Date: 10/9/2012

To: Bill Martin
c: Tom Evans

From: Tom Evans FA/ORNL

Level 3 Milestone Deliverable – L3:RTM.PRT.P5.04

Milestone Due Date: 9/30/2012

Milestone Completion Date: 9/30/2012

Description of Milestone:
Demonstration of embedded Sn/MOC capability on defined problem (needs TBD), including results analysis with user and computational resource requirements, to support FY12 decision on viability and support of embedded Sn/MOC for meeting FY13 full-core pin-resolved transport milestone.

Completion Proof of the Milestone:
See the attached report detailing the coupled MoC-Sn results that satisfied this milestone.

Tasks to Complete the Milestone:
- We did some early preliminary investigations regarding the feasibility of embedded Sn/MoC methods; we concluded that these would not satisfy CASL neutronics requirements for runtime or accuracy on 3D full core problems.
- As an alternative, we investigated coupling MoC with Cartesian Sn transport. The results of this study are attached to this memo.
- The preliminary results of this study have motivated new coupled 2D/3D transport approaches for full core reactor analysis that we will pursue in FY13.

Person(s) Responsible for Completing the Milestone:
Tom Evans, ORNL; Josh Jarrell, ORNL; Mitch Young, Michigan

Regards,

Thomas M. Evans
Executive Summary

This document discusses the implementation of Method of Characteristics-assisted cross section homogenization for Denovo. Two mesh generation techniques are implemented and results are presented for a number of small test cases.

Contents

1 Introduction 2

2 Method 2

2.1 Mesh Techniques ................................................................. 2

2.2 Generating flux-and-volume weighted cross sections ...................... 3

2.2.1 Total cross section and fission source cross section .................... 4

2.2.2 Fission neutron energy distribution, χ ................................. 5

2.2.3 Scattering cross sections .................................................. 5

2.3 Homogenized solution .......................................................... 5

3 Results 6

3.1 Problem 1: Single Pin, Composite Mesh .................................... 6

3.2 Problem 2: 3x3 array with guide tube ...................................... 9

3.3 Problem 3: 3x3 with guide tube using hybrid mesh ....................... 11

3.4 Problem 4: 3x3 IFBA array with guide tube ............................... 14

3.5 MoC Convergence Study ....................................................... 18

4 Conclusions 19

5 Future Work 19

A Implementation 20

A.1 Cartesian submeshing ......................................................... 20

A.2 FSR indexing ................................................................. 20

A.3 Homogenized solution ....................................................... 21

A.4 Workflow ........................................................................... 21

A.4.1 Composite Mesh .......................................................... 21
1 Introduction

The neutral particle radiation transport code Denovo contains a discrete ordinates ($S_N$) solver, which operates on a strictly orthogonal two- or three-dimensional spatial mesh. While sufficient for many systems, the orthogonal mesh presents challenges when attempting to model spatially complex systems which are not inherently rectangular, such as nuclear reactor cores. Currently Denovo handles such systems by applying an orthogonal mesh to the problem geometry and calculating volume weighted cross sections within each mesh element. Accurately modeling some systems, such as IFBA (integral fuel burnable absorber) pins, requires a prohibitively fine mesh to achieve accurate results. Recently, the two-dimensional Method of Characteristics (MoC) approach was incorporated into Denovo to allow for solutions on more general mesh. In this paper, we investigate the feasibility of using an MoC-obtained scalar flux solution to generate volume-and-flux weighted cross sections for use in Denovo’s $S_N$ solver. This initial work is restricted to two-dimensional reactor lattice geometries.

2 Method

2.1 Mesh Techniques

Two approaches to merging the MoC and $S_N$ geometries were considered. We label these the “composite” and “hybrid” mesh techniques, and each handles the communication between the MoC and $S_N$ meshes in a different manner.

The composite technique forms a completely new mesh on which to perform the MoC calculation which is a union of the conventional MoC geometry and the orthogonal mesh typically used in Denovo. This results in a direct relationship between each flat source region (FSR) to a parent orthogonal mesh element. In this case, the mapping from the MoC flat source regions to the orthogonal mesh elements is many-to-one. Fig. 1 depicts the composite mesh flat source regions and orthogonal mesh for a simple two-ring geometry.

In contrast, the hybrid mesh uses a conventional MoC mesh geometry for the MoC calculation in which the actual flat source regions are allowed to span multiple orthogonal mesh elements. The mapping in this
Figure 1: A simple example of a composite mesh. Contiguous colored regions represent individual flat source regions in the MoC calculation, of which the first 40 are labeled. The orthogonal mesh is ultimately used for the $S_N$ calculation.

case is many-to-many. In other words, the scalar flux that belongs to a flat source region in the MoC mesh is assumed to be constant, regardless of which orthogonal element it is in, and can be applied to multiple orthogonal regions. The MoC solution has no knowledge of the orthogonal grid. Fig. 2 shows the hybrid mesh flat source regions. Note that the flat source regions span multiple Cartesian mesh regions. The flux within each colored region is used in all of the Cartesian elements that they touch.

2.2 Generating flux-and-volume weighted cross sections

Weighting cross sections by flux as well as volume conserves reaction rates within a coarse mesh region for any average flux within the coarse mesh region so long as an accurate fine mesh flux is used to perform the homogenization. The goal to preserve the overall reaction rates within each cell motivates the following sections.
Figure 2: A simple example of a hybrid mesh. Contiguous colored regions indicate the flat source regions that are used in the MoC calculation. The orthogonal mesh is ultimately used in the $S_N$ calculation.

2.2.1 Total cross section and fission source cross section

The total and fission cross sections, as well as fission neutron production factor, $\nu$ are homogenized in order to preserve the reaction rates from the finer-mesh MoC solution. The total interaction rate is given by the formula

$$R = \Sigma_t \phi V,$$

where $\phi$ is the scalar flux, $\Sigma_t$ is the total macroscopic cross section and $V$ is the volume of the region of interest. To homogenize the cross sections for the individual flat source regions within a parent Cartesian mesh element, the following sum is used

$$\Sigma_{t,V} = \frac{\sum_{i \in V} \Sigma_{t,i} \phi_i V_i}{\sum_{i \in V} \phi_i V_i}.$$
A similar equation holds for the homogenization of the fission cross section and the neutron production factor, $\nu$.

$$\nu \Sigma_{f,V} = \frac{\sum_{i \in V} \nu \Sigma_{f,i} \phi_i V_i}{\sum_{i \in V} \phi_i V_i}$$

(3)

2.2.2 Fission neutron energy distribution, $\chi$

Homogenizing the $\chi$ distribution requires different treatment than the other cross sections since we wish to preserve the neutron production rates in each group. The value of $\chi$ in each orthogonal mesh is calculated as the weighted average using the total fission rate in each subregion as the weighting factor. Since the fission rate in subregion $i$ and group $g$ is given as $\phi_{i,g} \nu \Sigma_{f,i,g} V_i$, we get the expression

$$\chi_{g,V} = \frac{\sum_{i \in V} V_i \chi_{i,g} \sum_{g' \in G} \phi_{i,g'} \nu \Sigma_{f,i,g'}}{\sum_{i \in V} V_i \sum_{g' \in G} \phi_{i,g'} \nu \Sigma_{f,i,g'}}.$$  

(4)

2.2.3 Scattering cross sections

The scattering matrix is homogenized based on the from-group flux. In other words, the scattering moments associated with $\Sigma_{s,g'g}$ (scattering from group $g'$ to group $g$) are weighted by the flux in group $g'$. The expression used to calculate the homogenized scattering cross sections is

$$\Sigma_{s,g'g}^l = \frac{\sum_{i \in V} \Sigma_{s,g'g}^l \phi_{g',i} V_i}{\sum_{i \in V} \phi_{g',i} V_i}$$

(5)

for the $l$–th moment from group $g'$ to group $g$.

2.3 Homogenized solution

Once the solution from the MoC calculation has been used to homogenize the material cross sections in each orthogonal mesh region, the resulting homogenized cross sections are used to construct a simple model which is representative of the original pin cell(s). As implemented, each orthogonal mesh element gets a unique material which contains the homogenized cross sections, which are calculated using the above formulas. For example, a single pin with an 8×8 Cartesian mesh would be represented as an 8×8 array of squares, each of which containing their respective homogenized cross sections.


3 Results

Several test cases were developed to investigate the effectiveness of the MoC-flux-weighted method. The goal was to quantify the accuracy benefit obtained by using the MoC-flux-weighted method. The test cases consist of a single pin cell, a 3x3 array of pins with a control rod guide tube in the center, and a 3x3 array of pins which incorporates several IFBA (integral fuel burnable absorber) pins.

Unless otherwise noted, all of the $S_N$ solutions used a quadruple range angular quadrature with 8 polar and 8 azimuthal angles per octant. Both the MoC and $S_N$ calculations used 8-group cross section data and isotropic scattering.

3.1 Problem 1: Single Pin, Composite Mesh

The single pin case was comprised of a single fuel pin with a helium gap and zircaloy cladding, surrounded by water moderator (see Fig. 3). A reference $S_N$ solution was obtained by using a fine spatial grid (100x100), giving an eigenvalue of 1.054759. The fast group (group zero) flux distribution from the reference solution is presented in Fig. 4. Pin-homogenized results were obtained at varying levels of spatial discretization, ranging from a 4x4 mesh to a 24x24 mesh. Fig. 5 shows the behavior of the error in the eigenvalue as the mesh is refined.

The results indicate some expected, yet important behavior. Most notably, the error incurred by the flux-and-volume-weighted $S_N$ solution is about one third of that incurred by the $S_N$ solution using only volume-weighted cross sections for the 4 × 4 mesh. While this is a great reduction in error, the MoC-assisted solution still has an error of about 400pcm. Upon refinement of the mesh, the error in both solutions decreases and the MoC-assisted solution is reliably smaller than that of the normal $S_N$ solution. At the finest mesh refinement investigated, both methods converge to the same result.

It is interesting to note that at coarse mesh refinements, the error in the MoC solution itself is significant.
This is because the MoC mesh relies on the same Cartesian mesh as the $S_N$ solution to obtain an accurate solution. The two solutions are therefore converging in tandem. In other words, the flux from the MoC calculation is relatively inaccurate, meaning that the homogenized cross sections will contribute somewhat to the error in the following $S_N$ calculation. This leads to the $S_N$ solution having a relatively constant additional error over the MoC solution (in this case about 100 pcm). The use of more efficient meshing techniques could result in a more accurate MoC solution and therefore a more accurate $S_N$ solution for the coarse mesh.
Figure 4: Group 0 flux distribution from the fine-mesh reference solution for Problem 1.

Figure 5: Error incurred by the MoC and $S_N$ solutions versus mesh refinement for the single pin with composite meshing (Problem 1).
3.2 Problem 2: 3x3 array with guide tube

A 3×3 array of pins was used to produce a larger and more complex geometry. The pins in the array are identical to those discussed in the single-pin case, with a water-filled control rod guide tube in the center, as depicted in Fig. 6. A reference solution was obtained using a 200×200 per pin Cartesian mesh and traditional, volume-weighted cross sections. The reference solution produced an eigenvalue of 1.012737. The fast flux distribution is presented in Fig. 7.

Fig. 8 shows the error in the eigenvalue as the 2D Cartesian mesh is refined. Actual data can be found in the included excel workbook. The error in this case is larger than that of the single-pin case, though we observe a similar reduction in error over the volume-weighted solution. As with the single-pin case, we see similar behavior in the error from the flux-and-volume-weighted $S_N$ solution and the MoC solution that was used to generate the cross sections. There appears in both cases to be a relatively constant additional error that is incurred by the $S_N$ calculation above the error associated with the MoC solution.
Figure 7: Group 0 flux distribution from the reference solution for problem 2.

Figure 8: Error incurred by the MoC and $S_N$ solutions versus mesh refinement for the non-IFBA 3x3 array and composite meshing.
3.3 Problem 3: 3x3 with guide tube using hybrid mesh

The same problem as described in §3.2 was analyzed using the hybrid mesh technique with a more converged MoC solution for each cartesian mesh refinement. A comparison of the solution convergence is presented in Fig. 9.

![Figure 9: Error comparison between hybrid and composite mesh techniques using the 3x3 non-IFBA array.](image)

At a glance, the results do not indicate a significant benefit from the hybrid mesh technique. However, it is important to consider the number of flat source regions necessary to perform the preliminary MoC solution. Under the composite mesh scheme, the number of flat source regions scales strongly with the 2D mesh refinement, while the hybrid mesh is not dependent on the number of orthogonal mesh elements that are used for the $S_N$ calculation (Fig. 10). For example, in the $24 \times 24$ mesh case, the composite mesh resulted in a total of 8,341 FSRs, while the hybrid mesh produced only 588. Using traditional MoC mesh geometry, a more accurate flux distribution can be obtained on a coarser mesh, which greatly improves computational efficiency. In contrast, the composite mesh requires additional FSRs as the orthogonal mesh is refined, adding significantly to the cost of the MoC calculation. A simple figure-of-merit (FOM) can be defined to quantitatively compare the two methods, which is based on the error in the eigenvalue, $\epsilon$ and the number of FSRs needed to perform the MoC calculation, $N$,

$$\text{FOM} = \frac{1}{\epsilon N} \times 10^5. \quad (6)$$

It is important to note that this figure of merit does not take into account the cost of the $S_N$ calculation, since it is the same for either meshing technique. Fig. 11 shows the FOM of the composite mesh and the
hybrid mesh for various orthogonal meshes. While the composite mesh maintains an almost constant, poor FOM for all of the orthogonal mesh levels, the hybrid method becomes significantly more efficient as the mesh is refined.

Another less obvious advantage to the hybrid mesh method is that it avoids the generation of very small flat source regions which are created when rings in the problem geometry are very close to intersections in the orthogonal mesh. Such situations result in a problem that is difficult to accurately trace with rays, possibly requiring prohibitively fine ray tracing in order to sample all of the flat source regions. These small FSRs have proven to be enough of an issue with the composite mesh that the hybrid technique will most likely be the only practical approach for realistic reactor problems.

![Graph showing the dependence of the number of FSRs on the cartesian mesh for the non-IFBA 3x3 array. The hybrid mesh used a constant 588 flat source regions.](image)

Figure 10: Dependence of the number of FSRs on the cartesian mesh for the non-IFBA 3x3 array. The hybrid mesh used a constant 588 flat source regions.
Figure 11: Figure-of-merit comparison between composite and hybrid meshes using the non-IFBA 3x3 array.
3.4 Problem 4: 3x3 IFBA array with guide tube

Another 3-by-3 pin array with several pins containing integral fuel burnable absorber (IFBA) layers was modeled using the hybrid mesh technique. Fig. 12 depicts the approximate geometry of the problem. A reference solution was obtained using the standard Denovo solver on a 200x200 mesh for each pin. A quadruple range quadrature with 8 polar and 8 azimuthal angles per octant was used for the reference solution as well as the coarse-mesh solutions. The reference solution produced an eigenvalue of 0.800719. The MoC mesh that was used to homogenize the cross sections contained 8 rings in the fuel region, 8 in the IFBA layer, a single ring for the helium gap and 4 rings for the cladding. The moderator region had 8 additional rings between the outside of the cladding and the edge of the pin cell. This MoC mesh was used to determine the flux distribution used for all of the orthogonal-mesh $S_N$ solutions.

Figs. 13 and 14 show the observed behavior of the eigenvalue and the error with respect to the reference solution as the Cartesian mesh is refined. Unlike the non-IFBA case, we observe a degree of variability in the convergence of the eigenvalue as the Cartesian mesh gets finer. Even so, the overall error tends to be comparable to that in the non-IFBA cases, which indicated that the flux-weighted method might be an effective means with which to approach pin-resolved transport. As indicated by Fig. 13, the non-flux-weighted solution consistently under predicts the eigenvalue by as much as 1,000 pcm. This occurs because the normal volume-weighted cross sections do not account for the flux depression that would be present in any strongly absorbing region, leading to an artificially larger homogenized cross section. This leads to an over prediction of absorption and therefore and under-predicted eigenvalue. Since conservative estimates of a system’s eigenvalue are typically desired, this can be a serious concern for reactor analysis and associated criticality safety analysis.

The erratic behavior in the eigenvalue is likely caused by streaming effects associated with the large discontinuities in the material cross sections in adjacent orthogonal mesh elements. Those elements which contain a piece of IFBA end up having a much higher total cross section than their non-IFBA neighbors. Since the IFBA layer is very thin, it is easy to have a configuration in which the corners of two mesh elements containing IFBA meet, creating an artificial streaming path. Geometry/orthogonal mesh combinations with more pronounced streaming paths tend to experience more error.
Figure 12: Approximate problem geometry for 3x3 IFBA case. Red pins indicate bare fuel, yellow indicate IFBA pins and the center location is a water-filled control rod guide tube.

Figure 13: Eigenvalues as calculated using flux-weighted and non-flux-weighted cross sections for the 3x3 IFBA case.
Figure 14: Absolute value of error in the eigenvalue of the 3x3 IFBA case.

Figure 15: Group 0 flux distribution from the 10x10-meshes-per-pin solution for the 3x3 IFBA case.
Figure 16: Group 7 flux distribution from the 10x10-meshes-per-pin solution for the 3x3 IFBA case.

Figure 17: Group 7 flux distribution from the reference solution for 3x3 IFBA case.
3.5 MoC Convergence Study

With the introduction of the hybrid mesh, it becomes more important to consider the degree to which the independent MoC solution has converged. In order to gain an understanding of the error behavior of the MoC calculation for the 3x3 case considered in 3.2 and 3.3 a simple convergence study was performed. The MoC mesh was gradually refined for several permutations of ray spacing and the number of segments per ring. Fig. 18 shows the convergence behavior of the system $k_{\infty}$.

![Figure 18: MoC solution convergence by number of mesh elements. Data are labelled by [ray spacing], [number of segments per ring].](image)

Unfortunately, the hybrid mesh technique has not yet been developed to support 8 segments per ring, but those cases were included in the study to act as somewhat of a reference solution. The case with 4 segments per ring and a ray spacing of 0.01cm was ultimately selected for use the analysis performed in 3.3, since it struck a balance between accuracy and computational cost.
4 Conclusions

This work examined the effectiveness of using flux-and-volume-weighted cross sections for modeling reactor fuel pins with relatively coarse orthogonal meshes for the discrete ordinates calculation. The goal was to characterize the error behavior of the method as the orthogonal mesh is refined, as well as to explore the effectiveness of two different mesh generation techniques.

For the small 2D cases, significant error reductions were achieved by using the MoC flux distribution to flux weight the cross sections. For the non-IFBA 3x3 pin case with a 4x4 mesh for each pin, the error was reduced from 1,500 pcm to about 660 pcm using the composite mesh. With an 8x8 mesh per pin, the error was reduced from about 660 pcm to approximately 380 pcm. These results indicate that the method can be relatively effective for reducing the error associated with using homogenized cross sections. Error is still relatively high for coarse meshes, so further analysis may be needed to determine if such coarse models can be used practically.

Between the two meshing techniques (hybrid/composite), the hybrid technique appears more practical. The small FSR problem with the composite mesh is enough of a practical issue to preclude its use in real-world reactor problems. Beyond the practical issue of the small FSRs, increased mesh efficiency is also gained by using the hybrid mesh, further encouraging its use. The traditional MoC mesh is more computationally efficient since the shape of the mesh elements lies along our a priori notion of what the flux distribution looks like in a fuel pin. As a result, the flux remains more constant within each mesh element, leading to a more accurate result for the same number of elements. Another benefit of the hybrid mesh approach is that it allows the user to converge the MoC solution independently from the $S_N$ solution. In the composite mesh, the accuracy of the MoC solution is dependent upon the coarseness of the orthogonal grid, and therefore both solutions will converge in tandem as the orthogonal grid becomes more fine. This behavior makes it difficult to characterize which solution step is dominating the error.

5 Future Work

The results from this work are promising enough to motivate further exploration. Extension of the 2D homogenized cross sections into 3D problems could result in a useful 2D/3D methodology for performing whole-core calculations.

The hybrid mesh approach could potentially be used for performing core depletion studies in which the local fine-mesh MoC solution is used to generate homogenized cross sections for a global $S_N$ solution.
Following the $S_N$ calculation, the global flux distribution could be projected back onto the MoC mesh, in which isotopic changes can be calculated. Once new isotopics are determined, the process can be repeated to arrive at the next state point.

A Implementation

A.1 Cartesian submeshing

To facilitate spatial collapsing of cross sections for a pin cell using a method of characteristics solver, it was necessary to modify the MoC capability in Denovo to impose a Cartesian grid over the standard MoC FSR mesh, thus making sets of FSRs which fit exactly into the orthogonal mesh used by the KBA algorithm. This functionality was implemented by introducing the following changes to the code:

- A new attribute was added to the `Pin` class, `num_cart`, which stores the number of orthogonal meshes to lay across the pin in each direction. The default value is 1, and any other value specified must be even.

- The `rtk_maps` class has a new member, `d_cart_id`, which stores which Cartesian mesh element within a pin that an FSR is in. This is needed later for homogenization.

- The functions in `KENO_Asembly_Generator` which build the `RTK_Map` and the keno input file are altered to either branch on `num_cart` or are redefined to handle the extra requirements of the Cartesian submeshing.

There are also several auxiliary python functions in `Hom_Utils.py` which assist in the cross-section homogenization process.

A.2 FSR indexing

The flat source regions are indexed differently in pins which make use of the Cartesian submesh than in pins which do not. This is because the Cartesian submesh routines operate on a 1/4 symmetry (since the `Pin_Mesher` class is used to find the volume fractions of the FSRs). As such, the geometry is built for the first quadrant and then reflected to the other quadrants. This results in identical, reflected FSRs having the contiguous indices. Fig. 1 depicts the indexing scheme for a sample pin cell using the composite mesh and
Fig. 2 shows the same pin using the hybrid mesh. As with the existing KENO geometry, the assembly gap is added after the array of pins.

### A.3 Homogenized solution

Once the solution from the MoC calculation has been used to homogenize the material cross-sections in each orthogonal mesh region, the resulting homogenized cross sections are used to construct a simple model which is representative of the original pin cell(s). As implemented, each orthogonal mesh element gets its own material type in a material database, and the proper materials are mapped to each region of the mesh. This process is performed by a python function, called `hom AMPX`, which reads in the original AMPX cross-section library that was used for the MoC calculation as well as the FSR-to-Cartesian-mesh map which is outputted from the modified `build rtk maps()` function in the `KENO Assembly Generator` class. `Hom AMPX` then produces a `PyXS_DB` object containing homogenized cross-sections for each mesh element.

### A.4 Workflow

While it is possible to fully integrate all of the functionality of the methods described in the previous section into Denovo, much of it has been left external to Denovo. As a result, analysis requires several steps. Briefly, pymoc is used to generate a cell mapping file and a flux file. The cell mapping file describes which MoC flat source regions correspond to which orthogonal mesh elements and the volume fraction of the orthogonal element the FSR comprises. The flux file stores the multigroup scalar flux for each FSR.

Below, the workflow associated with each mesh approach is described. Example pymoc and pykba scripts are provided with this document for both methods.

#### A.4.1 Composite Mesh

Since the composite mesh approach incorporates a single mesh that carries the Cartesian discretization and the MoC geometry, the cell and flux files can easily be created in tandem with the generation of the RTK maps. This allows a single pymoc run to set up the mesh and perform the MoC calculation itself. The pymoc scrips simply needs to apply a “num_cart” tag to each pin, specifying the number of desired Cartesian mesh elements in each direction. The pymoc script must then generate the flux file using the `print_flux()` function in `HomUtils.py`. A pykba script can now be written to prepare homogenized cross sections using the `hom_ampx()` function in `HomUtils.py`.
A.4.2 Hybrid Mesh

The hybrid mesh approach is somewhat more complicated since the same mesh is not being used to obtain the MoC solution and impose the Cartesian mesh. This adds the requirement of needing to create the cell file separately from the pymoc script used to determine the flux on the MoC mesh. There is a standalone pymoc file that is used just for the generation of the scalar flux, and a separate pymoc file just for generating the cell mapping file for the desired orthogonal mesh. In other words, there is one script for generating the flux file and a different script for generating the cell map. Once the cell map and flux file have been generated, a pykba script can be prepared similarly to the composite mesh.