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

Prof. Alireza Haghighat

Virginia Tech

Virginia Tech Transport Theory Group (VT^{3}G)

Director of Nuclear Engineering and Science Lab (NSEL) at Arlington

Nuclear Engineering Program, Mechanical Engineering Department

Completed June 2011

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

Particle Transport Theory

Objective

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

$$n(\vec{r}, E, \hat{\Omega}, t)d^3rdEd\Omega$$
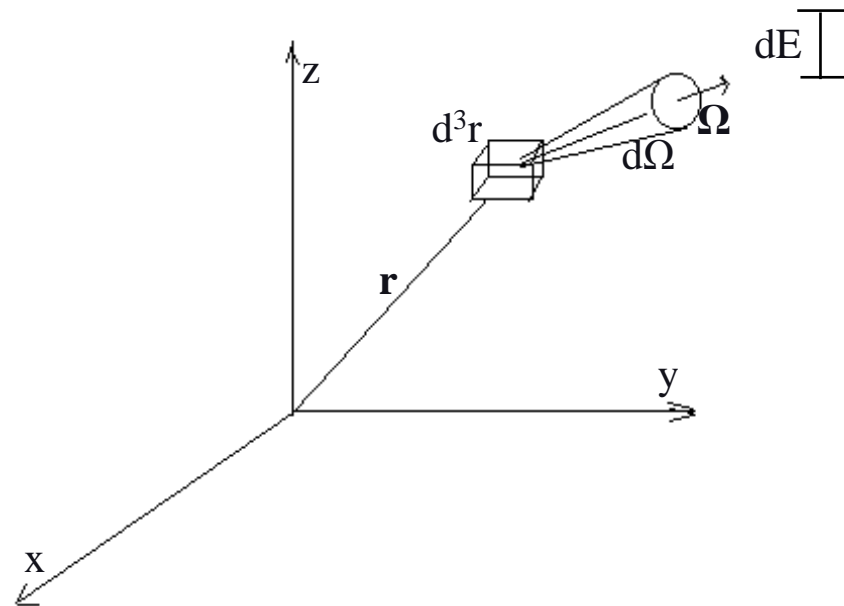
Number density is used to determine <u>angular flux/current</u>, <u>scalar flux and current density</u>, <u>partial currents, and reaction rates</u>.

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 <u>experiments</u> 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**

streaming   collision

$$\hat{\Omega}.\nabla\Psi(\vec{r},E,\hat{\Omega}) + \sigma(\vec{r},E)\Psi(\vec{r},E,\hat{\Omega}) =$$

scattering

$$\int_{0}^{\infty} dE' \int_{4\pi} d\Omega' \sigma_s(\vec{r},E'\rightarrow E,\hat{\Omega}'\rightarrow\hat{\Omega})\Psi(\vec{r},E',\hat{\Omega}) +$$

fission   Independent source

$$\frac{\chi(E)}{4\pi}\int_{0}^{\infty} dE' \int_{4\pi} d\Omega' \upsilon\sigma_f(\vec{r},E')\Psi(\vec{r},E',\hat{\Omega}) + S(\vec{r},E,\hat{\Omega})$$
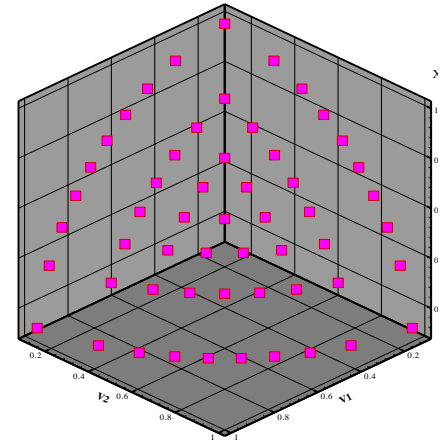
- **Integral form**

$$\psi(\vec{r},E,\hat{\Omega}) = \int_{0}^{R} d\,|\,\vec{r}-\vec{r'}\,|\,Q(r')e^{-\tau_E(\vec{r},\vec{r'})} + \psi(\vec{r}_s,E,\hat{\Omega})e^{-\tau_E(\vec{r},r_s')}$$

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
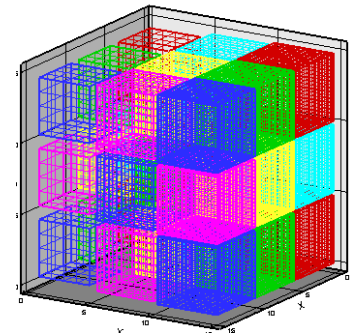
  $$\hat{\Omega}_m . \nabla \Psi(\vec{r}, E, \hat{\Omega}_m) + \sigma(\vec{r}, E) \Psi(\vec{r}, E, \hat{\Omega}_m) = q(\vec{r}, E, \hat{\Omega}_m)$$
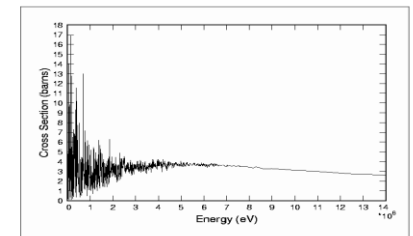
- *Spatial variable*

  Integrated over <u>fine meshes</u> using FD or FE methods

  $$\Psi_{m,g,A} = \frac{\int_{\Delta V_{ijk}} d^3 r \Psi_{m,g}(\vec{r})}{\Delta V_{ijk}}$$
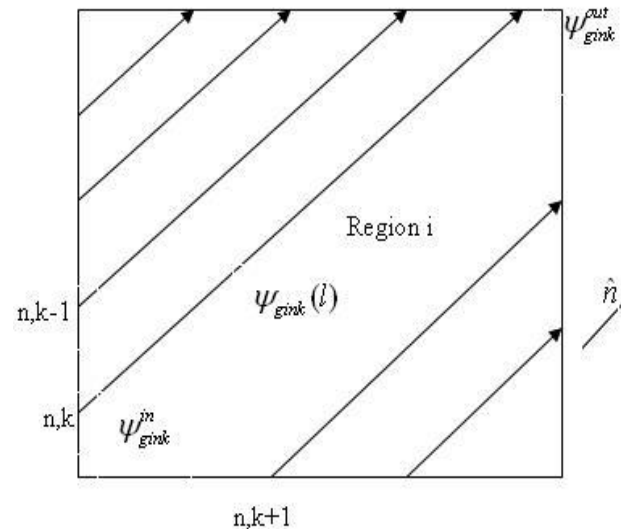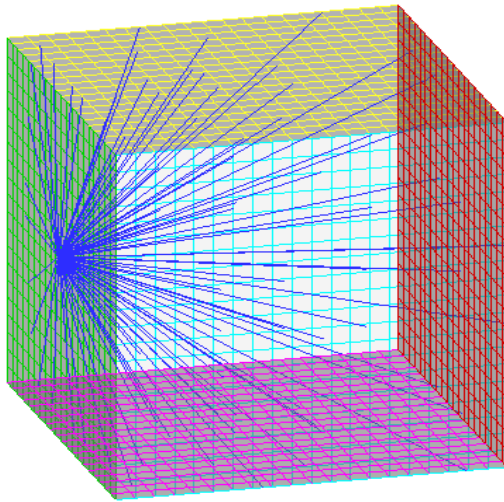
- *Energy variable*

  Integrate over energy intervals to prepare multigroup cross sections, σ_g

Integral - Solution method

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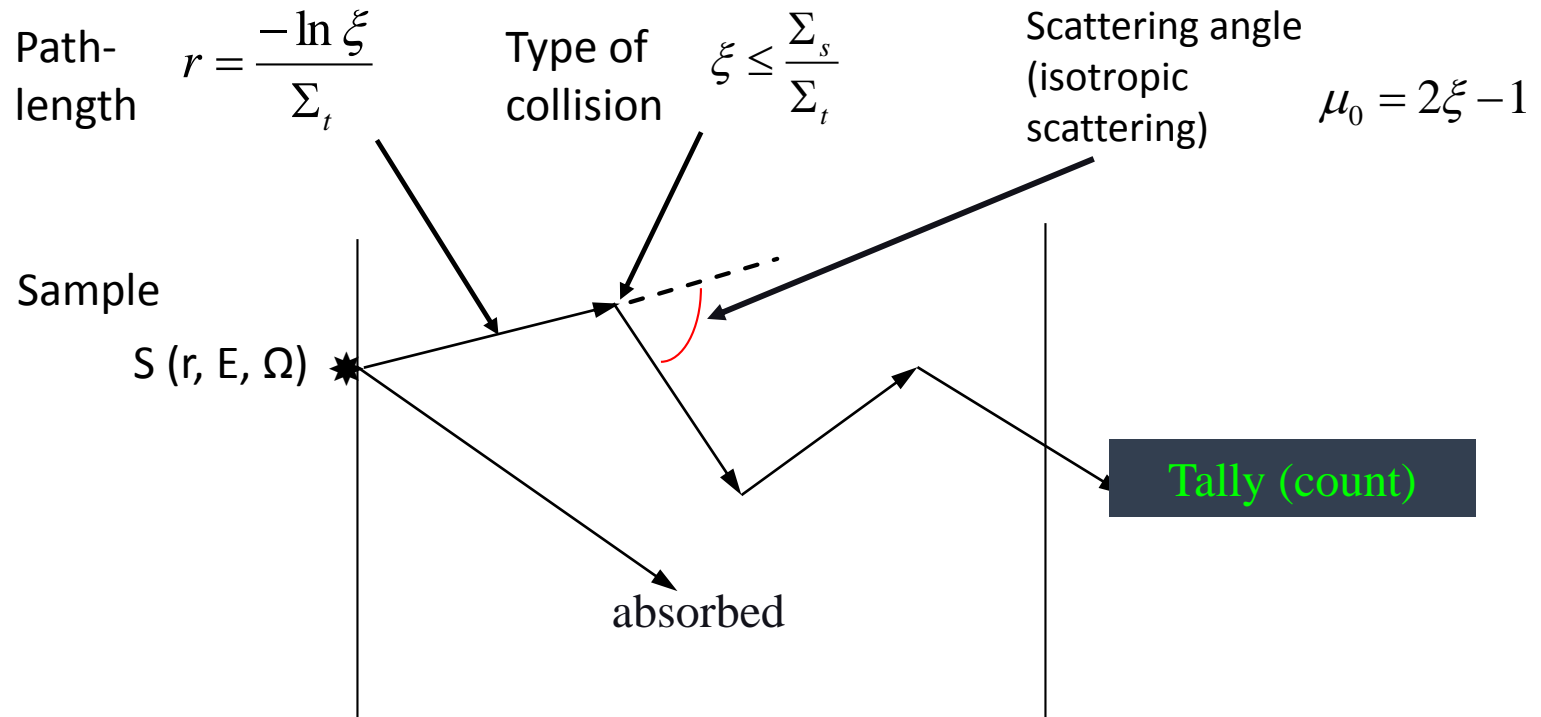

$$\psi_{g,m,i,k}(t_{m,i,k}) = \psi_{g,m,i,k}(0)\exp(-\sigma_{g,i}t_{m,i,k}) + \frac{Q_{g,m,i}}{\sigma_{g,i}}(1-\exp(-\sigma_{g,i}t_{m,i,k}))$$

Deterministic - Issues/Challenges/Needs

- Robust <u>numerical</u> formulations (e.g., adaptive differencing strategy)

- Algorithms for improving <u>efficiency</u> (i.e., acceleration techniques – synthetic formulations and pre-conditioners)

- Use of advanced computing <u>hardware & software</u> environments

- Pre- and post-processing <u>tools</u>

- Multigroup <u>cross section</u> preparation
- <u>Benchmarking</u>

Monte Carlo Methods

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

Path-length $r = \dfrac{-\ln \xi}{\Sigma_t}$

Type of collision $\xi \leq \dfrac{\Sigma_s}{\Sigma_t}$

Scattering angle (isotropic scattering) $\mu_0 = 2\xi - 1$

Sample

S (r, E, Ω)

Tally (count)

absorbed

Issue:

Precise expected values; i.e., small relative error, $R_{\bar{x}} = \dfrac{\sigma_{\bar{x}}}{\bar{x}}$

Variance Reduction techniques are needed for real-world problems!

Deterministic vs. Monte Carlo

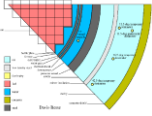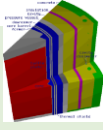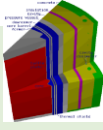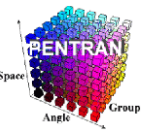| Item	Deterministic	MC
Geometry	Discrete/ Exact	Exact
Energy treatment – cross section	Discrete	Exact
Direction	Discrete/ Truncated series	Exact
Input preparation	Difficult	simple
Computer memory	Large	Small
Computer time	Small	Large
Numerical issues	Convergence	Statistical uncertainty
Amount of information	Large	Limited
Parallel computing	Complex	Trivial
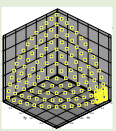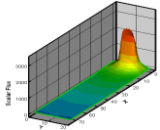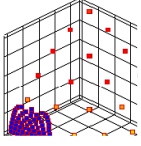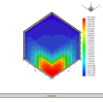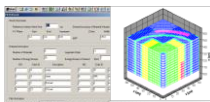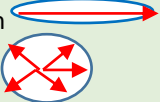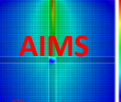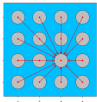
Why not MC only?

- 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)

Years	Milestones	Contributors
1986-1989	• Vector computing of 1-D Sn spherical geometry algorithm • Development an adjoint methodology for simulation TMI-2 reactor	Prof. Haghighat
1989-1992	• Vector and parallel processing of 2-D Sn algorithms • Simulation of Reactor Pressure Vessel (RPV)	Prof. R. Mattis, Pitt. Prof. B. Petrovic, GT
1992-1994	• Parallel processing of 2-D Sn algorithms & Acceleration methods • Determination of uncertainties in the RPV transport calculations	Dr. M. Hunter, W Prof. B. Petrovic, GT
1994-1995	• 3-D parallel Sn Cartesian algorithms • Monte Carlo for Reactor Pressure Vessel (RPV) benchmark using Weight-window generator; deterministic benchmarking of power reactors	Dr. G. Sjoden, DOD Dr. J. Wagner, ORNL
1995-1997	• 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 • A³MCNP (Automated Adjoint Accelerate MCNP)	Dr. B. Petrovic Dr. G. Sjoden, DOD Dr. J. Wagner, ORNL
1997-2001	• 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	Dr. V. Kucukboyaci, W Dr. B. Petrovi, GT Prof. Haghighat Prof. Hgahighat
2001-2004	• 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 (Contributon Point-wise cross-section Driven) for generation of multigroup libraries	Dr. G. Longonil, PNNL Dr. A. Patchimpattapong, IAEA Dr. A. Alpan, W
2004-2007	• TITAN hybrid parallel transport code system & a new version of PENMSH called PENMSHXP • ADIES (Angular-dependent Adjoint Driven Electron-photon Importance Sampling) code system	Dr. C. Yi, GT Dr. B. Dionne, ANL
2007-2011	• INSPCT-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)	W. Walters, PhD Cand. Dr. C. Yi, GT Dr. M. Wenner, W
2011-2013	• 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	K. Royston, PhD Cand. W. Walters, PhD Cand.
2014-2015	• 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)	N. Roskoff, PhD Stud. K. Royston, PhD Cand. W. Walters, PhD Cand.

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.
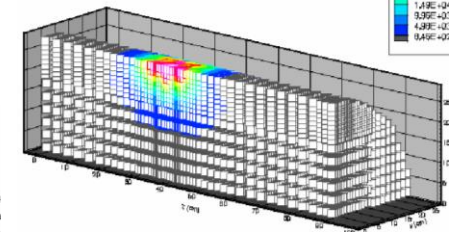
Achieved excellent agreement with the experimental results (within the experimental uncertainties).
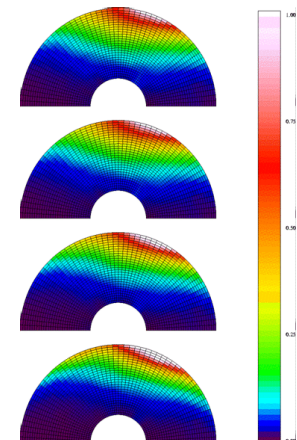
Importance Function

Time-sequenced MCA
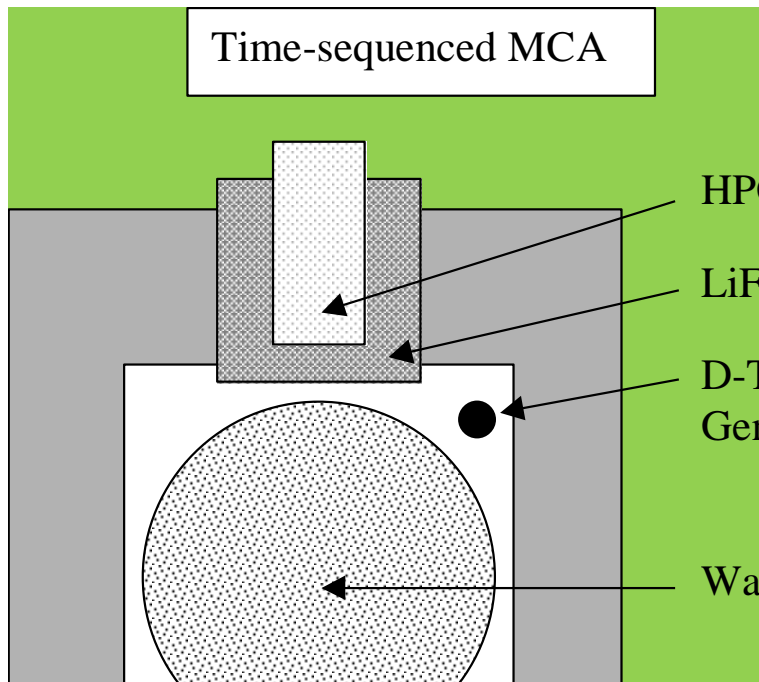
Mesh distribution

HP

LiF

D-T
Gen

Wa
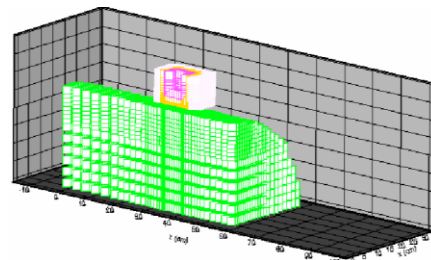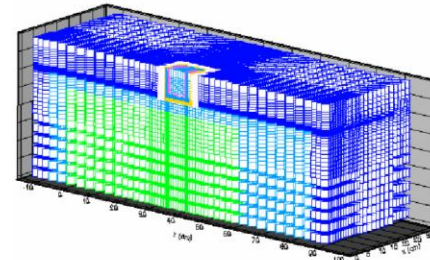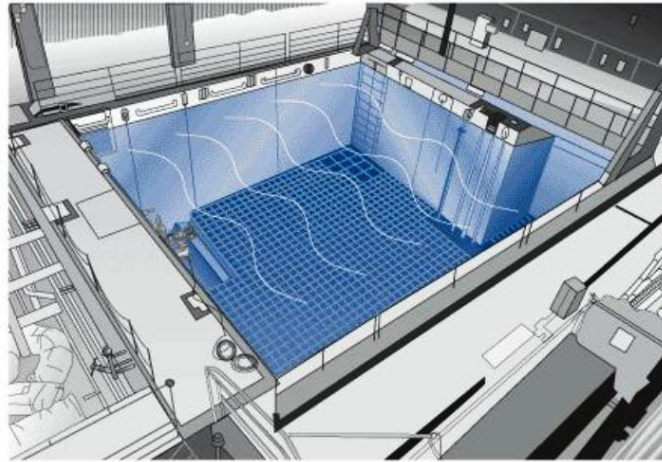
Detector Contribution

Spent Fuel Pool Inspection
(Development of a tool for safeguards)
(funded by LLNL)

Objective – Identification of missing/moved assemblies for safeguards

Approach – <u>On-line</u> 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

$$R = <\sigma_d \psi> = \int_{V_d} dV \int_0^\infty dE \int_{4\pi} d\Omega \; \sigma_d(\vec{r}, E) \psi(\vec{r}, E, \hat{\Omega})$$

- "Adjoint" approach

$$R = <S\psi^+>$$

Where,
- S is particle source, and
- ψ^+ is adjoint ("importance") function

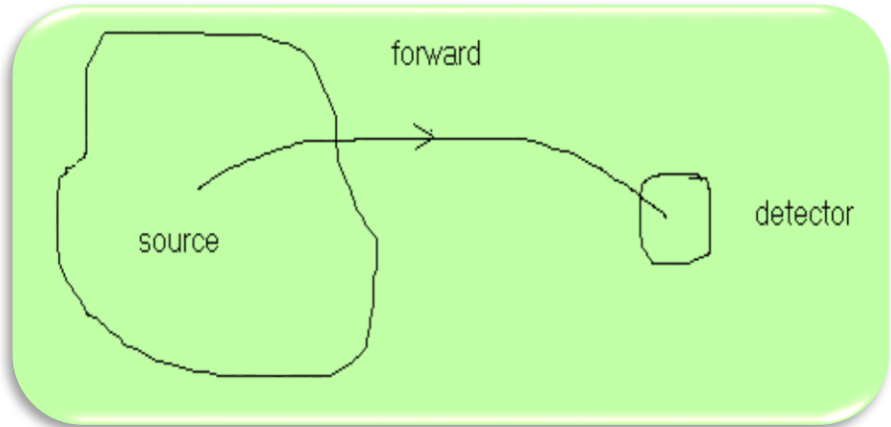# Demonstration

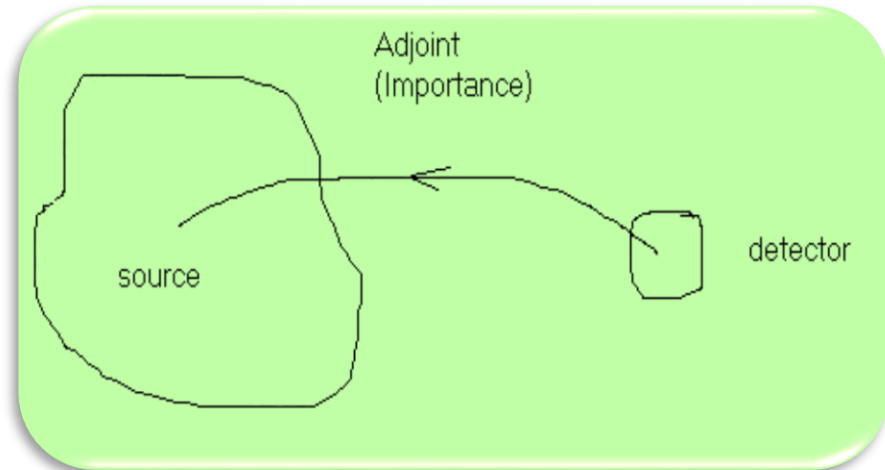| Standard |
|---|

$$R = <\sigma_d \phi>$$

Where,

$$H\phi = S$$

Adjoint Methodology

$$R = <S\phi^+>$$

Where,

$$H^+\phi^+ = \sigma_d$$

Development of INSPCT-S tool – A MRT algorithm

$$R_n = <S_n \phi_n^+>$$

Source ($S = S_{intrinsic} + S_{subcritical-Multiplication}$)

Stage 1. Intrinsic Source
- Spontaneous fission & (α, n) from fuel burnup calculation (ORIGEN-ARP)

(Created a database)

Multiplication Source by Assembly (Fission Matrix)

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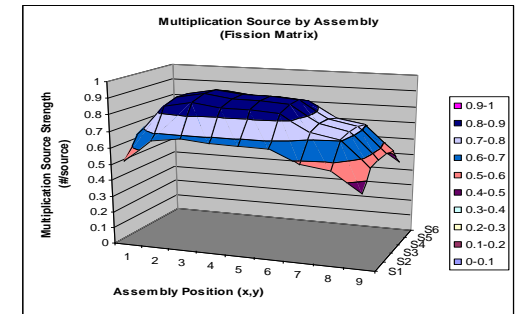- **Fission-matrix (FM) method**
  - **Use MCNP Monte Carlo to obtain $a_{i,j}$ for each pool type**

(Created a database for coef. a_{ij})
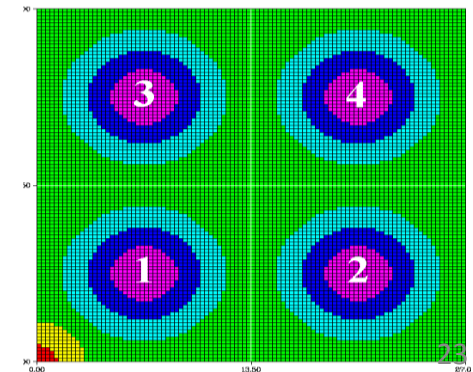
$$F_i = \sum_{i=1}^{N} a_{i,j}(F_j + S_j^{int.})$$

Adjoint function

Stage 3 – Determine adjoint function using the deterministic parallel PENTRAN Sn transport code

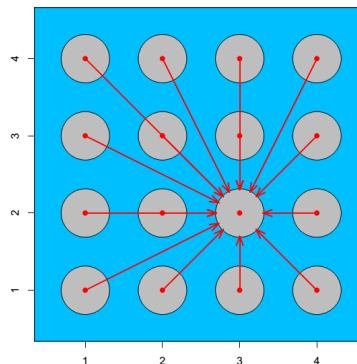(Created a database for multigroup adjoint for different lattice sizes)

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

6x9 array of assemblies

Corner Assemblies

**Edge Assemblies**

Interior Assemblies
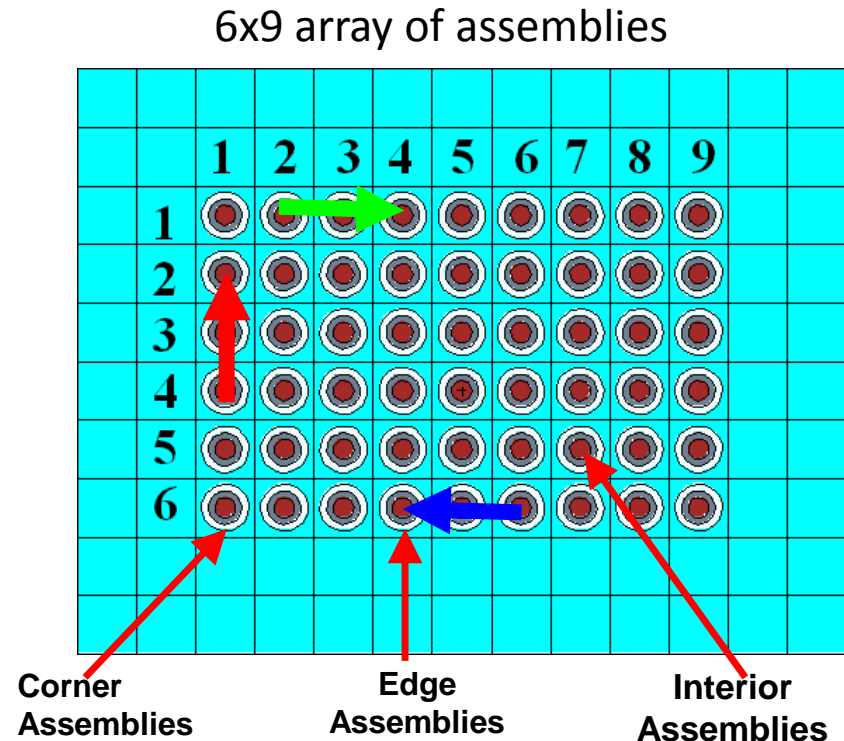
Testing the Simplified FM Methodology

- 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
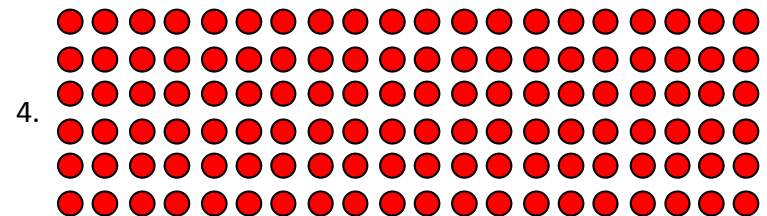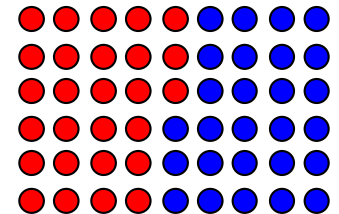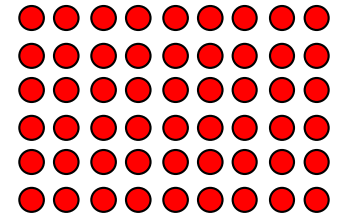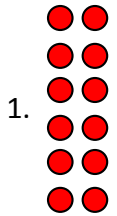
- MCNP calculation as a benchmark

1.

2.

3.

4.

FM Results

- Excellent agreement with Monte Carlo (<1%)

Assembly Arrangement Case	M (MCNP)	M (Fission Matrix)	Difference	MCNP Uncertainty 1-σ
2x6, uniform	1.7133	1.7104	-0. 29%	0.0008
9x6, uniform	1.9988	1.9966	-0. 22%	0.0007
9x6, non-uniform	2.0033	1.9968	-0.65%	0.0013
20x6, uniform	2.0513	2.0444	-0. 69%	0.0012

- 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 $R_n = <S_n \phi_n^+>$ (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

		src file	C:\Users\ali\Documents\haghD\ufttg\LLNL\INSPCT-s\se.dsrc			
COLUMNS	8	fm file	C:\Users\ali\Documents\haghD\ufttg\LLNL\INSPCT-s\se	Response Tolerance	
ROWS	6	imp file	C:\Users\ali\Documents\haghD\ufttg\LLNL\INSPCT-s\se	15.00%	

Burnup

(x,y)	1	2	3	4	5	6	7	8
1	9000	9000	9000	9000	9000	9000	9000	9000
2	10000	10000	10000	10000	10000	10000	10000	10000
3	11000	11000	11000	11000	11000	11000	11000	11000
4	12000	12000	12000	12000	12000	12000	12000	12000
5	13000	13000	13000	13000	13000	13000	13000	13000
6	14000	14000	14000	14000	14000	14000	14000	14000

Cooling time

(x,y)	1	2	3	4	5	6	7	8
1	1	2	5	10	15	20	30	40
2	1	2	5	10	15	20	30	40
3	1	2	5	10	15	20	30	40
4	1	2	5	10	15	20	30	40
5	1	2	5	10	15	20	30	40
6	1	2	5	10	15	20	30	40

Response (experimental)

(x,y)	0.5	1.5	2.5	3.5	4.5	5.5	6.5	7.5	8.5
0.5									
1.5		0.6				0.3			
2.5				0.8					
3.5									
4.5				1.4					
5.5						1.2			
6.5									

OUTPUT

Detector Normalization 5.28E-10   run

Independent Source

(x,y)	1	2	3	4	5	6	7	8
1	4.60E+07	3.39E+07	2.84E+07	2.48E+07	2.21E+07	1.94E+07	1.56E+07	13036948
2	6.89E+07	5.30E+07	4.49E+07	3.86E+07	3.39E+07	2.91E+07	2.26E+07	18101692
3	1.00E+08	8.04E+07	6.86E+07	5.84E+07	5.06E+07	4.29E+07	3.23E+07	25047256
4	1.42E+08	1.17E+08	1.01E+08	8.51E+07	7.33E+07	6.15E+07	4.53E+07	34204842
5	1.98E+08	1.67E+08	1.45E+08	1.22E+08	1.04E+08	8.67E+07	6.28E+07	46492994
6	2.68E+08	2.32E+08	2.01E+08	1.69E+08	1.44E+08	1.19E+08	8.52E+07	62072007

Fission Source

(x,y)	1	2	3	4	5	6	7	8
1	4.03E+07	4.68E+07	4.25E+07	3.66E+07	3.11E+07	2.57E+07	1.98E+07	12521188
2	6.88E+07	8.12E+07	7.41E+07	6.34E+07	5.32E+07	4.33E+07	3.26E+07	20199639
3	9.82E+07	1.17E+08	1.07E+08	9.08E+07	7.54E+07	6.05E+07	4.47E+07	27169878
4	1.32E+08	1.58E+08	1.44E+08	1.21E+08	9.98E+07	7.93E+07	5.78E+07	34751134
5	1.62E+08	1.92E+08	1.73E+08	1.45E+08	1.19E+08	9.42E+07	6.80E+07	40823941
6	1.49E+08	1.74E+08	1.56E+08	1.30E+08	1.06E+08	8.38E+07	6.03E+07	36229288

Response(Calculated)

(x,y)	0.5	1.5	2.5	3.5	4.5	5.5	6.5	7.5	8.5
0.5	0.123198	0.230998	0.221266	0.193583	0.166627	0.141523	0.114534	0.083892	0.036611
1.5	0.305453	0.580498	0.561644	0.491674	0.420538	0.353999	0.28285	0.203871	0.087599
2.5	0.467647	0.897903	0.880437	0.770597	0.653747	0.543993	0.427576	0.301569	0.127819
3.5	0.658686	1.271323	1.252983	1.094413	0.922518	0.761393	0.591298	0.410909	0.172554
4.5	0.879337	1.696988	1.669344	1.453015	1.219392	1.002015	0.772365	0.532245	0.222616
5.5	1.029258	1.978009	1.923356	1.665877	1.394836	1.14574	0.880125	0.605581	0.253192
6.5	0.57336	1.093457	1.058167	0.914234	0.765146	0.628852	0.482792	0.332139	0.139652

Response Difference

(x,y)	0.5	1.5	2.5	3.5	4.5	5.5	6.5	7.5	8.5
0.5									
1.5		3.36%				-15.25%			
2.5				3.82%					
3.5									
4.5				-3.65%					
5.5						4.74%			
6.5									

29

Extension to commercial spent fuel pools

Criticality Safety & Safeguards applictions

# 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.
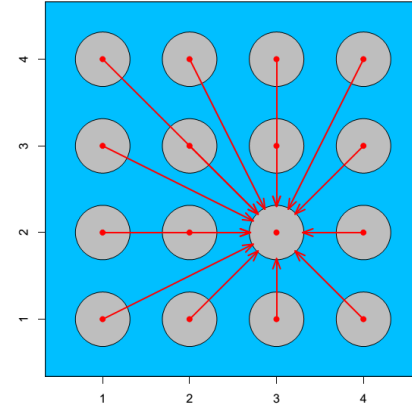
- **Subcritical multiplication** formulation

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

  - $b_{i,j}$ 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 x N = 740,710,656 total coefficients (6 GB of memory is needed)

- The standard approach is clearly NOT feasible

9x9 segment of spent fuel pool
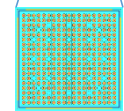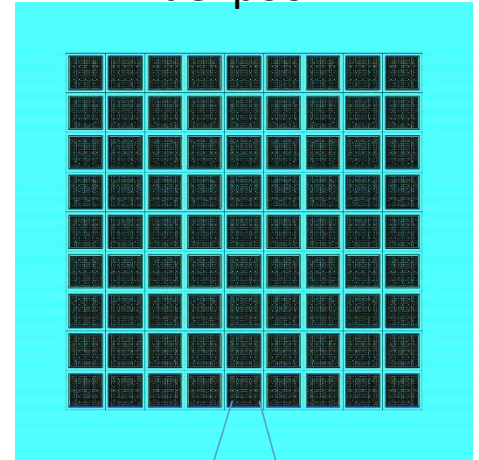
Assembly with 19x19 fuel pins

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**

FM approach for eigenvalue problems

- In addition to subcritical multiplication discussed for the INSPCT-s tool, FM approach can
    - Solve for k and fission density (fundamental eigenfuction) 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, $k^{(m)} = \sum_{i=1}^{N} F_i^{(m)}$
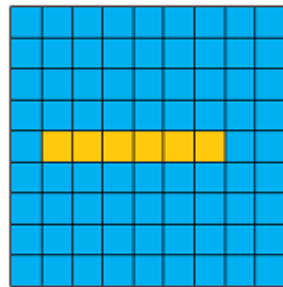
Test Problems (9x9 assemblies)
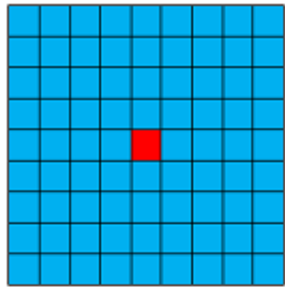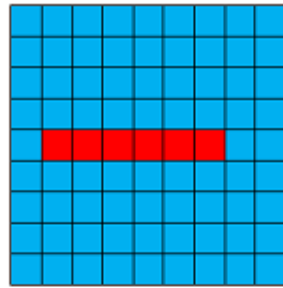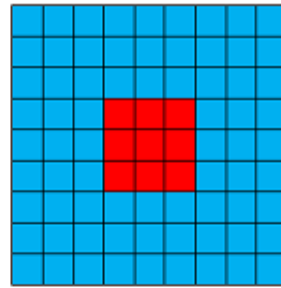
(a) Case 1   **(b)** Case 2   **(c)** Case 3   **(d)** Case 4
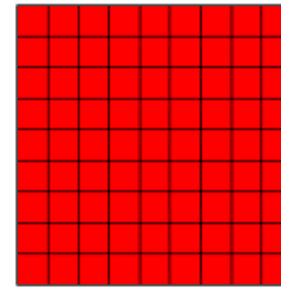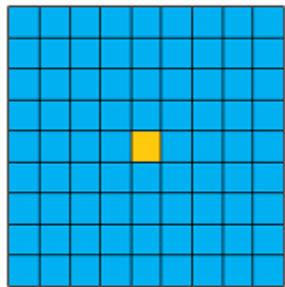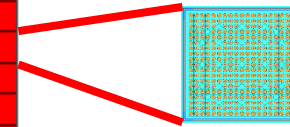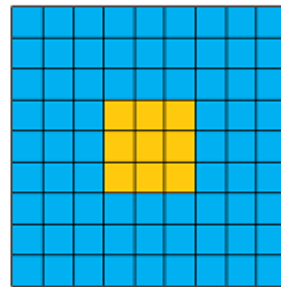
(e) Case 5   **(f)** Case 6   **(g)** Case 7   **(h)** Case 8
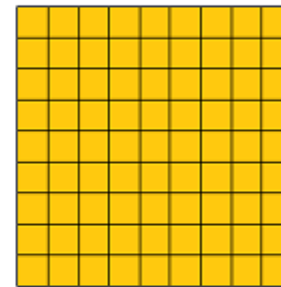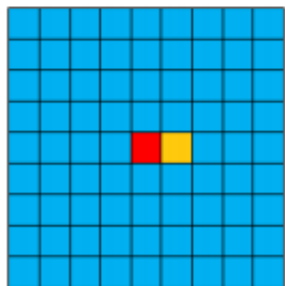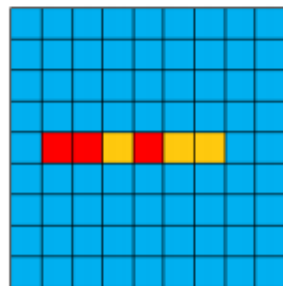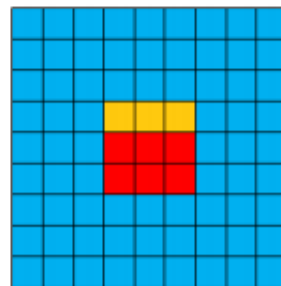
(i) Case 9   **(j)** Case 10   **(k)** Case 11   **(l)** Material Legend

Empty Assembly
4.95% Assembly
4.45% Assembly
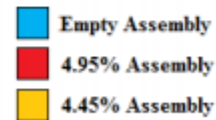
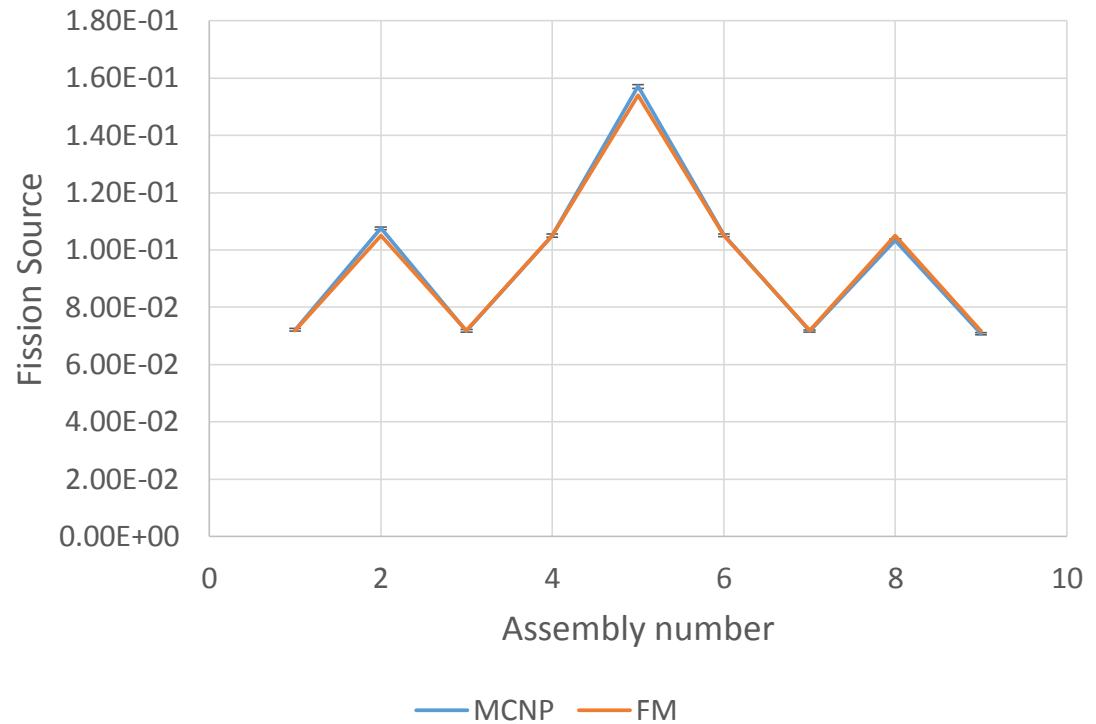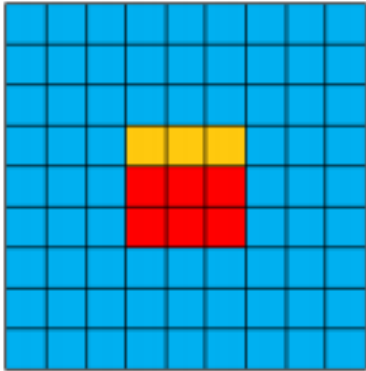| Case #	Number of Assemblies	Fuel Type
1	1x1	4.95%
2	6x1	4.95%
3	3x3	4.95%
4	9x9	4.95%
5	1x1	4.45%
6	6x1	4.45%
7	3x3	4.45%
8	9x9	4.45%
9	2x1	Mixed
10	6x1	Mixed
11	3x3	Mixed

Case 3 Eigenfunction
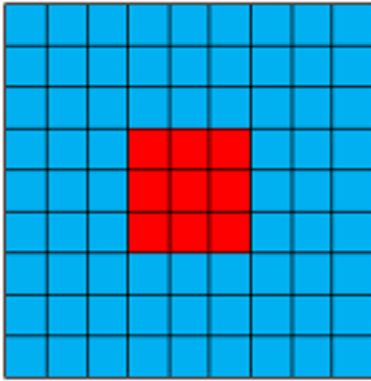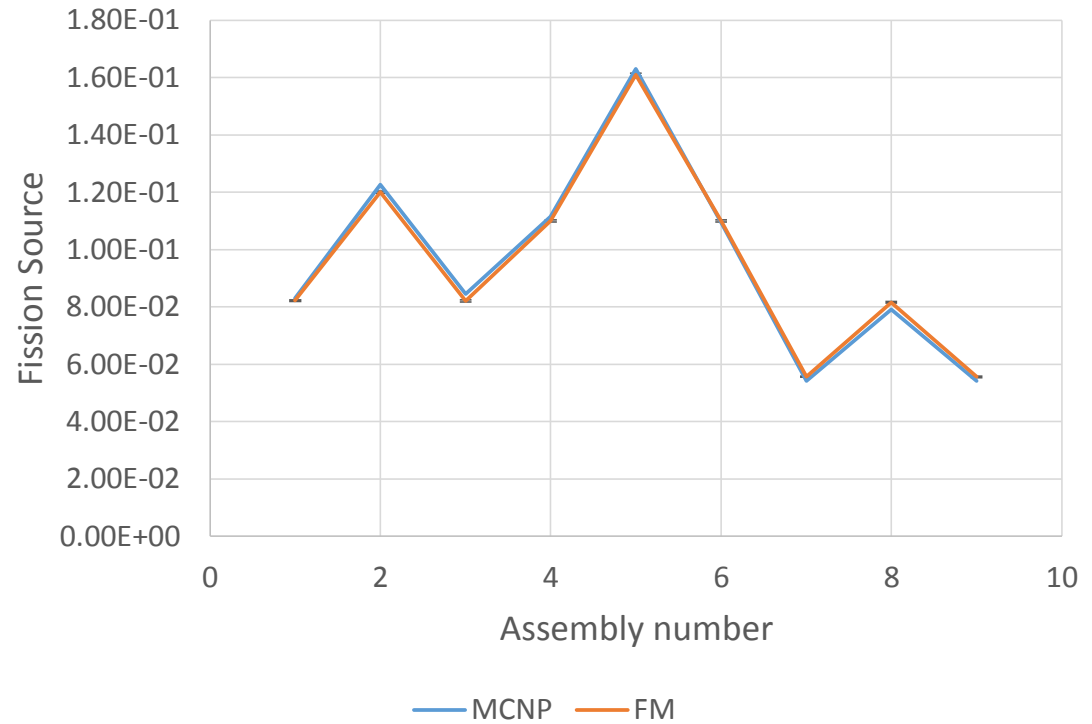

Reference Solution
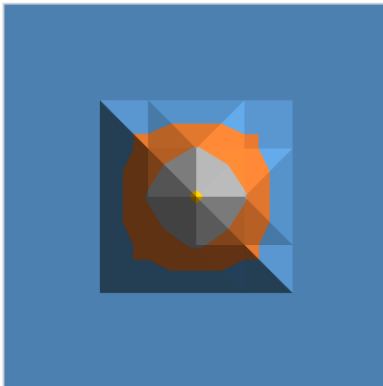
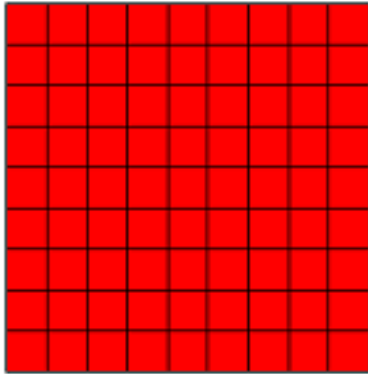Comparison of FM with MC

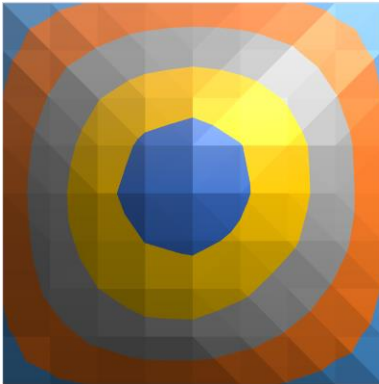# Case 11 Eigenfunction
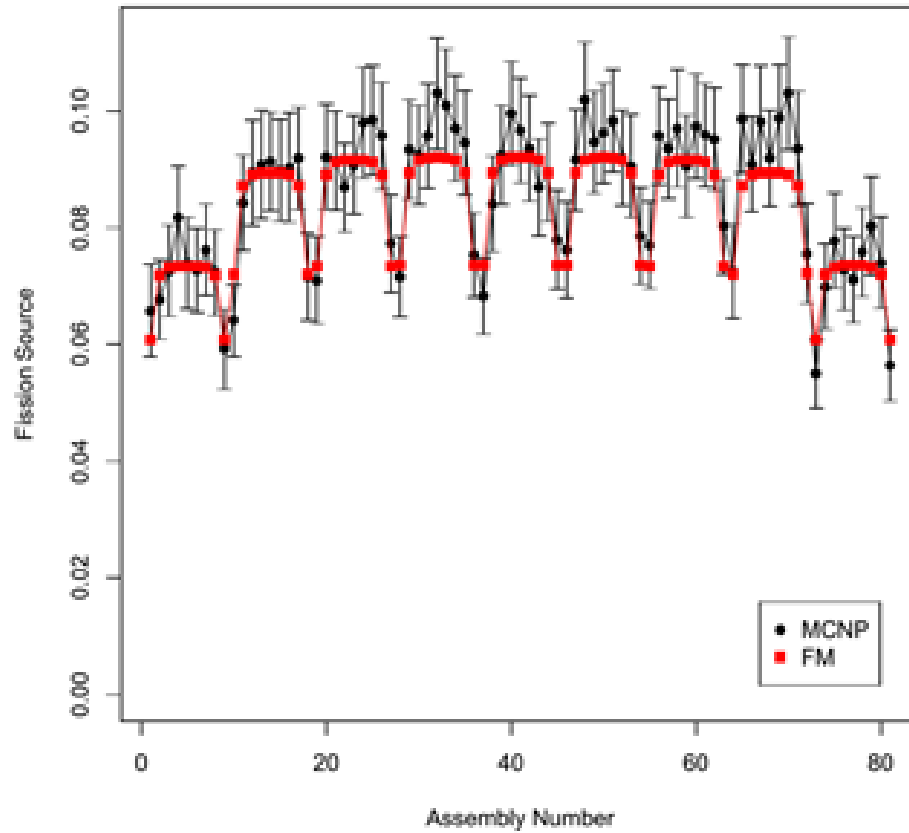

Reference Solution

Comparison with FM with MC

Case 4 Eigenfunction distribution

Reference Solution

Comparison with FM with MC

Comparison of calculated M - FM vs. MCNP

Case	FM		MCNP			Error in M (FM vs MCNP)	Speedup (FM vs MCNP)
	M	Time (min)	M	Time (min)	1-σ Uncertainty		
1x1	3.343353	0.092	3.33155	18591	0.0010	0.35%	202156
6x1	4.328244	0.213	4.31336	25122	0.0010	0.35%	117695
3x3	5.428051	0.965	5.40992	6500	0.0011	0.35%	6739
9x9	6.697940	8.17	6.67674	2421	0.012	0.32%	296

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?