LAPPEENRANTA-LAHTI UNIVERSITY OF TECHNOLOGY LUT
School of Energy Systems
Energy Technology

Yohannes Molla

OPENMC MODELING OF THE CRITICAL MYRRHA CONFIGURATION: AN EMPHASIS IN CROSS SECTION HOMOGENIZATION

Examiners: Professor Juhani Hyvärinen (LUT University)

Asst. Professor Heikki Suikkanen (LUT University)

Mentor: Dr. Augusto Hernandez Solis (SCK•CEN, Belgian Nuclear Research Center)

ABSTRACT

Yohannes Molla
School of Energy Systems
Examiners: Prof. Juhani Hyvärinen (LUT University)
Asst. Prof. Heikki Suikkanen (LUT University)
Mentor: Dr. Augusto Hernandez Solis (SCK•CEN, Belgian Nuclear Research Center)
Master's Thesis 2019
87 Pages, 35 Figures, 9 Tables

Keywords: MYRRHA, OpenMC, PHISICS, Homogenization, Multi-group Monte Carlo

MYRRHA-ADS is a research reactor, which is under design and development by the Belgian Nuclear Research Center, SCK•CEN. The reactor will have innovative applications such as, transmutation of high-level waste via fast spectrum, radioisotope production and silicon doping using thermal neutrons moderated by water loops. It is the first of its kind to be accompanied by a particle accelerator and cooled by Lead-Bismuth eutectic (LBE).

The strongly heterogeneous nature of this reactor offers a rare opportunity to study both thermal and fast spectra. Given its unique character, a critical core configuration based on the MYRRHA-1.6 was chosen to investigate the capabilities of the state-of-the-art OpenMC Monte Carlo code in computing global and local 3D neutronic observables. Moreover, homogenized multi-group macroscopic cross sections were generated using OpenMC for further utilization in a deterministic core simulator.

Benchmarking of OpenMC with respect to a well-validated Monte Carlo code, MCNP at core level, is one of the novel works of this thesis. The k_{eff} difference and the relative difference of energy integrated flux in the active region were within acceptable range. In addition, a code verification between a homogenized multi-group Monte Carlo (MGMC) OpenMC model and the PHISICS deterministic simulator was carried-out. Here, a significant eigen value bias was observed. Further, a comparison was made between the continuous energy heterogeneous model and the MGMC model showed that there is a highly significant eigen value difference. Possible reasons for these biases were identified and possible ramification techniques are suggested.

ACKNOWLEDEGMENTS

Foremost, I would like to thank Professor Juhani Hyvärinen for giving me the opportunity to become one of his esteemed students. I also owe gratitude to his savvy advice and mentorship, which has been an inspiration throughout my study years.

I would also like to take this opportunity to thank Asst. Professor Heikki Suikkanen for the supervision of this work. His criticial reviews, insightful comments, suggestions and ideas were pivotal for the final composure of the thesis.

My utmost gratitude goes-out to my advisor, mentor and a friend in need, Dr. Augusto Hernandez Solis, who has not only been supportive of my work through his motivation, enthusiasm and immense knowledge sharing, but also for making my stay in Mol, Belgium as smooth and memorable as possible. Thank you is not enough!

I thank colleagues and scientists at SCK•CEN with whom I have had the pleasure of working with. The Belgian Nuclear Research Center Academia also deserves an acknowledgment for giving opportunities for students like myself to have a hands-on research practices with respected experts of the Nuclear field.

Whenever I think of the person I have become today, I remember my parents, siblings and my extended family, who have implanted the attitude of working hard towards the greater good of mankind, whilst having love and respect for humanity and nature, in my formative years. I am forever grateful for having you in life. You all are always in my thoughts!

Finally, to the one person who have been by my side in happiness or hardships for the past eleven years. Your relentless encouragements and passion for my betterment made all these possible. But, most of all I am profoundly grateful for the love, care and the unshakeable confidence that you have in me, which transcends all bounds. Dearest Winnie, like I always say, I will always Love You!

Yohannes Molla

February 2020

Espoo, Finland

TABLE OF CONTENTS

SYMBOLS AND ABBREVIATIONS

Latin letters

A	L^{th} order coefficient of Legendre polynomial	
D	Diffusion coefficient	
E	energy	J, MeV
E'	energy of next phase space	J, MeV
J	neutron current	$1/(cm^2 s)$
k	multiplication factor/ eigen value	
n	angular neutron density	
N	neutron density	
$\bar{P}$	scattering probability matrix	
P	Legendre polynomial	
P_J	Number of source sites in the J^{th} mesh	
P_s	probability of survival	
R	reaction rate	$1/(cm^3 s)$
S	surface	
t	time	sec

v	velocity	m/s
V	volume	cm^3
w_i	pre-collision weight of particles	
W	total starting number of particles	

Greek letters

Δ	change	
δ	diagonals of isotropic scattering matrix	
ϕ	angular flux	$1/cm^2/s$
Φ	scalar flux	$1/cm^2/s$
ρ	reactivity	pcm
Σ	macroscopic cross-section	1/cm
v	number of new neutrons per fission	
χ	neutron fission spectrum	
Ω	solid angle	◦, rad
μ, η, ξ	direction cosines	

Subscript

eff	effective
f	fission

g	group
g'	next group
i	event index
j	multiplicity reaction
K, n	region in a volume integral
ℓ	number of orders in Legendre polynomials
r	position
s	scattering
t	total
tr	transport
x	reaction

Symbols

ℓ_i	length of the i^{th} trajectory
υ_j	scattering multiplicity for the j^{th} multiplicity reaction

Abbreviations

ACE	A Compact ENDF
ADONIS	Accelerator Driven Optimized Nuclear Irradiation System
ADS	Accelerator Driven System
API	Application Programming Interface
BOC	Beginning of Cycle
CMFD	Coarse Mesh Finite Difference
ENDF	Evaluated Nuclear Data File
ERANOS	European Reactor Analysis Optimized calculation System
FASTEF	Fast Spectrum Transmutation Experimental Facility
HDF	Hierarchical Data Format
HLW	High-level Long-lived radioactive Waste
INL	Idaho National Laboratory
INSTANT	Intelligent Nodal and Semi-structured Treatment for Advanced Neutron Transport
IPS	In-Pile test Section
JEFF	Joint Evaluated Fission and Fusion File
LBE	Lead-Bismuth Eutectic

MC	Monte Carlo
MCNP	Monte Carlo N-Particle
MGMC	Multi-group Monte Carlo
MGXS	Multi-group Cross Section
MGSXS	Multi-group Scattering Cross Section
MOX	Mixed Oxide
MPI	Message Passing Interface
MYRRHA	Multi-purpose hYbrid Research Reactor for High-tech Applications
OMP	Open Message Passing
PHISICS	Parallel and Highly Innovative Simulation for INL Code System
VTT	Teknologian tutkimuskeskus VTT
XT	eXperimental facility demonstrating the technical feasibility of Transmutation

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

The global electricity demand is increasing with a fast pace, especially since 2010. For instance, according to a report by the International Energy Agency it has risen by 2.3% in the previous year (IEA Report, 2019). Consecutively, this high demand put a heavy hand on the overall countries energy policy, which also depends on the security, sustainability, and affordability of the energy source. So far, no one energy source can guaranty to deliver sufficient power while respecting these three requirements. For example, renewable energy sources may be favored as sustainable; however, they are intermittent and expensive. In the case of fossil fuels, even though they are relatively secure and affordable, if all energy production relayed on them, the global initiative to minimize CO_2 would fall into jeopardy.

Therefore, it can reasonably be predicted that nuclear energy would be a viable energy source, at least for the foreseeable future. Nevertheless, most power-producing reactors of the present time are thermal spectrum reactors, and they rely on a small fraction of the natural uranium reserve. Hence, the development of fast spectrum reactor technology has been looked forward by many as a mitigation strategy for this challenge since it can increase the energy yield from natural uranium. Moreover, this technology provides a sustainable solution for nuclear energy's Achilles heel, i.e., the disposal of long-lived high-level radioactive waste. Even though, some nuclear power countries decide to store their high-level-waste in underground repositories, for most others, it is unacceptable since the length of time required to stabilize the waste is beyond historical records. For the later ones, transmuting the long-lived HLW to reduce the time scale is more appealing.

The idea of MYRRHA ADS was first conceived by parties who are concerned about fuel development and transmutation of HLW. Hence, MYRRHA is a Lead Bismuth eutectic (LBE) cooled fast spectrum reactor with a capability of sustaining a constant neutron flux level in a subcritical state via spallation reactions and forming the basis of an accelerator-driven-system. Most of all, the development of this reactor is a groundbreaking achievement for science and research in the field of nuclear science.

1.1 Scope of the work

This Master's thesis set-out to model the critical MYRRHA 1.6 core using an open-source Monte Carlo code known as OpenMC and to produce a database of spatially homogenized

and energy-collapsed multi-group macroscopic cross-sections. Then, the generated multi-group constants are implemented into the multi-group Monte Carlo (MGMC) feature of OpenMC and the deterministic transport solver of PHISICS toolkit. The aim is twofold, firstly it is to benchmark MGMC with respect to continuous energy heterogenous model. This has helped to investigate how well the physical characteristics were preserved during the process of homogenization. Secondly, it is to verify the PHISICS code with respect to MGMC simulation model of MYRRHA 1.6.

In addition, one section is dedicated to benchmarking OpenMC 0.10.0 simulation results with a well-established reference code, MCNP 6.2. The thesis also includes in-depth explanations about the MYRRHA reactor and Monte Carlo calculations from the perspective of OpenMC, such as:

- The historical path that led to the current design of MYRRHA reactor.
- The theoretical aspect of Monte Carlo calculation in relation to reactor physics.
- The usage and features of the OpenMC code from the user point of view.
- The mathematical approach to the homogenization process in the context of OpenMC.
- Result estimation in OpenMC.
- And, result interpretation.

Lastly, the perspective of the writer on the outcomes of the simulation and suggestions for future work are illustrated.

2 MYRRHA REACTOR

The name MYRRHA signifies a Multi-purpose hYbrid Research Reactor for High-tech Applications. The fact that it can be operated either in critical or sub-critical conditions will make the reactor one of its kind upon its completion. Obviously, a core loading alternation will be required to shift the reactor from one mode to another. The sub-critical chain reaction is supplemented by a spallation neutron source. A high energy proton beam is designed to collide with lead-bismuth eutectic, LBE, a material which is also planned to be used as a core coolant, to cause a spallation neutron source (Malambu & Stankovskiy, 2014).

Handling the High-Level and Long-Lived Radioactive Waste has always been a challenge for the nuclear industry. Numerous ideas were forwarded in the past, on how to handle this issue. To this end, spent fuel underground repositories have become widely accepted by some nuclear power countries. For instance, Finland is a pioneer in building a permanent underground repository. However, scientists on the other side of the isle argued that a geological solution will take tens of thousands of years for a nuclear waste to decay to a safe level. Hence, they proposed an accelerator driven transmutation of waste. Charles Bowman was one of the proponents of this idea since the 90's. MYRRHA is also a progeny of this idea.

The application goal of MYRRHA ADS can be summarized into three major parts (Abderrahim, et al., 2012):

- To demonstrate the concept of transmutation of minor actinides.
- To validate the ADS concept.
- To operate as a material irradiation facility.

2.1 Historical background

The idea of building such an innovative reactor was conceived in the 1990's at the Belgian Nuclear Research Center, SCK•CEN, based on the ADONIS project (1995-1997) (Abderrahim, et al., 2012). The ADONIS (Accelerator Driven Optimized Nuclear Irradiation System) project was executed in the framework of studying about the coupling of a proton accelerator, a spallation target and a subcritical core. In addition, it was intended for medical radioisotope production. ADONIS was planned to be a light water reactor fitted with a 150

MeV accelerator at 1.5MWth nominal power. However, the project was terminated at the design stage with an idea of extending it to a larger ADS multi-purpose research facility.

The work on MYRRHA began in 1998 and in 2002 the first design 'MYRRHA Draft 1' was submitted to the International Technical Guidance Committee for comment. Then, in 2005 an upgraded 'MYRRHA Draft 2' was published. The last draft became a springboard for the MYRRHA/XT-ADS design.

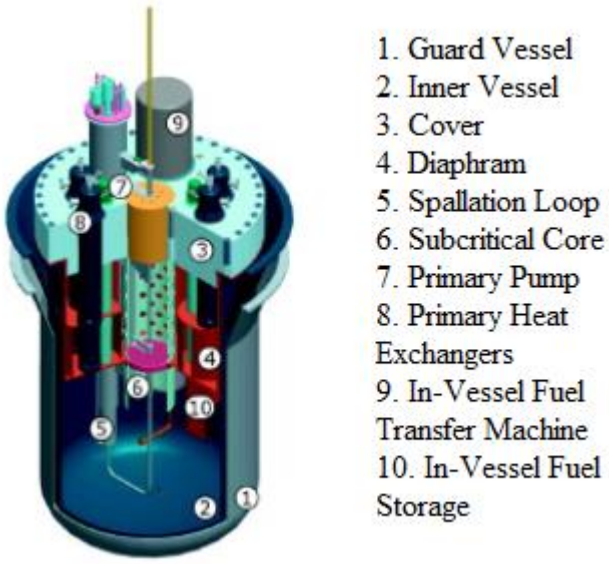


Figure 1: MYRRHA/XT-ADS model. (Abderrahim, et al., 2012)

Figure 1 shows the MYRRHA/XT-ADS, it was designed as a pool type reactor cooled by Lead Bismuth Eutectic (LBE). The main design parameters of the reactor are listed in Table 1 (Abderrahim, et al., 2012).

Even though this reactor design complied with the main design and safety requirements, it was not able to fulfill the objectives of MYRRHA. That is, MYRRHA/XT-ADS was only capable of operating in sub-critical mode, and it was not able to reach the required irradiation target.

Table 1: MYRRHA XT-ADS main parameters. (Abderrahim, et al., 2012)

Nominal reactor power	57 MW
Core cooling power	70 MW
Primary side inlet temperature	300 ^{o}C
Primary side outlet temperature	400 ^{o}C
Coolant velocity	2 m/s
Primary coolant	LBE
Secondary coolant	Steam
Tertiary coolant	Air

These hindrances triggered the necessity for future design improvement with the aforementioned objective. Thus, since 2009 the project commenced with a name MYRRHA-FASTEF. Figure 2 depicts the MYRRHA-FASTEF reactor.

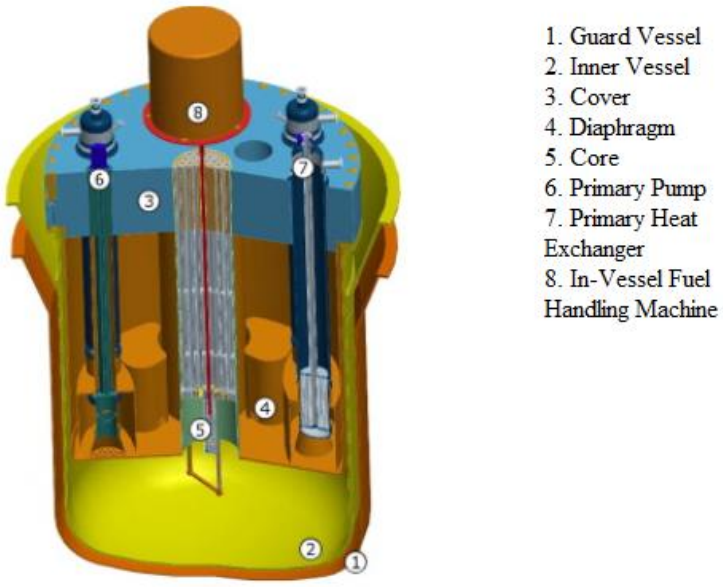


Figure 2: MYRRHA-FASTEF. (Abderrahim, et al., 2012)

As a critical reactor, the reactivity control and shout down mechanisms were needed in this design. Thus, the reactivity safeguards were added. Moreover, the power and power density

were increased to meet the high flux requirement. In addition, the safety systems of the facility were modified. Table 2 shows the design parameters of the reactor.

In 2014 a successful design revision was made by the Central Design Team to address further requests regarding core management. The revised design is named MYRRHA Rev1.6. The design reform was made under the following guidelines (Malambu & Stankovskiy, 2014):

I. Operating well below the maximum cladding temperature.
II. Maximize the fuel burn up discharge for as long as possible and minimize the fresh fuel intake in every cycle.
III. Reduction of irradiation damage on the outer barrel.
IV. Optimization of radioisotope production and silicon doping.

Table 2: MYRRHA-FASTEF main parameters. (Abderrahim, et al., 2012)

Nominal power	100 MW
Core inlet temperature	270 ^{o}C
Core outlet temperature	410 ^{o}C
Coolant velocity in core	2 m/s
Coolant pressure drop	2.5 bar
Primary coolant	LBE
Secondary coolant	Steam
Tertiary coolant	Air

2.2 MYRRHA Rev1.6 critical core design description

Core design is mainly governed by neutronic and thermal-hydraulic limitations (Malambu & Stankovskiy, 2014). As it was stated in the previous sub-sections, MYRRHA is a fast reactor with an objective of burning minor actinides and fission products. Hence, there is a quest for high fast flux for transmutation of this HLW. However, high and fast flux generation, which follows high power production is limited by coolant velocity and

maximum cladding temperature. Table 3 illustrates the key reactor physics design parameters of MYRRHA 1.6 critical core.

Table 3: Key design parameters of MYRRHA 1.6 critical core. (Malambu & Stankovskiy, 2014)

Core configuration	Starting up BOL	Equilibrium BOC
No. of fuel assembly	78	108
Admissible max. power (at 466°C Cladding temp.) [MW_{th}]	88	96
Admissible linear power (at 466°C Cladding temp.) [W/cm]	217	212

2.2.1 Core layout

The MYRRHA core design consists of hexagonal lattices of two kinds (Figure 3). The ones marked by black dot are accessible from the top. These sub-channels include In-Pile-Section (IPS) for material irradiation, control rod banks, shutdown systems and the spallation target for sub-critical operation. The rest of the channels are going to be accessed from the bottom. The second group of sub-channels includes fuel, shielding and reflector sub-assemblies. The whole core is contained in a barrel, but between the core assembly and the barrel there is additional steel shielding. (Eynde, et al., 2015)

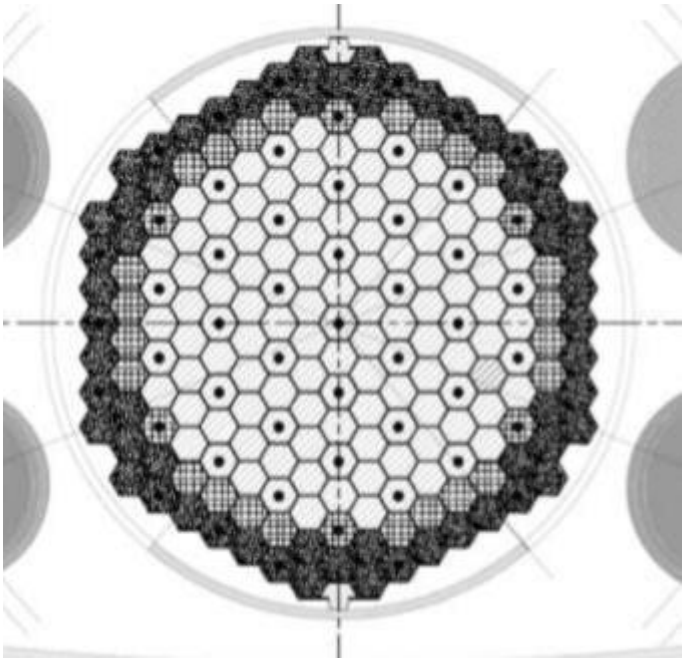


Figure 3: The MYRRHA core lay out. Channels marked with a black dot are accessible from the top during operation. (Malambu & Stankovskiy, 2014)

2.2.2 Critical core fuel management

One way of achieving the required flux intensity without exceeding the coolant mass flow rate and cladding temperature thresholds, is by implementing an appropriate fuel loading pattern. To begin with a 30 wt. % of enriched MOX is chosen based on the neutronic requirements and market availability (Eynde, et al., 2015). A more detailed explanation about fuel assembly configuration and dimensions can be found in Section 6.

At the beginning of life (BOL) 78 fresh fuel assemblies will be loaded with an addition of a fresh batch of six fuel assemblies at most inner position of the core in every cycle (each cycle consists of 90 days). The older batches will be reshuffled from in-to-out until the 18^{th} batch. The equilibrium cycle, also known as the beginning of cycle (BOC), will commence after loading the 18^{th} batch. Then, at the end of every cycle (EOC) six assemblies of the oldest batch will be removed and the same number of fresh fuels will be added with in-to-out shuffling manner. The fuel pin and coolant material specification are shown in Table 4.

Table 4: Core material specification. (Solis, 2018)

Fuel material	MOX
Enrichment	30% [HM]
Fuel density	10.5 [g/cm^3]
Admissible max. fuel temp.	1300 [K]
Cladding material	15-15 Ti SS
Cladding density	7.95 [g/cm^3]
Admissible max. cladding temp.	700 [K]
Coolant	LBE
Coolant density	10.3 [g/cm^3]
Control rod	B_4C

3 REACTOR PHYSICS CALCULATIONS

The importance of reactor physics had become vivid since the 1940's when it was discovered that a sustained chain reaction is a requirement for any civilian or military nuclear application. Nuclear reaction is the interaction between a flux of neutrons in phase space and fuel nuclide as a function of time.

However, since the neutron flux is coupled with the three aspects of the phase space in a complex manner, it is not possible to solve the equation analytically. Thus, a solution can only be obtained either through statistical method or numerical methods. The most common statistical and numerical methods known are Monte Carlo method and deterministic methods, respectively.

As mentioned in Section 1.1 one of the emphases in this thesis is generation of homogenized multi-group cross sections. Homogenized multi-group cross sections are the probability of occurrence of different interaction constants that are spatially averaged, and energy condensed (i.e. integrated). These multi-group cross sections have multiple applications in reactor physics. They are fundamentally used in diffusion theory and transport calculation codes. In addition, they can also be used in Monte Carlo codes to reduce the running time of calculations (Pirouzmand & Mohammadhasani, 2015).

3.1 Monte Carlo method in reactor physics

The application domain of the Monte Carlo method is far beyond reactor physics calculations. It is involved in all areas of engineering. The method is preferred since it is an intuitive, simple and statistical method of analyzing complex problems consisting of several well-defined sub-tasks. It is preferred in reactor physics calculations mostly for its simplicity and the accurate results it produces (Leppänen, 2007). Moreover, discretization and homogenization of the geometry of the reactor is not needed as in the case of deterministic methods (Leppänen, 2007). Due to these characteristics it is favorable in benchmarking deterministic codes, and in scientific researches of experimental nature (Hebert, 2016).

In addition, some characteristics of neutron-nucleus interaction, such as: the fact that neutrons interact with their surrounding and not amongst themselves, linearity of the transport process, the Markov process and the isotropic nature of materials in space makes the method even simpler. (Hebert, 2016)

Albeit, there has been a common misunderstanding that Monte Carlo codes solve the neutron transport equation. This is a rather wrong understanding since Monte Carlo is a statistical method, which deals with a random walk of a single neutron at a time (Leppänen, 2007). Actually, one of the strengths of the method is its ability to estimate integral reaction rates without solving the equation for flux distribution. The method works in such a way that various interactions that may occur between a particle and the surrounding nuclei during its lifetime are randomly sampled and simulated. In reactor Monte Carlo calculations, neutrons are introduced into a nuclear system in batches (Leppänen, 2007). Since the method is based on stochastic statistics, the accuracy of the result depends on both the number of neutrons in each batch and the total number of batches (Cai, 2014). Besides, the standard-deviation of the statistics indicates the accuracy of the computation (Hebert, 2016).

In reactor Monte Carlo simulations, there are three main processes that should be considered to determine the neutron transport. Firstly, the source location should be sampled based on its probability distribution. Second is tracking of neutron location, reactions, energy and trajectory. Thirdly, its collecting and analyzing results. An algorithm of Monte Carlo (MC) particle simulation is shown in Figure 4. (Wu, 2019)

A major drawback of this kind of simulation is that high precision results demand substantial computing cost. Evidently, it is not practical to solve transport calculations of large scale with this method. Therefore, deterministic methods become the dominant approach in this regard. (Cai, 2014)

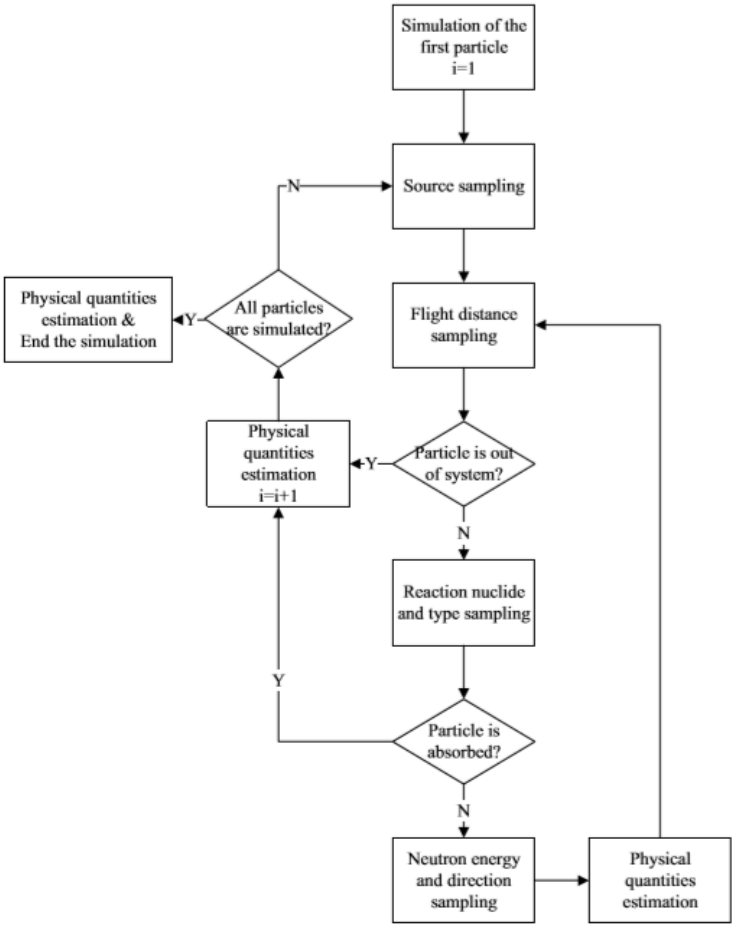


Figure 4: Algorithm of Monte Carlo transport simulation. (Wu, 2019)

For advanced and complete understanding of the mathematical proof of Monte Carlo method for reactor physics, one may refer to Jakko Leppanen's dissertation (Leppänen, 2007) or from the book by Hebert Alain (Hebert, 2016). Other authors like Lewis and Miller, Spanier and Gelbard, Llux and Koblinger also wrote extensively about the theoretical background of the method.

Based on the mathematical theory, a number of Monte Carlo based continuous-energy codes have been developed. Such codes as MCNP, TRIPOLI and SERPENT are among the well-known and well-established. However, for this thesis a niche Monte Carlo code known as OpenMC is used to calculate the effective multiplication factor, tally flux distribution and generate multi-group cross sections.

3.2 OpenMC: A Monte Carlo code

OpenMC is a Monte Carlo particle transport code for neutron criticality calculations. It was first released to the public in December 2012. Massachusetts Institute of Technology (MIT) developed it as a part of a project to bring about scalable parallel algorithms for future supercomputers. The code is written in standard Fortran 2008 (up to its version 0.10.0), and it possesses a number of attractive features to the user. First and for most, it is an open source code, thus many under privileged researchers will have access to a power of Monte Carlo code. Secondly, it is equipped with an extensive range of python functions such as Python API, which makes creating executable files, and post processing relatively easier. In addition, it supports both continuous-energy and multi-group transport data. (Romano, et al., 2014)

3.2.1 User input

The user input is designed in such way that it is comprehensive to a new user and at the same time, convenient for the developers to modify and extend the code script. The code requires the following inputs from the user in order to model and execute:

- A description of the geometry.
- A description of the nuclides and density of the constituent material.
- The number of particles to simulate and score.
- A list of required physical quantities.

The input file in OpenMC is structured in XML format, unlike other similar codes which use ASCII file with cards. The eXtensible Markup Language was chosen to organize the inputs in a sensible manner, so that the above-mentioned premises could be fulfilled, i.e. input insertion becomes easy to be visually inspected and determined, also it is easier for programmers to write the script that reads the input. Thus, the inputs are categorized into multiple files that are logically assigned (Romano, et al., 2012). The compulsory XML files for every simulation are:

- **`materials.xml`** – Contains the material composition file of the model. The materials are listed by their nuclide composition and density at a given temperature (see Figure 5).
- **`geometry.xml`** – Contains the material filled geometry file of the model.

- **`settings.xml`** – Is a file that contains all the simulation parameters.

In addition, there are another three optional ones, which are named as tallies.xml, plots.xml and cmfd.xml.

Figure 5 and Figure 6 below show material and geometry definitions in OpenMC consecutively.

```
m101 = openmc.Material(name='Fuel Batch 1')
m101.set_density('g/cc' ,10.499)
m101.temperature = 1300.0
m101.add_nuclide('O16'  ,1.16723E-01,'wo')
m101.add_nuclide('U234' ,1.55964E-04,'wo')
m101.add_nuclide('U235' ,4.40621E-03,'wo')
m101.add_nuclide('U238' ,6.13744E-01,'wo')
m101.add_nuclide('Pu238',5.95819E-03,'wo')
m101.add_nuclide('Pu239',1.48338E-01,'wo')
m101.add_nuclide('Pu240',7.04005E-02,'wo')
m101.add_nuclide('Pu241',1.40236E-02,'wo')
m101.add_nuclide('Pu242',2.00668E-02,'wo')
m101.add_nuclide('Am241',6.18314E-03,'wo')
materials_file = openmc.Materials([m101])
materials_file.export_to_xml()
```

Figure 5: Material definition in OpenMC.

```
# Instantiate Cell
cell2_fa        = openmc.Cell()
# Axial planes of the domain
z1n             = openmc.ZPlane(z0=-32.5)
z1p             = openmc.ZPlane(z0= 32.5)
# Surface of the domain
surf_12         = openmc.ZCylinder(x0=0, y0=0, R=0.2710)
# Creating a region: combin surface & planes
cell2_fa.region = -surf_12 & -z1p & +z1n
# Filling the region with material
cell2_fa.fill   = m101
root            = openmc.Universe([cell2_fa])
geometry        = openmc.Geometry(root)
geometry.export_to_xml()
```

Figure 6: Defining geometry in OpenMC.

Once the inputs are provided to the code, it will start performing the simulation one particle at a time (for detailed execution program flow). Readers, interested in detailed execution program flow, are advised to take a look at the OpenMC manual (Romano, 2018).

3.2.2 Geometry

OpenMC code uses constructive solid geometry (CSG) to define the geometry of the model. CSG is a mathematical technique, which enables the user to create complex regions by combining primitive regions using Boolean operators.

The geometry construction hierarchy of the code organized in such a way that surfaces join to create a region. However, the surfaces should first be built by a composition of planes. This means that to model a region, one has to begin by defining its most primitive predecessor, the plane. After, the surfaces are created; they must be referenced as a half-space to form a volume. A half-space of a surface can be defined as a region whose points satisfy a positive or negative inequality of the surface equation. Thus, the negative half-space represents the inside of the surface, while the positive depicts the outside.

In the Python API, planes and surfaces are created through subclasses of **`openmc.Plane`** (this has a 3D functionality) and **`openmc.Surface`** consecutively. The half-spaces of the region are represented by an antecedent, - or +, operator depending on the position of the space with respect to the region. Moreover, half-spaces can be combined by Boolean operators, such as (& for intersection, | for union and ~ for complement) to create volumes, which are recognized as cells by the code. Thus, an **`openmc.Cell`** subclass is a region bounded by half-space of quadric surfaces. In addition, a bounding-box can be determined automatically for regions bounded by half-spaces of cylinders, spheres and axis-aligned planes. Then, a collection of cells that may be repeated as part of the geometry can be comprised in subclass **`openmc.Universe`**. Generally, universes are used either to be assigned as a fill for a cell or in lattice formation, and they may also be translated and/ or rotated.

On the other hand, a nuclear reactor usually has repeated structures that occur in a regular pattern. An example for such a structure could be a lattice. In such cases, OpenMC provides a way to define lattice structures through the **`openmc.RectLattice`** and **`openmc.HexLattice`** classes.

In OpenMC, by default a surface is created by particles that pass through it, i.e. a surface is infinite unless and otherwise a boundary is applied to it. Thus, it is necessary to apply a boundary condition to specify a change of behavior for particles passing through the surface. There are three types of boundary conditions for a given particle in a specific surface. These are vacuum, periodic and reflective boundary conditions.

Lastly, the highest level in the hierarchy of the geometry functionality is the root universe. The root universe is the one that is used to create the XML file of the geometry. (Romano, 2018)

3.2.3 Geometry plotting

There are two options of plotting geometry in OpenMC. The first one is a two-dimensional slice plot, which allows the user to view the created geometry along a cut plane. The colors of the plots can be assigned either by material type or cell identity. In addition, there is an option to selectively include or exclude regions. The **`plots.xml`** file, which contains the preferences of the user, relays the information to the machine to create the plot. The created plot is written to a **`.ppm`** file, which is viewable in any Linux format. The file can also be converted to other graphic formats easily.

The instance of plotting starts with an **`openmc.Plot`** command. Followed by **`plot.basis`** which determines the orientation of the plot, and the **`plot.origin`** command specifies the origin of the plot. The width and pixels of the plot can be specified as **`Plot.width`** and **`Plot.pixels`** consultatively. Finally, the plots will be generated by executing the **`openmc.plot_geometry()`** function. Figure 7 shows the API command to create a plot in OpenMC.

```
plot_file = openmc.Plots(plot_xy)
plot_xy = openmc.Plot()
plot_xy.filename = 'plot_xy'
plot_xy.origin = [0, 0, 0]
plot_xy.width = [200, 200]
plot_xy.pixels = [3500, 3500]
plot_xy.color_by = 'material'

plot_file.export_to_xml()
```

Figure 7: Python API for geometry plotting.

Moreover, OpenMC has a capability for three-dimensional visualization by using graphic viewers known as ParaView and VisIt. The images are created the same way as the 2D images with the user specifying a grid of voxels. After the voxels are produced, the Phyton code imbedded in OpenMC converts the file into SILO or VTK files so that the image can be visualized by 3D graphic viewer software.

3.2.4 Tallies

In OpenMC the tally system has maximum flexibility in specifying physical results while maintaining scalability. By definition, a tally in MC reactor physics codes is a combination of filters and scores. Mathematically, it can be represented as follows.

$$X = \underbrace{\int dr \int d\Omega \int dE}_{\text{filter}} \underbrace{f(r,\Omega,E)}_{\text{Score}} \varphi(r,\Omega,E) \tag{4.1}$$

The filters determine which event in a phase space should be registered. On the other hand, a score identifies the physical quantity that is to be registered based on the chosen filters. OpenMC has given the user a freedom to choose from a wide selection of filters and scores, which are relevant to neutron tracking. Also, the user can choose to tally the scores using analog estimator, track length estimator or collision estimator. Analog estimator counts the number of actual reactions and determines the reaction rate based on the count. Whereas, a collision estimator registers the tally at every collision even if there is no reaction. On the other hand, track length estimator follows each particle and scores its contribution regardless of collision or reaction. Moreover, it is also the default estimator, unless the tallies require post collision attributes, in which case it would be a collision estimator. A list of available filters and scores, and a detailed mathematical explanation of the estimators can be seen from the user manual.

```
import openmc

# Open OpenMC's HDF5 statepoint file for the 100th batch
sp = openmc.StatePoint("statepoint.100.h5")

# Extract Tally objects for the flux and all reaction rates
rxn_rates = sp.get_tally(name = "reaction rates")
fluxes = sp.get_tally(name = "fluxes")

# Slice a Tally with only the "total" reaction rates
total = rxn_rates.get_slice(scores = ["total"])

# Compute the total MGXS with tally arithmetic
total_mgxs = total/flux
```

Figure 8: MGXS Calculation with Tally Arithmetic. (Boyd, et al., 2019)

It is possible to calculate cross sections of interest using the scattering and fission reaction rate outputs that are filtered by energy dependency. The multi-group cross section can also be generated directly by the code by using a coarse mesh finite difference (CMFD) solver (see Figure 8). The generated multi-group cross sections can be used as an input for deterministic codes.

Learning from the draw backs of other codes that suffered from severe performance issues when tallying a large number of quantities, OpenMC is equipped with a mapping technique that determines a tally to bin combination needed for a given phase space coordinates. The technique works in such a way that for each filter variable there exist a list of tallies to bin combination which can be scored for each value of the aforementioned filter.

3.2.5 Eigenvalue

The eigenvalue in the transport equation is defined as the multiplication rate of neutrons in a reactor. It can also be expressed statistically as the ratio of the population of current generation's neutrons to the population of their predecessors. Evaluating the eigenvalue statistically requires tracking the random walk of a large number of neutrons through many generations. Previously, this way of estimating the multiplication rate was impossible due to computational limitations, but recent advancements in computing performance made it possible to use Monte Carlo codes for such type of calculations.

Most of these codes, OpenMC included, determine the neutron random walk by sampling the appropriate probability density functions. That is, in cases where fission is observed, the spatial coordinates of the fission site, the sampled outgoing energy and direction of the fission neutron and the weight of the neutron (the probability constant of interaction) are

tallied. In OpenMC the fission site information is stored in an array field called fission bank. And the sampled source sites are stored in source bank array. (Romano, et al., 2014)

On the other hand, since the fission source for the first batch of neutrons cannot be known in priori, the first simulation will begin with an arbitrary source distribution. Then on the forthcoming simulations the source site is selected based on the recorded fission sites from the previous simulation. This iterative process will continue for a predefined number of iterations. To eliminate the effect of the arbitrary source, the simulation of the first few batches will be discarded. And to maintain the starting number of neutrons, the fission production is normalized after each simulation.

Lastly, OpenMC uses a global tally concept for the effective multiplication factor estimation. These estimators are divided into three, known as: analog, collision and track-length estimators (for detailed explanation on the estimator types see Section 5). (Romano, et al., 2012)

3.2.6 MGXS module

After the model geometry and material are defined, the MGXS generation workflow commence by creating MGXS subclasses. The MGXS subclass is a part of the python API that computes macroscopic cross section in group bins from tallies. It also has a library for different groups, spatial domains and reaction types (Boyd, et al., 2019). The generated multi-group constants can be applied in fine-mesh heterogeneous deterministic neutron transport codes.

Multi-group cross sections are calculated for spatially discretized regions in the geometry. A region could be as inclusive as a fuel assembly, just a fuel pin or a constructive solid geometry. This integration of spatial zones over discrete regions is known as spatial homogenization. (Romano, 2018)

On the other hand, critical systems usually have continuous energy domains ranging from 10^{-5} eV to 10^{7} eV. The multi-group approximation divides this range into a number of energy groups. The integration of neutron energies in a discrete group is known as energy condensation. (Romano, 2018)

In OpenMC there are two ways to instant MGXS objects. The first one is a manual instantiation of the subclasses. This option gives the user an ability to specify the spatial interest of domain, an

energy-group and the type of nuclides for which the multi-group cross section can be generated through the **`MGXS.by_nuclide`** attribute (see Figure 9). Alternatively, the second method is an automated instantiation by using a data library. As it is shown in

```
import openmc

# Create geometry for the model
fuel = openmc.Cell(...)
clad = openmc.Cell(...)
moderator = openmc.Cell(...)

...

# Specify energy group structure
groups = openmc.mgxs.EnergyGroups([0.0, 1.0, 20.0e6])

# Create MGXS library
library = openmc.mgxs.Library(geometry)
library.energy_groups = groups

# Specify desired reactions and spatial domains
library.mgxs_types = ['transport', 'fission', 'total', 'scatter matrix']
library.domain_type = 'cell'
library.domains = [fuel, clad, moderator]

# Add tallies to collection and export to XML
tallies = openmc.Tallies()
library.add_to_tallies_file(tallies)
tallies.export_to_xml()

# Run OpenMC and read in results of simulation
openmc.run()
sp = openmc.StatePoint('statepoint.500.h5')
library.load_from_statepoint(sp)
```

Figure 10, this instantiation permits the user to compute multiple cross sections for multiple spatial domains. (Boyd, et al., 2019)

The resulting tallies after the simulation are written in HDF5 state point file. Then, the Python API, i.e. the state point object read the tallies and load them via **`MGXS.load_from_statepoint()`** attribute to the MGXS object. After MGXS data is created, it can be displayed as an output, saved to a file or converted to Pandas Data Frame. (Boyd, et al., 2019)

```
import openmc

# Create geometry for the model
fuel = openmc.Cell(...)
clad = openmc.Cell(...)
moderator = openmc.Cell(...)

...

# Specify energy group structure
groups = openmc.mgxs.EnergyGroups([0.0, 1.0, 20.0e6])

# Create MGXS object for the fuel only
total_xs = openmc.mgxs.TotalXS(domain = fuel, energy_groups = groups)

# Add tallies to collection and export to XML
tallies = openmc.Tallies()
tallies + = total_xs.tallies.values()
tallies.export_to_xml()

# Run OpenMC and read in results of simulation
openmc.run()
sp = openmc.StatePoint('statepoint.500.h5')
total_xs.load_from_statepoint(sp)
```

Figure 9: MGXS Calculation with manual object instantiation. (Boyd, et al., 2019)

```
import openmc

# Create geometry for the model
fuel = openmc.Cell(...)
clad = openmc.Cell(...)
moderator = openmc.Cell(...)

...

# Specify energy group structure
groups = openmc.mgxs.EnergyGroups([0.0, 1.0, 20.0e6])

# Create MGXS library
library = openmc.mgxs.Library(geometry)
library.energy_groups = groups

# Specify desired reactions and spatial domains
library.mgxs_types = ['transport', 'fission', 'total', 'scatter matrix']
library.domain_type = 'cell'
library.domains = [fuel, clad, moderator]

# Add tallies to collection and export to XML
tallies = openmc.Tallies()
library.add_to_tallies_file(tallies)
tallies.export_to_xml()

# Run OpenMC and read in results of simulation
openmc.run()
sp = openmc.StatePoint('statepoint.500.h5')
library.load_from_statepoint(sp)
```

Figure 10: MGXS Calculation with Library-Automated Object Instantiation. (Boyd, et al., 2019)

3.2.7 MGMC simulation

One of the features that makes OpenMC unique is its capability to perform multi-group Monte Carlo (MGMC) simulation. This mode can work with either isotropic or anisotropic flux weighted homogenized multi-group macroscopic cross sections. Moreover, the

scattering matrix can be introduced using Legendre polynomials, histogram, tabular angular distributions or as transport corrected isotropic matrix. Then, the special geometry upon which the simulation will be executed has to be specified. Currently, OpenMC supports material, cell, universe and mesh domain types. (Romano, 2018)

Comparing the fission rate of MGMC simulation with the heterogenous Monte Carlo model is a good way to verify the accuracy of the homogenized and energy condensed macroscopic cross sections (Romano, 2018). With this intent in mind and to also compare it with a deterministic model that uses the same multi-group constants generated by OpenMC, a MGMC simulation was executed using the reduced model (see Section 6.2).

3.2.8 Simulation output

The code is capable of delivering the simulation results, such as k_{eff} and tally outputs as both ASCII file and Hierarchical Data Format (HDF5), which is a binary file. This makes viewing outputs with HDFView or PyTables (a Python package that can manipulate HDF5 data) easier. From the programming point of view, writing the results to a disk is more efficient with HDF5. In addition, it can also perform parallel I/O since the API has standard calls for this purpose. (Romano, et al., 2012)

Even though, the number and format of the output files depend on the user's preference, the most common ones are:

- **`tallies.out` –** Is an ASCII formatted file containing the mean value and standard deviation of a user defined tallies.
- **`summary.h5`** – Is an HDF5 file containing the description of the geometry and materials.
- **`statepoint.#.h5`** – Is also an HDF5 file containing the results of the simulation. This is the file that is used in all post process calculations in this thesis.

3.2.9 Nuclear data processing

Nuclear data is the base of all nuclear calculations. This data can be produced either by experimental setup or computational models. Obviously, the most tangible and trusted option would be the one that is found experimentally and evaluated by nuclear model calculations. Usually nuclear applications of research or industrial nature access an evaluated nuclear data

from ENDF (Evaluated Nuclear Data file) library. The raw data should then be processed by cross section processing codes, such as NJOY (Hebert, 2016).

The ENDF system is developed by Cross Section Evaluation Working Group (CSEWG) with sufficient accuracy to define cross sections over a large energy domain. The format of the library and the type of data included for a particular encounter is decided by this Working Group. In the ENDF library the file system is organized into two types. The first type, ENDF/A contains an arbitrary number of data for each isotope. Whereas, the ENDF/B file is comprised of a single evaluated and recommended data per interaction type. (Trkov, et al., 2018)

Figure 11 shows the steps of data library creation in the case of experimental data. In the data library creation process, Monte Carlo codes, such as OpenMC are employed as lattice calculators to produce the reactor database.

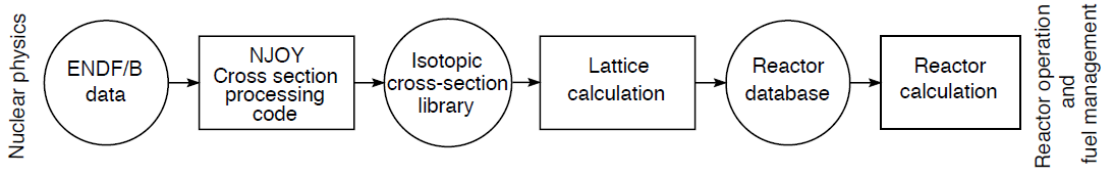


Figure 11: Data library creation steps. (Hebert, 2016)

OpenMC uses the ACE data format to represent neutron interaction with nuclei (Romano, et al., 2014). There are two types of ACE formatted data, known as Type 1 and Type 2. The basic difference between these two tables is that, the first one is formatted and independent, whereas the second one is unformatted and machine dependent but more compact and easier to read. Each type contains numerous classes of data (Conlin, et al., n.d.).

The ACE file is generated using the NJOY nuclear data processing system. NJOY is a Los Alamos National Laboratory product, which can generate applicable point wise libraries from ENDF files. The code is comprised of a set of modules with a well-defined processing task. Each module may be linked with another to prepare libraries of various nuclear applications. (MacFarlane, et al., 2010)

Depending on secondary energy and angle distribution laws of the ACE format data, OpenMC can simulate all nuclear reactions producing secondary neutrons, fission and scattering. Besides, using the same cross section libraries as other Monte Carlo codes, such as MCNP, allows OpenMC to compare simulation results. (Romano, et al., 2014)

Moreover, it has some special features to properly treat some peculiar physical situations. For instance, neutron scattering kinematics with a vibrating nucleus is approached by free gas approximation. Also, the probability table method is used to account for self-shielding in the resonance region. In addition, the eigenvalue problems are solved by a method of successive generations. (Romano, et al., 2014)

Furthermore, OpenMC keep scores of collision, absorption and track-length estimators, then calculates a minimum variance combined estimator based on the covariance matrix. It also gives a possibility for the user to define a mesh over which the Shannon entropy could be calculated. (Romano, 2018)

Shannon entropy is a mathematical quantity, which can be used to assess the convergence of the fission source distribution. In Monte Carlo simulations the first few calculation batches will be discarded to eliminate the bias of the initial guess (see Section 3.2.5), and it is ambiguous to determine at which point to start scoring the tallies. To this end, Shannon entropy of the fission distribution is proved to be effective in determining the convergence of the fission distribution, since it converges to a single steady value as the source becomes free of the initial guess bias. (Brown, 2000)

Computation of the Shannon entropy requires tallying of the fission sites in a fissionable domain. Then, the scored fission sites can be estimated as the fission source for the second batch of calculations. When a source distribution is estimated in such a way, the effect of the initial source site bias will diminish after a sufficient number of batches of calculations, and this is implied by a constant Shannon entropy value. The fission source sites can be tallied by imposing a mesh grid over the fissionable region. Mathematically the Shannon entropy can be expressed as follows: (Brown, 2000)

$$H_{src} = -\sum_{J=1}^{N} P_J \cdot ln_2(P_J) \tag{3.1}$$

Where,

H_{src} is the Shannon entropy.

P_J is the number of source sites in J^{th} grid. And, N is the number of mesh grids.

3.2.10 Parallel computing

Since Monte Carlo high fidelity particle transport simulations are slow to converge, various coding techniques have been developed to accelerate the process. One of these techniques is parallel computing. Parallel computing is based on the idea that each particle simulation is independent; therefore, Monte Carlo method is inherently parallel. According to (Wu, 2019), there are three types of parallelization methods, namely parallel on particles, region decomposition and data decomposition.

Of the above three, the most usual parallelization method is parallel on particles. This method has two options for parallelization known as parallel computing for fixed-source problems and parallel computing for eigenvalue problems. The idea behind the first of these options is to equally distribute particles to each processor for every independent simulation, and the final result can be obtained by merging the results from each simulation. On the other hand, parallelization for eigenvalue problem requires the fission sources to be sampled and stored in fission banks post each-iteration. Then, the resulting fission sources will be redistributed for the next generation calculation using the message passing interface (MPI). (Wu, 2019) and (Romano, et al., 2014)

The third and fourth techniques are more useful in terms of pre-allocating or sharing a memory within a node, especially for high-fidelity Monte Carlo calculations. In the case of region decomposition, the model will be discretized into smaller regions and assigned to specific memory locations. For particles crossing regional boundaries, their information will be passed to the memory location of their current residence by shared memory parallelism. Whereas, in data decomposition algorithm, the tally bins will be distributed equally to the available processors and evaluated. (Wu, 2019) and (Romano, et al., 2014)

OpenMC is capable of using both distributed memory and shared memory parallel computing. Shared memory is useful when the simulation is carried out on a single node with multiple processors, since each processor can simulate a particle independently. In OpenMC this feature is implemented through OpenMP. Thus, a system with Fortran compiler that supports OpenMP is required. (Romano, 2018)

Whereas, for simulations launched on a cluster or supercomputer, parallelizing the work throughout the nodes using MPI could save the calculation time drastically. Employing this

feature in OpenMC requires the implementation of OpenMPI or MPI CH. Therefore, one of these implementations should be installed in the system. (Romano, 2018)

4 GLOBAL HOMOGENIZATION PROCESS

Even though, the most accurate calculation method of the transport equation is the global fine calculation, it is unrealistic in industrial scale. Since a reactor core is composed of numerous types of material with various geometry, calculating the eigenvalue and other interaction rates of the heterogenous composition is prohibitive in financial and time sense. Therefore, homogenization of the special reaction system (reactor) is essential while preserving the global reaction rates and multiplication factor. This technique provides a group of constants that can be used to solve the full-core problem with much less computational effort.

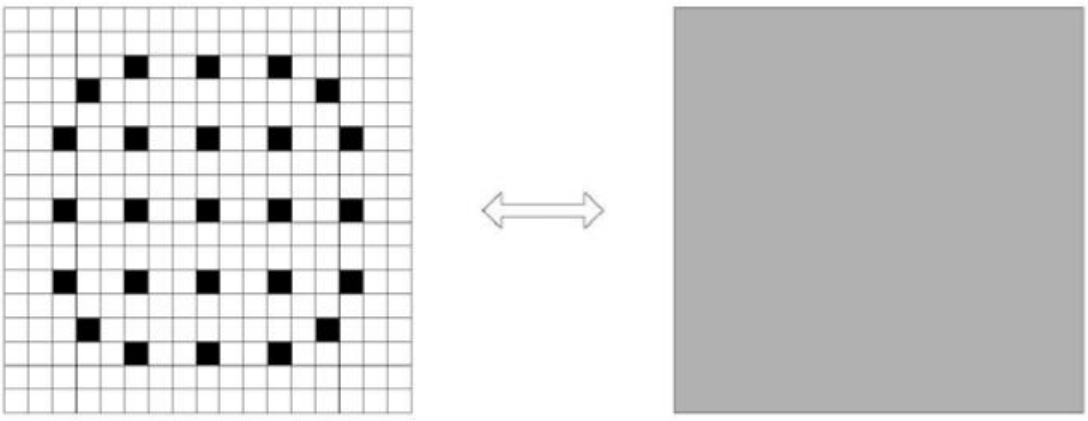


Figure 12: Demonstration of a homogenization at assembly level. (Hall, 2015)

4.1 The purpose of homogenization and energy condensation

On one hand the importance of accuracy in calculating neutronic parameters, and on the other the limitation of computational capacity demands the utilization of deterministic codes in case of core calculation. Thus, deterministic codes circumvent the computational challenge by coupling energy intensive calculations with few-group calculations via spatial homogenization and group condensation.

All the deterministic codes require the continuous energy dependence to be discretized into energy groups. Even if, the interaction data is discretized into numerous groups with other codes, it still is unfit to be implemented into reactor-scale calculations. Therefore, the discretized data should be reduced further with a homogenization process. In addition, the

nuclear interaction data collected from experimental measurements, also known as the nuclear data file, can only be used by Monte Carlo codes.

It a known fact that homogenized group constants are affected by fuel type and local operating conditions. Therefore, the details of such conditions should be taken into consideration in lattice calculations (assembly level flux distribution and multiplication factor computations) (Smith, 1986).

In addition, for the codes that employ a pin-by-pin or nodal diffusion method the cross-sections applied in these nodal simulation codes are group collapsed usually from two to four groups. Here, it should be noted that a straightforward energy collapse of cross sections will not preserve the properties of the lattice physics in diffusion theory models. The assembly transport calculation needs to be performed with sufficient number of energy groups to consider the spectral interactions between materials in the core. Also, the nonuniform fuel depletion affects the nodal averaged quantities and flux shape. Thus, it is important to develop an accurate method for homogenizing reactor assemblies, which takes into consideration the inter-assembly transport effects.

4.2 Global homogenization theory

A reliable input cross section data plays an indispensable role in an effort to mimic the neutronic interactions in a reactor using calculation codes. However, a truly heterogeneous transport equation constitutes unknowns in the order of 10^{11}. Under the current computational power of state-of-the-art computers solving such equation rigorously is not possible. In order to minimize the complexity and economize the time consumption the transport equation can be solved in two steps at reactor core level as the sub-assembly calculation and the core calculation.

The homogenized multi-group cross sections are produced by the first step. Although, homogenization has its merits, it is not without side effects. For instance, in standard assembly homogenization, it is impossible to maintain sub-regional quantities in a homogenized region.

The neutron transport equation in a heterogeneous reactor can be expressed as follows.

$$\nabla \cdot J_g(r) + \Sigma_{x,g}(r)\phi_g(r) = \sum_{g=1}^{G}\left[\frac{1}{k_{eff}M_{gg'}(r)} + \Sigma_{gg'}(r)\right]\phi_{g'}(r) \qquad (4.1)$$

Where,

$J_g(r) = \int d\Omega\Omega \cdot \varphi_g(r,\Omega)$, is the net current between faces.

$\phi_g(r) = \int d\Omega \cdot \varphi_g(r,\Omega)$, is the flux distribution before homogenization.

$M_{gg'}(r) = \chi v\Sigma_{fg}(r)$, is the multiplication of neutron fission spectrum, new neutrons per fission and fission cross section prior to homogenization.

$\Sigma_{gg'}(r) = \frac{1}{2}\int d\mu_o\Sigma_{gg'}(r,\mu_o)$; $\mu_o = \Omega \cdot \Omega'$, is neutron scattering cross section before homogenization.

r is a position vector.

Ω is a direction vector.

In the above equation energy continuity and smooth flux density distribution are maintained. However, as it has been stated in the previous section, core calculation simulators required a discretized form. Obviously, a substantial amount of information will be lost during the process. Nevertheless, current homogenization calculations resort at least to conserve the following three physical quantities.

- The node averaged group reaction rates,

$$\int_{V_k} \bar{\Sigma}_{x,g}\bar{\phi}(r)\,dV = \int_{V_k} \Sigma_{x,g}\phi_g(r)dV \qquad (4.2)$$

- The interfacial group current,

$$\int_{S_i^k} \nabla \cdot \bar{J}_g(r) \cdot dS = \int_{S_i^k} \nabla \cdot J_g(r) \cdot dS \qquad (4.3)$$

Where,

$\bar{J}_g(r) = -\bar{D}_g(r)\nabla\,\bar{\phi}(r)$.

The *k* in V_k refers to a region in a volume integral for spatial homogenization.

And S_i^k is the k^{th} surface of homogenized region i.

Therefore, Eqn (4.3) can be rewritten as:

$$\int_{S_i^k} \bar{D}_g(r)\nabla\,\bar{\phi}(r)\cdot dS = \int_{S_i^k} \nabla\cdot J_g(r)\cdot dS \tag{4.4}$$

In all the equations, *x* denotes the interaction type and *g* is an energy group index.

- And thirdly, the eigenvalue of the reactor.

$$\sum_{k=1}^{K}\int_{S_i^k} \bar{D}_g(r)\nabla\,\bar{\phi}(r)\cdot dS + \int_{V_k} \bar{\Sigma}_{x,g}\bar{\phi}(r)\,\mathrm{d}V =$$

$$\sum_{g=1}^{G}\int_{V_k}\left[\frac{1}{k_{eff}\bar{M}_{gg'}(r)} + \bar{\Sigma}_{gg'}(r)\right]\bar{\phi}_{g'}(r)\,dV \tag{4.5}$$

Where,

$\bar{\Sigma}_{x,g}$ is homogenized interaction cross section.

$\bar{\phi}_g$ is flux distribution after homogenization.

$\bar{D}_g$ is the diffusion coefficient.

In OpenMC the concept of "equivalence" is preserved intrinsically through global tallying. Assuming that all of the homogenized parameters are spatially constant within a node, the ideal homogenized cross section can be calculated from the conservation of reaction rate as follows.

$$\bar{\Sigma}^k{}_{x,g} = \frac{\int_{V_k}\Sigma_{x,g}\phi_g(r)dV}{\int_{V_k}\bar{\phi}(r)\,dV} \tag{4.6}$$

Literatures indicated that the homogenized cross section generated from the above equation may not guarantee the conservation of the integral reaction rates especially in sub-regional interfaces in homogenized regions. Also, the continuity of the current between interfaces will be affected. (Wang & Pan, 2019)

4.3 Particle angular distribution modeling

The motion of a particle in three-dimensional domain is represented by its solid angle normal to the direction of the particle, as shown in Figure 13 (Hebert, 2016).

$$\boldsymbol{V}_n = V_n \Omega \tag{4.7}$$

Where,

$$V_n = |\boldsymbol{V}_n| \; and \; |\Omega| = 1$$

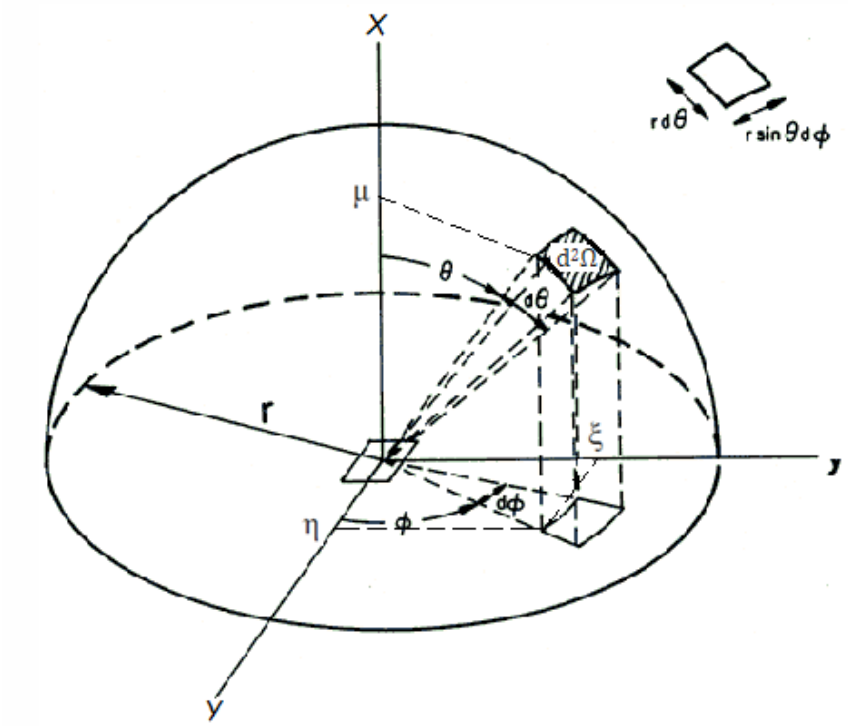


Figure 13: Solid angle. (Howell, 2011)

A solid angle is a space included inside a conical surface. It can be defined in terms of its three direction cosines as follows.

$$\Omega = \mu_i + \eta_j + \xi_k \tag{4.8}$$

Where,

$$\mu^2 + \eta^2 + \xi^2 = 1$$

Here, the domain for the zenith is $0 \leq \theta \leq \pi$ and $0 \leq \phi \leq 2\pi$ for the azimuth. Also, the direction cosines can be written in terms of the angles as follows (Hebert, 2016).

$$\mu = cos\theta, \eta = \sqrt{1-\mu^2 cos\phi} \ \ and \ \xi = \sqrt{1-\mu^2 sin\phi} \tag{4.9}$$

Thus, the solid angle is the quotient of the infinitesimal area swept-out on the surface of the unit sphere and the square of the radius, and can be expressed as:

$$d^2\Omega = \frac{rd\theta r sin\theta d\phi}{r^2} \tag{4.10}$$

This can be reduced to:

$$d^2\Omega = sin\theta d\theta d\phi \tag{4.11}$$

Particle transport quantities are continuous, and their distribution direction is defined by either the cosine μ or the solid angle Ω. In the case of the earlier the function can be approximated by L-order Legendre polynomial expansion. (Hebert, 2016)

$$f(\mu) = \sum_{\ell=0}^{L} A_\ell P_\ell(\mu) \tag{4.12}$$

And, the L^{th} order coefficient can be calculated as:

$$A_\ell = \frac{2+1}{2}\int_{-1}^{1} f(\mu) P_\ell d\mu \tag{4.13}$$

The Legendre polynomial $P_\ell(\mu)$ can be expressed mathematically as:

$$P_\ell(\mu) = \sum_{n=0}^{N} (-1)^n \frac{(2\ell-2n)!}{2^\ell n!\,(\ell-n)!\,(\ell-2n)!} x^{\ell-2n} \tag{4.14}$$

Where,

$$N = \begin{cases} \dfrac{\ell}{2} \ for \ even \ \ell \\ \dfrac{\ell-1}{2} \ for \ odd \ \ell \end{cases}$$

The solid angle can also be approximated in a similar fashion by L-order real spherical harmonics expansion (Hebert, 2016).

4.4 MGXS generation with OpenMC

The new capability introduced in OpenMC transport code for MGXS generation aims to mitigate the aforementioned draw backs of homogenized cross section generation (see

Section 4.2). In addition, this method replaces a separate calculation step for resonance self-shielding carried out by deterministic physics codes. Hence, it is possible to compute improved few-group constants needed by fine-mesh multi-group transport simulators (Boyd, et al., 2019). In this regard Serpent code is the pioneer to generate MGXS directly as a simulation output. Earlier trend was to use MCNP and other MC codes, and post process the resulting reaction rate estimates.

OpenMC being a statistical code uses stochastic integration, tallies and arithmetic combinations of tallies reaction rates to generate MGXS. The following equation shows a tally estimator of a reaction rate x.

$$\langle \Sigma_x, \varphi \rangle = \iiint_{VSE} \Sigma_x(r, E)\varphi(r, E, \Omega) dE d\Omega dr \tag{4.15}$$

Where,

$\langle , \rangle$ represents inner products in phase-space.

Σ_x is macroscopic cross section.
φ is angular neutron flux.
V is integration bound in space r.
S is solid angle Ω.
E is energy.

From Eqn (4.15) it should be noted that OpenMC uses angular flux with varying energy and space to collapse MGXS instead of scalar flux. Although, for this thesis the entire angular domain is integrated during MGXS generation since it is preferred by most multigroup transport codes, OpenMC is capable of producing angular-dependent MGXS.

Since cross-section generation is based on neutron-nucleus interaction, it is worth to discuss in brief about one of a fairly complicated interaction, i.e. collision interaction and its dynamics. A neutron-nucleus collision in its simplest form scatters the neutron without it penetrating the nucleus. This type of collision is known as potential scattering reaction. Potential scattering reaction is elastic since it conserves both momentum and kinetic energy of the pair.

In most cases collisions are more complex than potential scattering. Often, neutron penetration into a nucleus creates a compound nucleus, which is unstable due to the transfer

of binding energy. There are a number of ways by which a nucleus could release its excitation energy. The mode of decay determines the type of reaction. For heavy nucleoids for instance, the most probable decay mode is fission. Hence, it is called fission reaction. In the same manner there is resonant scattering reaction, radiative capture reaction and so on.

In nuclear physics calculations, there are two parameters that determine the probable reaction type. The first group of parameters is macroscopic cross-sections; these are used to compute the reaction rate. Secondly, the velocity and direction of remitted particles as a function of nuclide temperature and characteristics of the incident neutron (Hebert, 2016). The following section discusses the concept of modeling neutron distribution and direction in a global configuration.

4.4.1 Transport corrected scattering cross section

Most transport codes including OpenMC do not have the capacity to treat the angular dependent total cross section. Therefore, the anisotropy of scattering is generally treated by simplified approximations, i.e. by using transport corrected cross section and isotropic scattering approximations. Obviously, this kind of simplification will have an impact on the multi-group anisotropic scattering matrix. (Yamamoto, et al., 2008)

Nevertheless, the transport corrected calculation is compulsory due to memory limitations. To this end, the moment of a scattering kernel with ℓth Legendre polynomial is represented as follows. (Boyd, et al., 2019)

$$\Sigma_{s\ell}(r, E' \to E) = \int_{-1}^{1} \Sigma_s(r, \mu, E' \to E) P_\ell(\mu) d\mu \tag{4.16}$$

Based on Eqn (4.16), a tally for spatially homogenized and energy-collapsed ℓth Legendre scattering moment can be defined as:

$$\langle \Sigma_{s\ell}, \varphi \rangle_{k,g' \to g} =$$

$$\int_{r \in V_k} \int_{4\pi} \int_{E_g}^{E_{g-1}} \int_{E_{g'}}^{E_{g'-1}} \Sigma_{s\ell}(r, E' \to E) \times \varphi(r, E, \Omega) dE' dE d\Omega dr \tag{4.17}$$

Anisotropic scattering effects have been taken to consideration to minimize the discrepancies of implementing isotropic flux, which were mentioned at the end of the pervious section. This requires a transport correction to be incorporated into the general transport equation

with isotropic scattering kernel. In OpenMC this is done by introducing an expression for the in-scatter approximation to the transport correction. In-scatter is a phenomenon by which, a neutron emerged in a phase space as a result of scattering. The in-scatter approximation is computed by summing the first Legendre scattering moment over all incident energy groups (Boyd, et al., 2019):

$$\Delta\bar{\Sigma}_{tr,k,g} = \sum_{g'=1}^{G} \langle \Sigma_{s1}, \varphi \rangle_{k,g' \to g} \tag{4.18}$$

Where,

$\Delta\bar{\Sigma}_{tr,k,g}$ is homogenized transport correction.

Σ_{s1} is first Legendre scattering moment.

Here, it is noted that not all researchers in the field agreed with Boyd's approach. For instance, Yamamoto on his paper (Yamamoto, et al., 2008) showed that it is important to consider higher order Legendre scattering moment in some cases. For instance, for MOX fueled reactors and small cores with high leakage, it is suggested that first order anisotropic scattering is not enough. (Yamamoto, et al., 2008)

In defense of the current transport code (OpenMC), even though in principle detailed anisotropic transportation is ideal in this regard, such detailed calculation or even implementing higher order anisotropic scattering kernel would be prohibitive from the point of view of computational time. Hence, the potential discrepancy due to the above simplification should be included in the error margin of the calculation code.

To resume with the calculation of the transport correction, it is subtracted from the total cross section and normalized by the flux to compute the transport-corrected total cross section (Boyd, et al., 2019):

$$\bar{\Sigma}_{tr,k,g} = \frac{\langle \Sigma_t, \varphi \rangle_{k,g} - \Delta\bar{\Sigma}_{tr,k,g}}{\langle \varphi \rangle_{k,g}} \tag{4.19}$$

Where,

$\bar{\Sigma}_{tr,k,g}$ is transport corrected cross section.

Σ_t is total macroscopic cross section.

Since the homogenized Legendre scattering integral in Eqn (4.19) includes the outgoing neutron energy, an analog tally estimator is used to compute the transport correction. The total collision and flux are computed in the same manner to maintain consistency.

4.4.2 Scattering matrix

The cumbersome anisotropic scattering matrix computation can be reduced to its isotropic counterpart by transport corrected cross section approximation. The isotropic scattering matrix can be computed with an inner product of scattering reactions over both incoming and outgoing energies. Thus, the transport corrected scattering matrix for the ℓth Legendre moment can be define as (Yamamoto, et al., 2008):

$$\bar{\Sigma}_{s\ell,k,g'\to g} = \frac{\langle \Sigma_{s\ell}, \varphi \rangle_{k,g'\to g}}{\langle \varphi \rangle_{k,g'}} \tag{4.20}$$

One way of calculating the transport-corrected isotropic scattering matrix approximation is by subtracting the transport correction function, Eqn (4.20), from the diagonal elements of the isotropic scattering matrix as follows (Yamamoto, et al., 2008).

$$\bar{\Sigma}_{s,k,g'\to g} = \frac{\langle \Sigma_{s0}, \varphi \rangle_{k,g'\to g} - \delta_{g,g'}\Delta\bar{\Sigma}_{tr,k,g}}{\langle \varphi \rangle_{k,g'}} \tag{4.21}$$

Notice that the transport correction from Eqn (4.20) is calculated using the in-scatter approximation. However, other researches show that transport codes are usually numerically unstable when the in-scatter of the ℓth Legendre flux moment is used to evaluate the diagonal element of an anisotropic scattering matrix. Therefore, for the sake of practicality, a total P0 flux is desirable as a weighting factor in such computations. (Yamamoto, et al., 2008)

Alternatively, the consistent scattering matrix can be calculated by multiplying the isotropic scattering cross section by a group-to-group scattering probabilities. However, this formulation is computed form the out-scattering cross section contrary to the previous resolution.

Even though, the out-scatter approximation of the transport cross section results in the under estimation of the k_{eff}, it is balanced with the over estimation of the transport correction of Eqn (4.20), which was calculated with the in-scatter approximation. This insures the preservation of the reaction rate balance by a total cross section. The out-scattering cross section, $\bar{\Sigma}_{s,k,g}$ can be tallied by track-length estimators. Then, the Legendre scattering

moment will be the product of the out-scattering cross section and scattering probability matrix $\bar{P}_{s\ell,k,g'\to g}$, as shown below.

$$\bar{\Sigma}_{s\ell,k,g'\to g} = \bar{\Sigma}_{s,k,g} \times \bar{P}_{s\ell,k,g'\to g} \tag{4.22}$$

Where,

$$\bar{P}_{s\ell,k,g'\to g} = \frac{\langle \Sigma_{s\ell}, \varphi \rangle_{k,g'\to g}}{\langle \Sigma_{s0}, \varphi \rangle_{k,g'}} \tag{4.23}$$

Then, the neutron multiplication due to the reactions can be included into the consistent scattering matrix formulation by multiplying Eqn (4.21) by the multiplicity matrix.

$$\bar{\Sigma}_{s\ell,k,g'\to g} = \upsilon_{k,g'\to g} \times \bar{\Sigma}_{s,k,g} \times \bar{P}_{s\ell,k,g'\to g} \tag{4.24}$$

The multiplicity matrix in-turn is calculated from the rate of scattering multiplicity reaction, which is given as:

$$\langle \upsilon\Sigma_s, \varphi \rangle_{k,g'\to g} =$$

$$\int_{r\in V_k} \int_{4\pi} \int_{E_g}^{E_{g-1}} \int_{E_{g'}}^{E_{g'-1}} \sum_j \upsilon_j \Sigma_j \times (r, E' \to E, \Omega) \times \varphi(r, E, \Omega) dE' dE d\Omega dr \tag{4.25}$$

Where,

υ_j is the scattering multiplicity for the j^{th} multiplicity reaction.

Finally, the scattering multiplicity matrix can be evaluated by dividing Eqn (4.24) by the isotropic scattering moment as follows.

$$\upsilon_{k,g'\to g} = \frac{\langle \upsilon\Sigma_s, \varphi \rangle_{k,g'\to g}}{\langle \Sigma_{s0}, \varphi \rangle_{k,g'\to g}} \tag{4.26}$$

4.4.3 Fission cross section

In nuclear physics calculations, nuclear fission is given an uttermost attention firstly because it is responsible for the continuation of the chain reaction. And secondly, because nuclear kinetics is complex by nature, and predicting the evolution in time of the neutron population in a multiplying medium is treacherous (Zohuri, 2019). There are a couple of approaches to compute this phenomenon. Current full-core statistic-based transport codes like OpenMC, use fission production rate estimators to calculate the fission production cross sections (Boyd, et al., 2019). These codes can calculate point kinetics parameters, such as, prompt

neutron lifetimes, effective delayed neutron fractions and precursor decay constants (Leppanen, et al., 2016). The general equation for fission production macroscopic cross section can be posed as follows.

$$\nu\bar{\Sigma}_{f,k,g} = \frac{\langle \nu\Sigma_f, \varphi \rangle_{k,g}}{\langle \varphi \rangle_{k,g}} \tag{4.27}$$

Eqn (4.26) is similar to the pervious reaction cross section equations with one difference, i.e. it is independent of the index g'. This is because when a compound nucleus decays with a half-life of 10-22 to 10-14 second, it has no memory of how it is formed (Hebert, 2016).

However, the index g' is implemented in the following equation in a different context. Eqn (4.27) below, shows the rate of first-generation neutrons, herein referred as g' undergoing a fission process to give birth to their descendants, denoted by index g.

$$\langle \nu\Sigma_f, \varphi \rangle_{k,g'\to g} =$$

$$\int_{r\in V_k} \int_{4\pi} \int_{E_g}^{E_{g-1}} \int_{E_{g'}}^{E_{g'-1}} \nu\Sigma_f(r, E' \to E) \times \varphi(r, E, \Omega) dE' dE d\Omega dr \tag{4.28}$$

In cases where the group-to-group fission production is needed, the above equation can be discretized by groupwise flux as follows.

$$\nu\bar{\Sigma}_{f,k,g'\to g} = \frac{\langle \nu\Sigma_f, \varphi \rangle_{k,g'\to g}}{\langle \varphi \rangle_{k,g}} \tag{4.29}$$

5 RESULT ESTIMATORS IN OPENMC

The discussions on the previous sections of this thesis were mainly focused on Monte Carlo simulation techniques and neutron interaction calculation procedures. However, collecting the results do also require as much attention as the simulation process.

In Monte Carlo simulation all interactions are based on discrete events. This is the main difference between stochastic codes and deterministic ones. This means, in Monte Carlo codes the reaction integral indicates the number of responses not the response rate (Leppänen, 2007). It is as such because the reaction rates are determined by counting the number of interactions in the phase space.

In OpenMC the recorded events, which are used to calculate group constants are known as scores. Such scores include absorption, scattering and fission reaction rates, and others like neutron flux estimations. The resulting scores then, can be evaluated in various ways to form statistical estimates of integral physical quantities. (Leppänen, 2007)

The conventional approach to obtain discretized values is to have the phase space variables divided into specified number of bins. Then the code estimates the integrated quantities inside each bin (Banerjee, 2010).

As in most stochastic codes, OpenMC implement standard deviation to determine the extent of the accuracy of the scores. However, a statistical test research done at VTT (a Finnish Technical Research Centre) shows that the confidence interval is highly dependent on adequate sampling and the number of computational cycles per batch. (Kaltiaisenaho, 2014)

5.1 Analog estimator

Analog estimator is the simplest method of collecting results. It is based on counting the nuclide-neutron interactions or simulated event sequences in region of interest (Leppänen, 2007). For instance, a reaction rate of a certain interaction can be estimated by scoring the number of sampled interaction types that can cause the required reaction type. Another example is the multiplication factor; it can be calculated using the estimates of consecutive batches (Romano, 2018). Mathematically, a volume integrated reaction rate is stated as:

$$R_x = \frac{1}{W} \sum_{i \in A} w_i \tag{5.1}$$

Where,

R_x is the reaction rate.
x is the type of reaction.
i is an index for each event.
W is the total starting number of particles.
w_i is pre-collision weight of particles.

Analog estimators are more practical to use, when the parameter to be calculated is derived from a complicated process that occurs during the simulation (Leppänen, 2007). In other cases, there are more efficient non-analog estimators available. Two of the most common Monte Carlo implicit flux estimators are explained in the following sections.

5.2 Collision estimator

A collision estimator, unlike the analog estimator, is based on counting collisions to estimate total interactions. Even though, a Monte Carlo simulation is an alternation of transports and collisions, if the transport kernel is normalized the absorption weighting can be adjusted by survival probability after each collision (Ragheb, 2013):

$$P_s = \frac{\Sigma_s}{\Sigma_t} \tag{5.2}$$

Where,

P_s is the probability of survival or non-absorption.

Then the weight of a particle prior to collision, w_i, will be (Ragheb, 2013):

$$w_i = w_{i-1}\frac{\Sigma_s}{\Sigma_t} \tag{5.3}$$

Thus, the flux estimate can be calculated by dividing the total macroscopic cross section over the pre-collision population, which is normalized by the starting weight of particles (Romano, 2018).

$$\phi = \frac{1}{W}\sum_{i\in C}\frac{w_i}{\Sigma_t(E_i)} \tag{5.4}$$

From Eqn (5.4), a reaction rate can be estimated by multiplying both sides with a macroscopic cross section of interest.

$$R_x = \frac{1}{W}\sum_{i\in C}\frac{w_i\Sigma_x(E_i)}{\Sigma_t(E_i)} \tag{5.5}$$

Where,

R_x is the reaction rate of type *x*.

Even though, counting collisions instead of individual reaction rates helps to reduce higher variances in low probability events, this method has a downside since it requires a homogenized phase space (Ragheb, 2013). This means that the transportation and collision kernels in the Neumann series must occur with uniform frequency in different volumes of interest.

5.3 Track length estimator

Track length estimator is conceptually different from the two estimators discussed above since it scores rather the neutron transport trajectory. In most MC simulations it is customary to use track length estimator for tallying volumetric flux. The reason is mainly because a lower variance estimate of the scalar flux can be obtained with this type of estimator. (Banerjee, 2010)

The volume integrated flux can be defined mathematically as follows (Romano, 2018):

$$\phi_v = \int dr \int d\Omega \int dE \int d\mathrm{t}\ \varphi(r,\Omega,E,t) \tag{5.6}$$

On the other hand,

$$\varphi(r,\Omega,E,t) = vn(r,\Omega,E,t) \tag{5.7}$$

Where,

n is the angular neutron density and *v* is velocity.

Thus, Eqn (5.6) can be rewritten as:

$$\phi_v = \int dr \int dE \int d\mathrm{tv} \int d\Omega\ n(r,\Omega,E,t) \tag{5.8}$$

And,

$$\int d\Omega\, n(r, \Omega, E, t) = N(r, E, t) \tag{5.9}$$

Also,

$$v\, dt = d\ell \tag{5.10}$$

Therefore, the differential unit of track length can be calculated by substituting the values in Eqn (5.9) and Eqn (5.10) into Eqn (5.8) as follows:

$$\phi_v = \int dr \int dE \int d\ell\, N(r, E, t) \tag{5.11}$$

The above equation indicates that the particle transport track can be employed as an estimator of the flux. And, the integral form can be discretized as follows:

$$\phi = \frac{1}{W} \sum_{i \in T} w_i \ell_i \tag{5.12}$$

Where,

T is a set of all trajectories in a particle's lifetime.

ℓ_i is the length of the i^{th} trajectory.

The reaction rate can then be calculated by multiplying both sides by the required macroscopic reaction cross section as in Eqn (5.5).

$$R_x = \frac{1}{W} \sum_{i \in T} w_i \ell_i \Sigma_x(E_i) \tag{5.13}$$

The use of track length estimator alienates the use of any filter that requires the knowledge of the state of the particle after collision. Thus, for tallies with energy or direction filters, for example scattering reaction, it is recommended to use analog estimator. (Romano, 2018)

6 MYRRHA CORE MODEL

Two detailed 3D computer models of the reactor core have been created using the OpenMC software (the steps followed, and the Python API commands applied to create the model are explained in Section 3.2). The first one is a model of MYRRHA Rev1.6 core design in its entirety, including the two outer BeO reflector rings (universe). This model is used to execute OpenMC code verification in reference to MCNP.

The second model is a radially and axially reduced model in space. The purpose for it is related to the cross-section homogenization process. A relatively compact model in space, will allow generating homogenized and energy-collapsed macroscopic cross-sections with low statistical variance. This would create confidence in the creation of a nodal database that would form the basis of the deterministic PHISICS input model.

6.1 Full MYRRHA Rev1.6 core model

This model (described from here-on for its critical mode of operation) is extended axially from z = -117.8 to z = +350, with its mid plane being at z = 0. Radially, the core model resembles an equilateral hexagon with a distance of 86 cm from the center to the longest end and bounded by a stainless-steel jacket. The core region comprises of 217 equilateral hexagons arranged in a concentric form. Out of these, 108 are hexagonal fuel assemblies with an axial height of 65 cm, and each of them containing 127 active rods. The pitch between the assemblies is 10.45 cm. At the periphery, the core is surrounded by two rings of reflector assembly, and each one contains seven rods made of BeO. Finally, the whole core geometry is divided into 19 axial nodes for the purpose of homogenization. In addition, the core model has six thermal IPSs for radio-isotope production and three fast IPSs for material testing. There are also three safety-rod assemblies in case of scram and six power shimmering assemblies included. A detailed explanation on how to create geometries of this model is given in Section 3.2.2 and Section 3.2.3 of this Thesis.

In the case of material setup, the density, nuclide proportions and temperature are defined based on the operation parameters of the reactor (see Table 4). At the beginning of cycle, BOC, there are 18 batches of 6 fuel assemblies each ranging from 0 to 60 MWd/kgMOX. However, the beginning of life starts with 78 fresh (single enrichment) fuel assemblies.

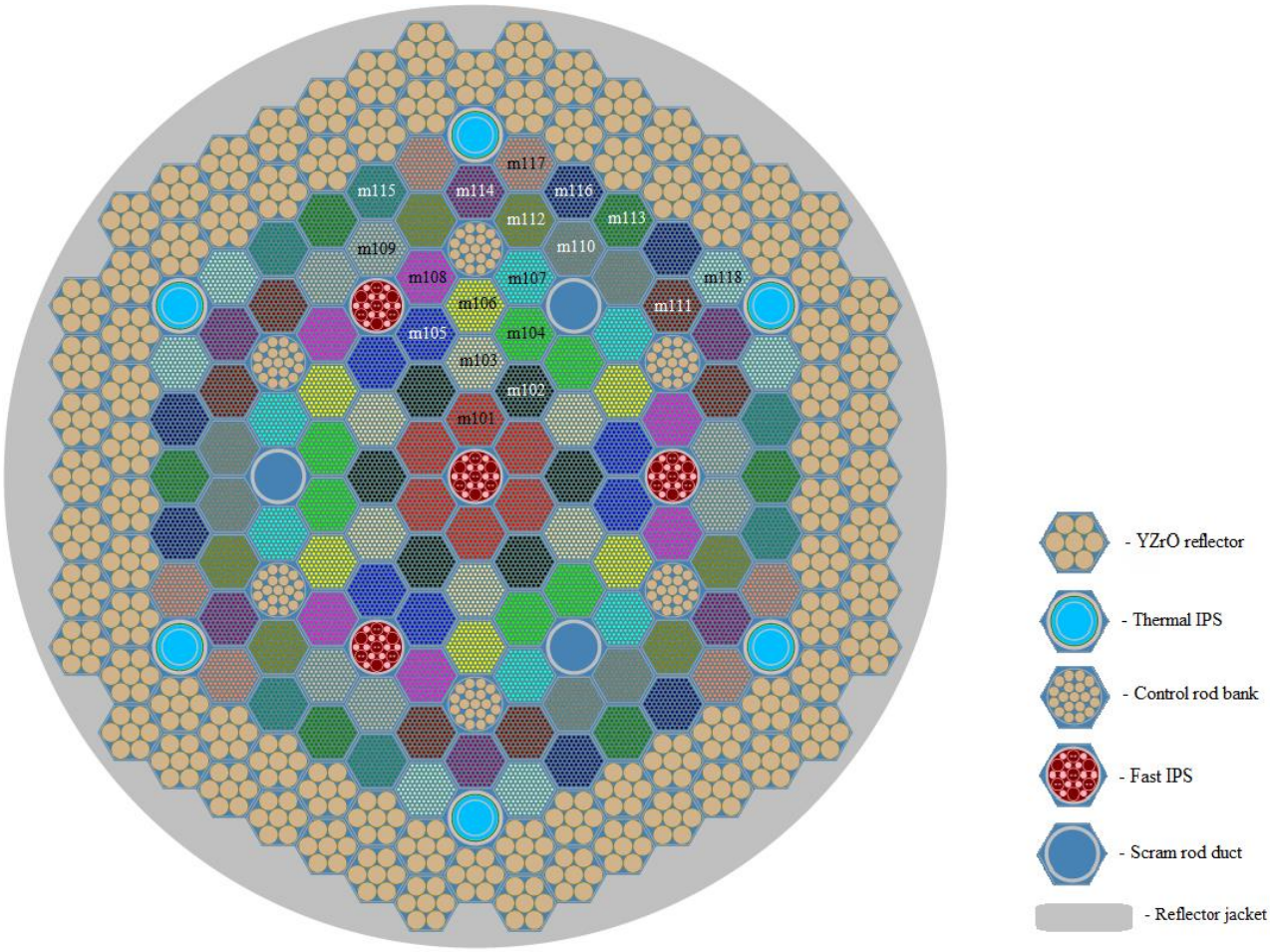


Figure 14: Radial cross-section at fuel mid-span of the MYRRHA 1.6 core model

Figure 14 shows the radial configuration of the reactor core, sectioned at the mid span of the fuel assembly region. In this figure the positions of control rod banks, safety rod banks, IPSs, reflector assemblies and all batches of fuel assemblies are portrayed and labeled.

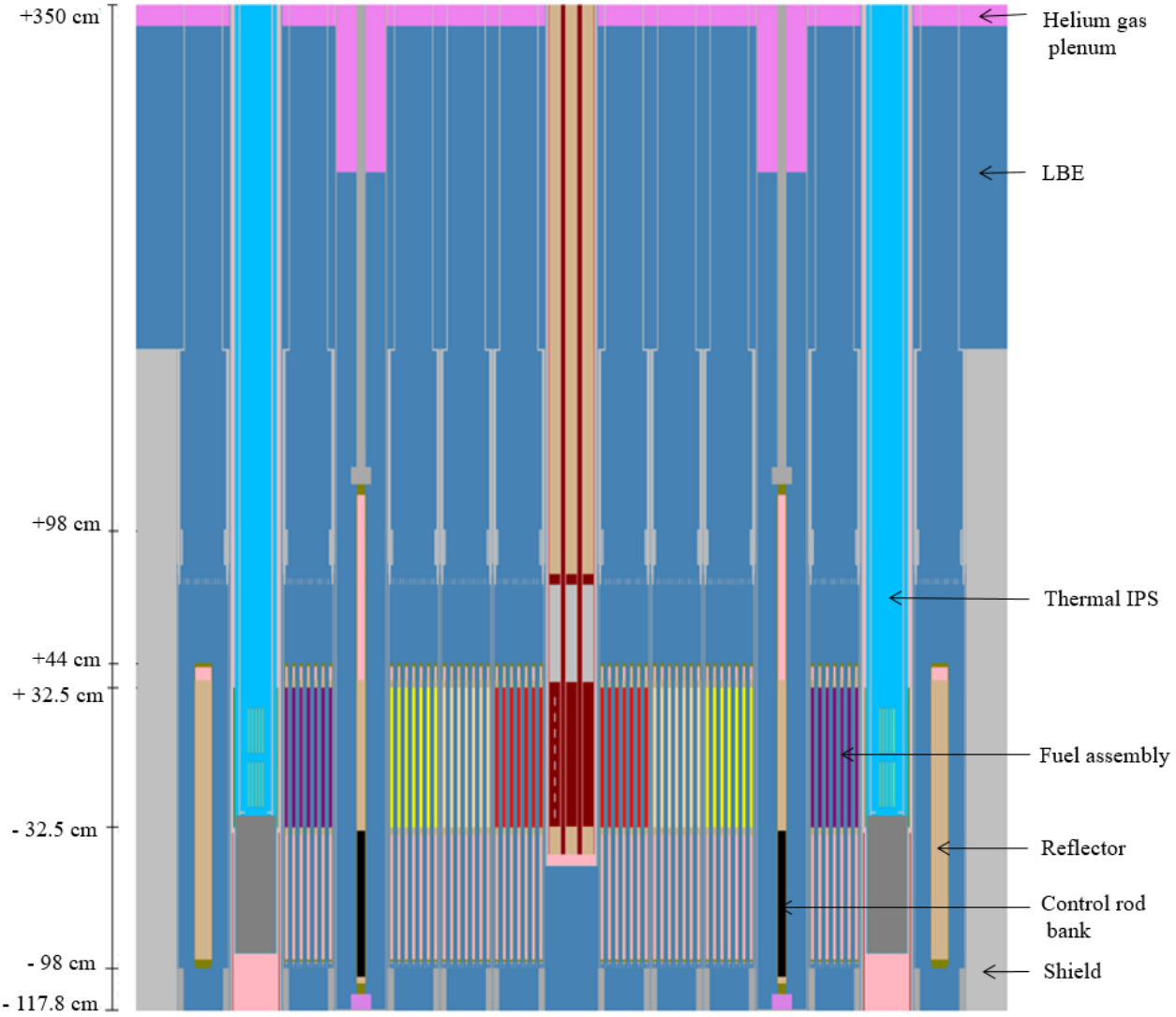


Figure 15: Axial mid-span cross-section view of the MYRRHA 1.6 core model in the yz-axis

In Figure 15 the yz-axis elevation cut view of the core model, sectioned at the origin is depicted. This figure contains some significant longitudinal values and components. The fuel assembly height, control rod height, reflector height and upper gas plenum are some of the components among the many.

In addition, Figure 16 below displays the elevation cut view, sectioned in xz-axis at the origin. In this figure, two fast IPS bins and the scram shaft with its bank of rods can be seen clearly.

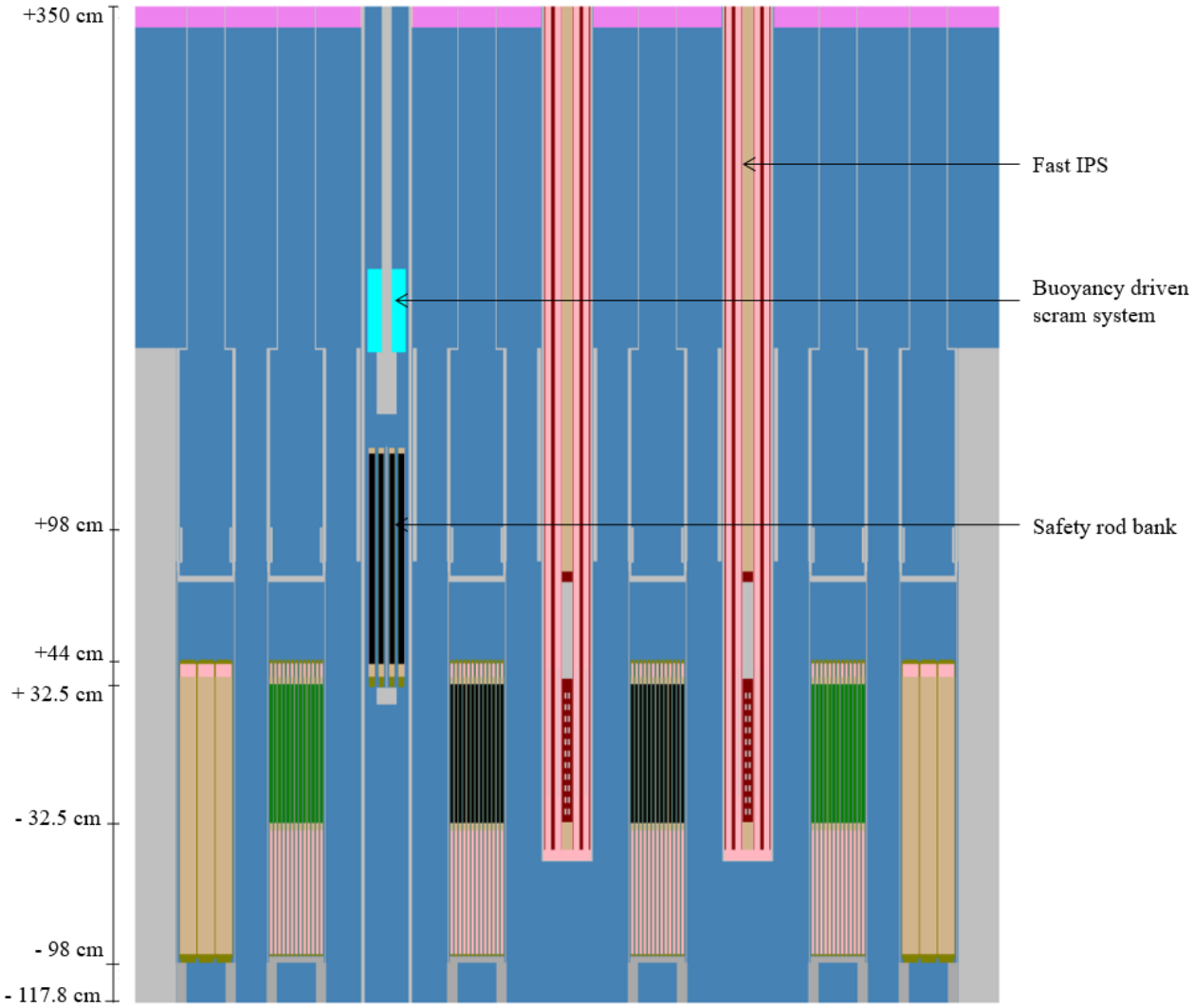


Figure 16: Axial mid-span cross-section view of the MYRRHA 1.6 core model in the xz-axis

6.2 Reduced model

As it can be seen from Figure 15, MYRHHA has a slim-cylindrical geometry with a height to diameter ratio of about 5.44:1 with fuel assemblies occupying the bottom third of the core height. In other words, the upper 2:3rd portion of the core is filled with coolant. This region is obviously neutron deficient and cannot be expected to give a reasonable statistical result in stochastic calculations. Therefore, it has become necessary to minimize the statistical errors while tallying volumetric-averaged reaction rates.

In contrast to the aforementioned full core model, this new model is reduced axially between elevations of + 44 cm and – 50.5 cm and a reflective boundary condition is imposed; while, radially black or vacuum boundary conditions still apply. There is still a enough portion of

LBE that axially extends for several neutron mean-free paths at the end of the fuel rods gas plenum (both at its upper and lower ends). Thus, assuming a reflective boundary conditions for this new reduced model is valid in order to preserve the shape of the flux at these heights with respect to the original model. In the end, the new 169 assembly-positions, divided into 19 axial nodes, will give a total of 3211 control volumes where multi-group constants are to be computed.

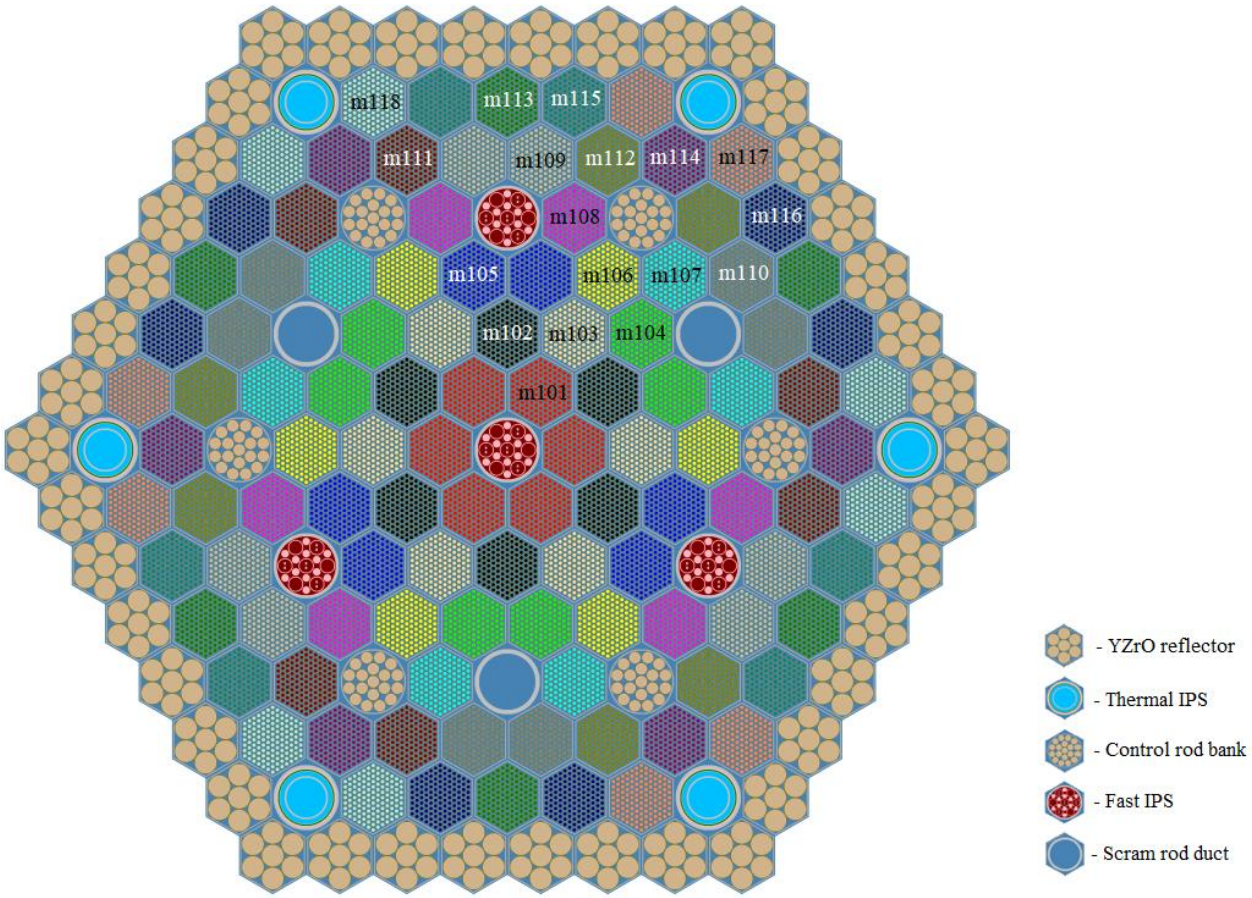


Figure 17: Reduced radial cross-section at fuel mid-span

Figure 17 shows the radial section view of the reduced model with a cut plane at the origin. From the figure it can be noticed that the core model is comprised of eight concentric hexagonal rings.

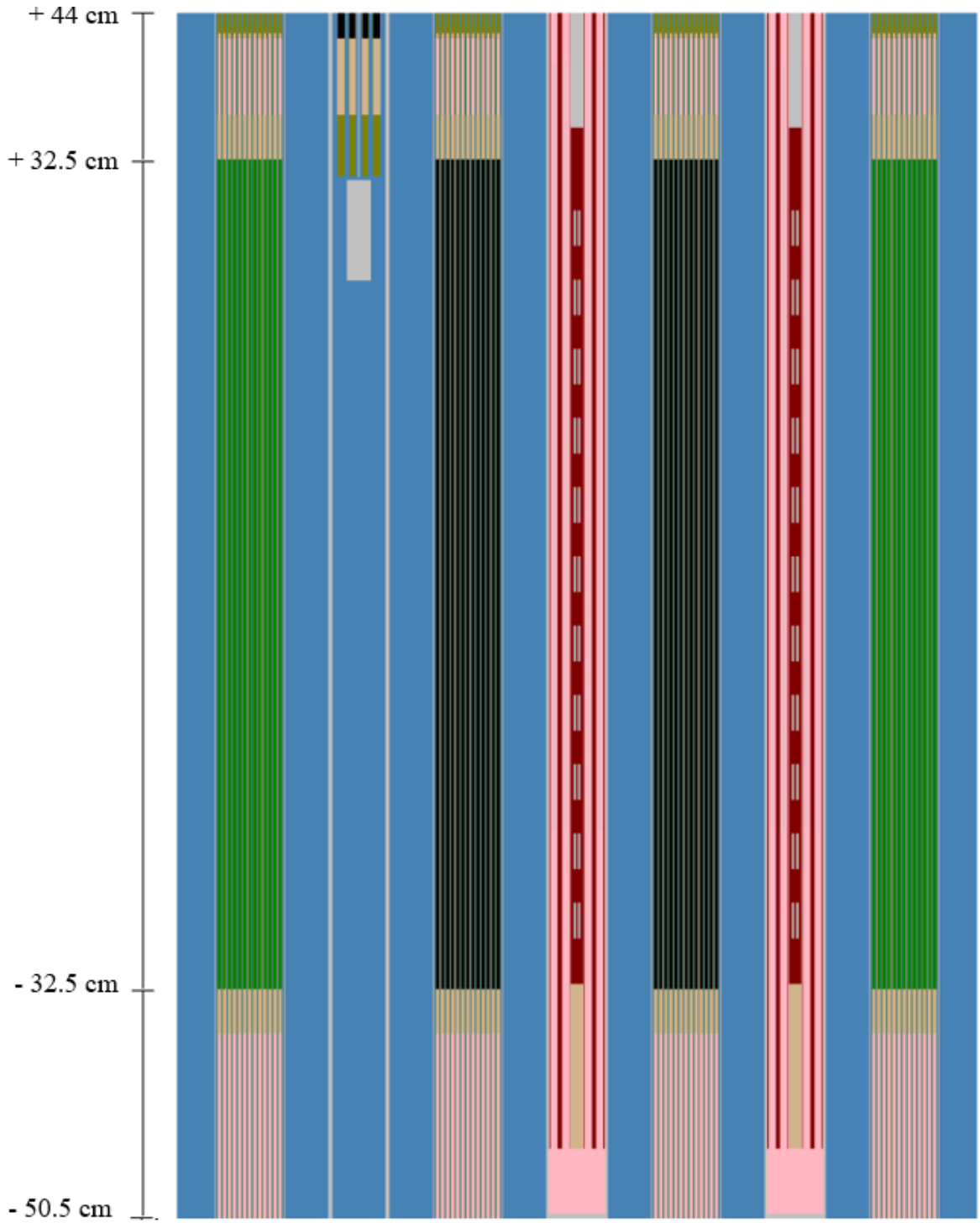


Figure 18: Reduced axial mid-span cross-section view in the xz-axis

In Figure 18 the axial xz-axis section of the reduced model is depicted. A comparison of this figure with Figure 16 reveals that a vast majority of the inactive region is excluded from simulation.

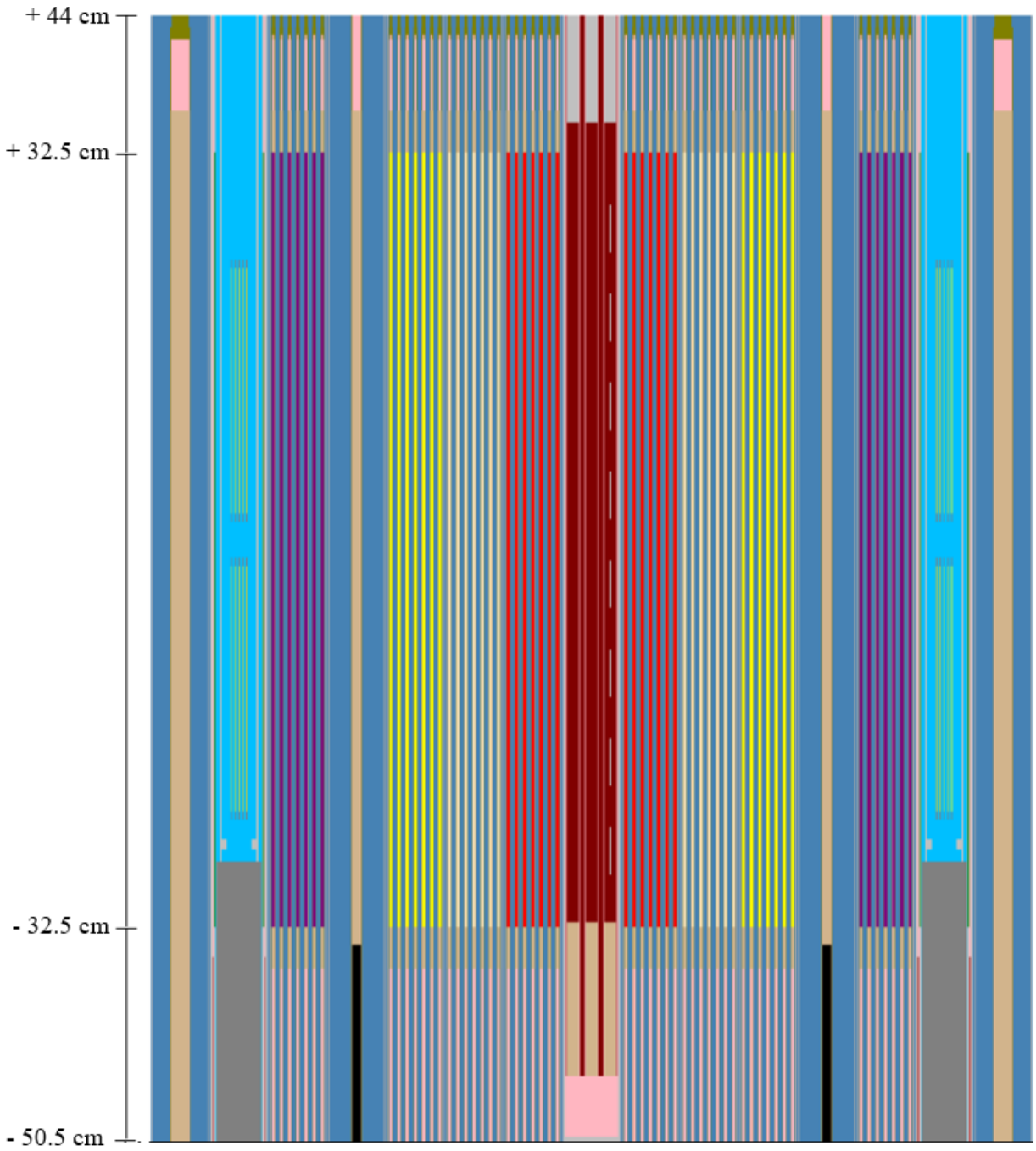


Figure 19: Reduced axial mid-span cross-section view in the yz-axis

On the other hand, Figure 19 portrays the yz-axis section view of the reduced model, where the cut plane is located at the origin. The special features of this view are the two thermal IPSs, the control rod and the central fast IPS.

6.3 Simulations

In this case study, three types of simulations were carried-out using OpenMC. The first one was executed on the full core model using a continuous energy neutron cross section library prepared for different nuclear temperatures from JEFF 3.1.2. This simulation result is used to benchmark the code with MCNP. The second simulation was performed on the reduced core model using the same library. The purpose of this simulation was to generate multi-group homogenized cross sections. The third simulation was carried-out using the homogenized cross sections. The purpose of this simulation was to compare continuous energy results versus multi-group ones. The output of this simulation was also compared with a deterministic code.

For all the simulations a total of $5x10^7$ particles were sampled in each cycle, with a total of 250 cycles, of which 50 were set to be inactive. The simulation was executed using SCK owned Fermi4 cluster. A single node with 4 physical processors working in parallel and 12 virtual processors linked to each physical processor was dedicated for the simulation.

7 RESULTS AND ANALYSIS

This section is dedicated to discussing the simulation results and milestone observations. For the ease of presentation and discussion, the findings are categorized into three major parts. The first sub-section presents a code verification benchmark between OpenMC and MCNP (only post processing of the results from this code is carried-out by the writer) using the full-scale model. The findings of code verification between OpenMC and MCNP are illustrated by comparing the volume normalized mesh tally that gives rise to the spatial distribution of the flux, along with a comparison between effective multiplication factors.

Secondly, studies related to nodal cross-section homogenization, based on the reduced model were carried-out. Homogenized multi-group constants, normalized to the volume of each of the 3211 nodes were generated through the process.

Finally, in the third sub-section the nodal cross-sections that were generated with OpenMC were used as input parameters for a deterministic core simulator known as PHISICS (here as well, only the post processing calculations were conducted by the writer). Afterwards, flux computations performed by PHISICS at different sites of the core are illustrated. In the end, a flux comparison between the reduced MYRRHA 1.6 model in OpenMC, both in continuous energy and in multi-group mode were presented in comparison with PHISICS.

7.1 OpenMC code verification

One way of evaluating the reliability of a relatively new code is by verifying it against other well-established codes. In neutronic codes the importance of verification is twofold. First, it helps to benchmark whether the physics has correctly mimicked the neutron-nucleus interactions. Secondly, programming errors can be identified and rectified. Albeit, it should be noted that no computer code is free of biases that are driven from drastic approximations, and uncertainties of results that may arise from the source data impurities.

Part of the objective of this thesis is to verify the flux density and multiplication factor calculated by OpenMC with the well-established Monte Carlo code, MCNP, and to compare the time taken by each code to simulate the case. MCNP is chosen to verify OpenMC in 3D full core calculations, because it has a long history of development and it has gone through numerous experimental validations. Thus, there is a profound consensus that it is the most reliable Monte Carlo code within the range of the aforementioned biases. In addition, since

both codes use the cross-section data from the same source, variation due to nuclear data uncertainties will be minimized.

7.1.1 Benchmark of spatial distribution of flux at the central channel

Even though evaluating the flux distribution is not a goal by itself, it is a means to calculate integral effects such as reaction rates, burn-ups and irradiation effects (Bowman, et al., 1977). In critical MYRRHA design the central channel is allocated for high flux irradiation of materials. Thus, it is to the interest of this thesis to check the true axial flux distribution at the central fast IPS slot. To this end, an energy integrated axial flux is calculated based on the track-length data from the simulations of OpenMC and MCNP. The following figure shows the axial flux plots from both codes.

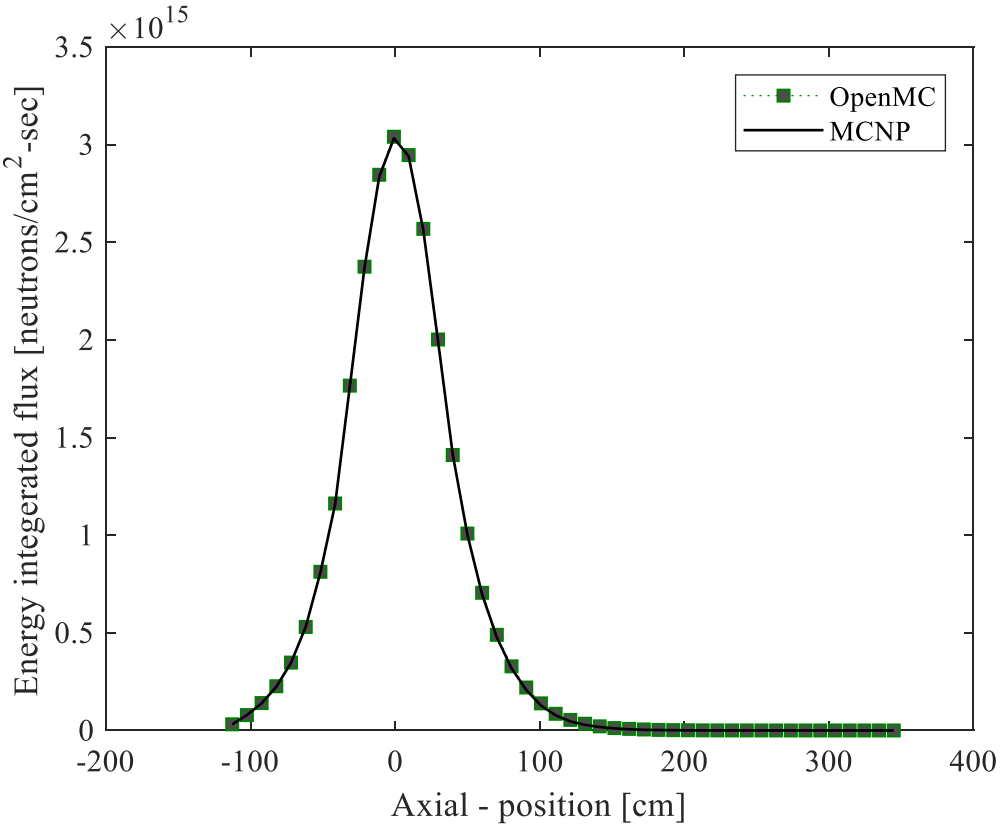


Figure 20: Energy integrated flux at the central channel

The revised MYRRHA Rev-1.6 design report (Malambu & Stankovskiy, 2014) stated that in the previous designs, there was a quest for a fast neutron flux in the order of 10^{15} at the central fast IPS. According to a flux calculation based on the data extracted from both codes,

the maximum flux is estimated to be $3.04 \cdot 10^{15}$ n/cm^2 sec. Besides, the axial flux distribution plot in Figure 20 confirms the aforementioned result.

7.1.2 Eigenvalue and other flux distribution benchmarks

As it is mentioned in the previous section, since OpenMC is a relatively new code, it is important to evaluate its computational accuracy before relaying on the code generated multi-group cross sections. For this reason, the energy integrated flux tally of OpenMC 0.10.0 is benchmarked vs. the MCNP 6.2 energy-integrated neutron flux.

In both cases the simulations were carried-out on a MYRRHA core model that was meshed into 30x30 radial nodes, and 46 axial nodes. The resulting normalized flux tallies are shown in Figure 21.

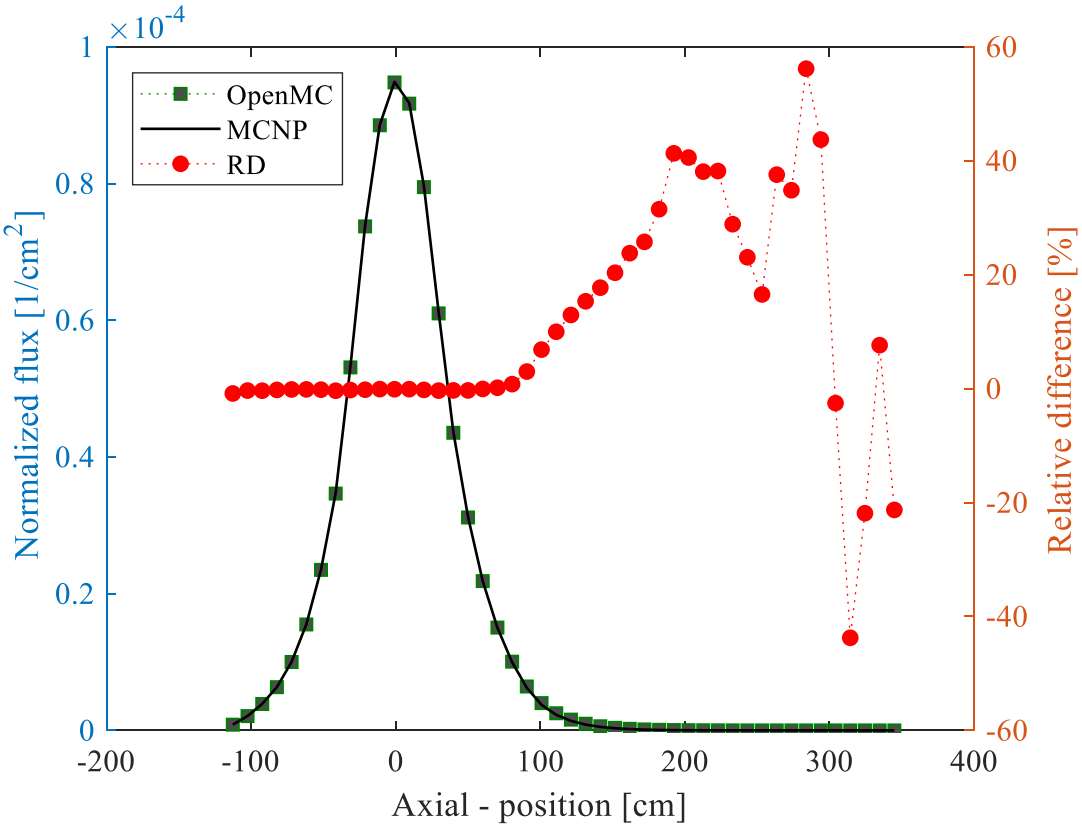


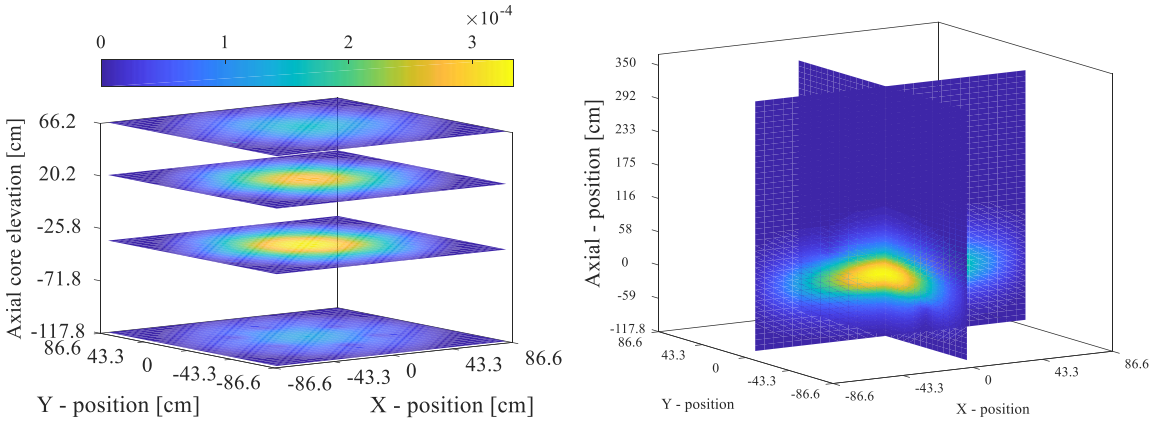
Figure 21: Full core radially averaged volumetrically normalized flux benchmark and the relative difference.

From Figure 21 it can be deduced that the result from OpenMC adequately resembles the average axial flux output from MCNP. An additional calculation of percentage relative

difference is executed to closely investigate the deviation between MCNP and OpenMC. The right y-axis of the above figure shows the percentage relative difference of average axial mesh tally flux in track-length of the two codes.

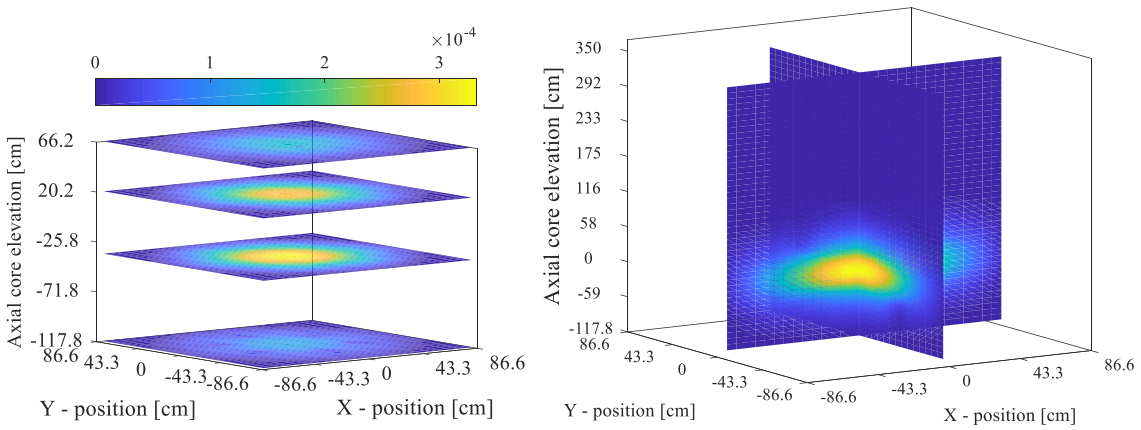
Figure 21 depicted a relative difference of < 1% in the lower reflected and fissile region. Whereas, in the upper region of the core, where the neutron density is negligible, differences more than 50% were registered. The seemingly significant difference in the upper region should not be considered as a programming bias. Rather, the relative difference is magnified due to the stochastic nature of Monte Carlo method. As it has been discussed in the earlier sections, in MC calculations a reliable result can be found at a higher population density. However, from the same figure it can be seen that above elevation +66 the flux density is close to zero. Therefore, it would not be far from truth to conclude that the high percentage difference arises from a lack of sample in the region. On the contrary, the strong similarity in the high-density region indicates that OpenMC a promising MC code at least in the academic realm.

In addition, the radial mesh tally track-length distribution estimated by both codes is plotted with a 3D color map at different elevations of the model. Figure 22 shows a 3D color map plot of the radial and axial flux distribution in mesh tally.



(a) (b)

Figure 22 (a) and (b): Color map of radial and axial flux distribution by OpenMC.



(c) (d)

Figure 22 (c) and (d): Color map of radial and axial flux distribution by MCNP.

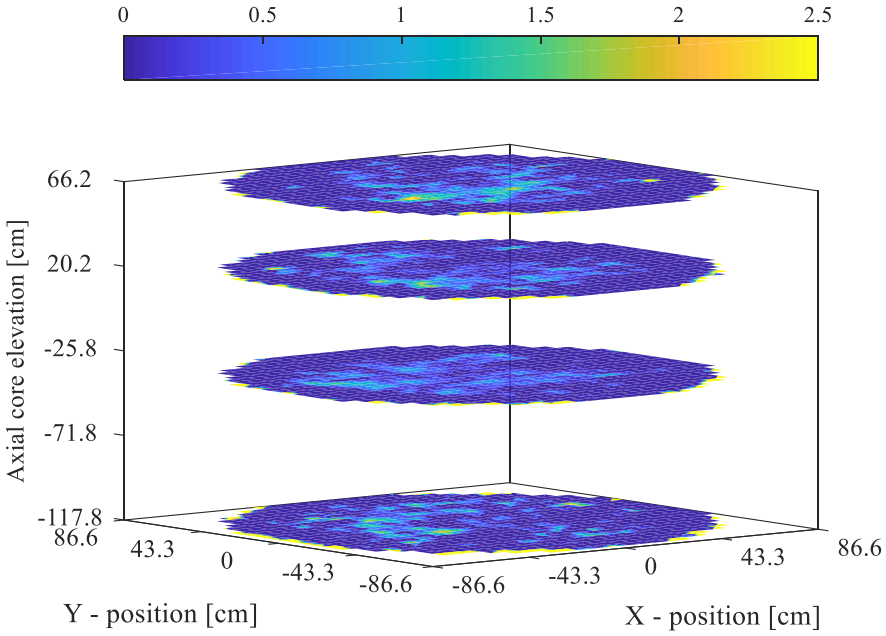
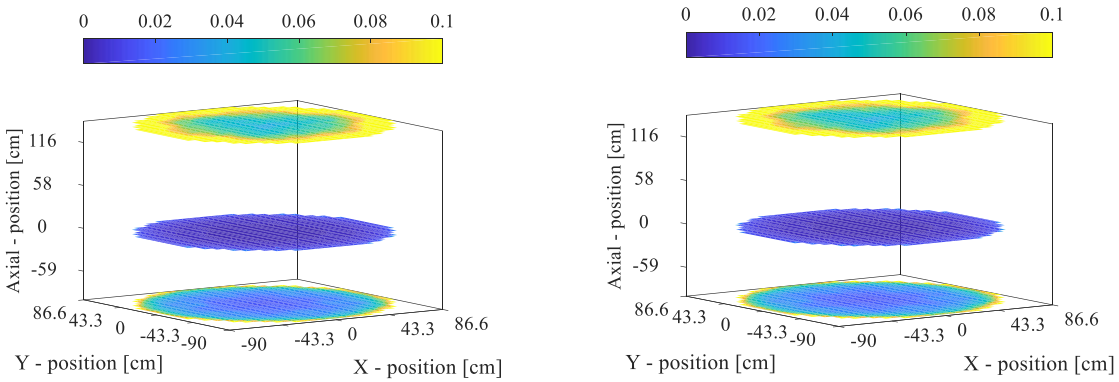


Figure 22 (e): the relative difference between OpenMC and MCNP.

In Figure 22 the color map shows that the flux distribution estimations by OpenMC are quite close to the one estimated by MCNP. From **Error! Reference source not found.**(e) the relative flux distribution difference between OpenMC and MCNP is less than 3%. Secondly, comparing the output of Figure 22 with Figure 21 the results show similarity in the sense that the flux density is negligible above elevation +100.

Further investigation was done on the radial and axial standard deviations to see the confidence interval at different regions of the core. As it can be seen from Figure 23(a), (b) and (c) relatively high unreliability was registered around the peripheries and the upper region of the core. These are regions where, there is less particle activity. Therefore, this result can be interoperated in such a way that in stochastic simulations, under-sampling can drastically affect the confidence interval of distribution estimators (see Section 5).



(a) (b)

Figure 23 (a) and (b): Color map of radial standard deviations by OpenMC and MCNP respectively.

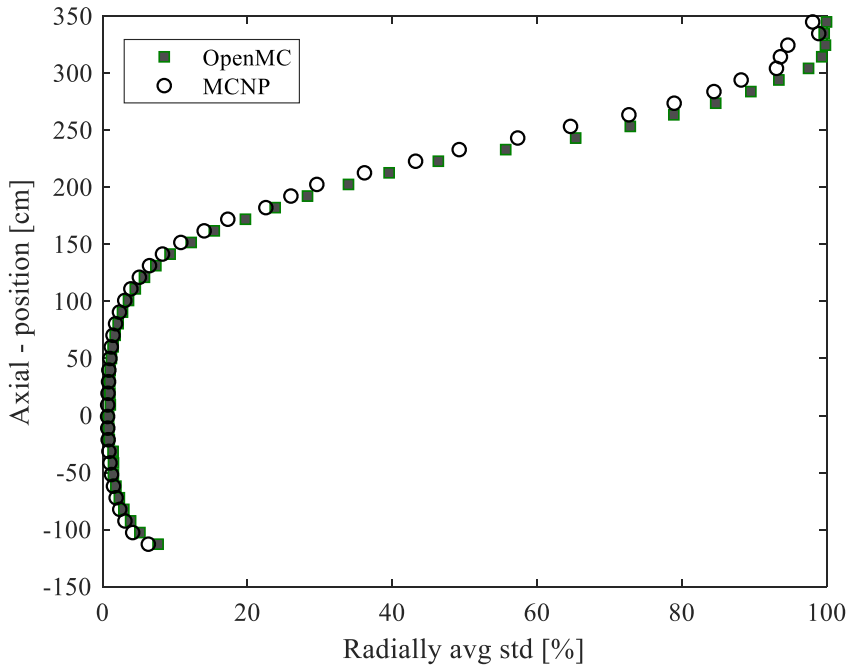


Figure 23 (c): Radially averaged standard deviation.

7.1.3 OpenMC Eigenvalue estimation and comparison

In critical core simulations, OpenMC uses eigenvalue as a criticality source method. Thus, it is essential to know the eigenvalue estimate, in order to determine whether the chain reaction is self-sustaining. As it is discussed in Section 3.2.5 the eigenvalue is estimated based on the ratio of two successive neutron cycles. Table 5 below shows the resulting k_{eff} for the BOC MYRRHA-1.6 with all control rods out of the core.

Table 5: OpenMC and MCNP k_{eff} result

	k_{eff}
OpenMC 0.10.0	1.01295 (+/- 11 pcm)
MCNP 6.2	1.01312 (+/- 9 pcm)

The result shows that the combined k_{eff} is 1.01295 with a standard deviation of 11 pcm, when all the shimmering control rods are inserted. From this the reactivity can be calculated using the following equation.

$$\rho = \frac{k_{eff} - 1}{k_{eff}} \times 10^5 \tag{7.1}$$

$$\rho = +1278\, pcm$$

The above calculation shows that there is an excess reactivity at the beginning of cycle (BOC), which could be controlled by burnable absorbers. In addition, the time statistics of the simulation showed that the total time elapsed is in the order of 10^4 seconds. While, a similar simulation model on MCNP has taken $1.644 \cdot 10^4$ seconds (Solis, 2018).

On the other hand, the OpenMC eigenvalue estimate was compared with the reference MCNP eigenvalue estimate. In both cases continuous energy distribution was considered, and the simulation was executed assuming vacuum boundary conditions in all sides of the model.

From Table 5 the eigenvalue bias can be calculated by comparing the eigenvalue from OpenMC with the reference eigenvalue from MCNP, as follows.

$$\Delta k_{eff} = \left(k_{eff}^{MCNP} - k_{eff}^{OpenMC}\right) \times 10^5 \tag{7.2}$$
$$\Delta k_{eff} = 17\ pcm$$

The calculation shows that the OpenMC eigenvalue is consistent with the reference MCNP with only 17 pcm difference (i.e. such difference lies within the statistical uncertainty given by both codes). This demonstrates that the global reaction rate is preserved with OpenMC.

7.2 Homogenized cross-sections

As it is stated in Section 3.2.6, OpenMC has a capacity to generate reaction constants based on the tally scores of reaction rate and flux estimates from simulation results. Moreover, it also constructs multi-group scattering matrices at different Legendre-orders. The results are calculated according to the OpenMC multi-group cross-section evaluation approach described in Section 4.2. For this purpose, the reduced MYRRHA model is chosen per the explanation in Section 6.2. Then, this model is divided into 19 spatially homogenized axial nodes and radially the existing hexagonal assembly boundary has become the homogenized nodal boundary.

On the other hand, the continuous energy is condensed into 20 groups. Before deciding to resort at the 20 energy groups, several simulations were executed with finer and coarser energy meshes. For instance, a 33 group ERANOS (Rimpault, et al., 2002) meshing for fast reactors was the first to be implemented since it is supported by previous experiments and researches. However, for the reason that will be shown in the upcoming figures and paragraphs finer meshing does not work for MYRRHA.

Figure 24 below shows the relative error of total MGXS normalized by its mean value. The graph shows that in general at lower energy groups the accuracy of the resulting cross-sections deviates from the mean value by a larger margin. Especially, the near center nodes show the highest percentage of normalized deviation from the mean. This has proven that using a fine energy mesh does not necessarily improve the quality of the cross-sectional data, however it may take a toll in computational requirements.

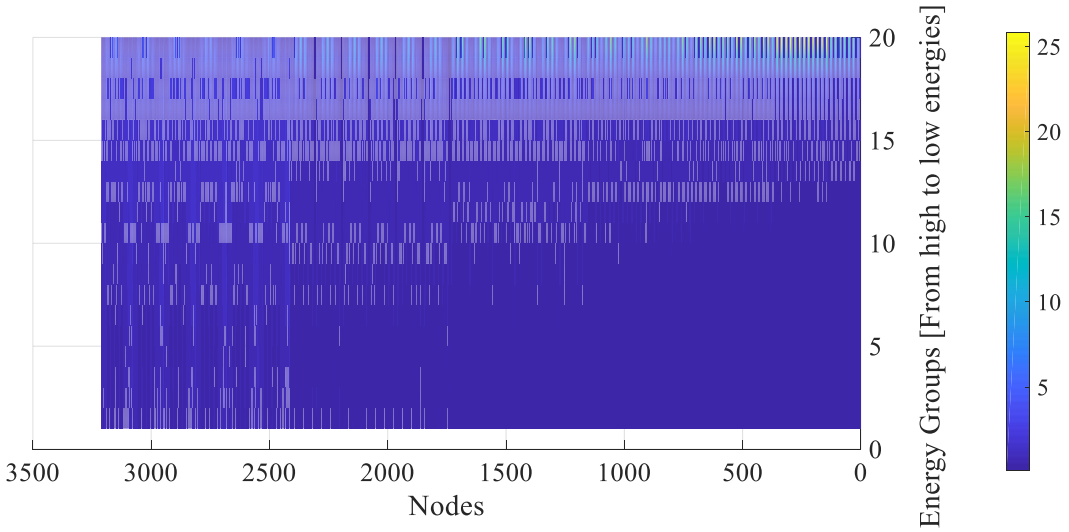


Figure 24: Relative error of the total MGXS.

Table 6: Energy mesh used for homogenization.

Energy [eV]					
	G I	G II	G III	G IV	G V
	20.00E+06	1.00E+06	4.98E+05	3.02E+05	1.83E+05
	G VI	G VII	G VIII	G IX	G X
	1.11E+05	6.74E+04	4.09E+04	2.48E+04	1.50E+04
	G XI	G XII	G XIII	G XIV	G XV
	9.12E+03	5.53E+03	3.35E+03	2.03E+03	1230
	G XVI	G XVII	G XVIII	G XIX	G XX
	749	454	304	149	91.7

Further analysis on the normalized standard deviation showed that the high frequency of deviation is below 3%. Figure 25 indicates that even though there were more than 2000 instances of deviations from their mean values, the percentages of these deviations were close to zero. On the other hand, the highest percentage of deviation, which was 25% had a frequency of occurrence only twice. From these findings, it can be deduced that since the majority of the values are near to the mean value, the accuracy of the calculation code is acceptable.

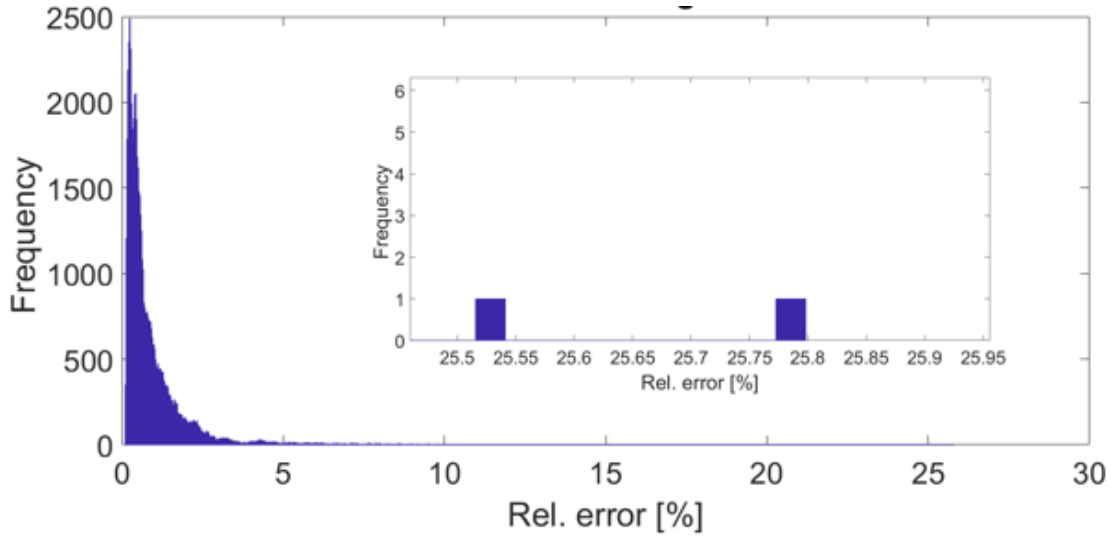


Figure 25: The frequency of the normalized deviation of MGTXS from its mean value.

7.2.1 Homogenized MGSXS generated by OpenMC

Deterministic nodal diffusion codes require absorption, fission and scattering cross-sections, and other constants, such as diffusion coefficient, fission spectra and neutron production (nu) to execute the diffusion equation. Of these parameters, it is interesting to investigate the scattering cross-section at different locations in the reactor core. The scattering matrix is appealing for two reasons. First, it requires further treatment of transport correction due to its angular dependences. Secondly, it is the only interaction constant that involves group transfer.

The OpenMC code is fitted with a sub-class that can be used to generate a scattering matrix of spatially homogenized and energy-integrated multi-group cross-sections with the cosine of the change-in-angle represented by one or more Legendre moments. Therefore, the energy groups, the property of the domain of interest and the order of the Legendre moment are the requirements to be set to produce the scattering matrix. (Romano, 2018)

With these in mind, the effect of scattering in a good scattering medium (water) and in a fissile region was investigated. Hence, two sites in the reactor were selected based on the importance of the material of those regions for scattering as stated in reactor physics literatures (Lamarsh & Baratta, 2014. P 258). According to these literatures, water is known to be a good scatterer and in the fissile region scattering is low due to the fast energy spectra. Therefore, the thermal IPS that is water medium was chosen as a representative of a good

scatterer, and the fuel assembly represented the second case. Figure 26 shows a scattering cross-section matrix in thermal IPS site.

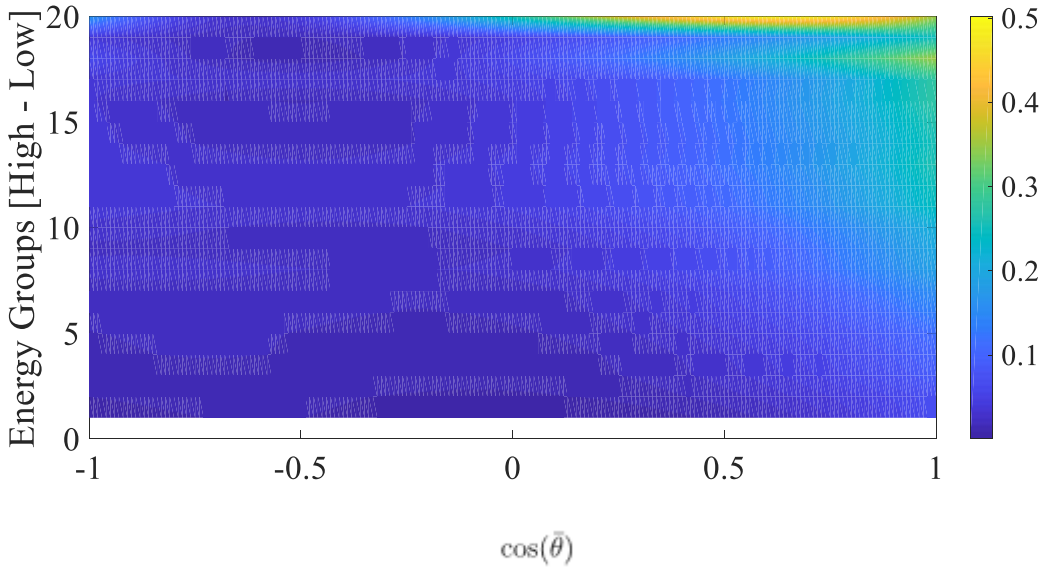


Figure 26: Multi-group scattering cross-section matrix in thermal IPS (water medium).

The above figure shows that in fast energy region the macroscopic scattering cross-section is low, and the angular dependency is also insignificant. On the other hand, at lower energy spectrum the angular dependency of the scattering cross-section is highly significant. Moreover, the general trend shows that the scattering cross-section increases at lower energy groups as expected.

Next, a transport corrected macroscopic scattering cross-section in the fissile region was investigated. Figure 27 shows the graph of the resulting values.

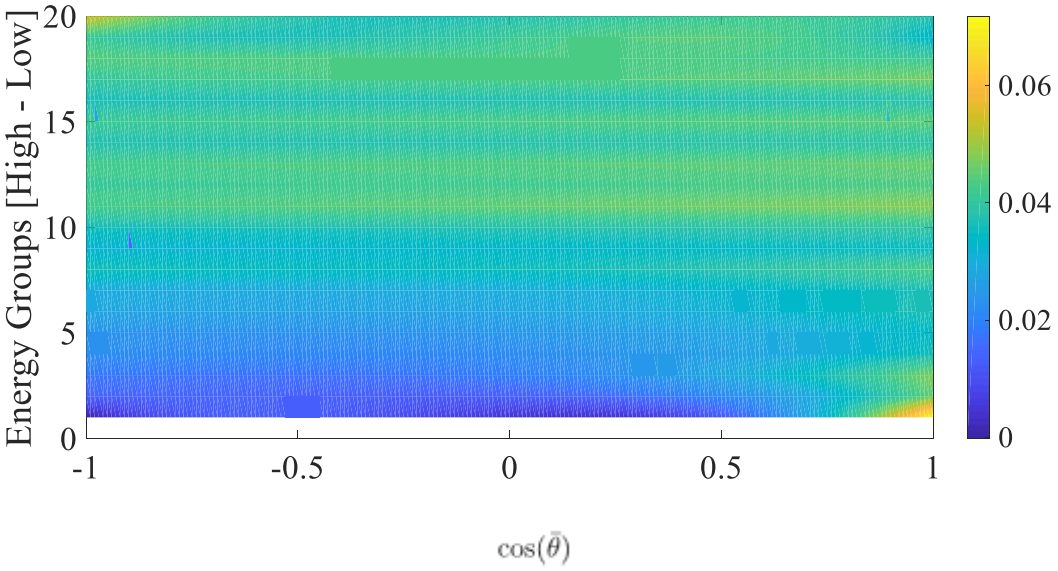
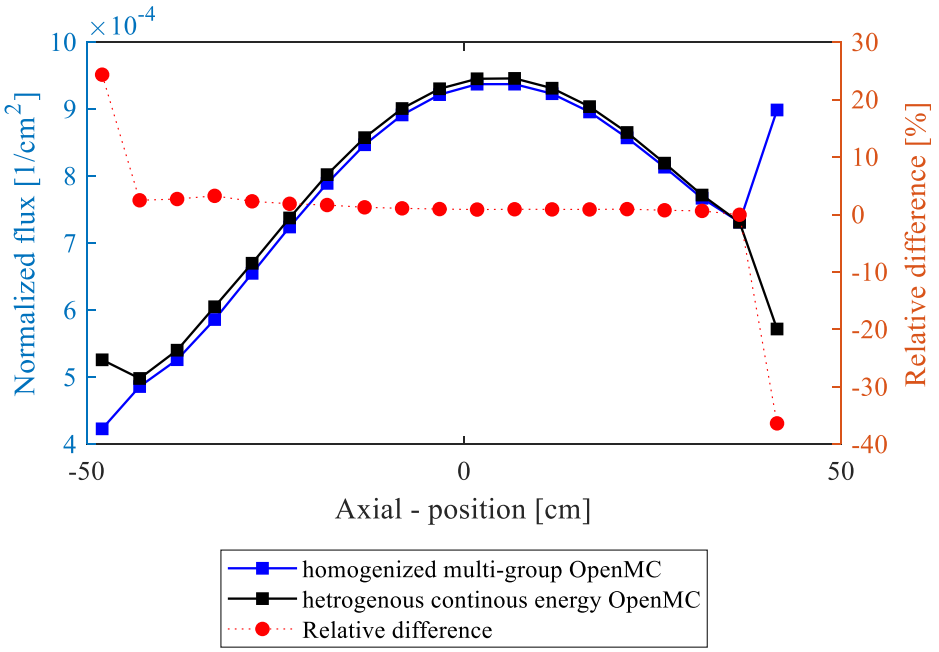


Figure 27: Multi-group scattering cross-section matrix in fuel assembly.

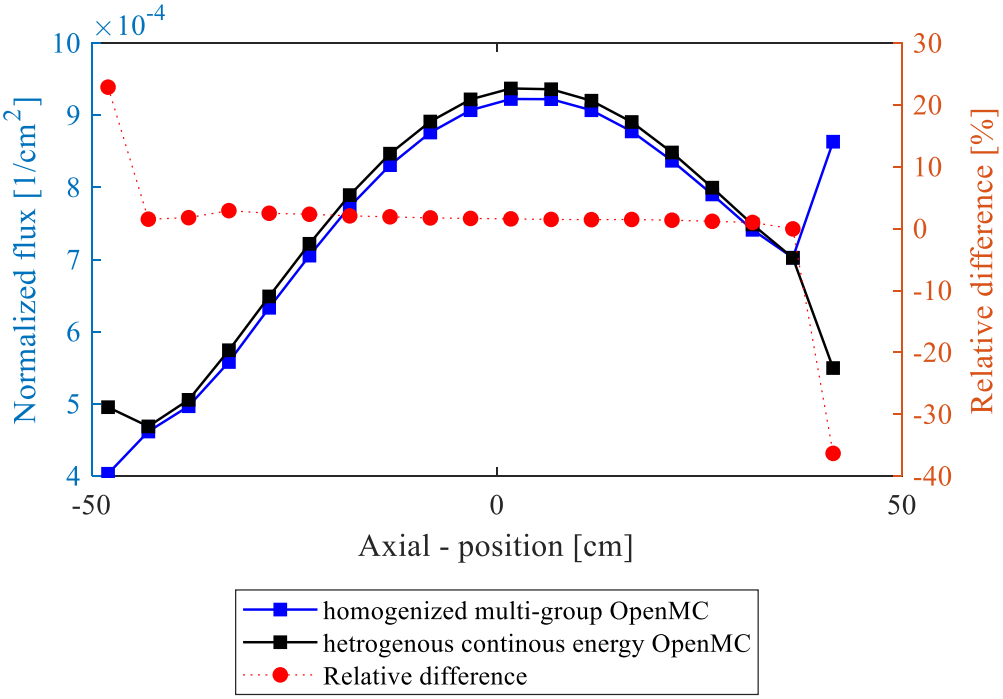
As it was expected, the scattering cross-section interval is smaller in the fuel assembly. In addition, from Figure 27 it can be observed that the angular dependency is significant rather at the higher energy region than the lower energy region. In comparison, Figure 26 and Figure 27 have proven that water has a higher scattering cross-section.

7.3 Comparison between heterogenous OpenMC and MGMC

The eigenvalue comparison between the two models shows that there is a difference of 489 pcm. In another context, a local level comparison of energy integrated and volume weighted flux was carried-out and the results are portrayed in Figure 28.



(a)



(b)

Figure 28 (a) and (b): Energy integrated flux comparison of heterogeneous and MGMC models at the central IPS and fresh fuel assembly respectively.

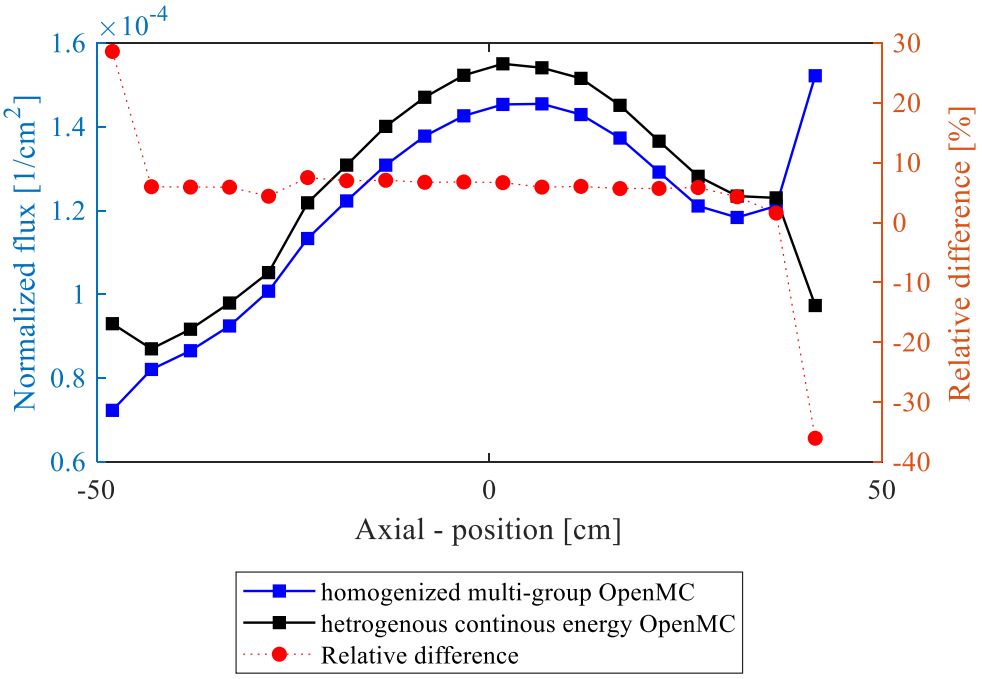


Figure 28 (c): Energy integrated flux comparison of heterogeneous and MGMC models at thermal IPS.

As it can be seen from the figure, the volume normalized fluxes are compared in the central IPS, the fuel assembly and thermal IPS respectively. In all these regions, the relative differences are within acceptable range except at the top and bottom ends. The fact that the trend of these divergences is similar in all cases, has to do with a boundary related issue. Due to the fact that the media is highly anisotropic in the core (i.e. relatively small core with a dominating fast spectra and thermal islands), the angular treatment in the modeling is fundamental. Therefore, higher than P3 expansion order in the Legendre scattering treatment is required if more accurate results are to be expected.

7.4 PHISICS flux output in specific regions of the core

In addition to the Monte Carlo codes, a deterministic package INSTANT (Intelligent Nodal and Semi-structured Treatment for Advanced Neutron Transport) from a reactor physics analysis toolkit PHISICS (Parallel and Highly Innovative Simulation for INL Code System), which is developed by Idaho National Laboratory was also used to simulate the core by

another colleague involved this project. The INSTANT solution taken into account in this work follows the discretization of the neutron transport equation based in the Variational Nodal Method. In space, this corresponds to a hybrid finite element method, while in angle it corresponds to a spherical harmonic expansion.

Comparing the outputs of this deterministic simulation are essential for this project, since the nodal input parameters of the PHISICS model correspond to the multi-group macroscopic cross-sections that were previously computed by the reduced MYRRHA 1.6 OpenMC model. Fluxes were then calculated in three significant regions of the core. The locations were selected based on their material composition, function and position. Below, is depicted the flux output of the central IPS. This site was selected due to its functional significance. As it was mentioned in Section 6.1 this slot has a specific purpose, hence a targeted value of true flux is expected to be attained.

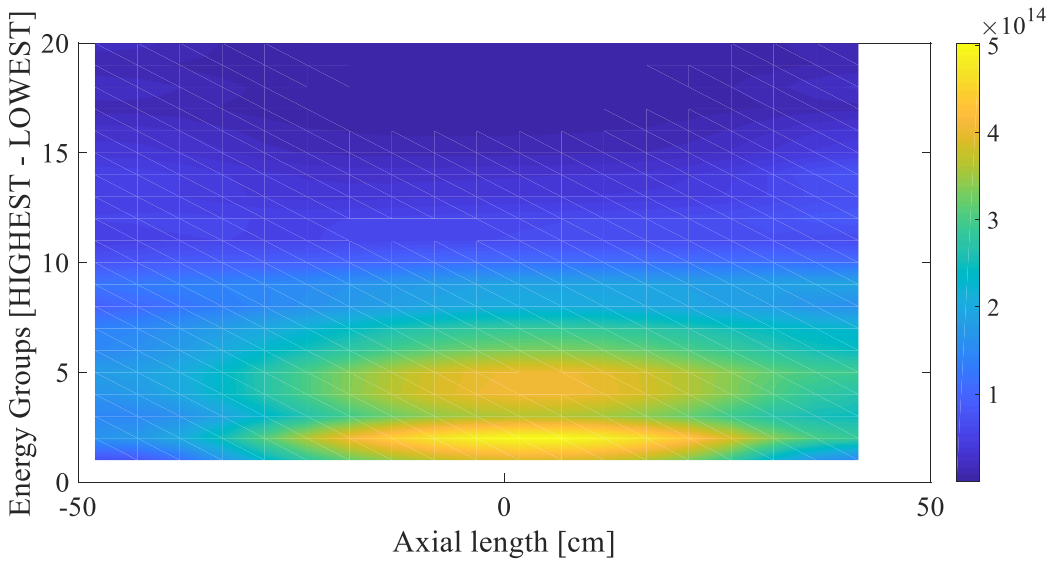


Figure 29: PHISICS flux output in the central IPS

In the central IPS, the maximum flux is registered at the fast energy region and the neutron flow density reduced towards the lowest energy region (see Figure 29). This phenomenon is expected since MYRRHA is a fast reactor and the central IPS is designated for fast neutron irradiation of materials.

Next, the flux in the fuel assembly was investigated. It is important to evaluate the fission rate density and its axial and radial distribution along the assembly. Some of the benefits of mapping the flux density distribution are (Bowman, et al., 1977):

- To determine safe and optimized operation of the reactor.
- To locate regions with maximum flux density distribution.
- And, for hydro-thermal calculations.

Figure 30 shows the true flux distribution in the fuel assembly. Figure 30 shows that the peak flux distribution is at the axial center of the cluster and in the highest energy group. This proves the theory that neutrons immerge with high energy immediately after a fission reaction.

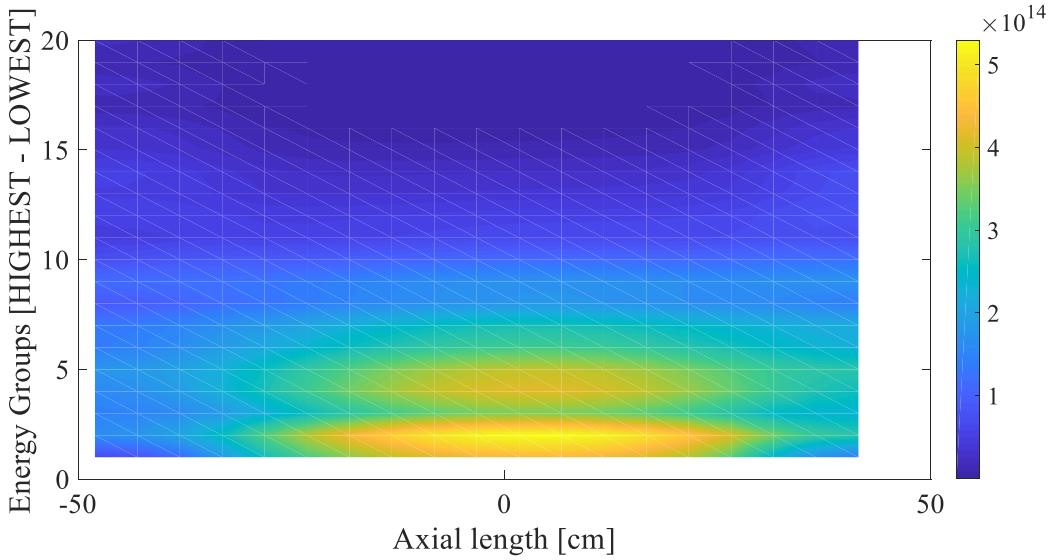


Figure 30: PHISICS flux output in the fuel assembly

One unique feature of MYRRHA 1.6 is that it has six thermal islands known as thermal IPSs, near the periphery for radionuclide production. These locations are of interest, due to the importance of ensuring neutrons in these regions are thermalized enough for the intended purpose. It is also advantageous to know the flux distribution in thermal IPSs owing to their strategic global position in the reactor, surrounded by relatively faster neutron spectrum. Hence, the output of flux distribution as calculated by the PHISICS code is shown in Figure 31.

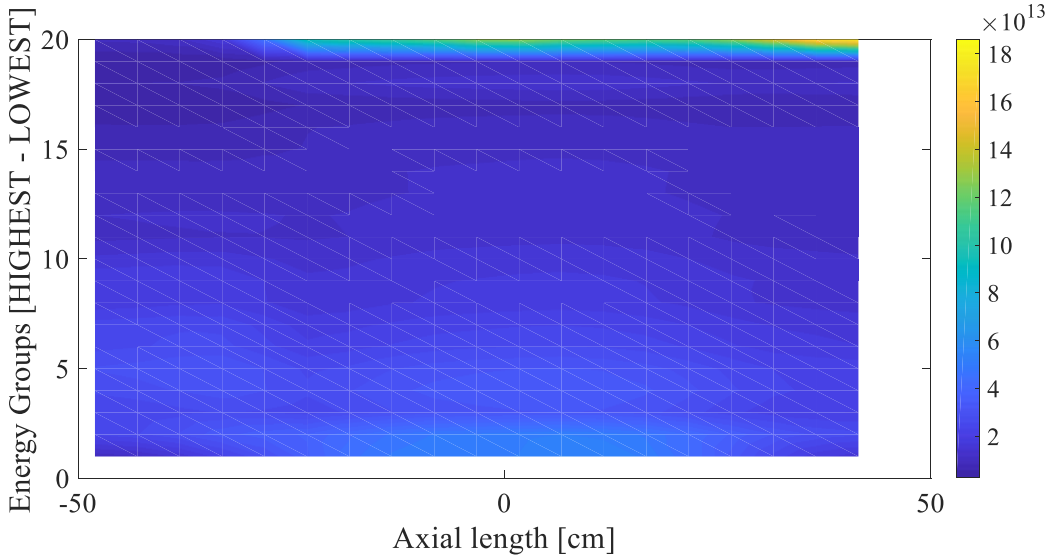


Figure 31: PHISICS flux output in the thermal IPS.

From Figure 31 the peak flux distribution is at the lowest energy spectrum. This is consistent with the intended design of the reactor.

7.5 Code comparison OpenMC vs PHISICS

Deterministic codes are programmed using either diffusion theory or neutron transport formula. PHISICS being a deterministic code, is programmed based on the transport equation. Even through, transportation equation considers all the possible interactions, it is a good trend to benchmark it against a stochastic code to verify that it is free of any calculation and/ or programming errors.

Hence, the eigen value and flux results of PHISICS were compared with the outputs of OpenMC model. Again, the group-homogenized and reduced MYRRHA model was used for this purpose. Table 7 illustrates the effective multiplication factor comparison between OpenMC – multi-group and PHISICS models.

Table 7: k_{eff} comparison between OpenMC homogenous model and PHISICS.

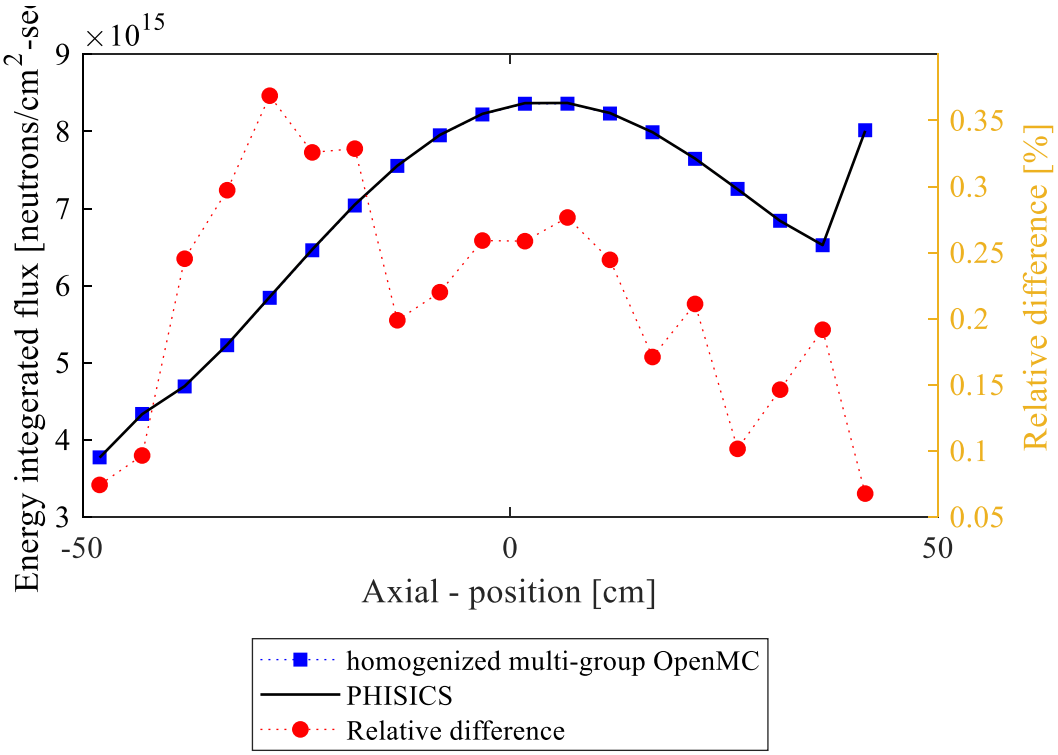
	k_{eff}
OpenMC – heterogeneous continuous energy for reduced model	1.04413 (+/- 3 pcm)
OpenMC – MGMC simulation	1.04902 (+/- 3 pcm)
PHISICS model	1.05073

Table 7 shows that the eigen value bias between the continuous energy OpenMC and PHISICS is 660 pcm. Moreover, comparisons between multi-group OpenMC and PHISICS, and multi-group OpenMC and continuous energy OpenMC also exhibit significant differences. In the first case the range is 171 pcm, whereas in the second comparison the difference is 489 pcm. These discrepancies are mainly attributed to the following factors: complex geometry of the core, high heterogeneity of the reactor (for instance, the presence of thermal IPSs in a fast reactor) and leakages (see Table 8).

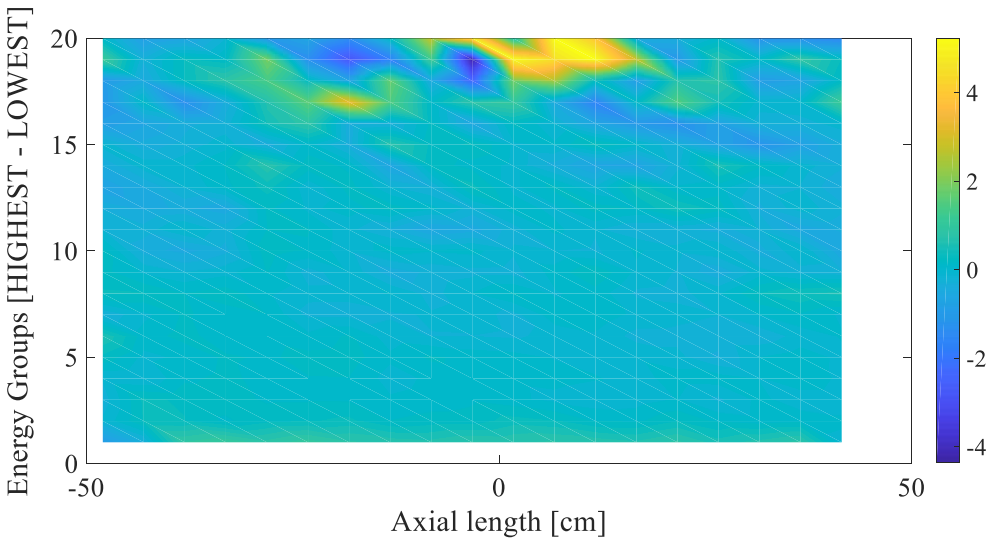
Table 8: OpenMC leakage fraction.

	Leakage fraction
Heterogeneous OpenMC	0.20241 (+/- 0.00002)
Homogeneous OpenMC	0.19876 (+/- 0.00002)

On the other hand, the flux estimates were compared at the local domains that were mentioned in the previous section (see Section 7.4). The figures below portray the energy-integrated fluxes and the relative differences of the group fluxes between the two codes in the respective locations.

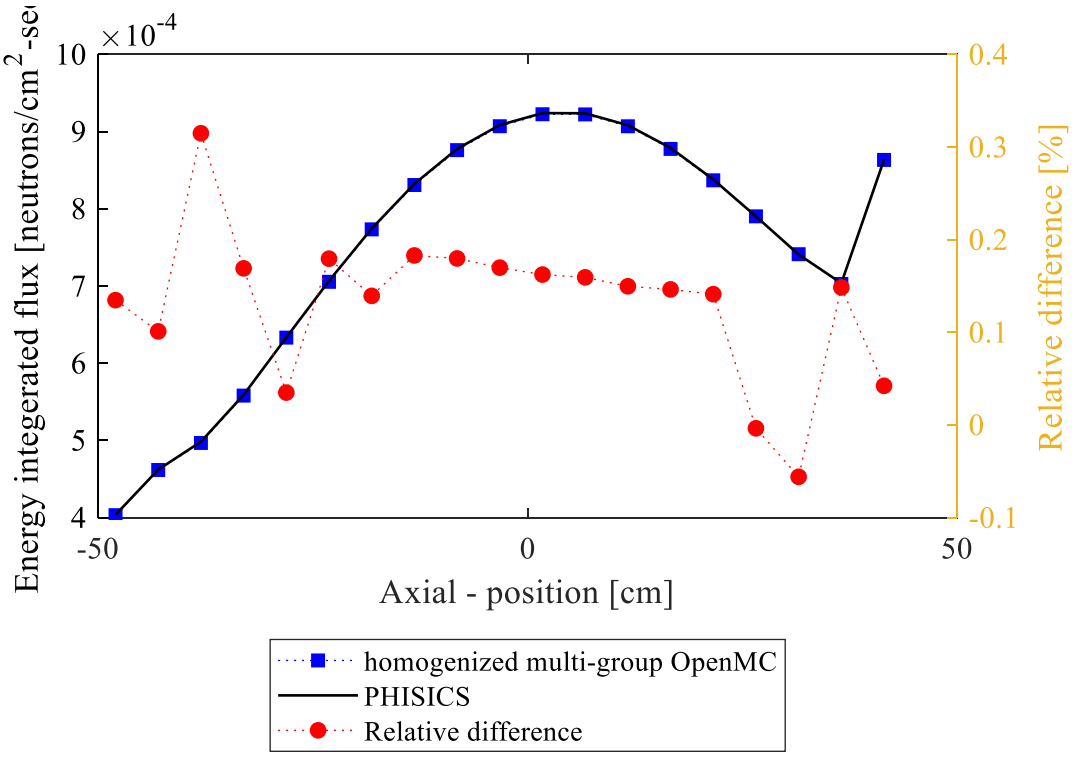


(a)

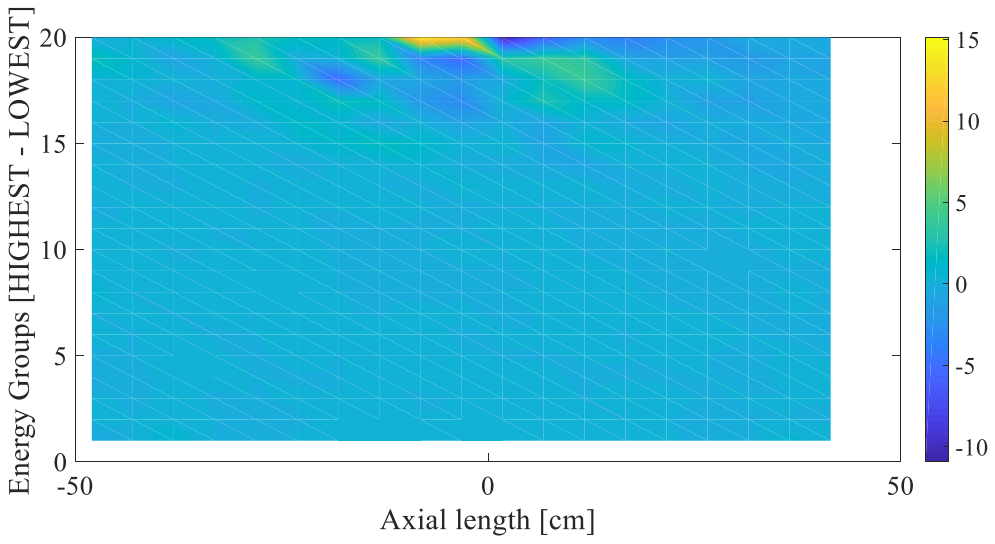


(b)

Figure 32: (a) Energy integrated fluxes. (b) Flux relative difference between OpenMC and PHISICS in the central IPS.

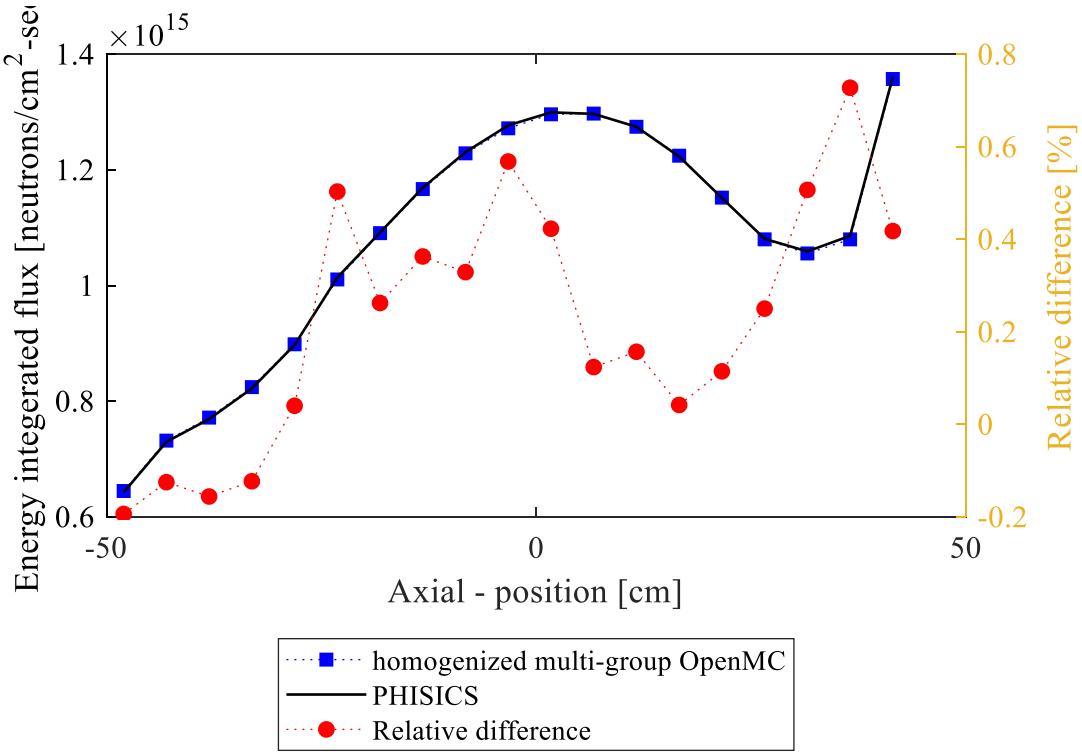


(a)

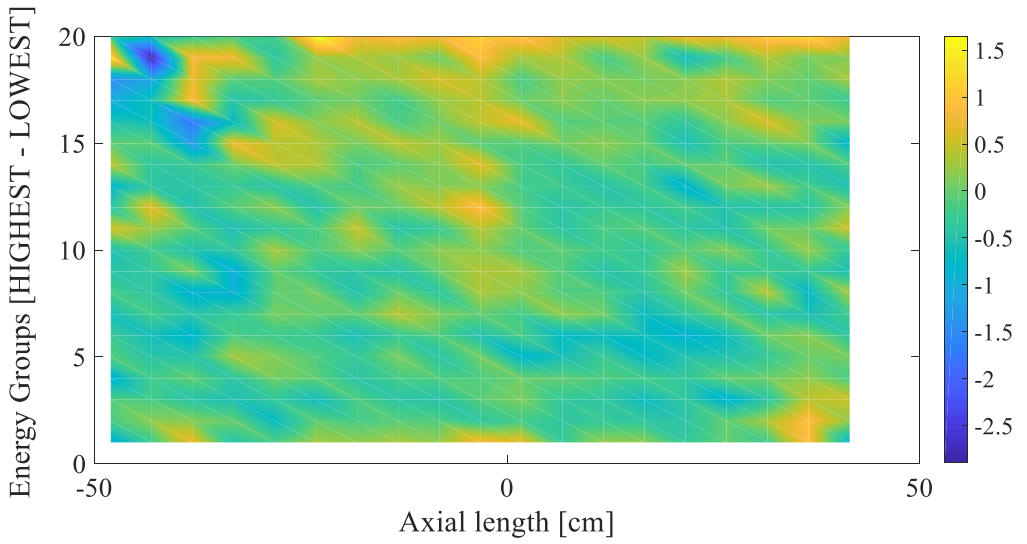


(b)

Figure 33: (a) Energy integrated fluxes. (b) Flux relative difference between OpenMC and PHISICS in the fresh fuel assembly.



(a)



(b)

Figure 34: (a) Energy integrated fluxes. (b) Flux relative difference between OpenMC and PHISICS in the thermal IPS.

According to Figure 32 (a), the maximum energy integrated relative difference of fluxes in the central IPS (highest energy spectrum) was registered as 0.35%. Also, from Figure 32 (b) the multi-group spectrum relative difference of fluxes in the same domain was not more than 5%. Elsewhere, in the fresh fuel assembly the maximum energy integrated flux relative difference between the two codes was < 0.4 %, and the multi-group relative difference was about 8 % (see Figure 33 (a) and (b)). The highest energy integrated flux relative difference was observed in the thermal IPS with 0.8%. Moreover, the group relative difference was about 1.5 % (see Figure 34 (a) and (b)). Given, the above results, it can be said that there is a reasonably acceptable agreement between OpenMC and PHISICS.

8 CONCLUSION AND RECOMMENDATION FOR FUTURE WORK

This work is the first of its kind to engage a Monte Carlo code, OpenMC, for a global 3D spatially homogenized generation of multi-group reaction constants. Also included in this thesis are, code verification with a reference model in MCNP and comparisons with a deterministic code, PHISICS.

In the process of modeling, it has been proven that compartmentalization of inputs into logical XML files made adding or modifying the parameters easy. In addition, the implementation of python API in the code made it possible to specify simulation parameters, simulation execution and retrieve the tallied estimates. Furthermore, the object-oriented data processing system of OpenMC has been proven to provide fast and efficient computational results, indexing and storage of tallies over the specified spatial domain.

One of the purposes of this thesis was to verify whether the outputs of OpenMC 0.10.0 are comparable with MCNP 6.2. In this regard, the comparison showed no major differences in mesh tally results. The highest relative difference of energy integrated flux in the active region was less than 1%, and the effective multiplication factor discrepancy was in the order of 17 pcm. In terms of computational time, OpenMC was found to be 1.7 times faster than MCNP (Solis, 2018). Such a difference was observed in the absence of any computational capability favoritism for one or the other, i.e. both simulations were launched on a supercomputer with similar number of physical nodes and hyper-threads. Therefore, from the perspective of this study OpenMC has proven to be more efficient, in terms of time management, than MCNP.

The other major work of this project was to generate homogenized group constants for a full core calculation that were applied into a nodal transport code. This was a straightforward process since OpenMC is one of the rare Monte Carlo codes that has the capability to produce multi-group cross-sections (be reminded that MCNP does not have this capability, specifically for scattering matrices). Besides, a novel approach implemented in OpenMC, i.e. using angular flux to calculate the reaction constants, has a significant benefit in terms of accuracy in the case of homogenized three-dimensional full-scale reactors. Two plots of a scattering cross-section matrix at different locations were shown as to exemplify the scattering cross-sections differences in different material compositions.

On the other hand, a significant eigenvalue bias of 489 pcm was observed between the continuous energy and group collapsed Monte Carlo models. The fact that this substantial difference existed even though the angular characteristics were taken into consideration, needs further investigation on the intra-assembly level leakage consideration and boundary condition biases of the simulation setup. For instance, higher order Legendre polynomial of the nodal scattering matrices can be introduced to reproduce more adequately the true physical behavior of the angular flux at the boundary.

Even though, the homogenized cross-sections generated by OpenMC do not represent the actual physical phenomena, they were plugged into a deterministic code to compare the outputs. The eigen value difference between multi-group homogenous OpenMC model and the PHISICS model was 171 pcm. In contrary, a satisfactory result was obtained from the energy integrated flux comparisons. The maximum energy integrated flux relative difference of the two codes, from the localized domains where the sample calculations carried-out was found to be about 0.8%. Both results are anticipated from the reactor physics point of view, as the differences may arise from statistical variances or approximation biases. Therefore, based on the aforementioned results, it can be said that in this particular case the INSTANT transport solver of PHISICS toolkit has been verified to be within acceptable margin in comparison to its counterpart MGMC model of OpenMC.

As a concluding remark, the writer would like to suggest a future development on eigen value bias correction methods such as super homogenization to minimize the gap between multi-group homogenous OpenMC model and the deterministic model. In addition, a higher order anisotropic flux resolution method could resolve the MGMC flux discrepancies near the top and bottom edges. Furthermore, the potential of this project to evolve into time dependent interaction characteristics analysis could be considered as the next stage.

9 SUMMARY

MYRRHA is a Belgian research reactor, currently in design and safety analysis stage. The project is funded by the Belgian government and its development is spearheaded by the Belgian Nuclear Research Center, SCK•CEN. Upon its completion, the reactor is mainly intended for irradiation purposes and transmutation of high-level waste. Which means that it has to have a capability to operate in both critical and subcritical conditions separately. Hence, it is equipped with an accelerator driven system in case of subcritical functionalities.

Moreover, its purpose drives the choice of MOX fuel with 30 wt.% enrichment at BOL, which will be followed by batches of various enrichment levels at BOC. The reactor is designed to operate at a fast spectrum with thermal regions allocated for material irradiation and cooled by lead-bismuth eutectic. The fact that it has a highly heterogenous configuration was seen as an opportunity to test a niche Monte Carlo reactor physics code, OpenMC.

In this study, OpenMC was used to model and homogenize the MYRRHA-1.6 configuration and generate the multi-group nodal macroscopic cross sections. In addition, the following comparisons were carried-out.

- Continuous energy heterogenous OpenMC versus MCNP.
- Multi-group Monte Carlo (MGMC) OpenMC versus deterministic code PHISICS.
- Heterogenous OpenMC versus MGMC OpenMC.

A comparison between OpenMC and MCNP showed a satisfactory match both in integrated flux distribution and fission rate especially in the assembly region. However, comparisons that involve the MGMC model were observed to have multiplication factors far from their counterparts, even though the relative integrated flux differences are within acceptable range. Lastly, these discrepancies have triggered recommendations, such as correction of multi-group constants and higher order angular dependency resolution.

REFERENCES

Abderrahim, et al., 2012. MYRRHA - A Multi-purpose Fast Spectrum Research Reactor. Energy Conversion and Management, p. 7.

Banerjee, K., 2010. KERNEL DENSITY ESTIMATOR METHODS FOR MONTE CARLO RADIATION TRANSPOR, s.l.: s.n.

Bowman, C. D., Carlson, A. D., Liskien, H. O. & Stewart, L., 1977. Neutron Standards and Applications. Gaithersburg, National Bureau of Standards.

Boyd, et al., 2019. Multigroup Cross-Section Generation with the OpenMC Monte Carlo Particle Transport Code. American Nuclear Society, p. 17.

Brown, F., 2000. On the Use of Shannon Entropy of the Fission Distribution for Assessing Convergence of Monte Carlo Criticality Calculations. PHYSOR.

Cai, L., 2014. Condensation and homogenization of cross sections for the deterministic transport codes with Monte Carlo method: Application to the GEN IV fast neutron reactors , Paris: Université Paris Sud.

Conlin, J., L. & Romano, P., n.d. A Compact ENDF (ACE) Format Specification, s.l.: s.n.

Eynde, G. V. et al., 2015. An Updated Core Design for the Multi-purpose Irradiation Facility MYRRHA. Journal of Nuclear science and Technology, 52(7-8), pp. 1053-1057.

Hall, A., 2015. Homogenization Methods for Full Core Solution of the Pn Transport Equations with 3-D Cross Sections , s.l.: university of Michigan.

Hebert, A., 2016. Applied Reactor physics. 2nd ed. Quebec: Presses internationales Polytechnique.

Howell, J., 2011. Themopedia. [Online] Available at: http://www.thermopedia.com/content/66/ [Accessed 20 August 2019].

IEA Report, G. E. &. C. S., 2019. The latest trends in energy and emissions in 2018, s.l.: IEA.

Kaltiaisenaho, T., 2014. Statistical Tests and the Underestimation of Variance in Serpent 2, Espoo: VTT.

Lamarsh, j. & Baratta, A., 2014. Introduction to Nuclear Engineering. 3rd ed. Harlow: Pearson Education Ltd..

Leppanen, J., Pusa, M. & Fridman, E., 2016. Overview of methodology for spatial homogenization in the Serpent 2 Monte Carlo code. Annals of Nuclear Energy, pp. 126-136.

Leppänen, J., 2007. Development of a New Monte Carlo Reactor Physics Code, D.Sc. Thesis: Helsinki University.

MacFarlane, E., R., Kahler & C., A., 2010. Methods for Processing ENDF/B-VII with NJOY. s.l.: Elsevier Inc..

Malambu, E. & Stankovskiy, A., 2014. Revised Core Design for MYRRHA-Rev1.6, s.l.: SCK-CEN.

Mbala, M., E., Stankovskiy & A., 2014. Revised Core Design for MYRRHA-Rev1.6, s.l.: SCK-CEN.

Pirouzmand, A. & Mohammadhasani, F., 2015. Multi-Group Neutron Cross Sections and Scattering Matrix Calculations Via Monte Carlo Method. Journal of Polytechnic, 18(4), pp. 235-240.

Ragheb, M., 2013. Estimation Methods in Monte Carlo Particle Transport Simulations. [Online]
Available at: https://mragheb.com/NPRE%20498MC%20Monte%20Carlo%20Simulations%20in%20Engineering/Estimation%20Methods%20in%20Monte%20Carlo%20Particle%20Transport%20Simulations.pdf
[Accessed 4 Augest 2019].

Rimpault, G., Plisson, D., Tommasi, J. & Jacqmin, R., 2002. THE ERANOS CODE AND DATA SYSTEM FOR FAST REACTOR NEUTRONIC ANALYSES. Physor.

Romano, et al., 2014. OpenMC: A State-of-the-art Monte Carlo code for research and development. Annals of Nuclear Energy, Issue 82, pp. 90-97.

Romano, P., F. & B., 2012. The OpenMC Monte Carlo particle transport code. Annals of Nuclear Energy, pp. 274-281.

Romano, P., 2018. The OpenMC Monte Carlo Code. [Online] Available at: https://docs.openmc.org/en/stable/index.html [Accessed 3 April 2019].

Smith, K., 1986. ASSEMBLY HOMOGENIZATION TECHNIQUES FOR LIGHT WATER REACTOR ANALYSIS. Prooress in Nuclear Eneroy, 17(3), pp. 303-335.

Solis, A., 2018. An introduction to OpenMC: A MYYRRHA cell case study, Mol: SCK-CEN.

Trkov, A., Herman, M. & Brown, A., 2018. ENDF-6 Formats Manual. [Online] Available at: https://www.bnl.gov/isd/documents/70393.pdf [Accessed 6 Sept. 2019].

Wang, K. & Pan, Q., 2019. One-step Monte Carlo global homogenization based on RMC code. Nuclear Engineering and Technology, pp. 1209-1217.

Wu, Y., 2019. Neutronics of Advanced Nuclear Systems. Anhui: Springer Nature Singapore Pte Ltd..

Yamamoto, A., Kitamura, Y. & Yamane, Y., 2008. Simplified Treatments of Anisotropic Scattering in LWR Core Calculations. Journal of Nuclear Science and Technology, 45(3), pp. 217-229.

Zohuri, B., 2019. Neutronic Analysis For Nuclear Reactor Systems. 2nd ed. Cham: Springer Nature Switzerland AG .