Hexagonal Geometries in MPACT

Tarek Ghaddar
Aaron Graham
Benjamin Collins
Brian J. Ade

August 2021

![OAK RIDGE National Laboratory]

Nuclear Energy and Fuel Cycle Division

HEXAGONAL GEOMETRIES IN MPACT

Tarek Ghaddar
Aaron Graham
Benjamin Collins
Brian Ade

August 2021

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

ORNL   Oak Ridge National Laboratory
CMFD   Coarse Mesh Finite Difference
MOC   Method of Characteristics

1. INTRODUCTION

The MPACT code is a high-fidelity light-water reactor analysis code using whole-core pin-resolved neutron transport calculations on modern parallel-computing hardware [1].

MPACT uses the 2D/1D method to solve 3D neutron transport problems by decomposing the problem into a stack of 2D slices, each of which is solved independently using the method of characteristics (MOC). The slices are then coupled axially using the P3 nodal expansion method (NEM-P3) for the 1D axial calculations. MPACT also employs the coarse mesh finite difference (CMFD) method to accelerate calculations.

This manuscript details work supporting advanced reactor designs using hexagonal pins and hexagonal assemblies such as the VVER-1000. If performed correctly, MOC is geometry agnostic. However, MPACT previously had optimizations in place for Cartesian geometries, specifically in the modularization and current calculations.

Sections 2. and 3. detail the changes made to MPACT to support MOC and CMFD calculations on hexagonal geometries. Section 4. reports results demonstrating solution consistency for problems run with and without CMFD acceleration, results demonstrating solution consistency when run in serial and parallel, and pincell results using the Monte Carlo code, McCard's benchmark results [2].

2. MODULARIZATION OF HEXAGONAL GEOMETRIES

MPACT takes advantage of the often repeated geometry of light-water reactors by ray tracing over *modules*. Modules are subdomains that exhibit a unique geometry. With small adjustments to ray spacing and angle, the modularization process can ensure that rays traced on modules link across the entire domain for each angle. Figure 1 [3] shows nine modules, each of which has modular rays that link across the module boundaries to form long rays across the problem domain.
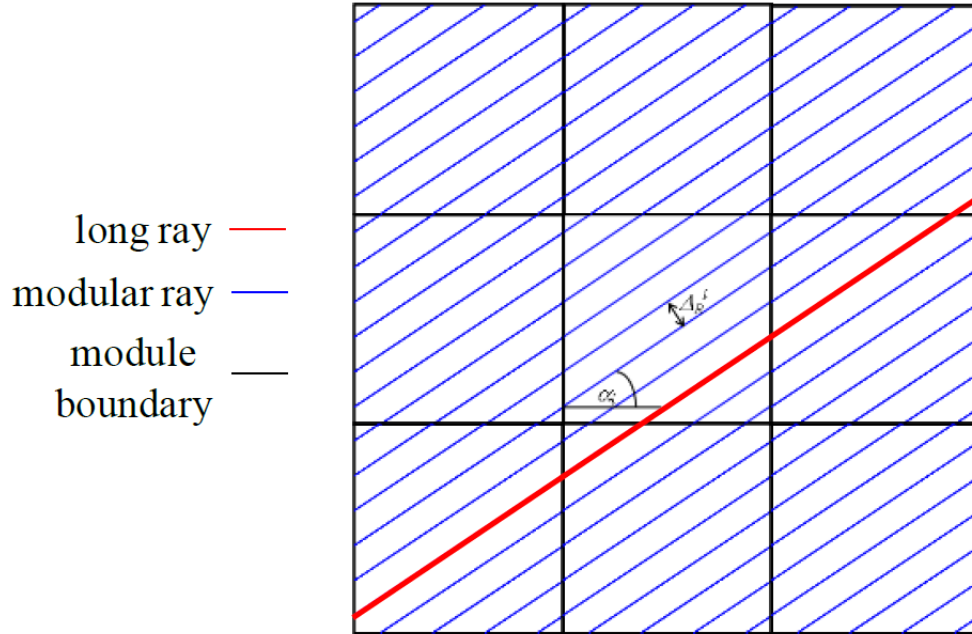
Figure 1. The modules, modular rays, and long ray with Cartesian modules in MPACT.

Modular rays are determined based on the desired ray spacing and set of angular directions. This process differs when using cartesian or hexagonal modules, so only the hexagonal modularization technique is discussed in this report.

2.1 HEXAGONAL MODULARIZATION IN MPACT

Figure 2 shows the the MPACT hexagonal module with the X, U, and P, sides labeled. It follows that:

- X: North/south faces

- U: East-south-east (ESE)/west-north-west (WNW) faces

- P: East-north-east (ENE)/west-south-west (WSW) faces
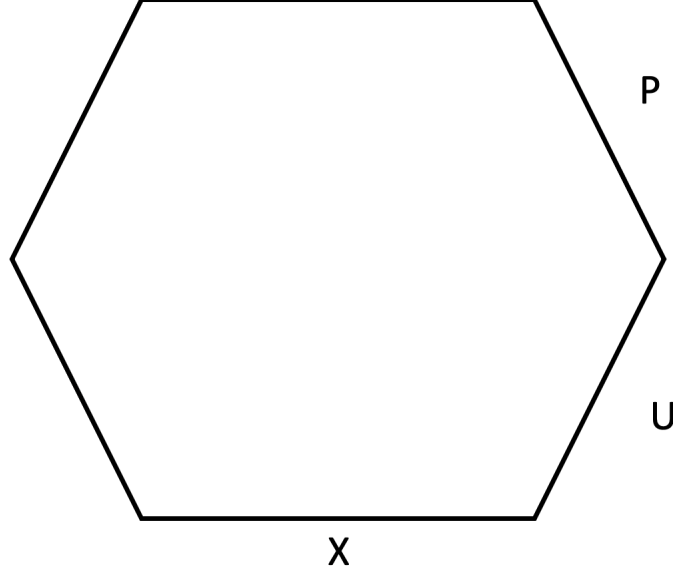
Figure 2. The hexagonal module in MPACT with the X, U, and P sides labeled.

The hexagonal modularization process in MPACT is influenced by previous work performed in the DeCART code [4]. The goals of modularization on hexagonal geometries are as follows:

1. Calculate adjusted angles and ray spacing on the sixth space (0°, 60°) that allow for rays to be linked across modules for each angle,

2. Expand the adjusted set of angles to the half space (0°, 180°) so that each angle has a reflected angle for the X, U, and P in the set.

Both codes begin the modularization process given a set of angles $\tilde{\alpha} < 60°$ and desired ray spacing $\Delta\tilde{A}$. Eqs. 1, 2, and 3 use this information to calculate the number of ray intersections on the X, U, and P sides:

$$N_X = \text{CEIL}[\sin(\tilde{\alpha})\frac{L}{\Delta\tilde{A}}], \tag{1}$$

$$N_U = \text{CEIL}[\sin(60° - \tilde{\alpha})\frac{L}{\Delta\tilde{A}}], and \tag{2}$$

$$N_P = N_X + N_U, \tag{3}$$

where L is the side length of the regular hexagonal module, and CEIL is the ceiling function that rounds to the nearest integer greater than or equal to its argument.

Using N_X and N_U, Eqs. 4 and 5 are used to calculate an adjusted ray angle α and ray spacing ΔA that satisfy that modularity condition:

$$\tan\alpha = \frac{\sqrt{3}}{2(N_U/N_X) + 1}, and \tag{4}$$

$$\Delta A = \frac{L\sin\alpha}{N_X}. \tag{5}$$

.

The calculation of the number of ray intersections on the X, U, and P sides, as well as the calculation of the modularized angles in $(0, 60°)$ and ray spacing, all mirror DeCART [4] exactly. This part of the modularization process ensures that rays for a given angle can be linked across modules.

The expansion of the angle set from $(0, 60°)$ to $(0, 180°)$ deviates from DeCART. Given a set of modularized angles from $(0, 60°)$, we define a few parameters:

- n_{az} = the number of azimuthal angles on the halfspace $(0, 180°)$,

- $n_{az,hex}$ = the number of azimuthal angles on the sixth space $(0, 60°)$,

- angles = an array storing the the angles on the halfspace (size n_{az}), and

- angles$_{hex}$ = an array storing the adjusted angles on the sixth space (size $n_{az,hex}$),

---
Algorithm 1: Angular Expansion

---
for $i=1,n_{az,hex}$ **do**

    angles(i) = angles$_{hex}$(i)

    angles(i + $n_{az,hex}$) = angles$_{hex}$(i) + 60°

    angles(i + 2 · $n_{az,hex}$) = 180° - angles$_{hex}$($n_{az,hex}$ - i + 1)

end

Algorithm 1 shows how angles are expanded to meet the reflection requirement. Using Algorithm 1 allows the number of ray intersections to be established in the second $(60°, 120°)$ and third $(120°, 180°)$ angular sextants based on the number of ray intersections calculated for the first angular sextant $(0, 60°)$.

Calculating the ray intersections using the expanded adjusted angles and Eqs. (1–3) leads to erroneous N_X, N_U, and N_P values for angles in the second and third angular sextants. These values do not meet the reflection capability. This is because Eqs. (1–3) calculate the ray crossings prior to the adjustment of the angles. However, the N_X, N_U, and N_P values from the first angular sextant can be used to calculate the number of ray crossings of its corresponding expanded angle in the second and third angular sextants.

For an angle in the second angular sextant:

$$N_{X_2} = N_{P_1},$$
$$N_{U_2} = N_{X_1},$$
$$N_{P_2} = N_{U_1}, , \tag{6}$$

where N_{P_1} is the number of rays crossing the P surface for an angle in the first sextant, and N_{X_2} is the number of rays crossing the X surface for the corresponding angle in the second angular sextant.

For an angle in the third angular sextant:

$$N_{X_3} = N_{U_1},$$
$$N_{U_3} = N_{P_1},$$
$$N_{P_3} = N_{X_1}. \tag{7}$$

To demonstrate this, consider a case that has the following:

- $n_{az} = 24$

- $n_{az,hex} = 8$

The third angle in the adjusted set on the first angular sextant, $\alpha_3 = 19.11°$, calculated with $N_X = 3$, $N_U = 6$, and $N_P = 9$, is shown in Fig. 5.
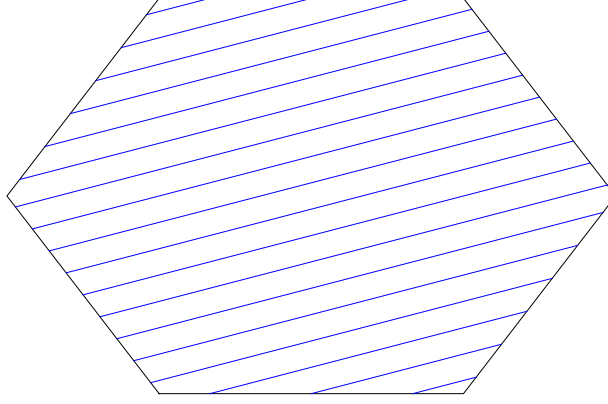
Figure 3. The rays on a hexagonal module with $\alpha_3 = 19.11°, N_X = 3, N_U = 6, N_P = 9$.

Following Algorithm 1, the corresponding angle in the second angular sextant is the 11th angle of the set, $\alpha_{11} = 79.11°$. Using Eq. (6), we calculate $N_X = 9$, $N_U = 3$, and $N_P = 6$, shown in Figure 4.
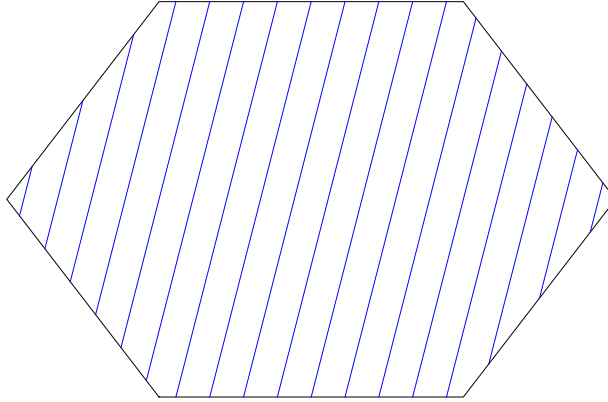
Figure 4. The rays on a hexagonal module with $\alpha_{11} = 79.11°, N_X = 9, N_U = 3, N_P = 6$.

Following Algorithm 1, the corresponding angle in the third angular sextant is the 19th angle of the set, $\alpha_{19} = 139.11°$. Using Eq. (7), we calculate $N_X = 6$, $N_U = 9$, and $N_P = 3$, shown in Figure 4.
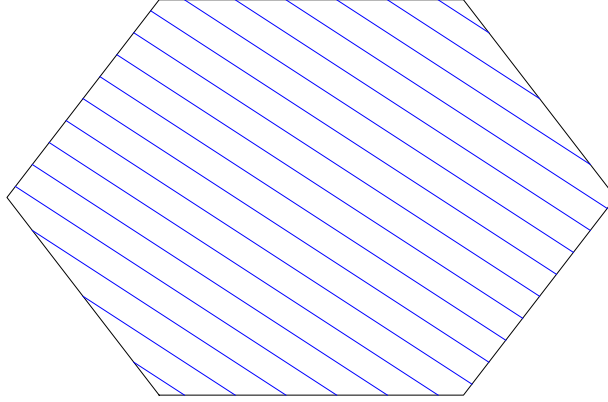
Figure 5. The rays on a hexagonal module with $\alpha_{19} = 139.11°, N_X = 6, N_U = 9, N_P = 3.$

Using α_3, we check that the reflected angles meet the reflection capability for each face (equivalent number of ray crossings for each reflected angle-pair). For the X face, α_3's reflected angle is α_{22}, as shown Figs. 6a and 6b. This reflected angle-pair meets the reflection capability with $N_X = 3$ for both angles.
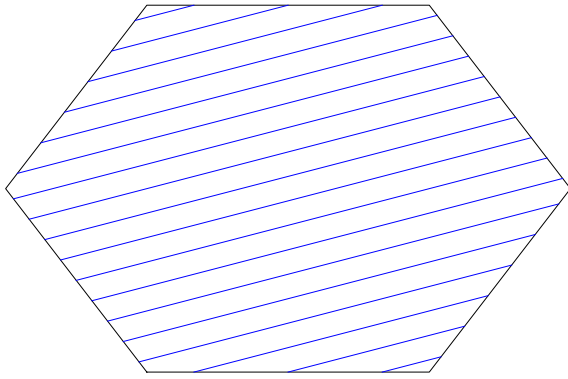
(a) $\alpha_3 = 19.11°$
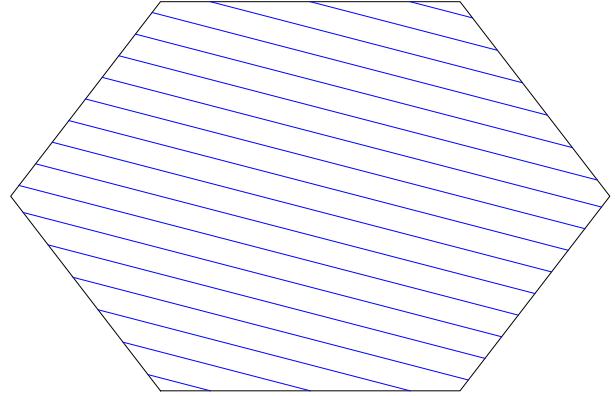
(b) $\alpha_{22} = 160.89°$

Figure 6. α_3 **and** α_{22} **with equivalent** $N_X = 3.$

For the U face, α_3's reflected angle is α_{14}, as shown in Figs. 7a and 7b. This reflected angle-pair meets the reflection capability with $N_U = 6$ for both angles.
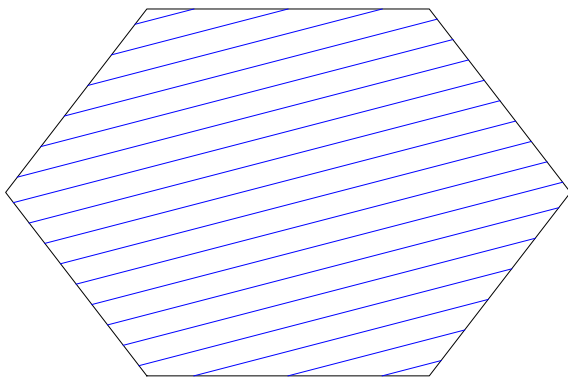
(a) $\alpha_3 = 19.11°$ **(b)** $\alpha_{14} = 100.89°$
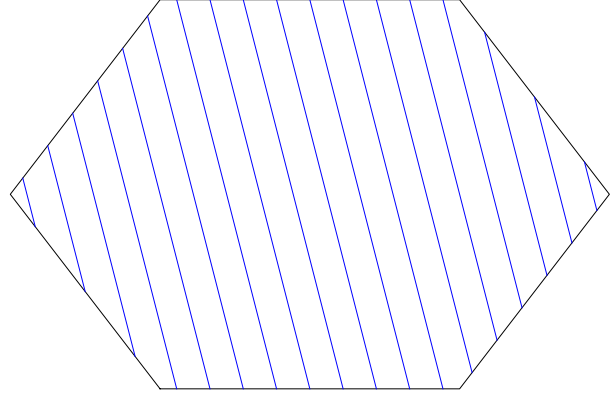
Figure 7. α_3 **and** α_{14} **with equivalent** $N_U = 6$**.**

For the P face, α_3's reflected angle is α_6, as shown in Figs. 8a and 8b. This reflected angle-pair meets the reflection capability with $N_P = 9$ for both angles.
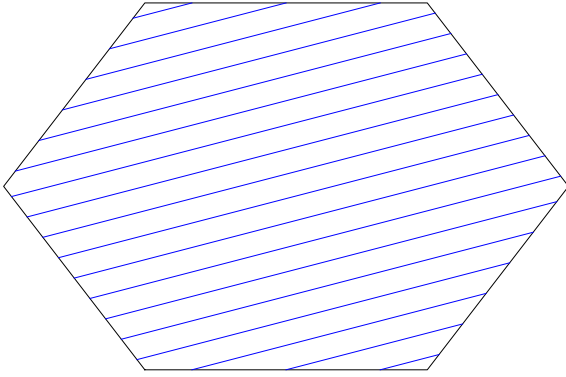
(a) $\alpha_3 = 19.11°$ **(b)** $\alpha_6 = 40.89°$
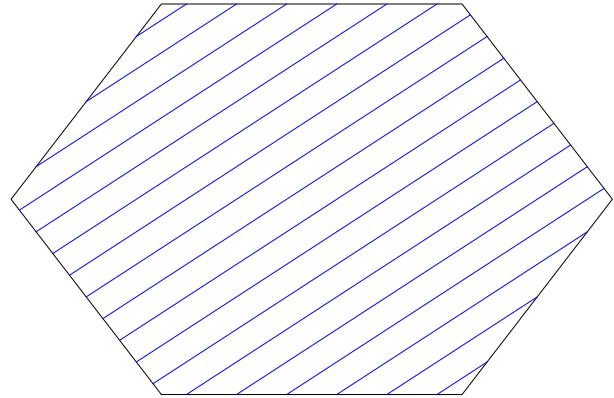
Figure 8. α_3 **and** α_6 **with equivalent** $N_P = 9$**.**

With modular rays that meet the reflection capability for reflected angle-pairs and link across modules to form core-long rays, MPACT can effectively perform the MOC sweep on hexagonal geometries. The next step is to ensure that currents are being calculated correctly for CMFD acceleration.

3. CURRENT CALCULATION FOR CMFD IN MPACT

The current contribution of a ray crossing a surface for an energy group and angle is calculated by Eq. (8):

$$\vec{j} \mathrel{+}= w_{\text{total}}(\psi_{\text{forward}} - \psi_{\text{backward}}), \tag{8}$$

where $\psi_{\text{forward/backward}}$ is the angular flux in the forward or backward direction of the ray, and ω_{total} is the weight assigned to that ray's contribution to the net current on that surface. It is calculated as follows:

$$w_{\text{total}} = dz dA w_\alpha \pi w_\theta \sin(\theta) \text{SIGN}(\vec{n} \cdot \vec{\Omega}) \tag{9}$$

$$\vec{n} \cdot \vec{\Omega} = (\Omega_x n_x + \Omega_y n_y + \Omega_z n_z)\text{DirSign}, \tag{10}$$

where dA is the ray spacing, w_α and w_θ are angular quadrature weights, π is a normalizing factor, $\sin(\theta)$ projects the quadrature weights to the 2D plane, and DirSign is +1 for positive surfaces (N and E) and -1 for negative surfaces (S and W).

Consider the CMFD cell, ray, and two surfaces in Fig. 9.

- SIGN($\vec{\Omega_1} \cdot \vec{n_1}$), as calculated by Eq. (10), yields a value of -1, flipping ψ_{backward}, or the inbound angular flux, to the positive contribution to the net current.

- SIGN($\vec{\Omega_1} \cdot \vec{n_2}$), as calculated by Eq. (10), yields a value of +1, keeping ψ_{forward}, or the inbound angular flux, as a positive contribution to the net current.

- SIGN($\vec{\Omega_2} \cdot \vec{n_3}$), as calculated by Eq. (10), yields a value of +1, keeping the outbound current contribution positive.

- SIGN($\vec{\Omega_2} \cdot \vec{n_4}$), as calculated by Eq. (10), yields a value of -1, keeping the outbound current contribution positive.
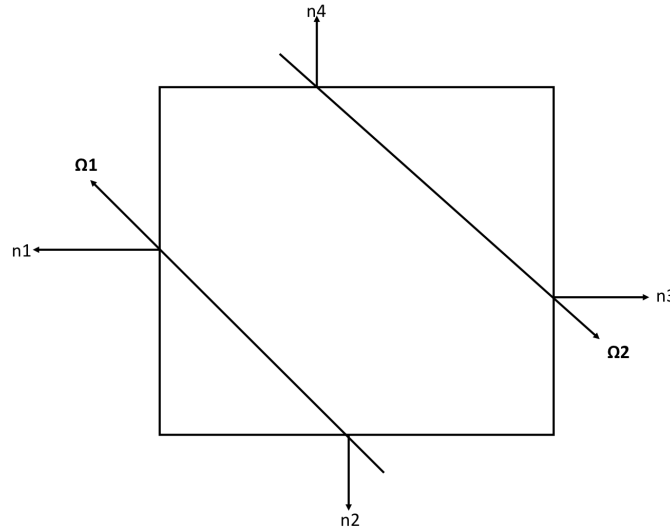

Figure 9. A Cartesian CMFD cell with two rays.

It follows that for "positive surfaces" (in Cartesian modules, this is the north and east surfaces), the outbound current contribution should be positive, with the inbound current contribution negative, and for

"negative surfaces," the inbound current contribution should be positive, with the outbound current contribution negative.

3.1 CURRENT MPACT IMPLEMENTATION FOR HEXAGONAL CMFD CELLS

The current calculation for hexagonal CMFD cells maintains Eqs. (8), (9), and (10).

Figure 10 shows a hexagonal CMFD cell with two rays crossing four surfaces.
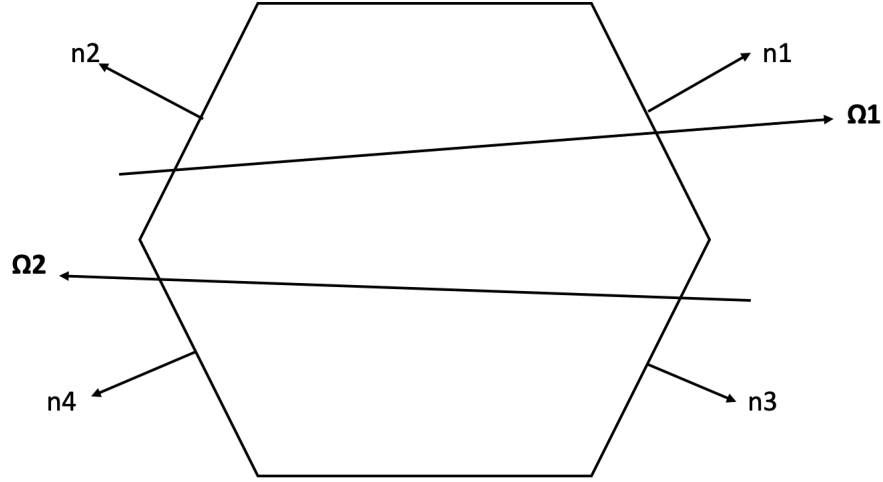
Figure 10. A hexagonal CMFD cell with two rays.

- SIGN($\Omega_1 \cdot n_1$), as calculated by Eq. (10), is +1, keeping the outbound current contribution positive, which matches Cartesian behavior for a positive surface.

- SIGN($\Omega_1 \cdot n_2$), as calculated by Eq. (10), is +1, keeping the inbound current contribution positive, which matches Cartesian behavior for a negative surface.

- SIGN($\Omega_2 \cdot n_3$), as calculated by Eq. (10), is -1, making the outbound current contribution (ψ_{backward}) positive, which matches Cartesian behavior for a positive surface.

- SIGN($\Omega_2 \cdot n_4$), as calculated by Eq. (10), is -1, making the inbound current contribution (ψ_{backward}) positive, which matches Cartesian behavior for a negative surface.

Some simple checks can be performed to build confidence in the current calculation on hexagonal geometries for use in CMFD acceleration. CMFD is only an accelerator, so the solution for a problem should be almost equivalent, regardless of whether or not CMFD is used.

4.   RESULTS

To test the hexagonal geometry in MPACT, the following are checked:

1. Solution consistency for problems run with and without CMFD, both in serial and parallel, and

2. The k_{eff} of the VVER-1000 fuel pin benchmark [2].

4.1   SOLUTION CONSISTENCY

Figure 11 shows the problem that was used to test solution consistency for serial runs with and without CMFD acceleration and parallel runs with and without CMFD acceleration. The parallel cases were run with 12 message-passing interface (MPI) processors using spatial decomposition.
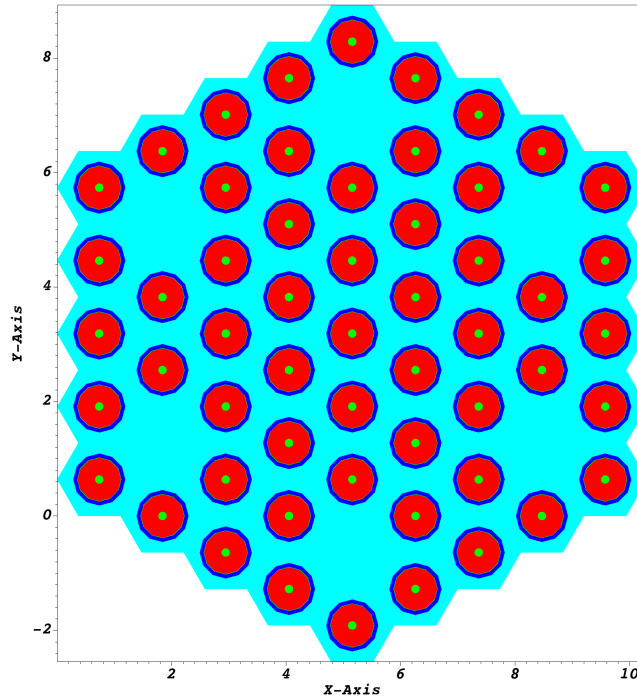
Figure 11. The problem geometry used to test solution consistency for serial and parallel runs with and without CMFD acceleration.

Table 1 demonstrates the consistency between serial cases with and without CMFD and parallel cases with and without CMFD acceleration. The serial MOC-only case is used as the reference solution.

Table 1. The test problem (Fig. 11) results run with serial MOC, serial MOC with CMFD acceleration, parallel MOC, and parallel MOC and CMFD.

Case	k_{eff}	Δk_{eff} (pcm)	Pin Power % RMS	Pin Power %Max
Serial MOC Only	1.2706401	0.0	0.0	0.0
Serial MOC+CMFD	1.2706422	0.165	1.76E-05	3.88E-05
Parallel MOC Only	1.2706396	0.039	7.82E-06	1.90E-05
Parallell MOC+CMFD	1.2706420	0.150	8.30E-06	2.34E-05

Figures 12a–12d plot the thermal fluxes for the cases tabulated in Table 1.

(a) Serial MOC only

(b) Serial MOC with CMFD acceleration

(c) Parallel MOC only

(d) Parallel MOC with CMFD acceleration

Figure 12. The thermal flux of the test cases run on Fig. 11 and tabulated in Table 1.

11

4.2   PINCELL SOLUTION VERIFICATION

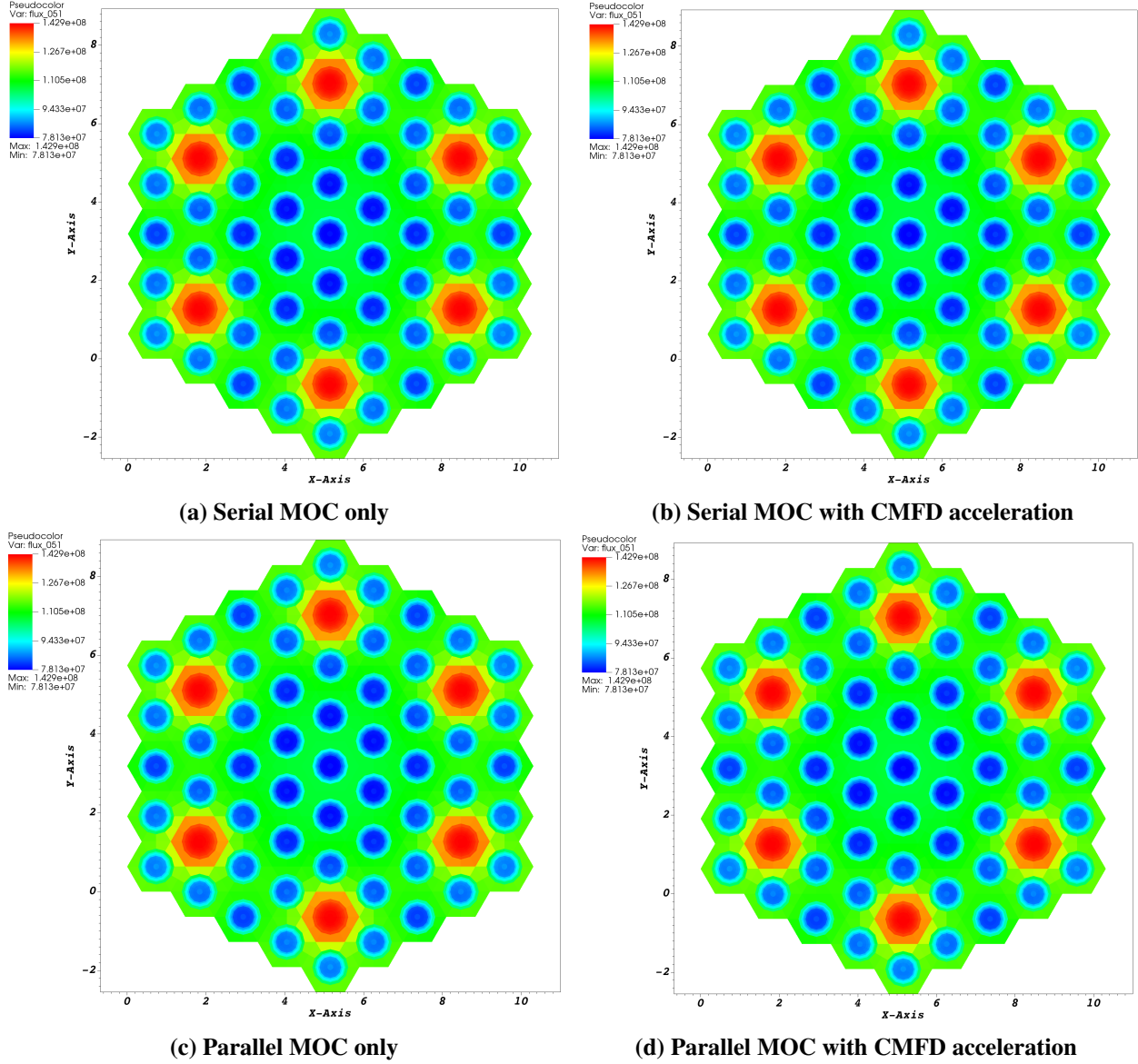The VVER-1000 fuel pin (shown in Fig. 13 was used to test MPACT's solutions on hexagonal geometry, and the continuous-energy Monte Carlo code McCard's solutions were used as the reference solution [2]. Each case was run with 36 azimuthal angles from (0,180°), 3 polar angles from (0,90°), and a ray spacing of 0.01 cm. Dimension, material composition, and case details can be found in the benchmark document [2].
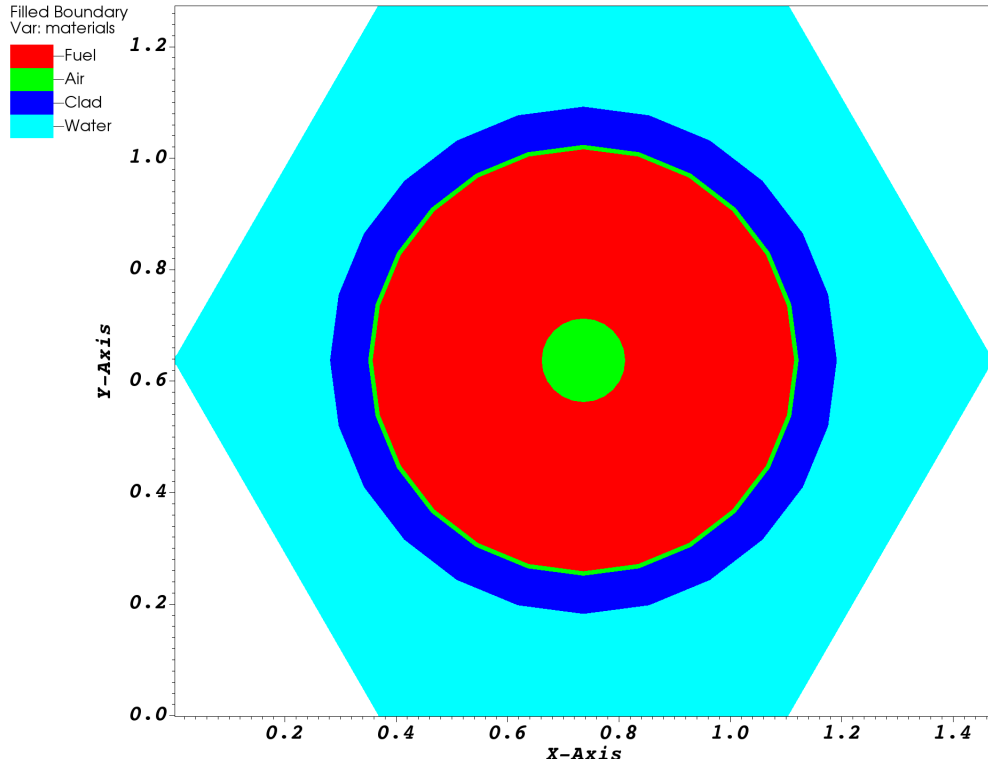
Figure 13. The VVER-1000 fuel pin geometry.

Table 2 tabulates the MPACT and McCard k_{eff} and the differences in their reactivities. The maximum reactivity difference is in case A01V15, and the minimum reactivity difference is in case A01V07. Case A01V15 is the fuel pin problem with 2,000 ppm soluble boron in the water, 2.0 weight percent [235]U, and the fuel, clad, and moderator temperatures at 600 K. Case A01V07 is the fuel pin problem with 1000 ppm soluble boron in the water, 2.0 weight percent [235]U, and the fuel, clad, and moderator temperatures at 300 K. Some discrepancy is expected, as the deterministic MOC and the stochastic Monte Carlo method are two different paths to solutions. However, for cases that exhibit larger discrepancies, some analysis can be done.

Table 2. The VVER fuel pin results from MPACT using McCard as a benchmark.

Case	MPACT k_{eff}	McCard k_{eff}	Δpcm
A01V01	1.276383	1.27732	73.36
A01V02	1.386073	1.38673	47.38
A01V03	1.211521	1.20996	-129.01
A01V04	1.307009	1.30515	-142.44
A01V05	1.200377	1.19902	-113.18
A01V06	1.295207	1.29376	-111.84
A01V07	1.074295	1.07433	3.26
A01V08	1.212733	1.21253	-16.74
A01V09	1.078295	1.07644	-172.33
A01V10	1.196971	1.19512	-154.88
A01V11	1.068443	1.06691	-143.69
A01V12	1.186258	1.18466	-134.89
A01V13	0.931487	0.93106	-45.86
A01V14	1.081247	1.08058	-61.73
A01V15	0.974415	0.97225	-222.68
A01V16	1.106239	1.10418	-186.47
A01V17	0.965599	0.96386	-180.42
A01V18	1.096441	1.09463	-165.44

Figure 14 plots the absolute value of the discrepancy (Δpcm) against the fuel temperature, clad temperature, and moderator temperature, and ppm of soluble boron for all the cases run. The largest trend in discrepancy between MPACT and McCard appears to be when materials are at 600 K, and to a slightly lesser extent, 900 K. Possible factors could be MPACT's cross section data at higher temperatures or differences in mesh resolution in fuel, water, and clad between the two code cases. Future work will explore these possibilities to improve understanding of the differences.
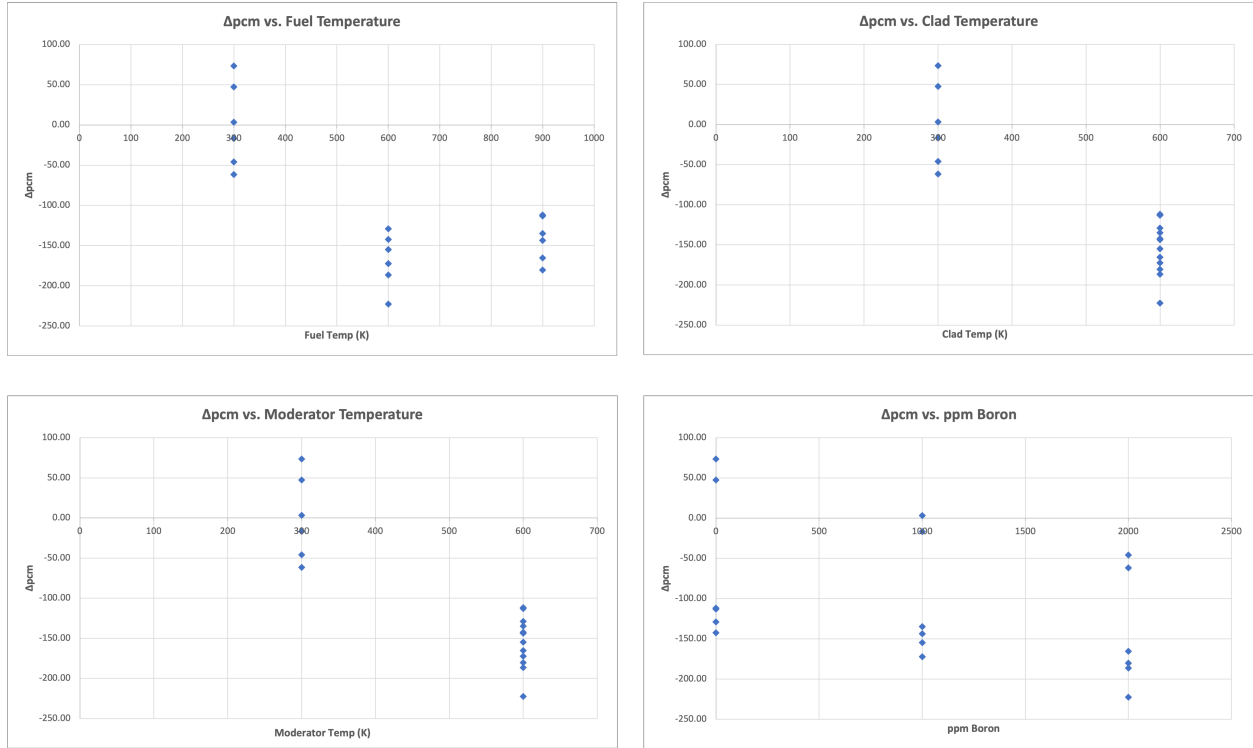
Figure 14. The reactivity discrepancy between MPACT and McCard plotted as functions of fuel temperature, clad temperature, moderator temperature, and ppm boron.

4.3 PRELIMINARY 2D/1D RESULTS

Preliminary testing has been performed on 2D/1D test cases. Figure 11 is extruded, and multiple axial slices were run, reflecting boundary conditions on the top and bottom faces. Theoretically, the 2D and 2D/1D representations of the problem should yield the same k_{eff}. Table 3 shows the expected consistency in k_{eff}.

Table 3. Test problem (Fig. 11) results run with serial MOC, serial MOC with CMFD acceleration, parallel MOC, and parallel MOC and CMFD.

Case	k_{eff}	Δk_{eff} (pcm)
1 axial slice	1.2706422	0.0
4 axial slices	1.2706420	0.015

5. REFERENCES

References

[1] B. Collins, S. Stimpson, B. W. Kelley, M. T. Young, B. Kochunas, A. Graham, E. W. Larsen, T. Downar, and A. Godfrey, "Stability and Accuracy of 3D Neutron Transport Simulations Using the 2D/1D Method in MPACT," *J. Comp. Phys.*, vol. 326, pp. 612–628, 2016.

[2] "VVER1000 Reactor Benchmark Problems," Tech. Rep. KAERI/TR-7339/2018, KAERI.

[3] MPACT Development Team, "MPACT Theory Manual Version 2.2.0," Tech. Rep. CASL-U-2016-1107-000, Oak Ridge National Laboratory.

[4] J.-Y. CHO, K.-S. KIM, H.-J. SHIM, J.-S. SONG, C.-C. LEE, and H.-G. JOO, "Whole Core Transport Calculation Employing Hexagonal Modular Ray Tracing and CMFD Formulation," *Journal of Nuclear Science and Technology*, vol. 45, no. 8, pp. 740–751, 2008.