

# Development and Benchmarking of an MOC-based Transport Solver with CMFD Acceleration Exploiting Parallel Architectures

Zafar Iqbal Zafar and Ser Gi Hong  
Hanyang University, 222 Wangsimni-ro, Seongdong-gu, Seoul, Korea  
hongsergi@hanyang.ac.kr

\*Keywords: MOC, CMFD, Parallel Computing, C5G7, and KAIST Benchmarks

## 1. Introduction

The method of characteristics (MOC) has extensively been developed over last couple of years for solving transport equations in nuclear engineering. Different aspects of MOC, including acceleration techniques, parallelizing algorithms, and its applications to diverse fields have been proven widely [1] [2], and incorporated to some codes like OpenMOC, STREAM, and DeCART [3] [4] [5].

This paper introduces a transport solver for a 2D reactor core problem using MOC including its detailed geometrical meshing capability, solving algorithm of the neutron transport equation, and compatibility with the multi-core systems. In this work, a MOC solver which can analyze 2D cores having various geometrical symmetries was developed and the standard Coarse Mesh Finite Difference (CMFD) acceleration scheme was implemented using the pin surface net currents. In particular, the MOC solver uses a pin cell-based modular ray tracing, which is very efficient in reducing the time for ray tracing. The geometry modeling is done with a similar way used in MCNP and so our MOC solver has great flexibilities in complicated shapes in fuel pins, guide tubes, and burnable poison pins. Also, OpenMP was applied in parallelization over the azimuthal directions and cyclic loops of the rays.

The validation of the MOC solver was performed against two benchmark problems, the C5G7 benchmark [6] and the KAIST benchmark 2b [7].

## 2. Methods

### 2.1 Geometry options

Depending upon the prevailing symmetries a problem could be modeled in one of the four geometry options (1) full core, (2) quarter core, (3) diagonally half core, or (4) octant core. The full core option has freedom of using any boundary condition on any side of the core. There is no restriction on the sizes of the pins or their meshing schemes also. However, all the other options require appropriate boundary conditions. Furthermore, a given flat source region (FSR) has to fully exist on one side of the symmetry line i.e. a

symmetry line cannot pass through an FSR. Similarly, the octant and diagonal symmetries require both boundary conditions and meshes to be symmetric about the diagonal lines. The Fig. 1 shows the available geometry options in the current version.

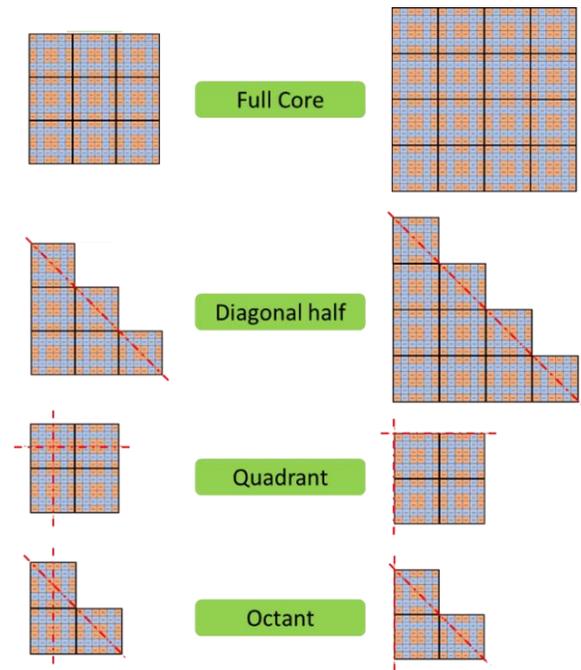


Fig. 1. Different geometry options in the solver

The geometry modeling starts by defining the FSRs using the predefined surfaces in geometrically distinct fuel pins. Then, the fuel assemblies are modeled using the fuel pins, and finally the core is modeled using the fuel assemblies.

### 2.2 MOC sweep

The contribution from a track  $k$  to the angular flux over an arbitrary FSR could be calculated using the following multigroup transport equation:

$$\frac{d\psi_{m,i,k,g,p}}{ds} + \Sigma_{i,g}^t \psi_{m,i,k,g,p} = Q_{i,g}, \quad (1)$$

where the source  $Q_{i,g}$  is given by

$$Q_{i,g} = \frac{\chi_{i,g}}{k_{eff}} \sum_{g'} \nu \Sigma_{i,g'}^F \phi_{i,g'} + \sum_{g'} \Sigma_{i,g' \rightarrow g}^S \phi_{i,g'}. \quad (2)$$

Integrating Eq. (2) with a track segment having  $s_{in}$  and  $s$  as the entering and leaving points for an FSR  $i$ , gives the change in angular flux along track  $k$  over FSR  $i$ . This change in angular flux along a track segment in an FSR is given by

$$\Delta \psi_{m,i,k,g,p} = \left( \psi_{m,i,k,g,p}(s_{in}) - \frac{Q_{i,g}}{\Sigma_{i,g}^t} \right) \left( 1 - e^{-\Sigma_{i,g}^t s_{m,i,k,p}} \right). \quad (3)$$

The polar angular quadrature suggested by Yamamoto for the numerical polar direction integration was used in this work. The change in angular flux is thus first numerically integrated to get the corresponding change in the scalar flux due to a track segment passing through that FSR as in Eq.(4).

$$\Delta \phi_{i,k,g} = \sum_p w_p \sin \theta_p \Delta \psi_{mi,k,g,p}. \quad (4)$$

Finally, the scalar flux for the energy group  $g$  and the FSR  $i$  is obtained by summing this  $\Delta \phi_{i,k,g}$  from all tracks  $k$  in the FSR and all azimuthal angles as follows:

$$\phi_{i,g} = \frac{4\pi}{\Sigma_{i,g}^t} \left[ Q_{i,g} + \frac{1}{4\pi A_i} \sum_k w_{m(k)} w_k \Delta \phi_{i,k,g} \right]. \quad (5)$$

### 2.3 CMFD sweep

The scalar flux obtained from Eq. (5) is correct numerically, however, it converges rather slowly. Out of the numerous accelerating techniques, coarse mesh finite difference method is used, because of its simplicity and power to converge in rather fewer iterations. In the CMFD module, the pin-wise coarse mesh diffusion problem is solved using the Jacobi method, which conserves the surface currents and reaction rates to the FSR-wise fine mesh transport problem. The non-linear coupling coefficients are calculated from the pin-wise net-current accumulated during the MOC sweep to ensure equivalence of the corresponding diffusion solution.

The current summation for the pin located at  $(I, J)$ , at its surface  $s$ , from track  $k$ , and polar angle  $p$  is used for finding the normalized surface net currents as follows:

$$\tilde{J}_{I,J,g,s} = \sum_{k \in s} \sum_p w_{m(k)} w_p w_k \sin \theta_p \psi_{m,k,p,g,p}. \quad (6)$$

The pin-averaged flux and cross sections for the pin at  $(I, J)$  from all FSRs  $i$  that exist inside this pin are calculated using Eq.(7a) and (7b), respectively. The

FSR area  $A_i$  and the pin area  $A_{I,J}$  are calculated numerically in the track-laying module once, only.

$$\bar{\phi}_{I,J,g} = \frac{\sum_i \phi_{i,g} A_i}{A_{I,J}} \quad (7a)$$

$$\Sigma_{I,J,g} = \frac{\sum_i \Sigma_{i,g} \phi_{i,g} A_i}{\sum_i \phi_{i,g} A_i} \quad (7b)$$

The pin-surface diffusion coupling coefficient ( $\hat{D}$ ) and the non-linear coupling coefficient ( $\tilde{D}$ ) for the surface  $s$  of the pin  $(I, J)$  with its neighboring location  $(I', J')$ ; (+) for the right and top surfaces and (-) otherwise, are calculated as follows:

$$\hat{D}_{I,J,g,s} = \frac{2 \cdot D_{g,I,J} \cdot D_{g,I',J'}}{D_{g,I,J} \cdot h + D_{g,I',J'} \cdot h}. \quad (8a)$$

$$\tilde{D}_{I,J,g,s} = \frac{\pm \hat{D}_{I,J,g,s} (\bar{\phi}_{g,I',J'} - \bar{\phi}_{g,I,J}) + \frac{J'_{I',J',g,s}}{h}}{\bar{\phi}_{g,I',J'} + \bar{\phi}_{g,I,J}}. \quad (8b)$$

Negative scalar fluxes could be encountered for certain cases if the non-linear correction coefficient was large. For instance, due to multiple types of the MOX fuel pins, CMFD acceleration can fail when newly updated non-linear correction factors are used. However, a damping factor of 0.5 as a rough guess is good to stabilize the acceleration scheme [3]. The damping factor of '0' would mean using correction coefficient from the previous iteration (i.e. slow convergence), and '1' would mean a fully new correction factor from the current iteration (i.e. faster convergence but lower stability).

The pin-wise flux is calculated for the pin  $(I, J)$  that has neighboring location pins  $(I', J')$ , iteratively using Jacobi method coupled with OpenMP parallel computing. The first term of Eq. (9) i.e. leakage sum, is performed over all four surfaces of the pin, and  $s'$  is simply the opposite surface of  $s$ , (like north for south and vice versa).

$$\frac{1}{h} \{ \sum L_s - L_{s'} \} + \Sigma_{g,I,J}^t \bar{\phi}_{g,I,J} = Q_{I,J}, \quad (9),$$

where

$$Q_{I,J} = \frac{\chi_{i,g}}{k_{eff}} \sum_{g'} \nu \Sigma_{g',I,J}^f \bar{\phi}_{g',I,J} + \sum_{g' \neq g} \Sigma_{I,J,g' \rightarrow g}^s \bar{\phi}_{g',I,J}$$

$$L_s = \hat{D}_s (\bar{\phi}_{g,I,J} - \bar{\phi}_{g,I',J'}) \pm \tilde{D}_s (\bar{\phi}_{g,I,J} + \bar{\phi}_{g,I',J'}).$$

After normalizing the CMFD flux to the pin-wise MOC flux, the FSR-wise MOC flux,  $\phi_{i,g}^{MOC}$ , is corrected as in Eq.(10). This flux prolongation and updating of effective multiplication factor from the CMFD module are the two improvements to the MOC solution that accelerate the convergence.

$$\phi_{i,g}^{MOC} \rightarrow \phi_{i,g}^{MOC} \times \frac{\bar{\phi}_{i,j,g}^{CMFD}}{\bar{\phi}_{i,j,g}^{MOC}} \quad (10)$$

Finally, the total source in the MOC module is calculated from this corrected FSR wise flux ( $\phi_{i,g}^{MOC}$ ) of Eq. (10) before remaining MOC sweep of Eq. (2).

#### 2.4 Parallelizing

OpenMP (Open Multi-Processing) is used for executing most of the loops in parallel. The input is read and processed over a single thread. Later, the tracks are generated in parallel. So, presence of the multiple unique pins does not take longer time than generating tracks for a single pin. Afterwards, these tracks are linked in parallel to form the core-wise global tracks and their closed loops.

Inside the MOC and the CMFD modules, most of the loops are parallelized using simple omp directives. The outer most loops of the MOC solver module is enlarged by combining azimuthal angles and corresponding loops of the global tracks. This is done to accommodate larger teams of the threads. Because the global tracks are not identical, so, the load on each thread is also not identical. That is the reason for using dynamic load balancing for this loop.

Thread isolation without absolute locking directives like atomic, are preferred to ensure safe data updating on the heap memory but the absolute isolation, e.g. provided by the critical directive is avoided due to their overhead time requirements. Moreover, due to extensive heap memory usage, the neutron current normalization, final data printing, and numerous other petty tasks are performed using a single thread.

#### 2.5 Benchmarking

The C5G7 benchmark geometrical description along with seven-group cross sections set, and the KAIST 2b benchmark along with its respective seven-group cross sections set are used for verification of the results. Both benchmarks (C5G7 and KAIST 2b) are defined for a quadrant core geometry. Although both of the benchmarks have clear octant symmetry, both quadrant and octant geometries could be modeled in the current version of the solver. However, simply following the given benchmarks, the quadrant geometries as shown in the Fig. 1 are analyzed in this work.

### 3. Results and discussion

Same set of MOC solution conditions is used for the both benchmark problems. That is, a ray spacing of 0.05 cm, 8 azimuthal angles in a quadrant, and 3 polar angles in a hemisphere are used.

#### 2.1 KAIST 2b benchmark

The pin power profile for the KAIST 2b benchmark problem with all control rods inserted is reproduced in Fig. 2. Higher power contribution from the MOX fuel assemblies is the reason for the darker shade. The instrument tubes are filled with coolant, and the control rods do not produce noteworthy amount of power. That is the reason for lighter shade at their locations.

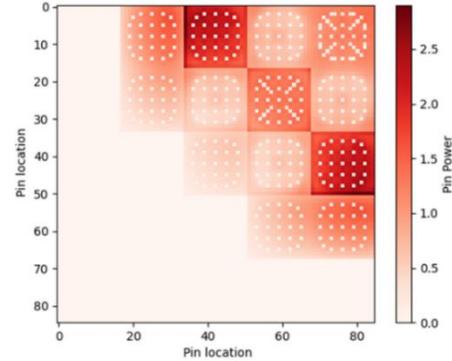


Fig. 2. Pin power profile with all CR inserted

Table I gives the effective multiplication factor and the relative maximum and minimum pin powers. The contribution to total power from the peripheral assemblies is less due to vacuum boundary condition on the reflector side of the core (bottom and left in the Fig. 2). Similarly, four fuel assemblies having control rods inserted also produced relatively less power, hence lighter shade in Fig. 2.

Table I: Results of KAIST 2b core analysis

Quantity	$k_{eff}$	$P_{max}^a$	$P_{min}^b$
Value	0.96286	2.895	0.114

<sup>a</sup>Maximum relative pin power,

<sup>b</sup>Minimum relative pin power

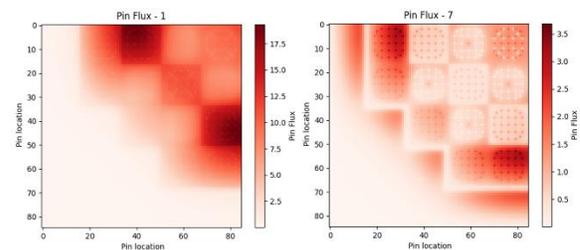


Fig. 3. Neutron flux profile in the KAIST benchmark 2b; left: fast group 1, right: Thermal group – 7

Although no noteworthy fast neutron flux exists beyond the baffle, plenty of thermal neutrons do exist (Fig. 3). So, presence of the extra water assemblies is justified. Relatively harder spectrum in the MOX

assemblies and softer in the  $UO_2$  assemblies are also properly produced. A reflective BC at the right and top of Fig. 3 could be attributed to the higher fast neutron flux near these surfaces, which constitute the core center-lines. Hence, all of the perceivable neutronic characteristics are successfully estimated by our solver.

## 2.2 C5G7 benchmark

The current solver was also applied to estimate the effective multiplication factor, the maximum pin power and minimum pin powers for the C5G7 benchmark. The calculated benchmark characteristics given in Table II are comparable to the reference values given in the benchmark. The maximum error in the calculated pin-power for all pins is less than 2%, which is comparable to other well-known packages like DeCART2D [5], also given in the benchmark solution. The relative error in  $k_{eff}$  is just -0.03%.

The  $Error_{max}$  in table II is the maximum error in any pin in the reactor core. This pin (and other pins with error of the order of 1%) are the fuel pins located adjacent to the reflector region. So, this could be attributed to the flat source approximation because there is a big change in the pin-source from the fuel region to the neighboring reflector region. A sensitivity analysis with a finer flat-source region and a linear source approximation are the planned improvements against the error of around 30 pcm compared to the reference MCNP calculation.

Table II Results of C5G7 benchmark analysis

Quantity	Calculated	*Reference	Relative error
$k_{eff}$	1.186896	1.186550	-0.03%
$P_{max}$	2.494	2.498	0.16%
$P_{min}$	0.235	0.23	-0.56%
$Error_{max}$	0.270	0.265	1.74%

\*MCNP results

## 2.3 Parallelizing

Currently, the parallel efficiency is around 40% on a 16 thread 8 core personal computer, when both MOC and CMFD solver modules were used an 11th Gen Intel i7 dual processor machine (Fig. 4). The single sweep time saturates to little less than one second after 8 threads. This could be attributed to false sharing or non-efficient load balancing. Increasing number of threads is definitely beneficial but it does not look as good as quoted by some other codes [3].

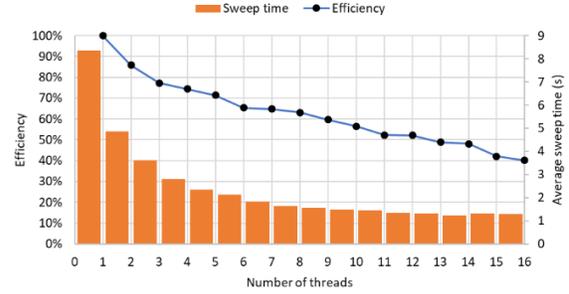


Fig. 4. Total time for a full sweep and parallel efficiency as a function of number of threads used

## 4. Conclusion

An MOC-based neutron transport solver was developed for 2D core problems. This solver concerns the reactor core in cartesian geometries incorporating different symmetry options. Use of the acceleration technique CMFD and running on multi-thread systems via OpenMP directives makes it fast enough to solve a 100-assembly core benchmark like C5G7 in less than one minute on a personal computer.

## REFERENCES

- [1] A. Hebert, Applied Reactor Physics, Montreal: Presses internationales Polytechnique, 2020.
- [2] D. G. Cacuci, "Vol. 2: Reactor Design," in Handbook of Nuclear Engineering, France, Springer Science & Business Media, 2010, p. 751.
- [3] W. Boyd, S. Shaner, L. Li, B. Forget and K. Smith, "The OpenMOC method of characteristics neutral particle transport code," Ann. Nucl. Energy, vol. 68, pp. 43-52, 2014.
- [4] Y. Zheng, S. Choi and D. Lee, "A new approach to three-dimensional neutron transport solution based on the method of characteristics and linear axial approximation," Journal of Computational Physics, vol. 350, pp. 25-44, 2017.
- [5] J. Cho, K. Kim, H.-Y. Kim, J. Lee, J. Seong-gyun and H. Joo, "DeCART2D v1.0 User's Manual," Korea Atomic Energy Research Institute, KAERI/TR-5116/2013, 2013.
- [6] E. E. Lewis, M. A. Smith, N. Tsoulfanidis, G. Palmiotti, T. A. Taiwo and R. N. Blomquist, "Benchmark Specification for Deterministic 2-D/3-D MOX Fuel Assembly Transport Calculations without Spatial Homogenisation (C5G7 MOX)," Nuclear Energy Agency, NEA/NSC/DOC(2001)4, 2001.
- [7] N. Z. Cho, "KAIST-Benchmark Problem 2B," [Online]. Available: <https://github.com/nzcho/Nurapt-Archives/tree/master>. [Accessed 2023].