Multi-stage Response-function Transport (MRT) Methodologies for Real-Time Calculations

Prof. Alireza Haghighat
Virginia Tech
Virginia Tech Transport Theory Group (VT³G)
Director of Nuclear Engineering and Science Lab (NSEL) at Arlington Nuclear Engineering Program, Mechanical Engineering Department

“Global Leaders’ Symposium on Reactor Physics”, In honor of the retirement of Prof. Nam Zin Cho, Korean Advanced Institute for Science and Technology (KAIST), Daejeon, Korea, Dec. 8, 2014
This lecture is dedicated to Prof. Nam Zin Cho
It is my honor and pleasure to participate at the retirement celebration for my dear friend and colleague, Prof. Nam Zin Cho.

Nam Zin you have made major contributions to nuclear science and engineering discipline and nuclear industry by training a large number of highly qualified and accomplished professionals who are making important contributions to nuclear industry in Korea and beyond, and by developing advanced computational methods in reactor physic and particle transport.
Objective

Determine the expected number of particles in a phase space \((d^3rdEd\Omega)\) at time \(t\):

\[
n(\vec{r}, E, \hat{\Omega}, t)d^3rdEd\Omega
\]

Number density is used to determine angular flux/current, scalar flux and current density, partial currents, and reaction rates.
Simulation Approaches

• **Deterministic Methods**
  - Solve the linear Boltzmann equation to obtain the expected flux in a phase space

• **Statistical Monte Carlo Methods**
  - Perform particle transport experiments using random numbers (RN’s) on a computer to estimate average properties of a particle in phase space
Deterministic – Linear Boltzmann Equation

• Integro-differential form

\[ \hat{\Omega}.\nabla \Psi(\vec{r}, E, \hat{\Omega}) + \sigma(\vec{r}, E) \Psi(\vec{r}, E, \hat{\Omega}) = \]
\[ \int_{0}^{\infty} dE' \int_{0}^{4\pi} d\Omega' \sigma_{s}(\vec{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) \Psi(\vec{r}, E', \hat{\Omega}) + \]
\[ \frac{\chi(E)}{4\pi} \int_{0}^{\infty} dE' \int_{0}^{4\pi} d\Omega' \nu \sigma_{f}(\vec{r}, E') \Psi(\vec{r}, E', \hat{\Omega}) + S(\vec{r}, E, \hat{\Omega}) \]

• Integral form

\[ \psi(\vec{r}, E, \hat{\Omega}) = \int_{0}^{R} d |\vec{r} - \vec{r}'| Q(\vec{r}') e^{-\tau_{E}(\vec{r}, \vec{r}')} + \psi(\vec{r}_{s}, E, \hat{\Omega}) e^{-\tau_{E}(\vec{r}, \vec{r}_{s})} \]
Integro-differential - Solution Method

• **Angular variable: Discrete Ordinates (Sn) method:**
  A discrete set of directions \{\hat{\Omega}_m\} and associated weights \{w_m\} are selected
  \[\hat{\Omega}_m \cdot \nabla \Psi(\vec{r}, E, \hat{\Omega}_m) + \sigma(\vec{r}, E) \Psi(\vec{r}, E, \hat{\Omega}_m) = q(\vec{r}, E, \hat{\Omega}_m)\]

• **Spatial variable**
  Integrated over fine meshes using FD or FE methods
  \[\Psi_{m,g,A} = \frac{\int d^3 r \Psi_{m,g}(\vec{r}) \Delta V_{ijk}}{\Delta V_{ijk}}\]

• **Energy variable**
  Integrate over energy intervals to prepare multigroup cross sections, \(\sigma_g\)
Integral - Solution method

- **Method of Characteristic (MOC):** Model is partitioned into coarse meshes and transport equation is solved along the characteristic paths \((k)\) (parallel to each discrete ordinate \((n)\)), filling the mesh, and averaged.

\[
\psi_{g,m,i,k} (t_{m,i,k}) = \psi_{g,m,i,k} (0) \exp(-\sigma_{g,i} t_{m,i,k}) + \frac{Q_{g,m,i}}{\sigma_{g,i}} (1 - \exp(-\sigma_{g,i} t_{m,i,k}))
\]
Deterministic - Issues/Challenges/Needs

- Robust **numerical formulations** (e.g., adaptive differencing strategy)

- Algorithms for improving **efficiency** (i.e., acceleration techniques – synthetic formulations and pre-conditioners)

- Use of advanced computing **hardware & software** environments

- Pre- and post-processing **tools**

- Multigroup **cross section** preparation

- **Benchmarking**
Monte Carlo Methods

- Perform an experiment on a computer; “exact” simulation of a physical process

\[
\begin{align*}
    r &= -\ln \xi / \Sigma_t \\
    \xi &\leq \frac{\Sigma_s}{\Sigma_t} \\
    \mu_0 &= 2\xi - 1
\end{align*}
\]

Path-length

Type of collision

Scattering angle (isotropic scattering)

Sample

\( S (r, E, \Omega) \)

Tally (count)

absorbed

Issue:

Precise expected values; i.e., small relative error, \( R_x = \frac{\sigma_x}{\bar{x}} \)

Variance Reduction techniques are needed for real-world problems!
<table>
<thead>
<tr>
<th>Item</th>
<th>Deterministic</th>
<th>MC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geometry</td>
<td>Discrete/ Exact</td>
<td>Exact</td>
</tr>
<tr>
<td>Energy treatment – cross section</td>
<td>Discrete</td>
<td>Exact</td>
</tr>
<tr>
<td>Direction</td>
<td>Discrete/ Truncated series</td>
<td>Exact</td>
</tr>
<tr>
<td>Input preparation</td>
<td>Difficult</td>
<td>simple</td>
</tr>
<tr>
<td>Computer memory</td>
<td>Large</td>
<td>Small</td>
</tr>
<tr>
<td>Computer time</td>
<td>Small</td>
<td>Large</td>
</tr>
<tr>
<td>Numerical issues</td>
<td>Convergence</td>
<td>Statistical uncertainty</td>
</tr>
<tr>
<td>Amount of information</td>
<td>Large</td>
<td>Limited</td>
</tr>
<tr>
<td>Parallel computing</td>
<td>Complex</td>
<td>Trivial</td>
</tr>
</tbody>
</table>
Why not MC only?

• Because of the difficulty in obtaining detail information with reliable statistical uncertainty in a reasonable time

• Example situations
  • Real-time simulations
  • Obtaining energy-dependent flux distributions,
  • Time-dependent simulations,
  • Sensitivity analysis,
  • Determination of uncertainties
Why not hybrid methods?

Deterministic-deterministic (differencing schemes, different numerical formulations, generation of multigroup cross sections, generation of angular quadratures, acceleration techniques)

Monte Carlo-deterministic (variance reduction with the of use deterministic adjoint)
• Vector computing of 1-D Sn spherical geometry algorithm
• Development an adjoint methodology for simulation TMI-2 reactor

• Vector and parallel processing of 2-D Sn algorithms
• Simulation of Reactor Pressure Vessel (RPV)

• Parallel processing of 2-D Sn algorithms & Acceleration methods
• Determination of uncertainties in the RPV transport calculations

• 3-D parallel Sn Cartesian algorithms
• Monte Carlo for Reactor Pressure Vessel (RPV) benchmark using Weight-window generator; deterministic benchmarking of power reactors

• Directional Theta Weight (DTW) differencing formulation
• PENTRAN (Parallel Environment Neutral Particle TRANsport) code system
• CADIS (Consistent Adjoint Driven Importance Sampling) formulation for Monte Carlo Variance Reduction
• A$^3$MCNP (Automated Adjoint Accelerate MCNP)

• Parallel Angular & Spatial Multigrid acceleration methods for Sn transport
• Hybrid algorithm for PGNNA device
• PENMSH & PENINP for mesh and input generation of PENTRAN
• Ordinate Splitting (OS) technique for modeling a x-ray CT machine

• Simplified Sn Even Parity (SSn-EP) algorithm for acceleration of the Sn method
• RAR (Regional Angular Refinement) formulation
• Pn-Tn angular quadrature set
• FAST (Flux Acceleration Simplified Transport)
• PENXMSH, An AutoCad driven PENMSH with automated meshing and parallel decomposition
• CPXSD (Contribution Point-wise cross-section Driven) for generation of multigroup libraries

• TITAN hybrid parallel transport code system & a new version of PENMSH called PENMSHXP
• ADIES (Angular-dependent Adjoint Driven Electron-photon Importance Sampling) code system

• INSPECT-S (Inspection of Nuclear Spent fuel-Pool Calculation Tool ver. Spreadsheet), a MRT algorithm
• TITAN fictitious quadrature set and ray-tracing for SPECT (Single Photon Emission Computed Tomography)
• FMBMC-ICEU (Fission Matrix Based Monte Carlo with Initial source and Controlled Elements and Uncertainties)

• New WCOS (Weighted Circular Ordinated Splitting) Technique for the TITAN SPECT Formulation
• Adaptive Collision Source (ACS) for Sn transport
• AIMS (Active Interrogation for Monitoring Special-nuclear-materials), a MRT algorithm

• TITAN-SDM includes Subgroup Decomposition Method for multigroup transport calculation
• Deterministic iterative image Reconstruction algorithm for SPECT (DR-SPECT)(ongoing)
• Real-Time Pool Simulation (RTPS) tool (ongoing)
Development of Transport Formulations for Real-Time Applications

• Even parallel, “fast” hybrid transport calculations are slow for real-time applications

• Develop a **Multi-stage, Response-function Transport (MRT) algorithm**
  
  • It is necessary to **partition** a problem into **stages** (sub-problems),
    
    • For each stage employ response method and/or adjoint function methodology
    
    • Pre-calculate response-function or adjoint-function using an accurate and fast transport code
    
    • Solve a linear system of equations to couple all the stages
Examples for **MRT Algorithms**

- **Nondestructive testing**: Optimization of the Westinghouse’s PGNNA active interrogation system for detection of RCRA (Resource Conversation and Recovery Act) (e.g., lead, mercury, cadmium) in waste drums (partial implementation of MRT; 1999)

- **Nuclear Safeguards**: Monitoring of spent fuel pools for detection of fuel diversion (2010)

- **Nuclear nonproliferation**: Active interrogation of cargo containers for simulation of special nuclear materials (SNMs) (2013) (in collaboration with GaTech)

- **Spent fuel safety and security**: Real-Time Pool Simulation (RTPS) for determination of eigenvalue, subcritical multiplication, and material identification (partly funded by I²S project, led by GaTech) (Ongoing)
Nondestructive Testing via Active Interrogation - Optimization of Pulsed Gamma Neutron Activation Analysis (PGNAA) device

Stage 1 - Determined the thermal neutron flux distribution throughout the waste using a time-dependent MCNP Monte Carlo calculation

Stage 2 - Determined the gamma flux at the face of a gamma detector using an “importance function” obtained from a 3-D PENTRAN deterministic calculation.

Achieved excellent agreement with the experimental results (within the experimental uncertainties).
Spent Fuel Pool Inspection (Development of a tool for safeguards) (funded by LLNL)

**Objective** – Identification of missing/moved assemblies for safeguards

**Approach** – On-line combination (via statistical minimization) of measured and computed detector responses to identify possible fuel diversion.
Need?

• Develop a fast and accurate computation tool which can estimate the detector response for various combinations of
  • Burnup
  • Cooling time
  • Pool lattice arrangement
  • Fuel type (enrichment)
How do we calculate the detector response?

• Standard or “forward” approach

\[
R = \langle \sigma_d \psi \rangle = \int_{V_d} dV \int_0^\infty dE \int_{4\pi} d\Omega \sigma_d(\vec{r}, E) \psi(\vec{r}, E, \Omega)
\]

• “Adjoint” approach

\[
R = \langle S \psi^+ \rangle
\]

Where,
• \( S \) is particle source, and
• \( \psi^+ \) is adjoint (“importance”) function
Demonstration

**Standard**

\[ R = \langle \sigma_d \phi \rangle \]

Where,

\[ H\phi = S \]

**Adjoint Methodology**

\[ R = \langle S\phi^+ \rangle \]

Where,

\[ H^+\phi^+ = \sigma_d \]
Development of INSPCT-S tool – A MRT algorithm

\[ R_n = \langle S_n \phi_n^+ \rangle \]

**Source** \((S = S_{\text{intrinsic}} + S_{\text{subcritical-Multiplication}})\)

---

**Stage 1. Intrinsic Source**
- Spontaneous fission & \((\alpha, n)\) from fuel burnup calculation (ORIGEN-ARP)
  (Created a database)

---

**Stage 2. Subcritical Multiplication**
- Fission-matrix (FM) method
  - Use MCNP Monte Carlo to obtain \(a_{i,j}\) for each pool type
  (Created a database for coef. \(a_{ij}\))

---

**Adjoint function**

**Stage 3 – Determine adjoint function using the deterministic parallel PENTRAN Sn transport code**
(Created a database for multigroup adjoint for different lattice sizes)
Fission Matrix (FM) Method

• *Eigenvalue* formulation

\[ F_i = \frac{1}{k} \sum_{j=1}^{N} a_{i,j} F_j \]

• Where, \( a \) is a coefficient matrix, \( F \) is fission density, and \( k \) is eigenvalue

• Above formulation provides the amount of fission neutrons generated in one location due to fission neutrons in every other location
Fission Matrix Formulation

• **Subcritical multiplication** formulation

\[ F_i = \sum_{j=1}^{N} (a_{i,j} F_j + b_{i,j} S_j), \]

• We have shown that for this application, we can consider

\[ a_{i,j} \cong b_{i,j} \]
Determination of FM Elements ($a_{i,j}$)

• Three assembly categories
  • Three fixed-source calculations for the three categories of assemblies to determine $a_{i,j}$

• Coupling is highly localized
  • Consider two only rows of assemblies surrounding each assembly

• Geometric similarity
  • For example, green, red and blue arrows indicate the edge assemblies with identical coefficients
Testing the Simplified FM Methodology

• Same $a_{i,j}$ coefficients used for every case
• Solve system of equations
\[ F_i = \sum_{i=1}^{N} a_{i,j} (F_j + S_j^{\text{int.}}) \]

• Four test spent fuel scenarios
  1. 2x6 array, uniform source
  2. 9x6 array, uniform source
  3. 9x6 array, 27 assemblies on the left with source strength 1, the rest with source strength 0.5
  4. 20x6 array, uniform source

• MCNP calculation as a benchmark
FM Results

• Excellent agreement with Monte Carlo (<1%)

<table>
<thead>
<tr>
<th>Assembly Arrangement Case</th>
<th>M (MCNP)</th>
<th>M (Fission Matrix)</th>
<th>Difference</th>
<th>MCNP Uncertainty 1-σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>2x6, uniform</td>
<td>1.7133</td>
<td>1.7104</td>
<td>-0.29%</td>
<td>0.0008</td>
</tr>
<tr>
<td>9x6, uniform</td>
<td>1.9988</td>
<td>1.9966</td>
<td>-0.22%</td>
<td>0.0007</td>
</tr>
<tr>
<td>9x6, non-uniform</td>
<td>2.0033</td>
<td>1.9968</td>
<td>-0.65%</td>
<td>0.0013</td>
</tr>
<tr>
<td>20x6, uniform</td>
<td>2.0513</td>
<td>2.0444</td>
<td>-0.69%</td>
<td>0.0012</td>
</tr>
</tbody>
</table>

• Very fast
  <1s for Fission-matrix method as compared to ~1hr for the standard Monte Carlo
Real-time Tool: INSPCT-S (Inspection of Nuclear Spent fuel-Pool Computing Tool – Spreadsheet)

INSPCT-S solves \[ R_n = \langle S_n \phi^+_n \rangle \] (Use of Fission Matrix & Adjoint)

By interpolation, source and adjoint function are determined using databases of the decay neutrons, fission matrix coefficients, and adjoint distributions.

INPUT

<table>
<thead>
<tr>
<th>src file</th>
<th>C:\Users\ali\Documents\haghD\ufttg\LLNL\INSPCT-s\se.dsrc</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLUMNS</td>
<td>8</td>
</tr>
<tr>
<td>fm file</td>
<td>C:\Users\ali\Documents\haghD\ufttg\LLNL\INSPCT-s\se.dfm</td>
</tr>
<tr>
<td>imp file</td>
<td>C:\Users\ali\Documents\haghD\ufttg\LLNL\INSPCT-s\se.dimp</td>
</tr>
</tbody>
</table>

OUTPUT

<table>
<thead>
<tr>
<th>Detector Normalization</th>
<th>5.28E-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rows</td>
<td>6</td>
</tr>
<tr>
<td>Response Tolerance</td>
<td>15.00%</td>
</tr>
</tbody>
</table>

Burnup

<table>
<thead>
<tr>
<th>(x,y)</th>
<th>1 1 1 1 1 1 1 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>9000</td>
<td>10000</td>
</tr>
<tr>
<td>11000</td>
<td>12000</td>
</tr>
<tr>
<td>13000</td>
<td>14000</td>
</tr>
<tr>
<td>15000</td>
<td>16000</td>
</tr>
</tbody>
</table>

Cooling time

<table>
<thead>
<tr>
<th>(x,y)</th>
<th>1 1 1 1 1 1 1 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2 5 10 15 20 30 40</td>
</tr>
<tr>
<td>2</td>
<td>5 10 15 20 30 40</td>
</tr>
<tr>
<td>3</td>
<td>10 15 20 30 40</td>
</tr>
<tr>
<td>4</td>
<td>15 20 30 40</td>
</tr>
<tr>
<td>5</td>
<td>20 30 40</td>
</tr>
<tr>
<td>6</td>
<td>30 40</td>
</tr>
</tbody>
</table>

Response (experimental)

<table>
<thead>
<tr>
<th>(x,y)</th>
<th>0.5 1.5 2.5 3.5 4.5 5.5 6.5 7.5 8.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.6 0.8 0.9 1.0 1.1 1.2 1.3 1.4 1.5</td>
</tr>
<tr>
<td>1.5</td>
<td>2.0 2.1 2.2 2.3 2.4 2.5 2.6 2.7 2.8</td>
</tr>
<tr>
<td>2.5</td>
<td>3.0 3.1 3.2 3.3 3.4 3.5 3.6 3.7 3.8</td>
</tr>
<tr>
<td>3.5</td>
<td>4.0 4.1 4.2 4.3 4.4 4.5 4.6 4.7 4.8</td>
</tr>
<tr>
<td>4.5</td>
<td>5.0 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8</td>
</tr>
<tr>
<td>5.5</td>
<td>6.0 6.1 6.2 6.3 6.4 6.5 6.6 6.7 6.8</td>
</tr>
<tr>
<td>6.5</td>
<td>7.0 7.1 7.2 7.3 7.4 7.5 7.6 7.7 7.8</td>
</tr>
</tbody>
</table>

Response Difference

<table>
<thead>
<tr>
<th>(x,y)</th>
<th>0.5 1.5 2.5 3.5 4.5 5.5 6.5 7.5 8.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8</td>
</tr>
<tr>
<td>1.5</td>
<td>1.1 1.2 1.3 1.4 1.5 1.6 1.7 1.8 1.9</td>
</tr>
<tr>
<td>2.5</td>
<td>2.1 2.2 2.3 2.4 2.5 2.6 2.7 2.8 2.9</td>
</tr>
<tr>
<td>3.5</td>
<td>3.1 3.2 3.3 3.4 3.5 3.6 3.7 3.8 3.9</td>
</tr>
<tr>
<td>4.5</td>
<td>4.1 4.2 4.3 4.4 4.5 4.6 4.7 4.8 4.9</td>
</tr>
<tr>
<td>5.5</td>
<td>5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9</td>
</tr>
<tr>
<td>6.5</td>
<td>6.1 6.2 6.3 6.4 6.5 6.6 6.7 6.8 6.9</td>
</tr>
</tbody>
</table>

VT Nuclear Science and Engineering Lab (NSEL) at Arlington
Extension to commercial spent fuel pools

Criticality Safety & Safeguards applications
Development of RTPS tool

• Develop a Real-Time spent fuel Pool Simulation (RTPS) tool for
  • Criticality safety
  • Safeguards and verification

• Standard approach - Full Monte Carlo calculations face difficulties in this area
  • Convergence is difficult due to low coupling between regions (due to absorbers)
    • Convergence can also be difficult to detect
  • Computation times are very long, especially to get detailed information
  • Changing pool configuration requires complete recalculation

• Fission Matrix approach – It can address the above issues
  • Fission matrix coefficients are pre-calculated using Monte Carlo
  • Computation times are much shorter, with no convergence issues
  • Detailed fission distributions are obtained at pin level
  • Changing pool assembly configuration does not require new pre-calculations
    • No additional Monte Carlo
Determination of fission Matrix (FM) Coefficients

- **Eigenvalue** formulation

\[ F_i = \frac{1}{k} \sum_{j=1}^{N} a_{i,j} F_j \]

- \( k \) is eigenvalue
- \( F_j \) is fission source, \( S_j \) is fixed source in cell \( j \)
- \( a_{i,j} \) is the number of fission neutrons produced in cell \( i \) due to a fission neutron born in cell \( j \).

- **Subcritical multiplication** formulation

\[ F_i = \sum_{j=1}^{N} \left( a_{i,j} F_j + b_{i,j} S_j \right), \]

- \( b_{i,j} \) is the number of fission neutrons produced in cell \( i \) due to a source neutron born in cell \( j \).
Developed a Multi-stage methodology for determination of FM coefficients

- As the computational cell size, use a single pin
  - $N = 9 \times 9 \times 336 = 27,216$ total fuel pins/fission matrix cells
  - Allows for good accuracy and pin-resolved fission rates

- Standard FM would require $N=27,216$ separate fixed-source calculations to determine the coefficient matrix
  - One calculation for each pin
  - A matrix of size $N \times N = 740,710,656$ total coefficients (6 GB of memory is needed)

- The standard approach is clearly NOT feasible
Developed a Multi-stage methodology for determination of coefficients \((a_{i,j} & b_{i,j})\)

- **Coefficients are calculated at different stages including:**
  - Pin-wise (axial and radial dependent) for one assembly for different burnups, coolants and lattice structures
  - For assemblies in the pool (pin-wise or regional)

**Notes**

- **Reduced the number of calculations**
  - Geometric similarity
  - Geometric symmetry
  - Degree of coupling
  - Sensitivity of the coefficients to different parameters

- **Reduced the amount of memory by indexing**
In addition to subcritical multiplication discussed for the INSPCT-s tool, FM approach can solve for $k$ and fission density (fundamental eigenfunction) using a power-iteration approach as follows:

$$F_i^{(0)} = \frac{1}{N}$$

$$F_i^{(m+1)} = \frac{1}{k^{(m)}} \sum_{j=1}^{N} a_{i,j} F_j^{(m)}$$

Where, $$k^{(m)} = \sum_{i=1}^{N} F_i^{(m)}$$
Test Problems (9x9 assemblies)

(a) Case 1  (b) Case 2  (c) Case 3  (d) Case 4
(e) Case 5  (f) Case 6  (g) Case 7  (h) Case 8

(i) Case 9  (j) Case 10  (k) Case 11  (l) Material Legend

<table>
<thead>
<tr>
<th>Case #</th>
<th>Number of Assemblies</th>
<th>Fuel Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1x1</td>
<td>4.95%</td>
</tr>
<tr>
<td>2</td>
<td>6x1</td>
<td>4.95%</td>
</tr>
<tr>
<td>3</td>
<td>3x3</td>
<td>4.95%</td>
</tr>
<tr>
<td>4</td>
<td>9x9</td>
<td>4.95%</td>
</tr>
<tr>
<td>5</td>
<td>1x1</td>
<td>4.45%</td>
</tr>
<tr>
<td>6</td>
<td>6x1</td>
<td>4.45%</td>
</tr>
<tr>
<td>7</td>
<td>3x3</td>
<td>4.45%</td>
</tr>
<tr>
<td>8</td>
<td>9x9</td>
<td>4.45%</td>
</tr>
<tr>
<td>9</td>
<td>2x1</td>
<td>Mixed</td>
</tr>
<tr>
<td>10</td>
<td>6x1</td>
<td>Mixed</td>
</tr>
<tr>
<td>11</td>
<td>3x3</td>
<td>Mixed</td>
</tr>
</tbody>
</table>
Case 3 Eigenfunction

Reference Solution

Comparison of FM with MC

Fission Source

Assembly number

MCNP

FM
Case 11 Eigenfunction

Comparison with FM with MC

Reference Solution
Case 4 Eigenfunction distribution

Comparison with FM with MC
Comparison of calculated M - FM vs. MCNP

<table>
<thead>
<tr>
<th>Case</th>
<th>FM</th>
<th>MCNP</th>
<th>Error in M (FM vs MCNP)</th>
<th>Speedup (FM vs MCNP)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>M</td>
<td>M</td>
<td>1-σ Uncertainty</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Time (min)</td>
<td>Time (min)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1x1</td>
<td>3.343353</td>
<td>3.33155</td>
<td>0.0010</td>
<td>0.35%</td>
</tr>
<tr>
<td>6x1</td>
<td>4.328244</td>
<td>4.31336</td>
<td>0.0010</td>
<td>0.35%</td>
</tr>
<tr>
<td>3x3</td>
<td>5.428051</td>
<td>5.40992</td>
<td>0.0011</td>
<td>0.35%</td>
</tr>
<tr>
<td>9x9</td>
<td>6.697940</td>
<td>6.67674</td>
<td>0.012</td>
<td>0.32%</td>
</tr>
</tbody>
</table>
RTPS tool

• The current version of the RTPS can quickly and accurately calculate the eigenvalue and eigenfunction for a spent fuel pool.

• By pre-calculating a database of FM coefficients for different conditions, pool simulation can be performed in real-time allowing for changes in configurations (assembly shuffling, removal and addition) for various burnup, cooling times, lattice structure, and enrichments.
Conclusions

• MRT methodology allows for development of real-time tools for analysis of nuclear systems

• Thus far, we have developed INSPCT-S and AIMS software tools for safeguards and nonproliferation applications

• Working on the RTPS tool for accurate and real-time evaluation of commercial spent fuel pools’ safety and security
  • Thus far, the tool provides subcritical multiplication, k and corresponding eigenfunction.
  • Future goal is material identification
Thanks!

Questions?