Department
2355 Bonisteel Boulevard, Ann Arbor, MI 48109, USA
seanec@umich.edu; wrm@umich.edu

² Los Alamos National Laboratory, Monte Carlo Codes Group
PO Box 1663, MS A143, Los Alamos, NM 87545, USA
fbrown@lanl.gov; bckiedro@lanl.gov

ABSTRACT

This paper describes the initial experience and results from implementing a fission matrix capability into the MCNP Monte Carlo code. The fission matrix is obtained at essentially no cost during the normal simulation for criticality calculations. It can be used to provide estimates of the fundamental mode power distribution, the reactor dominance ratio, the eigenvalue spectrum, and higher mode spatial eigenfunctions. It can also be used to accelerate the convergence of the power method iterations. Past difficulties and limitations of the fission matrix approach are overcome with a new sparse representation of the matrix, permitting much larger and more accurate fission matrix representations. Numerous examples are presented. A companion paper (Part I – Theory) describes the theoretical basis for the fission matrix method.

Key Words: Monte Carlo, neutron transport, eigenvalues

1. INTRODUCTION

Continuous-energy Monte Carlo codes such as MCNP [1] simulate neutron behavior using the best available nuclear data, accurate physics models, and detailed geometry models. Reactor criticality calculations for k_{eff} and the power distribution are carried out iteratively, using the power method, where batches of neutrons are simulated for a single generation. The fission matrix approach was proposed in the earliest works on Monte Carlo criticality calculations [2-4] and has been tried by many researchers over the years. The present work takes advantage of the very large computer memories available today and a new sparse matrix representation to overcome past difficulties. A companion paper (Part I - Theory [5]) describes the theoretical basis for this fission matrix capability.

2. FISSION MATRIX METHOD

2.1 Fission Matrix Equations

As derived in [5], if the physical problem is segmented into N spatial regions, and the k-effective form of the integral transport equation is integrated over the volumes of each initial region J , with $\vec{r}_0 \in V_J$, and final region I , with $\vec{r} \in V_I$, then the following equations are obtained:

$$S_I = \frac{1}{k} \cdot \sum_{J=1}^N F_{I,J} \cdot S_J \quad (1)$$

where

$$F_{i,j} = \int_{\bar{r} \in V_i} d\bar{r} \int_{\bar{r}_0 \in V_j} d\bar{r}_0 \frac{S(\bar{r}_0)}{S_j} \cdot H(\bar{r}_0 \rightarrow \bar{r}), \quad S_j = \int_{\bar{r} \in V_j} S(\bar{r}') d\bar{r}' \quad (2)$$

S_j is the fission neutron source in region J , and H is an energy- and angle-averaged Green's function. The matrix element $F_{i,j}$ is equal to the number of next-generation fission neutrons born in region I due to one average fission neutron starting in region J . The matrix \bar{F} is called the *fission matrix*. The fundamental mode eigenvalue of this matrix is formally identical to k -effective, and the fundamental mode eigenfunction is the regionwise fission neutron source distribution. In matrix-vector form, Eq. (1) is

$$\bar{S} = \frac{1}{k} \cdot \bar{F} \cdot \bar{S} \quad (3)$$

where \bar{S} is a vector of length N giving the single-generation production of neutrons in each region from fission, and \bar{F} is a full matrix of size $N \times N$. Higher eigenmodes of Eq. (3) can be determined according to:

$$\begin{aligned} \bar{S}_n &= \frac{1}{k_n} \cdot \bar{F} \cdot \bar{S}_n & n &= 0, 1, \dots, N \\ k_0 &> |k_1| > |k_2| > \dots > |k_N| \end{aligned} \quad (4)$$

where the subscript n refers to the mode, with $n=0$ the fundamental mode. For a problem with N regions in the mesh for the fission matrix, \bar{F} is an $N \times N$ matrix with N discrete eigenvalues. Because \bar{F} is a nonsymmetric matrix, the eigenvalues and eigenvectors may be complex, although the fundamental mode must be strictly real and nonnegative.

2.2 Monte Carlo Estimation of the Fission Matrix

In this section, we describe the choices implemented in MCNP6 for the initial proof-of-principle testing of the fission matrix method, including the region shapes and size, the initial source guess, the tallying procedure, the iteration strategy, and parallel computing issues.

2.2.1 Regions for fission matrix tallies

The choice of region shapes and sizes for determining the fission matrix is arbitrary, as long as all fissionable regions in the physical problem are covered. For the initial testing in MCNP6, we have chosen to use a simple, uniform, 3D, Cartesian mesh, with different numbers of mesh intervals permitted in the x -, y -, and z -directions. The mesh overlays the detailed Monte Carlo geometry for the physical problem and must encompass all fissionable regions in the problem. The choice of mesh for tallying the fission matrix does not affect the ordinary Monte Carlo tracking in any manner. Using a uniform mesh permits determining the ijk spatial mesh indices for tallies of the source at x, y, z at only trivial computing cost. (Further development will extend the mesh choices to nonuniform Cartesian and cylindrical meshes.)

2.2.2 Initial source guess

While the initial guess for the fission neutron source distribution is arbitrary for criticality calculations, the use of a uniform volumetric source in fissionable regions of the problem is the most prudent choice. This ensures that tallies of fission matrix elements will be made for all fissionable regions in the initial stages of the power iteration process used in MCNP. (Future development will automate this and include stratified sampling techniques.)

2.2.3 Tallying the fission matrix elements

The elements of the fission matrix can be estimated at essentially no extra cost during the normal Monte Carlo criticality simulation – simply remember the region a fission neutron was born in (J), determine the region a next-generation fission neutron is produced in (I), and tally the (I, J)-th element of the fission matrix. For Monte Carlo codes that use a fixed number of starting source neutrons for every cycle, the region tallies are simply incremented by 1 for each neutron; for MCNP with a varying number of neutrons starting each cycle, the tallies need to be incremented by M_0/M , where M_0 is the number of neutrons starting the initial cycle, and M is the number that started the current cycle. Before solving the fission matrix equations, the tallies need to be normalized by dividing each (I, J)-th element by the total number of starters in region J .

Tallies for fission matrix elements can be made using only the locations of fission neutron sources at the start and end of each batch, without incurring any overhead during the random walk simulation of the neutrons. This approach eliminates any inter-process communications overhead during MPI parallel processing, since the entire fission matrix estimation can be performed on the master node, using only the existing “fission bank” information. (If fission matrix tallies were instead made during the neutron random walks, for instance using a path length estimator, then the entire set of fission matrix tallies would need to be stored on each parallel computing node. The entire set of fission matrix tallies would then need to be accumulated across all nodes, which could involve many GB of MPI message-passing at considerable overhead cost.)

If a coarse mesh is used to define the spatial regions, then the fission matrix tallies cannot be made until after the fission source distribution has converged, since the spatial weighting functions in Eq. (2) correspond to the stationary source distribution. However, if a fine enough mesh is used such that

$$\frac{S(\bar{v}_j)}{S_j/V_j} \approx 1 \quad \text{for } \bar{v}_j \in V_j \quad (5)$$

then Eq. (2) becomes independent of the spatial weighting functions, and valid tallies can be made even before the source distribution converges.

2.2.4 Tally updates and iteration strategy

During a standard k-effective calculation, at the end of each cycle the $F_{i,j}$ estimators are updated by tallying the fission neutron weight using the starting and ending mesh region numbers for each point in fission bank. If the mesh is fine enough that Eq. (5) is valid, the tallies may be accumulated over cycles even during the inactive cycles, prior to convergence of the fission source distribution. In practice, we have chosen to begin the fission matrix tallies at the 4th iteration cycle, and to accumulate the tallies over all subsequent cycles. (Single-cycle estimates of the fission matrix are never used.)

At any desired cycle, the $F_{i,j}$ tallies may be normalized by dividing by the total accumulated source in the starting regions (i.e., J index) to form the normalized fission matrix. Then the eigenvalues and eigenvectors of the fission matrix may be found by simple power iteration. If higher modes are desired, then Hotelling deflation or direct solvers may be also be used. In practical application, the fission matrix tallies could be accumulated for all problem iterations, and then the fission matrix eigenvalues and eigenvectors could be determined only after the Monte Carlo calculation had completed. If it is desired to obtain the fission matrix solution during the Monte Carlo calculation, to potentially use it to accelerate the overall source

convergence, then the fundamental mode eigenvector could be determined at the end of any cycle. When using the fission matrix solution to accelerate source convergence, only the fundamental mode eigenvector is required.

2.3 Sparse Fission Matrix Representation

The principal limitation on the accuracy of the fission matrix approach is, and always has been, the size of the regions for each fission matrix element. Typically, a regular 3D spatial mesh with $N = N_I \times N_J \times N_K$ elements is used, giving an $N \times N$ fission matrix, with N^2 entries. A 100x100x100 spatial mesh would give rise to a fission matrix with 10^{12} elements, which could not be stored even on today's computers.

To overcome this limitation, we investigated the use of a sparse fission matrix. Clearly, not every region in a large 3D problem is tightly coupled to every other region; fission neutrons induce most further fissions in neighboring regions, and few or none in distant regions. To investigate this, we incorporated tallies into MCNP to diagnose the fractions of induced fission neutrons in neighboring regions, and examined the structure of the fission matrix for a typical 2D PWR problem. Fig. 1 shows the structure of the fission matrix for the 15x15x1 mesh case, where each mesh element corresponds to an assembly-sized region. It is evident from the banded structure of the fission matrix that neutrons from one assembly cause nearly all of their fissions in that assembly and the nearest 2 neighboring assemblies in each direction. Only about 0.5% of the fission matrix tallies correspond to more distant fissions. Thus, storing a sparse, banded fission matrix (rather than the full matrix) offers a promising mechanism for mitigating the storage problem. Fig. 2 shows the sparse fission matrix structure obtained by limiting the coupling to only 2 nearest neighbor assemblies in each direction. Of course, as the spatial mesh is refined,

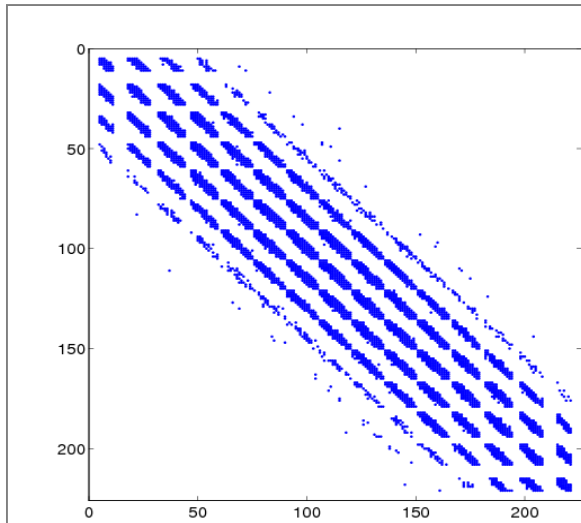


Fig. 1. Fission matrix structure for a 2D whole-core PWR model, for a 15 x 15 x 1 spatial mesh. Matrix dimensions are 225 x 225. Points in blue are non-zero elements of the fission matrix.

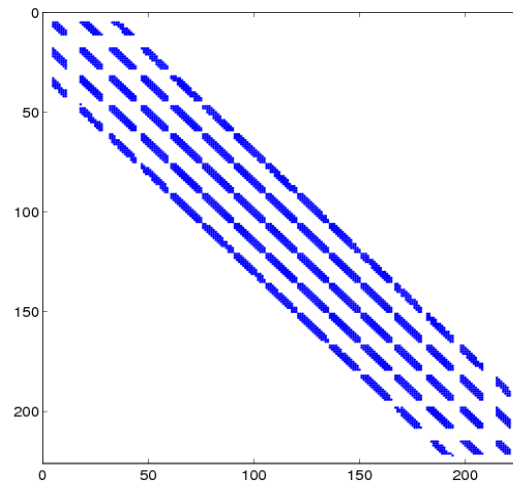


Fig. 2. Sparse fission matrix structure for a 2D whole-core PWR model, for a 15 x 15 x 1 spatial mesh. Spatial coupling limited to ± 2 neighboring assemblies.

more neighbor bands will need to be retained. For the testing described below, we have used the sparse fission matrix representation, with the number of bands in the matrix chosen to include spatial regions corresponding to the nearest 2 neighboring assemblies in each direction. That is,

for a 15x15x1 spatial mesh, the fission matrix sparse storage is 225x25 elements, rather than the full 225x225 elements. For the 30x30x1 mesh, the sparse matrix representation is 900x81; for the 60x60x1 mesh, the matrix is 3600x289. Additionally, the very few tallies outside of the matrix bands were accumulated in the nearest banded fission matrix element in order to preserve overall neutron balance.

3. EXAMPLES OF HIGHER MODE ANALYSIS USING THE FISSION MATRIX

In this section we provide a several examples for 2D and 3D reactors where MCNP was used to compute a fission matrix, and then higher eigenvalues and eigenfunctions of the fission neutron source distribution are obtained from the fission matrix. All MCNP calculations were performed with continuous-energy collision physics using ENDF/B-VII.0 cross-section data [6].

3.1 2D Whole-core PWR

A 2D whole-core PWR model is shown in Fig. 3 (previously used in [7], based on [8]). The fission matrix was accumulated during standard KCODE calculations with 500K neutrons/batch. Tallies for the fission matrix elements were made only for the 4th and successive batches. k_{eff} , the fundamental mode eigenfunction, and the dominance ratio from the fission matrix were determined via an iterative method. Higher-mode eigenvalues and eigenfunctions for the fission matrix were determined by either a direct non-symmetric matrix routine or by using Matlab.

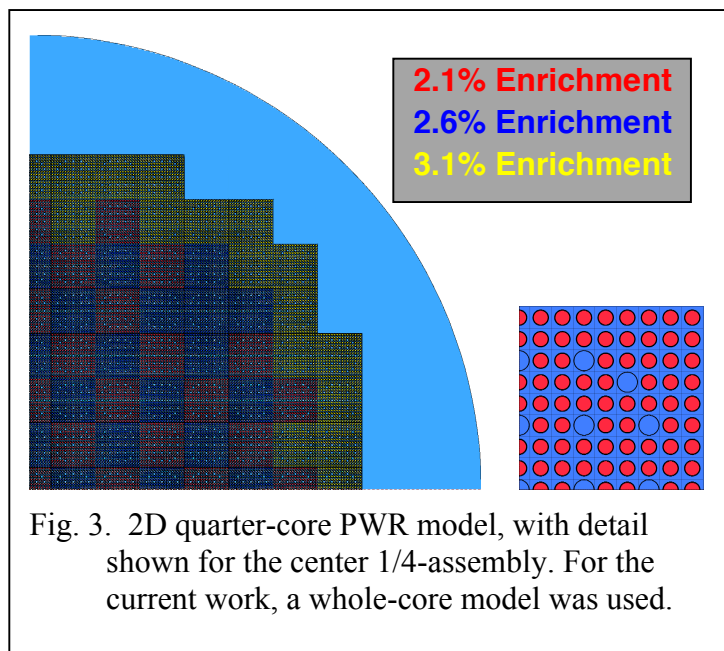


Fig. 3. 2D quarter-core PWR model, with detail shown for the center 1/4-assembly. For the current work, a whole-core model was used.

Fig 4. shows the fundamental mode eigenfunction for various spatial resolutions used in tallying the fission matrix, corresponding to full, $\frac{1}{4}$, $\frac{1}{16}$ th, and $\frac{1}{64}$ th of the assembly size. As discussed in Part I [5], a convergence study of the eigenvalue spectrum with mesh refinement was used to determine that the 120x120 mesh provided sufficient resolution to produce accurate eigenvalues and eigenfunctions.

Fig. 5 shows the fundamental eigenmode (i.e., the fission neutron source distribution) and 15 higher eigenmodes for the 120x120x1 mesh case. These plots are especially interesting, since the higher eigenmodes cannot normally be obtained directly from a Monte Carlo calculation. For this calculation, the spatial mesh included 14,400 regions, and the fission matrix size was 14,400x14,400. During the calculation, the dominance ratio k_1/k_0 was obtained every cycle, and all 14,400 eigenvalues and eigenfunctions were obtained after the MCNP calculation using Matlab.

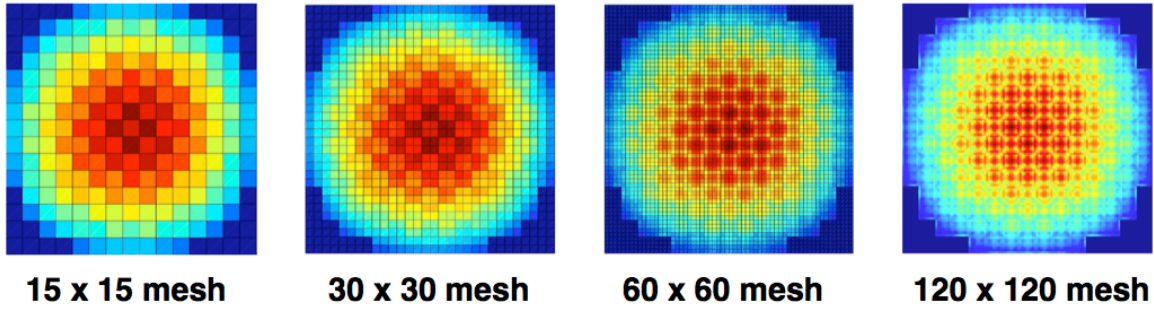


Fig. 4. 2D Whole-core PWR model: Fundamental mode eigenfunctions for fission neutron source distribution, obtained from the fission matrix for different spatial tally meshes. 500K neutrons/cycle, fission matrix tallies for cycles 4-55, sparse fission matrix limited to nearest neighbors within ± 2 assembly pitch distances.

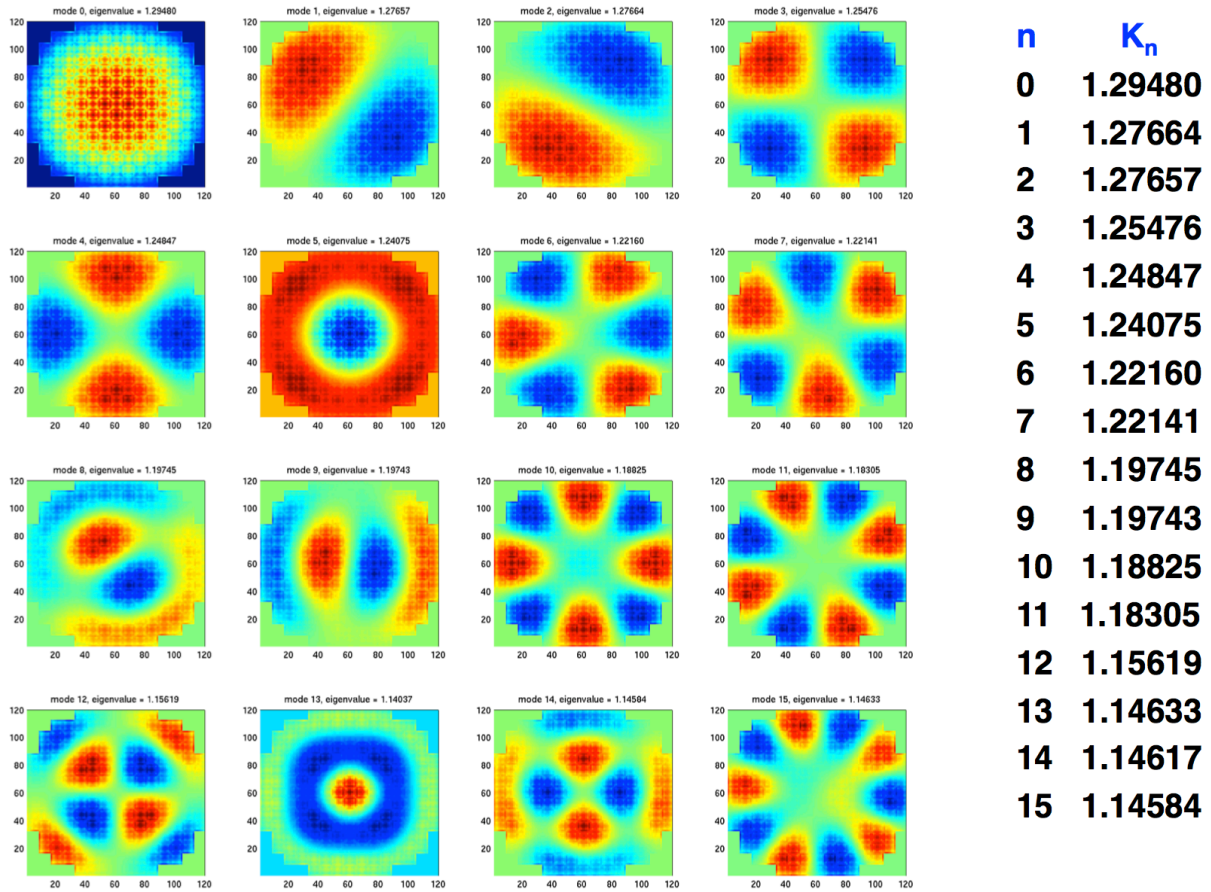


Fig. 5. 2D Whole-core PWR model: First 16 eigenfunctions and eigenvalues for fission neutron source distribution, obtained using a 120x120 tally mesh for the fission matrix (14,400 tally regions, 14,400x14,400 fission matrix). 5M neutrons/cycle, cycles 4-100.

3.2 Kord Smith Challenge Problem

This is a detailed 3D whole core reactor model developed for computer performance benchmarking [9]. The geometry is shown in Fig. 6. For normal MCNP criticality calculations, this problem takes about 55 cycles to converge the source distribution. The fission matrix tallies were performed during cycles 4-55, using 1M neutrons/cycle, a 42x42x20 tally mesh (1/4-assembly, with 20 axial segments), and a 35280x4913 sparse fission matrix. Fig. 7 shows the fundamental and 14 higher modes for the fission neutron source distribution, and Table I shows the corresponding eigenvalues.

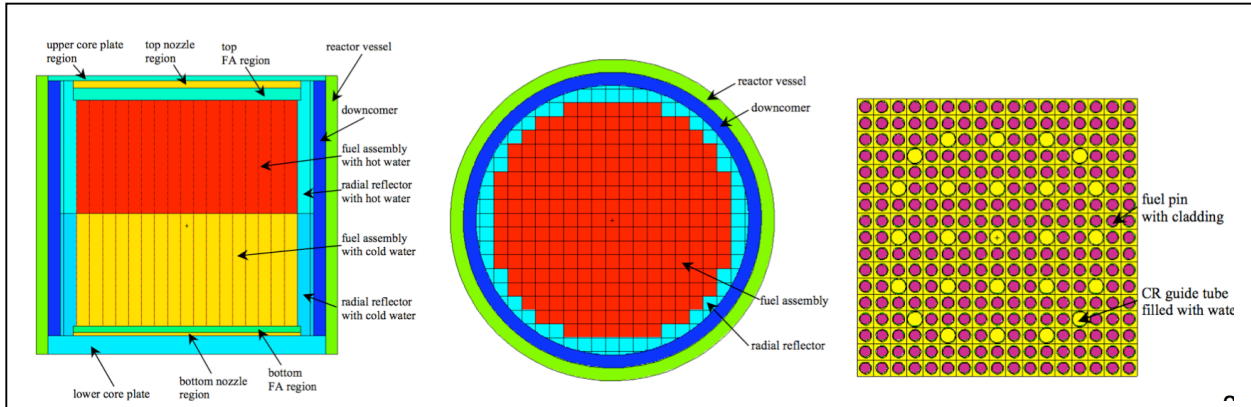


Fig. 6. Kord Smith Model: geometry [9]

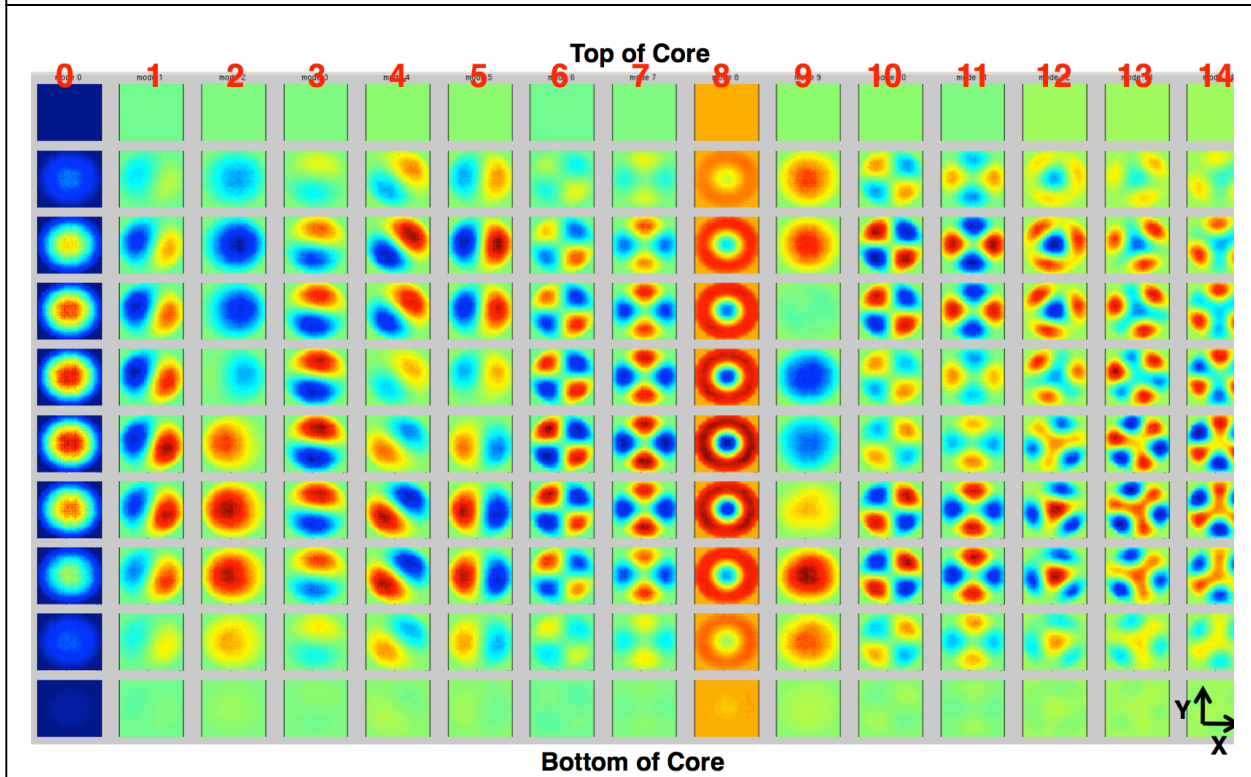


Fig. 7. Kord Smith Model: First 15 eigenfunctions of the fission neutron source distribution, x-y slices through the core at 10 elevations.

Table I. Kord Smith Model: First 15 eigenvalues

n	k_n	n	k_n
0	0.99919	8	0.96043
1	0.98483	9	0.95671
2	0.98362	10	0.95178
3	0.98469	11	0.95078
4	0.96956	12	0.94524
5	0.96950	13	0.94497
6	0.96693	14	0.94472
7	0.96591	-	-

3.3. Fuel Storage Vault

This is Benchmark Problem 1 from the OECD/NEA Source Convergence Benchmarks [10]. The problem contains 36 large, loosely coupled spent fuel assemblies in water surrounded by concrete reflector. A sole assembly has concrete reflector on two sides, as opposed to one or zero for the others. Consequently, this single assembly is by far the most reactive, with a total fission rate over a factor of 10,000 greater than the least reactive assembly. Conventional Monte Carlo requires around 2000 cycles for fission source convergence with a flat initial guess.

Fig. 8 shows the convergence behavior of the standard MCNP and fission matrix results. The fission matrix is tallied for cycles 3-200, with a batch size of 1 M. The spatial mesh is 96x12x10, corresponding in the x-y plane to sixteen mesh regions for every assembly. By cycle 30, the fission matrix gives a reasonably converged fundamental eigenvector. Thus we see the potential for excellent source convergence acceleration with the fission matrix.

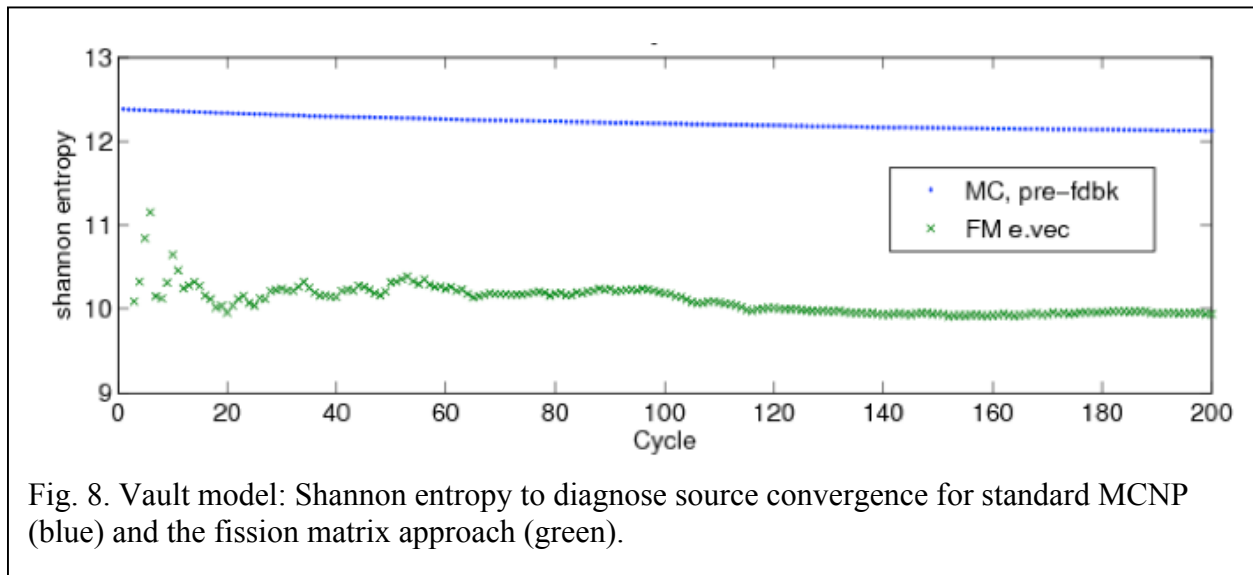


Fig. 9 shows the first 16 eigenfunctions for the fuel storage vault problem, and Table II gives the corresponding eigenvalues.

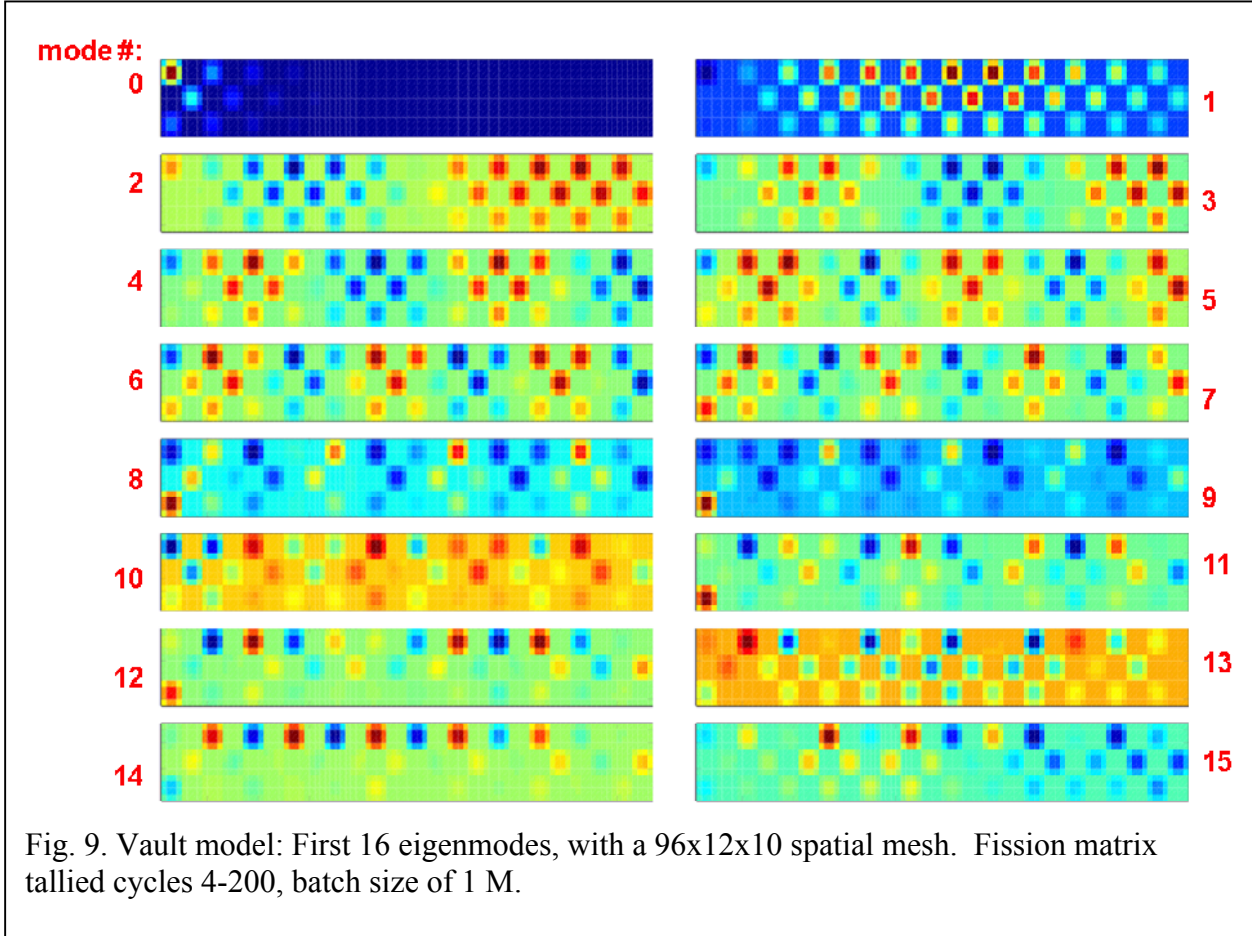


Table II. First 16 eigenvalues for the fuel storage vault problem with a 96x12x10 spatial mesh.

n	k_n	n	k_n
0	0.88947	8	0.87785
1	0.88653	9	0.87658
2	0.88600	10	0.87536
3	0.88533	11	0.87488
4	0.88399	12	0.87363
5	0.88275	13	0.87309
6	0.88112	14	0.87279
7	0.87945	15	0.87245

3.4. Advanced Test Reactor

The final problem examined is the Advance Test Reactor (ATR) at Idaho National Laboratory [11], shown in Fig. 10. Used primarily for the study of radiation effects, this core has a complex serpentine-shape fuel arrangement that does not easily adhere to a Cartesian mesh. There are 40 curved fuel assemblies with 93% enriched uranium aluminide powder fuel; each wraps 45 degrees. Each assembly has 19 plates of thickness 0.2 cm; the actual thickness of the fuel within each plate is 0.05 cm.

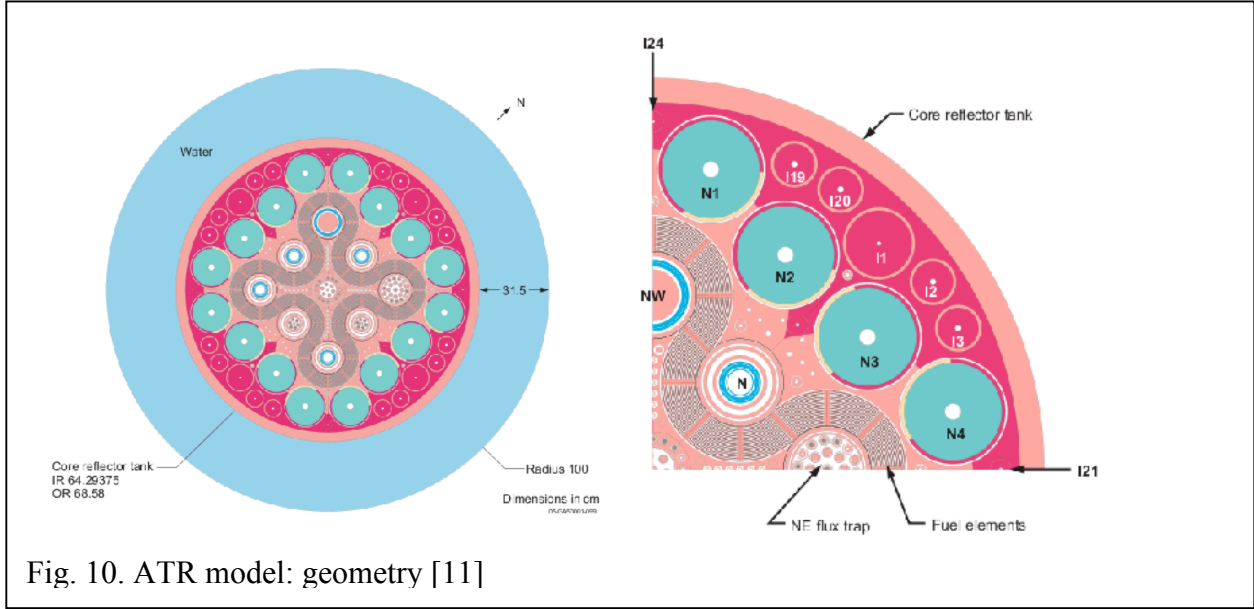


Fig. 10. ATR model: geometry [11]

Fig. 11, the fission matrix structure for a 50x50x1 spatial mesh, shows that all of the fissionable regions are highly-coupled, giving a full, dense matrix. A sparse fission matrix representation is not suitable for this problem.

Fig. 12 shows the first 16 eigenfunctions for the ATR model, using a 100x100x1 spatial mesh for fission matrix tallies. The fission matrix tallies were made for cycles 2-200, with 1M neutrons/cycle.

Table III gives the first 16 eigenvalues for the ATR problem.

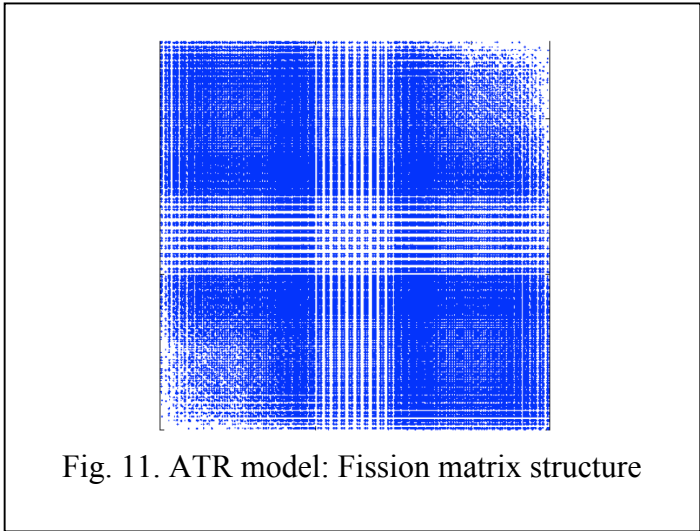
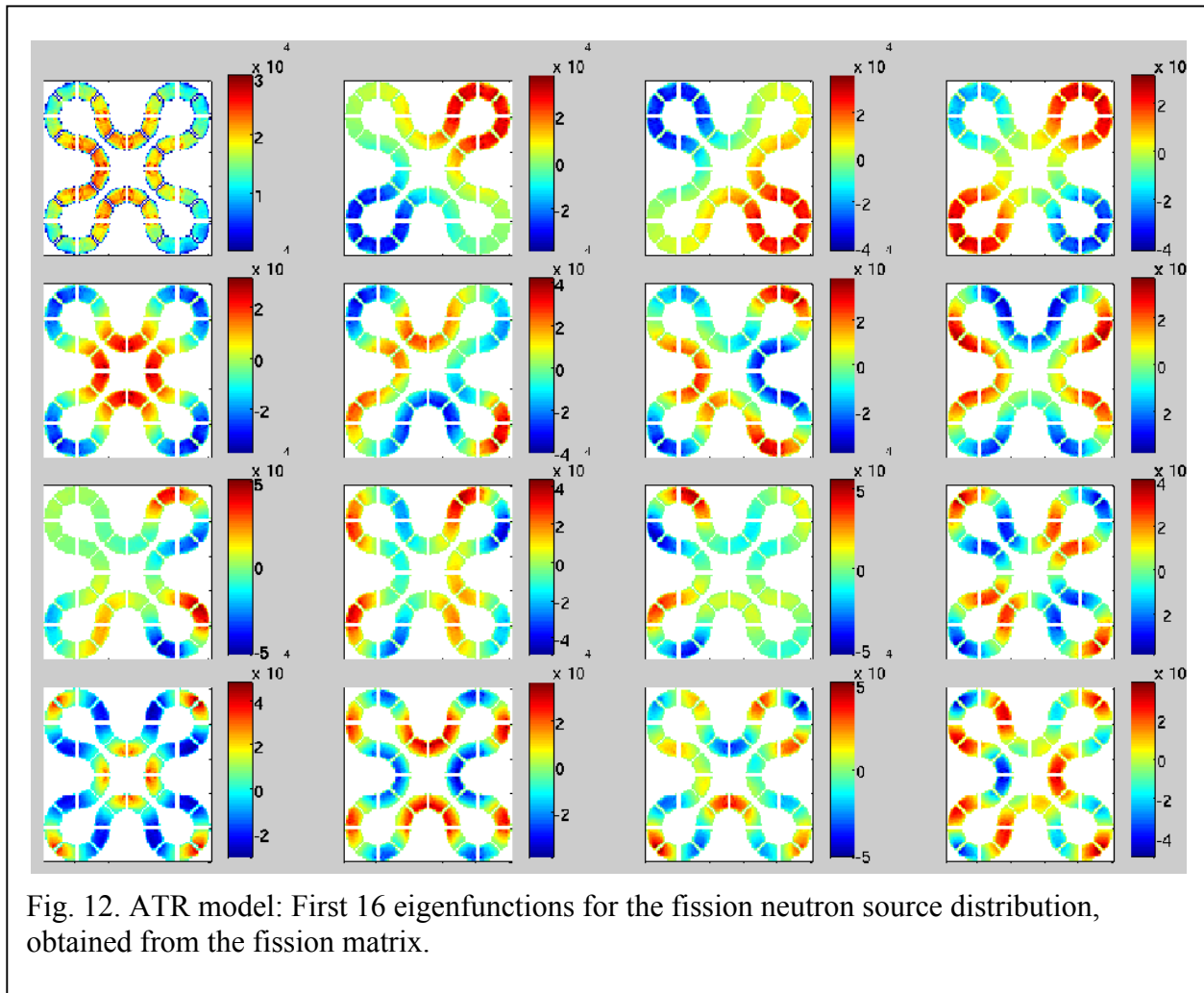


Fig. 11. ATR model: Fission matrix structure

Table III. First 16 eigenvalues for the ATR, with a 100x100x1 mesh.

n	k_n	n	k_n
0	0.99490	8	0.47004
1	0.85630	9	0.46173
2	0.84612	10	0.45794
3	0.78265	11	0.41144
4	0.64564	12	0.32865
5	0.55461	13	0.29454
6	0.55207	14	0.28401
7	0.53659	15	0.28327



4. WORK IN PROGRESS

We are also investigating the use of the fission matrix to accelerate the power method convergence of Monte Carlo criticality calculations. Because the fission matrix can be determined accurately with only a few batches during the inactive portion of the calculation, the fundamental eigenmode can be used to bias the fission neutron source, forcing the source distribution based on Monte Carlo histories to converge more quickly. Initial testing of this method is encouraging, and further study and development are in progress.

5. CONCLUSIONS AND FUTURE WORK

We have described the initial experience and results from implementing a fission matrix capability into the MCNP Monte Carlo code. The fission matrix is obtained at essentially no cost during the normal simulation for criticality calculations. It can be used to provide estimates of the fundamental mode power distribution, the reactor dominance ratio, the eigenvalue spectrum, and higher mode spatial eigenfunctions. It can also be used to accelerate the convergence of the power method iterations. Past difficulties and limitations of the fission matrix approach are overcome for many problems with a new sparse representation of the matrix, permitting much larger and more accurate fission matrix representations.

ACKNOWLEDGMENTS

This work was supported by the US DOE/NNSA Nuclear Criticality Safety Program and by the US DOE/NNSA Advanced Simulation & Computing Program.

REFERENCES

1. X-5 Monte Carlo Team, "MCNP – A General Monte Carlo N-Particle Transport Code, Version 5, Volume I: Overview and Theory," LA-UR-03-1987, *Los Alamos National Laboratory* (2003).
2. K.W. Morton, "Criticality Calculations by Monte Carlo Methods", United Kingdom Atomic Energy Research Establishment, Harwell, Report T/R-1903 (1956).
3. E.L. Kaplan, "Monte Carlo Methods for Equilibrium Solutions in Neutron Multiplication", Lawrence Radiation Laboratory, UCRL-5275-T (1958).
4. J.M. Hammersely & D.C. Handscomb, *Monte Carlo Methods*, Chapter 8, Methuen & Co. (1964).
5. F.B. Brown, S.E. Carney, B.C. Kiedrowski, W.R. Martin, Fission Matrix Capability for MCNP, Part I - Theory", this conference (2013).
6. M.B. Chadwick, et al., "ENDF/B-VII.0: Next Generation Evaluated Nuclear Data Library for Nuclear Science and Technology", *Nuclear Data Sheets* **107**, 2931–3060 (2006).
7. F.B. Brown, "A Review of Best Practices for Monte Carlo Criticality Calculations", ANS NCSD-2009, Richland, WA, Sept 13-17 (2009).
8. M. Nakagawa & T. Mori, "Whole Core Calculations of Power Reactors by use of Monte Carlo Method", *J. Nuc. Sci. and Tech.*, **30** [7], pp 692-701 (1993).
9. J.E. Hogenboom, W.R. Martin, B. Petrovic, "Monte Carlo Performance Benchmark for Detailed Power Density Calculation in a Full Size Reactor Core", *OECD/NEA* (2010).
10. F. B. Brown, R. C. Little, A. Sood, D. K. Parsons, "MCNP Calculations for the OECD/NEA Source Convergence Benchmarks", *Trans. Am. Nucl. Soc.*, **87**, 150 (2003).
11. S. S. Kim, B. G. Schnitzler, et. al., "Serpentine Arrangement of Highly Enrichment Water-Moderated Uranium-Aluminide Fuel Plates Reflected by Beryllium", Idaho National Laboratory, HEU-MET-THERM-022, (2005).