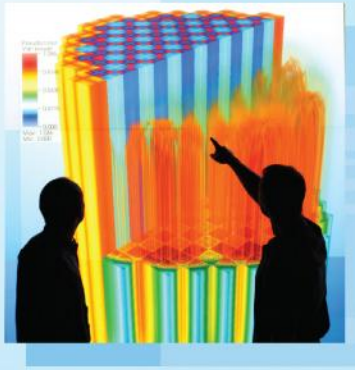
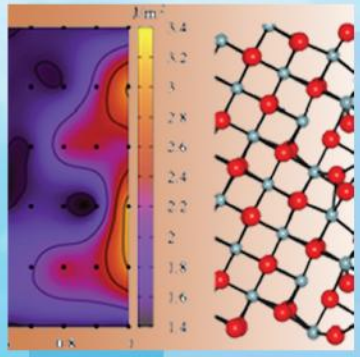
Power uprates
and plant life extension



Engineering design
and analysis

Science-enabling
high performance
computing



Fundamental science

Plant operational data

Consortium for Advanced Simulation of LWRs

VERA Problem 10: Restart and Shuffling in MPACT

Revision 1
February 26, 2015

Brendan Kochunas
Daniel Jabaay
Thomas Downar

U.S. DEPARTMENT OF ENERGY | Nuclear Energy

REVISION LOG

Revision	Date	Affected Pages	Revision Description
0	12/4/2014	All	Initial version
1	2/27/2015	All	Fixed typos from copy-paste in Appendix A, added section 3.3, updated results in Section 5. Additions to Section 4.1. Updated Section 3 and 6

Document pages that are:

Export Controlled None

IP/Proprietary/NDA Controlled None

Sensitive Controlled None

Requested Distribution:

To: N/A

Copy: N/A

EXECUTIVE SUMMARY

This report documents the work performed to meet the objectives of the RTM milestone on implementation of fuel shuffling and the restart capability into MPACT. The principle objective of the milestone was to add the capability for MPACT to write and read a restart file when running with an arbitrary number of processors, as well as to provide functionality to shuffle the written assembly information to other assembly locations. This capability is necessary for modeling multi-cycle core depletion cases since assemblies are shuffled in the core after a single burnup cycle is completed.

These features were added by creating a new module within MPACT that provided the necessary read and write capabilities. This feature is different than the checkpoint file previously implemented in MPACT which was able to be written at any time during the eigenvalue calculation. The new multi-state solver object within MPACT controls the execution of each state point eigenvalue solve, so this object is the client of the restart file object. The primary data being written to the file is cross section information which exists on the cross section mesh object. The file object also requires knowledge of the problem domain and geometry and therefore depends on both the `XSMesh` and `ModularMesh` modules. The shuffle file is written and read to/from the HDF5 files, so it will also depend on the `HDF5_FileType` module.

The testing of this feature was performed progressively, starting with a single level single pin and increasing in size and complexity to VERA Problem 10 which was the full core depletion of cycle 1 of Watts Bar and restart/fuel shuffling to prepare in the input for cycle 2 depletion.

Problem 10 was simulated by first simulating problem 9. Cycle 1 was simulated on Falcon using 4234 processes and a total wall time of nearly 30 hours and nearly 60% of that time being attributable to COBRA-TF. Good agreement was observed for the cycle 1 reactivity. Cycle 2 used shutdown decay and solved HZP and HFP conditions. Because no reference solution exists yet, it is difficult to ascertain the accuracy of the Cycle 2 solutions.

TABLE OF CONTENTS

REVISION LOG ii

EXECUTIVE SUMMARY iii

TABLE OF CONTENTS iv

FIGURES v

ACRONYMS vi

1. Introduction 1

2. Input Changes 2
   2.0 Input Cards 2
   2.1 Card Descriptions 2

3. File Structure and Implementation 4
   3.0 Restart File 4
      3.0.0 Depletion region file indexing 6
   3.1 Object Design 8
   3.2 Parallel Read and Write 11
      3.2.0 Overview of Parallelism in MPACT 11
      3.2.1 Parallel Writing Communication 11
      3.2.2 Parallel Reading 13
   3.3 Symmetry 13

4. Integral Testing 15
   4.0 Unit Testing 15
   4.1 Verification Testing 15

5. Simulation of VERA Benchmark Problem 10 18
   5.0 Model Description 18
   5.1 Results and Discussion 21
      5.1.0 Cycle 1 Results 21
      5.1.1 Cycle 2 BOC results 23
      5.1.2 Performance of Isotopic Restart File I/O 26

6. Conclutions and Future Work 27
   6.0 Conclusions 27
   6.1 Future Work 27

REFERENCES 28

Appendix A - VERA Input 29

Appendix B - T/H Parameter Sensitivity 54

FIGURES

Figure 1: Shuffle Label Syntax Description 2
Figure 2: Illustration of Assembly 1-D Pin Indexing in Restart File 7
Figure 3: Illustration of Pin-local Composition Region Indexing in Restart File 7
Figure 4: Fortran code snippet illustrating looping structure for assembly composition data 8
Figure 5: UML Legend 9
Figure 6: Fuel Shuffle File Software Architecture Diagram 9
Figure 7: Software Design for MPACT Fuel Shuffle File Implementation 10
Figure 8: Example of a Two Plane Assembly Parallelized in Space, Angle, and Ray in MPACT[2] 11
Figure 9: Illustration of Mirror and Rotational Symmetry unfolding 13
Figure 10: VERA Problem 5 Assembly, Poison, and Control Rod Layout in Quarter Symmetry[3] 18
Figure 11: Problem 10 WBN1 Cycle 2 Core Loading Pattern[3] 19
Figure 12: Radial partitioning in MPACT 20
Figure 13: Comparison of Cycle 1 Critical Boron 21
Figure 14: Relative Component Run Times of VERA-CS with 4234 cores on Falcon 23
Figure 15: Cycle 2 BOC Relative Fission Rate Distribution at Hot Zero Power 24
Figure 16: Cycle 2 BOC at Hot Full Power 25

TABLES

Table 1: VERA Common Input Card Descriptions 2
Table 2: HDF5 Restart File Structure 5
Table 3: Symmetry Translation Operations 14
Table 4: Verification Tests 16
Table 5: Symmetry Verification Unit Tests 17
Table 6: Cycle 1 Simulated Operating History 20
Table 7: Measured and Simulated Cycle 1 States 22
Table 8: Calculation Component Total Run Times for Cycle 1 (Including Restarts) 22
Table 9: I/O Performance of Isotopic Restart File for a single state with 2378 processors on EOS .. 26

ACRONYMS

1D	One-Dimensional
2D	Two-Dimensional
API	Application Program Interface
BOC	Beginning of Cycle
CASL	Consortium for Advanced Simulation of Light Water Reactors
EFPD	Effective Full Power Days
EOC	End of Cycle
HFP	Hot Full Power
HZP	Hot Zero Power
IFBA	Integral Fuel Burnable Absorber
I/O	Input / Output
MPACT	Michigan Parallel Characteristics Transport Code
ORNL	Oak Ridge National Laboratory
PPMB	Part Per Million Boron
QA	Quality Assurance
T/H	Thermal-Hydraulics
UM	University of Michigan
UML	Unified Modeling Language
VERA	Virtual Environment for Reactor Applications
VERA-CS	Virtual Environment for Reactor Applications – Core Simulator
WABA	Wet Annular Burnable Absorber

1. INTRODUCTION

The reactor core simulator for VERA-CS in the Consortium for Advanced Simulation of Light Water Reactors (CASL) is the MPACT code which is now jointly developed by UM and ORNL. The key characteristics of MPACT include the subgroup method and the embedded self-shielding method (ESSM) for resonance treatment, a whole core solver with a 2-D (MOC)/1-D (Nodal-Diffusion) synthesis method, and the matrix exponential method for burnup based on the SCALE-ORIGEN solver. As part of the CASL core simulator, VERA-CS, MPACT is coupled with COBRA-TF for thermal-hydraulic feedback and ORIGEN for tracking the isotopic inventory to provide the core depletion capability.

Light Water Reactors operate for a burnup or depletion cycle which typically lasts a year or more. At the end of a cycle the core is shutdown, and undergoes an outage where maintenance is performed, new fuel is loaded, and current fuel assemblies are shuffled into different locations or removed from the core. In order to model the core and deplete for more than one cycle with VERA-CS, the following capabilities are required by MPACT:

- Store the depleted fuel compositions for subsequent calculations
- Load these fuel compositions easily into an arbitrary user specified fuel assembly location within the core

An automated fuel shuffling capability was implemented in MPACT to minimize user requirements and to obviate the need for the user to print and input the isotopic vector for each depletion zone in each fuel assembly for the entire core.

The previously implemented capability for MPACT to continue running a case was limited to the checkpoint file. This capability allowed for a to be written during the iterative solution process or after convergence of a steady state solution. This feature allowed a user to circumvent machine wall time limits by specifying a checkpoint prior to the end of the allotted time and then start a new job with refreshed wall time using the checkpoint file. However, a key limitation of this capability was that it did not permit anything about the case to be changed. Additionally, it has been observed that the parallel write operation for this file did not scale particularly well, most likely because of contention for I/O resources. A more general "restart" capability was designed based on the concept of an "isotopic restart" which enables a user to load in fuel compositions from a database. Not only does this provide a restart capability, but it facilitates fuel shuffling for multi-cycle analysis and restarts for branch calculations, both of which are necessary code features of a core simulator required to meet the needs of CASL challenge problems. Therefore, an additional feature has been developed and added to MPACT that enables fuel shuffling and restart as a part of the scope of this milestone.

The objectives of this milestone were to create the restart file structure, implement the ability to read and write such a file in parallel, have MPACT be able to shuffle the file data, and demonstrate this feature by running a two cycle depletion problem of the Watts-Bar reactor core, also known as the VERA Benchmark Progression Problem 10.

The following section of the report describes the required changes to the VERA input. The file structure and details of the implementation are provided in Section 3. The feature testing is described in Section 4. Then Section 5 presents the demonstration of the capability for Problem 10. Finally the work for this milestone is summarized and areas of future work are given.

2. INPUT CHANGES

2.0 Input Cards

The addition of restart files and fuel shuffling required more input cards both in the VERA common input as well as MPACT's XML input processor. The input cards were specified in the document "Initial Specification for Core Shuffling" [1]. The list of VERA common input cards and a summary of their descriptions are:

Table 1: VERA Common Input Card Descriptions

Card Name	Card Inputs	Block	Input Format
restart_write	<restart_file> <restart_label>	[STATE]	(string, string)
restart_read	<restart_file> <restart_label>	[STATE]	(string, string)
restart_shuffle	<restart_files> <restart_labels>	[STATE]	(string array, string array)
shuffle_label	<array of assembly x and y labels>	[STATE]	(string array)
op_date	<date string>	[STATE]	(string)
cycle	<core cycle number>	[CORE]	(string)
unit	<core unit number>	[CORE]	(string)
rotate_map	<array of assembly rotation values>	[CORE]	(integer array)

2.1 Card Descriptions

The cards **`restart_write`**, **`restart_read`**, and **`restart_shuffle`** require a filename and a label name that will both be used in the respective operation. When writing a restart file, the filename may be either a new file or existing file. If an existing file is being used, then the label name must be different than any pre-existing labels in the specified file. When reading or shuffling from a restart file, both the filename and the label name must refer to existing files and datasets, respectively.

The **`shuffle_label`** card is an array in the same shape as any core map input, but the syntax of the assembly labels is very specialized. An assembly label in the map not only specifies the previous assembly to place in the current core position, but also the unit and cycle as well. An example for an assembly from unit two cycle four is shown below:

```
<unit number>  <cycle number>   <x label>   <y label>

                3A-03 2:4H-14 1:G-07

                      2:4H-14
Syntax: <unit number>:<cycle number><X Label>-<Y Label>
```

Figure 1: Shuffle Label Syntax Description

The unit and cycle numbers are optional; if the cycle number is not specified in the label it is assumed to be from the previous cycle and the input card "cycle" is then used and reduced by one. The cycle number and unit number are used to compare information from within the shuffle file to ensure the correct assembly information is retrieved. The **`op_date`** card can be specified for each

state point. For core shuffling, it is a required input. The **`cycle`** card is required for any depletion case, and the unit card is optional and defaults to “1”.

The **`rotate_map`** card is also an array that is the same as a core map input and is only used when shuffling. When writing a restart file, the current rotation integer for that assembly position is written to the file. During a core shuffle, the input `rotate_map` value indicates the number of clock-wise 90° rotations from the orientation specified by the assembly's geometric description given in the `ASSEMBLY` block of the input. This rotation value is then used to compute the relative rotation to be applied to the assembly data based on the orientation assembly's data when the entry was written. Therefore, positive differences mean a clockwise rotation, and negative differences mean a counter-clockwise rotation. A more detailed description of the inputs and their nuances can be found in the full input specification document [1].

3. FILE STRUCTURE AND IMPLEMENTATION

3.0 Restart File

The restart file structure was created to store the fuel assembly compositions in such a way that the data is tied to the input geometry and independent of the simulation mesh, parallel decomposition or symmetry, thus allowing for the data to be easily used in a variety of downstream simulations and analyses that may not even involve the same tools that generated the data.. When restarting a case, the only data truly needed is the isotope number densities and cross section library indices. Additional data is needed for fuel shuffling, such as rotation, cycle number, and assembly labels. The type of restart file was chosen to be a binary HDF5 [5] file to minimize the way the file stores the data in addition to limiting the amount of data written. The structure of the HFD5 restart file is shown below in Table 2.

The format for the file is arranged for storing several different state points for a given core which are specified by the "`RESTART_<label>`". The root directory in the HDF5 file is given by the name of the core from the input file. The `ZAID` list is needed because it is assumed that the same cross section library may not be used when loading composition data.

For each restart label, more information is stored since each restart can be more unique. Each restart requires the cycle number since it is possible to store multiple cycle's worth of data for each core. The same logic applies for the unit number since most nuclear power plant sites have more than one reactor. The operation date is written so isotopic information from decay time can be obtained from fuel that was discharged from one cycle, stored in the spent fuel pool for a certain amount of time, and then reintroduced into another cycle. The number of assemblies in both the x and y coordinate directions and the x and y labels for identifying assembly locations within the core exist if the user at some point desired to switch between full and quarter symmetry when running a problem.

The bulk of the data is provided for each unique fuel assembly. The original physical rotation index of the assembly is stored to indicate the orientation of the assembly within the core when the data was written. Additionally, a symmetry option is included, and this is required because a simulation may be run with symmetry and an assembly can be shuffled into a different quadrant requiring that the orientation of the compositions within the assembly preserve any symmetry. The serial number is specified as a way for the user to track a specific known assembly. Shuffling by serial number is not currently implemented.

The assembly label is taken from the assembly position and used to lookup the corresponding x and y label values. Currently, the assembly type is not stored, but will be added in the future The number of pins in the x, y, and z direction is stored so the dimensions of `nxsreg_pin` are known. This information will be used to loop over the data structure. The initial heavy metal mass is needed for the bulk assembly burnup calculation. The total assembly exposure, or burnup, is needed to keep track of how long the assembly has been burned in the reactor. This value will become especially important when multi-cycle analysis and equilibrium cycle analysis are performed.

Table 2: HDF5 Restart File Structure

CORE_<Unit Label>	RESTART_<restart_string>	Assembly_<IAF>_<JAF>	Data Format
Zaid list	-	-	(integer array)
RESTART_<restart_string>	-	-	(HDF5 Group)
-	**version**	-	(integer)
-	**Cycle Label**	-	(string)
-	**Unit Number**	-	(string)
-	**Op Date**	-	(string)
-	**Nxasy**	-	(integer)
-	**Nyasy**	-	(integer)
-	**Xlabel**	-	(string array)
-	**Ylabel**	-	(string array)
-	**Assembly_<IAF>_<JAF>**	-	(HDF5 Group)
	-	**symmetry**	(string)
-	-	**Rotation**	(integer)
-	-	**Serial Number**	(string)
-	-	**Label**	(string) not used
-	-	**Assembly Type**	(string) not used
-	-	**Nxpin**	(integer)
-	-	**Nypin**	(integer)
-	-	**Nz**	(integer)
-	-	**Initial Heavy Metal Mass**	(real)
-	-	**Total Assembly Exposure**	(real)
-	-	**Nxsreg_pin**	(integer)
-	-	**Isotope Index Map**	(integer array)
-	-	**Nuclide ID**	(integer array)
-	-	**Fuel Composition**	(real array)
-	-	**Axial Mesh**	(real array)
-	-	**Axial Heavy Metal Mass**	(real array)
-	-	**XS Region Volume**	(real array)
-	-	**Pin-average Exposures**	(real array, optional)
-	-	**Crud Values**	(real array, optional)
-	-	**Gap Values**	(real array, optional)
-	**INSERT_<IAF>_<JAF>**	-	(HDF5 Group)
	-	**symmetry**	(string)
-	-	**Rotation**	(integer)
-	-	**Nxpin**	(integer)
-	-	**Nypin**	(integer)
-	-	**Nz**	(integer)
-	-	**Nxsreg_pin**	(integer)

-	-	**Isotope Index Map**	(integer array)
-	-	**Nuclide ID**	(integer array)
-	-	**Composition**	(real array)
-	-	**Axial Mesh**	(real array)
-	-	**XS Region Volume**	(real array)

The `nxsreg_pin` is a 3-D array of the number of depletable cross section regions and is therefore independent of composition regions, for a given pin location in the assembly. The isotope index map is similar to `nxsreg_pin` in that it is used to specify the number of isotopes in a given cross section region. The "Nuclide ID" data set contains the indices for the corresponding ZAID integer array, and the "Fuel Composition" data set contains the corresponding number densities for the given isotopes. The axial mesh is specified for the assembly for the case when a user defines a new problem where the mesh is different from the previous case where the restart file was written and interpolation/homogenization becomes necessary. The axial heavy metal mass is written so there can be an axial representation of the burnup distribution within an assembly. The cross section region volume is required for the case where the depletion mesh within a pin changes thus requiring some mapping/homogenization of the data onto the mesh of the new simulation depletion mesh.

Pin averaged exposures are useful for explicitly tracking the high burnup pin(s) within an assembly throughout the assembly's life. However, this data was not included in the current implementation. The current scope of the capability is to only include the compositions of the depletion zones in each fuel pin segment.

The INSERT group is for insert compositions. These cannot be shuffled, but are typically required for restarts since the inserts are depletable. They contain a subset of the information in the Assembly group, essentially everything but information related to exposure.

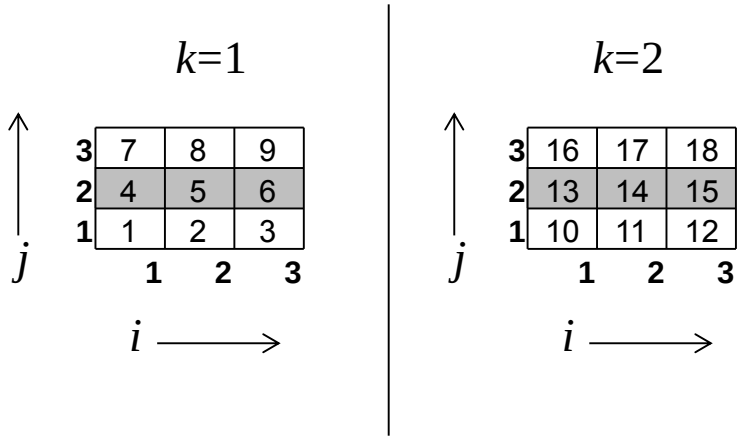
Future work will include adding compositions for control rods and potentially shuffling inserts. Relative to other currently planned future work, PACT will be coupled to a code that provides chemistry feedback, and it will be necessary to track any values relevant to CRUD deposition on the fuel pins such as the composition and heat transfer coefficient of the CRUD layer. MPACT will also be coupled to a fuel performance code or T/H code that potentially needs the restart file to track the fuel conductivity and/or fuel-clad gap conductance or thickness. In multi-physics simulations these values are expected to be needed to perform mid-cycle restarts and multi-cycle analysis, although this data may not be directly used by MPACT.

3.0.0 Depletion region file indexing

To facilitate the usability of the fuel composition data, a new indexing is developed that applies to the data in the file. This indexing is mostly independent of the simulation details of MPACT when the data was generated. The purpose of this section is to describe the indexing used for a given assembly's data within the restart file. This information will be particularly important for anyone wishing to develop a reader.

The 1-D indexing may be used for the pins. It applies to all pin locations and does not depend on whether or not a pin has fuel or not. In other words, for a 17x17 assembly with 50 axial edit regions the dimensions will be `nxpinasy`=17, `nypinasy`=17, `nz`=50. These correspond to the dimensions of `nxsreg_pin` and the 1-D pin indexing will be from 1 to `nxpinasy*nypinasy*nz` , in this example

14450. The 3-D to 1-D indexing is illustrated by Figure 2 for a 3x3 assembly, which also includes the functions for converting the 1-D index to the 3-D index and vice versa.



Convert 3-D to 1-D:

```
ipin = ((k-1)*nypinasy+(j-1))*nxpinasy+i
```

Convert 1-D to 3-D:

```
i = MOD(ipin-1,nxpinasy)+1
j = MOD(ipin-1,nxpinasy*nypinasy)/nxpinasy+1
k = (ipin-1)/(nxpinasy*nypinasy)+1
```

Figure 2: Illustration of Assembly 1-D Pin Indexing in Restart File

The fuel composition edit region indexing is continuous within each pin region and follows the 1-D pin indexing. Presently in MPACT, some pins may have azimuthal dependent cross sections which comes from pins split on the symmetric core boundary. The pin-local 1-D cross section indexing is illustrated in Figure 3. Essentially it goes from the outer-most radii to the inner-most radii and this process repeats for each half or fourth of the pin proceeding counter-clockwise from the northeast quadrant.

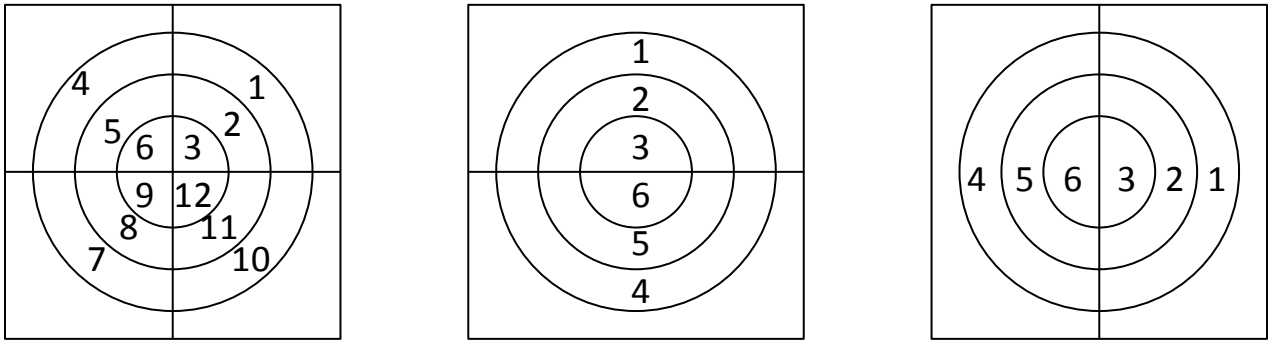


Figure 3: Illustration of Pin-local Composition Region Indexing in Restart File

Finally, Figure 4 illustrates the loop for how to stream through the isotopic data, looping over the composition edit regions consecutively.

```
ixsasy=0
iso=0
DO ipin=1,nxpinasy*nypinasy*nz
  ipinx=MOD(ipin-1,nxpinasy)+1
  ipiny=MOD(ipin-1,nxpinasy*nypinasy)/nxpinasy+1
  ipinz=(ipin-1)/nxpinasy*nypinasy)+1
  DO ixspin=1,nxsreg_pin(ipinx,ipiny,ipinz)
    ixsasy=ixsasy+1
    DO i=1,niso_xs(ixsasy)
      iso=iso+1
      !Nuclide_ID(iso)
    ENDDO
  ENDDO
ENDDO
```

Figure 4: Fortran code snippet illustrating looping structure for assembly composition data

All data in each assembly is stored for the full geometry of the assembly regardless of the problem symmetry at the time the data is written to the file. One memory saving approach that is transparent to any read client of the file is that symmetric assembly locations do not duplicate file data. Instead an HDF5 hard link (reference counted pointer) is created to the data for the symmetric location. For example, if a problem is run with quarter core symmetry, the data for the symmetric quadrants is not duplicated in the restart file, but instead just points to the data in the one quadrant that was modeled. This feature allows the client to read the data for any full core assembly position, but keeps the file size minimized by avoiding the duplication of data. This treatment of symmetry is described in more detail in 3.3.

It is important to note that in the current implementation, the axial edit regions of the assembly above and below the active fuel are counted as well as the regions above and below the top and bottom nozzles. An area of future work is to adjust the contents such that the mesh regions above and below the assembly nozzles are not included in the file.

Another important note is that in the current implementation, the file does not contain information to indicate how many radii or azimuthal sectors there are in each pin location, nor does the file contain radii information. This implementation limits the ability to homogenize or map the data onto a new geometry, but adding these values is planned as an area of future work.

3.1 Object Design

A new object was created within MPACT based on the requirements specified in [1], from which the coding architecture and high level design were derived. Since this file type is a restart file type, the assumption is that the data will only be printed at the end of a state point. This feature is different than the checkpoint file which can be written at any time during the eigenvalue calculation. The multi-state solver object within MPACT controls the execution of each state point eigenvalue solve, so this object will be the client of the restart file object. The primary data being written to the file is cross section information which exists on the cross section mesh object. The file object will also need to have knowledge of the problem domain and geometry. The restart file object will depend on both the `XSMesh` and `ModularMesh` modules. Lastly, the shuffle file will be writing to and reading from HDF5 files, so it will also have a dependency on the `HDF5_FileType` module. Figure 6 and Figure 7 are known as UML class diagrams, which are a type of diagram frequently used to illustrate

features of software. A short key for the meaning of the diagram objects in Figure 6 and Figure 7 is given in Figure 5**.** Figure 6 describes the architecture of the restart file object shown below:

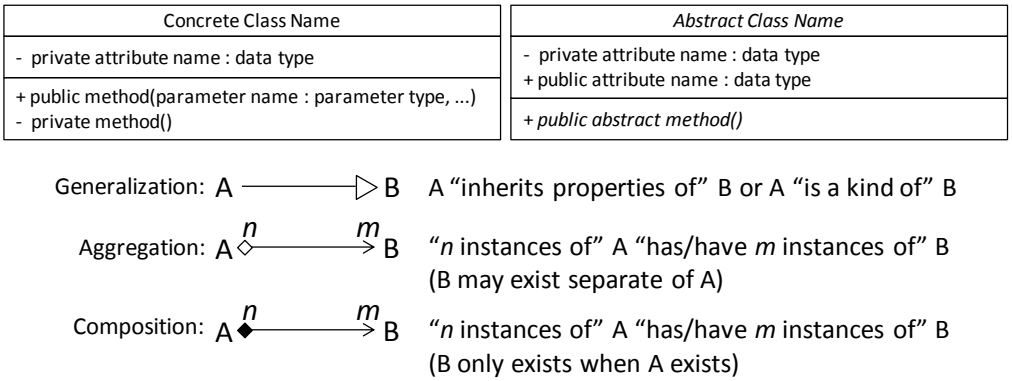


Figure 5: UML Legend

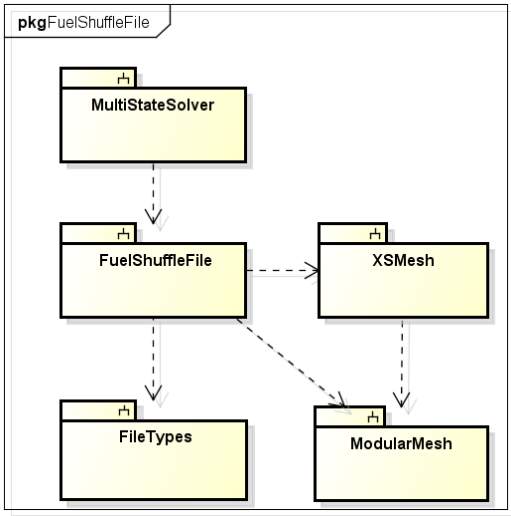


Figure 6: Fuel Shuffle File Software Architecture Diagram

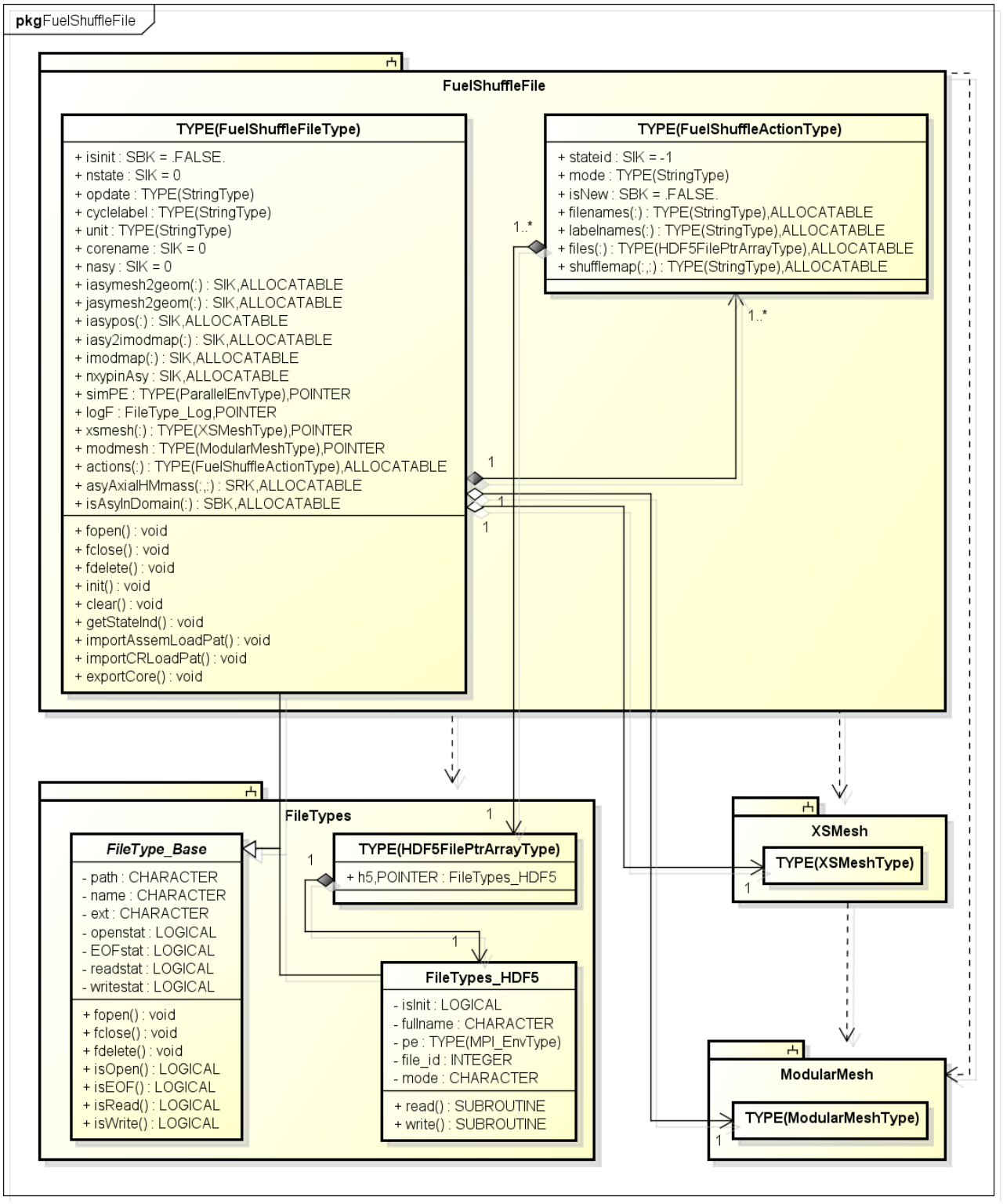


Figure 7: Software Design for MPACT Fuel Shuffle File Implementation

3.2 Parallel Read and Write

3.2.0 Overview of Parallelism in MPACT

In MPACT, there are several ways to parallelize a problem. While MPACT has ray parallelism that uses shared memory based OpenMP, the parallelisms of concern when implementing this restart feature are angle and spatial parallelism which require distributed memory MPI. The primary reason for this distinction is because it is a non-trivial task to write data to a single file while that data is spread across multiple processors that requires communication to consolidate. When angular decomposition is used, the parallel scheme is straightforward. The spatial domain is replicated on the given process with a reduced set of angles over which to perform the neutron transport calculation. Spatial decomposition can be more flexible.

MPACT can be spatially decomposed on an assembly, quarter assembly, or pin-wise basis for varying numbers of axial sections depending on how the model is constructed. For the modeling of the Watts-Bar Unit 1 reactor, MPACT breaks up the spatial domain on a quarter assembly basis for spatial decomposition. It would be possible to decompose on just an assembly basis or even a pin-wise basis if there were the right circumstances (e.g. a core with an even number of assembly in both x and y dimensions, or no inter-assembly gap, respectively). However, the Watts-Bar core geometry does not meet either of these criteria.

A simple illustration of the parallelism is shown in Figure 8 below, where the two by two by two block would be an assembly of two planes:

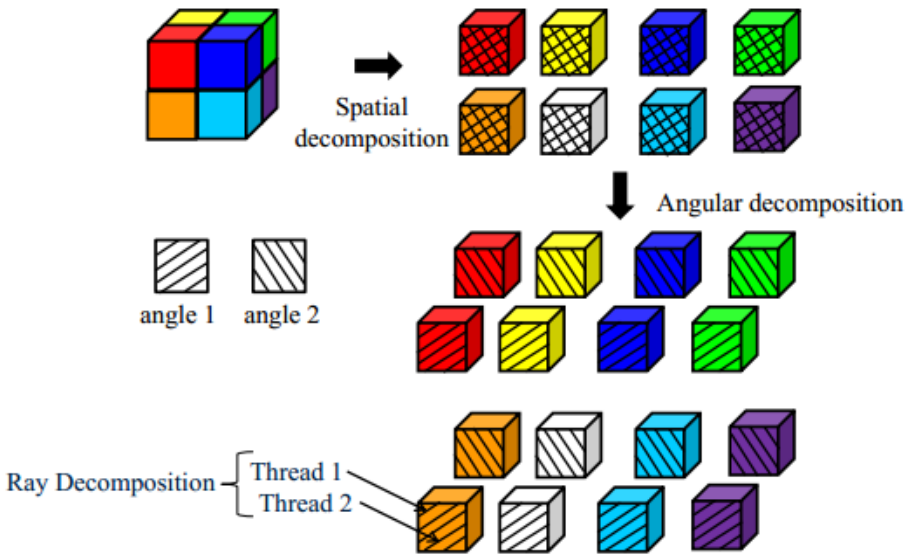


Figure 8: Example of a Two Plane Assembly Parallelized in Space, Angle, and Ray in MPACT[2]

3.2.1 Parallel Writing Communication

The communication and I/O model for the file's write operation has only the root process perform the actual I/O. The spatial domains that are duplicated for angle decomposition do not participate in the write operation at all. Consequently, there are only *nspace* processes participating in the write operation at any one time. For a given assembly, all spatial domains count the number of composition regions and isotope data that are within that assembly, and this information is gathered

on the root process. The gather operation is performed using non-blocking send and receive operations. Note that it is expected that for some decompositions a process may not have any data for a given assembly. Those processes that have data for the assembly then order the data into a send buffer and use non-blocking sends to communicate that information to the root process.

The root process determines which other domains will be sending data to be written using the dimension information communicated initially. Once it receives this information, the root process will allocate receive buffers to collect the incoming data. The root process then allocates I/O buffers to order all the data for the entire assembly for a single write operation.

This algorithm is expected to perform reasonably well because:

- It minimizes collective communication operations
- The gather operations are performed with non-blocking communication, so "useful work" may be done while the messages are in transit.
- The number of I/O operations are minimized.

In terms of the average number of clock cycles for network communication versus disk I/O, the disk I/O will typically be higher. Therefore, in general it is cheaper to send a message than perform additional I/O. This algorithm however still creates a bottle-neck in the write operation, requiring that all the composition data for an entire assembly be collected onto a single process. This scheme creates memory overhead, but it should not be prohibitive. The upper bound for the expected amount of composition data for a single assembly is estimated as:

rings fuel per pin = 10
of IFBA rings per pin = 1
# of clad rings per pin = 1
# of crud mesh pin = 1
number of azimuthal divisions per ring = 8
number of isotopes per depletion zone = 300
number of fuel pin edit regions per assembly = 264*50

13*8*300*264*50*(4+8 bytes) = <u>4.6 GB</u>

However, since MPACT does not currently have an azimuthal depletion capability or CRUD layer modeling capability and the typical number of rings used in a fuel pin is 3, the *expected amount of data for an entire assembly is **227 MB***, which is far more manageable. A longer term task should be to revisit the I/O buffers to have them write blocks of data, so the upper bound case can be practical to execute. It is recommended that this activity probably need not take place until after a CRUD and azimuthal depletion capability are implemented into MPACT.

Since the file is to store each assembly's data in full symmetry, if the simulation models only half an assembly due to symmetry, the local data is replicated to account for the partial symmetry of the model.

3.2.2 Parallel Reading

In comparison to the parallel writing, the parallel reading is relatively trivial. Every process needs their data, and therefore every process sets up the index maps locally to determine what data they need and all processes read the file in parallel with no communication taking place amongst the various processes. Currently one limitation of the read is that the code expects the depletion mesh within each pin location to be identical to what has been defined with the input geometry and meshing parameters.

3.3 Symmetry

To facilitate usability of the isotopic restart file over a greater number of cases, it is structured in such a way as to make it appear identical for quarter symmetric models and full symmetry models. This supports features for symmetry unfolding (e.g. running in quarter symmetry then restarting or shuffling into a full symmetry model), and using shuffle patterns specified for the full core in all shuffle cases. This is accomplished without the unnecessary duplication of data.

Although every attempt is made to make this transparent to the user, there are subtle differences to the actual file structure. This section attempts to describe some of those differences in the file structure that are relevant to the developer, but not necessarily relevant to the typical user.

When a file is written from a case in full symmetry, all the data from the simulation model is exported to the file. Nothing special is done. All of the special treatments apply only to cases where the isotopic restart file is written from a quarter symmetric model. Thus the remainder of the discussion in this section focuses on the quarter symmetric case.

The basic approach to making the isotopic restart file appear as though it has a full core's worth of data is to correctly unfold the data based on the symmetry boundary conditions. There are two symmetry boundary conditions that are supported: mirror and rotational. The unfolding procedure is different for mirror and rotational symmetry. This is illustrated in Figure 9. Note that for 1/8th symmetric problems the mirror and rotational boundary conditions are equivalent.

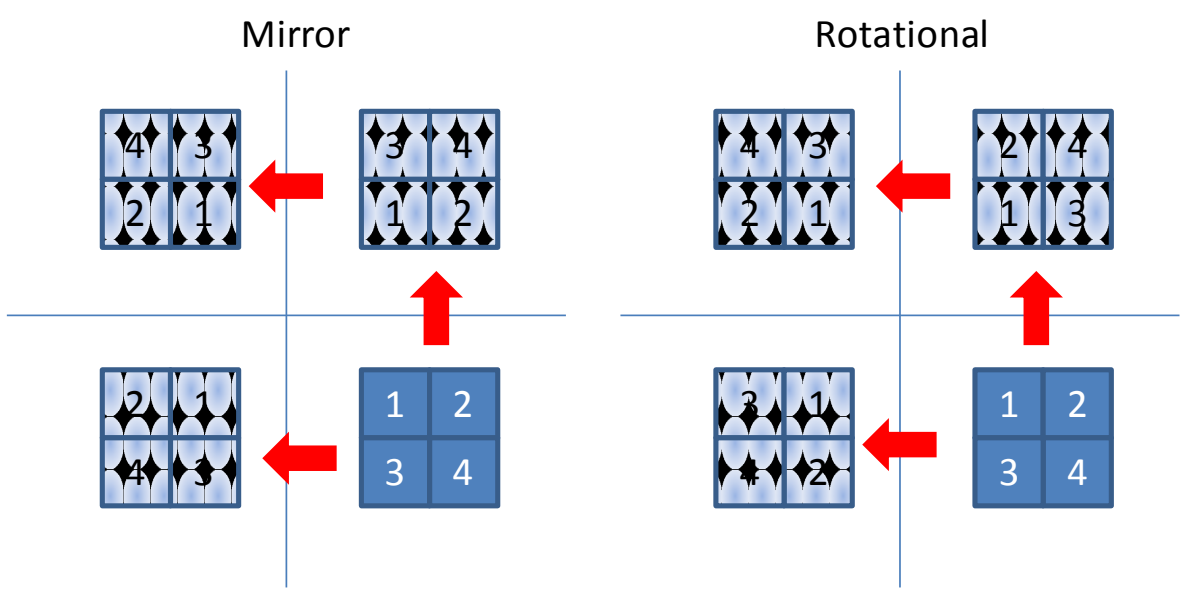


Figure 9: Illustration of Mirror and Rotational Symmetry unfolding

For assemblies that are bisected by a symmetry boundary, the data gets unfolded so that the a full assembly's worth of data is present in the file for reading. In the case of mirror symmetry, this requires only knowing the compositions of the assembly that is bisected by the symmetry boundary. However, if the boundary condition is rotational, then the rotational assembly position is used to unfold the bisected assembly. Conversely, if reading into a symmetric location only that part of the assembly's data in the file is read into the simulation. The rest is discarded. This is an approximation; an alternative approximation that may be added in the future is to average the all quadrants of the assembly's data into the quadrants that will be loaded into the simulation.

For all the fuel locations not in the quarter-symmetric simulation, the northeast, northwest, and southwest quadrants of the core, "virtual" entries are created for these assemblies in the HDF5 file using the "hard link" feature of HDF5. This prevents the duplication of data and thus allows for a file size ~4x smaller than the full core model.

On the subsequent read of the metadata entry, symmetry, for the assembly locations is used to inform MPACT what data manipulation to perform on any read operations to insure that the data is loaded in the correct orientation of the full core model. Values for symmetry follow the form <sym_type>-<axes>, therefore some expected values are `mirror-x`, `mirror-y`, `rotation-xy`, etc. For assemblies that are in the simulation the symmetry tag give in the file is "none". The symmetry values only have meaning to any read operation. Table 3 below summarizes the necessary transformations for the x-y pin coordinate in the lattice.

Table 3: Symmetry Translation Operations

Symmetry Type	x-coordinate transformation	y-coordinate transformation
none	$i = i_0$	$j = j_0$
mirror-x	$i = nx-i_0+1$	$j = j_0$
mirror-y	$i = i_0$	$j = ny-j_0+1$
mirror-xy	$i = nx-i_0+1$	$j = ny-j_0+1$
rotation-x (SE → NE=1 CW)	$i = ny-j_0+1$	$j = i_0$
rotation-y (SE → SW=3 CW)	$i = nx-i_0+1$	$j = ny-j_0+1$
rotation-xy (SE → NW=2 CW)	$i = j_0$	$j = nx-i_0+1$

Note that if the fuel pin has azimuthally dependent isotopics then the within pin distribution of isotopics must also be rotated or mirrored.

One final note with the symmetry is that if a physical rotation is applied to any of symmetry cases, then the coordinate transformation needed to perform the rotation should be applied after the symmetry operations. If the symmetry case is mirror, then the rotations go from clockwise (CW) to counter-clockwise (CCW).

4. INTEGRAL TESTING

4.0 Unit Testing

When writing new modules in MPACT, it is required that a unit test be constructed to test the newly created routines and objects to ensure they operate as expected for typical use case. The procedures and attributes listed in Figure 7 are tested, except for the `fclose`, `fopen`, `fdelete`, and `importCRLoadPat` routines which are not currently used. The functionality of each procedure tested how the attributes on the object are changed or the outputs produced by the object. The `initialize` and `clear` routines test and verify that each attribute on the object is set to the correct value after each routine is called. The `exportCore` routine tests the capability to write a restart file. The values of the file are then read back into the test and checked against reference values for each entry in the HDF5 file. The `importAssemLoadPat` routine tests the ability to read a restart file, read the file and shuffle the data to the correct assembly location, and read the file with shuffling and rotating the data. In this section, the `XSMesh` values for the test are checked against reference number densities for each region.

4.1 Verification Testing

In the case of reading and writing restart files, it was also beneficial to have a set of small problems to progressively test functionality and systematically debug issues that occur. The problems that were used started with a single axial level of a single assembly with a single pin and increased in complexity to a multi-level three-by-three assembly cross where each assembly was comprised of three-by-three pins. A table of the 16 cases used is shown below in Table 4. Of these 16 tests, case numbers 1, 14, and 16 were added to the automated test suite.

All 16 cases were run initially to generate the corresponding restart file. Each case was then restarted from the file, and the restart eigenvalue was compared to the case's initial eigenvalue to ensure consistency. For shuffling and shuffling with rotation capability, select multi-assembly cases from the suite were tested. The assemblies of different enrichments were shuffled on the periphery to symmetric locations which preserved the eigenvalue and allowed for pin power comparisons to ensure the shuffling and rotation were working as expected.

A total of 29 automated tests were added to protect this feature.

Table 4: Verification Tests

Case Number	Number of Assemblies		Number of Pins per Assembly		Number of Axial Levels		Symmetry	
	1x1	3x3	1x1	3x3	1 (2-D)	5 (3-D)	Full	Quarter
1	X		X		X		X	
2	X		X		X			X
3	X		X			X	X	
4	X		X			X		X
5	X			X	X		X	
6	X			X	X			X
7	X			X		X	X	
8	X			X		X		X
9		X	X		X		X	
10		X	X		X			X
11		X	X			X	X	
12		X	X			X		X
13		X		X	X		X	
14		X		X	X			X
15		X		X		X	X	
16		X		X		X		X

In addition to these tests which were run as a part of the development, numerous verification tests were added to verify the symmetry unfolding. The tests described in Table 5 were added to verify the symmetry capability. In addition to these tests there are tests for verifying the automated decay calculation.

Table 5: Symmetry Verification Unit Tests

Test Name	Feature Tested
MPACT_Drivers_testVerify_shuffle_mirror-x_rot0	Reading from NE quadrant with 0 physical rotation (mirror symmetry)
MPACT_Drivers_testVerify_shuffle_mirror-x_rot1	Reading from NE quadrant with 1 physical rotation (mirror symmetry)
MPACT_Drivers_testVerify_shuffle_mirror-x_rot2	Reading from NE quadrant with 2 physical rotation (mirror symmetry)
MPACT_Drivers_testVerify_shuffle_mirror-x_rot3	Reading from NE quadrant with 3 physical rotation (mirror symmetry)
MPACT_Drivers_testVerify_shuffle_mirror-y_rot0	Reading from SW quadrant with 0 physical rotation (mirror symmetry)
MPACT_Drivers_testVerify_shuffle_mirror-y_rot1	Reading from SW quadrant with 1 physical rotation (mirror symmetry)
MPACT_Drivers_testVerify_shuffle_mirror-y_rot2	Reading from SW quadrant with 2 physical rotation (mirror symmetry)
MPACT_Drivers_testVerify_shuffle_mirror-y_rot3	Reading from SW quadrant with 3 physical rotation (mirror symmetry)
MPACT_Drivers_testVerify_shuffle_mirror-xy_rot0	Reading from NW quadrant with 0 physical rotation (mirror symmetry)
MPACT_Drivers_testVerify_shuffle_mirror-xy_rot1	Reading from NW quadrant with 1 physical rotation (mirror symmetry)
MPACT_Drivers_testVerify_shuffle_mirror-xy_rot2	Reading from NW quadrant with 2 physical rotation (mirror symmetry)
MPACT_Drivers_testVerify_shuffle_mirror-xy_rot3	Reading from NW quadrant with 3 physical rotation (mirror symmetry)
MPACT_Drivers_testVerify_shuffle_rotation-x_rot0	Reading from NE quadrant with 0 physical rotation (rotational symmetry)
MPACT_Drivers_testVerify_shuffle_rotation-x_rot1	Reading from NE quadrant with 1 physical rotation (rotational symmetry)
MPACT_Drivers_testVerify_shuffle_rotation-x_rot2	Reading from NE quadrant with 2 physical rotation (rotational symmetry)
MPACT_Drivers_testVerify_shuffle_rotation-x_rot3	Reading from NE quadrant with 3 physical rotation (rotational symmetry)
MPACT_Drivers_testVerify_shuffle_rotation-y_rot0	Reading from SW quadrant with 0 physical rotation (rotational symmetry)
MPACT_Drivers_testVerify_shuffle_rotation-y_rot1	Reading from SW quadrant with 1 physical rotation (rotational symmetry)
MPACT_Drivers_testVerify_shuffle_rotation-y_rot2	Reading from SW quadrant with 2 physical rotation (rotational symmetry)
MPACT_Drivers_testVerify_shuffle_rotation-y_rot3	Reading from SW quadrant with 3 physical rotation (rotational symmetry)
MPACT_Drivers_testVerify_shuffle_rotation-xy_rot0	Reading from NW quadrant with 0 physical rotation (rotational symmetry)
MPACT_Drivers_testVerify_shuffle_rotation-xy_rot1	Reading from NW quadrant with 1 physical rotation (rotational symmetry)
MPACT_Drivers_testVerify_shuffle_rotation-xy_rot2	Reading from NW quadrant with 2 physical rotation (rotational symmetry)
MPACT_Drivers_testVerify_shuffle_rotation-xy_rot3	Reading from NW quadrant with 3 physical rotation (rotational symmetry)

5. SIMULATION OF VERA BENCHMARK PROBLEM 10

5.0 Model Description

The cases that were run to exercise the restart and shuffling functionality are the VERA benchmark Problems 9 and 10. It was necessary to rerun Problem 9 with the restart capability because Problem 10 needs to be restarted from the end of Problem 9 (EOC1). Problem 9 is the full core WBN1 Cycle 1 with neutronics coupled to thermal hydraulic feedback. A brief summary of the core description is:

- 193 Fuel assemblies, 56 full or partial assemblies modeled in quarter symmetry
- 17, 19, and 20 Assemblies with 3.1%, 2.6%, and 2.1% enrichments respectively in quarter symmetry
- 8 Banks of B4C Control Rods with AIC tips, for a total of 18 rods in quarter symmetry
- 6, 12, 4, 3, and 2 Banks of Pyrex burnable inserts of clusters of 24, 20, 16, 12, and 8 rodlets respectively

The layout of the core is shown below in Figure 10:

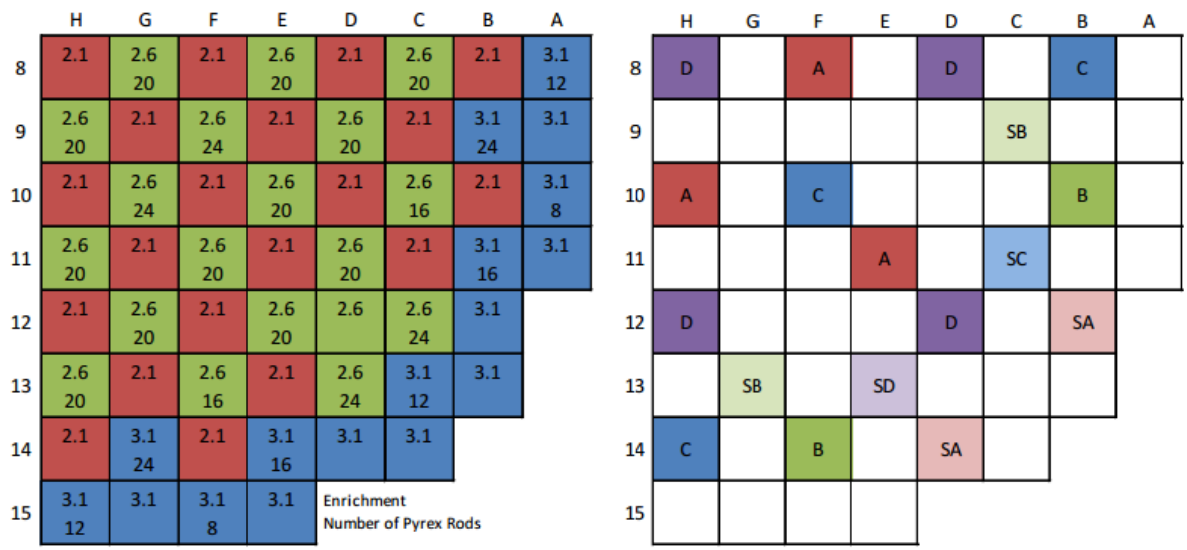


Figure 10: VERA Problem 5 Assembly, Poison, and Control Rod Layout in Quarter Symmetry[3]

A full description of the problem is presented in "VERA Core Physics Benchmark Progression Problem Specifications". The VERA benchmark Problem 10 differs in several ways from Problem 9. The chief differences taken from [4] are:

- All burnable Pyrex absorbers are removed
- Feed assemblies enriched to 3.709% are added
- Feed assemblies are loaded with various configurations of IFBA and/or WABA poisons
- Fuel is shuffled from end of cycle 1 in Problem 9.

The primary difference is that the fuel from cycle 1 is shuffled at the beginning of cycle 2 in Problem 10. Figure 11 below describes the fuel loading map for cycle 2. The purple assemblies describe the fresh feed assemblies being added, along with what IFBA and WABA loading are present. All other assemblies reference their previous x and y label location in cycle 1. The specific IFBA and WABA loading maps are described in further detail in [4].

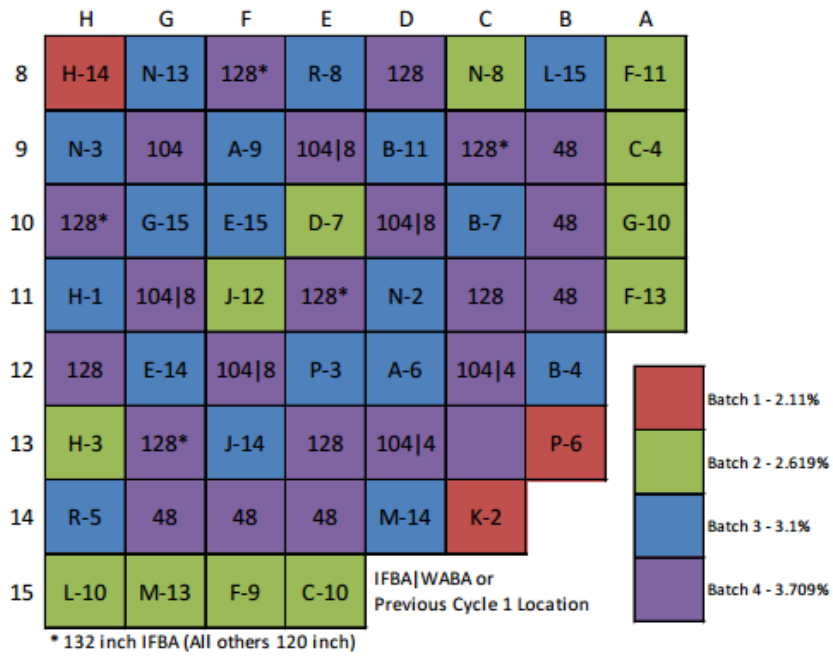


Figure 11: Problem 10 WBN1 Cycle 2 Core Loading Pattern[3]

For the purposes of demonstration, the cycle operation is approximated from the specification given in [4] and the approximated operation history used here is shown in Table 6. The approximations include constant power over the cycle except for ramp-up and coast-down and constant rod position.

The MOC discretization used the Chebyshev-Gauss quadrature with 16 azimuthal angles per octant and 4 polar angles per octant and a ray spacing of 0.05 cm. Transport corrected P0 scattering based on the neutron leakage conservation with the out-scatter approximation was used for the scattering treatment. The newest cross section library for MPACT, `mpact47g_70s_v4.0_11032014.fmt`, was used for the transport cross sections and the MPACT's internal depletion capability was used. The internal depletion capability was used because at the time the ORIGEN coupling required the 56-group structure which is known to have issues with the transport correction.

For the thermal-hydraulics in CTF the direct-moderator heating fraction was set to 2% and the heat transfer coefficient for the fuel-clad gap was set to be 10,000 W/m^2-K. Previously the value of 4,500 W/m^2-K has been recommended, however this value appears to be more appropriate for BOL before the gap has closed. The value of 10,000 W/m^2-K was obtained from a parametric study shown in Appendix B to achieve a target fuel temperature of 835 K in a pin cell calculation.

Table 6: Cycle 1 Simulated Operating History

State Index	Exposure (EFPD)	Power (%)	Inlet Temp (F)	Bank D Position
1	0	1.00E-06	557.0	219
2	9	65.7	557.6	192
3	32	99.7	558.1	219
4	45	97.7	557.9	217
5	60	97.7	557.9	217
6	80	97.7	557.9	217
7	100	97.7	557.9	217
8	120	97.7	557.9	217
9	160	97.7	557.9	217
10	200	97.7	557.9	217
11	240	97.7	557.9	217
12	280	97.7	557.9	217
13	320	97.7	557.9	217
14	360	97.7	557.9	217
15	398.6	97.7	557.9	217
16	410.7	89.9	557.1	224
17	423.6	78.8	556.3	228
18	441	64.5	554.9	230

The simulation was performed on the Falcon compute cluster at INL's HPC Center and the problem was decomposed in MPACT with 4234 spatial domains (58 axial and 73 radial). CTF used 56 of these processors for assembly decomposition. The radial decomposition for 41, 56, and 73 radial domains in MPACT is illustrated in Figure 12. It was determined that the 41 radial domains is more suitable than the 56 domains that have been used in several previous calculations because the maximum subdomain size in either case is the same (1.5 assembly by 1.5 assembly). Therefore, using 56 processors only increases the amount of load imbalance and the computation time is still limited by the largest subdomain. The 41 radial partitions provides better load balance and therefore better parallel efficiency and computational resource utilization. When the resources are available though, using a radial decomposition of 73 domains still demonstrates good load balance and efficiency.

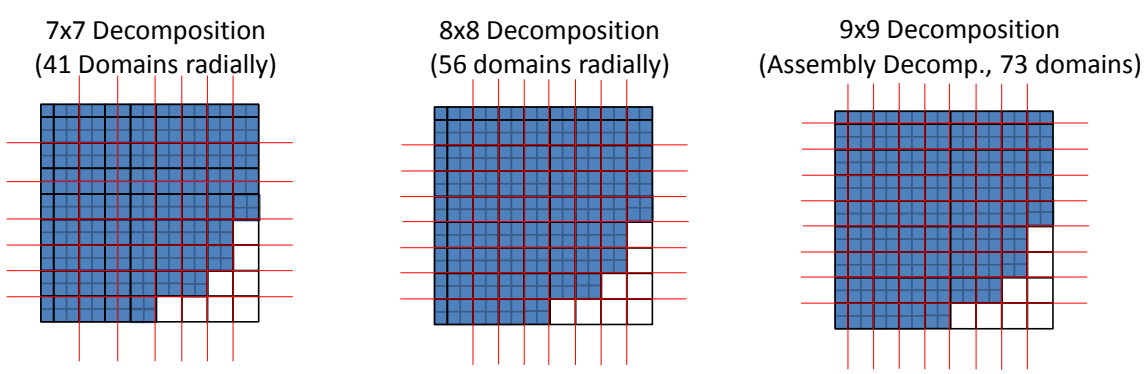


Figure 12: Radial partitioning in MPACT

Cycle 1 was modeled with 18 states. Due to wall-time limitations the simulation was run in 3 steps which required 2 restarts. In shuffling the fuel from cycle 1 to cycle 2 the exact post-irradiation cool down time was not known, so a 1 month decay was assumed between EOC1 state and BOC2 state. However, since there is currently no reference solution or measurement data for cycle 2, this approximation was deemed acceptable for demonstration purposes. The VERA inputs used for the calculation is provided in Appendix A.

5.1 Results and Discussion

5.1.0 Cycle 1 Results

The simulation and comparison of cycle 1 was the focus of the VERA problem 9 benchmark and therefore was not the principal focus of the work in this report. However, some results and comparison to measurement are included because a new cross section library was used, as well as a different heat transfer coefficient for the fuel-clad gap value in the 1-D heat conduction of the fuel.

The results are provided in Figure 12 and in Table 5 below. Somewhat better agreement with measurement was achieved than previous results, although there are still notable differences. From the results in Table 7 the only direct points of comparison are the HZP state and the 32 EFPD exposure case. The HZP and 32 EFPD cases have reasonably good comparison to measurement with a difference of ~16 ppmB in reactivity. Most of the rest of the simulated points appear to differ within 50 ppmB of measurement as noted by the black dashed lines in Figure 13.

The measured critical boron values could be adjusted using measured reactivity coefficients to estimate the criticality at the simulated points, however this process was not performed here because the computed fuel temperatures are not validated and are known to be a significant source of the discrepancy with measurement.

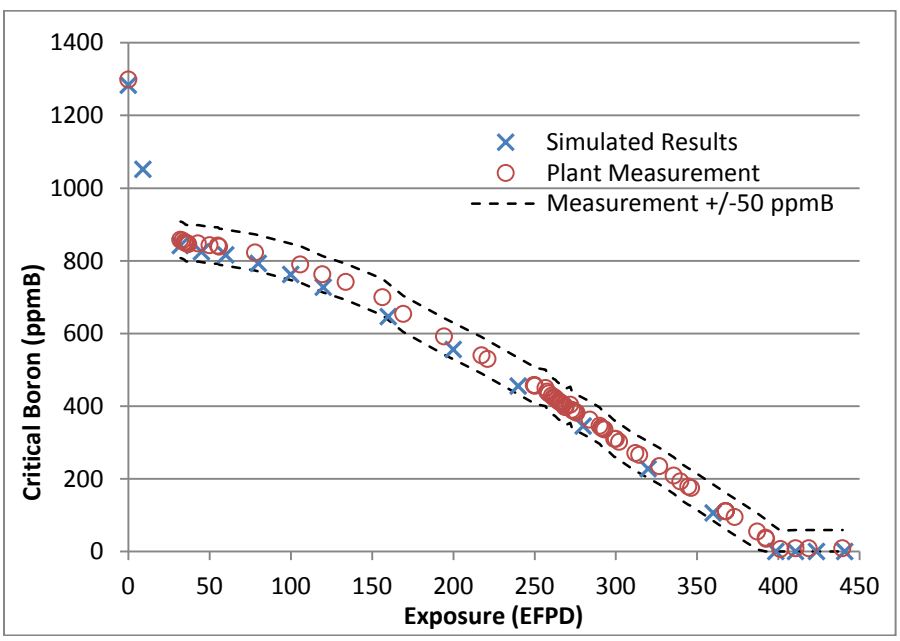


Figure 13: Comparison of Cycle 1 Critical Boron

Table 7: Measured and Simulated Cycle 1 States

Simulated						Measured			
Exposure (EFPD)	Power (%)	Bank D	Boron (ppmB)	keff	Fuel Temp (K)	Exposure (EFPD)	Power (%)	Bank D	Boron (ppmB)
0	1.0E-06	186	1283.24	1.00000	291.7	**0**	0		1299
9	65.7	192	1051.82	1.00000	756.9	---	---	---	---
32	99.7	217	841.86	1.00000	873.4	**32**	99.7	219	858
45	97.7	217	824.89	1.00000	866.0	**42.8**	100	215	848
60	97.7	217	816.23	1.00000	865.8	**55.9**	99.9	214	839
80	97.7	217	792.92	0.99999	865.3	**78**	99.9	208	823
100	97.7	217	762.36	1.00000	864.8	**105.8**	99.8	217	790
120	97.7	217	727.31	1.00000	864.4	**119.4**	99.8	212	763
160	97.7	217	646.34	1.00000	863.6	**156.4**	99.9	218	700
200	97.7	217	555.92	0.99999	862.8	**194.3**	98.9	215	592
240	97.7	217	455.29	1.00000	862.1	**249.6**	99.9	216	458
280	97.7	217	345.27	1.00000	861.4	**284**	99.9	218	363
320	97.7	217	227.61	1.00000	860.9	**314.5**	99.5	214	266
360	97.7	217	106.04	1.00000	860.5	**367.7**	100	216	111
398.6	97.7	217	1.0E-07	0.99853	860.1	**401.4**	99.6	217	7
410.7	89.9	224	1.0E-07	0.99790	833.3	**410.7**	89.9	216	9
423.6	78.8	228	1.0E-07	0.99765	796.0	**418.8**	83.4	228	9
441	64.5	230	1.0E-07	0.99705	750.0	**439.5**	65.3	227	9

The total run time by computational component for cycle 1 is given in Table 8. Figure 14 shows the relative time spent in each computational component. Additionally, when performing a restart, the solution for the restarted state must be recomputed and cycle 1 was completed with 2 restarts. The time spent re-computing the restarted states was 03:21:48 and the fraction of the total time spent re-computing restarts was 11.3%.

Table 8: Calculation Component Total Run Times for Cycle 1 (Including Restarts)

Component		Total Time (hh:mm:ss)	Average Time per State (hh:mm:ss)
MPACT	MOC+CMFD	08:06:51	00:25:37
	Subgroup	3:09:58	00:10:00
	Depletion	00:13:16	00:00:42
	Restart File I/O	01:02:27	00:03:17
	Other	00:09:23	00:00:30
COBRA-TF	Solve	17:06:54	00:54:03
Total		29:48:49	01:34:09

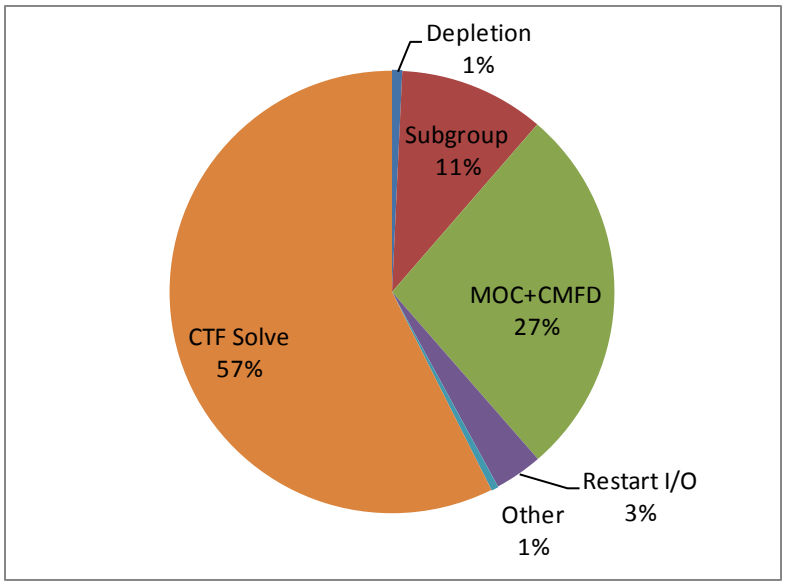


Figure 14: Relative Component Run Times of VERA-CS with 4234 cores on Falcon

5.1.1 Cycle 2 BOC results

The cycle 2 BOC condition was computed for HZP and HFP. The power distributions for the HZP state is shown in Figure 15. Figure 16 shows the relative fission rate, fuel temperature and coolant density distributions for HFP conditions. The critical boron for HZP and HFP was computed to be 1427.43 ppmB and 1251.22 ppmB, respectively. Of particular note in these results compared to cycle 1 is that at HZP the peaking factor is very high (3.56). However, the HFP peaking factor of 1.74 is much lower and appears to be within typical design goals for F_q. Additionally, the power peaking seems to occur near the top or bottom of the fuel stack rather than in the middle. This effect appears to be due in part to the use of the enriched blankets and the IFBA in several batch 4 assemblies that does not extend the full length of the fuel stack. As expected, all peaking occurs in the fresh assemblies. The power suppression due to the WABA is also visible in the HFP fission rate and fuel temperature distributions.

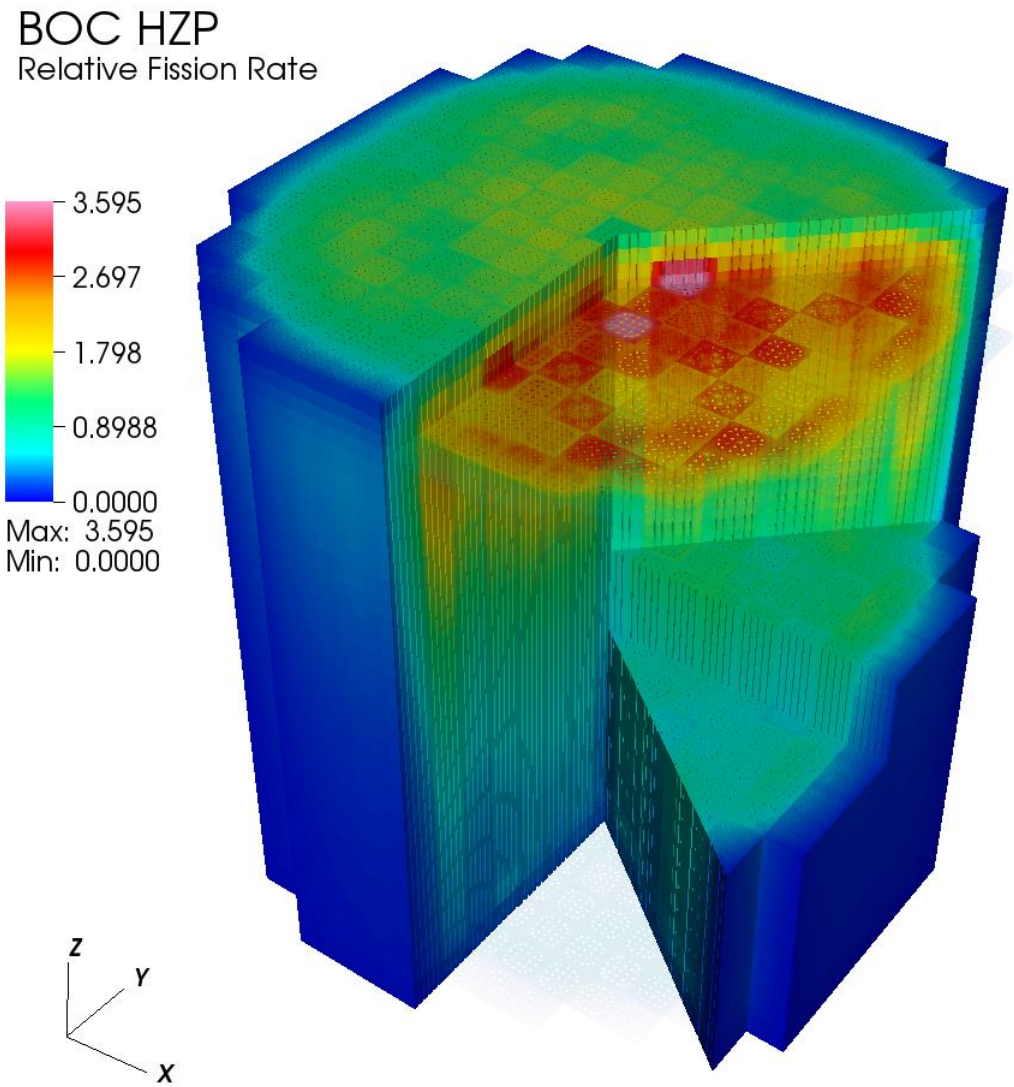
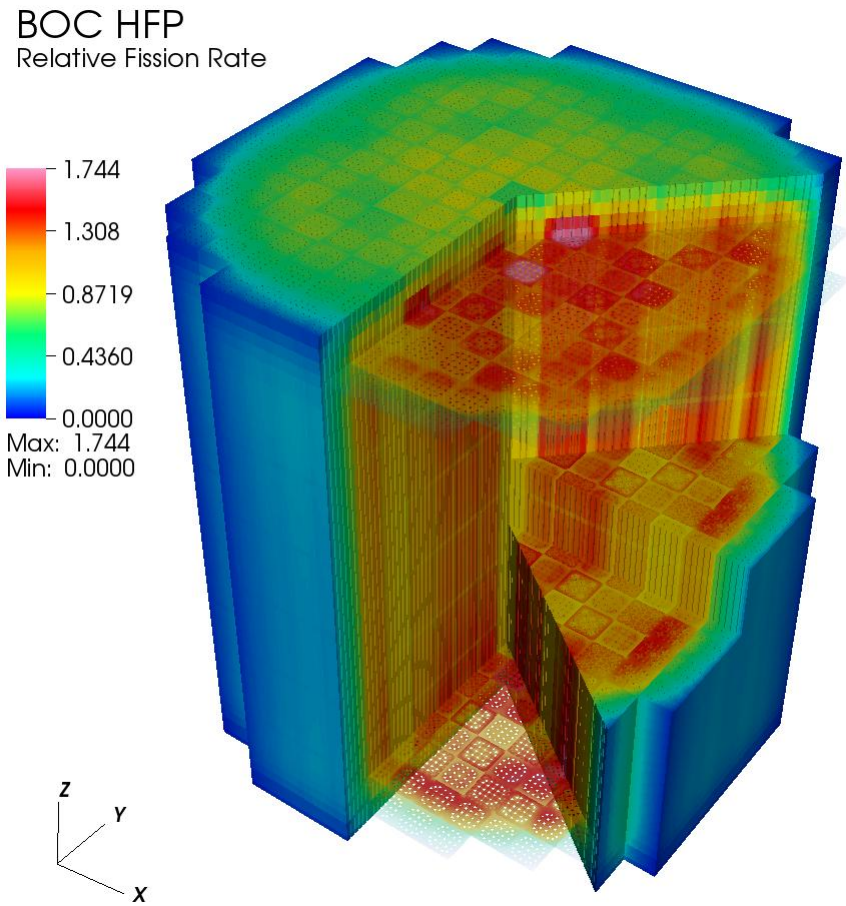
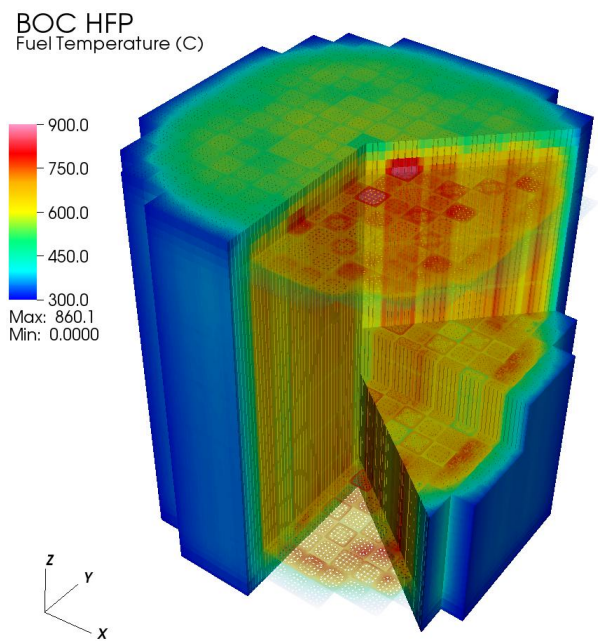


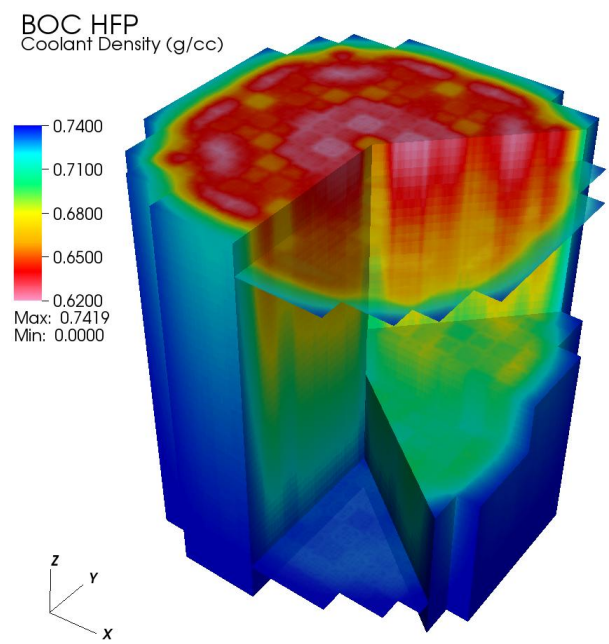
Figure 15: Cycle 2 BOC Relative Fission Rate Distribution at Hot Zero Power



(a) Relative Fission Rate



(b) Fuel Temperature



(c) Coolant Density

Figure 16: Cycle 2 BOC at Hot Full Power

5.1.2 Performance of Isotopic Restart File I/O

As noted previously, the isotopic restart file uses the HDF5 format. The I/O operations were timed for an earlier full core simulation performed on the EOS cluster as a part of the problem 10 work. This data included timings for write, restart read, and shuffle read. This operation occurred in parallel as using 2378 spatial partitions. Additionally, the times for the read and write operation were measured with and without using compression on the isotopic restart file. The compression used was provided by ZLIB [4] with a compression level of 5. The compression levels offered are 0 through 9, with 0 being less compression but less overhead, and 9 being the most overhead and compression. The compression level of 5 was chosen as a good balance of compression and minimal overhead. In testing of the compression algorithms used in ZLIB it appears that the improved compression is generally not worth the overhead when using compression levels higher than 5. Table 9 below summarizes the I/O performance.

Table 9: I/O Performance of Isotopic Restart File for a single state with 2378 processors on EOS

Operation	No Compression	Compression	Difference
Write	90 s	220 s	2.4x
Restart Read	< 1s	< 1s	---
Shuffle Read	---	26 s	---
File Size	4.5 GB	2.2 GB	59%

From the data in this table it can be concluded that the compression has little impact on the restart read operation, and that the time for this operation is trivial. The compression adds a notable increase to the write time increasing it by a factor of 2.4x, but given that it takes approximately 2 hrs to solve a steady-state problem, a total time of 220 s is negligible compared to the steady state-solve time. The reduction in file size from compression is nearly 60%, which certainly justifies the increased write times. The shuffle read time is notably higher than the restart read time, but its magnitude is still quite small.

The read and write operations were also timed for the simulation run on Falcon using 4234 processors, as described in Section 3.2, and compression. This data is given in Table 8 below.

Table 8: I/O Performance of Isotopic Restart File for a single state with 4234 processors on Falcon

Operation	Compression
Write	200 s
Restart Read	170 s
Shuffle Read	12 s
File Size	2.4 GB

In summary, it is apparent that the I/O operations for the isotopic restart file should not become a bottleneck and that the algorithm for the operations is reasonably scalable. A single state-point worth of composition data is manageable at < 2.5 GB, but including a cycle's worth of composition data might become prohibitive as this file size would be around 40 GB. Since the overhead for reading and writing is not significant but the reduction in file size is significant, the use of compression is recommended as the default setting.

6. CONCLUTIONS AND FUTURE WORK

6.0 Conclusions

A new isotopic restart and shuffle capability has been implemented in MPACT. This addition includes new input for VERAIn and parallel I/O algorithms for MPACT to generate, read, and shuffle the isotopic restart file. Using this capability, the VERA benchmark progression problem 10 was performed which involved simulating cycle 1 and generating the isotopic restart file for end of cycle 1 conditions. This data was subsequently shuffled and combined with new feed assemblies to build a cycle 2 core loading. The cycle 2 BOC state was then simulated for HFP and HZP conditions. Since a reference solution has not been made available for this problem, only the qualitative performance of the core was assessed (e.g. the relation of the power distribution to the known location of the WABA, IFBA, fresh and burned assemblies, etc.). Based on the successful completion of the sequence of test problems, the objectives of this milestone have been completed.

6.1 Future Work

Several tasks were identified for future work in order to continue to support and enhance the isotopic restart and shuffling capability in MPACT. These include:

1. (low priority) Enable tracking composition data for control rods
2. (low priority) Enable shuffling of inserts
3. (low priority) Enable shuffling of control rods
4. (medium priority) Replace axial heavy metal mass with pin-wise exposure data
5. (low priority) Remove regions above and below assembly nozzles from `nz` dimension. Corresponding data spaces with an `nz` dimension currently evaluate zeros for these regions.
6. (low priority) Modify I/O buffers to write chunks of data instead of data for entire assembly at once to minimize memory overhead
7. (medium priority) Add pin information for number of rings and radii to allow for homogenization/mapping of data during read.
8. (medium priority) Implement homogenization and mapping of data when reading data from file.
9. (low priority) Shuffling by label
10. (medium priority) Perform additional input processing based on restart file data to reduce input redundancy. (e.g. store reference to core geometry, store all state variables).

In terms of expanding the capability for tracking and shuffling compositions of inserts and control rods, it is suggested that the approach used for assemblies simply be replicated for these structures. However, in the case of control rods there will be new challenges since these structures traverse the axial mesh and thus a technique to resolve this behavior will need to be developed. A likely approach would be to track a separate axial mesh of the control rod and homogenize/de-homogenize the compositions as the rod moves. In order to perform a more complete analysis of problem 10, additional information would be necessary such as the public specifications of the Tritium Producing Burnable Absorber Rods (TPBARs) and the actual shutdown time between the end of cycle 1 and the beginning of cycle 2 which would enable the accurate prediction of the isotope decay times. The cycle 2 zero power physics testing should also be documented (e.g. rod bank positions, boron concentrations, flux maps) to further confirm the cycle 2 core loading. It would also be worthwhile to enable the isotopic restart files produced by MPACT to be read into another code (e.g. SHIFT) to increase the versatility and further verify the shuffle/restart capability implemented in MPACT.

REFERENCES

[1] A. Godfrey and S. Palmtag, "Initial Specification for Core Shuffling," 22 August 2014.

[2] B. Kochunas, "A Hybrid Parallel Algorithm for the 3-D Method of Characteristics Solution of the Boltzmann Transport Equation on High Performance Compute Clusters", *PhD Dissertation*, University of Michigan, http://deepblue.lib.umich.edu/handle/2027.42/100072 (2013)

[3] A. Godfrey, "VERA Core Physics Benchmark Progression Problem Specifications" 29 August 2014 CASL-U-2012-0131-004

[4] P. Deutsch and J.-L. Gailly, " ZLIB Compressed Data Format Specification version 3.3", RFC Editor, 1996.

[5] The HDF Group. Hierarchical Data Format, version 5, 1997-2014. http://www.hdfgroup.org/HDF5/

APPENDIX A - VERA INPUT

p10_cycle1.inp

```
[CASEID]
  title 'Operational Reactor - Watts Bar Unit 1 Cycle 1 - Public'

[STATE]
  op_date "1996/01/18"
  power 1e-6                            ! %
  pressure 2250
  tinlet 557.0 F                         ! F
  tfuel 565 K                            ! K
  modden 0.743                           ! g/cc
  boron 1285                             ! ppm
  sym qtr
  search "boron"
  feedback on
  rodbank SA 230
          SB 230
          SC 230
          SD 230
           A 230
           B 230
           C 230
           D 219

[STATE]
  power         65.7
!  tinlet       557.6 F
!  rodbank    D 192
  deplete EFPD   9.0
  restart_write p10_restart_STATE_02.h5 09EFPD

[STATE]
  power         99.7
!  tinlet       558.1 F
!  rodbank    D 219
  deplete EFPD  32.0
  restart_write p10_restart_STATE_03.h5 32EFPD

[STATE]
  power         97.7
!  tinlet       557.9 F
!  rodbank    D 217
  deplete EFPD 45.0
  restart_write p10_restart_STATE_04.h5 45EFPD

[STATE]
  deplete EFPD 60.0
  restart_write p10_restart_STATE_05.h5 60EFPD

[STATE]
  deplete EFPD 80.0
  restart_write p10_restart_STATE_06.h5 80EFPD

[STATE]
  deplete EFPD 100.0
  restart_write p10_restart_STATE_07.h5 100EFPD
```

```
[STATE]
  deplete EFPD 120.0
  restart_write p10_restart_STATE_08.h5 120EFPD

[STATE]
  deplete EFPD 160.0
  restart_write p10_restart_STATE_09.h5 160EFPD

[STATE]
  deplete EFPD 200.0
  restart_write p10_restart_STATE_10.h5 200EFPD

[STATE]
  deplete EFPD 240.0
  restart_write p10_restart_STATE_11.h5 240EFPD

[STATE]
  deplete EFPD 280.0
  restart_write p10_restart_STATE_12.h5 280EFPD

[STATE]
  deplete EFPD 320.0
  restart_write p10_restart_STATE_13.h5 320EFPD

[STATE]
  deplete EFPD 360.0
  restart_write p10_restart_STATE_14.h5 360EFPD

[STATE]
  deplete EFPD 398.6
  restart_write p10_restart_STATE_15.h5 398.6EFPD

[STATE]
  power        89.9
!  tinlet       557.1 F
!  rodbank    D 224
  deplete EFPD 410.7
  restart_write p10_restart_STATE_16.h5 410.7EFPD

[STATE]
  power        78.8
!  tinlet       556.3 F
!  rodbank    D 228
  deplete EFPD 423.6
  restart_write p10_restart_STATE_15.h5 423.6EFPD

[STATE]
  power        64.5
  tinlet       554.9 F
!  rodbank    D 230
!  deplete EFPD 441.0
  restart_write p10_fuel_restart.h5 EOC1
```

```
[CORE]
  unit   1
  cycle  1
  size  15                ! assemblies across core
  rated 3411 131.68       ! MW, Mlbs/hr
  apitch 21.5
  height 406.337

  core_shape
    0 0 0 0 1 1 1 1 1 1 1 0 0 0 0
    0 0 1 1 1 1 1 1 1 1 1 1 1 0 0
    0 1 1 1 1 1 1 1 1 1 1 1 1 1 0
    0 1 1 1 1 1 1 1 1 1 1 1 1 1 0
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
    0 1 1 1 1 1 1 1 1 1 1 1 1 1 0
    0 1 1 1 1 1 1 1 1 1 1 1 1 1 0
    0 0 1 1 1 1 1 1 1 1 1 1 1 0 0
    0 0 0 0 1 1 1 1 1 1 1 0 0 0 0

  assm_map
   B1000
   B2000  B1000
   B1000  B2000  B1000
   B2000  B1000  B2000  B1000
   B1000  B2000  B1000  B2000  B2000
   B2000  B1000  B2000  B1000  B2000  B3000
   B1000  B3000  B1000  B3000  B3000  B3000
   B3000  B3000  B3000  B3000

  insert_map
     -
    20  -
     - 24  -
    20  - 20  -
     - 20  - 20  -
    20  - 16  - 24 12
     - 24  - 16  -  -
    12  -  8  -

  crd_map
    1
    - -
    1 - 1
    - - - 1
    1 - - - 1
    - 1 - 1 - -
    1 - 1 - 1 -
    - - - -
```

```
  crd_bank
    D  -  A  -  D  -  C  -
    -  -  -  -  - SB  -  -
    A  -  C  -  -  -  B  -
    -  -  -  A  - SC  -  -
    D  -  -  -  D  - SA
    - SB  - SD  -  -  -
    C  -  B  - SA  -
    -  -  -  -

! the detector map is upside down for denovo
  det_map
            1 - - 1 - - -
        1 - - - 1 - - 1 - 1 -
      - 1 - 1 - - 1 - - - - - 1
      - - - - 1 - - 1 - - 1 - -
    1 - - - 1 - - 1 - - 1 - - - 1
    - - - - 1 - 1 - - - - 1 - - -
    - 1 - - - - - - 1 - 1 - - - 1
    1 - 1 - 1 - 1 - - 1 - 1 1 1 -
    - - - 1 - - 1 - - 1 - - 1 - -
    1 - 1 - - 1 - 1 - - - - - 1 -
    - - - - 1 - - - 1 - 1 - 1 - -
      1 1 - - - - 1 - - - - - -
      - - - - - - 1 - 1 - 1 - 1
        1 - - 1 - 1 - - - - -
            - - 1 - - 1 -

  baffle ss 0.19 2.85
! vessel cs 219.71 21.99 - not functional yet

  lower_plate ss  5.0 0.5   ! mat, thickness, vol frac
  upper_plate ss  7.6 0.5   ! mat, thickness, vol frac

! lower_ref  mod 20.0 1.0   ! reduce this as much as possible for now
! upper_ref  mod 20.0 1.0

  xlabel   R  P  N  M  L  K  J  H  G  F  E  D  C  B  A
  ylabel  01 02 03 04 05 06 07 08 09 10 11 12 13 14 15

  mat he     0.0001786
  mat inc    8.19
  mat ss     8.0
  mat zirc   6.56 zirc4
  mat spring 0.4 ss 0.05 he 0.95

[ASSEMBLY]
  include WBN1_C1+2_ASY.inp

[INSERT]
  title "Pyrex"
  npin 17

  mat pyrx1 2.25 pyrex-vera

  cell 1  0.214 0.231 0.241 0.427 0.437 0.484 / he ss he pyrx1 he ss
  cell 2                          0.437 0.484 /            spring ss    ! plenum
  cell 3                                0.484 /                   ss    ! plug
```

```
rodmap  PY8
   -
   - -
   - - -
   1 - - -
   - - - - -
   - - - - - 1
   - - - - - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PLEN8
   -
   - -
   - - -
   2 - - -
   - - - - -
   - - - - - 2
   - - - - - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PLUG8
   -
   - -
   - - -
   3 - - -
   - - - - -
   - - - - - 3
   - - - - - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PY12
   -
   - -
   - - -
   1 - - -
   - - - - -
   - - - - - -
   - - - 1 - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PLEN12
   -
   - -
   - - -
   2 - - -
   - - - - -
   - - - - - -
   - - - 2 - - -
   - - - - - - - -
   - - - - - - - - -
```

```
rodmap  PLUG12
   -
   - -
   - - -
   3 - - -
   - - - - -
   - - - - - -
   - - - 3 - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PY16
   -
   - -
   - - -
   1 - - -
   - - - - -
   - - - - - 1
   - - - 1 - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PLEN16
   -
   - -
   - - -
   2 - - -
   - - - - -
   - - - - - 2
   - - - 2 - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PLUG16
   -
   - -
   - - -
   3 - - -
   - - - - -
   - - - - - 3
   - - - 3 - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PY20
   -
   - -
   - - -
   1 - - -
   - - - - -
   - - - - - 1
   1 - - 1 - - -
   - - - - - - - -
   - - - - - - - - -
```

```
rodmap  PLEN20
   -
   - -
   - - -
   2 - - -
   - - - - -
   - - - - - 2
   2 - - 2 - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PLUG20
   -
   - -
   - - -
   3 - - -
   - - - - -
   - - - - - 3
   3 - - 3 - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PY24
   -
   - -
   - - -
   1 - - 1
   - - - - -
   - - - - - 1
   1 - - 1 - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PLEN24
   -
   - -
   - - -
   2 - - 2
   - - - - -
   - - - - - 2
   2 - - 2 - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PLUG24
   -
   - -
   - - -
   3 - - 3
   - - - - -
   - - - - - 3
   3 - - 3 - - -
   - - - - - - - -
   - - - - - - - - -

axial   8  13.221 PLUG8  15.761 PY8  376.441 PLEN8  398.64
axial  12  13.221 PLUG12 15.761 PY12 376.441 PLEN12 398.64
axial  16  13.221 PLUG16 15.761 PY16 376.441 PLEN16 398.64
axial  20  13.221 PLUG20 15.761 PY20 376.441 PLEN20 398.64
axial  24  13.221 PLUG24 15.761 PY24 376.441 PLEN24 398.64
```

```
[CONTROL]
  title "B4C with AIC tips"
  npin 17
  stroke  365.125 230     ! approx for 1.5875 step sizes and 230 max stroke

  mat aic 10.2
  mat b4c 1.76

  cell 1  0.382 0.386 0.484 / aic he ss
  cell 2  0.373 0.386 0.484 / b4c he ss
  cell 3        0.386 0.484 / spring ss !plenum
  cell 4              0.484 /        ss !plug

  rodmap AIC
     -
     - -
     - - -
     1 - - 1
     - - - - -
     - - - - - 1
     1 - - 1 - - -
     - - - - - - - -
     - - - - - - - - -

  rodmap B4C
     -
     - -
     - - -
     2 - - 2
     - - - - -
     - - - - - 2
     2 - - 2 - - -
     - - - - - - - -
     - - - - - - - - -

  rodmap PLEN
     -
     - -
     - - -
     3 - - 3
     - - - - -
     - - - - - 3
     3 - - 3 - - -
     - - - - - - - -
     - - - - - - - - -

  rodmap PLUG
     -
     - -
     - - -
     4 - - 4
     - - - - -
     - - - - - 4
     4 - - 4 - - -
     - - - - - - - -
     - - - - - - - - -
```

```
  axial 1     15.131
        PLUG  17.031
         AIC 118.631
         B4C 377.711
        PLEN 388.411

[DETECTOR]
  title "Incore instrument thimble"
  npin 17

  mat he 0.0001786
  mat ss 8.0

  cell 1  0.258 0.382 / he ss

  rodmap  LAT
     1
     - -
     - - -
     - - - -
     - - - - -
     - - - - - -
     - - - - - - -
     - - - - - - - -
     - - - - - - - - -

  axial 1  0.0 LAT 406.337

[EDITS]
  include axial_mesh_1.inp

[MPACT]
  vis_edits         none
  jagged            true
  coupling_method   ctf
  ith_hgap          10000
  ith_dhfrac        0.02
!quad_set
  quad_type         CHEBYSHEV-GAUSS
  polars_octant     4
  azimuthals_octant 16
  ray_spacing       0.05
!iteration_control
  flux_tolerance    1e-4
  num_inners        2
  k_tolerance       1e-5
  up_scatter        1
  num_outers        200
  scattering        TCP0
!cmfd
  cmfd              cmfd
  cmfd_solver       mgnode
  k_shift           1.15
  cmfd_num_outers   20
!2D1D
  split_TL          true
  TL_treatment      lflat
  nodal_method      nem
!parallel
!  num_space         14906
```

```
  par_method        EXPLICITFILE
  par_file          partition_41r.txt
  num_angle         1
  num_energy        1
  num_threads       2
!xs_library
  xs_filename       mpact47g_70s_v4.0.fmt
  xs_type           ORNL
  dep_filename      MPACT.dpl
  dep_edit          false
  dep_substep       2
  subgroup_set      4
!mesh

  mesh fuel  3 1 1 / 8 8 8 8 8 8
  mesh gtube 3 1   / 8 8 8 8 8

  meshing_method    nonfuel
  automesh_bounds  2.0 10.0

[COBRATF]
  nfuel   8             ! number of fuel rings in conduction model
  nc      1             ! conduction option - radial conduction
  irfc    2             ! friction factor correlation default=2
  dhfrac  0.02          ! fraction of power deposited directly into coolant
  hgap    10000.0       ! gap conductance
  epso    0.001
  oitmax  5
  iitmax  40
  gridloss END 0.9070   ! spacer grid loss coefficient
  gridloss MID 0.9065   ! spacer grid loss coefficient
  dtmin   0.000001
  dtmax   0.1
  tend    0.1
  rtwfp   1000.0
  maxits  100000
  courant 0.8
  parallel 1

[COUPLING]
  epsk        1.0  ! pcm
  epsp        1.0e-4
  rlx_power   0.65
  rlx_tfuel   1.0
  rlx_den     1.0
  maxiter     1000
```

p10_cycle2.inp

```
[CASEID]
  title 'Operational Reactor - Watts Bar Unit 1 Cycle 1 - Public'

[STATE]
  op_date "1996/01/18"
  power 1e-6                            ! %
  pressure 2250
  tinlet 557.0 F                         ! F
  tfuel 565 K                            ! K
  modden 0.743                           ! g/cc
  boron 1285                             ! ppm
  sym qtr
  search "boron"
  feedback on
  rodbank SA 230
          SB 230
          SC 230
          SD 230
           A 230
           B 230
           C 230
           D 219

[STATE]
  power         65.7
  tinlet       557.6 F
  rodbank    D 192
  deplete EFPD   9.0
  restart_write p10_restart_STATE_02.h5 09EFPD

[STATE]
  power         99.7
  tinlet       558.1 F
  rodbank    D 219
  deplete EFPD  32.0
  restart_write p10_restart_STATE_03.h5 32EFPD

[STATE]
  power         97.7
  tinlet       557.9 F
  rodbank    D 217
  deplete EFPD 45.0
  restart_write p10_restart_STATE_04.h5 45EFPD

[STATE]
  deplete EFPD 60.0
  restart_write p10_restart_STATE_05.h5 60EFPD

[STATE]
  deplete EFPD 80.0
  restart_write p10_restart_STATE_06.h5 80EFPD

[STATE]
  deplete EFPD 100.0
  restart_write p10_restart_STATE_07.h5 100EFPD

[STATE]
  deplete EFPD 120.0
  restart_write p10_restart_STATE_08.h5 120EFPD
```

```
[STATE]
  deplete EFPD 160.0
  restart_write p10_restart_STATE_09.h5 160EFPD

[STATE]
  deplete EFPD 200.0
  restart_write p10_restart_STATE_10.h5 200EFPD

[STATE]
  deplete EFPD 240.0
  restart_write p10_restart_STATE_11.h5 240EFPD

[STATE]
  deplete EFPD 280.0
  restart_write p10_restart_STATE_12.h5 280EFPD

[STATE]
  deplete EFPD 320.0
  restart_write p10_restart_STATE_13.h5 320EFPD

[STATE]
  deplete EFPD 360.0
  restart_write p10_restart_STATE_14.h5 360EFPD

[STATE]
  deplete EFPD 398.6
  restart_write p10_restart_STATE_15.h5 398.6EFPD

[STATE]
  power        89.9
  tinlet       557.1 F
  rodbank    D 224
  deplete EFPD 410.7
  restart_write p10_restart_STATE_16.h5 410.7EFPD

[STATE]
  power        78.8
  tinlet       556.3 F
  rodbank    D 228
  deplete EFPD 423.6
  restart_write p10_restart_STATE_15.h5 423.6EFPD

[STATE]
  op_date "09/05/1997"
  power        64.5
  tinlet       554.9 F
  rodbank    D 230
  deplete EFPD 441.0
  restart_write p10_fuel_restart.h5 EOC1

[CORE]
  unit   1
  cycle  1
  size  15               ! assemblies across core
  rated 3411 131.68      ! MW, Mlbs/hr
  apitch 21.5
  height 406.337
```

```
core_shape
  0 0 0 0 1 1 1 1 1 1 1 0 0 0 0
  0 0 1 1 1 1 1 1 1 1 1 1 1 0 0
  0 1 1 1 1 1 1 1 1 1 1 1 1 1 0
  0 1 1 1 1 1 1 1 1 1 1 1 1 1 0
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  0 1 1 1 1 1 1 1 1 1 1 1 1 1 0
  0 1 1 1 1 1 1 1 1 1 1 1 1 1 0
  0 0 1 1 1 1 1 1 1 1 1 1 1 0 0
  0 0 0 0 1 1 1 1 1 1 1 0 0 0 0

assm_map
 B1000
 B2000  B1000
 B1000  B2000  B1000
 B2000  B1000  B2000  B1000
 B1000  B2000  B1000  B2000  B2000
 B2000  B1000  B2000  B1000  B2000  B3000
 B1000  B3000  B1000  B3000  B3000  B3000
 B3000  B3000  B3000  B3000

insert_map
   -
  20  -
   - 24  -
  20  - 20  -
   - 20  - 20  -
  20  - 16  - 24 12
   - 24  - 16  -  -
  12  -  8  -

crd_map
  1
  - -
  1 - 1
  - - - 1
  1 - - - 1
  - 1 - 1 - -
  1 - 1 - 1 -
  - - - -

crd_bank
  D  -  A  -  D  -  C  -
  -  -  -  -  - SB  -  -
  A  -  C  -  -  -  B  -
  -  -  -  A  - SC  -  -
  D  -  -  -  D  - SA
  - SB  - SD  -  -  -
  C  -  B  - SA  -
  -  -  -  -
```

```
! the detector map is upside down for denovo
  det_map
            1 - - 1 - - -
        1 - - - 1 - - 1 - 1 -
      - 1 - 1 - - 1 - - - - - 1
      - - - - 1 - - 1 - - 1 - -
    1 - - - 1 - - 1 - - 1 - - - 1
    - - - - 1 - 1 - - - - 1 - - -
    - 1 - - - - - - 1 - 1 - - - 1
    1 - 1 - 1 - 1 - - 1 - 1 1 1 -
    - - - 1 - - 1 - - 1 - - 1 - -
    1 - 1 - - 1 - 1 - - - - - 1 -
    - - - - 1 - - - 1 - 1 - 1 - -
      1 1 - - - - 1 - - - - - -
      - - - - - - 1 - 1 - 1 - 1
        1 - - 1 - 1 - - - - -
            - - 1 - - 1 -

  baffle ss 0.19 2.85
! vessel cs 219.71 21.99 - not functional yet

  lower_plate ss  5.0 0.5   ! mat, thickness, vol frac
  upper_plate ss  7.6 0.5   ! mat, thickness, vol frac

! lower_ref  mod 20.0 1.0   ! reduce this as much as possible for now
! upper_ref  mod 20.0 1.0

  xlabel   R  P  N  M  L  K  J  H  G  F  E  D  C  B  A
  ylabel  01 02 03 04 05 06 07 08 09 10 11 12 13 14 15

  mat he     0.0001786
  mat inc    8.19
  mat ss     8.0
  mat zirc   6.56 zirc4
  mat spring 0.4 ss 0.05 he 0.95

[ASSEMBLY]
  include WBN1_C1+2_ASY.inp

[INSERT]
  title "Pyrex"
  npin 17

  mat pyrx1 2.25 pyrex-vera

  cell 1  0.214 0.231 0.241 0.427 0.437 0.484 / he ss he pyrx1 he ss
  cell 2                          0.437 0.484 /            spring ss    ! plenum
  cell 3                                0.484 /                   ss    ! plug

  rodmap  PY8
     -
     - -
     - - -
     1 - - -
     - - - - -
     - - - - - 1
     - - - - - - -
     - - - - - - - -
     - - - - - - - - -
```

```
rodmap  PLEN8
   -
   - -
   - - -
   2 - - -
   - - - - -
   - - - - - 2
   - - - - - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PLUG8
   -
   - -
   - - -
   3 - - -
   - - - - -
   - - - - - 3
   - - - - - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PY12
   -
   - -
   - - -
   1 - - -
   - - - - -
   - - - - - -
   - - - 1 - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PLEN12
   -
   - -
   - - -
   2 - - -
   - - - - -
   - - - - - -
   - - - 2 - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PLUG12
   -
   - -
   - - -
   3 - - -
   - - - - -
   - - - - - -
   - - - 3 - - -
   - - - - - - - -
   - - - - - - - - -
```

```
rodmap  PY16
   -
   - -
   - - -
   1 - - -
   - - - - -
   - - - - - 1
   - - - 1 - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PLEN16
   -
   - -
   - - -
   2 - - -
   - - - - -
   - - - - - 2
   - - - 2 - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PLUG16
   -
   - -
   - - -
   3 - - -
   - - - - -
   - - - - - 3
   - - - 3 - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PY20
   -
   - -
   - - -
   1 - - -
   - - - - -
   - - - - - 1
   1 - - 1 - - -
   - - - - - - - -
   - - - - - - - - -

rodmap  PLEN20
   -
   - -
   - - -
   2 - - -
   - - - - -
   - - - - - 2
   2 - - 2 - - -
   - - - - - - - -
   - - - - - - - - -
```

```
  rodmap  PLUG20
     -
     - -
     - - -
     3 - - -
     - - - - -
     - - - - - 3
     3 - - 3 - - -
     - - - - - - - -
     - - - - - - - - -

  rodmap  PY24
     -
     - -
     - - -
     1 - - 1
     - - - - -
     - - - - - 1
     1 - - 1 - - -
     - - - - - - - -
     - - - - - - - - -

  rodmap  PLEN24
     -
     - -
     - - -
     2 - - 2
     - - - - -
     - - - - - 2
     2 - - 2 - - -
     - - - - - - - -
     - - - - - - - - -

  rodmap  PLUG24
     -
     - -
     - - -
     3 - - 3
     - - - - -
     - - - - - 3
     3 - - 3 - - -
     - - - - - - - -
     - - - - - - - - -

  axial   8  13.221 PLUG8  15.761 PY8  376.441 PLEN8  398.64
  axial  12  13.221 PLUG12 15.761 PY12 376.441 PLEN12 398.64
  axial  16  13.221 PLUG16 15.761 PY16 376.441 PLEN16 398.64
  axial  20  13.221 PLUG20 15.761 PY20 376.441 PLEN20 398.64
  axial  24  13.221 PLUG24 15.761 PY24 376.441 PLEN24 398.64

[CONTROL]
  title "B4C with AIC tips"
  npin 17
  stroke  365.125 230     ! approx for 1.5875 step sizes and 230 max stroke

  mat aic 10.2
  mat b4c 1.76

  cell 1  0.382 0.386 0.484 / aic he ss
  cell 2  0.373 0.386 0.484 / b4c he ss
```

```
  cell 3        0.386 0.484 / spring ss !plenum
  cell 4              0.484 /        ss !plug

  rodmap AIC
     -
     - -
     - - -
     1 - - 1
     - - - - -
     - - - - - 1
     1 - - 1 - - -
     - - - - - - - -
     - - - - - - - - -

  rodmap B4C
     -
     - -
     - - -
     2 - - 2
     - - - - -
     - - - - - 2
     2 - - 2 - - -
     - - - - - - - -
     - - - - - - - - -

  rodmap PLEN
     -
     - -
     - - -
     3 - - 3
     - - - - -
     - - - - - 3
     3 - - 3 - - -
     - - - - - - - -
     - - - - - - - - -

  rodmap PLUG
     -
     - -
     - - -
     4 - - 4
     - - - - -
     - - - - - 4
     4 - - 4 - - -
     - - - - - - - -
     - - - - - - - - -

  axial 1     15.131
        PLUG  17.031
         AIC 118.631
         B4C 377.711
        PLEN 388.411

[DETECTOR]
  title "Incore instrument thimble"
  npin 17

  mat he 0.0001786
  mat ss 8.0
```

```
  cell 1  0.258 0.382 / he ss

  rodmap  LAT
     1
     - -
     - - -
     - - - -
     - - - - -
     - - - - - -
     - - - - - - -
     - - - - - - - -
     - - - - - - - - -

  axial 1  0.0 LAT 406.337

[EDITS]
  include axial_mesh1.inp

[MPACT]
  vis_edits         none
  jagged            true
  coupling_method   ctf
  ith_hgap          10000
  ith_dhfrac        0.02
!quad_set
  quad_type         CHEBYSHEV-GAUSS
  polars_octant     4
  azimuthals_octant 16
  ray_spacing       0.05
!iteration_control
  flux_tolerance    1e-4
  num_inners        2
  k_tolerance       1e-5
  up_scatter        1
  num_outers        200
  scattering        TCP0
!cmfd
  cmfd              cmfd
  cmfd_solver       mgnode
  k_shift           1.15
  cmfd_num_outers   20
!2D1D
  split_TL          true
  TL_treatment      lflat
  nodal_method      nem
!parallel
  num_space         4234
  par_method        EXPLICITFILE
  par_file          partition_73r.txt
  num_angle         1
  num_energy        1
  num_threads       1
!xs_library
  xs_filename       mpact47g_70s_v4.0_11032014.fmt
  xs_type           ORNL
  dep_filename      MPACT.dpl
  dep_edit          false
  dep_substep       2
  subgroup_set      4
!mesh
```

```
  mesh fuel  3 1 1 / 8 8 8 8 8 8
  mesh gtube 3 1   / 8 8 8 8 8

  meshing_method    nonfuel
  automesh_bounds  2.0 10.0

[COBRATF]
  nfuel   8             ! number of fuel rings in conduction model
  nc      1             ! conduction option - radial conduction
  irfc    2             ! friction factor correlation default=2
  dhfrac  0.026         ! fraction of power deposited directly into coolant
  hgap    10000.0       ! gap conductance
  epso    0.001
  oitmax  5
  iitmax  40
  gridloss END 0.9070   ! spacer grid loss coefficient
  gridloss MID 0.9065   ! spacer grid loss coefficient
  dtmin   0.000001
  dtmax   0.1
  tend    0.1
  rtwfp   1000.0
  maxits  100000
  courant 0.8
  parallel 1

[COUPLING]
  epsk        1.0  ! pcm
  epsp        1.0e-4
  rlx_power   0.65
  rlx_tfuel   1.0
  rlx_den     1.0
  maxiter     1000
```

WBN1_C1+2_ASY.inp

```
!This is a VERA Input include file for the assemblies used in Watts Bar Unit 1
!cycle 1. This also applies to the VERA Benchmark Progression Problems 3
!through 10.
!
!Note that "he", "zirc", "mod", "spring", "lplug", "uplug" should be materials
!defined in the input that uses this file.

  title "WBN1C1+2 Assembly"
  npin 17
  ppitch 1.26

  fuel U21   10.257 94.5 / 2.11
  fuel U2613 10.257 94.5 / 2.613
  fuel U26   10.257 94.5 / 2.619
  fuel U31   10.257 94.5 / 3.1
  fuel U37   10.257 94.5 / 3.709
  mat ifba   3.85
        zr   0.81306
      b-10   0.09347
      b-11   0.09347

  cell 1   0.4096        0.418 0.475 / U21      he zirc
  cell 2   0.4096        0.418 0.475 / U26      he zirc
  cell 3   0.4096        0.418 0.475 / U31      he zirc
  cell 4   0.4096        0.418 0.475 / U37      he zirc  !Normal Fuel
  cell X   0.4096 0.4106 0.418 0.475 / U37 ifba he zirc  !Normal Fuel+IFBA
  cell B   0.4096        0.418 0.475 / U2613    he zirc  !Blanket enrichment
  cell G                 0.561 0.602 /         mod zirc  ! guide/instrument tube
  cell 5                 0.418 0.475 /      spring zirc  ! plenum with spring
  cell 6                       0.475 /              mod  ! lower end plug
  cell 7                       0.475 /              mod  ! upper end plug

  lattice FUEL21
       G
       1 1
       1 1 1
       G 1 1 G
       1 1 1 1 1
       1 1 1 1 1 G
       G 1 1 G 1 1 1
       1 1 1 1 1 1 1 1
       1 1 1 1 1 1 1 1 1

  lattice FUEL26
       G
       2 2
       2 2 2
       G 2 2 G
       2 2 2 2 2
       2 2 2 2 2 G
       G 2 2 G 2 2 2
       2 2 2 2 2 2 2 2
       2 2 2 2 2 2 2 2 2
```

```
  lattice FUEL31
       G
       3 3
       3 3 3
       G 3 3 G
       3 3 3 3 3
       3 3 3 3 3 G
       G 3 3 G 3 3 3
       3 3 3 3 3 3 3 3
       3 3 3 3 3 3 3 3 3

!This is a VERA Input include file for the assemblies used in Watts Bar Unit 1
!cycle 2. This also applies to the VERA Benchmark Progression Problem 10.
!
!This defines the 4 feed assemblies for cycle 2

  lattice BLANKET
       G
       B B
       B B B
       G B B G
       B B B B B
       B B B B B G
       G B B G B B B
       B B B B B B B B
       B B B B B B B B B

  lattice FUEL37
       G
       4 4
       4 4 4
       G 4 4 G
       4 4 4 4 4
       4 4 4 4 4 G
       G 4 4 G 4 4 4
       4 4 4 4 4 4 4 4
       4 4 4 4 4 4 4 4 4

  lattice IFBA048
       G
       X 4
       4 4 4
       G X 4 G
       4 4 4 X 4
       4 4 4 4 X G
       G X 4 G X 4 4
       4 4 4 4 4 4 4 4
       4 4 4 4 4 4 4 4 X

  lattice IFBA104
       G
       X 4
       X 4 4
       G X X G
       X 4 4 X 4
       X 4 4 X X G
       G X X G X X 4
       X 4 4 X 4 4 4 4
       4 4 4 4 4 4 4 4 X
```

```
  lattice IFBA128
       G
       X 4
       X 4 4
       G X X G
       X 4 4 X 4
       X 4 4 X X G
       G X X G X X 4
       X 4 4 X 4 4 X 4
       4 X 4 4 X 4 4 4 X

  lattice PLEN
       G
       5 5
       5 5 5
       G 5 5 G
       5 5 5 5 5
       5 5 5 5 5 G
       G 5 5 G 5 5 5
       5 5 5 5 5 5 5 5
       5 5 5 5 5 5 5 5 5

  lattice LPLUG
       G
       6 6
       6 6 6
       G 6 6 G
       6 6 6 6 6
       6 6 6 6 6 G
       G 6 6 G 6 6 6
       6 6 6 6 6 6 6 6
       6 6 6 6 6 6 6 6 6

  lattice UPLUG
       G
       7 7
       7 7 7
       G 7 7 G
       7 7 7 7 7
       7 7 7 7 7 G
       G 7 7 G 7 7 7
       7 7 7 7 7 7 7 7
       7 7 7 7 7 7 7 7 7
!
!Cycle 1 batches
  axial B1000   6.053 LPLUG 11.951 FUEL21 377.711 PLEN 393.711 UPLUG 397.510
  axial B2000   6.053 LPLUG 11.951 FUEL26 377.711 PLEN 393.711 UPLUG 397.510
  axial B3000   6.053 LPLUG 11.951 FUEL31 377.711 PLEN 393.711 UPLUG 397.510
!
!Cycle 2 batches
  !NO IFBA
  axial B4000 6.053 LPLUG 11.951 BLANKET 27.191 FUEL37 362.471 BLANKET 377.711
        PLEN 393.711 UPLUG 397.510

  !48 IFBA
  axial B4048 6.053 LPLUG 11.951 BLANKET 27.191 FUEL37 42.431 IFBA048 347.231
        FUEL37 362.471 BLANKET 377.711 PLEN 393.711 UPLUG 397.510

  !104 IFBA
  axial B4104 6.053 LPLUG 11.951 BLANKET 27.191 FUEL37 42.431 IFBA104 347.231
```

```
        FUEL37 362.471 BLANKET 377.711 PLEN 393.711 UPLUG 397.510

  !128 IFBA for 120"
  axial B4128 6.053 LPLUG 11.951 BLANKET 27.191 FUEL37 42.431 IFBA128 347.231
        FUEL37 362.471 BLANKET 377.711 PLEN 393.711 UPLUG 397.510

  !128 IFBA for 132"
  axial B4128L 6.053 LPLUG 11.951 BLANKET 27.191 IFBA128 362.471 BLANKET 377.711
        PLEN 393.711 UPLUG 397.510

  grid MID zirc 875  3.810  !name, material, mass (g), height (cm)
  grid END inc  1017 3.866

!Axial locations (center of grid) of grid types in assembly (cm)
  grid_axial
      END  13.884
      MID  75.2
      MID 127.4
      MID 179.6
      MID 231.8
      MID 284.0
      MID 336.2
      END 388.2

  lower_nozzle  ss 6.053 6250.0  ! mat, height (cm), mass (g)
  upper_nozzle  ss 8.827 6250.0  ! mat, height (cm), mass (g)
```

axial_mesh_1.inp

```
!Axial mesh 1

  axial_edit_bounds
    11.951
    15.817
    27.191
    34.811
    42.431
    50.147
    57.863
    65.579
    73.295
    77.105
    85.17
    93.235
   101.3
   109.365
   117.43
   125.495
   129.305
   137.37
   145.435
   153.5
   161.565
   169.63
   177.695
   181.505
   189.57
   197.635
   205.7
   213.765
   221.83
   229.895
   233.705
   241.77
   249.835
   257.9
   265.965
   274.03
   282.095
   285.905
   293.97
   302.035
   310.1
   318.165
   326.23
   334.295
   338.105
   347.231
   354.851
   362.471
   370.091
   377.711
```

APPENDIX B - T/H PARAMETER SENSITIVITY

For this mini-study a 3x3 rod cluster was simulated for full height. The heat transfer coefficient of the fuel-clad gap h_{gap} and direct moderator heating fraction dh_{frac} were varied and the metric compared is the "core" averaged fuel temperature. The target core averaged fuel temperature is 835 K. The goal of this study is to quantify the effect of these parameters on the core averaged fuel temperature and to determine the combination of h_{gap} and dh_{frac} that result in a core averaged fuel temperature near 835 K.

Table B.1: Fuel Temperature Sensitivity on dh_{frac} (h_{gap} = 4,500 [W/m^2-K])

dh_{frac}	Avg. Fuel Temp. (K)	ΔT (K)
0.000	956.0	
0.020	945.8	10.2
0.025	943.3	12.7
0.030	940.8	15.2

Table B.2: Fuel Temperature Sensitivity on h_{gap} (dh_{frac} = 0)

h_{gap} [W/m^2-K]	Avg. Fuel Temp. (K)	ΔT (K)
2,000	1202.1	
4,000	981.0	221.2
6,000	905.7	296.5
8,000	867.6	334.5
10,000	844.6	357.5

Using a combined dh_{frac} = 0.02 and h_{gap} = 10,000 [W/m^2-K] and resulted in a core average fuel temperature of 837 K.