Reactor Agnostic Multi-Group Cross Section Generation for Fine-Mesh Deterministic Neutron Transport Simulations

by

William Robert Dawson Boyd III
B.S., Georgia Institute of Technology (2010)
M.S., Massachusetts Institute of Technology (2014)

Submitted to the Department of Nuclear Science and Engineering
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Nuclear Science and Engineering
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
February 2017

Author ..... Signature redacted .....
Department of Nuclear Science and Engineering
November 4, 2016

Certified by .. Signature redacted .....
Kord Smith
KEPCO Professor of the Practice of Nuclear Science and Engineering
Thesis Supervisor

Certified by ..... Signature redacted .....
Benoit Forget
Associate Professor of Nuclear Science and Engineering
Thesis Supervisor

Accepted by .... Signature redacted .....
Ju Li
Battelle Energy Alliance Professor of Nuclear Science and Engineering
Professor of Materials Science and Engineering
Chair, Committee on Graduate Students

Reactor Agnostic Multi-Group Cross Section Generation for Fine-Mesh Deterministic Neutron Transport Simulations

by

William Robert Dawson Boyd III

Submitted to the Department of Nuclear Science and Engineering
on November 4, 2016, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Nuclear Science and Engineering

Abstract

A key challenge for full-core transport methods is reactor agnostic multi-group cross section (MGXS) generation. Monte Carlo (MC) presents the most accurate method for MGXS generation since it does not require any approximations to the neutron flux. This thesis develops novel methods that use MC to generate the fine-spatial mesh MGXS that are needed by high-fidelity transport codes. These methods employ either engineering-based or statistical clustering algorithms to accelerate the convergence of MGXS tallied on fine, heterogeneous spatial meshes by Monte Carlo.

The traditional multi-level approach to MGXS generation is replaced by full-core MC calculations that generate MGXS for multi-group deterministic transport codes. Two pin-wise spatial homogenization schemes are introduced to model the clustering of pin-wise MGXS due to spatial self-shielding spectral effects. The Local Neighbor Symmetry (LNS) scheme uses a nearest neighbor-like analysis of a reactor geometry to determine which fuel pins should be assigned the same MGXS. The inferential MGXS (*i*MGXS) scheme applies unsupervised machine learning algorithms to "noisy" MC tally data to identify clustering of pin-wise MGXS without any knowledge of the reactor geometry. Both schemes simultaneously account for spatial self-shielding effects while also accelerating the convergence of the MC tallies used to generate MGXS.

The LNS and *i*MGXS schemes were used to model MGXS clustering from radial geometric heterogeneities in a suite of 2D PWR benchmarks. Both schemes reduced U-238 capture rate errors by up to a factor of four with respect to schemes which neglect to model MGXS clustering. In addition, the schemes required an order of magnitude fewer MC particle histories to converge MGXS for multi-group deterministic calculations than a reference MC calculation. These results demonstrate the potential for single-step MC simulations of the complete heterogeneous geometry as a means to generate reactor agnostic MGXS for deterministic transport codes. The LNS and *i*MGXS schemes may be valuable for reactor physics analyses of advanced LWR core designs and next generation reactors with spatial heterogeneities that are poorly modeled by the engineering approximations in today's methods for MGXS generation.

Thesis Supervisor: Kord Smith
Title: KEPCO Professor of the Practice of Nuclear Science and Engineering

Thesis Supervisor: Benoit Forget
Title: Associate Professor of Nuclear Science and Engineering

Acknowledgments

This thesis was based upon work supported by the U.S. Department of Energy Office of Science Advanced Scientific Computing Research's Center for Exascale Simulation of Advanced Reactors (CESAR) under Contract DE-AC02-06CH11357. The author was additionally supported by the Idaho National Laboratory and the National Science Foundation Graduate Research Fellowship Grant No. 1122374. This research made use of the resources of the High Performance Computing Center at Idaho National Laboratory, which is supported by the Office of Nuclear Energy of the U.S. Department of Energy and the Nuclear Science User Facilities under Contract No. DE-AC07-05ID14517.

I would first like to thank my thesis co-advisors Professors Kord Smith and Benoit Forget for empowering me with the intellectual freedom to explore a novel topic for my thesis research. The encouragement they provided during our numerous enlightening discussions helped me surmount the challenging obstacles that I faced with this thesis research. I am grateful for their personal and professional guidance which has instilled in me the values of hard work and integrity, as well as a practical view on how to effect the change I seek to make in the world. I look forward to maintaining a collegial friendship with each of them as I embark upon the next chapter in my life.

I am incredibly thankful for my former and current peers within the Computational Reactor Physics Group (CRPG) with whom I have developed deep friendships that will outlast my time at MIT. I am especially grateful for Samuel Shaner, my roommate, officemate, collaborator and friend with whom I created the OpenMOC code. I am thankful for my productive collaboration with Paul Romano to develop a Python interface, including a module for multi-group cross section generation, for the OpenMC code. I am appreciative of the many fruitful conversations that I shared with Nathan Gibson to diagnose the bias induced by the flux separability approximation in multi-group cross section generation. Adam Nelson was gracious to correspond with me by email to share his experience in generating multi-group cross sections with OpenMC for use in the MPACT code. Finally, my collaborations with many other fellow students, including Nicholas Horelik, Bryan Herman, Jon Walsh, Derek Lax, Sterling Harper, Logan Abel,

Matthew Ellis, Geoffrey Gunow, Lulu Li, Derek Gaston and John Tramm made it fun and exciting to contribute to open source code projects as a member of CRPG.

I am grateful for my loving parents, Bill and Gail Boyd, who have tirelessly supported each and every opportunity I have sought out in my educational career. I cannot thank them enough for the sacrifices that they have made throughout their lives to enable me to pursue my dreams. Lastly, I am incredibly thankful for my girlfriend Frances Chiang and her love and patience during the five and a half years of late nights and working weekends that I have spent working on this thesis. Her calm and steady hand helped me remain focused on both our short- and long-term goals, and I am incredibly excited to continue our partnership together in the coming years.

Table of Contents

List of Figures

List of Tables

List of Algorithms

Definitions and Acronyms

ASCII	American Standard Code for Information Interchange
API	Application Programming Interface
BC	Boundary Condition
BEAVRS	Benchmark for Evaluation and Validation of Reactor Simulations
BP	Burnable Poison
BWR	Boiling Water Reactor
BFS	Breadth-First Search
CG	Combinatorial Geometry
CRGT	Control Rod Guide Tube
CMFD	Coarse Mesh Finite Difference
FSR	Flat Source Region
GMM	Gaussian Mixture Model
HDF5	Hierarchical Data Format 5
HPC	High-Performance Computing
LNS	Local Neighbor Symmetry
LWR	Light Water Reactor
MC	Monte Carlo
MGXS	Multi-Group Cross Sections
MOC	Method of Characteristics
pcm	per cent mille
PWR	Pressurized Water Reactor
SPH	SuPerHomogénéisation
Q-Q	Quantile-Quantile
XML	eXtensible Markup Language

Part I

Introduction

Chapter 1

Introduction

1.1 Motivation

Numerical simulation has long played an important role in nuclear reactor physics and engineering. The nuclear industry relies on computational modeling of the neutron physics in reactors to predict core reactivity, power distributions, fuel depletion, and transient behavior to ensure the safety and reliability of the current fleet of Light Water Reactors (LWRs). Predictive simulations are necessary to evaluate innovations which seek to improve reactor safety and fuel cycle economics, such as reduced safety margins, accident-tolerant fuels, and extended cycle lengths. In addition, simulation is used to assess the technical competencies of advanced reactor technologies such as Small Modular Reactors (SMRs), Sodium Fast Reactors (SFRs), Molten Salt Reactors (MSRs), High Temperature Gas Reactors (HTGRs), among other proposed designs.

Many Generation III+ reactors, such as the Westinghouse AP1000™Pressurized Water Reactor (PWR), optimize performance with complicated core designs. A variety of reactivity control mechanisms – including partially-inserted control rods, "grey" control poisons, Integral Fuel Burnable Absorbers (IFBA), soluble boron, etc. – along with axial enrichment zoning are used to improve performance metrics such as power peaking factors. The reactor analysis methods in widespread use today assume a "smoothly" varying flux distribution, and are not well-suited to model highly localized flux gradients which result from these complex core configurations. New high-fidelity simulation tools

are needed to accurately capture neutron physics in advanced reactor designs.

The development and deployment of neutron physics simulations is governed by tradeoffs between accuracy and speed. High-fidelity simulations are accurate and flexible since they make few approximations, but require significant computational time and resources. On the other hand, the assumptions and approximations made by low-fidelity methods reduce the number of variables which greatly improves the time-to-solution. However, low-fidelity models are designed for specific applications and lose their predictive power when employed in settings outside of the scope for which they were intended. As a result, it is common to employ a mix of high- and low-fidelity tools for reactor analysis – for example, high-fidelity tools are frequently used to inform and benchmark low-fidelity models for use within a narrow envelope of design parameters. This thesis develops a new approach within the same vein by employing continuous energy Monte Carlo neutron transport simulations to generate accurate multi-group cross sections for computationally efficient fine-mesh deterministic transport methods.

1.2 Background

A key trend in recent years has been the steady progress towards full-core neutron transport-based reactor analysis tools. The standard methods used for reactor analysis today continue to be based on diffusion theory, which enables orders of magnitude computational performance improvements with respect to transport methods. Diffusion-based methods coupled with accurately modeled cross section data have proven to be sufficiently accurate for software tools used by reactor analysts, designers and regulators in industry and academia. However, these techniques rely on a number of assumptions and approximations which are not valid for all reactor types. For example, some Generation IV reactor design concepts are significantly more challenging to model than Light Water Reactor (LWR) designs due to the high degree of spectral coupling between geometrically disparate zones within the core. Although the computational requirements for full-core transport-based simulations have precluded their widespread deployment, the continuing growth of cheap parallel processing power has made the prospects for

such tools increasingly feasible.

Monte Carlo (MC) particle transport methods are often looked to as the "gold standard" for the future of nuclear reactor core depletion calculations. Monte Carlo methods are by their very nature *reactor agnostic* in that they permit an accurate treatment of the core geometry and spectral coupling. Another appealing characteristic of MC methods is their ability to use continuous energy cross sections. An accurate treatment of evaluated cross section data permits high-fidelity core spectral calculations. Although new scalable parallel algorithms have enabled codes to achieve excellent scaling on 100,000s of cores, production tools for industrial workstations remain out of reach for the foreseeable future. The primary reason for this is that the inverse square root convergence rate inherent to MC makes it computationally intractable to realize an acceptably low uncertainty for each tally of interest – except on large supercomputers. Furthermore, full-core Monte Carlo calculations require a terabyte of memory or more to store the tallied quantities, which is inaccessible except on the world's largest computing machines. Finally, the accurate energy treatment largely renders MC methods inefficient for modern computational hardware. The stochastic treatment of the nonlinear neutron energy characterization is challenging to vectorize and results in highly disjoint memory accesses to cross section data with minimal cache reuse.

An attractive alternative to MC are deterministic methods – such as Discrete Ordinates (S_N), Simplified P_N (SP_N), and the Method of Characteristics (MOC). Deterministic methods typically do not make use of continuous energy cross section data and instead discretize the energy domain through the multi-group energy approximation as shown in Fig. 1-1. The multi-group approximation considerably reduces the necessary dataset footprint for simulation. In addition, the multi-group approximation enables deterministic methods to be more effectively formulated for vectorization and cache reuse than MC methods. However, the approximation requires an *a priori* estimate of the neutron flux to compute the multi-group cross sections (MGXS) in each energy group and spatial zone in order to solve for the flux distribution throughout the core.

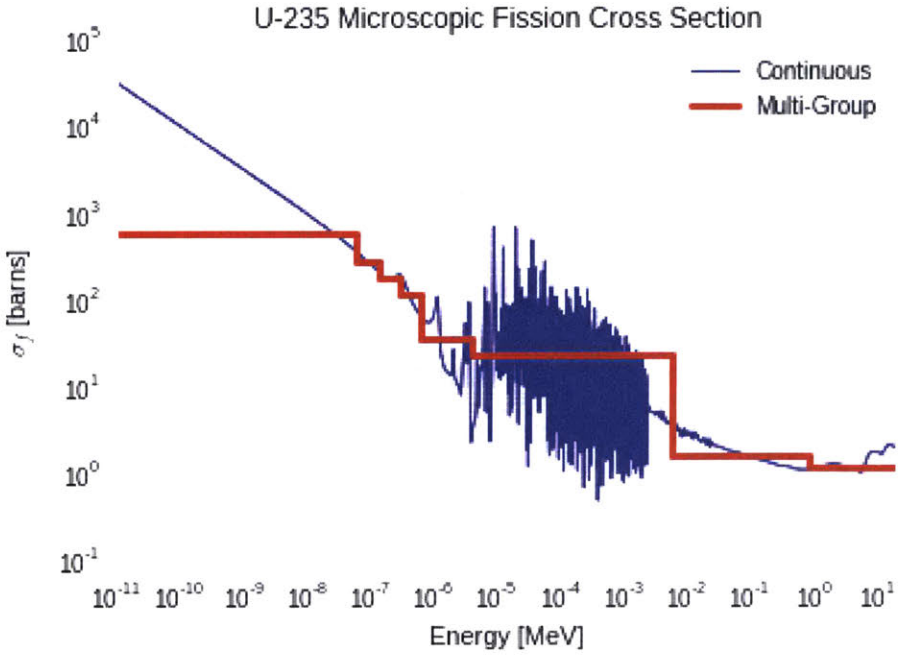


Figure 1-1: U-235 continuous energy and 16-group fission cross section.

This thesis is motivated by the desire to obtain Monte Carlo-quality solutions with computationally efficient deterministic neutron transport methods.

Many different engineering prescriptions have been developed to generate MGXS for specific reactor configurations and spectra. In general, MGXS generation schemes use a multi-level approach to decouple the energy, angular and spatial dimensions as depicted in Fig. 1-2. The multi-level approach typically applies high-fidelity models of the energy self-shielding physics to low-fidelity geometric models of unique core components. The complexity of the energy treatment is then reduced at each level as larger and more complex geometric models are considered.

For example, the first stage for LWR MGXS generation attempts to capture energy self-shielding effects within simplified geometric models such as infinite fuel pin cells. This step typically condenses continuous energy cross sections to $\mathcal{O}(100)$ groups. These MGXS are then used in a heterogeneous lattice physics calculation of an individual fuel assembly within an infinite lattice. The lattice physics calculation models spatial self-shielding effects between pins of various material compositions and condenses the

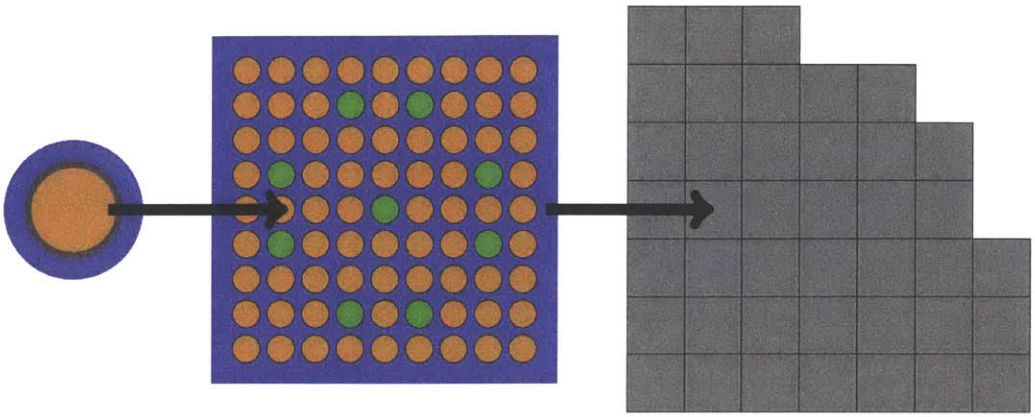


Figure 1-2: Current multi-level framework for reactor analysis.

MGXS to a coarse energy structure of $\mathcal{O}(10)$ groups. In addition, the MGXS may be spatially-homogenized across the entire fuel assembly or across each fuel pin within each assembly. Finally, the spatially-homogenized coarse MGXS for each fuel assembly are used in a full-core calculation composed of many fuel assemblies.

The multi-level approach uses a combination of models of varying complexity to optimize overall simulation speed with accuracy. However, this is typically done at the expense of generality. For example, some prior knowledge of the neutron energy spectra is required to design approximations to the flux for a particular reactor configuration. Furthermore, multi-level MGXS generation schemes do not generally model inter-assembly physics or the effect of reflectors and other core heterogeneities on the spatial distribution of the flux. Instead, geometric heuristics are often used to embed spatial self-shielding effects in MGXS for similarly shielded spatial zones (*e.g.*, fuel pins with similar neighboring pins). The approximations to the energy and spatial variation of the flux introduce approximation error in full-core calculations and limit the core design parameter space for which multi-level schemes may be applied. New reactor agnostic MGXS generation methods are needed to enable deterministic transport-based methods to be as accurate and flexible as Monte Carlo in full-core calculations.

This thesis investigates the use of Monte Carlo methods to generate MGXS for full-core deterministic reactor analysis. Monte Carlo presents a natural approach to replace engineering prescriptions to approximate the flux with a stochastic approximation of the exact flux. The advantage of a MC-based approach is that all of the relevant

physics modeled in MC may be directly embedded into MGXS. This improvement in accuracy comes at the computational expense of converging group constant tallies to acceptably low uncertainties. MC methods have increasingly been used to generate few group constants for coarse mesh diffusion, most notably by the Serpent MC code [1]. However, there exist few rigorous and comprehensive analyses of MGXS generation for heterogeneous fine-mesh deterministic transport methods.

This thesis develops and evaluates MC-based methods to generate MGXS for fine-mesh deterministic neutron transport codes.

In addition, MC-based MGXS generation methods to date have retained the multi-level geometric framework to tabulate MGXS for individual reactor components – such as infinite fuel pins and/or assemblies – for subsequent use in full-core multi-group calculations. Although the use of MC within a multi-level scheme eliminates the need to approximate the flux in energy, it does not account for spatial self-shielding effects throughout a reactor core. This thesis abandons the multi-level framework in place of a full-core MC calculation which simultaneously accounts for all energy and spatial effects in a single step.

In theory, full-core MC calculations can be used to tally MGXS in each spatial zone (*e.g.*, 100 axial depletion zones within each of 50,000+ fuel pins in a PWR core) to account for the spatial variation of the flux. However, such simulations have not been employed for practical reasons – in particular, the large memory footprint and computational expense of performing such calculations has been prohibitive for MC codes until recent years. Furthermore, roughly the same number of particle histories would be required to converge the MGXS tallies in each spatial zone as would be required for a direct full-core calculation by MC. Hence, it would be more sensible to simply use MC to compute the solution to the full-core eigenvalue problem directly rather than use it to fully embed spatial self-shielding effects in MGXS for deterministic transport codes. Therefore, in order for MC to be practical for reactor agnostic fine-mesh MGXS generation, a new method is required to accelerate the convergence of the MGXS tallies in each fine-mesh region to a degree that is not possible for conventional full-core Monte

Carlo simulations.

This thesis proposes to use statistical clustering methods to accelerate the convergence of full-core MC calculations for MGXS generation. This novel approach relies on the fact that many distinct spatial zones across a reactor core experience similar spatial self-shielding effects, and therefore have similar MGXS. The stochastic nature of MC simulations will contribute statistical "noise" to the tally estimates for the MGXS. As a result, the MGXS estimates for similarly self-shielded spatial zones will form clusters which will converge as more particle histories are simulated. The goal of this thesis is to develop and apply algorithms to identify MGXS clusters from "noisy" Monte Carlo tally data and to predict the true mean of each cluster prior to convergence. This methodology aims to generate MGXS for deterministic neutron transport codes in a reactor agnostic and computationally efficient manner.

This thesis uses statistical clustering algorithms to accelerate full-core MC calculations which simultaneously model all energy and spatial self-shielding effects for fine-mesh MGXS generation in a single step.

1.3 Thesis Objectives

The subject matter of this thesis is organized along two main themes:

- ***Approximation Error*** – Quantify and diagnose approximation error in MGXS generated from MC methods for simple heterogeneous benchmark problems.
- ***Statistical Clustering*** – Develop statistical clustering methods to accelerate the convergence of MGXS on heterogeneous MC tally meshes.

The first theme of this thesis rigorously assesses the efficacy of MGXS generation with MC for fine-mesh transport calculations. Some of the approximations made by MC-based MGXS generation are quantified, including the energy and spatial dependence of condensed MGXS. An in-depth analysis of systematic bias resulting from constant-in-

angle total MGXS is presented, along with a scheme based on SuPerHomogénéisation (SPH) factors to compensate for this loss in accuracy.

The second theme of this thesis develops a new methodology to simultaneously capture local and global spatial self-shielding effects in MGXS for full-core calculations. This scheme applies statistical clustering methods to accelerate the convergence of MGXS tallied on fine, heterogeneous spatial meshes in Monte Carlo. The latent variable model which inspires the clustering paradigm is presented, along with a discussion of the implementation of a data pipeline to evaluate clustering algorithms for MGXS generation. A series of increasingly complex heterogeneous benchmarks are modeled to empirically compare the accuracy and convergence of the approach with more traditional multi-level schemes for MC-based MGXS generation.

1.4 Thesis Outline

This thesis is segmented into five Parts. Part I is comprised of this introductory chapter.

Part II discusses the relevant background information for this thesis. Chap. 2 reviews multi-group neutron transport theory and considers some common approximations made in MGXS generation and multi-group transport codes. Chap. 3 introduces Monte Carlo as an approach to generate MGXS, and highlights relevant studies in the literature which have used MC to generate MGXS. Chap. 4 presents the simulation workflow developed for this thesis to evaluate MC for MGXS generation, including the OpenMC, OpenMOC and OpenCG codes.

Part III diagnoses common sources of approximation error in MGXS generation and multi-group transport methods. Chap. 5 quantifies the impact of multi-group approximation error for simple, heterogeneous PWR geometries. Chap. 6 presents an algorithmic approach to mitigate systematic biases resulting from constant-in-angle total MGXS using SPH factors, and motivates the need for future work to address this issue.

Part IV develops a novel approach based on statistical clustering methods to accelerate full-core MC calculations for MGXS generation. Chap. 7 presents a series of heterogeneous PWR benchmark models along with reference reaction rates and eigenvalues

computed with OpenMC to validate this new methodology. Chap. 8 quantifies the impact of using MGXS which reflect inter-pin and inter-assembly spatial self-shielding effects on the solutions computed by multi-group deterministic transport methods. Chap. 9 illustrates the emergence of MGXS clusters due to spatial self-shielding with a variety of visual aids. Chap. 10 outlines a new methodology which applies unsupervised clustering methods to accelerate the convergence of MGXS tallied with MC. Chap. 11 evaluates the impact of clustered MGXS on the accuracy and convergence of the eigenvalue solutions computed by deterministic transport methods.

Part V summarizes the progress made in this thesis to chart a path forward for MC-based MGXS generation for full-core deterministic transport methods in Chap. 12.

Part II

Background

Chapter 2

Approximations in Multi-Group Transport Theory

This chapter presents an overview of some of the approximations made by methods which solve the multi-group form of the neutron transport equation. This chapter begins by reviewing the continuous energy steady-state neutron transport equation in Sec. 2.1. The following sections present simplifications to the angular (Sec. 2.2), energy (Sec. 2.3) and spatial dependence (Sec. 2.4) of the equation. The approximations are not specific to a particular approach for solving the transport equation and may be employed by either stochastic or deterministic methods. Sec. 2.5 concludes with a discussion of how these approximations present challenges for accurate MGXS generation.

2.1 Background

The field of reactor physics is concerned with computing the distribution of nuclear reaction rates throughout a nuclear reactor core. Nuclear reaction rates are dependent on two fundamental quantities: the density of neutrons and the probability of interaction. The angular neutron flux $\psi(\mathbf{r}, \mathbf{\Omega}, E)$ models the neutron density[1] as the path length traveled by neutrons per unit volume and is dependent on a neutron's spatial position

[1]Unlike the common definition of flux used in other areas of science and engineering, the angular flux ψ is the product of the volume density and speed of neutrons in phase space.

$\mathbf{r}$, direction of motion $\mathbf{\Omega}$ and energy E[2,3]. The macroscopic cross section $\Sigma_x(\mathbf{r}, E)$ is defined as the probability of interaction x per unit of length traveled by a neutron at some position and energy. A reaction rate $\mathcal{R}_x$ can be simply computed as the product of the angular flux and cross section:

$$\mathcal{R}_x(\mathbf{r}, \mathbf{\Omega}, E) = \Sigma_x(\mathbf{r}, E)\psi(\mathbf{r}, \mathbf{\Omega}, E) \tag{2.1}$$

The macroscopic cross section Σ_x is proportional to a quantity known as the microscopic cross section σ_x. The microscopic cross section is a property of a particular nuclide and is measured experimentally for various reaction types x which include fission f, radiative capture γ and scattering s[4]. The macroscopic cross section is the sum of the microscopic cross sections of each nuclide i weighted by its number density N_i:

$$\Sigma_x(\mathbf{r}, E) = \sum_i N_i(\mathbf{r})\sigma_{i,x}(E) \tag{2.2}$$

The microscopic cross section is highly dependent on the energy of the incoming neutron. As illustrated in Fig. 2-1, a cross section varies several orders of magnitude within an energy interval on the order of an eV near nuclear resonances. The probability of some interactions also depends on other properties which characterize the output channel of the reaction. For example, the scattering cross section σ_s depends on the energy and direction of motion of the outgoing neutron. The macroscopic cross section varies in space when nuclide densities depend on the position within a heterogeneous system.

[2]Vector-valued quantities are expressed in boldface font.
[3]This thesis focuses on steady-state calculations and time dependence is neglected for simplicity.
[4]Scattering as defined here includes both inelastic and elastic scattering channels.

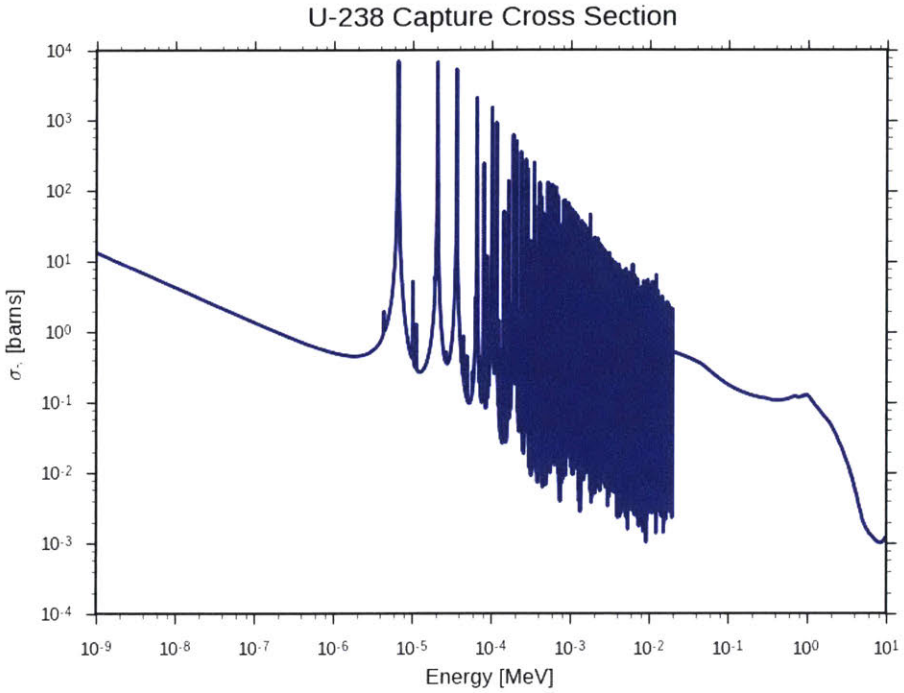


Figure 2-1: The continuous energy capture cross section for U-238.

Although cross sections are experimentally measured, the neutron flux must be calculated analytically or with simulation. The steady-state Boltzmann transport equation [2] is integro-differential in the neutron angular flux $\psi(\mathbf{r}, \mathbf{\Omega}, E)$ and balances the rate of change of the population of neutrons in phase space to the difference between the production and loss rates of neutrons within a closed system:

$$\begin{aligned} &\mathbf{\Omega} \cdot \nabla \psi(\mathbf{r}, \mathbf{\Omega}, E) + \Sigma_t(\mathbf{r}, E)\psi(\mathbf{r}, \mathbf{\Omega}, E) \\ &= \int_0^\infty \int_{4\pi} \Sigma_s(\mathbf{r}, \mathbf{\Omega}' \rightarrow \mathbf{\Omega}, E' \rightarrow E)\psi(\mathbf{r}, \mathbf{\Omega}', E')\mathrm{d}\mathbf{\Omega}'\mathrm{d}E' + Q(\mathbf{r}, \mathbf{\Omega}, E) \end{aligned} \tag{2.3}$$

The first term on the left hand side of the equation represents the streaming of neutrons within space and the second term is the total neutron collision rate determined by the total cross section Σ_t. On the right hand side, the first term models the scattering of neutrons at some energy E' and direction $\mathbf{\Omega}'$ into energy E and direction $\mathbf{\Omega}$. The final term represents a generic source Q of neutrons. In the case of critical systems, such as nuclear reactors, Q is a source of fission neutrons:

$$Q(\mathbf{r}, \mathbf{\Omega}, E) = \frac{1}{k_{eff}} \int_0^\infty \int_{4\pi} \nu\Sigma_f(\mathbf{r}, \mathbf{\Omega}' \rightarrow \mathbf{\Omega}, E' \rightarrow E)\psi(\mathbf{r}, \mathbf{\Omega}', E')\mathrm{d}\mathbf{\Omega}'\mathrm{d}E' \tag{2.4}$$

The fission production cross section $\nu\Sigma_f$ represents the probability of neutrons emitted at energy E and angle Ω resulting from fission events precipitated by neutrons at E' and Ω'. The eigenvalue k_{eff} of a critical system represents the multiplication of neutrons from fission and forces balance between neutron sources and losses due to absorption and leakage.

A solution for the neutron flux must computed from the transport equation in order to compute reaction rate distributions. The accurate determination of the neutron flux is primarily challenged by the complicated energy structure of the cross sections. In addition, the distribution of neutrons in LWRs spans 11 orders of magnitude from a few MeV at birth from fission emission to death by absorption at energies as low as 10^{-5} eV. As a result, analytical solutions to Eqn. 2.3 are intractable without significant simplifying assumptions.

Instead, numerical simulation is used to solve the transport equation for the flux. Monte Carlo may be employed to exactly treat the energy dependence in Eqn. 2.3[5], but it is computationally burdensome and impractical for routine nuclear reactor analysis. Although space and angle may be discretized using standard techniques for the solution of partial differential equations, special treatment must be given to the energy variable. The following sections introduce approximations used to reduce the dimensionality of the equation to permit tractable multi-group calculations.

Nuclear reactor simulations calculate the neutron multiplication factor k_{eff} and reaction rate spatial distributions. Monte Carlo methods are the most accurate approach, but are not yet practical for full-core analysis.

[5]The treatment is only as exact as the uncertainties in measured nuclear cross section data will permit.

2.2 Approximations in Angle

2.2.1 Isotropic Fission Source

The neutrons emitted from fission form a nearly isotropic distribution independent of the energy or angle of the incoming neutron[6]. As a result, the fission production cross section can be approximated as $\nu\Sigma_f(\mathbf{r},\mathbf{\Omega}'\rightarrow\mathbf{\Omega},E'\rightarrow E)\approx\nu\Sigma_f(\mathbf{r},E'\rightarrow E)$. This permits the fission source in Eqn. 2.4 to be written as:

$$Q(\mathbf{r},\mathbf{\Omega},E)=\frac{1}{4\pi k_{eff}}\int_0^\infty\int_{4\pi}\nu\Sigma_f(\mathbf{r},E'\rightarrow E)\psi(\mathbf{r},\mathbf{\Omega},E')\mathrm{d}\mathbf{\Omega}\mathrm{d}E' \tag{2.5}$$

This expression may be further simplified in terms of the scalar neutron flux $\phi(\mathbf{r},E)$:

$$Q(\mathbf{r},\mathbf{\Omega},E)=\frac{1}{4\pi k_{eff}}\int_0^\infty\nu\Sigma_f(\mathbf{r},E'\rightarrow E)\phi(\mathbf{r},E')\mathrm{d}E' \tag{2.6}$$

$$\phi_g(\mathbf{r},E)=\int_{4\pi}\psi(\mathbf{r},\mathbf{\Omega},E)\mathrm{d}\mathbf{\Omega} \tag{2.7}$$

The isotropic approximation reduces the dimensionality of the fission production term in the transport equation and simplifies the derivation of the approximations in the following sections.

2.2.2 Angular Expansion of the Scattering Kernel

Unlike the fission source, the source of neutrons from scattering cannot be treated as isotropic since it is strongly dependent on the relationship between the incoming and outgoing directions of motion. The dimensionality of the scattering source term in the transport equation – known as the double differential scattering kernel – is commonly reduced with basis function expansions in angle [3,4]. The angular flux is first expanded as an infinite sum of spherical harmonic functions $Y_\ell^m(\mathbf{\Omega})$ and angular flux moments $\psi_\ell^m(\mathbf{r},E)$:

[6]This approximation is only valid for a large number of fission events as is the case in a nuclear reactor.

$$\psi(\mathbf{r},\mathbf{\Omega},E)=\sum_{\ell=0}^{\infty}\frac{2\ell+1}{4\pi}\sum_{m=-\ell}^{\ell}\psi_{\ell}^{m}(\mathbf{r},E)Y_{l}^{m}(\mathbf{\Omega}) \tag{2.8}$$

$$\psi_{\ell}^{m}(\mathbf{r},E)=\int_{4\pi}\psi(\mathbf{r},\mathbf{\Omega},E)Y_{\ell}^{m}(\mathbf{\Omega})\mathrm{d}\mathbf{\Omega} \tag{2.9}$$

Similarly, the angular dependence of the scattering cross section $\Sigma_s(\mathbf{r},\mathbf{\Omega}'\rightarrow\mathbf{\Omega},E'\rightarrow E)$ can be treated with a basis function expansion. The scattering cross section can be simplified without approximation by noting that the distribution over the change in direction μ is independent of the incoming angle $\mathbf{\Omega}'$ in isotropic media. The re-parametrized scattering cross section $\Sigma_s(\mathbf{r},\mu,E'\rightarrow E)$ can then be expanded as an infinite sum of Legendre polynomials $P_\ell(\mu)$ and scattering moments $\Sigma_{s,\ell}(\mathbf{r},E'\rightarrow E)$:

$$\Sigma_s(\mathbf{r},\mu,E'\rightarrow E)=\sum_{\ell=0}^{\infty}\frac{2\ell+1}{2}\Sigma_{s,\ell}(\mathbf{r},E'\rightarrow E)P_{\ell}(\mu) \tag{2.10}$$

$$\Sigma_{s,\ell}(\mathbf{r},E'\rightarrow E)=\int_{-1}^{1}\Sigma_s(\mathbf{r},\mu,E'\rightarrow E)P_{\ell}(\mu)\mathrm{d}\mu \tag{2.11}$$

The expansions of the angular flux in spherical harmonics and the scattering cross section in Legendre polynomials may be substituted into the scattering kernel. The spherical harmonic addition theorem can be applied to simplify the kernel in terms of only the real components $R_\ell^m(\mathbf{\Omega})$ of the spherical harmonics:

$$\begin{aligned}&\int_0^{\infty}\int_{4\pi}\Sigma_s(\mathbf{r},\mathbf{\Omega}'\rightarrow\mathbf{\Omega},E'\rightarrow E)\psi(\mathbf{r},\mathbf{\Omega}',E')\mathrm{d}\mathbf{\Omega}'\mathrm{d}E'\\&=\int_0^{\infty}\sum_{\ell=0}^{\infty}\frac{2\ell+1}{4\pi}\Sigma_{s,\ell}(\mathbf{r},E'\rightarrow E)\psi_{\ell}^{m}(\mathbf{r},E')R_{\ell}^{m}(\mathbf{\Omega})\mathrm{d}E'\end{aligned} \tag{2.12}$$

No approximation has been made to the scattering kernel's angular dependence in Eqn. 2.12. In practice, however, the expansion is truncated to a finite number of spherical harmonics L to make the transport equation computationally tractable. The transport equation with the scattering source expansion and isotropic fission source is then:

$$\begin{aligned}
&\mathbf{\Omega}\cdot\nabla\psi(\mathbf{r},\mathbf{\Omega},E)+\Sigma_t(\mathbf{r},E)\psi(\mathbf{r},\mathbf{\Omega},E)\\
&=\int_0^\infty\sum_{\ell=0}^{L}\frac{2\ell+1}{4\pi}\sum_{m=-\ell}^{\ell}\Sigma_{s,\ell}(\mathbf{r},E'\rightarrow E)\psi_\ell^m(\mathbf{r},E')R_\ell^m(\mathbf{\Omega})\mathrm{d}E'\\
&\quad+\frac{1}{4\pi k_{eff}}\int_0^\infty\nu\Sigma_f(\mathbf{r},E'\rightarrow E)\phi(\mathbf{r},E')\mathrm{d}E'
\end{aligned}\tag{2.13}$$

2.2.3 Transport Correction

The scattering matrix and flux moments substantially increase the memory and storage requirements for calculation schemes which model anisotropic scattering with a finite moment expansion as introduced in Sec. 2.2.2. In practice, many methods attempt to implicitly model anisotropic scattering effects with transport-corrected cross sections. These approaches seek to define a correction which makes the transport equation with isotropic scattering in the laboratory system ($L=0$) equivalent to the general equation with anisotropic scattering. Various transport corrections are thoroughly detailed in the TRANSX [5] and NJOY [6] manuals, each of which follows the approach taken by Bell, Hansen and Sandmeier in [7]. This section summarizes the derivation in [3].

First, consider a truncated form of the Legendre polynomial expansion of the scattering cross section $\Sigma_s(\mathbf{r},\mu,E'\rightarrow E)$ in Eqn. 2.10:

$$\begin{aligned}
\Sigma_s(\mathbf{r},\mu,E'\rightarrow E)&=\sum_{\ell=0}^{\infty}\frac{2\ell+1}{2}\Sigma_{s,\ell}(\mathbf{r},E'\rightarrow E)P_\ell(\mu)\\
&\approx\sum_{\ell=0}^{L}\frac{2\ell+1}{2}\tilde{\Sigma}_{s,\ell}(\mathbf{r},E'\rightarrow E)P_\ell(\mu)+\Delta\Sigma_{tr}(\mathbf{r},E'\rightarrow E)\delta(\mu-1)
\end{aligned}\tag{2.14}$$

where $\tilde{\Sigma}_{s,\ell}$ is a modified form of the scattering moment $\Sigma_{s,\ell}$ and $\Delta\Sigma_{tr}$ is a transport correction term. The Kronecker delta function $\delta(\mu-1)$ is used to make the correction term forward peaked in order to best capture the first order anisotropies in thermal reactors. The coefficients $\tilde{\Sigma}_{s,\ell}$ and $\Delta\Sigma_{tr}$ are defined such that the Legendre moments of the scattering cross section in Eqn. 2.11 are preserved for $0\leq\ell\leq L+1$:

$$\Sigma_{s,\ell}(\mathbf{r}, E' \rightarrow E) = \int_{-1}^{1} \Sigma_s(\mathbf{r}, \mu, E \rightarrow E) P_\ell(\mu) \mathrm{d}\mu$$

$$\approx \int_{-1}^{1} \sum_{\ell'=0}^{L} \frac{2\ell'+1}{2} \tilde{\Sigma}_{s,\ell'}(\mathbf{r}, E' \rightarrow E) P_{\ell'}(\mu) P_\ell(\mu) \mathrm{d}\mu \qquad (2.15)$$

$$+ \int_{-1}^{1} \Delta\Sigma_{tr}(\mathbf{r}, E' \rightarrow E) \delta(\mu - 1) P_\ell(\mu) \mathrm{d}\mu$$

The following simultaneous system of equalities for $0 \leq \ell \leq L$ follows from the identity $P_\ell(1) = 1$ and the orthogonality relation of the Legendre polynomial basis set:

$$\tilde{\Sigma}_{s,\ell}(\mathbf{r}, E' \rightarrow E) + \Delta\Sigma_{tr}(\mathbf{r}, E' \rightarrow E) = \Sigma_{s,\ell}(\mathbf{r}, E' \rightarrow E) \qquad (2.16)$$

$$\Delta\Sigma_{tr}(\mathbf{r}, E' \rightarrow E) \doteq \Sigma_{s,L+1}(\mathbf{r}, E' \rightarrow E) \qquad (2.17)$$

For isotropic in lab scattering with $L = 0$ the Eqns. 2.14 and 2.17 simplify in terms of only the zeroth and first order scattering moments:

$$\Sigma_s(\mathbf{r}, \mu, E' \rightarrow E) \approx \frac{1}{2}\left[\Sigma_{s,0}(\mathbf{r}, E' \rightarrow E) - \Sigma_{s,1}(\mathbf{r}, E' \rightarrow E)\right] + \Sigma_{s,1}(\mathbf{r}, E' \rightarrow E)\delta(\mu - 1) \qquad (2.18)$$

The transport-corrected scattering cross section in Eqn. 2.18 is then substituted into the transport equation in Eqn. 2.13 with an isotropic scattering kernel and rearranged to produce:

$$\mathbf{\Omega} \cdot \nabla\psi(\mathbf{r}, \mathbf{\Omega}, E) + \Sigma_t(\mathbf{r}, E)\psi(\mathbf{r}, \mathbf{\Omega}, E) - \int_0^\infty \Sigma_{s,1}(\mathbf{r}, E' \rightarrow E)\phi(\mathbf{r}, E')\mathrm{d}E'$$

$$= \frac{1}{4\pi} \int_0^\infty \left[\Sigma_{s,0}(\mathbf{r}, E' \rightarrow E) - \Sigma_{s,1}(\mathbf{r}, E' \rightarrow E)\right] \phi(\mathbf{r}, E')\mathrm{d}E' \qquad (2.19)$$

$$+ \frac{1}{4\pi k_{eff}} \int_0^\infty \nu\Sigma_f(\mathbf{r}, E' \rightarrow E)\phi(\mathbf{r}, E')\mathrm{d}E'$$

where the relation $\phi = \psi_0^0$ for the scalar flux has been used to completely remove the angular dependence from the isotropic scattering kernel. The transport correction term

$\Delta\Sigma_{tr}(\mathbf{r}, E)$ can be defined such that it can be lumped into new transport-corrected total and scattering cross sections $\tilde{\Sigma}_t$ and $\tilde{\Sigma}_s$ as follows:

$$\Delta\Sigma_{tr}(\mathbf{r}, E) \equiv \frac{\int_0^\infty \Sigma_{s,1}(\mathbf{r}, E' \rightarrow E)\phi(\mathbf{r}, E')\mathrm{d}E'}{\phi(\mathbf{r}, E)} \tag{2.20}$$

$$\tilde{\Sigma}_t(\mathbf{r}, E) = \Sigma_t(\mathbf{r}, E) - \Delta\Sigma_{tr}(\mathbf{r}, E) \tag{2.21}$$

$$\tilde{\Sigma}_s(\mathbf{r}, E' \rightarrow E) = \Sigma_{s,0}(\mathbf{r}, E' \rightarrow E) - \Delta\Sigma_{tr}(\mathbf{r}, E)\delta(E' - E) \tag{2.22}$$

This definition of the transport correction is termed the *in-scatter approximation* in the literature [8]. Finally, the transport equation in Eqn. 2.19 can be simplified with the substitution of the corrected total and scattering cross sections:

$$\begin{aligned}\mathbf{\Omega} \cdot \nabla\psi(\mathbf{r}, \mathbf{\Omega}, E) + \tilde{\Sigma}_t(\mathbf{r}, E)\psi(\mathbf{r}, \mathbf{\Omega}, E) = &\frac{1}{4\pi}\int_0^\infty \tilde{\Sigma}_s(\mathbf{r}, E' \rightarrow E)\phi(\mathbf{r}, E')\mathrm{d}E' \\ &+ \frac{1}{4\pi k_{eff}}\int_0^\infty \nu\Sigma_f(\mathbf{r}, E' \rightarrow E)\phi(\mathbf{r}, E')\mathrm{d}E'\end{aligned} \tag{2.23}$$

The neutron fission source is isotropic in reactors. The scattering source is simplified with a basis function expansion in angle. A transport correction is commonly used to account for anisotropic scattering in simulations that simplify the scattering source as isotropic.

2.3 Approximations in Energy

2.3.1 Energy Discretization

The multi-group approach used to solve the transport equation subdivides the neutron's energy into discrete bins known as energy groups. The energy groups are indexed starting at 1 for high energies and ending with G for the lowest energies of interest. An energy group $g \in \{1, 2, \ldots, G\}$ spans a range of energies from $\left[E_g, E_{g-1}\right]$ where E_0 is

the highest energy under consideration and E_g is the upper bound of group g[7]. First, a group-wise angular flux ψ_g and scalar flux ϕ_g is defined for each energy group:

$$\psi_g(\mathbf{r},\mathbf{\Omega}) = \int_{E_g}^{E_{g-1}} \psi(\mathbf{r},\mathbf{\Omega},E)\mathrm{d}E \tag{2.24}$$

$$\phi_g(\mathbf{r}) = \int_{E_g}^{E_{g-1}} \phi(\mathbf{r},E)\mathrm{d}E \tag{2.25}$$

The continuous energy transport equation with isotropic fission and scattering sources and transport-corrected cross sections in Eqn. 2.23 can be transformed into its multi-group form by integrating over each energy group[8]:

$$\begin{aligned}
&\mathbf{\Omega}\cdot\nabla\psi_g(\mathbf{r},\mathbf{\Omega}) + \int_{E_g}^{E_{g-1}} \left[\tilde{\Sigma}_t(\mathbf{r},E)\psi(\mathbf{r},\mathbf{\Omega},E)\right]\mathrm{d}E \\
&= \int_{E_g}^{E_{g-1}} \left[\frac{1}{4\pi}\sum_{g'=1}^{G}\int_{E_{g'}}^{E_{g'-1}} \tilde{\Sigma}_s(\mathbf{r},E'\rightarrow E)\phi(\mathbf{r},E')\mathrm{d}E'\right]\mathrm{d}E \\
&\quad + \int_{E_g}^{E_{g-1}} \left[\frac{1}{4\pi k_{eff}}\sum_{g'=1}^{G}\int_{E_{g'}}^{E_{g'-1}} \nu\Sigma_f(\mathbf{r},E'\rightarrow E)\phi(\mathbf{r},E')\mathrm{d}E'\right]\mathrm{d}E
\end{aligned} \tag{2.26}$$

The integrals over incoming neutron energy in the scattering kernel and fission source in Eqn. 2.26 are treated as summations of discrete integrals over each incoming energy group. Although the streaming term is easily expressed in terms of the multi-group flux ψ_g, the total collision, scattering and fission terms are defined as integral quantities in energy. These three terms can be simplified by multiplying each by unity in the form of ψ_g/ψ_g and ϕ_g/ϕ_g:

[7]This convention derives from the fact that neutrons are emitted at high energies from fission and are absorbed at lesser energies in LWRs.

[8]The multi-group approximation with energy discretization may be similarly applied to the transport equation in Eqn. 2.13 if anisotropic scattering is explicitly treated with scattering moments.

$$\begin{aligned}
&\boldsymbol{\Omega}\cdot\nabla\psi_g(\mathbf{r},\boldsymbol{\Omega})+\left[\frac{\int_{E_g}^{E_{g-1}}\tilde{\Sigma}_t(\mathbf{r},E)\psi(\mathbf{r},\boldsymbol{\Omega},E)\mathrm{d}E}{\psi_g(\mathbf{r},\boldsymbol{\Omega})}\right]\psi_g(\mathbf{r},\boldsymbol{\Omega})\\
&=\frac{1}{4\pi}\sum_{g'=1}^{G}\left[\frac{\int_{E_g}^{E_{g-1}}\int_{E_{g'}}^{E_{g'-1}}\tilde{\Sigma}_s(\mathbf{r},E'\rightarrow E)\phi(\mathbf{r},E')\mathrm{d}E'\mathrm{d}E}{\phi_{g'}(\mathbf{r})}\right]\phi_{g'}(\mathbf{r})\\
&+\frac{1}{4\pi k_{eff}}\sum_{g'=1}^{G}\left[\frac{\int_{E_g}^{E_{g-1}}\int_{E_{g'}}^{E_{g'-1}}\nu\Sigma_f(\mathbf{r},E'\rightarrow E)\phi(\mathbf{r},E')\mathrm{d}E'\mathrm{d}E}{\phi_{g'}(\mathbf{r})}\right]\phi_{g'}(\mathbf{r})
\end{aligned}\tag{2.27}$$

The bracketed terms in Eqn. 2.27 are defined as the MGXS for total, scattering and fission production reactions. The MGXS are the averages of the corresponding continuous energy cross sections weighted by the angular neutron flux ψ in each energy group. The MGXS $\tilde{\Sigma}_{t,g}$, $\tilde{\Sigma}_{s,g'\rightarrow g}$ and $\nu\Sigma_{f,g'\rightarrow g}$ are defined below for completeness:

$$\tilde{\Sigma}_{t,g}(\mathbf{r},\boldsymbol{\Omega})\equiv\frac{\int_{E_g}^{E_{g-1}}\tilde{\Sigma}_t(\mathbf{r},E)\psi(\mathbf{r},\boldsymbol{\Omega},E)\mathrm{d}E}{\psi_g(\mathbf{r},\boldsymbol{\Omega})}\tag{2.28}$$

$$\tilde{\Sigma}_{s,g'\rightarrow g}(\mathbf{r})\equiv\frac{\int_{E_g}^{E_{g-1}}\int_{E_{g'}}^{E_{g'-1}}\tilde{\Sigma}_s(\mathbf{r},E'\rightarrow E)\phi(\mathbf{r},E')\mathrm{d}E'\mathrm{d}E}{\phi_{g'}(\mathbf{r})}\tag{2.29}$$

$$\nu\Sigma_{f,g'\rightarrow g}(\mathbf{r})\equiv\frac{\int_{E_g}^{E_{g-1}}\int_{E_{g'}}^{E_{g'-1}}\nu\Sigma_f(\mathbf{r},E'\rightarrow E)\phi(\mathbf{r},E')\mathrm{d}E'\mathrm{d}E}{\phi_{g'}(\mathbf{r})}\tag{2.30}$$

With these definitions of the MGXS, the multi-group form of the transport equation in Eqn. 2.26 can be expressed succinctly in terms of the group-wise fluxes:

$$\begin{aligned}
\boldsymbol{\Omega}\cdot\nabla\psi_g(\mathbf{r},\boldsymbol{\Omega})+\tilde{\Sigma}_{t,g}(\mathbf{r},\boldsymbol{\Omega})\psi_g(\mathbf{r},\boldsymbol{\Omega})=&\frac{1}{4\pi}\sum_{g'=1}^{G}\tilde{\Sigma}_{s,g'\rightarrow g}(\mathbf{r})\phi_{g'}(\mathbf{r})\\
&+\frac{1}{4\pi k_{eff}}\sum_{g'=1}^{G}\nu\Sigma_{f,g'\rightarrow g}(\mathbf{r})\phi_{g'}(\mathbf{r})
\end{aligned}\tag{2.31}$$

Thus far, no approximations have been made in the energy discretization of multi-group transport equation given in Eqn. 2.31. However, the expression for the total multi-group cross section $\tilde{\Sigma}_{t,g}$ in Eqn. 2.28 presents a complication since it is dependent on the unknown angular flux. One common approximation used to eliminate the angular dependence is presented in the following section.

2.3.2 Flux Separability Approximation

The angular dependence of the total cross section is often treated with the flux separability approximation. Flux separability makes the simplifying assumption that the energy and angular dependence of the flux varies independently such that the angular flux can be written as the product of the scalar neutron flux $\phi(\mathbf{r}, E)$ and some function $W(\mathbf{r}, \mathbf{\Omega})$:

$$\psi(\mathbf{r}, \mathbf{\Omega}, E) = \phi(\mathbf{r}, E) W(\mathbf{r}, \mathbf{\Omega}) \tag{2.32}$$

The angular dependence of the $\tilde{\Sigma}_{t,g}$ may then be eliminated by inserting Eqn. 2.32 into Eqn. 2.28, factoring out $W(\mathbf{r}, \mathbf{\Omega})$ and writing $\tilde{\Sigma}_t$ in terms of the scalar flux:

$$\begin{aligned} \tilde{\Sigma}_{t,g}(\mathbf{r}) &\equiv \frac{\int_{E_g}^{E_{g-1}} \tilde{\Sigma}_t(\mathbf{r}, E) \phi(\mathbf{r}, E) W(\mathbf{r}, \mathbf{\Omega}) \mathrm{d}E}{\phi_g(\mathbf{r}) W(\mathbf{r}, \mathbf{\Omega})} \\ &= \frac{\int_{E_g}^{E_{g-1}} \tilde{\Sigma}_t(\mathbf{r}, E) \phi(\mathbf{r}, E) \mathrm{d}E}{\phi_g(\mathbf{r})} \end{aligned} \tag{2.33}$$

Although flux separability is a simple and commonly used approach to reduce the complexity of the "true" multi-group total cross section, it is not always valid and may not preserve neutron balance. The impact of the flux separability approximation is systematically investigated and quantified in Secs. 5 and 6 for some simple PWR benchmark models.

2.3.3 Scattering Production

Although neutron production is dominated by fission in nuclear reactors, there may also be non-negligible production of neutrons due to scattering multiplicity (n, xn) reactions. Production from scattering is typically accounted for with a factor $\nu_{scatt,g' \to g}$ which represents the average number of neutrons produced in a scattering reaction. The factor $\nu_{scatt,g' \to g}$ may be lumped into the scattering matrix $\nu_{scatt,g' \to g} \Sigma_{s,g' \to g}$ and substituted directly into the scattering source term in the multi-group transport equation[9].

[9] It should be noted that (n, xn) reactions typically have different angular and energy distributions than those used to treat (n, n) scattering reactions. Although the energy dependence may be lumped into the multi-group scattering matrix, the angular dependence must be embedded in the angular expansion of the scattering kernel.

The k_{eff} eigenvalue is only defined for the fission operator in the transport equation. As a result, deterministic methods which compute the eigenvalue using the fission production, absorption and leakage rates will fail to capture scattering production in the multiplication factor. In order to account for (n, xn) reactions, a "corrected" absorption cross section $\tilde{\Sigma}_{a,g} = \Sigma_{a,g} - (\nu_{scatt,g'\rightarrow g} - 1)\Sigma_{s,g'\rightarrow g}$ must be used to preserve neutron balance. Alternatively, (n, xn) reactions will be accounted for if the eigenvalue is computed as the ratio of successive fission sources in iterative deterministic methods.

2.3.4 Fission Matrix Condensation

The scattering and fission production matrices dominate the memory storage requirements for MGXS since they depend on both incoming and outgoing energy groups. In thermal reactors, the energy distribution of neutrons produced in fission is nearly independent of the incoming neutron energy, and the fission production matrix can be condensed into two single dimensional vectors for each energy group. The fission spectrum χ_g is introduced as a probability distribution over outgoing fission energies:

$$\chi_g \equiv \frac{\sum_{g'=1}^{G} \nu\Sigma_{f,g'\rightarrow g}\phi_{g'}}{\sum_{g=1}^{G}\sum_{g'=1}^{G} \nu\Sigma_{f,g'\rightarrow g}\phi_{g'}} \tag{2.34}$$

The group-wise fission production cross section $\nu\Sigma_{f,g'}$ is defined as the sum of $\nu\Sigma_{f,g'\rightarrow g}$ over all outgoing groups and signifies the probability of fission occurring in group g:

$$\nu\Sigma_{f,g'} \equiv \sum_{g'=1}^{G} \nu\Sigma_{f,g'\rightarrow g} \tag{2.35}$$

Upon substituting χ_g and $\nu\Sigma_{f,g}$ into Eqn. 2.31 one obtains:

$$\mathbf{\Omega}\cdot\nabla\psi_g(\mathbf{r},\mathbf{\Omega}) + \tilde{\Sigma}_{t,g}(\mathbf{r})\psi_g(\mathbf{r},\mathbf{\Omega}) = \frac{1}{4\pi}\sum_{g'=1}^{G}\tilde{\Sigma}_{s,g'\rightarrow g}(\mathbf{r})\phi_g(\mathbf{r}) + \frac{\chi_g}{4\pi k_{eff}}\sum_{g'=1}^{G}\nu\Sigma_{f,g'}(\mathbf{r})\phi_{g'}(\mathbf{r}) \tag{2.36}$$

The multi-group transport equation is defined in terms of the group-wise fluxes and cross sections. MGXS are the flux-weighted averages of the continuous energy cross sections that preserve reaction rates in each group. Flux separability uses the scalar instead of the angular flux to weight the total cross section. The fission matrix is condensed into the fission production cross section and the energy spectrum vectors.

2.4 Approximations in Space

2.4.1 Spatial Homogenization

Up to this point, the MGXS have been defined as continuously varying in space. In practice, most deterministic methods used to solve the multi-group transport equation make the simplifying assumption that material properties are constant across each spatial mesh cell. Spatial homogenization is used to compute flux-weighted volume-averaged cross sections within each mesh cell k with volume V_k as follows:

$$\tilde{\Sigma}_{t,k,g} \equiv \frac{\int_{\mathbf{r}\in V_k} \left[\int_{E_g}^{E_{g-1}} \tilde{\Sigma}_t(\mathbf{r}, E)\phi(\mathbf{r}, E)\mathrm{d}E \right] \mathrm{d}\mathbf{r}}{\int_{\mathbf{r}\in V_k} \phi_g(\mathbf{r})\mathrm{d}\mathbf{r}} \tag{2.37}$$

$$\tilde{\Sigma}_{s,k,g'\to g} \equiv \frac{\int_{\mathbf{r}\in V_k} \left[\int_{E_g}^{E_{g-1}} \int_{E_{g'}}^{E_{g'-1}} \tilde{\Sigma}_s(\mathbf{r}, E'\to E)\phi(\mathbf{r}, E')\mathrm{d}E'\mathrm{d}E \right] \mathrm{d}\mathbf{r}}{\int_{\mathbf{r}\in V_k} \phi_{g'}(\mathbf{r})\mathrm{d}\mathbf{r}} \tag{2.38}$$

$$\nu\tilde{\Sigma}_{f,k,g'\to g} \equiv \frac{\int_{\mathbf{r}\in V_k} \left[\int_{E_g}^{E_{g-1}} \int_{E_{g'}}^{E_{g'-1}} \nu\Sigma_f(\mathbf{r}, E'\to E)\phi(\mathbf{r}, E')\mathrm{d}E'\mathrm{d}E \right] \mathrm{d}\mathbf{r}}{\int_{\mathbf{r}\in V_k} \phi_{g'}(\mathbf{r})\mathrm{d}\mathbf{r}} \tag{2.39}$$

Spatial homogenization may be applied to compute the fission spectrum $\chi_{k,g}$ from the fission production cross section in Eqn. 2.39. The spatially-homogenized MGXS must be appropriately defined to preserve reaction rates such that the following equation holds true for each spatial zone k and energy group g:

$$\mathbf{\Omega} \cdot \nabla \psi_g(\mathbf{r}, \mathbf{\Omega}) + \tilde{\Sigma}_{t,g} \psi_g(\mathbf{r}, \mathbf{\Omega}) = \frac{1}{4\pi} \sum_{g'=1}^{G} \tilde{\Sigma}_{s,k,g' \to g} \phi_{g'}(\mathbf{r}) + \frac{\chi_{k,g}}{4\pi k_{eff}} \sum_{g'=1}^{G} \nu\Sigma_{f,k,g'} \phi_{g'}(\mathbf{r}) \tag{2.40}$$

This is the form of the multi-group transport equation solved by the deterministic OpenMOC code in this thesis.

Spatially-homogenized MGXS preserve reaction rates in discrete spatial zones.

2.5 MGXS Generation

The preceding sections described approximations reducing the dimensionality of the transport equation to permit efficient computational simulation. The solution of the transport equation in Eqn. 2.40 requires knowledge of the energy condensed and spatially-homogenized MGXS in Eqns. (2.37) to (2.39). This section describes the challenges to computing MGXS and outlines the standard multi-level approach for MGXS generation. The chapter concludes with a brief introduction to the potential for Monte Carlo methods as an alternative pathway for accurate MGXS generation.

2.5.1 Challenges

Given a solution to the transport equation for the multi-group flux, general reaction rate distributions may be computed (*e.g.*, radiative capture, recoverable fission energy) with multi-group cross sections for each nuclide and reaction type of interest. Energy condensation and spatial homogenization may be applied to compute multi-group microscopic cross sections $\sigma_{x,i}$ for each reaction type x and nuclide i:

$$\sigma_{x,i,k,g} = \frac{\int_{\mathbf{r} \in V_k} \int_{E_g}^{E_{g-1}} \sigma_{x,i}(\mathbf{r}, E) \phi(\mathbf{r}, E) \mathrm{d}E \mathrm{d}\mathbf{r}}{\int_{\mathbf{r} \in V_k} \int_{E_g}^{E_{g-1}} \phi(\mathbf{r}, E) \mathrm{d}E \mathrm{d}\mathbf{r}} \tag{2.41}$$

The accurate determination of MGXS depends on an accurate understanding of the spatial and energy dependence of the microscopic cross sections and the flux, each of which is depicted in Fig. 2-2. The spatial variation of $\sigma_{x,i}$ is known from the reactor's

geometric configuration[10], and the energy dependence is known from experimental data. However, the spatial and energy dependence of the flux are unknown which presents a "chicken-or-the-egg" type of quandary for multi-group calculations. This predicament is aptly described in the NJOY manual [6]:

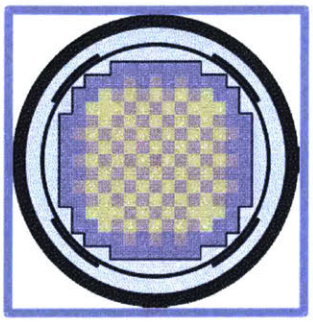
> *"Wait a minute," you ask, "the purpose of solving the transport equation is to get the flux, but I have to know the flux to compute the multi-group constants!" This conundrum is the source of much of the "art" in using multi-group methods.*

The MGXS are needed to compute the flux, but the flux is needed to compute MGXS. As a result, an informed guess is generally made about the flux distribution in order to compute MGXS. Such estimates of the flux introduce further approximations in addition to those presented in Secs. 2.2 to 2.4. The following section discusses a high-level overview of the standard multi-level approach to estimating the flux for MGXS generation.

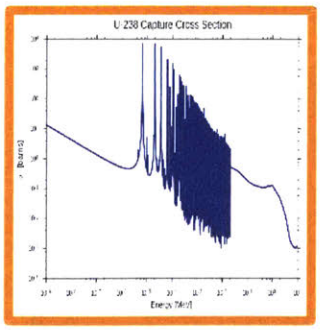
2.5.2 Standard Multi-Level Approach

The standard techniques for multi-group cross section generation use a multi-level framework to draw a compromise between accuracy and computational efficiency. The goal in this process is to define a library of MGXS which enforces an equivalence between an accurate fine-mesh transport calculation and a corresponding computationally efficient coarse mesh transport or diffusion calculation. The angular, energy and spatial variation of the flux is treated with varying degrees of complexity at each level as illustrated in Fig. 2-3. The process begins with a resonance self-shielding calculation that solves the slowing down problem with point-wise nuclear cross section data in a simple infinite medium, slab or pin cell geometry. The self-shielding calculation produces an MGXS library with $\mathcal{O}(100)$ energy groups that is next used by a lattice physics calculation to capture spectral interactions within and between fuel pins in each unique fuel assembly.

[10]The microscopic cross section $\sigma_{x,i}$ is a property unique to each nuclide and reaction and does not vary in space. However, the volume V_k may span different regions which may or may not contain nuclide n. The discrete spatial variation in the number density N_i may be modeled as $\sigma_{x,i}(\mathbf{r}, E) = \sigma_{x,i}(E)\mathbb{1}_{V_k}(\mathbf{r})$ with the indicator function $\mathbb{1}_{V_k}(mathbf r)$ equal to 1 if $\mathbf{r} \in V_k$ and 0 otherwise.

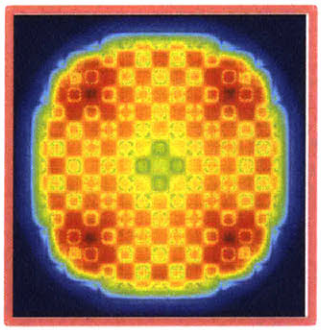


(a)

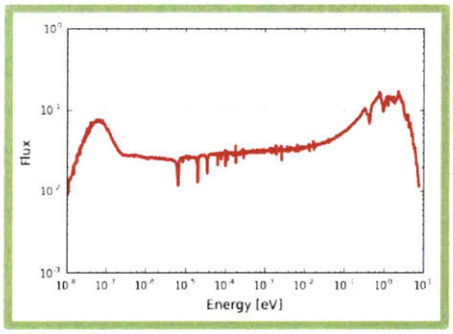


(b)

$$\sigma_{x,i,k,g} = \frac{\int_{\mathbf{r}\in V_k} \int_{E_g}^{E_{g-1}} \sigma_{x,i}(\mathrm{r}, E)\phi(\mathrm{r}, E)\mathrm{d}E\mathrm{d}\mathbf{r}}{\int_{\mathbf{r}\in V_k} \int_{E_g}^{E_{g-1}} \phi(\mathrm{r}, E)\mathrm{d}E\mathrm{d}\mathbf{r}}$$



(c)



(d)

Figure 2-2: The calculation of multi-group cross sections requires knowledge of the spatial and energy variation of the continuous energy cross section and flux. The color-coded position **r** and energy E variables correspond to the figures with the matching colored outlines. The microscopic cross section $\sigma_{x,i}$ depends on reactor configuration (a) and neutron energy (b). The flux ϕ varies with position (c) and energy (d).

The lattice physics calculation performs spatial homogenization across each assembly and produces a condensed library with $\mathcal{O}(2-10)$ groups. This assembly-homogenized few group MGXS library is finally used by a full-core nodal diffusion calculation.

The primary challenge throughout this multi-level scheme is to incorporate spectral interactions known as *self-shielding effects* in energy and space. *Energy self-shielding* refers to the impact that resonance interactions may have on the shape of the flux in energy. For example, the sharp depressions in PWR flux spectra at energies near the large U-238 thermal capture resonances is an example of energy self-shielding (see Fig. 2-2d). These depressions must be accurately captured in the flux used to generate MGXS. *Spatial self-shielding* refers to the impact of the material properties in different

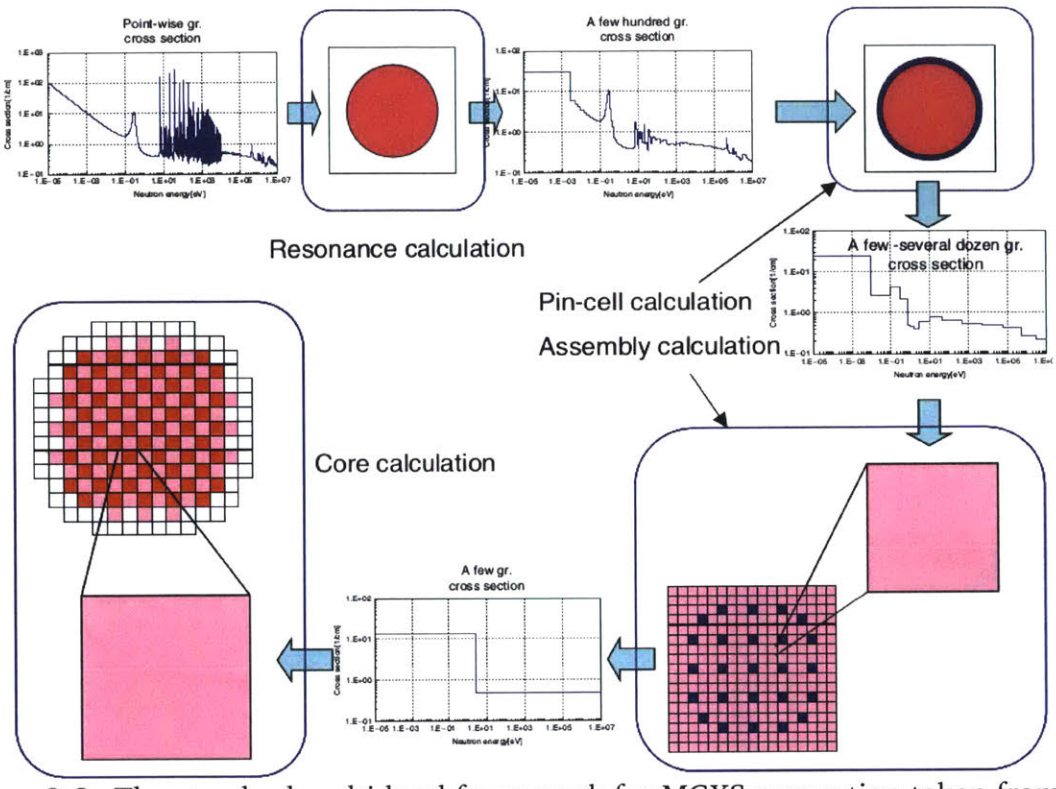


Figure 2-3: The standard multi-level framework for MGXS generation taken from the *Handbook of Nuclear Engineering* [4]. The energy dependence of the nuclear cross sections is successively coarsened with resonance self-shielding and lattice physics calculations to generate multi-group libraries for full-core simulations.

spatial zones on the flux in other nearby zones. For example, the outer rim of an LWR fuel pin shields the center of the pin from neutrons in certain energy groups due to reactions such as U-238 resonance absorption.

There is a long history of approximations for the flux for MGXS generation [4]. These include the narrow, wide and intermediate resonance (NR, WR, and IR) approximations used to model the flux in a resonance self-shielding calculation. Lattice physics calculations typically use equivalence in dilution or subgroup methods to model physics in heterogeneous geometries. These approximations are based on simplifying assumptions regarding the geometric configuration, resonance interference effects (mutually overlapping resonances) and temperature distributions[11]. Although ultra-fine methods may be used to compute reference solutions for comparison, they are typically too

[11] Although the temperature dependence of cross sections has been neglected thus far for brevity, it must be captured in MGXS libraries in order to model Doppler broadening for reactivity coefficient calculations.

computationally expensive for routine use.

As a result of the various approximations employed to model the flux spectrum, the conventional approach for MGXS generation is subject to significant engineering prescriptions for different reactor designs. For example, modifications must be made to lattice physics calculations which account for the impact of local spatial heterogeneities, such as fuel rods adjacent to burnable absorbers. Similarly, it is typical to use infinite boundary conditions (reflective, periodic, white) in resonance self-shielding and lattice physics calculations. This may lead to approximation error in full-core calculations which must model the neutron leakage spectra with vacuum boundary conditions, as well as the flux at the interface between different adjacent assemblies and/or a reflector. As a result, adjustments must be made to the standard multi-level approach in order for the final full-core calculation to accurately account for the effects of spatial heterogeneity.

2.5.3 An Alternative Pathway with Monte Carlo

The multi-level approach of approximations used to generate MGXS is the "Achilles heel" of multi-group methods. The approximate flux used to compute MGXS, along with other approximations inherent in multi-group calculations, require careful use of multi-group methods for reactor analysis. As a result, Monte Carlo must be used to generate reference solutions for benchmarking and verification analysis of multi-group methods. In recent years, there has been growing interest in the use of MC in the MGXS generation process [9–37]. since it offers a potential pathway to use a stochastic approximation to the exact flux in MGXS generation.

Continuous energy Monte Carlo is considered the "gold standard" for neutron transport calculations since it is reactor agnostic and samples over the entire phase space of angle, energy and position. MC is not widely used in routine reactor analysis, however, since its computational expense renders it many orders of magnitude slower than deterministic multi-group methods. Some recent work in the community adopts a hybrid approach which combines the resonance self-shielding and lattice physics stages in a single MC simulation. These schemes minimize the computational expense by modeling

sub-components (*e.g.*, fuel pins and assemblies) rather than a full reactor. Although the hybrid approach has the advantage of using the "true" flux sampled in MC to generate MGXS, special treatment must still be given to accommodate infinite boundary conditions and neutron leakage spectra.

The following chapter introduces stochastic integration with MC and its specific application to compute MGXS. In addition, the chapter reviews past and present efforts to generate MGXS with MC for neutron diffusion and transport codes.

Highlights

- Numerical methods approximate the angular, energy and spatial variables in the neutron transport equation to make it computationally tractable.
- Multi-group theory treats a neutron's energy with a finite set of discrete energy groups. Multi-group theory is valid if MGXS are defined to preserve reaction rates in each energy group.
- Energy condensation and spatial homogenization are used to compute MGXS in each energy group and spatial zone. MGXS generation requires the neutron flux in energy and space to average the continuous energy cross sections.
- Standard MGXS generation methods use a multi-level approach to approximate the neutron flux, accounting for self-shielding effects. Flux approximations are based on engineering prescriptions for specific reactor configurations and spectra and are not generalizable to new core designs.
- Monte Carlo is a promising approach for MGXS generation since it accurately produces an unbiased estimate of the flux without approximation.

Chapter 3

MGXS Generation with Monte Carlo

In the preceding chapter it was observed that many approximations are made in multi-group theory and the generation of multi-group cross sections. Monte Carlo is an approach to replace some of the steps in the standard multi-level framework for MGXS generation with a natural and reactor agnostic treatment of energy and spatial self-shielding. This chapter presents a brief overview of MC tallies and statistics in Sec. 3.1, outlines the necessary computation needed to generate MGXS with MC in Sec. 3.2, and discusses past studies which applied MC for MGXS generation in Sec. 3.3.

3.1 Overview of Monte Carlo Methods

Monte Carlo methods have been successfully applied to neutron transport calculations for many decades. A detailed accounting of the physics models and algorithms used in MC methods can be found in the manuals for the Serpent [1], MCNP [38], and OpenMC [39] Monte Carlo particle transport codes. This section presents a few key aspects related to tallies and statistics, and follows directly from the manual for the OpenMC [39] code which is used throughout this work.

3.1.1 Monte Carlo Tallies

MC simulations sample the particle distribution in order to compute integral quantities of interest called *tallies*. A tally is an integral of a *scoring function* f weighted by the neutron distribution, or flux, across some region of phase space. A general form for tally $\mathcal{T}$ is given by the following integral expression:

$$\mathcal{T} = \int_V \int_S \int_E f(\mathbf{r}, \mathbf{\Omega}, E)\psi(\mathbf{r}, \mathbf{\Omega}, E)\mathrm{d}E\mathrm{d}\mathbf{\Omega}\mathrm{d}\mathbf{r} \tag{3.1}$$

In OpenMC parlance, the integration bounds over space, angle and energy are termed *filters*, while the scoring function f is simply known as a *score*. Various scores may be used to compute volume-integrated fluxes, reaction rates and functional expansions. MC does not perform the integration in Eqn. 3.1 with the exact flux specified at all points in phase space. Instead, MC performs stochastic integration by sampling the particle population across the entirety of phase space to compute a *statistical estimate* $\hat{\mathcal{T}}$ of the true integral $\mathcal{T}$.

There are a number of different techniques to estimate a tally. The first and most general method is known as an *analog estimator*. An analog estimator $\hat{\mathcal{T}}$ increments a tally by the particle weight w_i each time an event i occurs from the set of all events of interest A (*e.g.*, fission events). The sum of particle weights is then normalized by the total weight of all particles W to compute the interaction frequency on a per-particle basis within the phase space volume of interest:

$$\hat{\mathcal{T}} = \frac{1}{W}\sum_{i \in A} w_i \tag{3.2}$$

Although analog estimators permit general filters and scoring functions, they may suffer from poor tallying efficiency if the size of set A is very small compared to the total number of events in a simulation. A *collision estimator* improves the tallying efficiency by incrementing a tally more frequently than is possible with analog estimators. In particular, a collision estimator increments a tally at each collision i from the set of all collisions C irregardless of the types of collisions that took place. By noting that the total collision rate is given by $R_t = \Sigma_t \phi$, a collision estimator for the flux $\hat{\phi}$ may be simply defined by dividing the particle weights by the total macroscopic cross section:

$$\hat{\phi} = \frac{1}{W} \sum_{i \in C} \frac{w_i}{\Sigma_t(E_i)} \tag{3.3}$$

The collision estimator depends on the energy of the incoming particle E_i in order to scale the particle weight by the energy-dependent total cross section. It follows that the collision estimator for a reaction rate $\hat{\mathcal{R}}_x$ is the product of the flux in Eqn. 3.3 and the cross section for the reaction type x of interest:

$$\hat{\mathcal{R}}_x = \frac{1}{W} \sum_{i \in C} \frac{w_i \Sigma_x(E_i)}{\Sigma_t(E_i)} \tag{3.4}$$

The collision estimator increases the number of events in C to improve the tallying efficiency with respect to analog tallies. A third method known as *track-length estimators* goes one step further and increments a tally each time a particle trajectory crosses the phase space of interest (*e.g.*, a spatial tally mesh zone) even if a collision did not take place. The track-length estimator makes use of a particle's distance traveled $\Delta\mathbf{r}$ to estimate the flux. A track-length estimator of the flux is therefore:

$$\hat{\phi} = \frac{1}{W} \sum_{i \in T} w_i \Delta\mathbf{r}_i \tag{3.5}$$

where the set T represents each particle trajectory through the phase space volume of interest. Similarly, a track-length estimate of a reaction rate x can be found by simply multiplying the flux in Eqn. 3.5 by the cross section:

$$\hat{\mathcal{R}}_x = \frac{1}{W} \sum_{i \in T} w_i \Delta\mathbf{r}_i \Sigma_x(E_i) \tag{3.6}$$

The tallying efficiency for track-length estimators is greatly improved by incrementing a tally for each particle trajectory. As a result, the confidence intervals are generally much tighter for track-length estimators than those for analog and collision estimators.

Each of the three estimators – analog, collision and track-length – may be useful for different scenarios. Although track-length and collision estimators improve statistics over analog estimators, they cannot be employed for all types of filters. For example, track-length tallies may not be used if the scoring function requires information about the outgoing particle since this is not available unless a collision has taken place. As discussed in Sec. 3.2, a mixture of estimators which tradeoff generality with efficiency must be used to generate MGXS from MC tallies.

3.1.2 Sample Statistics

MC performs stochastic integration with one of a number of different estimators. In each case, the tally estimator $\hat{\mathcal{T}}$ is computed as a sample mean of of all of the events or particle trajectories simulated. An unbiased estimate of the sample mean is given by:

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{3.7}$$

where each of the N samples is given by a random variable x_i. For MC codes that use batch-based statistics, such as OpenMC, N is the number of batches of particles simulated and x_i is the tally estimator for the i^{th} batch of particles.

Each of the random variables x_i is sampled from some probability distribution representative of the physics in the simulation. The sampling distribution is normal such that $x_i \sim \mathcal{N}(\mu, \sigma^2)$ where $\mathcal{N}(\mu, \sigma^2)$ signifies a normal distribution with mean μ and variance σ^2 if each batch of particles is independent. An unbiased estimate of the variance σ^2 of the normal distribution from which the population x_i is drawn may be estimated using Bessel's correction for the sample variance s^2:

$$s^2 = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \bar{x})^2 \tag{3.8}$$

As $N \to \infty$ the population variance estimator will approach the true variance σ^2 of the underlying distribution. In the case of batch-based statistics, the variance σ^2 will be determined by the number of particles simulated per batch – the more particle histories simulated per batch, the smaller σ^2 will be, and vice versa.

In general, it is more useful to quantify the uncertainty of a tally estimator than it is to compute the sample variance. The variance of the sample mean is representative of the distribution from which the random variable $\bar{x}$ is drawn and indicates the degree of confidence one may have in a tally estimator. By the Central Limit Theorem, the sample mean $\bar{x}$ will converge to the mean of a normal distribution if the samples x_i are uncorrelated. An unbiased estimate of the variance of the sample mean can be derived from the Bienaymé formula to give:

$$s_{\bar{x}}^2 = \frac{1}{N-1}\left(\frac{1}{N}\sum_{i=1}^{N} x_i^2 - \bar{x}^2\right) \tag{3.9}$$

A key observation is that the standard deviation of the sample mean is directly proportional to $1/\sqrt{N}$ if the samples x_i are uncorrelated. This necessarily implies that the uncertainties on a tally estimator can be made arbitrarily small given enough simulated particle histories. However, recent studies have shown that tally estimators in eigenvalue calculations are not generally independent and identically distributed realizations due to correlated fission sources between batches [40, 41].

Monte Carlo provides statistical estimators for quantities such as energy- and volume-integrated reaction rates and fluxes. Track-length estimators are more statistically efficient than analog and collision estimators, but are not generally applicable for scoring functions dependent on outgoing neutron energy.

3.2 MGXS Generation with Monte Carlo

This section describes how multi-group cross sections may be computed using stochastic integration. Sec. 3.2.1 outlines the types of OpenMC tallies needed to generate MGXS – including the scores, filters and estimators for each tally – and the arithmetic combinations used to combine different tallies. Sec. 3.2.2 illustrates how the uncertainties of the MGXS may be estimated using error propagation theory.

3.2.1 Tally Types Needed for MGXS Generation

The types of MGXS needed to solve the neutron transport equation were outlined in Chap. 2, including expressions for the transport-corrected total cross section and scattering matrix, and the fission production cross section and emission spectrum. This section outlines the types of tallies needed to compute these MGXS. It is important to note that the flux separability approximation (Sec. 2.3.2) is applied in the tally formulations for each of the group constants.

3.2.1.1 Inner Product Notation

The following sections use angle bracket notation $\langle \cdot, \cdot \rangle$ to represent inner products in phase space. This may correspond to integrals over incoming and/or outgoing energy, space, and angle. Using this notation, a tally estimator for reaction rate x is represented as follows:

$$\langle \Sigma_x, \psi \rangle = \int_V \int_S \int_E \Sigma_x(\mathbf{r}, E)\psi(\mathbf{r}, E, \mathbf{\Omega}) \mathrm{d}E \mathrm{d}\mathbf{\Omega} \mathrm{d}\mathbf{r} \tag{3.10}$$

This notation is specialized throughout this section with subscripts to indicate the subsets of phase space that are integrated over in the inner product. In particular, subscript k refers to a volume integral over V_k for some region of space k for spatial homogenization (Sec. 2.4.1), while subscript g corresponds to an integral over energies with $E \in [E_g, E_{g-1}]$ for energy condensation (Sec. 2.3.1). For example, the microscopic reaction rate for reaction x by nuclide i is denoted as:

$$\langle \sigma_{x,i}, \psi \rangle_{k,g} = \int_{\mathbf{r} \in V_k} \int_{4\pi} \int_{E_g}^{E_{g-1}} \sigma_{x,i}(\mathbf{r}, E)\psi(\mathbf{r}, E, \mathbf{\Omega}) \mathrm{d}E \mathrm{d}\mathbf{\Omega} \mathrm{d}\mathbf{r} \tag{3.11}$$

The inner product of a function with unity, such as the spatially-homogenized and energy-integrated flux is denoted by:

$$\langle \psi \rangle_{k,g} \equiv \langle \psi, \mathbb{1} \rangle_{k,g} = \int_{\mathbf{r} \in V_k} \int_{4\pi} \int_{E_g}^{E_{g-1}} \psi(\mathbf{r}, E, \mathbf{\Omega}) \mathrm{d}E \mathrm{d}\mathbf{\Omega} \mathrm{d}\mathbf{r} \tag{3.12}$$

Finally, the superscripts a and $t\ell$ are given to those inner products computed with analog and track-length estimators, respectively – *i.e.*, $\langle \cdot, \cdot \rangle^a$ is an analog tally estimator and $\langle \cdot, \cdot \rangle^{t\ell}$ is a track-length tally estimator of the corresponding inner products.

3.2.1.2 General Reaction Cross Section

A general spatially-homogenized and energy condensed macroscopic multi-group cross section for reaction x, spatial zone k and energy group g can be computed with track-length tally estimators in OpenMC. The MGXS is simply the ratio of the group-wise reaction rates $\langle \Sigma_x, \psi \rangle_{k,g}^{t\ell}$ and fluxes $\langle \psi \rangle_{k,g}^{t\ell}$:

$$\hat{\Sigma}_{x,k,g} = \frac{\langle \Sigma_x, \psi \rangle^{t\ell}_{k,g}}{\langle \psi \rangle^{t\ell}_{k,g}} \tag{3.13}$$

Likewise, a microscopic MGXS for nuclide i can be computed as follows:

$$\hat{\sigma}_{x,i,k,g} = \frac{\langle \sigma_{x,i}, \psi \rangle^{t\ell}_{k,g}}{\langle \psi \rangle^{t\ell}_{k,g}} \tag{3.14}$$

These estimators are used for reaction types which are only dependent on the incoming energy of a neutron, such as total and radiative capture reactions.

3.2.1.3 Total Cross Section

The total macroscopic cross section Σ_t is a special case of Eqn. 3.14, with track-length estimators for the total collision rate and flux:

$$\hat{\Sigma}_{t,k,g} = \frac{\langle \Sigma_t, \psi \rangle^{t\ell}_{k,g}}{\langle \psi \rangle^{t\ell}_{k,g}} \tag{3.15}$$

As discussed in Sec. 2.2.3, a transport correction is often used to incorporate anisotropic scattering effects into the transport equation with an isotropic scattering kernel. An expression for the in-scatter approximation [8] to the transport correction can be computed with an OpenMC tally for the first Legendre scattering moment[12]. The inner product for this tally is given by:

$$\langle \Sigma_{s1}, \psi \rangle_{k,g' \to g} = \int_{\mathbf{r} \in V_k} \int_{4\pi} \int_{E_g}^{E_{g-1}} \int_{E_{g'}}^{E_{g'-1}} \Sigma_{s1}(\mathbf{r}, E' \to E) \psi(\mathbf{r}, E', \mathbf{\Omega}) \mathrm{d}E' \mathrm{d}E \mathrm{d}\mathbf{\Omega} \mathrm{d}\mathbf{r} \tag{3.16}$$

An analog estimator must be used in OpenMC since the tally includes an integral over the outgoing neutron energy. The spatially-homogenized and energy condensed transport-corrected total cross section given in Eqn. 2.20 is computed by summing over all incoming energy groups:

[12] It is assumed that scattering multiplicity is included in the scattering moments as discussed in Sec. 2.3.3.

$$\Delta\hat{\Sigma}_{tr,k,g} = \sum_{g'=1}^{G} \langle \Sigma_{s1}, \psi \rangle^{a}_{k,g' \to g} \tag{3.17}$$

The transport correction is then subtracted from the group-wise total collision rate and normalized by the flux to compute the transport-corrected total cross section:

$$\hat{\hat{\Sigma}}_{t,k,g} = \frac{\langle \Sigma_t, \psi \rangle^{a}_{k,g} - \Delta\hat{\Sigma}_{tr,k,g}}{\langle \psi \rangle^{a}_{k,g}} \tag{3.18}$$

Note that since the transport correction must be computed using an analog estimator, the total collision and flux in Eqn. 3.18 must also be computed with analog estimators.

3.2.1.4 Scattering Matrix

The isotropic scattering matrix is computed with an inner product of scattering reactions over both incoming and outgoing energies. An analog estimator must be used since the integral is dependent on the neutron's outgoing energy. Similar to the first Legendre moment in Eqn. 3.16, the isotropic scattering moment is given by the following expression:

$$\langle \Sigma_{s0}, \psi \rangle_{k,g' \to g} = \int_{\mathbf{r} \in V_k} \int_{4\pi} \int_{E_g}^{E_{g-1}} \int_{E_{g'}}^{E_{g'-1}} \Sigma_{s0}(\mathbf{r}, E' \to E) \psi(\mathbf{r}, E, \mathbf{\Omega}) \mathrm{d}E' \mathrm{d}E \mathrm{d}\mathbf{\Omega} \mathrm{d}\mathbf{r} \tag{3.19}$$

The isotropic scattering matrix is then:

$$\hat{\Sigma}_{s,k,g' \to g} = \frac{\langle \Sigma_{s0}, \psi \rangle^{a}_{k,g' \to g}}{\langle \psi \rangle^{a}_{k,g'}} \tag{3.20}$$

The transport correction in Eqn. 3.17 can be applied by subtracting it from the diagonal elements in the matrix to compute the transport-corrected scattering matrix:

$$\hat{\hat{\Sigma}}_{s,k,g' \to g} = \frac{\langle \Sigma_{s0}, \psi \rangle^{a}_{k,g' \to g} - \delta_{g,g'} \Delta\hat{\Sigma}_{tr,k,g}}{\langle \psi \rangle^{a}_{k,g'}} \tag{3.21}$$

3.2.1.5 Fission Production Cross Section

The fission production cross section was condensed in Eqn. 2.35 to make it independent of the energies of the neutrons emitted from fission. It is therefore straightforward to treat the fission product macroscopic cross section $\nu\Sigma_f$ as a special case of Eqn. 3.14, with track-length estimators for the total collision rate and flux:

$$\hat{\nu\Sigma}_{f,k,g} = \frac{\langle \nu\Sigma_f, \psi \rangle^{t\ell}_{k,g}}{\langle \psi \rangle^{t\ell}_{k,g}} \tag{3.22}$$

3.2.1.6 Fission Energy Spectrum

Unlike the fission production cross section, the fission spectrum is dependent on the outgoing neutron energy and must be computed with analog estimators. The fission production matrix from group g' into group g is given by the following inner product:

$$\langle \nu\Sigma_f, \psi \rangle_{k,g'\to g} = \int_{\mathbf{r}\in V_k} \int_{4\pi} \int_{E_g}^{E_{g-1}} \int_{E_{g'}}^{E_{g'-1}} \nu\Sigma_f(\mathbf{r}, E' \to E)\psi(\mathbf{r}, E, \mathbf{\Omega}) \mathrm{d}E' \mathrm{d}E \mathrm{d}\mathbf{\Omega} \mathrm{d}\mathbf{r} \tag{3.23}$$

The fission spectrum in Eqn. 2.35 can then be computed from this tally by summing over incoming and outgoing energy groups:

$$\hat{\chi}_{k,g} = \frac{\sum_{g'=1}^{G} \langle \nu\Sigma_f, \psi \rangle^{a}_{k,g'\to g}}{\sum_{g=1}^{G}\sum_{g'=1}^{G} \langle \nu\Sigma_f, \psi \rangle^{a}_{k,g'\to g}} \tag{3.24}$$

This expression for the fission spectrum will result in a normalized discrete probability distribution for the energy of neutrons emitted from fission.

3.2.1.7 Summary

The tallies needed to generate MGXS libraries were outlined in detail in the preceding sections, and are summarized in Table 3.1. The scores and filters correspond to the notation used by the OpenMC code to describe the scoring function and integration

bounds used in Eqn. 3.1. The energy group structure for energy condensation is specified by `energy` and/or `energyout` filters in the table. The regions for spatial homogenization are specified by `material` or `cell` filters, although this could potentially include `universe`, `distribcell` and `mesh` filters as well.

Table 3.1: The types of tallies used in MGXS generation with OpenMC.

Name	Symbol	Tally	Score	Filters	Estimator
General	$\hat{\Sigma}_{x,k,g}$	$\langle \Sigma_x, \psi \rangle_{k,g}$	reaction x	`material/cell energy`	`track-length`
		$\langle \psi \rangle_{k,g}$	`flux`	`material/cell energy`	`track-length`
Total	$\hat{\Sigma}_{t,k,g}$	$\langle \Sigma_t, \psi \rangle_{k,g}$	`total`	`material/cell energy`	`track-length`
		$\langle \psi \rangle_{k,g}$	`flux`	`material/cell energy`	`track-length`
Transport-Corrected Total	$\hat{\tilde{\Sigma}}_{t,k,g}$	$\langle \Sigma_t, \psi \rangle_{k,g}$	`total`	`material/cell energy`	`analog`
		$\langle \Sigma_{s1}, \psi \rangle_{k,g' \to g}$	`nu-scatter-1`	`material/cell energyout`	`analog`
		$\langle \psi \rangle_{k,g}$	`flux`	`material/cell energy`	`analog`
Scattering Matrix	$\hat{\Sigma}_{s,k,g' \to g}$	$\langle \Sigma_{s0}, \psi \rangle_{k,g' \to g}$	`nu-scatter-0`	`material/cell energy energyout`	`analog`
		$\langle \psi \rangle_{k,g}$	`flux`	`material/cell energy`	`analog`
Transport-Corrected Scattering Matrix	$\hat{\tilde{\Sigma}}_{s,k,g' \to g}$	$\langle \Sigma_{s0}, \psi \rangle_{k,g' \to g}$	`nu-scatter-0`	`material/cell energy energyout`	`analog`
		$\langle \Sigma_{s1}, \psi \rangle_{k,g' \to g}$	`nu-scatter-1`	`material/cell energyout`	`analog`
		$\langle \psi \rangle_{k,g}$	`flux`	`material/cell energy`	`analog`
Fission Production	$\nu\hat{\Sigma}_{f,k,g}$	$\langle \nu\Sigma_f, \psi \rangle_{k,g}$	`nu-fission`	`material/cell energy`	`track-length`
		$\langle \psi \rangle_{k,g}$	`flux`	`material/cell energy`	`track-length`
Fission Spectrum	$\hat{\chi}_{k,g}$	$\langle \nu\Sigma_f, \psi \rangle_{k,g' \to g}$	`nu-fission`	`material/cell energy energyout`	`analog`

3.2.2 Uncertainty Propagation

As discussed in the preceding sections, MGXS may be computed using arithmetic combinations of tally estimators for reaction rates and fluxes. Each tally estimator is a random variable with an associated uncertainty estimated by the variance of the sample mean in Eqn. 3.9. As a result, each multi-group cross section computed for a spatial zone and energy group is itself a random variable from a distribution with some unknown variance. It is therefore useful to estimate the uncertainty of MGXS computed from

MC tallies in order to quantify whether the MGXS are known with enough precision for accurate multi-group calculations.

Estimates of the variance may be deduced from standard error propagation theory. Such analysis is widely discussed in the literature [42]. A few key equations necessary to estimate the variance for MGXS are reproduced here. The arithmetic combinations of interest for MGXS generation include addition, subtraction, multiplication and division.

Consider two random variables X and Y, generated from distributions with variances σ_X^2 and σ_Y^2 which are arithmetically combined into a new random variable Z with variance σ_Z^2. The random variables X and Y may correspond to tallies for reaction rates and the flux, while Z could correspond to a MGXS. The following expressions can be derived for the variance σ_Z^2 for binary combinations of X and Y:

$$Z = X + Y \qquad \sigma_Z^2 = \sigma_X^2 + \sigma_Y^2 + 2\sigma_{XY} \tag{3.25}$$

$$Z = X - Y \qquad \sigma_Z^2 = \sigma_X^2 + \sigma_Y^2 - 2\sigma_{XY} \tag{3.26}$$

$$Z = XY \qquad \sigma_Z^2 \approx Z^2\left[\left(\frac{\sigma_X}{X}\right)^2 + \left(\frac{\sigma_Y}{Y}\right)^2 + 2\frac{\sigma_{XY}}{Z}\right] \tag{3.27}$$

$$Z = \frac{X}{Y} \qquad \sigma_Z^2 \approx Z^2\left[\left(\frac{\sigma_X}{X}\right)^2 + \left(\frac{\sigma_Y}{Y}\right)^2 - 2\frac{\sigma_{XY}}{Z}\right] \tag{3.28}$$

These expressions are given in terms of the covariance σ_{XY} of X and Y:

$$\sigma_{XY} = \mathbb{E}[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])] \tag{3.29}$$

where $\mathbb{E}[\cdot]$ is the expectation operator. The covariance is not generally computable using the standard formulation for a tally estimator in a Monte Carlo simulation. Although it would be possible to estimate the covariance using ensemble statistics[13], this is not often feasible. Instead, the covariance terms in Eqns. (3.25) to (3.28) are typically neglected. In general, the random variables for reaction rates and fluxes in the same volume of phase space are highly correlated, and neglecting the covariance leads to a poor approximation for the variance of MGXS. However, it should be noted that division is the primary operation needed to combine tallies to compute MGXS. Since the reaction rates and flux tallies must be positively correlated, the covariance term in Eqn. 3.28

[13]The covariance could be estimated from the results of an ensemble of independent MC simulations.

reduces the estimate of the covariance. It therefore follows that a conservative estimate of the variance for MGXS is obtained by neglecting the covariance.

A mixture of analog and track-length MC tallies for reaction rates and fluxes are used to generate spatially-homogenized and energy condensed MGXS. Error propagation theory is used to estimate the MGXS uncertainties.

3.3 A Literature Review of MGXS Generation with MC

The last two decades have seen growing interest in Monte Carlo as a means to generate MGXS libraries. This section presents a brief overview of the literature which documents these efforts. As discussed in Sec. 3.3.1, most of the work to date has been directed at generating homogenized few-group constants for coarse mesh diffusion-based codes. Sec. 3.3.2 reviews a few recent theses which develop MC-based methods to generate MGXS for fine-mesh transport-based simulations, which is the motivation for this thesis.

3.3.1 MGXS for Coarse Mesh Diffusion Calculations

Most MC-based MGXS generation schemes to date focus on generating few-group constants for coarse mesh diffusion codes[14]. These schemes aim to improve the accuracy of standard diffusion codes for analysis of atypical core configurations for which the simplifications made by multi-level deterministic MGXS generation methods are not necessarily applicable. These efforts replace the separate resonance self-shielding and deterministic lattice physics calculation steps in multi-level approaches (see Sec. 2.5.2) with fully-detailed MC calculations of each assembly to compute the few-group constants needed by whole core diffusion codes. The widely used Serpent code discussed in Sec. 3.3.1.1 has led this trend over the last decade, and a few authors have applied the MCNP and McCARD codes in a similar fashion as will be highlighted in Secs. 3.3.1.2 and 3.3.1.3. The latter subsections summarize studies which used the MC21, MVP-BURN,

[14]In this context, *coarse mesh* refers to the use of one or a few homogenized mesh cells per assembly.

RCP01 and VIM codes to generate MGXS for coarse mesh diffusion calculations.

3.3.1.1 Serpent

Serpent is a continuous energy Monte Carlo code developed by the VTT Technical Research Centre of Finland, and is one of the most widely used MC particle transport codes in the world [1]. Serpent was initially created as part of Leppänen's thesis [9] to generate few-group constants for nodal diffusion codes. More recently, Serpent has developed into a general purpose reactor physics burnup code with features including isotopic depletion, on-the-fly Doppler broadening and support for CAD geometries.

Serpent is designed to be a drop-in replacement for deterministic resonance self-shielding and lattice physics calculation codes and can generate homogenized few-group constants for coarse mesh sub-assembly geometries. Serpent is uniquely designed for coarse mesh MGXS generation since it uses the Woodcock-Delta method [43] to greatly reduce the computational expense of tracking particles in geometries with complicated surface crossings. Unlike deterministic multi-level approaches, Serpent uses MC to precisely model self-shielding effects in complicated geometries. In addition, Serpent simplifies the validation of downstream diffusion codes which use the MGXS it generates since it is also capable of computing full-core reference solutions.

One of the challenges for lattice physics calculations is the appropriate treatment of net current between assemblies. In order to address this, Serpent employs a two-step energy condensation scheme including an infinite lattice calculation followed by a B_1 leakage correction [10]. The scheme first performs an infinite lattice calculation for each unique assembly and tallies MGXS in the WIMS 69-group structure. The MGXS are used to form the B_1 equations for the homogenized system which are solved for the critical flux spectrum accounting for inter-assembly leakage. The critical flux is finally used to collapse the 69-group MGXS into few-group constants. Although the B_1 equations make assumptions that are not always true – such as an energy-independent buckling – the approach has been demonstrated to largely resolve 20% errors in 2-group diffusion coefficients compared to those collapsed with the infinite flux for simple PWR benchmarks [10]. However, the B_1 leakage correction does have some known

deficiencies, including its inability to treat non-fissile regions such as homogenized reflectors or transmutation cross sections in burnup calculations [11].

Much of the research focus for Serpent has revolved around the need to accurately compute diffusion coefficients. Unlike the total, scattering, and fission production MGXS discussed in Chap. 2, diffusion coefficients do not have a continuous energy counterpart in transport theory. Serpent uses an approximate method to compute the Selengut-Goertzel diffusion coefficient from the inverse of the transport cross section[15]. In particular, Serpent tallies a 69-group MGXS transport cross section, and condenses its inverse with the B_1 leakage corrected spectra to compute diffusion coefficients. In addition, since it is not straightforward to compute current-weighted tallies in MC, Serpent makes a heuristic approximation and uses a scalar flux-weighted scattering moment to compute the transport cross section (see Sec. 2.2.3). More recently, the Serpent team has investigated a novel current-weighted scheme for directional diffusion coefficients [12], and introduced Liu's novel Cumulative Migration Method (CMM) [25] to mitigate shortcomings in the current approach [11].

Although Serpent is the most popular tool for MGXS generation, its use thus far has been specifically focused on coarse mesh diffusion applications. To the first author's knowledge, there have not yet been any published works which use Serpent to generate MGXS for fine-mesh transport calculations.

3.3.1.2 MCNP

The widely used MCNP code [38] developed by Los Alamos National Laboratory has also been employed to generate few-group MGXS for coarse mesh diffusion methods. Perhaps the most comprehensive study can be found in Pounders' thesis [14] which, like Serpent, aimed to improve the accuracy of standard diffusion codes by generating improved diffusion coefficients with Monte Carlo. In addition to the reactor problems considered in Pounders' thesis, MCNP has also been used to generate diffusion coefficients for analysis of spent fuel storage lattices [13].

[15]This is an approximate method since the energy collapse of the transport cross section is not equivalent to the energy collapse of its inverse, which is what is needed to collapse the diffusion coefficient.

Pounders considered several approaches to compute diffusion coefficients with Monte Carlo and implemented each in MCNP. The simplest method computed the diffusion coefficient as the inverse of the transport cross section. This formulation is most similar to that used in Serpent, but did not include a fine-to-few-group collapse with the B_1 leakage corrected spectra. Pounders also considered the flux-limited diffusion coefficient [44] widely used in radiation hydrodynamics codes. In addition, Pounders developed an approach to use directional neutron currents to appropriately weight the first order scattering moment needed to compute the transport cross section. Lastly, Pounders' most novel contribution, termed the "stochastic diffusion coefficient," attempted to embed higher order anisotropies in the diffusion coefficient by combining direction-dependent diffusion coefficients with volume-integrated flux gradient tallies.

Pounders' results demonstrated that the stochastic diffusion coefficient produced the best results since it mitigated the approximations imposed by P1 theory. However, the analysis was limited to simple 1D slab and 2D pin cell problems since the implementation of stochastic diffusion coefficients for arbitrary 3D reactor geometries would be challenging. Most interestingly, Pounders' thesis concluded that the error resulting from diffusion theory's assumption of a linearly anisotropic flux dominated the error induced by each of the diffusion coefficient formulations. Pounders' thesis has served as the basis for the development of new methods to generate MGXS with MCNP, including new diffusion coefficient formulations [15] and an approach to represent the Legendre expansion of the scattering kernel with equiprobable cosine bins [16].

A few recent studies have evaluated new leakage models for multi-group diffusion coefficient generation with MCNP. Yun and Cho evaluated a hybrid approach to compute diffusion coefficients with MCNP using an albedo-corrected leakage spectrum [17, 18]. This approach iterated between a Monte Carlo lattice and deterministic full-core calculations to converge nodal diffusion parameters to preserve surface currents to match a criterion defined by any arbitrary leakage model. Cho evaluated the method for a PWR fuel assembly and demonstrated a slight improvement in the solution compared to a diffusion calculation with MGXS weighted by an infinite spectrum.

Yamamoto [19] developed a method to model neutron leakage with a correction

term in the transport equation which is equivalent to the B_1 leakage correction method. The correction term was introduced in MCNP as a complex-valued particle weight in order to generate anisotropic diffusion coefficients [20]. Few-group diffusion coefficients for pin cell and assembly benchmarks exhibited good agreement with those generated by a reference deterministic code, but no criticality calculations were performed to validate eigenvalues and power distributions computed by a diffusion code.

3.3.1.3 McCARD

The McCARD Monte Carlo transport code developed by Seoul National University has been utilized in a number of studies to generate few-group constants for full-core diffusion analysis [21–23]. The over-arching objective in these studies was to develop and evaluate a method to model the leakage spectrum in MC lattice physics calculations. An approach was independently developed which used the solution to the homogeneous B_1 equations which is equivalent to the methodology used in the Serpent code [10]. In summary, McCARD tallied fine group MGXS and used the critical spectrum solved from the B_1 equations to collapse the fine group MGXS into few-group constants. Like Serpent, McCARD uses the flux rather than the current to weight the first order scattering moment tallies to approximately compute fine group transport cross sections. The studies published in the literature validated the MGXS generated with McCARD with a reference deterministic lattice code. In addition, good agreement was demonstrated between reference MC and deterministic diffusion solutions for the eigenvalues and power distributions for 3D PWR and gas-cooled, TRistructural-ISOtopic (TRISO) fueled Very High Temperature Reactor (VHTR) benchmark configurations. Finally, the B_1 leakage corrected spectra was shown to significantly improve the eigenvalue estimates and assembly-wise power distributions with respect to MGXS computed from an infinite lattice spectrum.

3.3.1.4 MC21

Herman utilized the MC21 code developed by Knolls Atomic Power Laboratory to evaluate two approximations used in Monte Carlo calculations of diffusion coefficients [24].

His work was motivated by approximations made by the multi-step energy condensation methodology used to compute diffusion coefficients in Serpent with a B_1 leakage corrected spectra. First, Herman considered whether to energy condense the fine group transport cross sections or diffusion coefficients when computing few-group diffusion coefficients. Second, he presented an approach to use a pre-tabulated form of the energy-dependent scattering cosine to correct the traditional form of the diffusion coefficient. His results demonstrated a significant reduction from more than 3% to nearly 0.25% in the reconstructed pin powers for a PWR assembly. Herman's corrected diffusion coefficient was implemented in the OpenMC code [45] for use in Coarse Mesh Finite Difference (CMFD) acceleration [46], and has inspired the development of the Cumulative Migration Method (CMM) to calculate diffusion coefficients with Monte Carlo [25].

3.3.1.5 MVP-BURN

Tohjoh used the MVP-BURN Monte Carlo code [26] to generate few-group constants for full core Boiling Water Reactor (BWR) analysis [27]. The objective of the study was to evaluate MC as an approach to generate 3-group constants for advanced BWR assemblies with novel geometries and reactivity controls. Although MVP-BURN could be used to tally reaction rates and fluxes to compute standard MGXS (*e.g.*, total, fission), it was not modified to directly generate diffusion coefficients. Unlike past work with Serpent and MCNP, the diffusion coefficient was computed with MVP-BURN by collapsing the total and scattering cross sections in energy, rather than the transport cross section or the diffusion coefficient itself. In particular, the diffusion coefficient was computed from tallied total and scattering cross sections and a simple heuristic for the average scattering cosine. Furthermore, the code system was unable to produce scattering matrices. Instead, up-scattering was assumed to be negligible and the downscattering cross sections were computed directly from the removal and capture MGXS. The group constants were validated with those produced from a deterministic lattice code, and a full-core nodal diffusion burnup calculation was performed to validate the eigenvalues and power distributions with reference solutions.

3.3.1.6 RCP01

Gast authored an early study which analyzed a variety of deterministic formulations of the diffusion coefficient to determine the most relevant candidate(s) for Monte Carlo [28]. Gast concluded that most formulations were not practically realizable with MC due to their reliance on spatially integrated flux gradients and current densities. Instead, he deduced that the Selengut-Goertzel diffusion coefficient based on the transport cross section was best suited for computation with MC, and implemented it in the RCP01 MC code [29] developed by Bettis Atomic Power Laboratory. The implementation applied an empirical correction factor to the final computed diffusion coefficient to address the approximation made by using flux-weighted rather than current-weighted tallies. The first author could not find any subsequent analyses in the publicly available literature which validate the implementation for coarse mesh diffusion calculations.

3.3.1.7 VIM

The VIM continuous energy Monte Carlo code is developed by Argonne National Laboratory [30] for neutron and photon transport calculations. VIM is capable of generating isotopic macroscopic or microscopic MGXS, including group-to-group scattering matrices, for deterministic applications. The code is capable of producing diffusion coefficients derived from tallied multi-group total and scattering cross sections. A data processing tool may be used to convert the computed MGXS into the ISOTXS file format accepted by ANL's deterministic diffusion and transport codes. The first author could not find any analyses in the publicly available literature which validate the MGXS generated by VIM for coarse mesh diffusion calculations.

3.3.2 MGXS for Fine-Mesh Transport Calculations

3.3.2.1 MCNP

Redmond [31] performed one of the earliest known in-depth analyses of MGXS generation with MC methods. Redmond studied two methods to compute group-to-group scattering moment matrices with the MCNP code [38]. The first approach – known

as the *direct method* – is equivalent to the analog estimator for computing scattering matrices presented in Sec. 3.2.1.4. The second approach – termed the *explicit method* – imposes a finite sampling of possible outgoing energies and angles for each scattering and fission event. The explicit method may thereby improve the sampling statistics for the scattering cross section and the fission spectrum. To the first author's knowledge, the explicit method is not implemented in any presently available production code nor are there results beyond Redmond's work which evaluate the sampling efficiency of the method.

More recently, Van der Marck [32] applied Redmond's methods in MCNP to generate MGXS to model the Petten High Flux Reactor (HFR) in the Netherlands. Van der Marck was concerned with the slow computational performance of computing MGXS with MCNP as a result of using track-length tallies. Although track-length tallies are an efficient statistical estimator, they require scoring to each tally every time a neutron travels between material zones, which can be exceedingly slow in MCNP. Instead, Van der Marck created a downstream data processing tool called ELNINJO to analyze MCNP binary files with collision data to compute collision estimators for MGXS generation. The ELNINJO code was developed to generate either MGXS for diffusion or transport theory codes, and was validated for a few simple 1D and 2D test cases.

Hoogenboom employed a similar downstream data processing approach in a tool to parse MCNP's PTRAC event file to compute multi-group scattering matrices [33]. The tool permitted the computation of multi-group scattering matrices within arbitrary geometric regions with no changes necessary to the MCNP source code or input files. The tool was used to generate 2-group constants in a simple 1D slab geometry which were then validated with respect to those generated with the deterministic SCALE code system [47]. There are no known published results to validate the efficacy of the MGXS generated from the tool in a downstream fixed source or eigenvalue calculation code.

Most recently, Yoshioka evaluated a methodology for MGXS generation with MCNP for deterministic diffusion or transport methods [34, 35]. Yoshioka developed a *weight-to-flux ratio* method to compute scattering matrices as an extension to an earlier study by Tohjoh with the MVP-BURN code [27]. The weight-to-flux ratio scheme was derived as a

simple tally scheme for 3-group down-scatter cross sections and neglected up-scattering and self-scattering. The MGXS generated with MCNP were validated with respect to those computed using a reference deterministic lattice code. In addition, steady-state, burnup and transient calculations were performed to validate eigenvalues, reactivity margins, Minimum Critical Power Ratio (MCPR), and power responses for a variety of BWR benchmark configurations. Interestingly, the MGXS generated by MCNP led to a negative bias of 300–600 per cent mille (pcm) in deterministic transport calculations for a BWR fuel pin cell model. Although the source of this bias was not identified, the results presented in Chap. 5 indicate that it may have been due in part to the flux separability approximation (see Sec. 2.3.2).

3.3.2.2 TRIPOLI-4

Cai [36] investigated the use of the TRIPOLI-4 code to generate MGXS with continuous energy Monte Carlo simulations. Unlike other published works, Cai validated the MGXS using the multi-group Monte Carlo solver in TRIPOLI-4 rather than a deterministic multi-group solver. Her thesis primarily focused on methods to enforce neutron balance between reference continuous energy and multi-group Monte Carlo calculations. Cai developed a method termed the "In-Group Scattering Correction" (IGSC) as a means to mitigate the flux separability approximation discussed in Sec. 2.3.2. The IGSC technique employed volumetric neutron current-weighted moments of the total cross section as correction terms to the diagonal entries in the scattering moment matrices.

Cai explored IGSC with a MC estimator of Todorova's current approximation for a variety of homogenized 2D and 3D fast reactor benchmarks. Although the IGSC method improved the consistency between continuous energy and multi-group MC results for some cases, it led to a several thousand pcm eigenvalue bias for a few of the benchmarks. Furthermore, Cai's results demonstrated the deficiency of using Todorova's current approximation in a heterogeneous geometry with strongly varying material properties. Cai developed and evaluated an alternative approximation to the current, termed the "Direction-X" current, to eliminate the approximations made in Todorova's current – namely, that the gradient of the flux is similar to the flux spectrum itself. Although the

"Direction-X" current greatly improved the results with respect to Todorova's current, its implementation and analysis was limited to 1D slab geometries.

3.3.2.3 OpenMC

Nelson [37] identified the convergence rate of scattering moments to be a key bottleneck to computing MGXS libraries, and developed an approach to mitigate this with deterministic tallies of the energy and angle for outgoing neutrons in collisions. Nelson's methodology permits track-length estimators for tallies which depend on the outgoing neutron energy, and scores to all outgoing energy groups for each particle trajectory. This is advantageous since it greatly improves the tallying efficiency and convergence rate of scattering moment matrices and the fission energy spectrum. However, Nelson's methodology requires a computationally expensive pre-processing step to transform each nuclide's nuclear data for use in an MC simulation. Furthermore, the pre-processing step is specific to the energy group structure used in an MGXS library, and the computational expense and memory footprint grow as $\mathcal{O}(NG^2)$ with the number of groups G and nuclides N in a simulation. Although Nelson implemented the scheme in a developmental version of OpenMC, it was not utilized by the first author in this thesis work.

3.3.3 Concluding Remarks

A large body of recent work has aimed to replace resonance scattering and lattice physics codes to instead compute MGXS with Monte Carlo for coarse mesh diffusion codes. These studies have primarily explored various formulations to tally diffusion coefficients, and various models to account for leakage in assembly homogenized few-group constants. In comparison, relatively less consideration has been given to MGXS generation for fine-mesh transport calculations. These efforts have focused on improving the statistical efficiency of MC tallies for scattering moment matrices and the fission spectrum, and alternatives to the flux separability approximation for more accurate total cross sections.

The work to date has demonstrated the promise for MC to replace many of the complicated steps and approximations made in the traditional multi-level framework

for MGXS generation. However, past studies have typically generated MGXS for subsets of a reactor geometry – such as individuals fuel assemblies – for subsequent use in multi-group full-core analysis. This thesis aims to build upon the progress made in this area by investigating the opportunity to directly use full-core continuous energy MC simulations to compute MGXS for full-core deterministic multi-group calculations. The subsequent chapters of this thesis develop and evaluate a novel methodology for statistically efficient spatial homogenization which accounts for self-shielding effects in MGXS tallied on a fine spatial mesh.

Highlights

- MGXS are computed from a mixture of analog and track-length Monte Carlo reaction rate and flux tally estimators. Error propagation theory is used to estimate the variance of MGXS computed from MC tallies.
- MC is increasingly used to generate few-group constants for coarse mesh diffusion calculations with tools such as the Serpent MC code.
- MC-based MGXS generation methods to date rely on a multi-level framework similar to that employed in conventional deterministic methods. These approaches generate MGXS for sub-assembly geometries with infinite boundary conditions to generate few-group constants for full-core analysis.
- Multi-level methods suffer from approximations used to treat neutron leakage and spatial heterogeneities such as neutron reflectors.
- Less attention has been directed to MC-based MGXS generation for fine-mesh transport calculations – the focus of this thesis.

Chapter 4

Simulation Workflow

4.1 A Simulation Triad

This thesis investigates Monte Carlo as a means to generate multi-group cross sections for fine-mesh transport codes. This work required the development of a "simulation triad" encompassing three primary codes as illustrated in Fig. 4-1. First, the OpenMC Monte Carlo code [45] was utilized to generate multi-group cross sections. Second, the MGXS were used by the OpenMOC Method of Characteristics (MOC) code [48] for deterministic multi-group transport calculations. Finally, the OpenCG library [49] enabled the processing and transfer of tally data on combinatorial geometry (CG) meshes between OpenMC and OpenMOC. In addition, a significant amount of infrastructural code was developed to process the results produced by OpenMC and OpenMOC. The results in this thesis may be most easily reproduced with the versions of OpenMC, OpenMOC and OpenCG itemized in Tab. 4.1.

Table 4.1: The Git SHA-1 commit hashes for each code used in this thesis.

Code	Date [MM-DD-YYYY]	Git Commit SHA-1
OpenMC	05-30-2016	698c223482a7d5f5df4dc83eeed65d99a2e52fbf
OpenMOC[16]	05-29-2016	b09e66be269703ca0a08d7a6afced1bc5984112a
OpenCG	05-14-2016	2c5e8f92f501d076f4ed06b70b09684a411b06b9

[16] OpenMOC was compiled with double precision floating point arithmetic.

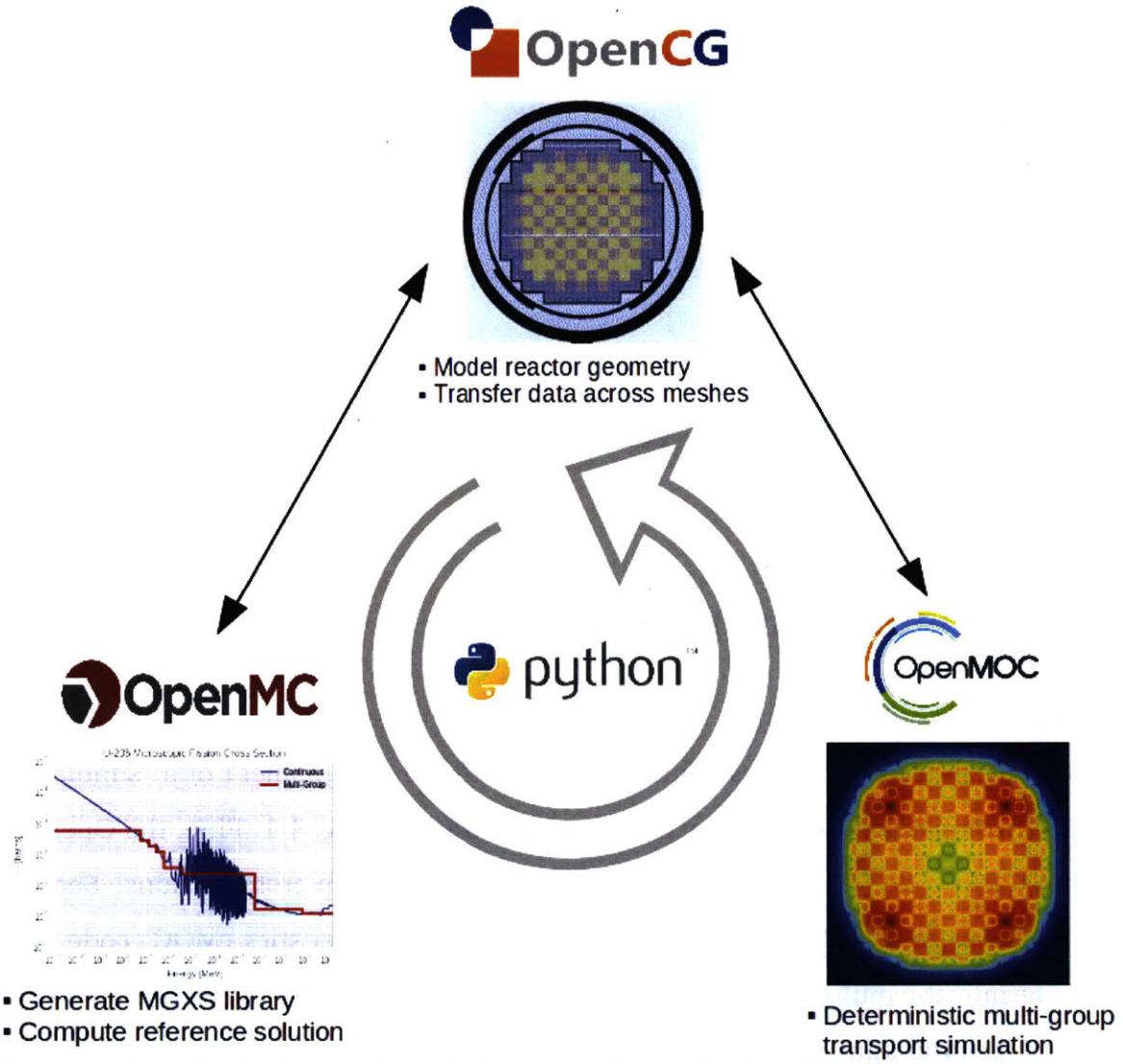


Figure 4-1: A simulation triad consisting of the OpenMC, OpenMOC and OpenCG codes "glued" together with Python formed the foundation for this thesis research.

This chapter describes the author's contributions to each component code in the simulation triad to support the objectives of this thesis – namely, the evaluation of intrinsic bias in MGXS for fine-mesh transport (Chaps. 5 to 6), and the development of a novel methodology for spatial homogenization based on unsupervised clustering (Chaps. 7 to 10). An overview of the OpenMC, OpenMOC, and OpenCG codes, along with the features added to each code to support this thesis, are presented in Secs. 4.2, 4.3, and 4.4, respectively.

A simulation triad consisting of OpenMC, OpenMOC and OpenCG was created to evaluate MGXS generated from MC within a Python-based framework.

4.2 OpenMC

The OpenMC code is a continuous energy Monte Carlo neutron transport code [45] with support for general constructive solid geometry models. OpenMC was initially created by Romano [50] to explore novel parallel algorithms for High-Performance Computing (HPC) architectures. The code was released for public use with the MIT open source license, and has attracted growing interest as a platform for the development of new physics methods and computational algorithms. Although this thesis could have plausibly used any continuous energy MC neutron transport code to generate MGXS, this author chose OpenMC for its general and extensible implementation, excellent parallel scalability, and open source license agreement which permits the modification of its codebase. The general physics and computational methods implemented in OpenMC will not be detailed here since they are well documented in the literature. The interested reader is referred to the online code manual [39] for further information.

This thesis developed new features for OpenMC to enable the processing of large tally datasets to generate MGXS. These contributions were motivated by the novel spatial homogenization technique presented in Chaps. 7 to 10 which required the calculation of microscopic MGXS for each nuclide in each spatial zone across a reactor core geometry. The tally datasets for this scheme are orders of magnitude larger than those generated by the multi-level approaches previously considered in the literature. In particular, the tally datasets are computed on a fine (*e.g.*, pin-wise) spatial tally mesh for a fully-detailed heterogeneous full-core geometry. This stands in contrast to multi-level approaches which compute MGXS for each unique fuel pin or assembly with infinite lattice boundary conditions for use in a multi-group full-core calculation. For example, the scheme introduced here would tally MGXS for each of the $\mathcal{O}(50,000)$ fuel pins in a full-core MC simulation of a PWR. In contrast, a multi-level approach would compute MGXS for the

$\mathcal{O}(10)$ of unique fuel pins or assemblies in the model with pin-wise or assembly-wise MC simulations.

This thesis' requirements for "big data" Monte Carlo calculations can be defined along two primary dimensions: scalable parallel algorithms for efficient MC tracking, sampling and tallying, along with flexible and robust tools for downstream data processing. The first of these dimensions has been a focal point for OpenMC development since its inception. OpenMC includes distributed memory parallelism via the Message Passing Interface (MPI) [51] and has been shown to scale with near perfect efficiency to 100,000s of processor cores [50]. In addition, shared memory parallelism is implemented with the OpenMP library [52] which reduces the simulation memory footprint by minimizing domain replication on multi-core processors. Furthermore, recent work has developed innovative schemes to manage tally datasets with memory footprints beyond that available on a single node in a typical HPC machine (a few tens of gigabytes) with tally servers [53] and spatial domain decomposition [54]. The efficient parallel algorithms already implemented in OpenMC were a key reason to use the code for this thesis work.

However, this thesis did involve the development of software tools to address the data processing needs for "big data" MC calculations. It is the author's opinion that these data processing tools uniquely position OpenMC as the only MC code presently capable of supporting the MGXS generation scheme introduced in Chaps. 7 to 10. For example, many commonly used MC codes store and retrieve tally data from American Standard Code for Information Interchange (ASCII) formatted files or flat binary files. Although these file formats may work well for small tally datasets, they do not scale well for the tally datasets used in this thesis. The data processing paradigm reinforced by many MC codes places a large burden on the user to write convoluted parsers to extract tally data without a generic set of tools to guide the process. In addition, many tally data stores are organized in a way that necessarily serializes tally data access without the metadata needed to index data in an efficient and parallel manner. Furthermore, many data stores are highly tailored to tallies on Cartesian or hexagonal meshes rather than the more complex unstructured meshes needed to generate MGXS for fine-mesh

transport codes. The software tools developed for this thesis attempt to mitigate these issues and formalize a flexible and scalable tally data model for OpenMC.

This section describes the features introduced to develop a next-generation tally data model in OpenMC and follows a recent paper by this author [55]. Sec. 4.2.1 presents a fully-featured Python Application Programming Interface (API) for OpenMC which formed the foundation for much of this work. An algorithm to simplify tally management on unstructured but repeated tally volumes is highlighted in Sec. 4.2.2, and a feature to use isotropic in lab scattering is examined in Sec. 4.2.3. Finally, a new module to generate MGXS was implemented atop many of the newly introduced features in OpenMC as discussed in Sec. 4.2.4. All of the feature implementations were peer reviewed and incorporated into the v0.7.1 release of OpenMC.

4.2.1 Python API

A fully-featured Python API was designed and implemented to enable programmatic pre- and post-processing for OpenMC. The API enables tight coupling of input generation, simulation execution, and tally data analysis within dynamic Python script "input files." In addition, the API makes it possible to leverage the extensive ecosystem of Python packages for scientific computing alongside OpenMC in a simulation workflow. The following sections describe the API and some of the core features which comprise the software stack developed to support the MGXS generation module created for OpenMC.

4.2.1.1 Overview

The Python API is a user-friendly, complementary (and optional) addition to the OpenMC codebase. OpenMC is written in Fortran 2008 and uses eXtensible Markup Language (XML) input files to describe the simulation materials, geometry, tallies, and settings. Although XML is often hailed as both human-readable and machine-readable, it is cumbersome to write by hand for large and complicated reactor models such as those modeled in this thesis. The Python API circumvents this process by leveraging Python's internal `ElementTree` API to generate the XML files used by the OpenMC executable.

Instead of writing XML files by hand, dynamic Python scripts are used to describe one or more OpenMC simulations, including those used to generate MGXS with OpenMC.

The OpenMC Python API adheres to object-oriented software design principles with extensible class definitions. A user instantiates, manipulates, and connects objects representing items such as the materials, geometry and tallies to construct an OpenMC simulation. This is a scalable alternative workflow to traditional "decks" of "cards" in which data characterizing a simulation is specified in opaque ASCII files (*e.g.*, integer identifiers for geometric primitives such as surfaces, cells, universes, etc.). The Python API provides classes and routines to represent all features provided by OpenMC's XML input specifications.

In addition to its functionality for input generation, the Python API also includes a rich framework of tally data processing utilities. The API eliminates the time intensive and error prone process of writing code to parse results from OpenMC's output files. The API is able to reconstruct the hierarchy of interconnected Python objects used to represent the materials, geometry and tallies from OpenMC's "statepoint" and "summary" Hierarchical Data Format 5 (HDF5) output files [56]. OpenMC's dynamic object-oriented data processing model – fusing the geometry and materials configuration with tallied data – enabled the rapid calculation, indexing and storage of MGXS from tallies on unstructured meshes for this thesis.

4.2.1.2 Pandas DataFrames

The Python API encapsulates numerical tally data using N-dimensional array objects from the NumPy package [57]. Although OpenMC's NumPy interface to tally data is more flexible than simply reporting the data in ASCII files, NumPy arrays are relatively opaque containers for managing large tally datasets. A single OpenMC `Tally` object used for MGXS generation may encompass many different energy groups, nuclides and reaction types, yet all of this data is tabulated in a single contiguous NumPy array. As a result, it is challenging to implement general algorithms to inspect, index, and manipulate tally data in NumPy arrays for specific groups, nuclides or reactions.

The Pandas Python package [58] was implemented in the Python API to enable

transparent tally data processing for MGXS generation. In particular, the `Tally` class includes a feature to construct a Pandas `DataFrame` object from tally data. Pandas DataFrames are modeled after data structures in the R programming language used to store data tables in a more accessible format than contiguous arrays. Pandas DataFrames support mixed-type data (*i.e.*, strings and numbers), and allow the use of string keys or labels to index each column or row. The Python API builds Pandas DataFrames by annotating tally data with the filters, nuclides, and scores associated with each tally bin.

Pandas' most advanced features are intended for scalable data manipulation operations such as sorting, merging, and joining datasets, and generating pivot tables. In addition, Python's powerful statistics and machine learning packages – such as SciPy [59], statsmodels [60] and scikit-learn [61] – are well integrated with Pandas and may be easily applied to `DataFrames` of tally data. Pandas DataFrames were extensively used to encapsulate tally data throughout the statistical data processing framework for the MGXS spatial homogenization methodology introduced in Chaps. 7 to 10.

4.2.1.3 Tally Slicing and Merging

Two useful and related features in the OpenMC Python API for MGXS generation are *tally merging* and *tally slicing* as depicted in Fig. 4-2. It is intuitively useful to systematically create individual `Tally` objects for each spatial zone and reaction type when generating the OpenMC inputs necessary to compute MGXS. However, this necessarily leads to a large number (10^2 – 10^3) of distinct tally objects for large, complex geometries, which poses a computational bottleneck since the overhead to tally in OpenMC scales as $\mathcal{O}(N)$ for N tallies.

To compensate for this, the Python API's `Tally` class automatically merges user-specified tallies for input generation. Similarly, the API supports the slicing of tallies to simplify downstream data processing which may comprise energy-, nuclide-, and/or reaction-dependent transformations of the tally data. Tally merging and slicing are extensively used throughout the statistical data processing framework for the MGXS spatial homogenization methodology introduced in Chaps. 7 to 10.

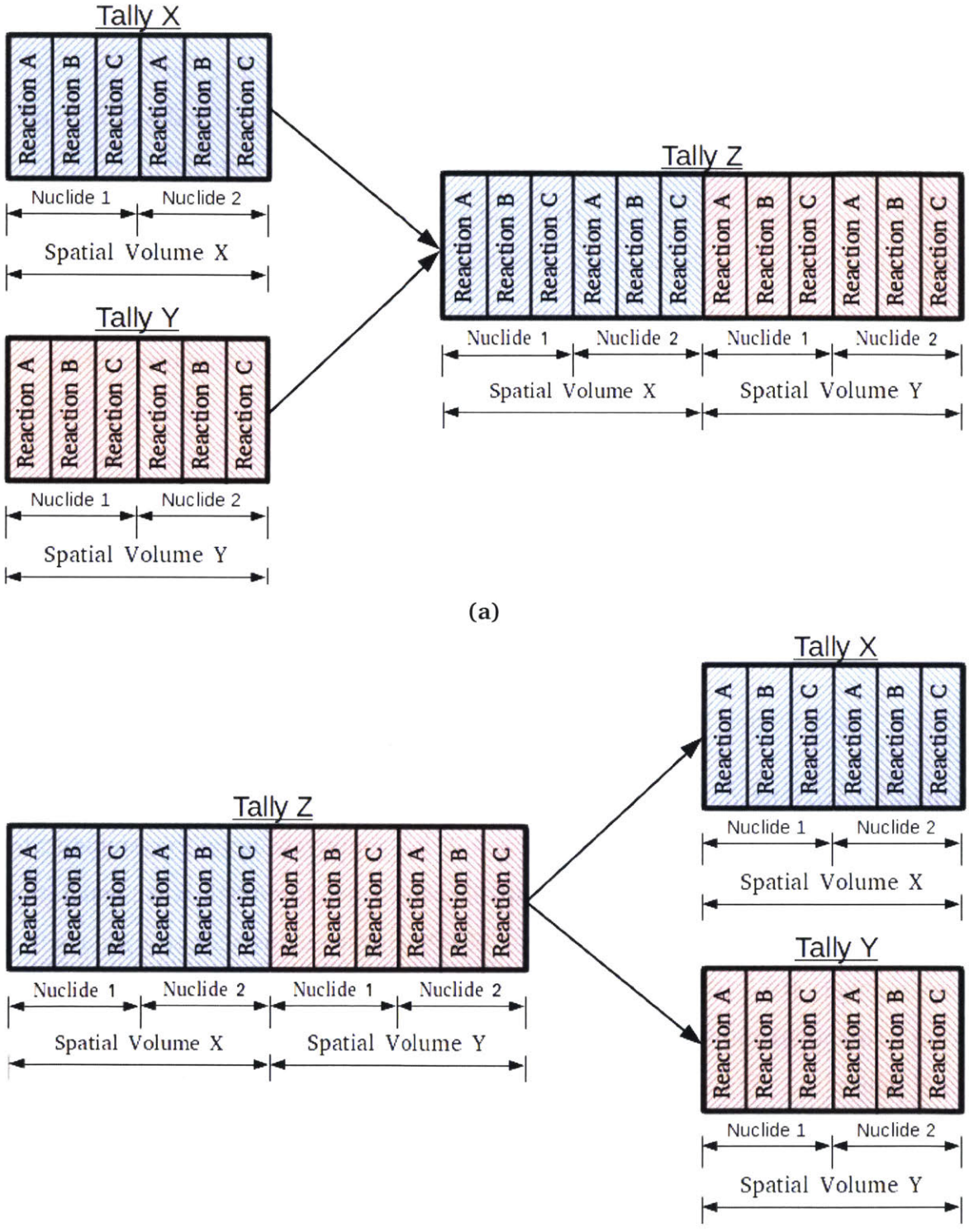


Figure 4-2: Two `Tally` objects for different spatial volumes are merged into a single `Tally` (a). A single `Tally` is sliced by spatial volume into two distinct `Tally` objects (b).

4.2.1.4 Tally Arithmetic

As discussed in Sec. 3.2.1, a variety of reaction rate and flux tallies must be arithmetically combined in order to compute MGXS with Monte Carlo (see Sec. 3.2.1). At the most general level, a reaction rate tally must be divided by a flux tally for each energy group, nuclide and tally volume (see Eqn. 3.14). In addition, the transport correction must be subtracted from the total cross section and scattering matrix (Eqns. 3.18 and 3.21), and a summation must be performed over energy groups to compute the fission emission spectrum (Eqn. 3.24). Furthermore, it is desirable to compute a variance estimator for each MGXS by propagating uncertainties as described in Sec. 3.2.2. The Python API provides a novel feature known as *tally arithmetic* to enable arithmetic combinations of tallies with efficient vectorized numerical operations across energy groups, nuclides and spatial tally zones.

Tally arithmetic is an object-oriented data processing feature which arithmetically combines two or more tallies and/or scalar values into new *derived tallies*. The objective of tally arithmetic is to rapidly transform tally data with automated uncertainty propagation. The tally arithmetic implementation in OpenMC overloads the operators for addition, subtraction, multiplication, division, and exponentiation in the Python API's `Tally` class. In addition, the `Tally` class supports summation or averaging operations across some or all of its filter, nuclide or score bins. The derived tallies produced from tally arithmetic provide the same rich functionality available for the `Tally` operands used in the arithmetic operation (*e.g.*, Pandas DataFrames, tally arithmetic).

Multi-group cross sections may be simply and efficiently computed with tally arithmetic. For example, the following code snippet illustrates how tally slicing and arithmetic are used to compute a total MGXS:

Listing 4.1: MGXS calculation with tally arithmetic.

```
import openmc

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

# Extract reaction rate and flux Tally objects
```

```
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
```

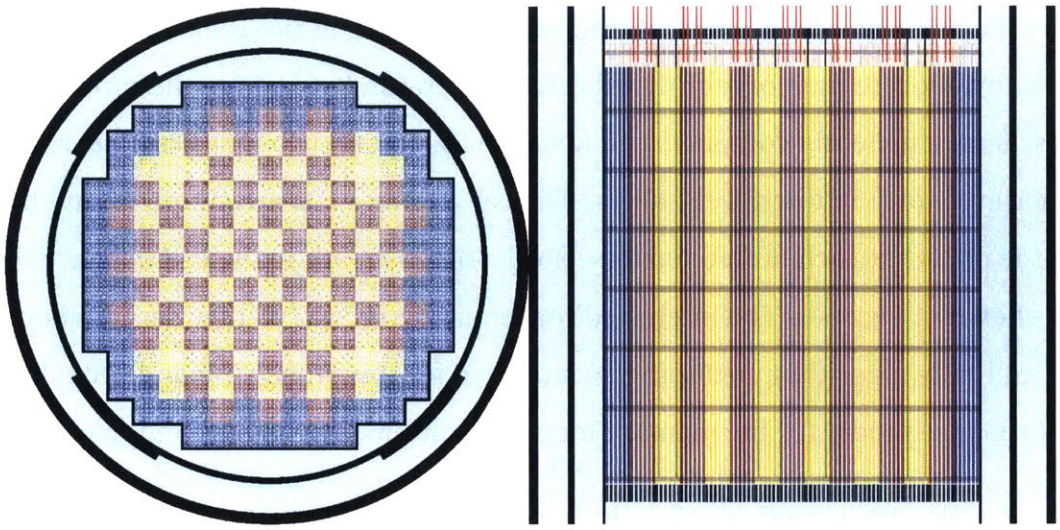
The total MGXS that is returned from the tally division operation is encapsulated within a `Tally` class. This is the approach used by the MGXS generation module created for OpenMC in Sec. 4.2.4.

It should be noted that the uncertainty propagation in tally arithmetic makes the assumption that tallies represent independent random variables (see Sec. 3.2.2). However, in many cases this assumption is untrue as tallies may be highly correlated. For example, there is a strong correlation between the flux and reaction rate tallies across the same material or cell, but this is not accounted for when these tallies are combined to compute MGXS with tally arithmetic. In the future it may be possible to improve this approximation with the inclusion of tally covariance matrices in tally arithmetic.

4.2.2 Distributed Cell Tallies

Many Monte Carlo codes, including OpenMC, use some variant of combinatorial geometry because it can represent arbitrary, repeating geometries such as fuel pins and assemblies. However, the Combinatorial Geometry (CG) approach is challenged by applications which require tallies in each instance of a repeated cell throughout a reactor geometry, such as the Benchmark for Evaluation and Validation of Reactor Simulations (BEAVRS) benchmark model [62] depicted in Fig. 4-3. The "brute force" solution is to instantiate a unique cell for each distinct tally zone. However, this defeats the purpose of using CG for its compact representation, and it is not scalable to problems with large tally datasets such as those considered in this thesis.

The *distributed cell tally* algorithm was implemented in OpenMC for D. Lax's M.S. thesis [63] to permit simply defined spatial tally zones across repeated cell instances. The



(a) (b)

Figure 4-3: The radial (a) and axial (b) views of the BEAVRS PWR core geometry [62]. The distributed cell tally algorithm provides a simple and efficient interface to tally within each of the 50,000+ fuel, guide tube, and burnable poison pin cells in complex heterogeneous models like the BEAVRS core.

distributed cell algorithm, commonly abbreviated as the *distribcell* algorithm, classifies each unique cell instance using *maps* and *offsets* which consume orders of magnitude less memory than would be required by the "brute force" approach. Only a single transparent line of XML input is necessary to define a distribcell tally which may span across an arbitrary number of instances for a particular cell. Furthermore, the Python API may be used to perform efficient vectorized transformations of distribcell tally data stored as contiguous NumPy arrays. The distribcell tally algorithm was used to compute spatially-varying MGXS across fuel pin cell instances for the spatial homogenization methodology introduced in Chaps. 7 to 10.

The distribcell tally algorithm consists of pre-processing and indexing stages. The pre-processing phase builds a mapping to index unique regions based on the unique combination of cells, universes and lattices used to construct each region. This is done by storing offset numbers in the data structures for each fill cell[17]. The offset maps are recursively constructed starting from the top-level universe and proceeding through each lower nested universe level in the combinatorial geometry. The pre-processing

[17]Any cell that is filled with a nested universe or lattice of cells.

algorithm tabulates the total number of instances of each cell in the geometry in order to dynamically allocate memory for distribcell tally arrays. At runtime, the on-the-fly indexing scheme efficiently computes cell instance IDs in order to index and score to the appropriate bin(s) in distribcell tallies. The cell instance IDs are simply found by summing the offsets at each nested universe level along the path to the cell instance in the CG as shown in Fig. 4-4. The interested reader is referred to [63] for details about the distribcell tally algorithm, including pseudocode for pre-processing and indexing and a performance model for the memory footprint consumed by offsets and maps.

4.2.3 Isotropic in Lab Scattering

As part of this thesis, a unique option for isotropic in lab scattering was implemented in the OpenMC code. The isotropic in lab (abbreviated as iso-in-lab) feature may be useful to quantify the ability of multi-group codes to capture anisotropic scattering effects with higher order scattering matrices or transport correction schemes (see Sec. 2.2.3). The iso-in-lab scattering feature was implemented as a "scattering" attribute for each nuclide/element in a simulation. When iso-in-lab scattering is specified for a nuclide/element, the outgoing neutron energy is sampled from the scattering laws prescribed by the continuous energy cross section library, but the outgoing neutron direction of motion is sampled from an isotropic in lab distribution.

Unless otherwise noted, isotropic in lab scattering was employed in OpenMC to generate the MGXS used in this thesis. Isotropic scattering is generally not a valid approximation for nuclear reactors and may induce large errors for thermal systems such as the PWR benchmarks modeled in this thesis. However, the iso-in-lab scattering feature enabled "apples-to-apples" comparisons between the reference eigenvalues and reaction rates produced by OpenMC and those computed from isotropic multi-group calculations with OpenMOC.

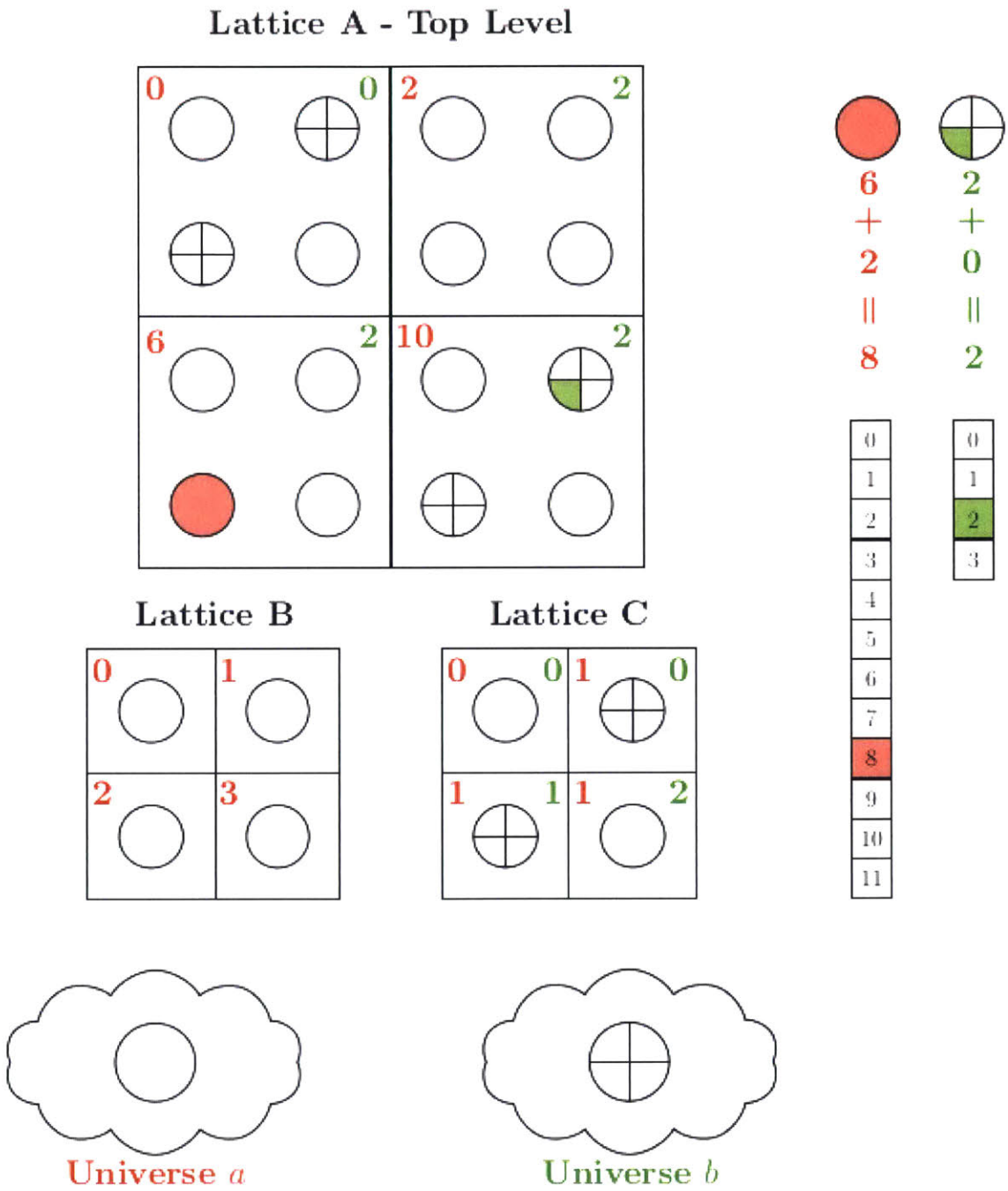


Figure 4-4: An example of the distribcell tally algorithm's on-the-fly indexing scheme [63]. Material-filled cells are defined in pin cell universes *a* and *b*, which are filled into the cells of lattices *B* and *C*, which are filled into the cells of lattice *A*. The colored numbers in each fill cell are the offsets for each base universe, which can be used to quickly compute a unique ID for each instance of a material cell.

4.2.4 MGXS Generation

The OpenMC Python API's `openmc.mgxs` module was implemented to generate multi-group cross sections. The `openmc.mgxs` module is built atop the underlying core features in the rest of the API to support a seamless interface for both input generation and downstream data processing of MGXS from Python. In particular, one may specify the MGXS to compute and the `openmc.mgxs` module will construct the necessary `Tally` objects. The `Tally` objects may be easily exported to XML input files for OpenMC, and used to containerize and process the tally data produced by an OpenMC simulation. The `openmc.mgxs` module thereby leverages the software stack of Pandas DataFrames, tally arithmetic, etc. provided by the OpenMC Python API.

The `openmc.mgxs` module can compute macroscopic or microscopic MGXS for individual nuclides or elements as well as collections of nuclides and elements[18]. MGXS can be computed in one or more arbitrary energy group structures. The module supports energy condensation in downstream data processing which is useful for exploring approximation bias in various energy group structures. For example, MGXS may be computed in a 16-group structure and the tally data subsequently condensed to 2-group, 8-group, 12-group, etc. structures for multi-group calculations[19]. The analysis in this thesis computed MGXS in the 70-group structure provided in Tab. A.8, and condensed the MGXS to the coarser group structures given in Appendix A as needed.

The `openmc.mgxs` module is designed to perform spatial homogenization on heterogeneous tally meshes for fine-mesh transport codes. In OpenMC parlance, MGXS may be computed for material, cell or universe spatial domains. In addition, the module supports MGXS calculations for repeated cell instances using distribcell spatial tally domains (see Sec. 4.2.2). The `openmc.mgxs` module may also perform spatial homogenization on structured Cartesian tally meshes for coarse mesh multi-group calculations. This thesis computed MGXS using distribcell tallies (*e.g.*, MGXS for each fuel pin instance in a geometry) to support the MGXS spatial homogenization introduced in Chaps. 7 to 10.

[18] A MGXS computed for a collection of nuclides/elements is the sum of the individual contributions from each nuclide/element.

[19] Energy condensation may be performed to arbitrarily defined coarse group structures with `openmc.mgxs` provided the coarse group boundaries coincide with boundaries in the fine group structure.

The `openmc.mgxs` module uses an object-oriented design based on an abstract `MGXS` class with subclasses for different reaction types. The `MGXS` subclasses are itemized in Tab. 4.2 and compute multi-group constants from MC tallies using the methods detailed in Sec. 3.2.1. It should be noted that some reaction types include variants which do or do not account for scattering multiplicity reactions (see Sec. 2.3.3). For example, the transport correction used by the `TransportXS` class does not include (n, xn) reactions while that employed by the `NuTransportXS` does account for scattering multiplicity. The `openmc.mgxs` module also includes a `Library` class which automates the construction of `MGXS` objects for different group structures, spatial domains, and reaction types.

Table 4.2: The MGXS types implemented by the `openmc.mgxs` module in OpenMC.

Class	Description
`TotalXS`	Total collision
`TransportXS`	Transport-corrected total collision
`NuTransportXS`	Transport-corrected total collision w/ scattering multiplicity
`AbsorptionXS`	Absorption
`CaptureXS`	Radiative capture
`FissionXS`	Fission
`NuFissionXS`	Fission neutron production
`KappaFissionXS`	Fission energy release
`ScatterXS`	Scattering
`NuScatterXS`	Scattering w/ scattering multiplicity
`ScatterMatrixXS`	Scattering matrix
`NuScatterMatrixXS`	Scattering matrix w/ scattering multiplicity
`Chi`	Fission emission spectrum
`ChiPrompt`	Prompt fission emission spectrum

The `openmc.mgxs` module was developed with general design principles to generate MGXS for any multi-group neutron transport code. Although the module does not explicitly support any multi-group codes, it can export MGXS data to a variety of data storage formats, including Comma-Separated Values (CSV) and HDF5. The exported MGXS files may be easily transformed into the database or input files required by a particular multi-group code. As discussed in the following section, this thesis developed a tightly integrated framework to pipeline MGXS generated by `openmc.mgxs` into the

multi-group OpenMOC code.

The OpenMC code was used to generate MGXS. A Python API was implemented for input generation and data processing, along with features including distributed cell tallies and isotropic in lab scattering. The `openmc.mgxs` Python module was created to generate MGXS from OpenMC tallies.

4.3 OpenMOC

The OpenMOC code is a multi-group neutron transport code implementing the deterministic Method of Characteristics (MOC) [48]. OpenMOC was initially developed to support a series of M.S. theses at MIT [64–66] and was later released for public use with the MIT open source license. Although this thesis could have plausibly used any multi-group code, OpenMOC was chosen for its flexible Python interface, computationally efficient algorithms, and the author's familiarity with the open source codebase.

This thesis inspired the development of new features and extensions which have been incorporated into the official OpenMOC codebase. This section begins with a brief overview of the MOC implementation in OpenMOC in Sec. 4.3.1. Various improvements were made to the OpenMOC Python interface as discussed in Sec. 4.3.2, while Sec. 4.3.3 details a module created to interface with MGXS generated by OpenMC. The parallel algorithms and numerical acceleration schemes in OpenMOC which were used by this thesis are mentioned in Secs. 4.3.4 and 4.3.5, respectively.

4.3.1 Methods Overview

The method of characteristics is a widely used technique for solving partial differential equations, including the Boltzmann form of the neutron transport equation [67]. Although not a stochastic formulation, MOC is a ray-based algorithm akin to Monte Carlo particle tracking-based methods. In contrast to Monte Carlo, MOC uses a fixed angular quadrature that is determined *a priori*. This quadrature is used to specify 1D character-

istics that cross the spatial domain. Prior to the physics computation, ray tracing must be performed to subdivide each characteristic into segments within different regions in the spatial mesh. Fig. 4-5 illustrates the spatial mesh and cyclic characteristic laydown used by the OpenMOC code.

MOC propagates the angular neutron flux along each characteristic through each spatial zone. For each segment, the angular flux is attenuated due to neutron absorption and enhanced due to neutron fission or scattering in the corresponding spatial zone. MOC uses the multi-group energy approximation such that this computation is performed for neutrons within discretized energy groups (see Sec. 2.3.1). Finally, an angular quadrature is applied to combine the average angular flux contribution from each characteristic to compute the average scalar flux in each zone and energy group.

(a) (b) (c)

Figure 4-5: The coolant and fuel materials (a), flat source region spatial mesh (b), and cyclic characteristic laydown (c) for a 4 × 4 fuel pin lattice taken from [68].

The OpenMOC code is capable of performing 2D MOC calculations for light water reactor core configurations. OpenMOC discretizes the 2D geometry into flat source regions (FSRs) which approximate the neutron source as constant across each spatial zone[20]. Furthermore, OpenMOC approximates the scattering source as isotropic in the lab coordinate system and is not yet able to use higher order scattering moment matrices. OpenMOC uses a power iteration scheme to solve for the dominant eigenvalue and eigenvector in criticality calculations. As part of this thesis, a general purpose fixed source solver was implemented to enable the detailed investigation of approximation

[20]The neutron source may include any combination of fission, scattering and fixed sources.

bias in MGXS as discussed in Chap. 6. The interested reader is referred to the online code manual [69] for more information detailing the MOC implementation in OpenMOC.

4.3.2 Python Interface

One of the key reasons that OpenMOC was used by this thesis was to leverage its flexible Python interface. The majority of the source code is written in C++ using object-oriented software design principles. The Simplified Wrapper Interface Generator (SWIG) [70] is deployed to expose the C++ classes and routines to the Python scripting language. The Python interface made it possible to tightly couple OpenMOC with the rich ecosystem of Python-based data processing and visualization tools for the spatial homogenization methodology developed in Chaps. 7 to 10.

As part of this work, a new scheme was implemented to efficiently link OpenMOC's Python and C++ CG data structures into a hierarchical tree-like data structure. This made it possible to use the OpenCG region differentiation algorithm discussed in Sec. 4.4.3 to derive geometries for OpenMOC. In addition, a thread-safe memory management model was realized which unifies the dynamic deallocation of objects shared between Python and C++ code. The memory model enabled 100s of OpenMOC simulations to be orchestrated by Python for the parametric studies considered in this thesis.

4.3.3 Multi-Group Cross Sections

OpenMOC uses multi-group macroscopic nuclear cross sections specified in any arbitrary energy group structure. Isotopic concentrations are not used since OpenMOC does not perform self-shielding or depletion calculations. Multi-group cross sections may be specified for each material or cell in the CG used in a simulation. OpenMOC requires total, fission production and scattering matrix cross sections along with a fission emission spectrum[21]. In addition, a fission cross section may be optionally supplied in order to compute spatially-varying fission reaction rates.

Cross section data is encapsulated by the `Material` class. A `Material` object may be

[21] Transport-corrected total cross sections and scattering matrices may be used in OpenMOC.

instantiated in Python and cross section data loaded into it from NumPy arrays [57]. For simulations with many different materials (such as those in this thesis), defining nuclear cross section data by hand in a Python script is cumbersome and error prone. In order to minimize this painstaking process, the `openmoc.materialize` module was implemented to automate the loading MGXS data into OpenMOC `Material` objects. This module can either import MGXS data from HDF5 binary files [56] or extract MGXS data from OpenMC `Library` Python objects (see Sec. 4.2.4). The `openmoc.materialize` module is designed to support large MGXS libraries such as those produced from the spatial homogenization methodology introduced in Chaps. 7 to 10. In addition, a scheme to compute SPH factors to ensure reaction rate consistency with OpenMC was implemented in `openmoc.materialize` and is discussed in detail in Chap. 6.

4.3.4 Parallelism

OpenMOC's high performance parallel solvers for both multi-core Central Processing Units (CPUs) and Graphic Processing Units (GPUs) were extensively used for the analyses presented in this thesis. A shared memory parallel solver implemented in C++ using OpenMP [71] has demonstrated excellent scalability on a range of multi-core architectures [68]. In addition, OpenMOC includes a highly parallel implementation which uses the CUDA programming language [72] to run on NVIDIA GPUs with 100s to 1000s of lightweight cores [73]. OpenMOC's parallel solvers made it possible to perform large parametric studies of 100s of MOC simulations to evaluate the spatial homogenization methodology introduced in Chaps. 7 to 10.

4.3.5 CMFD Acceleration

OpenMOC uses the Coarse Mesh Finite Difference (CMFD) acceleration scheme to greatly reduce the number of iterations required to converge criticality calculations [48]. CMFD acceleration functions by using the solution of a coarse mesh diffusion problem to accelerate the convergence of the fine-mesh MOC transport problem. The details of the CMFD implementation in OpenMOC are beyond the scope of this thesis, and the

interested reader is referred to the online code manual [69] for more information.

CMFD acceleration was extensively used to accelerate the eigenvalue calculations in this thesis. All simulations which employed CMFD used the novel k-Nearest Neighbors prolongation scheme created by Shaner [74] and implemented in OpenMOC to improve the convergence rate and stability of CMFD. In particular, all simulations used three neighbors along with a successive over-relaxation (SOR) factor of unity. Along with OpenMOC's parallel solvers, CMFD acceleration made it feasible to perform large parametric studies to generate the results presented in this thesis.

The OpenMOC code was used to perform 2D deterministic multi-group MOC calculations. The `openmoc.materialize` module was improved to support MGXS generated from OpenMC. OpenMOC's parallel algorithms and CMFD acceleration enabled parametric studies of 100s of OpenMOC simulations.

4.4 OpenCG

The OpenCG code [49] was created to simplify the process of creating and transferring data mapped to combinatorial geometries for OpenMC and OpenMOC. Combinatorial geometry[22] is commonly used by neutron transport simulation codes since it:

- Permits description of an *arbitrarily accurate* unstructured geometric mesh
- Provides a *compact representation* with minimal input description
- Represents $\mathcal{O}(n)$ components with $\mathcal{O}(\log n)$ memory requirements
- Utilizes a hierarchical *tree data structure* with scalable $\mathcal{O}(\log n)$ traversals

Although many codes utilize CG, it is overly burdensome to manually write geometric input files for multiple simulation tools for code verification of a single reactor model. In addition, the compact geometric representation is not well-suited for large scale analysis of spatially-varying data in nuclear reactor cores – such as distributed cell tally data in OpenMC – without new algorithms to guide and automate the process. This

[22] CG is often referred to as constructive solid geometry in the neutron transport literature.

thesis developed a new CG modeling tool called OpenCG to accelerate the building of complicated reactor geometries, enable rapid cross-code verification and facilitate large scale data processing.

OpenCG is a simple-to-use Python library which may be used to construct a single geometry for use in both OpenMC and OpenMOC. In addition, OpenCG enables data transfer between OpenMC and OpenMOC as illustrated in Fig. 4-1. In particular, OpenCG accommodates the generation of MGXS from distributed cell tallies on OpenMC's CG mesh and maps them to the flat source region spatial mesh used by OpenMOC. A variant of OpenMC's distributed cell tally algorithm (see Sec. 4.2.2) was implemented in OpenCG to make this possible. In addition to transferring MGXS data, OpenCG enabled the comparison of pin-wise reaction rate distributions computed from OpenMC distribcell tallies with those computed on OpenMOC's Flat Source Region (FSR) mesh.

The following sections outline the key features implemented in OpenCG for this thesis and follows a recent paper by this author [49]. Sec. 4.4.1 discusses the compatibility modules which tie OpenCG, OpenMC and OpenMOC into a simulation triad. In addition, two novel algorithms known as Local Neighbor Symmetry (LNS) and region differentiation were developed in OpenCG to enable the spatial homogenization methodology introduced in this thesis, and are presented in Secs. 4.4.2 and 4.4.3, respectively.

4.4.1 Compatibility Modules

The simulation triad of OpenCG, OpenMC and OpenMOC shown in Fig. 4-1 is made possible with compatibility modules. The compatibility modules allow the construction of a single geometry using OpenCG's Python CG primitives for surfaces, cells, universes and lattices. The OpenCG geometry may then be exported to OpenMC or OpenMOC using compatibility modules developed for each code.

For example, OpenMC includes a Python API with object-oriented CG primitives. This API's primitives include routines to directly export themselves to the XML input file format used by OpenMC. The OpenCG-OpenMC compatibility module allows OpenCG's primitives to be transformed into the corollaries within the OpenMC Python API, and

vice versa. Likewise, the OpenCG-OpenMOC compatibility module allows OpenCG's primitives to be transformed into the corollaries within the OpenMOC Python/C++ code. In summary, the compatibility modules enable the rapid, automated exportation of various OpenCG geometries directly to OpenMC and OpenMOC.

OpenCG's object-oriented Python software model permits greater freedom for geometric parameter optimization than can be easily achieved with traditional ASCII or XML input files. In particular, the compatibility module framework enabled a dynamic workflow between OpenMC and OpenMOC for MGXS generation and code verification in this thesis. For example, OpenCG's region differentiation algorithm (see Sec. 4.4.3) was used to produce 100s of geometries composed with different MGXS libraries to evaluate the spatial homogenization methodology introduced in Chaps. 7 to 10.

4.4.2 Local Neighbor Symmetry

One of the unique algorithms implemented in OpenCG explicitly for this thesis is known as Local Neighbor Symmetry (LNS) identification [49]. The LNS algorithm is motivated by this thesis' objective to accurately predict spatial zones that experience similar spectral self-shielding effects. The LNS algorithm performs a systematic analysis of a CG tree data structure to identify neighbor cells, or pairs of cells which are adjacent to one another. The neighbor cells are assembled into a heuristic which groups like spatial zones with common LNS identifiers. The LNS algorithm is analogous to the geometric templates used in lattice physics codes such as CASMO [75] to identify fuel pins which have similar MGXS in a fuel assembly.

The LNS algorithm identifies the unique symmetry for the path to a region in a combinatorial geometry as described in Alg. 4-1. LNS performs a Breadth-First Search (BFS) to find neighbors on each level of the CG tree. For example, BFS is used to find neighbor cells for a particular cell within a universe. Similarly, BFS is used to find neighbor universes adjacent to a particular lattice cell. The neighbor cells and universes on each of the k levels of a CG tree are connected to form a k-partite graph[23]

[23] A k-partite graph is a graph whose graph vertices can be partitioned into k disjoint sets so that no two vertices within the same set are adjacent [76].

as depicted in Fig. 4-6. Finally, the k-partite graph is used as an argument to a hash function to compute the LNS identifier (*e.g*, a non-negative integer) for the particular region represented by the path. This algorithmic formulation is general to any arbitrary combinatorial geometry, including those commonly used to model LWRs with nested rectilinear lattices.

Algorithm 4-1: Local Neighbor Symmetry Identification

```
1:  procedure COMPUTENEIGHBORSYMMETRY(path)
2:      G ← ∅                                          ▷ Initialize empty set for graph
3:      k ← length(path)                               ▷ Find number of independent sets
4:      for i := 1, k do
5:          if type(path[i]) is UNIVERSE then
6:              G ← G ∪ {path[i]}                      ▷ Append universe to graph
7:          else if type(path[i]) is LATTICE then
8:              N ← BREADTHFIRSTSEARCH(path[i])        ▷ Find lattice cell neighbors
9:              G ← G ∪ {N}                            ▷ Append neighbors to graph
10:         else if type(path[i]) is CELL then
11:             N ← BREADTHFIRSTSEARCH(path[i])        ▷ Find cell neighbors
12:             G ← G ∪ {N}                            ▷ Append neighbors to graph
13:         end if
14:     end for
15:     return HASH(G)                                 ▷ Return k-partite graph hash
16: end procedure
```

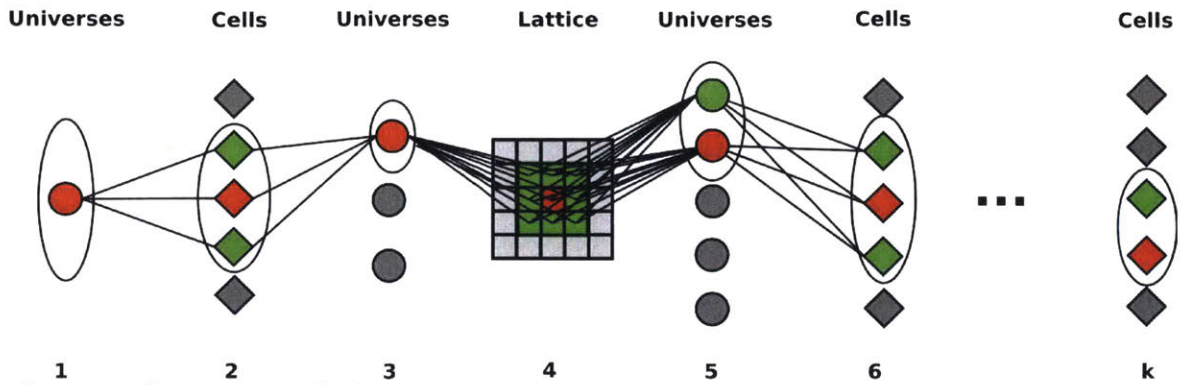


Figure 4-6: An example k-partite graph structure used to identify local neighbor symmetries. Red nodes correspond to the universes/cells encapsulating a region of interest, green nodes correspond to the neighbors of that region, and gray nodes correspond to universes/cells which are not neighbors. Red and green nodes at each level are combined into an argument for a hash function to generate a LNS ID for the region of interest.

Several parameters are incorporated into OpenCG's LNS API to provide the user with various methods to adjust the number of symmetries discovered by the algorithm. For

example, OpenCG includes *general* and *unique* neighbor identifiers. General neighbors includes all neighbor cells/universes found by BFS – including duplicates – as separate nodes in the k-partite graph. For example, a single universe may be placed multiple times around a lattice cell, each instance of which will be independently discovered by BFS and replicated as a distinct node within the graph. On the contrary, unique neighbors does not include duplicate cells/universes – only unique cells/universes are represented by distinct nodes in each independent set in the k-partite graph. A few diagrams of general and unique LNS applied to two geometries are depicted in Fig. 4-7.

To this author's knowledge there is no other implementation of LNS aside from that in the OpenCG code. The LNS algorithm was used extensively in this thesis as a deterministic "clustering" technique for the spatial homogenization methodology introduced in Chaps. 7 to 10. In particular, LNS was used to predict which fuel pins have similar MGXS in LWRs due to spatial self-shielding effects induced by neighboring fuel, guide tube and absorber pins. The LNS scheme served as a proxy to the traditional geometric template approach used in lattice physics codes to group pins with like MGXS. The spatial homogenization methodology developed in Chaps. 7 to 10 incorporates MC tally data in unsupervised clustering in an attempt to outperform LNS' analysis based solely on the geometry.

4.4.3 Region Differentiation

As previously noted, one of the advantages of combinatorial geometry is that it can take advantage of patterned structures with repeating primitives. In certain use cases, however, it may be necessary to replicate certain primitives which have the same geometric properties but experience very different radiation and/or thermal hydraulic conditions, and hence have different material properties in a transport simulation. For example, the LNS algorithm is useful for identifying groups of fuel pin cell instances which may have similar MGXS. However, in order for a transport code to make use of LNS, a replica of each cell must be made to represent each of its different LNS identifiers.

The process of manually constructing a CG with many replicated but geometrically

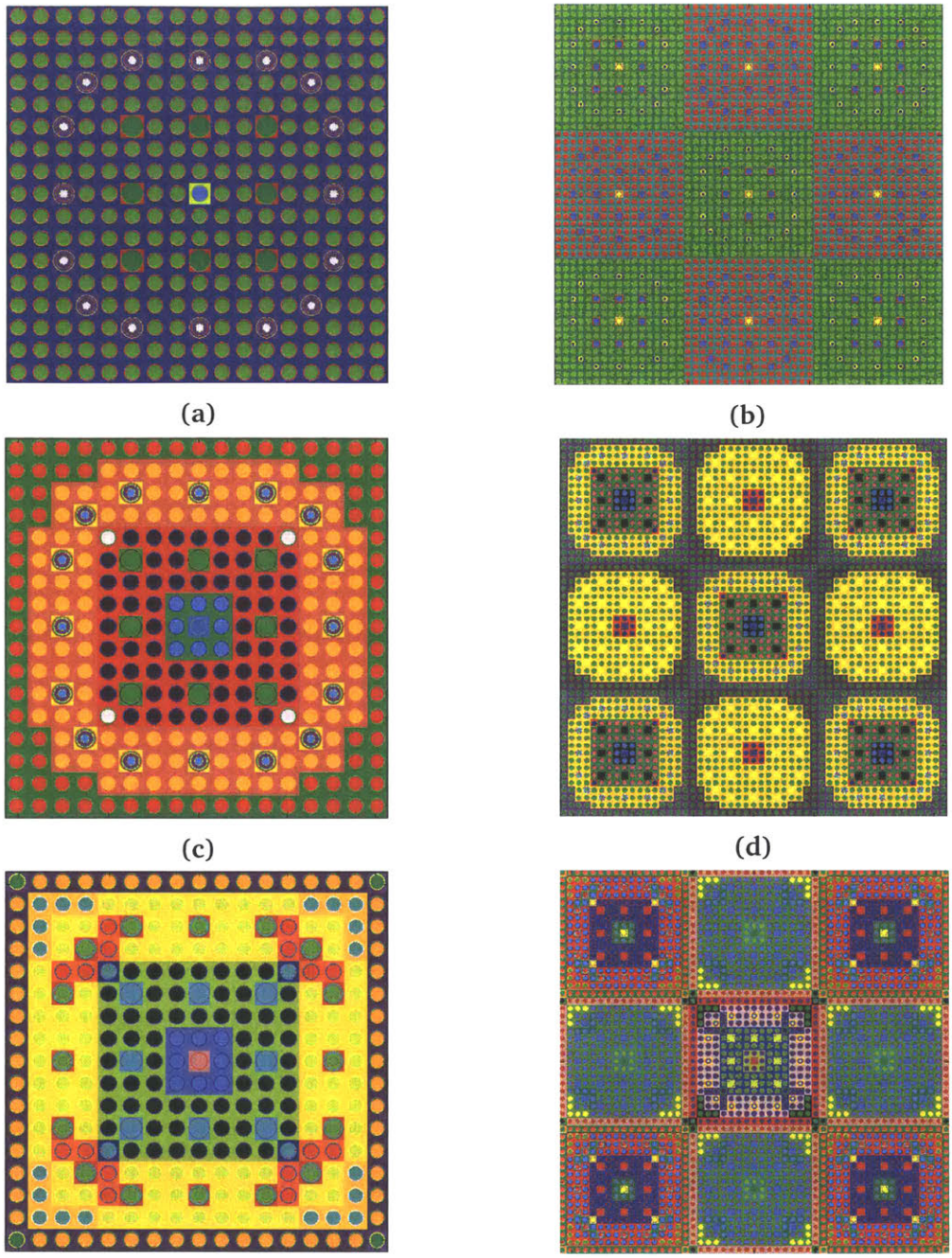


Figure 4-7: Two rectilinear lattice geometries are depicted to illustrate the use of local neighbor symmetry identification [49]. The *cells* are depicted for (a) a 17×17 PWR lattice and (b) a 3×3 colorset of two different 17 × 17 PWR assemblies each with burnable absorbers, guide tubes and instrument tubes. The *unique neighbor* symmetry identifiers are color-coded in (b) and (c) for the assembly and colorset, respectively. Likewise, the *general neighbor* symmetry identifiers are color-coded in (d) and (f).

identical cells is very time consuming and prone to errors. To address this issue, a novel algorithm termed *region differentiation* was implemented in OpenCG to efficiently and systematically reconstruct a CG with replicated cells [49]. A characterization of the region differentiation algorithm applied to a CG tree data structure is shown in Fig. 4-8.

The region differentiation algorithm is implemented in OpenCG and presents an interface which takes in a set of arbitrarily formed *region groupings*. A region grouping is the set of all regions, that reference a particular cell in the geometry. Alternatively, a region grouping can be thought of as a set of repeated instances of a cell throughout a combinatorial geometry. Two or more region groupings corresponding to the same cell designate specific cell instances that should be replicated. The region differentiation algorithm replicates cells, universes, and lattices for each region grouping.

A naive or brute-force implementation of the region differentiation algorithm would scale as $\mathcal{O}(kn!)$ in both memory and time for k nested universe/cell levels and n region groupings. The reason is that *a priori*, the algorithm does not know from which region groupings various cell instances will combine with one another to form universes, lattices and/or cells, which must themselves be differentiated for each possible combination of region groupings. To avoid factorial scaling, OpenCG makes use of dynamic programming to efficiently differentiate primitives one level at a time within the geometry.

The region differentiation algorithm iterates over each level of nested universes and cells. At each step, the paths for each region starting from the current level in the CG tree and ending with the root universe node are hashed and stored in a hash table linked to the region grouping. Next, the algorithm manages *primitive collisions* when two or more region groupings with the same path in the hash table point to the same primitive (cell, universe or lattice). To resolve primitive collisions, the algorithm differentiates the primitive for each region grouping involved in the collision. As primitive collisions are resolved, the algorithm merges any region groupings with paths that hash to same value in the hash table. The algorithm's termination condition is reached when the hash table only has one entry – *i.e.*, paths for all regions hash to the same value.

The region differentiation algorithm was an indispensable component of the simulation triad used to explore novel MGXS generation techniques in this thesis. In particular,

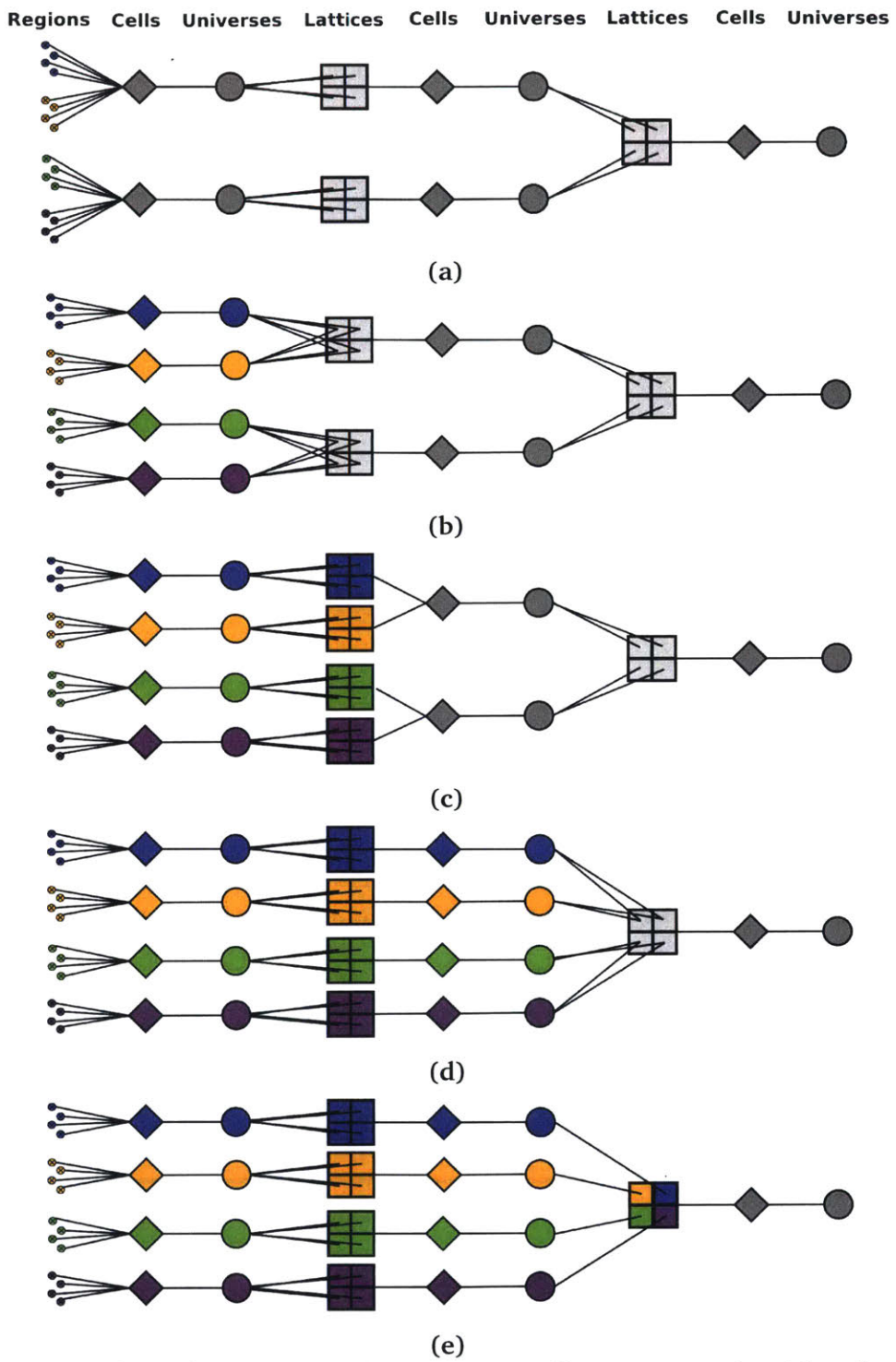


Figure 4-8: A few of the stages of the region differentiation algorithm [49]. The regions (cell instances) to be differentiated are grouped and colored blue, orange, green and purple in (a). The first levels of cells and universes for each region group are differentiated in (b). The same is done for the lattices in (c). The algorithm continues to recursively differentiate cells, universes and lattices until no region groups collide at any level of the CG tree in (e).

region differentiation made it possible to rapidly construct geometries to reflect the assignment of MGXS to arbitrary collections of fuel pins for the spatial homogenization methodology introduced in Chaps. 7 to 10. Fig. 4-9 illustrates a flow diagram where the LNS algorithm identifies the region groupings input to the region differentiation algorithm. The geometry produced from region differentiation may then be exported for use in OpenMC or OpenMOC using the compatibility modules discussed in Sec. 4.4.1.

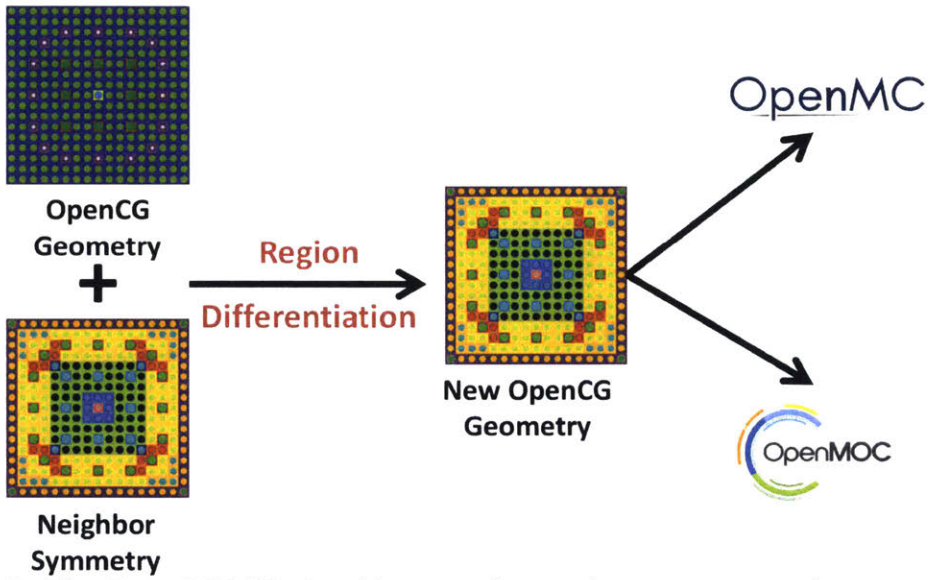


Figure 4-9: The OpenCG LNS algorithm may be used to generate region groupings for the region differentiation algorithm to create a geometry for OpenMC or OpenMOC.

The OpenCG code was implemented to facilitate data transfer between OpenMC and OpenMOC. The LNS and region differentiation algorithms supported the novel spatial homogenization methodology developed in Chaps. 7 to 10.

Highlights

- A framework consisting of the OpenMC, OpenMOC and OpenCG were used to explore MC-based MGXS generation methods for fine-mesh transport calculations.

- A fully-featured Python API was developed to support input generation and downstream processing of large tally datasets for OpenMC.
- The `openmc.mgxs` Python module was created to generate MGXS from OpenMC tallies. The distributed cell tally algorithm was implemented to generate pin-wise MGXS in large, heterogeneous geometries.
- New features were added to the deterministic multi-group OpenMOC code to enable it to use MGXS generated by `openmc.mgxs` in 2D MOC calculations.
- The OpenCG code was created to facilitate data processing and transfer on the combinatorial geometry meshes used by OpenMC and OpenMOC.
- OpenCG's LNS and region differentiation algorithms were crucial components for the spatial homogenization methodology developed in Chaps. 7 to 10.

Part III

Approximation Error

Chapter 5

Quantifying MGXS Approximations

As discussed in Chapter 2, a number of approximations are made in the multi-group formulation of the neutron transport equation and the MGXS generation process. These approximations remain even if one uses the "true" scalar flux for energy condensation and spatial homogenization, as is the case with energy condensation and spatial homogenization in Monte Carlo[24]. The observed bias between a continuous energy MC and a multi-group deterministic calculation reflects the convolution of these approximations, which may (or may not) lead to some degree of fortuitous cancellation of error. Although this thesis is motivated by the need for MGXS for full-core calculations, it is instructive to investigate the approximations inherent in multi-group theory and quantify their impact for simple benchmark models.

In this chapter, a series of case studies is devised to systematically quantify biases inherent to the energy condensation and spatial homogenization process(es) in multi-group transport theory. The results underline the complex interactions between discretizations in energy, space and angle. Various convergence studies with respect to each of these variables are presented to quantify the resulting magnitude of the bias induced between continuous energy Monte Carlo and multi-group deterministic calculations. The results in this chapter illustrate the loss in accuracy resulting from scalar flux-weighted total MGXS due to the flux separability approximation (see Sec. 2.3.2), and highlight the need

[24]By the Law of Large Numbers, the MC flux will be exactly correct in the limit of an infinite number of particle histories, though it is only a "noisy" proxy for a finite number of particles.

for models of the angular dependency in MGXS for fine-mesh deterministic neutron transport. Finally, this chapter verifies the data pipeline used to compute multi-group cross sections with OpenMC for use in OpenMOC (see Chap. 4).

5.1 Case Studies

This chapter investigates the loss in accuracy resulting from approximations made in both the MOC equations as well as the MGXS generation scheme with OpenMC. The benchmarks are designed to illustrate the emergence of the approximation errors as spatial heterogeneity is introduced in the geometric models. The approximation errors are quantified for a variety of geometric and material configurations based on a standard PWR. In each case, the bias $\Delta\rho$ compares the eigenvalue $k_{eff}^{OpenMOC}$ computed with MGXS in OpenMOC to that of the reference eigenvalue k_{eff}^{OpenMC} computed with continuous energy cross sections in OpenMC in units of pcm:

$$\Delta\rho = \left(k_{eff}^{OpenMOC} - k_{eff}^{OpenMC}\right) \times 10^5 \tag{5.1}$$

In each case study, the role of angular discretization in MOC is quantified through convergence studies of the number of azimuthal angles and the track spacing used in the deterministic MOC calculations. The effects of energy discretization are analyzed for MGXS tallied in the CASMO [77] energy group structures ranging from 1 – 70 groups (see App. A). For each of the case studies with heterogeneous geometries, the spatial domain is discretized in OpenMOC's FSR mesh with constant-by-material MGXS to quantify the interaction between the energy and spatial approximations. Spatial discretization studies show the impact of tallying MGXS in each of the FSRs used in the discretized OpenMOC geometry. Finally, MGXS libraries were tallied using OpenMC's iso-in-lab feature (see Sec. 4.2.3) to quantify the impact of the isotropic in lab scattering approximation used in OpenMOC. Inter-pin spatial self-shielding effects are not treated here as they are studied in detail in Chaps. 7 to 10. In each case study, OpenMOC was converged to a criterion of 10^{-7} on the root mean square of the energy-integrated fission

source in each FSR.

5.1.1 Homogeneous Infinite Medium

An initial series of case studies were performed for a homogeneous infinite medium problem. The isotopic composition of the infinite medium was a homogenized mixture derived from the 1.6% enriched UO_2 PWR fuel pin in the BEAVRS PWR model [62] and is described in Tab. 5.1. No approximation is made in the multi-group formulation of the transport equation in the case of homogeneous infinite media. As a result, neutron balance should be exactly preserved within numerical precision in deterministic calculations with MGXS computed in any energy group structure, assuming the MC tallies used to compute MGXS are sufficiently converged.

Table 5.1: Homogeneous infinite medium isotopic composition.

Nuclide	Density [atoms / b-cm]
H-1	4.12377E-2
O-16	2.06218E-2
Zr-90	3.00904E-3
U-235	1.62310E-4
U-238	9.79198E-3

The reference eigenvalues computed with continuous energy cross sections in OpenMC are shown in Tab. 5.2 for both normal anisotropic as well as iso-in-lab scattering. As one would expect for a homogeneous infinite medium, the eigenvalues for anisotropic and iso-in-lab scattering agree to within one standard deviation of the mean. The reference calculations were computed for 100 batches of 10^8 particles per batch. The `openmc.mgxs` module was used to compute 70-group libraries of $\hat{\Sigma}_{t,g}$, $\hat{\Sigma}_{s,k,g'\to g}$, $\nu\hat{\Sigma}_{f,g}$, and $\hat{\chi}_g$ from OpenMC tallies (see Tab. 3.1).

Table 5.2: Reference k_∞^{OpenMC} for a homogeneous infinite medium.

Anisotropic	Isotropic in Lab
1.15908 ± 0.00001	1.15907 ± 0.00001

5.1.1.1 Angular Discretization

The first case study investigated the sensitivity of the OpenMOC eigenvalue to the angular discretization used in the MOC calculation. Tab. 5.3 presents the bias $\Delta\rho$ between OpenMC and OpenMOC for a matrix of azimuthal angles and track spacings. The results for both normal and iso-in-lab scattering indicate consistent agreement of the eigenvalues irregardless of track discretization. This result is expected since the neutron source is isotropic in homogeneous infinite media and does not depend on the angular discretization used to solve the eigenvalue problem.

Table 5.3: Convergence study of the eigenvalue bias $\Delta\rho$ with varying azimuthal angle quadratures and track spacings for a homogeneous infinite medium.

	Track Spacing [cm]					
	0.1	0.01	0.001	0.1	0.01	0.001
# Angles	Anisotropic			Isotropic in Lab		
4	1.3	1.3	1.3	−0.1	−0.1	−0.1
8	1.3	1.3	1.3	−0.1	−0.1	−0.1
16	1.3	1.3	1.3	−0.1	−0.1	−0.1
32	1.3	1.3	1.3	−0.1	−0.1	−0.1
64	1.3	1.3	1.3	−0.1	−0.1	−0.1
128	1.3	1.3	1.3	−0.1	−0.1	−0.1

5.1.1.2 Energy Condensation

A second case study investigated the variation of the OpenMOC eigenvalue with the energy group structure used in the MOC calculation. Tab. 5.4 presents the bias $\Delta\rho$ between OpenMC and OpenMOC for a matrix of energy group structures and FSR spatial discretizations. The OpenMOC calculations each used 128 azimuthal angles and 0.01 cm track spacing. Although the eigenvalues differ by approximately 10 pcm for 1 and 2 groups, the OpenMOC eigenvalues match the eigenvalues computed analytically from the 1- and 2-group MGXS to within 1 pcm. The 10 pcm bias may be due to numerical roundoff error since the MGXS library was tallied in 70 groups in OpenMC and condensed to the coarser group structures with data processing by the `openmc.mgxs` module.

Table 5.4: Convergence study of the eigenvalue bias $\Delta\rho$ with varying energy group structures for a homogeneous infinite medium.

# Groups	Anisotropic	Isotropic in Lab
1	−11.1	−10.5
2	−9.5	−7.1
4	−0.1	−0.5
8	0.3	0.0
16	−0.2	0.5
25	1.8	0.1
40	1.6	0.1
70	1.3	−0.1

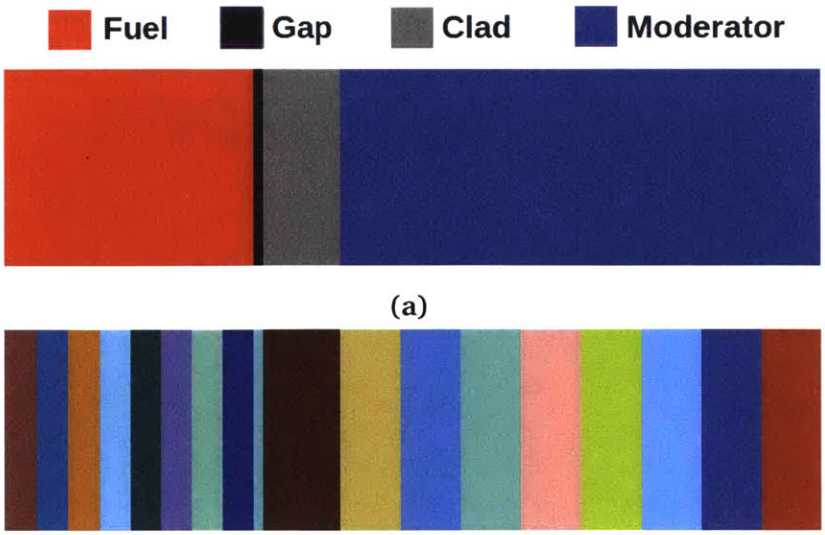
The eigenvalues for a homogeneous infinite medium agree to within nearly 10 pcm for all MOC angular discretizations and energy group structures.

5.1.2 1D Slab

A simple slab model was constructed to quantify approximation errors in a 1D heterogeneous geometry with spatial self-shielding. The slab model was constructed as an "equivalent" 1D model to the 2.4% enriched UO_2 PWR fuel pin in the BEAVRS PWR model [62]. The geometric configuration of UO_2 fuel, helium gap, zirconium clad and water moderator is illustrated in Fig. 5-1 and the dimensions for each material zone are shown in Tab. 5.5. The width of each spatial region was chosen to preserve the volumetric fraction of each material in the slab with those in the corresponding 2D fuel pin[25]. Reflective boundary conditions were applied to all x, y and z boundaries in the geometry. The isotopic composition of each material in the slab is identical to the BEAVRS fuel pin and is itemized in Tab. 5.6.

The reference eigenvalues computed with continuous energy cross sections in OpenMC are shown in Tab. 5.7 for both normal anisotropic as well as iso-in-lab scattering. The reference calculations were computed for 100 batches of 10^7 particles per batch. The reference eigenvalues for the two cases vary by nearly 75 pcm due to anisotropies re-

[25] A truly "equivalent" slab would preserve the mean chord lengths of each material in the 2D fuel pin.



(a)

(b)

Figure 5-1: A 1D slab with fuel, clad and moderator (a). Linearly-spaced tally zones were defined in the fuel and moderator in OpenMC and as FSRs in OpenMOC (b).

Table 5.5: 1D slab dimensions.

Material	Width [cm]
Fuel	0.19177
Gap	0.00777
Clad	0.06109
Water	0.36929

Table 5.6: 1D slab isotopic composition.

Material	Nuclide	Density [atoms / b-cm]
UO_2	O-16	4.58508E-2
	U-235	5.58415E-4
	U-238	2.24186E-2
Helium	He-4	2.40428E-4
Zircaloy	O-16	6.14041E-4
	Fe-56	2.71837E-4
	Zr-90	4.35958E-2
Water	H-1	4.95774E-2
	B-10	8.02369E-6
	B-11	3.22964E-5
	O-16	2.47320E-2

sulting from thermal scattering in the moderator. The `openmc.mgxs` module was used to compute 70-group libraries of $\hat{\Sigma}_{t,k,g}$, $\hat{\hat{\Sigma}}_{t,k,g}$, $\hat{\Sigma}_{s,k,g'\to g}$, $\hat{\hat{\Sigma}}_{s,k,g'\to g}$, $\nu\hat{\Sigma}_{f,g}$, and $\hat{\chi}_g$ from OpenMC tallies (see Tab. 3.1).

Table 5.7: Reference k_{eff}^{OpenMC} for a 1D slab.

Anisotropic	Isotropic in Lab
1.16275 ± 0.00003	1.16202 ± 0.00003

5.1.2.1 Angular Discretization

The first case study investigated the convergence of the OpenMOC eigenvalue with the angular discretization used in the MOC calculation. Tab. 5.8 presents the bias $\Delta\rho$ between OpenMC and OpenMOC for a matrix of azimuthal angles and track spacings. Two different 70-group MGXS libraries were computed from OpenMC tallies with anisotropic and iso-in-lab scattering. No transport correction was applied to the total cross section $\hat{\Sigma}_{t,k,g}$ or the scattering matrix $\hat{\Sigma}_{s,k,g'\to g}$. No spatial discretization was applied to the materials for the FSR mesh in OpenMOC. The results for anisotropic and iso-in-lab scattering exhibit a bias resulting from the multi-group approximation which converges to 170 and 280 pcm, respectively. The magnitude of the bias appears to converge with 128 azimuthal angles and was largely insensitive to the track spacing. The MGXS tallied with iso-in-lab scattering in OpenMC eliminates the isotropic scattering approximation in OpenMOC but increases the overall bias by over 100 pcm with respect to the anisotropic case due to the cancellation of other approximation errors.

Table 5.8: Convergence study of the 70-group eigenvalue bias $\Delta\rho$ with varying azimuthal angle quadratures and track spacings for a 1D slab.

	Track Spacing [cm]					
	0.1	**0.01**	**0.001**	**0.1**	**0.01**	**0.001**
# Angles	**Anisotropic**			**Isotropic in Lab**		
4	731	731	731	840	840	840
8	472	339	323	581	448	432
16	282	166	149	390	275	257
32	154	135	131	262	243	239
64	158	146	156	266	254	265
128	165	164	169	274	272	278
256	161	171	172	270	280	280
512	165	170	172	274	278	281

5.1.2.2 Energy Condensation and FSR Discretization

The second case study investigated the variation of the OpenMOC eigenvalue with the energy group structure used in the MOC calculation. This case study simultaneously varied the FSR discretization used in the OpenMOC simulation. The fuel and moderator were modeled with 1 – 16 equal volume FSRs in each material. The gap and clad were each modeled with a single FSR. Tab. 5.9 presents the bias $\Delta\rho$ between OpenMC and OpenMOC for a matrix of energy group structures and FSR spatial discretizations. In each case, the MGXS used in OpenMOC were tallied by material rather than FSR in OpenMC (*i.e.*, the spatial tally mesh corresponded to Fig. 5-1a). The OpenMOC calculations each used 128 azimuthal angles and 0.01 cm track spacing. The slab was discretized into 1 – 16 equal volume FSRs in the fuel and moderator.

The results illustrate a strong interaction between the energy and spatial meshes used to solve the multi-group transport equation. The eigenvalue bias varies by up to 150 pcm between energy group structures and nearly 400 pcm between FSR discretizations. The application of the transport correction reduces the bias by 55 pcm for the 70-group structure and 16× FSR discretization. The use of the iso-in-lab scattering feature reduces the bias by an additional 50 pcm with respect to anisotropic scattering.

Table 5.9: Convergence study of the eigenvalue bias $\Delta\rho$ with varying energy group structures and FSR spatial discretizations for a 1D slab with *MGXS tallied by material*.

	FSR Discretization				
	1×	2×	4×	8×	16×
# Groups	Anisotropic without Transport Correction				
1	70	71	72	72	71
2	106	−16	−80	−101	−85
4	112	−2	−63	−90	−87
8	158	−13	−109	−149	−160
16	182	−9	−117	−162	−173
25	146	−37	−143	−190	−201
40	155	−40	−155	−205	−217
70	164	−38	−156	−208	−221
	Anisotropic with Transport Correction				
1	87	88	88	88	87
2	285	211	175	167	189
4	188	113	75	51	63
8	211	81	13	−23	−24
16	232	82	1	−39	−40
25	148	−1	−89	−123	−128
40	148	−14	−111	−149	−156
70	153	−16	−118	−158	−166
	Isotropic in Lab				
1	132	134	135	135	133
2	289	167	102	81	97
4	226	112	50	24	26
8	296	126	30	−10	−21
16	321	131	22	−22	−33
25	258	75	−31	−78	−88
40	262	67	−47	−98	−110
70	272	71	−48	−100	−113

5.1.2.3 Spatial Homogenization and FSR Discretization

Finally, a case study was performed to investigate the sensitivity of the OpenMOC eigenvalue to the spatial tally mesh used to compute MGXS. This study was identical to that presented in Sec. 5.1.2.2, but the **MGXS were computed using a tally mesh in OpenMC identical to the FSR mesh used by OpenMOC.** Tab. 5.10 presents the bias $\Delta\rho$ for a matrix of energy group structures and MGXS spatial tally zone meshes. In each case, the MGXS were tallied on the FSR mesh with 1 – 16 equal volume subdivisions in the fuel and moderator, with a single subdivision each in the gap and clad (*i.e.*, the spatial tally mesh corresponded to Fig. 5-1b). The OpenMOC calculations each used 128 azimuthal angles and 0.01 cm track spacing.

The trends analyzed in Tab. 5.9 emerge in a nearly identical manner with spatially-dependent[26] MGXS in Tab. 5.10. In particular, the eigenvalue bias grows in magnitude with more energy groups, and is largely invariant to FSR spatial discretization or the elimination of the isotropic scattering approximation with iso-in-lab scattering. Most importantly, the results in Tab. 5.10 indicate that spatial self-shielding effects (*e.g.*, variations in the flux energy spectrum across the fuel) captured with spatially-dependent scalar flux-weighted MGXS for each FSR do not have a substantial impact on the systematic errors in the eigenvalue for this 1D slab problem.

A systematic bias of -100 to -200 pcm exists between OpenMC and OpenMOC for a 1D slab. The bias varies with the FSR discretization and grows with more energy groups. The bias is partially eliminated with iso-in-lab scattering, and is largely invariant to the spatial mesh used to generate MGXS.

[26] In this context, spatially-dependent refers to MGXS defined on the FSR rather than the material mesh.

Table 5.10: Convergence study of the eigenvalue bias $\Delta\rho$ with varying energy group structures and FSR spatial discretizations for a 1D slab with *MGXS tallied by FSR*.

	FSR Discretization				
	1×	2×	4×	8×	16×
# Groups	Anisotropic without Transport Correction				
1	68	107	111	78	113
2	104	9	−63	−107	−82
4	110	10	−36	−80	−71
8	156	−11	−86	−144	−141
16	180	−10	−97	−158	−153
25	144	−39	−127	−180	−173
40	153	−42	−138	−196	−190
70	162	−39	−142	−202	−198
	Anisotropic without Transport Correction				
1	85	90	103	92	98
2	283	213	189	167	201
4	186	120	96	62	86
8	209	84	26	−19	−11
16	230	84	11	−40	−31
25	146	−3	−82	−125	−118
40	147	−16	−107	−152	−147
70	152	−19	−114	−161	−156
	Isotropic in Lab				
1	139	97	127	162	140
2	296	145	100	90	88
4	233	125	77	61	51
8	304	136	45	7	−9
16	329	139	38	−6	−23
25	266	83	−11	−54	−70
40	270	75	−29	−73	−89
70	280	76	−33	−81	−93

5.1.3 2D Fuel Pin Cell

A PWR fuel pin cell model was constructed to quantify approximation errors in a 2D heterogeneous geometry with spatial self-shielding. The pin cell is identical to the 2.4% enriched UO_2 PWR fuel pin in the BEAVRS PWR model [62]. The geometric configuration of UO_2 fuel, helium gap, zirconium clad and water moderator is illustrated in Fig. 5-2 and the dimensions for each material zone are shown in Tab. 5.11. Reflective boundary conditions were applied to all boundaries in the geometry. The isotopic compositions of each material in the fuel pin were identical to those used in the 1D slab (see Tab. 5.6).

Table 5.11: 2D fuel pin dimensions.

Material	Dimension [cm]
Fuel Outer Radius	0.39218
Gap Outer Radius	0.40005
Clad Outer Radius	0.45720
Pin Pitch	1.25984

(a) (b) (c)

Figure 5-2: A PWR fuel pin cell with fuel, gap, clad and moderator (a). Radial tally zones were defined in each material in OpenMC (b). The tally zones were further subdivided into angular sectors for the FSR mesh in OpenMOC (c).

The reference eigenvalues computed with continuous energy cross sections in OpenMC are shown in Tab. 5.12 for both normal anisotropic and iso-in-lab scattering. The reference calculations were computed for 100 batches of 10^7 particles per batch. The

reference eigenvalues for the two cases vary by 65 pcm due to anisotropic thermal scattering in the moderator, roughly the same as that for the 1D slab geometry. The `openmc.mgxs` module was used to compute 70-group libraries of $\hat{\Sigma}_{t,k,g}$, $\hat{\hat{\Sigma}}_{t,k,g}$, $\hat{\Sigma}_{s,k,g'\to g}$, $\hat{\hat{\Sigma}}_{s,k,g'\to g}$, $\nu\hat{\Sigma}_{f,g}$, and $\hat{\chi}_g$ from OpenMC tallies (see Tab. 3.1).

Table 5.12: Reference k_{eff}^{OpenMC} for a 2D fuel pin.

Anisotropic	Isotropic in Lab
1.17486 ± 0.00003	1.17421 ± 0.00002

5.1.3.1 Angular Discretization

The first case study investigated the convergence of the OpenMOC eigenvalue with the angular discretization used in the MOC calculation. Tab. 5.13 presents the bias $\Delta\rho$ for a matrix of azimuthal angles and track spacings. Two different 70-group MGXS libraries were computed from OpenMC tallies with anisotropic and iso-in-lab scattering. No transport correction was applied to the total cross section $\hat{\Sigma}_{t,k,g}$ or the scattering matrix $\hat{\Sigma}_{s,k,g'\to g}$. A single radial tally zone was applied to each material for the MGXS calculation with OpenMC. The fuel and moderator were each discretized into 5 equal volume radial rings, and each material zone was discretized into 8 angular sectors for the FSR mesh in OpenMOC. The results for both normal anisotropic scattering and iso-in-lab scattering exhibit a few hundred pcm bias which appears to converge with 128 azimuthal angles and is largely invariant with the track spacing. The MGXS tallied with iso-in-lab scattering in OpenMC converge to an eigenvalue that is roughly 95 pcm less than that computed for the anisotropic case, with a bias of approximately -170 pcm. Although the bias is of the same order of magnitude as that observed for the 1D slab in Tab. 5.8, the trend is reversed for anisotropic and isotropic in lab scattering.

Table 5.13: Convergence study of the 70-group eigenvalue bias $\Delta\rho$ with varying azimuthal angle quadratures and track spacings for a 2D fuel pin.

	Track Spacing [cm]					
	0.1	0.01	0.001	0.1	0.01	0.001
# Angles	Anisotropic			Isotropic in Lab		
4	372	425	427	467	519	521
8	−377	−419	−418	−282	−325	−323
16	−436	−408	−412	−341	−314	−318
32	−324	−338	−331	−230	−244	−236
64	−256	−296	−285	−162	−202	−191
128	−285	−276	−267	−190	−182	−173
256	−277	−267	−265	−182	−173	−171
512	−273	−263	−264	−179	−169	−170

5.1.3.2 Energy Condensation and FSR Discretization

The second case study investigated the variation of the OpenMOC eigenvalue with the energy group structure used in the MOC calculation. This case study simultaneously varied the FSR discretization used in the OpenMOC simulation. The FSR mesh in the fuel and moderator consisted of 1 – 16 equal volume radial rings in each material with 8 angular sectors. The gap and clad were modeled with a single radial zone subdivided into 8 angular sectors. Tab. 5.14 presents the bias $\Delta\rho$ between OpenMC and OpenMOC for a matrix of energy group structures and FSR spatial discretizations. In each case, the MGXS used in OpenMOC were tallied by material rather than FSR in OpenMC (*i.e.*, the spatial tally mesh corresponded to Fig. 5-2a). The OpenMOC calculations used 128 azimuthal angles and 0.01 cm track spacing. Each of the materials in the fuel pin was discretized into 8 angular sectors. The fuel and moderator were each discretized into 1 – 16 equal volume radial rings.

As was demonstrated for the 1D slab, the results for the fuel pin indicate a strong interaction between the energy and spatial discretization. The eigenvalue bias exhibits a swing of ∼350 pcm between energy and spatial meshes. The bias exceeds 200 pcm for all scattering approximations. The application of the transport correction only reduces

Table 5.14: Convergence study of the eigenvalue bias $\Delta\rho$ with varying energy group structures and FSR spatial discretizations for a 2D fuel pin with *MGXS tallied by material*.

	FSR Discretization				
	1×	2×	4×	8×	16×
# Groups	**Anisotropic without Transport Correction**				
1	65	66	66	66	66
2	21	−23	−54	−65	−64
4	−60	−100	−129	−143	−151
8	−77	−137	−183	−204	−215
16	−74	−141	−194	−219	−230
25	−130	−194	−245	−272	−281
40	−133	−201	−257	−286	−296
70	−134	−204	−263	−294	−304
	Anisotropic with Transport Correction				
1	51	52	52	52	51
2	35	6	−13	−19	−11
4	−60	−89	−109	−125	−126
8	−76	−123	−158	−181	−184
16	−69	−124	−165	−192	−196
25	−126	−180	−223	−249	−252
40	−131	−190	−239	−267	−271
70	−133	−194	−246	−276	−280
	Isotropic in Lab				
1	79	80	80	80	80
2	140	96	65	53	55
4	26	−14	−43	−57	−65
8	25	−35	−81	−102	−113
16	34	−33	−86	−110	−122
25	−32	−95	−147	−173	−182
40	−39	−107	−163	−192	−202
70	−40	−110	−169	−199	−210

the bias by up to 25 pcm depending on the energy group structure. The use of the iso-in-lab scattering feature reduces the bias by roughly $1/3$ or 100 pcm. As was observed

for the 1D slab, the converged bias is negative for all scattering approximations.

5.1.3.3 Spatial Homogenization and FSR Discretization

A final case study was performed to investigate the sensitivity of the OpenMOC eigenvalue to the spatial tally mesh used to compute MGXS. This case study was identical to that presented in Sec. 5.1.3.2, but in this case the **MGXS were computed using a tally mesh in OpenMC identical to the FSR mesh used by OpenMOC.** Tab. 5.15 presents the bias for a matrix of energy group structures and MGXS spatial tally zone meshes. In each case, the MGXS were tallied on the FSR mesh used in OpenMOC with 1 – 16 equal volume rings in the fuel and moderator, with a single ring each for the gap and clad (*i.e.*, the spatial tally mesh corresponded to Fig. 5-2b). The OpenMOC calculations used 128 azimuthal angles and 0.01 cm track spacing.

The trends analyzed in Tab. 5.14 emerge in a similar manner with spatially-dependent MGXS. In particular, the eigenvalue bias grows in magnitude with more energy groups and FSRs but is largely insensitive to the the elimination of the isotropic scattering approximation. As was observed for the 1D slab, the overall systematic error between OpenMC and OpenMOC is not resolved and in fact increases with greater spatial resolution of the MGXS. The results in Tab. 5.15 indicate that spatial self-shielding effects captured with spatially-varying scalar flux-weighted MGXS for each FSR do not have a substantial impact on the systematic errors in the eigenvalue.

A systematic bias of -200 to -300 pcm exists between OpenMC and OpenMOC for a 2D fuel pin. The bias varies with the FSR discretization and grows with more energy groups. The bias is partially eliminated with iso-in-lab scattering, and is invariant to the spatial mesh used to generate MGXS.

Table 5.15: Convergence study of the eigenvalue bias $\Delta\rho$ with varying energy group structures and FSR spatial discretizations for a 2D fuel pin with *MGXS tallied by FSR*.

	FSR Discretization				
	1×	2×	4×	8×	16×
# Groups	Anisotropic without Transport Correction				
1	67	60	63	98	92
2	22	−27	−56	−55	−51
4	−58	−101	−128	−128	−135
8	−75	−139	−182	−194	−197
16	−73	−142	−190	−209	−207
25	−128	−198	−246	−271	−268
40	−131	−209	−261	−288	−288
70	−132	−214	−267	−296	−297
	Anisotropic with Transport Correction				
1	53	61	75	66	72
2	37	11	1	−10	4
4	−58	−83	−92	−114	−109
8	−74	−117	−145	−175	−170
16	−67	−118	−154	−186	−183
25	−124	−181	−221	−253	−245
40	−130	−191	−238	−272	−265
70	−131	−196	−245	−281	−274
	Isotropic in Lab				
1	80	92	55	83	66
2	141	87	29	50	34
4	27	−15	−43	−45	−57
8	26	−34	−85	−90	−102
16	35	−35	−91	−101	−111
25	−31	−105	−158	−170	−182
40	−38	−114	−174	−189	−202
70	−39	−117	−182	−196	−211

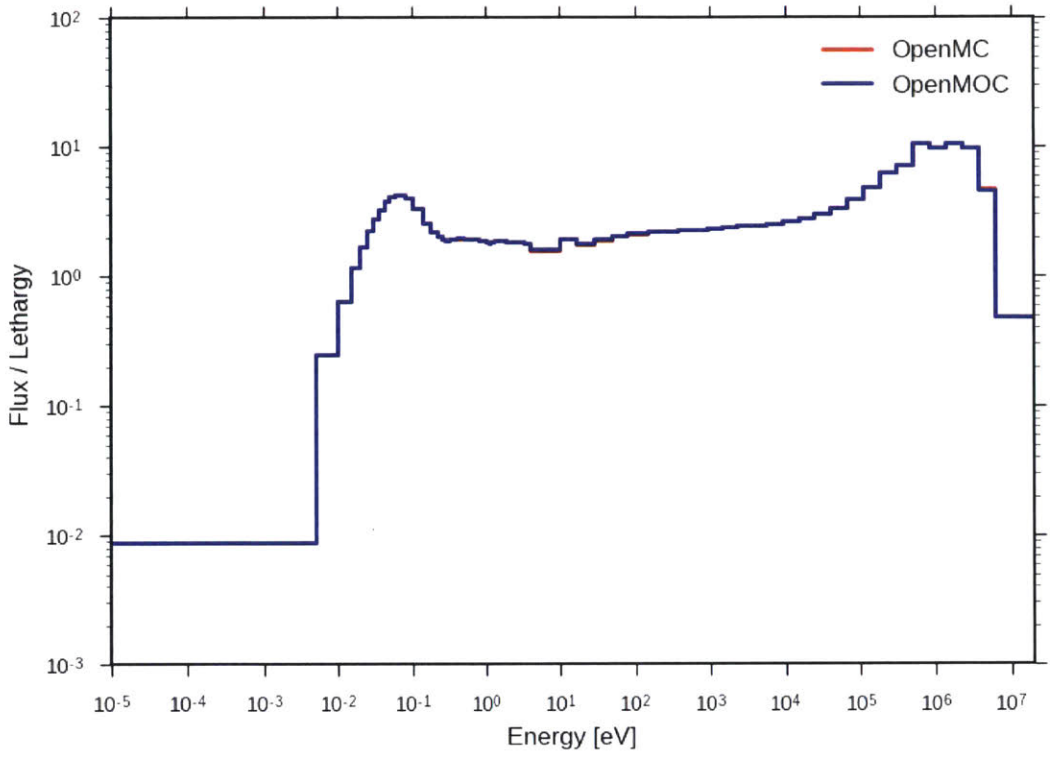
5.2 Diagnosing the Error

The emergence of a negative systematic bias in the eigenvalue with fine energy and spatial discretization for the heterogeneous 1D slab and 2D fuel pin models led to an analysis of the flux spectra computed by OpenMC and OpenMOC. The 70-group volume-averaged energy-dependent flux in the fuel for both benchmarks is illustrated in Fig. 5-3. All of the characteristic trends that one would expect to see in an LWR spectra are easily identifiable. In particular, the fission peak at fast energies, the $1/E$ slowing down flux for epithermal energies, and the Maxwellian peak at thermal energies are visible for the slab and fuel pin. The OpenMC flux is barely visible since there is little difference between the two flux spectra when jointly plotted for all groups. The two spectra do appear to differ slightly in the epithermal regime where there is a noticeable degradation in the flux due to resonance capture and scattering. The following sections investigate the deviations in the flux in resonance groups and estimate how they impact the bias in the eigenvalues predicted by OpenMC and OpenMOC.

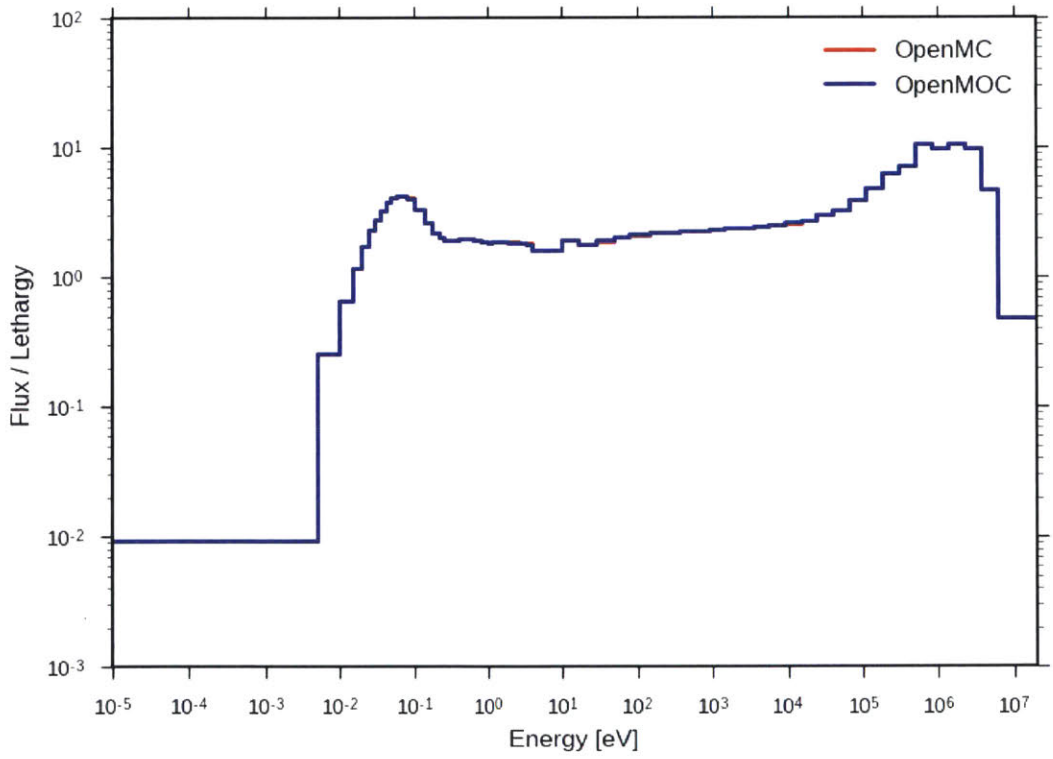
5.2.1 Energy-Dependent Flux Error

Upon further inspection, it was noted that OpenMOC's flux exhibited large errors with respect to the reference OpenMC flux in those energy groups which isolate large U-238 capture resonances. The error in the 70-group flux is illustrated in Fig. 5-4 for the FSRs nearest and furthest from the moderator, along with the average error across all FSRs in the fuel. These plots were generated for the benchmark models with a 16× FSR discretization with MGXS tallied on the FSR mesh. The plots correspond to the case studies in Tables 5.10 and 5.15 with iso-in-lab scattering.

As illustrated in the figures, there is a striking error of up to 1.5% for the fluxes in the slab and up to 2.5% for the fuel pin in the innermost FSR in groups 24, 25 and 27. These energy groups contain the three largest U-238 capture resonances between 4 and 48.052 eV. In addition, the flux errors appear to "build up" as the energy decreases through the resonance region, and the magnitude of the capture resonances increase. The one notable exception to this is group 26 (4 – 9.877 eV) which does not include

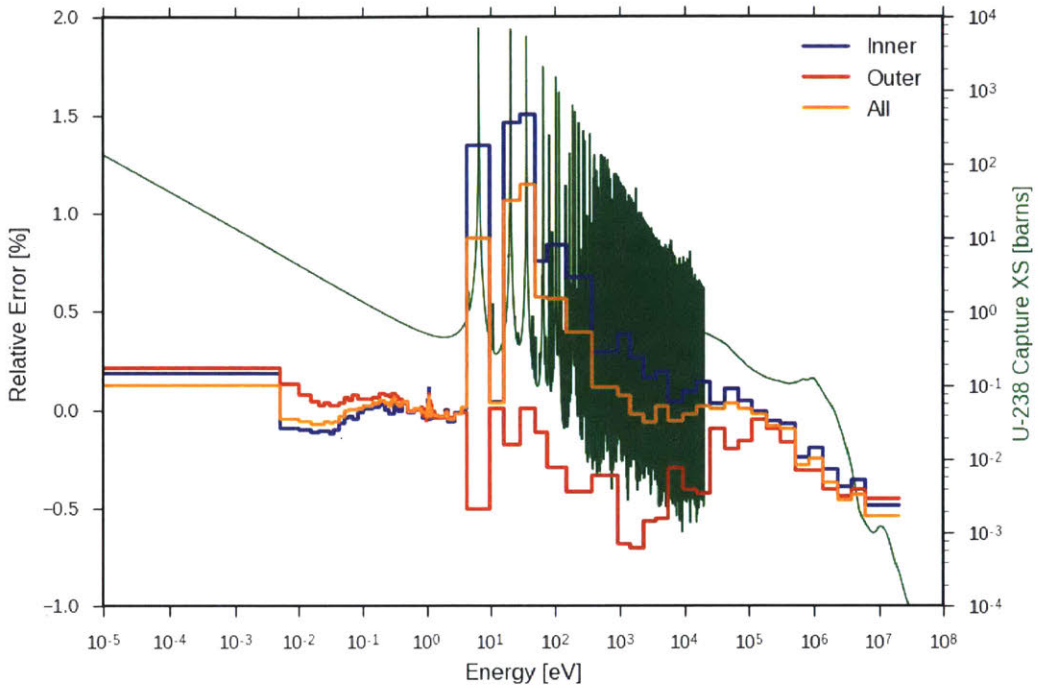


(a)

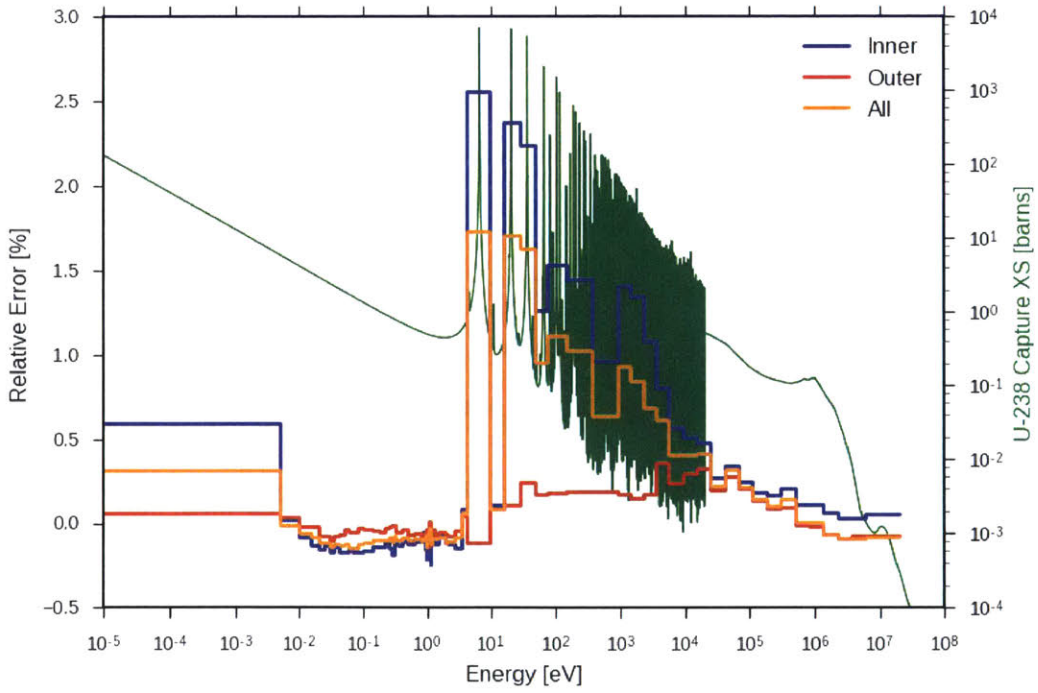


(b)

Figure 5-3: The flux spectrum in energy in a 1D slab (a) and 2D fuel pin (b).



(a)



(b)

Figure 5-4: The energy-dependent relative error of the OpenMOC scalar flux with respect to the reference OpenMC flux in a 1D slab (a) and 2D fuel pin (b) for the innermost, outermost and all FSRs. The results correspond to the case studies presented in Tables 5.10 and 5.15.

a U-238 capture resonance. These observations stand in contrast to the flux errors in the outermost FSR for which no remarkable trend can be discerned. The average error across all FSRs is roughly 2/3 that of the innermost FSR and exhibits the same features in the resonance region. The positive error in the flux in groups with large capture resonances indicates that capture reaction rates are over-predicted in those groups by OpenMOC, contributing to the negative bias in the eigenvalue. Furthermore, these results indicate a strong relationship between the energy and spatial distribution of the flux errors, as is explored in the following section.

5.2.2 Spatially-Dependent Flux Error

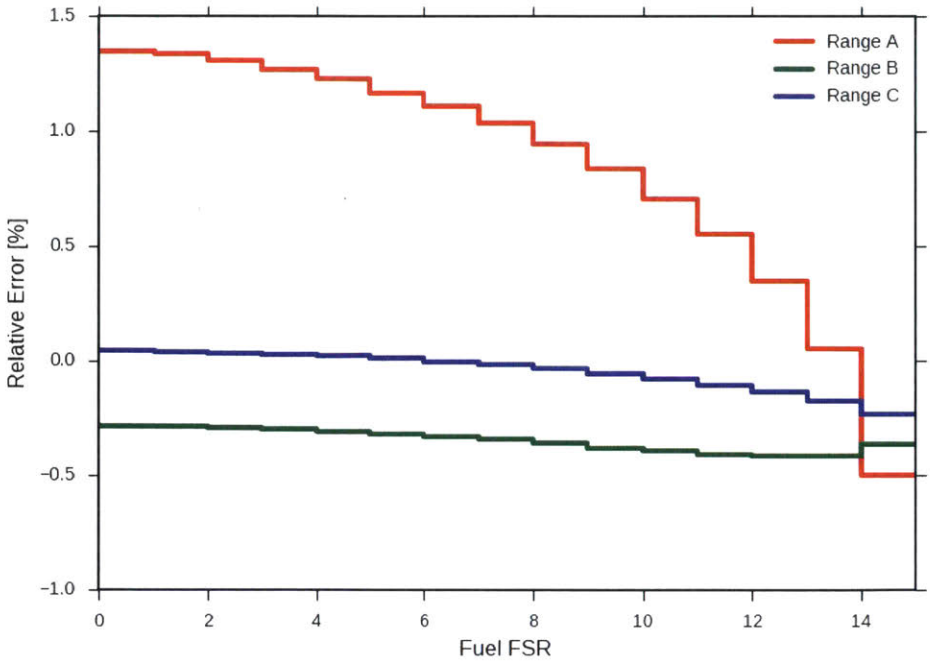
The results presented in Fig. 5-4 indicate significant errors in resonance groups, and in particular, group 27 in the 70-group calculation. Furthermore, the results indicated a large difference in the error profile for those FSRs nearest and furthest from the moderator. These trends were studied further to better understand the spatial variation of the flux error across the 16 FSRs in the fuel for the slab and pin geometries. In this analysis, the error of the flux was considered in the three different *ranges* of energy group structures itemized below:

- **Range A** – group 27 encompassing the U-238 capture resonance at 6.67 eV
- **Range B** – groups 11 – 27 spanning the resonance region from 4 eV – 408.5 keV
- **Range C** – groups 1 – 70 spanning the entire energy regime from 0 – 20 MeV

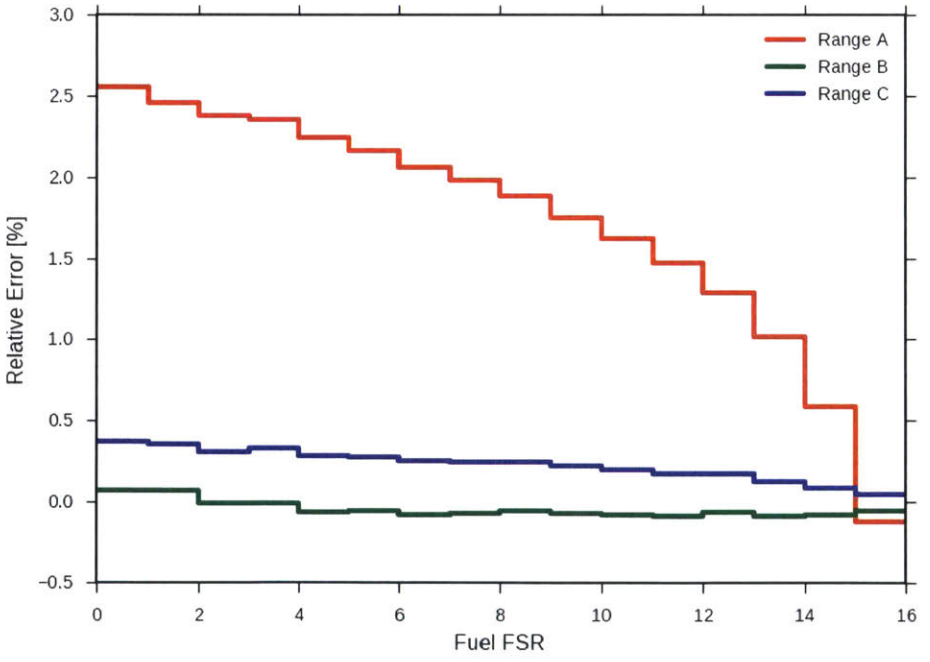
Fig. 5-5 highlights the spatial dependence of the error across the fuel for each energy range. These plots were generated for the benchmark models with a 16× FSR discretization with MGXS tallied on the FSR mesh. The plots correspond to the case studies in Tables 5.10 and 5.15 with iso-in-lab scattering.

The range A error in the slab monotonically decreases from a maximum of nearly 1.5% to a minimum of ∼-0.5% in those FSRs furthest and nearest the moderator, with a similar trend observed for the fuel pin. Furthermore, the trend accelerates in outermost 3 – 4 FSRs nearest the moderator, where the error drops by nearly half of its value at the center of the slab and pin. The error profiles for energy ranges B and C exhibit the

same decreasing trend from the inside to the outside of the slab and pin, but the error magnitude never exceeds 0.5% in magnitude. The systematic error trends in energy and space imply that the negative eigenvalue bias is driven by a poor prediction of the reaction rates in resonance groups, as investigated in the following section.



(a)



(b)

Figure 5-5: The spatially-varying relative error of the OpenMOC scalar flux with respect to the reference OpenMC flux for a 1D slab (a) and 2D fuel pin (b) in energy Ranges A, B, and C. The results correspond to Tables 5.10 and 5.15.

Fig. 5-7 illustrates the spatial dependence of the normalized U-238 capture reaction rates in Range A across the fuel for the slab and pin cell. The "rim effect" of U-238 capture

(and subsequent production of Pu-239) in the outermost ring nearest the moderator is easily seen for both geometries. The capture rates are 5× greater in the outermost slab/ring than in the innermost slabs/rings. The interior zones experience a highly self-shielded flux since neutrons at energies coinciding with the U-238 capture resonance at 6.67 eV are absorbed in the outermost ring before they can further penetrate the fuel. Although the Range A capture rates in the interior regions are relatively small, the largest errors appear in those zones as shown in Fig. 5-5. Taken together, the reaction rates and flux errors convolve to produce a non-negligible error in the volume-integrated U-238 capture rate across the slab/pin as shown in next section.

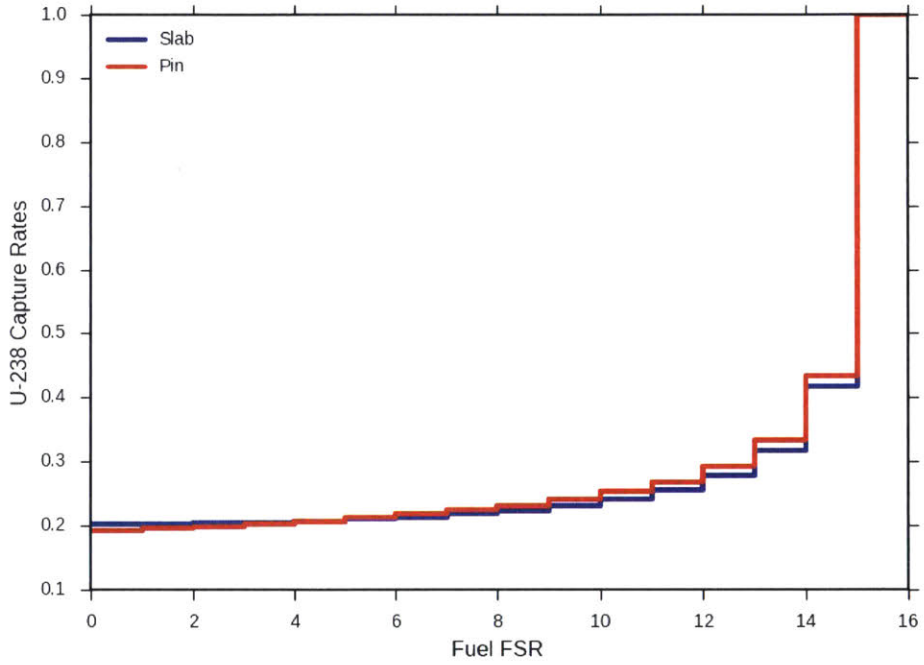


Figure 5-6

Figure 5-7: The normalized spatially-varying U-238 capture rates tallied by OpenMC for a 1D slab 2D fuel pin in Range A. The results correspond to Tables 5.10 and 5.15.

5.2.3 Reaction Rate Errors in the Resonance Region

The errors in the multi-group flux directly correspond to equal errors in the reaction rates since the MGXS are computed from the reference reaction rate and flux tallies. The results presented in Figs. 5-4 and 5-5 revealed significant errors in resonance groups, and in particular, group 27 in the 70-group calculation. In this section, the capture and

absorption rate errors are evaluated to quantify how much of the negative eigenvalue bias in the two geometries can be attributed to the mis-predicted reaction rates in the resonance groups. This analysis was performed for different energy group structures to determine if errors in the reaction rates emerge with more energy groups, similar to the eigenvalue bias observed in the case studies in Sec. 5.1.

The relative error for both U-238 capture and total absorption for all nuclides is quantified in Tab. 5.16 for energy ranges A, B and C (see Sec. 5.2.2). As in Secs. 5.2.1 and 5.2.2, the analysis considered the slab and fuel pin benchmark models with a 16× FSR discretization with MGXS tallied on the FSR mesh. The OpenMC iso-in-lab scattering feature was used to generate MGXS for OpenMOC.

As illustrated in the table, the U-238 capture and total absorption rate errors each grow with the number of energy groups. This trend is particularly pronounced for U-238 capture in the 6.67 eV resonance group which approaches 0.6% and 1.35% for the slab and pin, respectively, when 25 or more groups are used in the multi-group calculation. The 0.08% and 0.17% error in the total absorption in range C directly corresponds to an under-prediction in the eigenvalue of 115 and 197 pcm for the slab and fuel pin, respectively, which closely matches the observed eigenvalue bias for each model[27]. Approximately 5 – 6% and 16 – 18% of the total absorption occurs in energy ranges A and B, with U-238 capture accounting for 80% and 70% of the total absorption in each energy range, respectively. Hence, the error in U-238 resonance capture alone under-predicts the eigenvalue by 105 and 145 pcm for the slab and pin, with approximately half of the error deriving from group 27 of 70 due to the resonance at 6.67 eV.

[27]This analysis neglects the contribution of scattering multiplicity (*e.g.*, (n,xn)) to the eigenvalue.

Table 5.16: The volume-integrated U-238 capture and total absorption rate percent relative errors. The errors correspond to the results in Tables 5.10 and 5.15.

	U-238 Capture			Total Absorption		
	Range A	Range B	Range C	Range A	Range B	Range C
# Groups	1D Slab					
1	−0.02	−0.02	−0.02	−0.11	−0.11	−0.11
2	0.05	0.05	0.03	0.05	0.05	−0.07
4	0.11	−0.08	0.06	0.12	−0.08	−0.05
8	0.19	−0.03	0.11	0.22	−0.03	0.00
16	0.21	−0.02	0.12	0.23	−0.02	0.02
25	0.58	0.36	0.26	0.61	0.37	0.06
40	0.58	0.43	0.30	0.61	0.44	0.07
70	0.59	0.46	0.30	0.61	0.47	0.08
	2D Fuel Pin					
1	−0.02	−0.02	−0.02	−0.07	−0.07	−0.07
2	0.11	0.11	0.07	0.11	0.11	−0.03
4	0.55	0.07	0.32	0.54	0.07	0.04
8	0.71	0.11	0.40	0.72	0.11	0.08
16	0.72	0.12	0.41	0.73	0.12	0.09
25	1.32	0.85	0.61	1.34	0.85	0.15
40	1.33	0.93	0.64	1.34	0.92	0.16
70	1.33	0.99	0.65	1.35	0.97	0.17

These results indicate that spatial self-shielding effects in resonance groups is not adequately captured by the MGXS and/or the multi-group calculation. Furthermore, this analysis illustrates the counter-intuitive result that the bias between continuous energy Monte Carlo and multi-group deterministic transport may in fact increase in magnitude with more energy groups. Although it is challenging to isolate the factors which convolve to bias the eigenvalue, the data presented here indicates that an over-prediction of U-238 capture in the resonance groups largely drives the error. These results will be discussed in greater depth in the context of angular-dependent MGXS in Chapter 6.

The negative eigenvalue bias is caused by an over-prediction of absorption, dominated by U-238 capture in the 6.67 eV resonance. The reaction rate error increases with more energy groups as observed for the eigenvalue bias.

Highlights

- OpenMOC calculations were performed using MGXS generated by OpenMC to compare eigenvalues for simple benchmarks.
- The eigenvalues closely agreed for homogeneous infinite media. The eigenvalues exhibited a bias of -200 to -300 pcm for heterogeneous geometries, including a 1D slab and 2D fuel pin.
- A series of case studies demonstrated the dependence of the bias with:
  - *Energy condensation* – The bias increased with more energy groups due to systematic errors in groups with large U-238 capture resonances.
  - *Spatial homogenization* – The spatial tally mesh used to generate MGXS in the fuel and moderator had no effect on the bias.
  - *Angular treatment* – MGXS generated with iso-in-lab scattering in OpenMC reduced the bias by <100 pcm or $\sim \frac{1}{3}$.
- The eigenvalue bias is largely attributable to an over-prediction of U-238 capture rates in resonance groups.
- The flux errors indicate that spatial self-shielding is not adequately modeled in spatially heterogeneous multi-group calculations even when the "true" scalar flux is used to compute MGXS.

Chapter 6

SuPerHomogénéisation Factors

The results in Chapter 5 demonstrated that using the "true" flux spectrum from Monte Carlo to perform energy condensation and spatial homogenization will not necessarily result in accurate deterministic multi-group calculations. Large systematic biases in the eigenvalue were observed for simple heterogeneous benchmark models. In Section 5.2 it was shown that the bias largely derives from errors in the multi-group reaction rates in the large thermal U-238 capture resonances. These results indicate that one or more of the approximations made in multi-group transport theory are invalidated in heterogeneous geometries and prevent an appropriate treatment of spatial self-shielding at the fuel/moderator interface in PWR geometries.

Chap. 2 discussed approximations made in multi-group theory to simplify the neutron transport equation in angle, energy and space. Many of these approximations were quantitatively studied in Chap. 5 – including energy group structure, spatial discretization mesh, and isotropic scattering – yet it was demonstrated that none of these led to the eigenvalue bias. This chapter investigates the flux separability approximation as the dominant factor contributing to the eigenvalue bias.

Sec 6.1 reviews some recent work by Gibson [78] to quantify the approximation error resolved with angular-dependent MGXS, which closely mirrors the trends observed in Chap. 5. The historical SPH factor concept is introduced in the context of angular-dependent MGXS in Sec. 6.2, and SPH factors are applied to simple heterogeneous benchmarks and the results analyzed in Sec. 6.3. This chapter concludes with a summary

of the shortcomings of the SPH approach in Sec. 6.4 and the need for new methods to account for the angular dependence in MGXS in Sec. 6.5.

6.1 Angular-Dependent MGXS

The flux separability approximation introduced in Sec. 2.3.2 led to the use of the scalar rather than the angular flux to condense the total cross section in energy. The mathematically proper treatment would instead use the angular flux to condense the total MGXS in angle, energy and space, resulting in **angular-dependent total MGXS**. Flux separability is commonly used since conventional MGXS generation schemes are generally incapable of approximating the angular dependence of the flux in the arbitrary geometries and spatial discretizations modeled by multi-group transport codes. As a result, multi-group codes are unable to reproduce the correct angular dependence of the neutron flux with scalar flux-weighted MGXS. Furthermore, flux separability is an approximation which may lead to non-trivial errors in downstream multi-group calculations, such as the eigenvalue bias observed in Chap. 5.

The flux separability approximation will necessarily hold in infinite homogeneous media – such as that considered in Sec. 5.1.1 – since the flux does not vary in angle or space. However, the flux may vary greatly by angle in a heterogeneous geometry with significant spatial self-shielding. For example, consider the flux from two different directions impinged upon the FSR for one radial ring and angular sector in a fuel pin in Fig. 6-1. The epithermal flux entering from the moderator will be unshielded and will likely be quite similar to the $1/E$ asymptotic spectrum. In contrast, the flux which traverses the fuel pin will be significantly depressed in the resonant groups. As a result, the reaction rates for the incoming flux will be greater than those for the outgoing flux in resonant groups, which would be reflected in angular-dependent total MGXS.

A recent PhD thesis by Gibson [78] studied the impact of using angular-dependent total MGXS on multi-group calculations to avoid the flux separability approximation. Gibson was motivated by his observation of reaction rate errors similar to those discovered in Sec. 5.2. In particular, he solved for the reference ultra-fine flux for a 2D PWR fuel

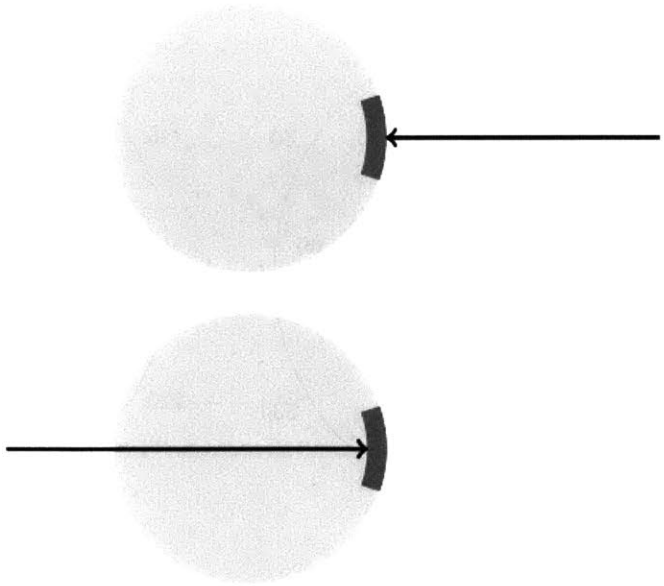
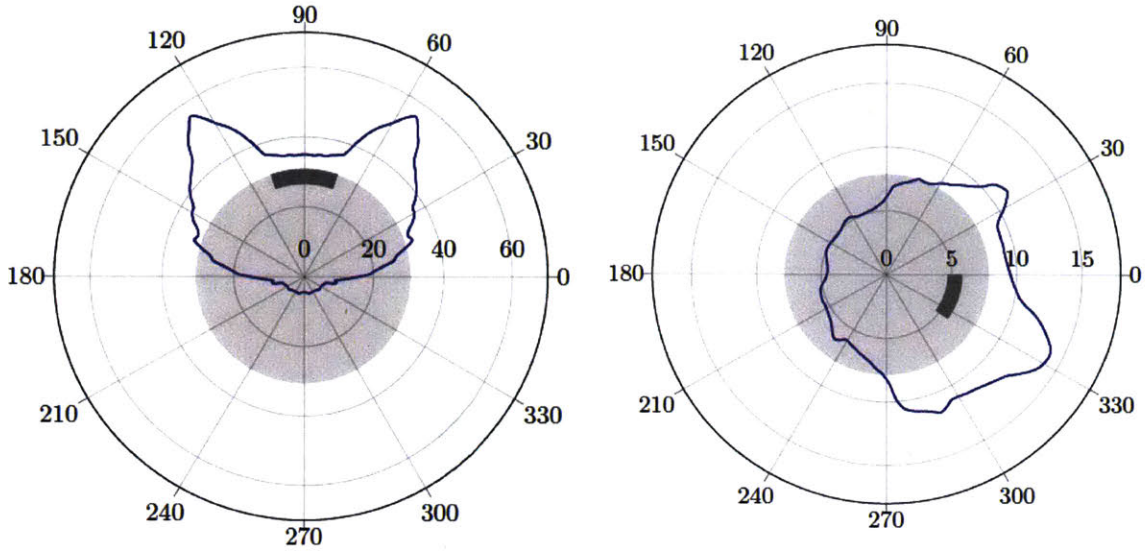


Figure 6-1: Angular flux impinged on an FSR from the moderator (top) and after traversing the fuel (bottom). *Image courtesy of N. Gibson [78].*

pin given a fixed source. The reference scalar flux was used to condense the continuous energy total cross sections to 69-group MGXS, which were employed in a fixed source transport calculation. The reaction rates computed from the ultra-fine and 69-group scalar fluxes differed by up to 1% for low-lying energy groups with large U-238 capture resonances, with an error profile in energy similar to that shown in Fig. 5-4. Gibson's results indicated that an improper treatment of self-shielding effects in heterogeneous geometries may lead to errors in resonant groups even when the exact scalar flux is used to collapse cross sections in energy and space.

Gibson further investigated this issue by using the reference ultra-fine angular flux to compute angular-dependent MGXS. Some examples of the angular-dependent capture MGXS generated for two different FSRs shaded in dark gray are shown in Fig. 6-2. The MGXS in the FSR at the fuel/moderator interface in Fig. 6-2a ranges from less than 5 to more than 50 barns for angles entering and leaving the fuel pin. The peaks near 60° and 120° are due to extra moderation experienced by neutrons streaming through the infinite rectilinear fuel pin lattice at those angles. The MGXS in FSRs in the interior of the fuel pin, such as the one shown in Fig. 6-2b, exhibit similar but less prominent properties since the flux is shielded in all directions.



(a) (b)

Figure 6-2: Angular-dependent capture MGXS for the 6.67 eV resonance group as a function of azimuthal angle for two different FSRs. The radial axis is given in units of barns and the azimuthal axis in units of degrees. *Image courtesy of N. Gibson [78].*

Gibson proceeded to show that the angular-dependent total MGXS eliminated the reaction rate errors observed with scalar flux-weighted MGXS. His analysis highlighted the tight coupling between angular dependence and spatial discretization. Notably, the reaction rate errors were reduced by an order of magnitude only when angular-dependent MGXS were paired with a fine FSR spatial discretization. This can be explained by the fact that the angular variation of the volume-integrated flux is diminished as the FSR mesh is coarsened. Thus, the spatial self-shielding effects at the fuel/moderator interface cannot be captured by angular-dependent total MGXS if the FSR discretization is unable to distinguish between neutrons entering and leaving the fuel pin.

Although angular-dependent total MGXS the most direct solution to eliminate the flux separability approximation, it is not a desirable approach for a number of reasons. Angular-dependent MGXS would significantly increase the memory footprint for MGXS libraries, and be complicated to accommodate in multi-group methods. Furthermore, angular-dependent MGXS is not attractive for MC-based MGXS generation since many more particle histories would be required to converge the MGXS in each discrete angular tally bin. The Consistent-P approximation [7] is an alternative method that embeds the

angular dependence of the total MGXS within the angular expansion of the scattering kernel (Eqn. 2.13) while retaining an angular independent total MGXS. Although the Consistent-P approximation is one viable approach used in many transport codes, it was not evaluated here since anisotropic scattering was not implemented in OpenMOC at the time of this writing. A third method known as SuPerHomogénéisation factors was discussed in this context by Gibson, who employed them to reduce heterogeneous resonant reaction rate errors (albeit to a lesser extent than he achieved with angular-dependent total MGXS). SPH is a relatively simple-to-implement method which does not require an anisotropic scattering kernel implementation. For this reason, the SPH factor scheme was evaluated in OpenMOC as discussed in the following sections.

Multi-group reaction rates are not preserved in heterogeneous geometries due to the flux separability approximation. Angular-dependent total MGXS or an equivalence scheme such as SPH factors are needed to resolve the bias between OpenMC and OpenMOC.

6.2 SuPerHomogénéisation Factors

As discussed in Chap. 2, multi-group theory is valid and consistent if and only if MGXS are defined to preserve reaction rates. In Chap. 5 it was determined that reaction rate preservation is not possible in heterogeneous geometries, even if the exact flux from MC is used to collapse the cross sections in energy and space. SPH factors were first proposed by Hébert [79] to preserve reaction rates during energy condensation and spatial homogenization. The SPH factor algorithm requires knowledge of a reference source that is used in a multi-group fixed source solver to derive multiplicative factors that adjust the total MGXS to force neutron balance.

The literature provides little explanation of the type of approximation errors that SPH factors are designed to mitigate. In theory, the SPH scheme adjusts MGXS to preserve reaction rates irregardless of the source of approximation error – including errors which derive from a poor treatment of energy and spatial self-shielding, a coarse

spatial and angular discretization of the multi-group calculation method, and/or a truncated approximation to the anisotropic scattering kernel. In this section, SPH factors are investigated as one approach to resolve the reaction rate errors observed in Chap. 5. The mathematical formulation behind SPH factors is described in Sec. 6.2.1, the iterative algorithm used to compute the factors is summarized in Sec. 6.2.2, and the implementation of SPH factors in OpenMOC is highlighted in Sec. 6.2.3.

6.2.1 Overview

The SPH algorithm enforces reaction rate preservation between a reference fine-mesh transport problem and a corresponding coarse mesh transport or diffusion problem in energy and space. SPH factors have traditionally been applied to spatially-homogenized few-group MGXS for coarse mesh diffusion applications. However, this section will introduce SPH factors to enforce equivalence between continuous energy Monte Carlo and deterministic multi-group transport methods.

The SPH scheme postulates the existence of a set of factors $\mu_{k,g}$ for each spatial zone k and energy group g which force the streaming and collision terms in the transport equation to balance with a fixed source $Q_{k,g}$:

$$\mathbf{\Omega} \cdot \nabla \psi_g(\mathbf{r}, \mathbf{\Omega}) + \mu_{k,g} \Sigma_{t,k,g} \psi_g(\mathbf{r}, \mathbf{\Omega}) = Q_{k,g}(\mathbf{\Omega}) \tag{6.1}$$

In this equation, the SPH factors are applied to correct the total MGXS in each region and group. The fixed source $Q_{k,g}$ is computed from the reference fine-mesh solution. In this case, the fixed source is treated as the sum of scattering and fission production sources in each energy group and spatial zone. For example, continuous energy Monte Carlo can be used to compute reference multi-group fluxes and MGXS, which are then combined to compute an isotropic source as follows:

$$Q_{k,g}(\mathbf{\Omega}) = \frac{1}{4\pi} \sum_{g'=1}^{G} \Sigma_{s,k,g' \to g} \phi_{k,g'} + \frac{\chi_{k,g}}{4\pi k_{eff}} \sum_{g'=1}^{G} \nu \Sigma_{f,k,g'} \phi_{k,g'} \tag{6.2}$$

Given the fixed source and total MGXS from MC, Eqn. 6.1 may be solved using any multi-group transport method, such as MOC. The challenge is to devise estimates to the true SPH factors $\mu_{k,g}$ which adequately preserve reaction rates. The following section describes the iterative scheme used to estimate SPH factors.

6.2.2 Algorithm

An iterative algorithm is used to estimate SPH factors from a series of multi-group fixed source calculations. First, the estimates $\mu_{k,g}^{(n)}$ at iteration n to the true SPH factors $\mu_{k,g}$ are introduced as a correction factor for the total cross section in Eqn. 6.1:

$$\mathbf{\Omega} \cdot \nabla \psi_g^{(n)}(\mathbf{r}, \mathbf{\Omega}) + \mu_{k,g}^{(n-1)} \Sigma_{t,k,g} \psi_g^{(n)}(\mathbf{r}, \mathbf{\Omega}) = Q_{k,g}(\mathbf{\Omega}) \tag{6.3}$$

A multi-group transport code (such as OpenMOC) may be used to solve Eqn. 6.3 with angular and volume integration to compute the scalar flux distribution. The SPH factor estimates $\mu_{k,g}^{(n)}$ are found from the ratio of the reference Monte Carlo scalar flux $\phi_{k,g}^{MC}$ to the flux $\phi_{k,g}^{(n)}$ computed from the fixed source calculation at iteration n,

$$\mu_{k,g}^{(n)} = \frac{\phi_{k,g}^{MC}}{\phi_{k,g}^{(n)}} \tag{6.4}$$

where the factors are initialized to unity on the first iteration:

$$\mu_{k,g}^{(0)} = 1 \tag{6.5}$$

The SPH factors are used to find a total MGXS which forces neutron balance in Eqn. 6.3. The initial total MGXS $\Sigma_{t,k,g}^{(0)}$ is computed from the reference MC flux and total reaction rate tallies. The SPH factors are then used to obtain a corrected total MGXS $\Sigma_{t,k,g}^{(n)}$ on each iteration:

$$\Sigma_{t,k,g}^{(n)} = \mu_{k,g}^{(n-1)} \Sigma_{t,k,g}^{(0)} \tag{6.6}$$

The series of fixed source problems defined by Eqn. 6.3 are solved until the SPH factors converge. A common convergence criterion is the maximum relative absolute deviation across energy groups and spatial zones:

$$res = \max_{k,g} \left| \frac{\mu_{k,g}^{(n)} - \mu_{k,g}^{(n-1)}}{\mu_{k,g}^{(n-1)}} \right| \tag{6.7}$$

A residual of 10^{-7} can typically be achieved with twenty or fewer iterations.

The scattering matrix $\Sigma_{s,k,g' \to g}$ and fission production cross section $\nu\Sigma_{f,k,g}$ are used to compute the reference fixed source in Eqn. 6.2, but are not needed in the iterative

scheme defined in Eqn. 6.3. However, in the context of this thesis, SPH factors are computed to preserve reaction rates in subsequent eigenvalue calculations. Therefore, the SPH factors must be applied to the scattering matrix and fission production cross sections to produce a fully-corrected MGXS library:

$$\Sigma^{(n)}_{s,k,g' \rightarrow g} = \mu^{(n-1)}_{k,g} \Sigma^{(0)}_{s,k,g' \rightarrow g} \tag{6.8}$$

$$\nu\Sigma^{(n)}_{f,k,g} = \mu^{(n-1)}_{k,g} \nu\Sigma^{(0)}_{f,k,g} \tag{6.9}$$

It should be noted that although the SPH-corrected MGXS are defined to preserve reaction rates, they will not preserve the group-wise scalar flux. However, the angular or scalar flux may be easily recovered from the fluxes $\tilde{\psi}_{k,g}$ and $\tilde{\phi}_{k,g}$ computed with the SPH-corrected MGXS in an eigenvalue or fixed source calculation:

$$\psi_{k,g} = \mu_{k,g} \tilde{\psi}_{k,g} \tag{6.10}$$

$$\phi^{(n)}_{k,g} = \mu_{k,g} \tilde{\phi}_{k,g} \tag{6.11}$$

The SPH iteration algorithm described here is summarized in Alg. 6-1. It should be noted that as presently posed, there is no unique solution to the set of SPH factors which preserve reaction rates. A unique solution may be found by forcing the factors to be unity in non-fissile zones (*e.g.*, moderator, clad and gap)[28]. This approach is motivated by the fact that resonances which lead to self-shielding errors – such as the U-238 capture resonances studied in Sec. 2.3.2 – are generally from isotopes in the fuel. However, the reaction rates in non-fissile zones will not be preserved since the MGXS in these zones remain uncorrected, but these errors are likely dominated by those in the fuel as was shown for the PWR benchmarks in Sec. 5.2.3.

[28]A formulation of the SPH algorithm which corrected MGXS in both fissile and non-fissile zones was also implemented. This formulation introduced an outer loop over each spatial zone in Alg. 6-1. The scheme is not presented here since it was unstable and the SPH factors diverged after a few outer iterations. Future work may develop a more rigorous approach to preserve reaction rates in each spatial zone.

[29]A series of G independent fixed source problems may be solved for each of the G groups. Alternatively, a single fixed source problem may simultaneously solve for all G groups, as is done in OpenMOC.

Algorithm 6-1: SPH Factor Algorithm

1: Initialize $\Sigma_{t,k,g}^{(0)}$, $\Sigma_{s,k,g'\rightarrow g}^{(0)}$, $\nu\Sigma_{f,k,g}^{(0)}$, and $\chi_{k,g}^{(0)}$ from MC tallies ▷ Tab. 3.1
2: Compute $Q_{k,g}$ from MC flux and MGXS ▷ Eqn. 6.2
3: Initialize $\mu_{k,g}^{(0)}$ to unity
4: **while** SPH factor residuals are not converged **do**
5: Update $\Sigma_{t,k,g}^{(n)}$ with SPH factors ▷ Eqn. 6.6
6: Solve fixed source transport problem[29] ▷ Eqn. 6.3
7: Compute new SPH factors $\mu_{k,g}^{(n)}$ ▷ Eqn. 6.4
8: Compute SPH factor residuals ▷ Eqn. 6.7
9: **end while**
10: Compute final MGXS with SPH factors ▷ Eqns. 6.6, 6.8, 6.9

6.2.3 Implementation in OpenMOC

The SPH scheme was implemented in the `openmoc.materialize` Python module of the OpenMOC code (see Sec. 4.3.3). The SPH algorithm was specifically implemented to work with the `Library` class included in the `openmc.mgxs` Python module for MGXS generation with OpenMC (see Sec. 4.2.4). In particular, the fixed source in Eqn. 6.2 is computed from the MC tallies in the `Library` object and used to construct a fixed source simulation in OpenMOC. The MGXS tabulated in the `Library` are loaded into OpenMOC, and the series of fixed source calculations in Alg. 6-1 is managed from Python. The MGXS data in the `Library` is updated with the SPH factors computed at each iteration. The final corrected MGXS `Library`, along with the SPH factors, are returned to the user for use in subsequent OpenMOC eigenvalue calculations.

Although the SPH algorithm is relatively simple, a few subtle issues made the implementation in OpenMOC less than straightforward. First, it should be noted that OpenMC tallies are volume-integrated, while reaction rates and fluxes in OpenMOC are volume-averaged. As a result, the MC tallies must be normalized to the volume used by OpenMOC for each FSR as opposed to the true volumes[30] to calculate the volume-averaged fixed source in each FSR. In addition, it was relatively complicated to map the MGXS data and SPH factors between the combinatorial geometry meshes used by OpenMC and OpenMOC. These challenges notwithstanding, SPH factors were implemented in OpenMOC v0.2.1, and used to reduce reaction rate errors as discussed

[30] OpenMOC estimates spatial volumes by integrating the characteristic track lengths across each FSR.

in the following section.

The SPH factor approach uses a reference fixed source to correct the total MGXS to preserve reaction rates between fine and coarse mesh methods.

6.3 Case Studies

A number of case studies were performed to evaluate the effectiveness of SPH factors in eliminating the bias observed between OpenMC and OpenMOC in Chap. 5. The 1D slab and 2D fuel pin benchmarks described in Secs. 5.1.2 and 5.1.3, respectively, were modeled with SPH-corrected MGXS for the FSRs in the fuel and compared to the original results. The MGXS were computed using isotropic in lab scattering and a spatial tally mesh corresponding to the FSR mesh (see Tabs. 5.10 and 5.15). The OpenMOC fixed source and eigenvalue calculations were performed with 128 azimuthal angles and 0.01 cm track spacing. The OpenMOC fixed source calculations were converged to 10^{-5} on the average FSR scalar flux. A convergence criterion of 10^{-7} was used to converge the SPH factors in Eqn. 6.7. Finally, the energy-integrated FSR fission source was converged to 10^{-7} in the eigenvalue calculations with SPH-corrected MGXS.

The eigenvalue bias for the 1D slab and 2D fuel pin with SPH-corrected MGXS are presented in Sec. 6.3.1. The error profiles of the energy-dependent and spatially-dependent fluxes within the fuel FSRs, as well as the SPH factors in energy and space, are analyzed in Secs. 6.3.2 and 6.3.3, respectively.

6.3.1 Impact of SPH on Eigenvalues

The first case study compared the eigenvalues computed with and without SPH factors. The eigenvalue bias $\Delta\rho$ between OpenMC and OpenMOC is presented for a matrix of energy group structures and FSR discretization in Tabs. 6.1 and 6.2 for a 1D slab and 2D fuel pin, respectively. The bias without SPH factors is reproduced from Tabs. 5.10 and 5.15 for comparison purposes.

The data in the tables illustrate a large reduction in the eigenvalue bias with SPH-corrected MGXS. With only a few exceptions, the bias is consistently positive and ranges between 10 – 40 pcm for both geometries. In the case of the slab, the bias of -93 pcm with the finest energy group structure and FSR discretization was reduced by a factor of 7× to only 13 pcm with SPH-corrected MGXS. The use of SPH factors had an even greater impact on the bias for the fuel pin, where the bias of over -211 pcm was reduced by 70× to just -3 pcm for the finest energy and spatial discretization.

Although these results illustrate much better agreement between OpenMC and OpenMOC, it is interesting to note that a non-negligible bias remains in most cases. As was noted in Sec. 6.2.2, the reaction rates in non-fissile zones such as the moderator are not preserved with SPH which may contribute to the lingering eigenvalue bias. In addition, the remaining bias may be due to the fact that the reaction rate balance enforced with SPH factors assumes that the eigenvalue calculation with a multi-group method will produce the same neutron source distribution as MC. However, the eigenvalue source is not necessarily conserved since approximation errors from spatial and angular discretization will impact the multi-group method's solution. In conclusion, the eigenvalues between OpenMC and OpenMOC will identically match if the reaction rate errors in non-fissile zones are negligible, and OpenMOC computes the same eigenvalue source with SPH-corrected MGXS as that found by OpenMC.

6.3.2 Impact of SPH on the Energy-Dependent Flux

A second case study investigated the impact of SPH factors on the energy-dependent flux errors identified in Sec. 5.2.1. The error of OpenMOC's 70-group flux with respect to the reference OpenMC flux is displayed in Fig. 6-3 for the slab and pin. The errors are shown for the innermost and outermost FSRs in the fuel, along with the average error across all FSRs in the fuel. The flux errors without SPH-corrected MGXS are reproduced from Fig. 5-4 and are illustrated with dashed lines for comparison purposes.

As expected, the flux error is greatly reduced with SPH-corrected MGXS. The flux error profile is nearly flat in energy for innermost, outermost and all FSRs, unlike the

Table 6.1: The impact of SPH factors on the eigenvalue bias $\Delta\rho$ with varying energy group structures and FSR spatial discretizations for a 1D slab.

	FSR Discretization				
	1×	2×	4×	8×	16×
# Groups	Without SPH				
1	139	97	127	162	140
2	296	145	100	90	88
4	233	125	77	61	51
8	304	136	45	7	−9
16	329	139	38	−6	−23
25	266	83	−11	−54	−70
40	270	75	−29	−73	−89
70	280	76	−33	−81	−93
	With SPH				
1	33	−10	18	51	30
2	36	13	27	34	16
4	18	26	28	33	18
8	27	37	29	22	12
16	27	38	30	20	9
25	28	39	34	25	12
40	28	39	31	23	12
70	31	41	31	20	13

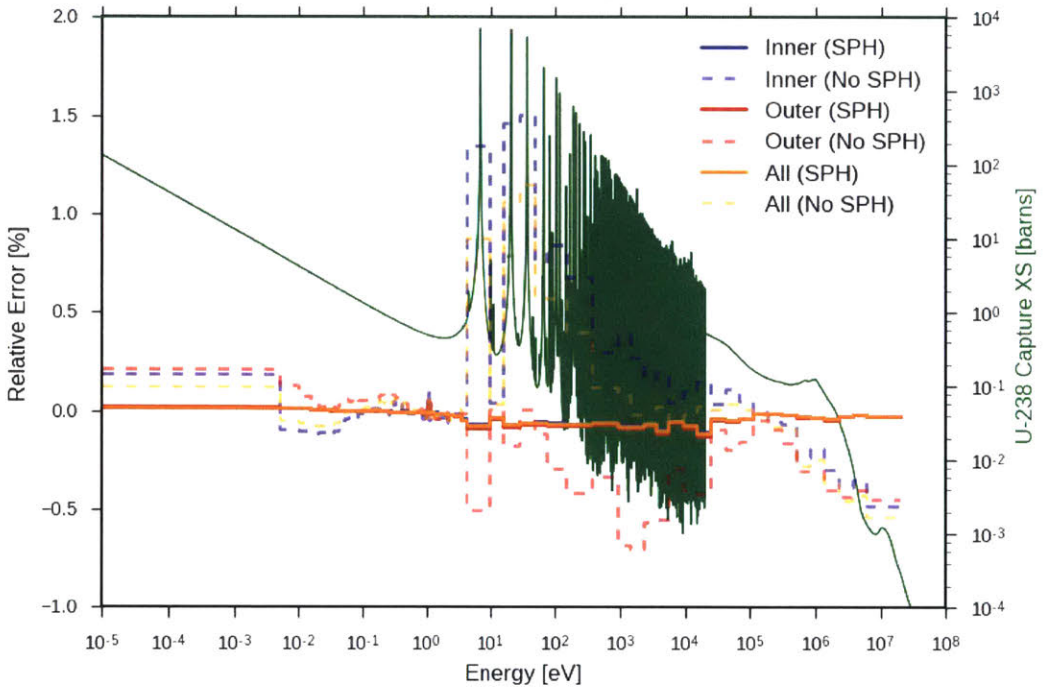
highly energy-dependent profiles observed without SPH. Furthermore, no systematic difference between the error profiles for the innermost and outermost FSRs can be discerned from the plots. The improvement in OpenMOC's flux – especially in those energy groups with large U-238 capture resonances – is responsible for the reduction in the eigenvalue bias with SPH-corrected MGXS presented in Sec. 6.3.1.

The SPH factors in each energy group are illustrated for the innermost and outermost FSRs in the slab and pin in Fig. 6-4. The SPH factors show nearly the opposite energy-dependent behavior of the flux errors without SPH factors. In general, the factors are less than unity in those groups with positive flux errors (*e.g.*, the fast groups), while the factors are greater than unity in groups with negative errors (*e.g.*, resonance groups in

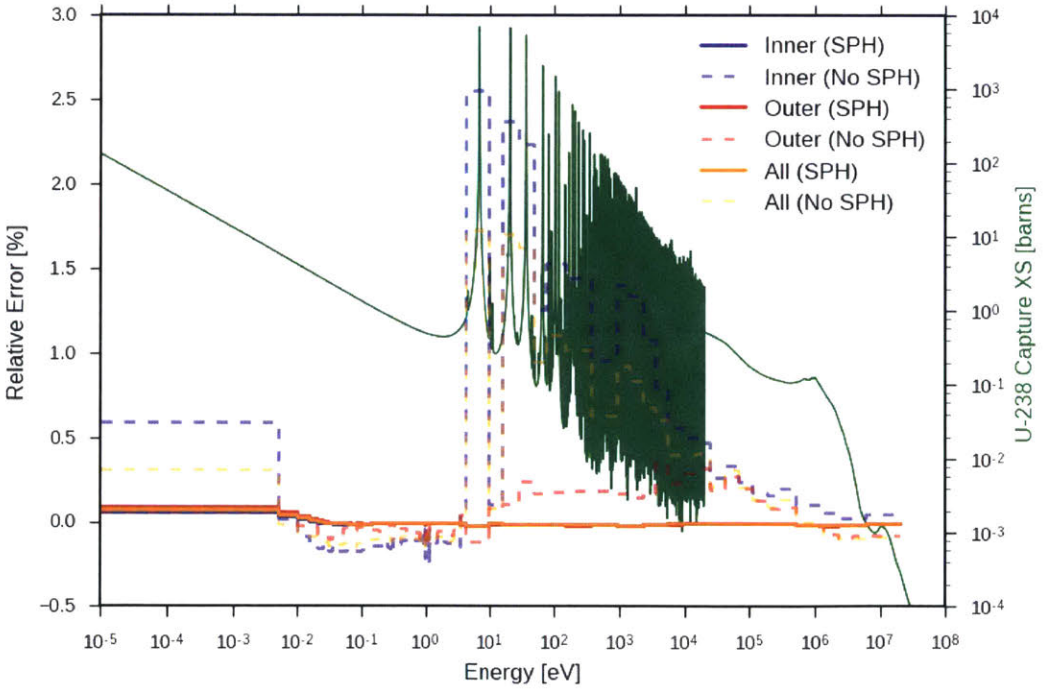
Table 6.2: The impact of SPH factors on the eigenvalue bias $\Delta\rho$ with varying energy group structures and FSR spatial discretizations for a 2D fuel pin.

	FSR Discretization				
	1×	2×	4×	8×	16×
# Groups	Without SPH				
1	80	92	55	83	66
2	141	87	29	50	34
4	27	−15	−43	−45	−57
8	26	−34	−85	−90	−102
16	35	−35	−91	−101	−111
25	−31	−105	−158	−170	−182
40	−38	−114	−174	−189	−202
70	−39	−117	−182	−196	−211
	With SPH				
1	19	25	−18	10	−14
2	25	20	−14	18	−6
4	7	10	2	12	1
8	4	13	0	12	2
16	5	13	0	10	4
25	5	13	2	12	−1
40	4	16	3	11	−2
70	4	17	2	12	−3

the innermost FSR). This behavior is expected since an SPH factor greater than unity will magnify the total cross section (see Eqn. 6.6) which will reduce the scalar flux, and vice versa. Furthermore, the deviation of the SPH factors from unity is roughly the same as the error in each respective group. For example, the flux in group 27 with the U-238 capture resonance at 6.67 eV exhibits an error of nearly 1.5% and 2.5% for the slab and pin, respectively. Similarly, the SPH factors in group 27 are approximately 0.984 and 0.970, or 1.6% and 3% reduced from unity, for the slab and pin.

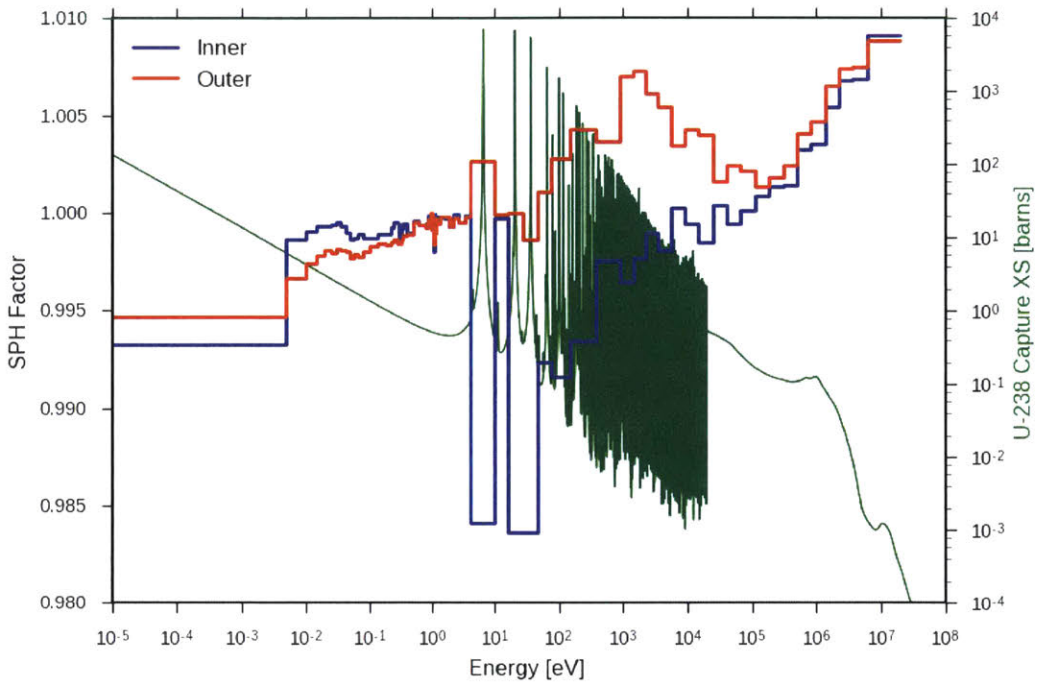


(a)

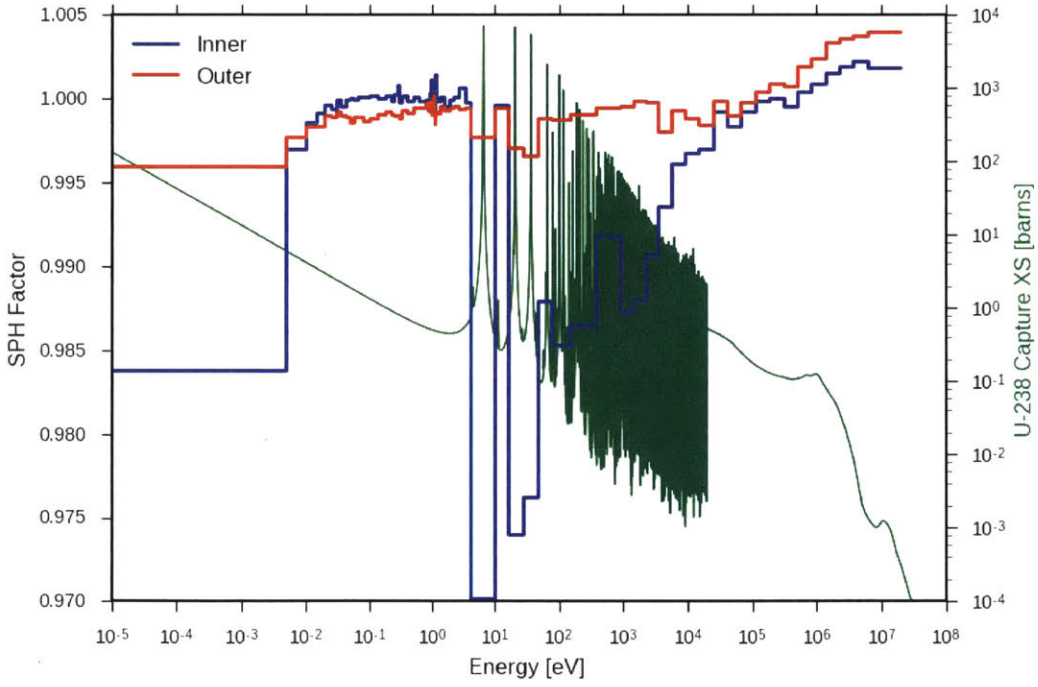


(b)

Figure 6-3: The energy-dependent relative error of the 70-group OpenMOC scalar flux with respect to the reference OpenMC flux in a 1D slab (a) and 2D fuel pin (b) for the innermost, outermost and all FSRs. The results correspond to Tables 6.1 and 6.2.



(a)



(b)

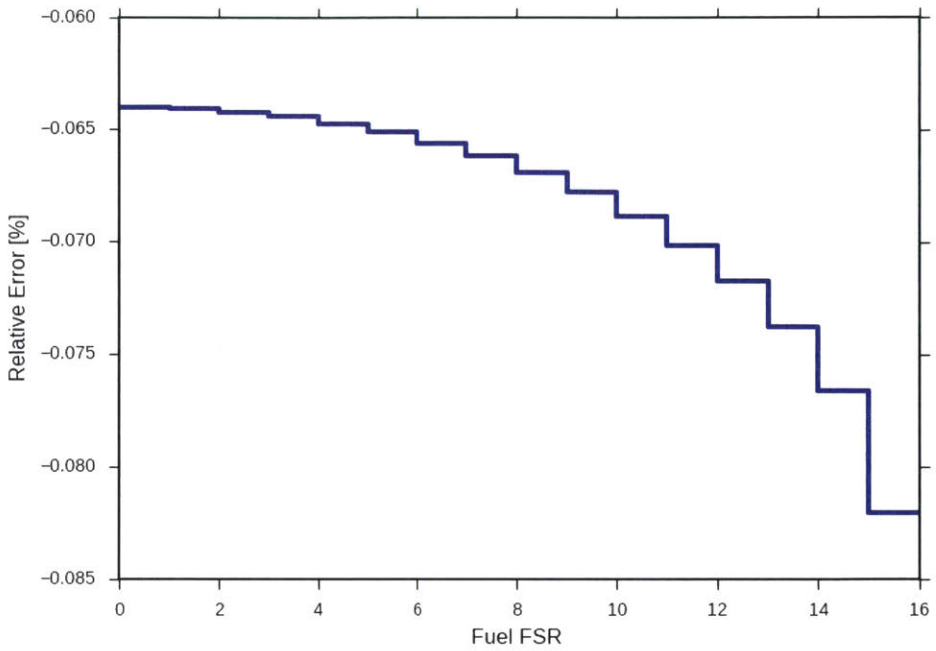
Figure 6-4: The energy-dependent SPH factors for a 70-group calculation in a 1D slab (a) and 2D fuel pin (b). The results correspond to Tables 6.1 and 6.2.

6.3.3 Impact of SPH on the Spatially-Dependent Flux

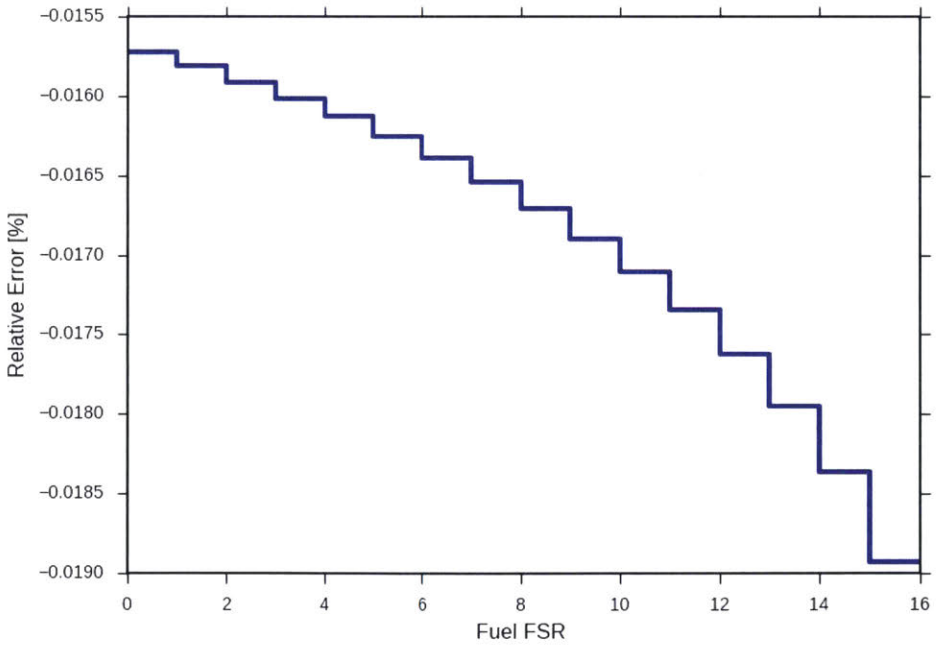
A final case study investigated the impact of SPH factors on the spatially-dependent 70-group flux errors identified in Sec. 5.2.2. The error of OpenMOC's flux with respect to the reference OpenMC flux is displayed in Fig. 6-5 for the slab and pin. The errors are shown for group 27 for each of the FSRs in the fuel. This plot can be compared to the spatially-dependent errors in Fig. 5-5 observed without SPH-corrected MGXS.

As expected, the flux error in all of the fuel FSRs is greatly reduced with the application of SPH factors. The maximum errors of approximately -0.082% and -0.019% with SPH compare to 1.3% and 2.5% without SPH for the slab and pin, respectively. The overall error profile as plotted appears similar in shape to the case without SPH factors in Fig. 5-5, though the fractional variation of the errors is far smaller with SPH. Interestingly, the errors with SPH are greatest in magnitude in the outermost FSRs nearest to the moderator. In contrast, the errors are greatest in magnitude in the innermost FSRs furthest removed from the moderator for the case without SPH factors. The reduction in the error across the fuel in group 27 is largely responsible for the reduction in the eigenvalue bias with SPH-corrected MGXS presented in Sec. 6.3.1.

The group 27 SPH factors in each of the FSRs in the fuel is illustrated in Fig. 6-6. The SPH factors show nearly the opposite spatially-dependent behavior of the flux errors without SPH factors. As discussed in Sec. 6.3.2, the factors are less than unity in those FSRs with positive flux errors, while the factors are greater than unity in groups with negative errors (*e.g.*, the outermost ring in the fuel pin). This behavior is expected since an SPH factor greater than unity will magnify the total cross section (see Eqn. 6.6) which will reduce the scalar flux, and vice versa. Furthermore, the deviation of the SPH factors from unity is roughly the same as the error in each respective group. For example, the flux in the innermost FSR exhibits an error of nearly 1.5% and 2.5% for the slab and pin, respectively. Similarly, the SPH factors in the innermost FSRs are approximately 0.984 and 0.970, or 1.6% and 3% reduced from unity, for the slab and pin.

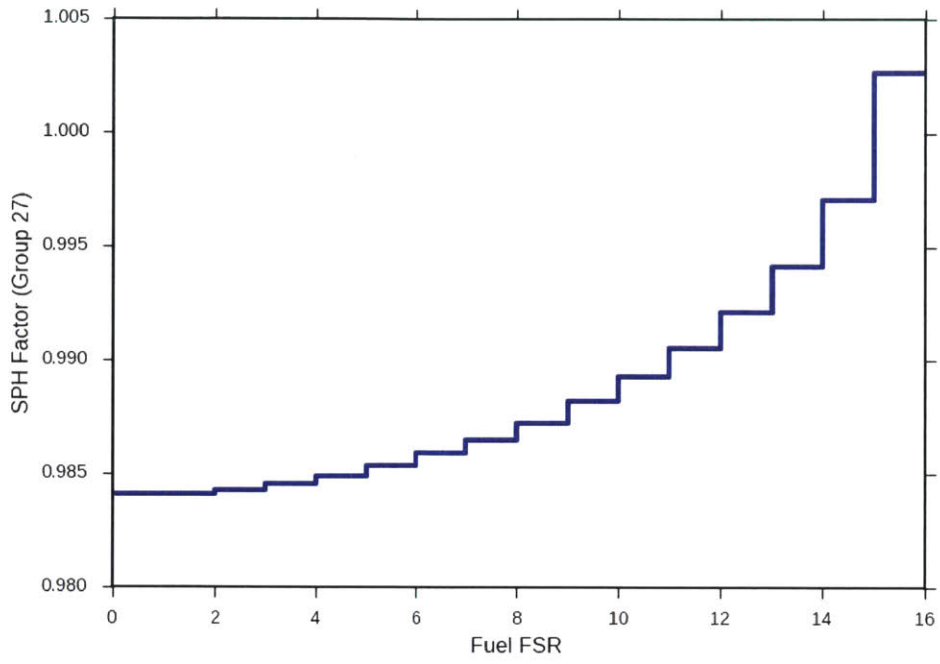


(a)

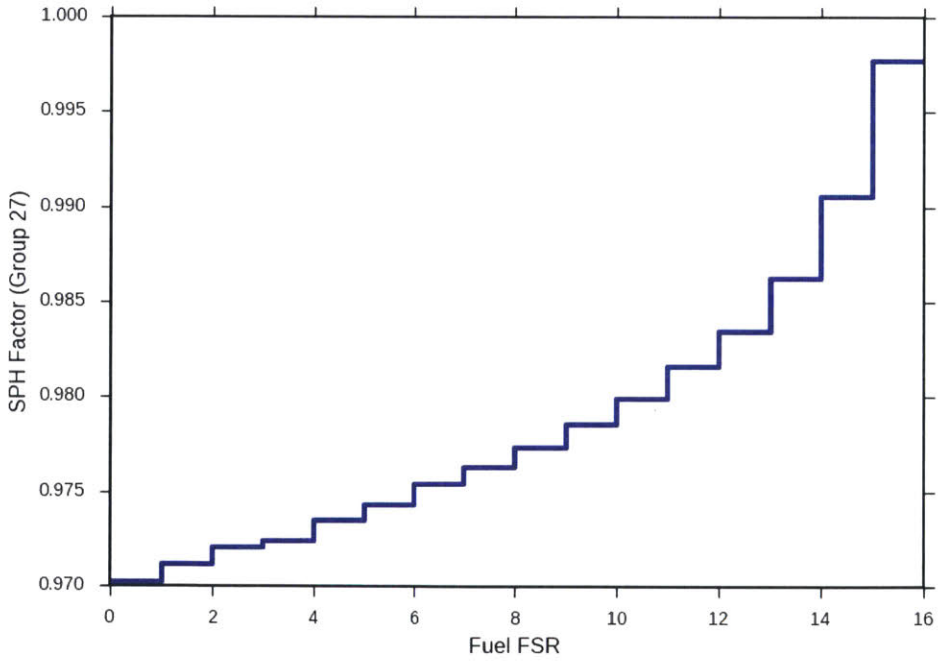


(b)

Figure 6-5: The spatially-varying relative error of the OpenMOC scalar flux with respect to the reference OpenMC flux for a 1D slab (a) and 2D fuel pin (b) in group 27. The results correspond to Tables 6.1 and 6.2.



(a)



(b)

Figure 6-6: The spatially-varying SPH factors for a 1D slab (a) and 2D fuel pin (b) in group 27. The results correspond to Tables 6.1 and 6.2.

The flux errors in the resonance groups, and the resulting eigenvalue bias between OpenMC and OpenMOC, is largely resolved with SPH factors.

6.4 Shortcomings of SPH Factors

This chapter identified the flux separability approximation as the culprit in the bias observed between OpenMC and OpenMOC for simple heterogeneous benchmarks in Chap. 5. The SPH factor approach was introduced as one method to force reaction rate preservation in multi-group methods with MGXS generated from Monte Carlo. Although the use of SPH factors was demonstrated to greatly improve the agreement between OpenMC and OpenMOC in Sec. 6.3, the SPH approach suffers from a number of shortcomings which may preclude it from further use in this setting in the future. In particular, the SPH scheme requires knowledge of the reference source distribution, is dependent on the spatial discretization mesh, and is indiscriminate between various sources of approximation error.

Perhaps the most significant weakness of the SPH approach in the context of MC-generated MGXS is that it requires knowledge of the true eigenvalue source distribution. The goal of generating MGXS with MC is to enable multi-group methods to accurately solve eigenvalue problems for the source and flux distributions. If a reference source must first be computed with MC in order to compute SPH-corrected MGXS, then there is no reason to perform a subsequent multi-group calculation since the solution is already known from MC.

Furthermore, the SPH scheme is wholly dependent on the spatial discretization used by downstream deterministic multi-group methods. In particular, the reference source in Eqn. 6.2 must be calculated from MC tallies in each of the spatial mesh cells used by the SPH iteration scheme. In the implementation presented here, the reference source must be computed from OpenMC tallies for each of the FSRs in the OpenMOC eigenvalue calculation. This is problematic since a finer FSR discretization will potentially require more particle histories to converge the tallies needed to compute the reference source

on the finer spatial tally mesh.

Finally, the SPH scheme attempts to preserve reaction rates between OpenMC and OpenMOC irregardless of the types of approximation errors which may lead to bias between the two codes. This chapter used SPH factors to mitigate the errors due to the flux separability approximation. However, the SPH scheme simultaneously attempts to also "correct" the MGXS to account for spatial, angular and energy discretization errors, approximation error due to the treatment of the scattering kernel, and/or any other approximation error inherent in multi-group methods. For example, the results presented in Sec. 6.3.1 illustrated a broadly consistent agreement between OpenMC and OpenMOC eigenvalues for all energy group structures and FSR discretizations, even though systematic trends in the bias were observed in both energy and space without SPH factors. This highlights the fact that SPH "corrected" for not only the lack of angular dependence in the total MGXS, but also errors inherent to the energy group structures and FSR discretizations. As a result, the same discretization parameters must be used in both the fixed source calculations in the SPH iteration scheme, as well as subsequent eigenvalue calculations with SPH-corrected MGXS[31].

[31] For OpenMOC, the same number of azimuthal angles, track spacing and FSR discretization must be used in both the SPH factor and eigenvalue calculations.

The SPH factor approach is complicated by the need to compute a reference fixed source with MC for the spatial discretization mesh used in multi-group methods. In addition, SPH factors do not simply correct for the error due to the flux separability approximation, but instead indiscriminately correct for all sources of approximation error between MC and multi-group methods.

6.5 Future Work

As a result of the shortcomings to the SPH approach, it is unclear whether the SPH factors may be broadly applied to correct for the flux separability approximation in MGXS generated from MC. Future work should investigate whether a universal set of SPH factors may be tabulated for known geometries (*e.g.*, PWR fuel pins). For example, if the SPH factors in the resonance groups are relatively invariant to the fuel enrichment, moderator density, burnup, neighboring pin types, etc. then a single set of SPH factors may be computed for an infinite pin cell and applied to each fuel pin in heterogeneous PWR lattice and full-core calculations.

Although it may be possible to universally apply pre-tabulated SPH factors to fixed geometric configurations, it will likely be necessary to develop alternative methods to account for the angular dependence of the total MGXS. For example, the angular dependence of the total MGXS may be adequately embedded into the scattering kernel using the Consistent-P approximation [7] (also known as the BHS approximation). Alternatively, a coarse set of angular-dependent MGXS may mitigate most of the bias observed between OpenMC and OpenMOC. For example, a simple approximation might model two different total MGXS for neutrons entering or leaving a fuel pin. Although a coarse angular scheme would not capture the high degree of angular variation illustrated in Fig. 6-2, it might capture enough to adequately resolve the bias. One challenge to this approach would be to define a general way to accommodate different FSR discretizations within each fuel pin cell.

Highlights

- The bias between OpenMC and OpenMOC is the result of the flux separability approximation which uses the scalar rather than the angular flux to collapse the total MGXS in energy and space.
- The most rigorous solution would require the use of angular-dependent total MGXS. However, most deterministic multi-group methods, including OpenMOC, are not equipped to use angular-dependent MGXS.
- SPH factors are introduced here as one approach to force reaction rate preservation in multi-group methods which use MC-generated MGXS.
- The SPH factor scheme corrects the total MGXS to enforce neutron balance with a reference fixed source computed from MC tallies.
- The flux errors and eigenvalue bias between OpenMC and OpenMOC was largely resolved with SPH factors for a 1D slab and 2D fuel pin.
- It is unclear if a generalizable scheme based upon SPH factors may be used to correct for the flux separability approximation.
- Future work should investigate methods to account for the angular dependence of total MGXS in order to adequately preserve reaction rates in fine-mesh transport methods with MC-generated MGXS.

Part IV

Statistical Clustering

Chapter 7

Benchmark Models and Reference Results

7.1 Motivation

This thesis is motivated by the desire to obtain Monte Carlo-quality solutions with computationally efficient deterministic neutron transport methods. In particular, this work evaluates the use of MC for reactor agnostic MGXS generation for high-fidelity neutron transport simulations. The analysis in Part III was dedicated to the evaluation of approximation errors in MGXS and multi-group transport methods which appear in the modeling of even simple heterogeneous benchmarks such as a 2D fuel pin cell. The chapters in Part IV develop a new approach based upon statistical clustering to capture spatial self-shielding effects which occur only in large, complex heterogeneous geometries. This chapter presents several heterogeneous PWR benchmark models which are used to evaluate the efficacy of the new methodology for MGXS generation throughout the subsequent chapters in Part IV.

Each of the heterogeneous benchmarks presented here is derived from the Benchmark for Evaluation And Validation of Reactor Simulations (BEAVRS) PWR model [62]. A series of six benchmarks were designed to introduce increasingly complex heterogeneous features – and corresponding spatial self-shielding effects – to the models in order to

understand their implications for accurate pin-wise MGXS generation. The impact of fuel enrichment, Control Rod Guide Tubes (CRGTs), Burnable Poisons (BPs), inter-assembly currents, water reflectors, steel baffles and the core barrel and vessel is considered. This chapter details the geometric and material specifications for the individual fuel assemblies, multiple assembly colorsets, and full BEAVRS core modeled with different MGXS generation schemes throughout Part IV.

In addition, this chapter presents the reference results for each of the six geometries used to evaluate the accuracy of MGXS generation with statistical clustering. Since this thesis aims to enable MC accuracy in deterministic transport simulations, OpenMC was used to generate the reference results for each of the six benchmarks. This chapter quantifies the source convergence rate for each of the benchmark models using Shannon entropy. In addition, a series of OpenMC simulations were used to calculate reference eigenvalues, pin-wise fission rates, and pin-wise U-238 capture rates for each benchmark.

The eigenvalue is a key integral quantity used to assess the reactivity of a reactor. The fission rates are directly related to the relative power density of each fuel pin which is important for fuel depletion as well as thermal hydraulic feedback. The U-238 capture rates result in the production of Pu-239 which contributes up to 40% of the power produced from fission in PWRs at the end-of-life (EOL). Hence, the spatial distributions of the fission rates and U-238 capture rates must be correctly modeled for accurate high-fidelity depletion calculations. The reference results for each of these three metrics were computed with highly converged OpenMC simulations. It is important to recall from Sec. 1.3 that this thesis aimed to generate MGXS with MC in significantly less time[32] than would be required to generate reference solutions with MC. The methodology developed in Part IV aims to make this possible with statistical clustering of "noisy" MC tallies for MGXS generation.

This chapter outlines the geometric and isotopic specifications for each of the six benchmark models in Sec. 7.2. The reference results computed with OpenMC for each of the benchmark models are presented in Sec. 7.3.

[32]In this case, "time" is synonymous with the number of particle histories simulated with MC.

7.2 Benchmark Configurations

The heterogeneous benchmarks modeled throughout Part IV were based on the BEAVRS PWR model [62]. The BEAVRS model is a highly-detailed PWR specification which was created to validate high-fidelity core analysis methods. Five of the six benchmark models are based upon sub-components (*e.g.*, fuel assemblies) of the full core BEAVRS model, while the sixth benchmark is of the full BEAVRS core model. Although BEAVRS is an axially heterogeneous 3D core model, each of the benchmarks were fabricated in 2D due to the geometric constraints in OpenMOC. In particular, the 2D radial heterogeneity in each benchmark is taken from the axial mid-plane of the BEAVRS model. The geometries used two planar surfaces perpendicular to the z-axis with reflective Boundary Conditions (BCs) to model the benchmarks as infinitely long in the axial direction (*e.g.*, each benchmark is infinitely homogeneous along the z-axis).

The materials and isotopic compositions in each of the benchmarks are detailed in Sec. 7.2.1, while the geometric specifications for each pin cell type – fuel pin, instrument tube, CRGT and BP – are tabulated in Sec. 7.2.2. The first three benchmarks were based upon individual BEAVRS fuel assemblies with different fuel enrichments and CRGT and BP locations as discussed in Sec. 7.2.3. The geometric configuration of 2×2 fuel assembly colorsets with and without a water reflector are highlighted in Sec. 7.2.4. Finally, the key parameters for the full BEAVRS core model are presented in Sec. 7.2.5.

7.2.1 Materials

The models described in this section were comprised of materials from the BEAVRS model. The six benchmarks include 1.6%, 2.4% and 3.1% enriched UO_2 fuel, borated water[33], zircaloy, helium, air, borosilicate glass and stainless steel. The densities and isotopic compositions for each material are detailed in the BEAVRS specifications [62] and are reproduced in App. B.1. Each of the materials was modeled with cross sections at 600K for hot zero power (HZP) conditions.

[33]The water consisted of 975 parts per million (ppm) boron, the critical concentration for the all rods out configuration [62]

7.2.2 Pin Cells

The six benchmarks were composed of four different types of pin cells – fuel pins, instrument tubes, control rod guide tubes and burnable poisons. Each of the four pin cell types is displayed in Fig. 7-1. The different material types are indicated with different colors. The fuel pin in Fig. 7-1a contains UO_2 fuel, a helium gap and zircaloy cladding. The CRGT in Fig. 7-1b is modeled in the control rod out configuration from above the dashpot and includes borated water surrounded by zircaloy cladding. The instrument tube in Fig. 7-1c is filled with air surrounded by two tubes of zircaloy cladding separated by borated water. The BP in Fig. 7-1d is the geometry from above the dashpot and consists of eight layers of air, steel, borosilicate glass and zircaloy. Each pin cell is surrounded by borated water which serves as the neutron moderator and coolant. The radii for each material zone is detailed in Tab. B.9. The pin cell pitch is 1.25984 cm.

7.2.3 Fuel Assemblies

The first three benchmark models are based upon 2D models of individual fuel assemblies extracted from the full core BEAVRS model. Each assembly consists of a 17×17 rectilinear array of pin cells with a total height and width of 21.41728 cm. The intra-pin egg-crate grid spacer and grid sleeve separating each fuel assembly in the full core BEAVRS model are not included in the individual models of each fuel assembly. The assemblies are modeled with reflective BCs to simulate an infinite repeating lattice of each fuel assembly.

The sequence of fuel assembly benchmarks are designed to investigate the impact of fuel enrichment, CRGTs and BPs on spatially self-shielded MGXS. The first fuel assembly benchmark depicted in Fig. 7-2 consists of 264 fuel pins with 1.6% enriched UO_2 fuel, 24 CRGTs, and a single central instrument tube. The second benchmark shown in Fig. 7-3 is of the same geometric configuration, but is composed of 3.1% enriched UO_2 fuel. The third benchmark illustrated in Fig. 7-4 includes 3.1% enriched UO_2 fuel with a mixture of 20 BPs, 4 CRGTs and a single central instrument tube. Although the BEAVRS model is composed of assemblies with many different BP configurations, only a single assembly with BPs was studied for practical reasons.

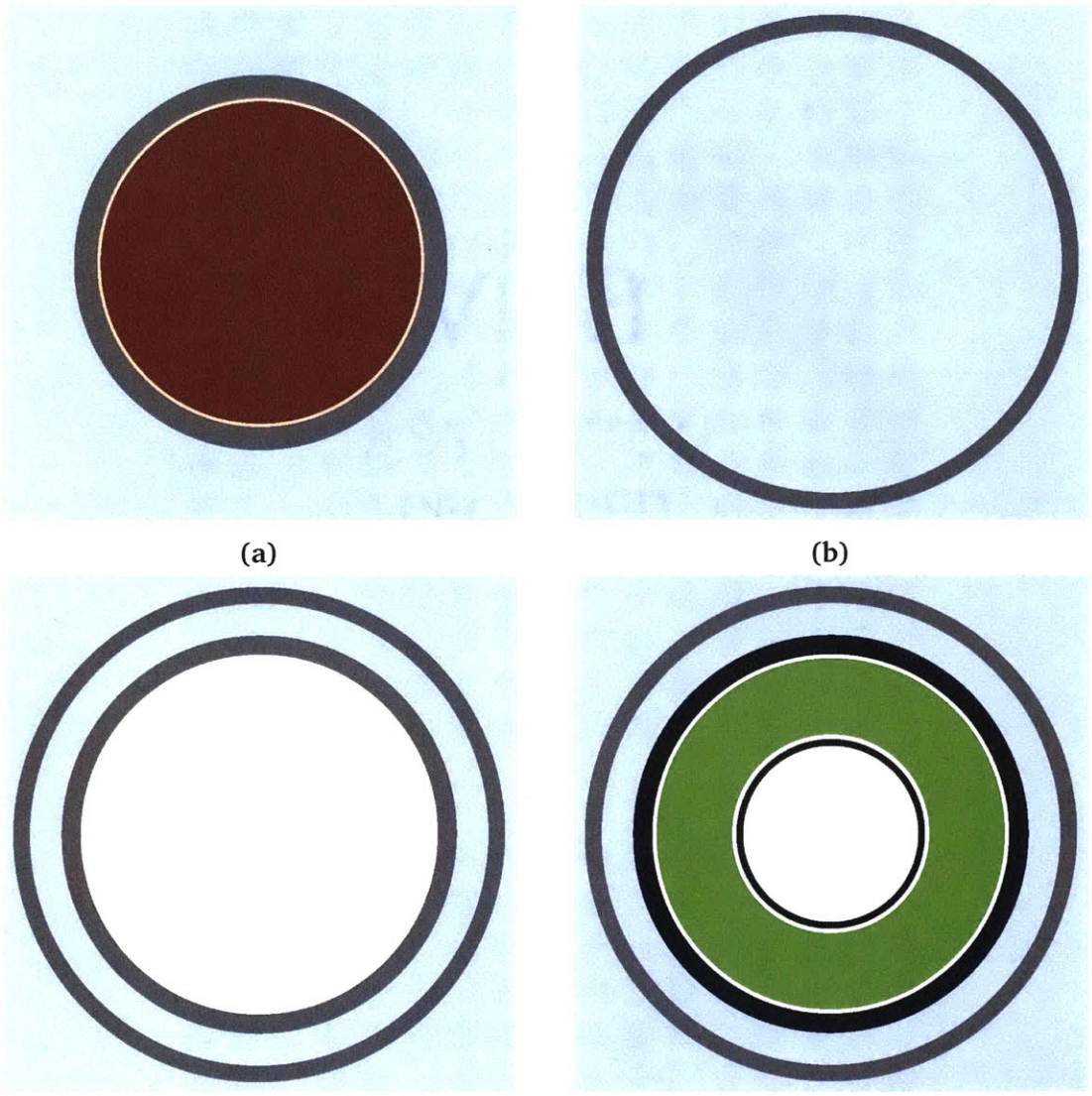


Figure 7-1: 1.6% enriched fuel pin (a), control rod guide tube (b), instrument tube (c) and burnable poison (d). Light blue is borated water, red is UO_2 fuel, gray is zircaloy, brown is helium, white is air and green is borosolicate glass.

7.2.4 2×2 Assembly Colorsets

Two benchmarks were constructed from 2×2 colorsets of fuel assemblies from the BEAVRS model presented in Sec. 7.2.3. The pitch between fuel assemblies in the colorsets is 21.41728 cm (the height/width of each assembly). The intra-pin egg-crate grid spacer and grid sleeve separating each fuel assembly in the full core BEAVRS model are not included in the 2×2 colorsets. The first colorset is modeled with periodic BCs on all sides to simulate an infinitely repeating lattice of fuel assemblies. The second

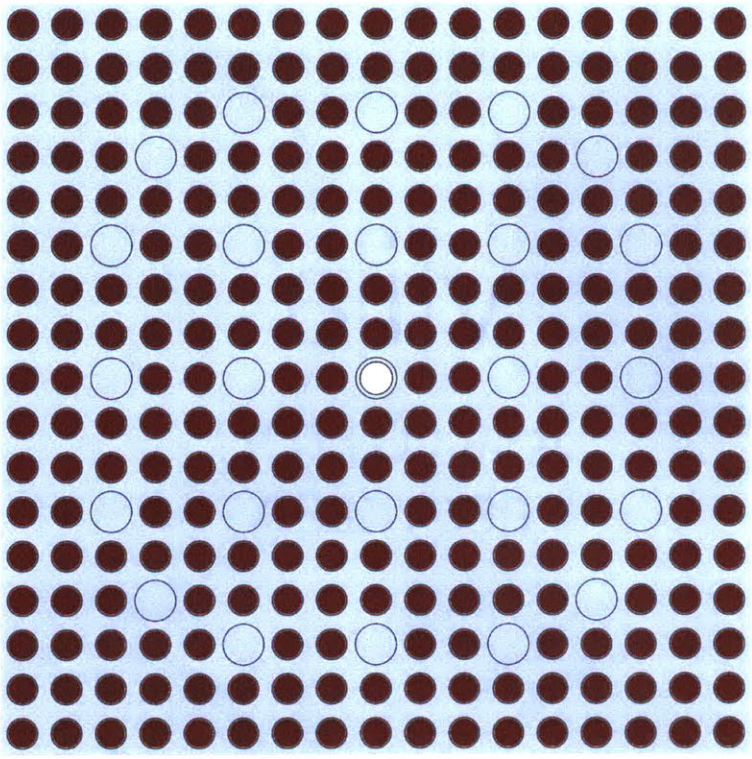


Figure 7-2: A 1.6% enriched UO_2 fuel assembly without BPs.

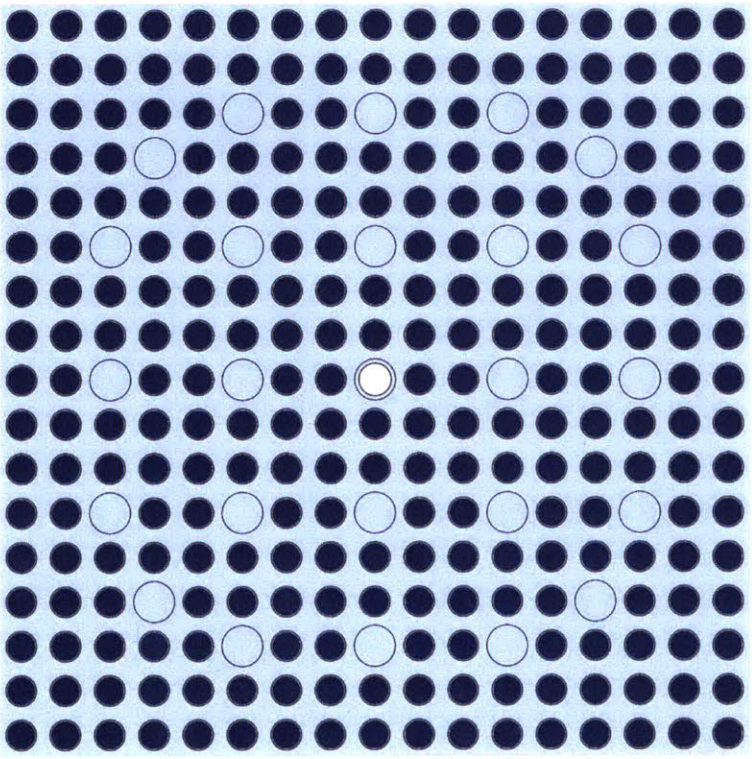


Figure 7-3: A 3.1% enriched UO_2 fuel assembly without BPs.

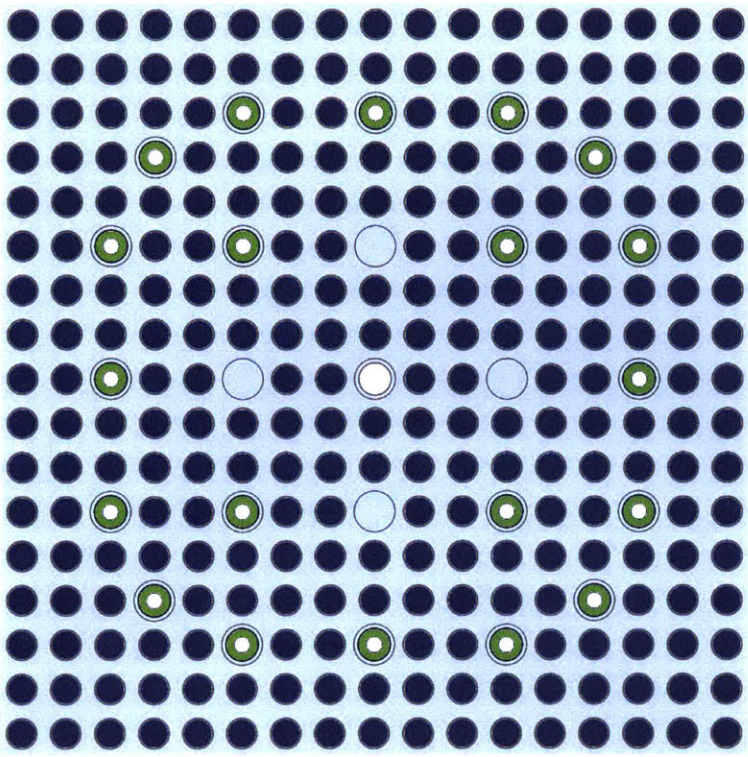


Figure 7-4: A 3.1% enriched UO_2 fuel assembly with 20 BPs.

colorset is surrounded by a water reflector on the bottom and right that is of the same width as a fuel assembly. The reflected colorset does not include the stainless steel baffle surrounding the fuel assemblies adjacent to the water reflector in the full core BEAVRS model. The reflected colorset includes reflective BCs on the top and left (adjacent to the fuel assemblies) with vacuum BCs on the bottom and right (adjacent to the reflector).

The first 2×2 colorset model shown in Fig. 7-5 is composed of a checkerboard pattern of the fuel assemblies of 1.6% enriched UO_2 without BPs (Fig. 7-2) and 3.1% enriched UO_2 with 20 BPs (Fig. 7-4). The first colorset benchmark is designed to investigate the effects of inter-assembly spatial heterogeneities on the spatially self-shielded MGXS of fuel pins of different enrichments (*i.e.*, from different fuel assemblies) placed adjacent to one another. The second benchmark illustrated in Fig. 7-6 is the same 2×2 colorset of fuel assemblies, but is surrounded by a water reflector. This benchmark is designed to quantify the impact of the moderation provided by the reflector, as well as the leakage of neutrons through the reflector, on spatially self-shielded MGXS.

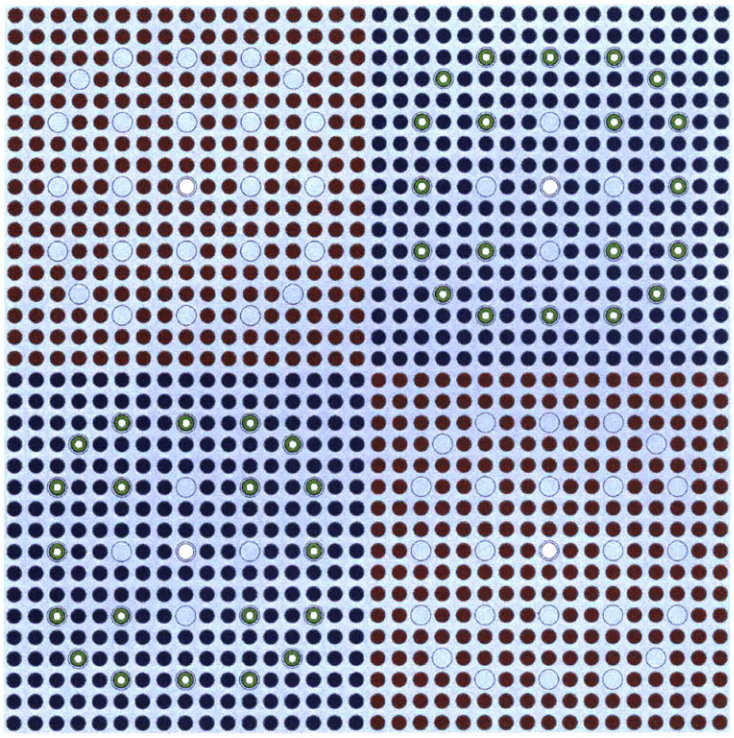


Figure 7-5: A 2×2 colorset of BEAVRS assemblies with periodic BCs.

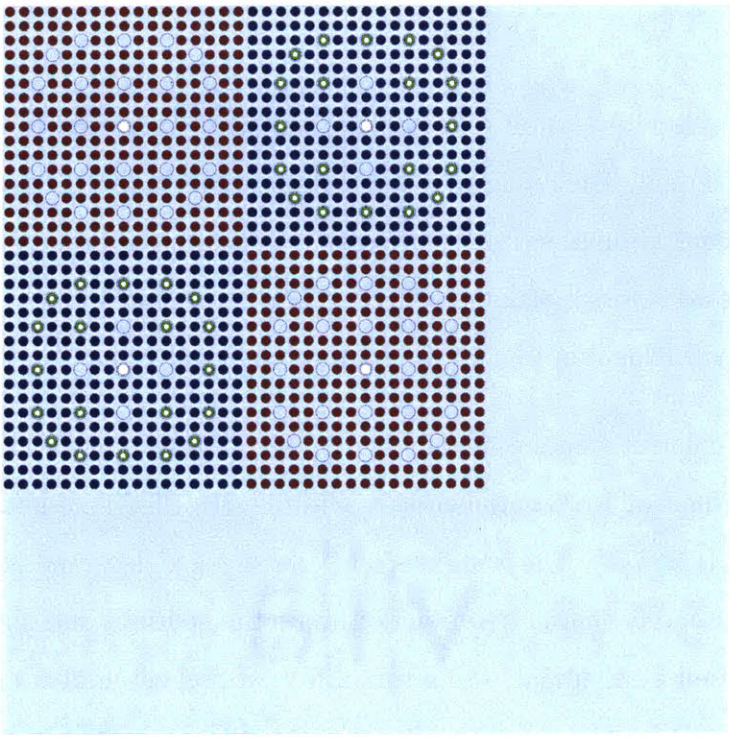


Figure 7-6: A 2×2 colorset of BEAVRS assemblies surrounded by a water reflector. Reflective BCs were applied on the left/top, with vacuum BCs on the right/bottom.

7.2.5 BEAVRS Quarter Core

The sixth and final benchmark is a 2D planar slice of the top right quadrant of the quadrant symmetric BEAVRS core model[34] at the axial mid-plane as shown in Fig. 7-7. The fuel assemblies in the model are configured according to the cycle 1 loading pattern detailed in the BEAVRS specifications [62]. The quarter core model includes assemblies with 1.6%, 2.4% and 3.1% enriched UO_2 fuel depicted as red, blue and yellow in Fig. 7-7, respectively. The benchmark model is the all rods out configuration with a critical boron concentration of 975 ppm [62].

The intra-pin egg-crate grid spacer and grid sleeve separating each fuel assembly in the 3D core BEAVRS model are not included in 2D model at the axial mid-plane. All other radial heterogeneities in the BEAVRS specifications are included in this model, including the inter-assembly water gaps, stainless steel baffle surrounding the fuel assemblies, core barrel, neutron shield panels and pressure vessel. The space outside of the pressure vessel is filled with air with vacuum BCs applied to planar surfaces on the top, bottom, left and right. The 2D quarter core BEAVRS model builds upon the preceding five simpler benchmarks to explore the impact of radial heterogeneities in a realistic core configuration on spatially self-shielded MGXS.

7.3 Reference Results

This section presents the reference results computed using OpenMC for each of the six benchmarks presented in the preceding section. As discussed in Sec. 7.3.1, OpenMC was employed to compute the Shannon entropy to determine the number of batches of particle histories needed to reach source stationarity. Sec. 7.3.2 presents the converged k_{eff} eigenvalue computed by OpenMC for each of the six benchmarks. Finally, Sec. 7.3.3 and Sec. 7.3.4 present the reference pin-wise spatial distributions of fission rates and U-238 capture rates in each benchmark as tallied in OpenMC along with their associated statistical uncertainties.

[34]The BEAVRS model was made quadrant symmetric by replacing all instrument tubes with empty CRGTs.

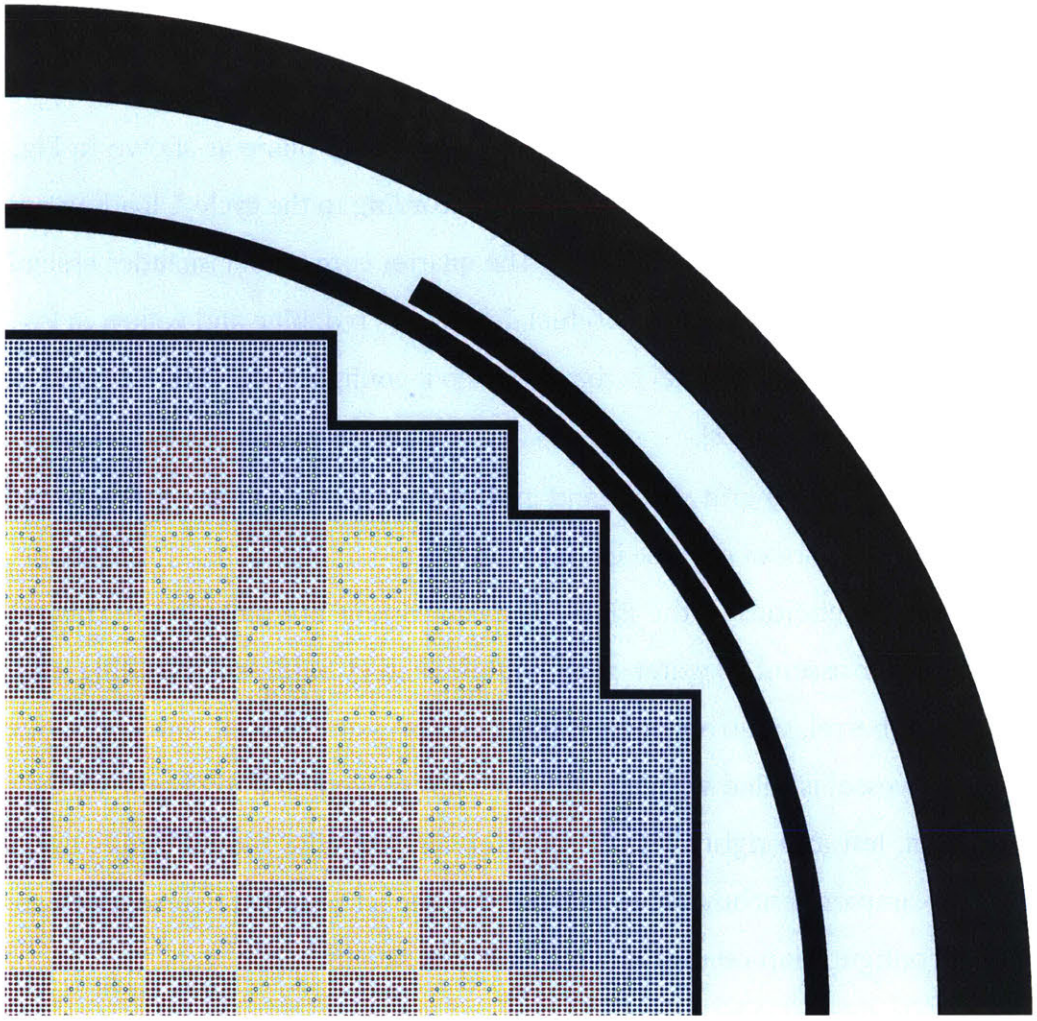


Figure 7-7: The 2D quarter core BEAVRS model.

The ENDF/B-VII.1 continuous energy cross section libraries evaluated at 600K provided by the MCNP code [38] were used by OpenMC for all simulations. It should be noted that isotropic in lab scattering was employed for all reference calculations with OpenMC's iso-in-lab feature (see Sec. 4.2.3). Although isotropic in lab scattering is a poor approximation for LWRs, it eliminated scattering source anisotropy as one possible cause of approximation error between OpenMC and OpenMOC[35] in order to isolate approximation errors resulting from spatially self-shielded MGXS.

The reference solutions for each assembly and colorset benchmark model was computed with 100 inactive and 900 active batches of 10^7 particle histories per batch. The

[35] At the time of this writing, OpenMOC employed an isotropic in lab neutron scattering kernel.

reference solution for the quarter core BEAVRS model was computed with 200 inactive and 800 active batches of 10^8 histories per batch. Each OpenMC simulation was performed in parallel on ten compute nodes on the Falcon supercomputer at Idaho National Laboratory. Each compute node contained two dual socket Intel Xeon E5-2680 CPUs with 12 cores and 132 gigabytes of DRAM[36]. Four MPI processes were launched on each node (two MPI processes per CPU) with 6 OpenMP parallel threads per MPI process.

7.3.1 Source Stationarity

The first metric that was evaluated for each of the six benchmarks was the stationarity of the fission source distribution. The initial distribution of fission source sites is uniformly distributed in space across fissile material zones by OpenMC for the first batch of particle histories. Each subsequent batch of fission source sites is drawn from a bank of fission sites populated during the preceding batch of particle histories. In eigenvalue calculations, the distribution of fission source sites must reach stationarity before tallying integral quantities such as the eigenvalue or reaction rates.

The Shannon entropy is a commonly used diagnostic to measure source stationarity in MC eigenvalue calculations [80]. To compute the Shannon entropy, a mesh of M mesh cells is superimposed across the geometry and the number of fission source cites in each mesh cell is tabulated. The empirical multinomial probabilities p_i for fission source sites is computed for each mesh cell i as the ratio of sites in that cell to the total number of source sites in the geometry. The Shannon entropy H is then computed from the multinomial probability distribution from the following equation:

$$H = \sum_{i=1}^{M} p_i \log_2 p_i \tag{7.1}$$

The Shannon entropy provides a single scalar value which characterizes the spatial distribution of fission source sites. By monitoring the value of H for each batch, one may determine the number of inactive batches needed to reach source stationarity.

The Shannon entropy was computed using a rectilinear pin-wise mesh for each of the

[36]Dynamic Random Access Memory.

six benchmark models and plotted in Fig. 7-8. The entropies in the plot are normalized to the entropy at the final 1000th batch in order to standardize the entropies for comparison in the same plot. The entropies for each of the three fuel assembly benchmarks, as well as the 2×2 colorset without a reflector, all lie very near unity and do not exhibit any convergence behavior in the plot. The entropy for the 2×2 colorset with a reflector appears to converge to near unity within 30 batches. Finally, the entropy for the high dominance ratio BEAVRS quarter core appears to converge within 200 batches.

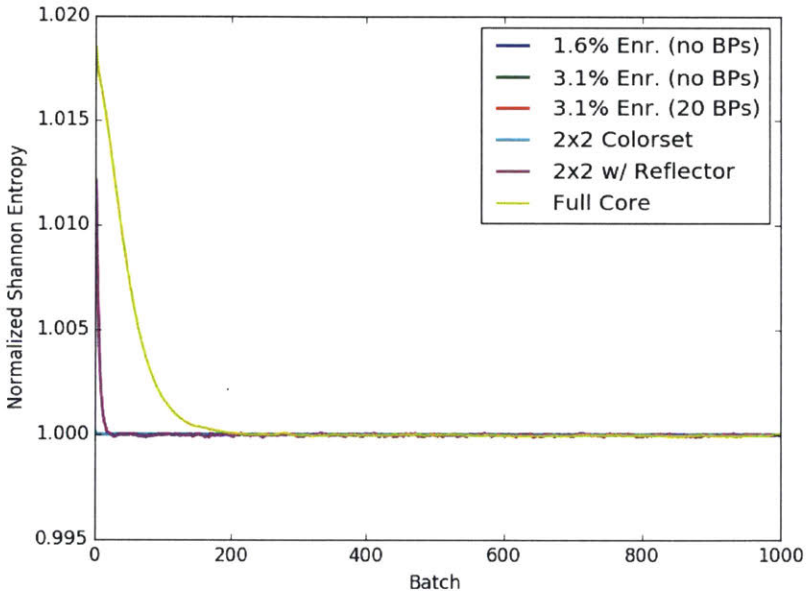


Figure 7-8: Shannon entropy source convergence for BEAVRS geometries.

As expected, the number of batches required to reach source stationarity increases with the benchmark size and heterogeneity, and the corresponding increase in the dominance ratio. This analysis of the Shannon entropy for each benchmark was used to determine an appropriately conservative number of inactive batches to employ in all subsequent OpenMC simulations of each benchmark model. In particular, all OpenMC simulations of the single assembly and 2×2 colorsets use 100 inactive batches prior to tallying the eigenvalue, reference reaction rate distributions and MGXS. Similarly, 200 inactive batches are used for the quarter core BEAVRS model.

7.3.2 Eigenvalues

The reference eigenvalues were computed for each of the six benchmarks and are listed in Tab. 7.1. The OpenMC "combined" eigenvalue estimator is reported along with the associated one sigma uncertainty of 1 pcm or less for each of the benchmarks. The total runtimes required for the OpenMC simulations to generate the reference eigenvalues are also reported in the table.

Table 7.1: Reference k_{eff}^{OpenMC} for heterogeneous benchmarks.

Benchmark	k_{eff}^{OpenMC}	Runtime [core-hours]
1.6% Enriched Assembly (no BPs)	0.99326 ± 0.00001	1,005
3.1% Enriched Assembly (no BPs)	1.21657 ± 0.00001	891
3.1% Enriched Assembly (20 BPs)	1.03315 ± 0.00001	879
2×2 Colorset	1.01814 ± 0.00001	903
2×2 Colorset w/ Reflector	0.94574 ± 0.00001	895
BEAVRS Quarter Core	1.02446 ± 0.00001	10,892

As expected, the eigenvalues increase with enrichment and decrease with the presence of BPs for the single fuel assembly benchmarks. The eigenvalue for the 2×2 colorset is reduced with the addition of a reflector due to leakage, and to a lesser extent, absorption in the reflector. Although the quarter core BEAVRS model is a critical configuration, the eigenvalue is nearly 2500 pcm supercritical due to the 2D and iso-in-lab scattering approximations[37]. The eigenvalues reported in Tab. 7.1 are used to validate the OpenMOC simulations with MGXS generated by OpenMC throughout the following chapters.

7.3.3 Fission Rate Spatial Distributions

The reference energy-integrated fission rate spatial distributions for each of the six benchmarks were computed using rectilinear, pin-wise tally meshes in OpenMC. The fission rates were volume-integrated across each fuel pin and include fission from all

[37] An identical OpenMC simulation of the 3D full core BEAVRS model using normal anisotropic scattering produced an eigenvalue of 0.99922 ± 0.00000.

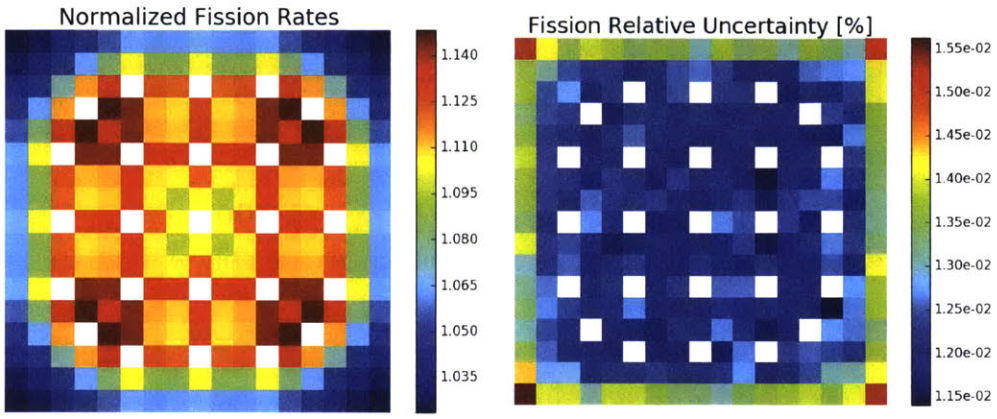
nuclides (only U-235 and U-238 for the fresh PWR UO_2 fuel in the benchmarks). The fission rates were normalized to the mean of all non-zero fission rates in each benchmark. The percent relative errors for the tallied fission rates was computed from the ratio of the sample standard deviation to the mean fission rate in each fuel pin. The fission rates for the quarter core BEAVRS model were appropriately tiled to display the equivalent distribution across the quadrant symmetric full core.

The fission rate spatial distributions and percent relative errors for each of the six benchmarks are presented as heatmaps in Figs. 7-9 to 7-14. The fission rates in the instrument tubes, CRGTs and BPs are all zero and are illustrated in white. The color bars for each of the heatmaps range from the minimum non-zero fission rate to the maximum fission rate for all fuel pins in each benchmark geometry. The statistical error at the final 1000^{th} batch is less than or equal to 0.5% for all of the benchmarks. These reference fission rate spatial distributions are used to validate the OpenMOC simulations with MGXS generated by OpenMC throughout the following chapters.

As illustrated in the figures, the fission rate distributions are strongly dependent on the spatially heterogeneous features in each benchmark geometry. For example, the CRGTs provide additional moderation and increase the fission rates in nearby fuel pins. The presence of BPs reduces the neutron population and therefore the fission rates for the surrounding fuel pins, while increasing the variation between the minimum and maximum fission rates. The presence of a reflector with a mixture of vacuum and reflective BCs induces a tilt in the fission rates across the assemblies in the 2×2 colorset, with the maximum fission rates located in the interior fuel pins and the minimum fission rates occurring near the reflector due to neutron leakage. The fission rates for the quarter core BEAVRS model in Fig. 7-14 form a highly complex spatial distribution due to the convolution of many different interacting spatial self-shielding effects.

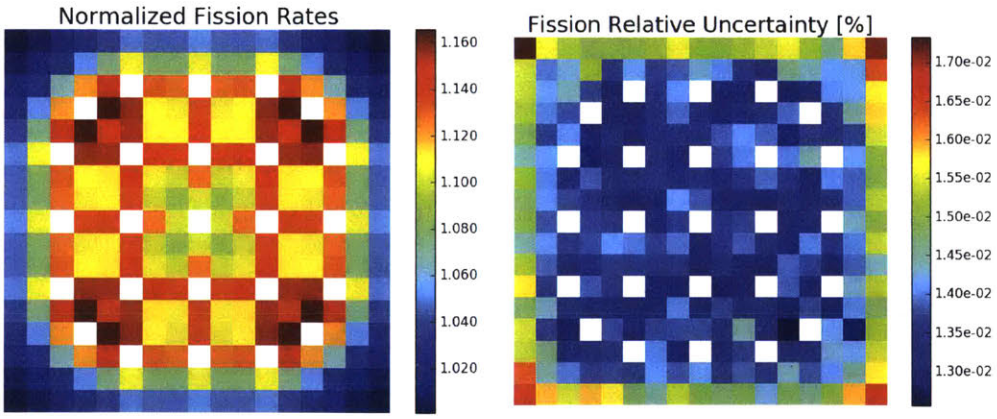
It should be noted that the fission rate distribution for the quarter core BEAVRS model is highly skewed due to the isotropic in lab scattering approximation. This can be seen from comparison of Fig. 7-14a with the true fission rate distribution calculated with normal anisotropic scattering in Fig. B-1a. The power distribution is highly sensitive to anisotropic scattering due to radial leakage out of the core, and is more peaked near the

corner reflectors when the isotropic approximation is made. Although isotropic scattering is unphysical, it allows direct comparisons of OpenMC and OpenMOC results in latter chapters to quantify the impact of various approaches to spatially homogenize MGXS without conflicting effects due to the isotropic scattering model used in OpenMOC.



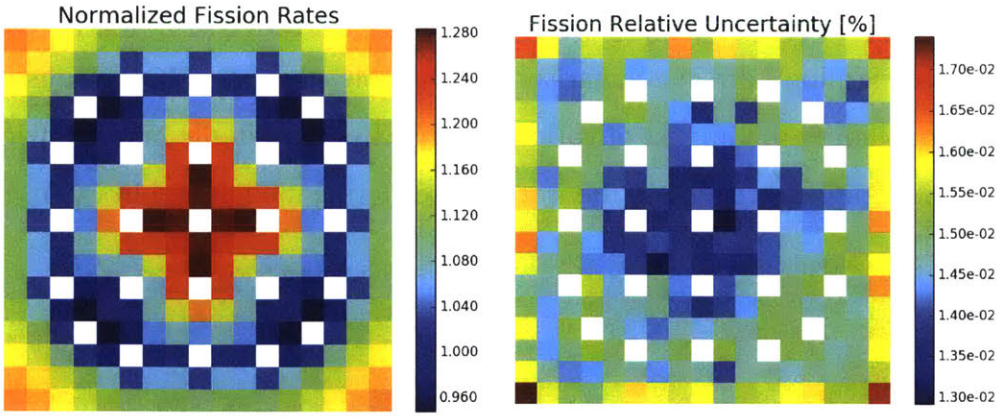
(a) (b)

Figure 7-9: Fission rates for a 1.6% enriched assembly.



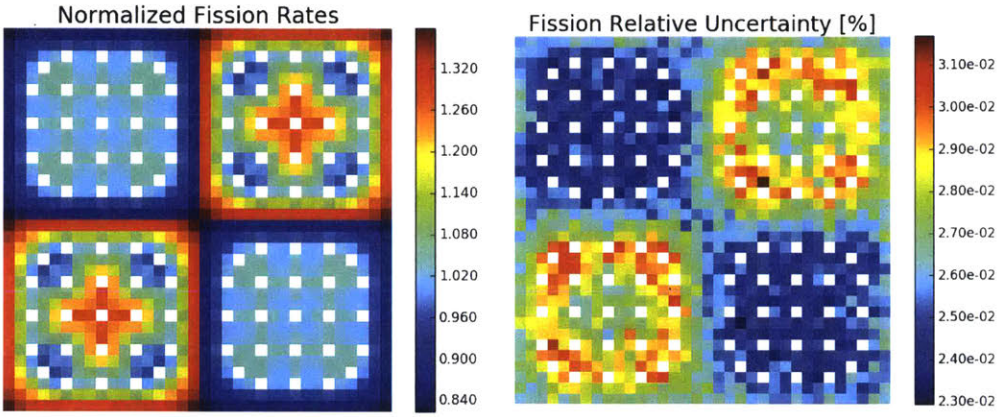
(a) (b)

Figure 7-10: Fission rates for a 3.1% enriched assembly.



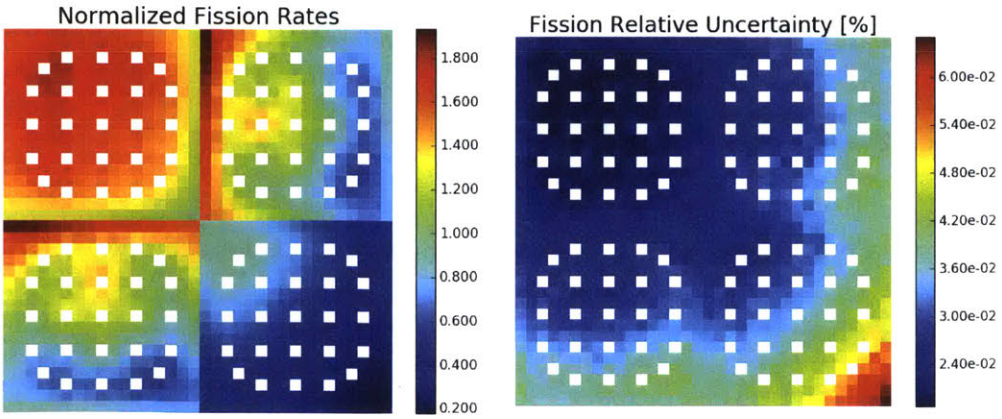
(a) (b)

Figure 7-11: Fission rates for a 3.1% enriched assembly with 20 BPs.



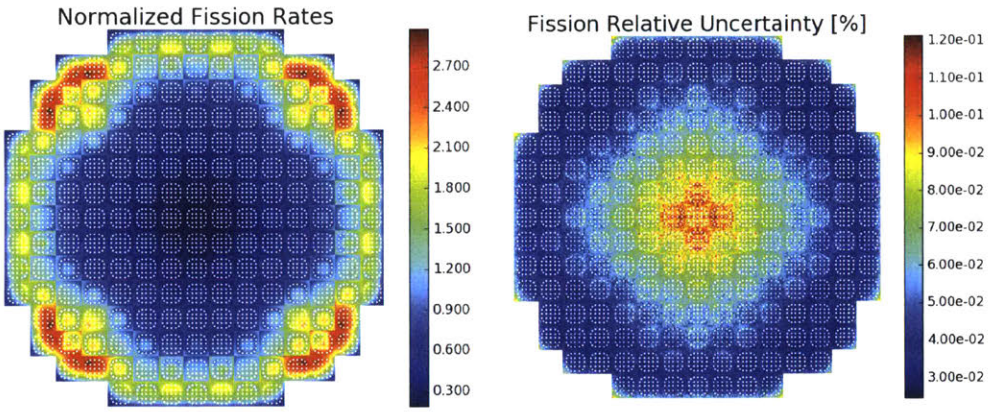
(a) (b)

Figure 7-12: Fission rates for a 2×2 colorset.



(a) (b)

Figure 7-13: Fission rates for a 2×2 colorset with a reflector.



(a) (b)

Figure 7-14: Fission rates for the 2D quarter core BEAVRS model.

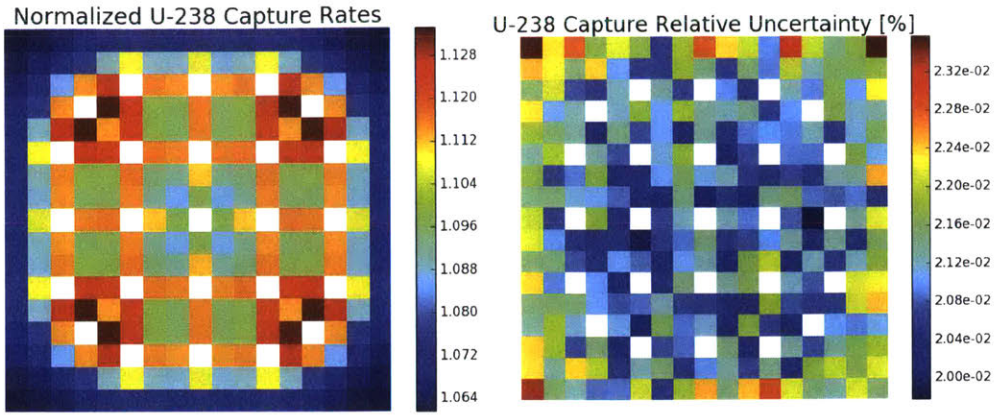
7.3.4 U-238 Capture Rate Spatial Distributions

The reference energy-integrated U-238 capture rate spatial distributions for each of the six benchmarks were computed using rectilinear, pin-wise tally meshes in OpenMC. The U-238 capture rates were volume-integrated across each fuel pin. The capture rates were normalized to the mean of all non-zero capture rates in each benchmark. The percent relative errors for the tallied capture rates was computed from the ratio of the sample standard deviation to the mean capture rate in each fuel pin. The capture rates for the quarter core BEAVRS model were appropriately tiled to display the equivalent distribution across the quadrant symmetric full core.

The U-238 capture rate spatial distributions and percent relative errors for each of the six benchmarks are presented as heatmaps in Figs. 7-15 to 7-20. The capture rates in the instrument tubes, CRGTs and BPs are all zero and are illustrated in white. The color bars for each of the heatmaps range from the minimum non-zero capture rate to the maximum capture rate for all fuel pins in each benchmark geometry. The maximum relative error at the final 1000^{th} batch is less than 0.5% for all of the benchmarks. These reference U-238 capture rate spatial distributions are used to validate the OpenMOC simulations with MGXS generated by from OpenMC throughout the following chapters.

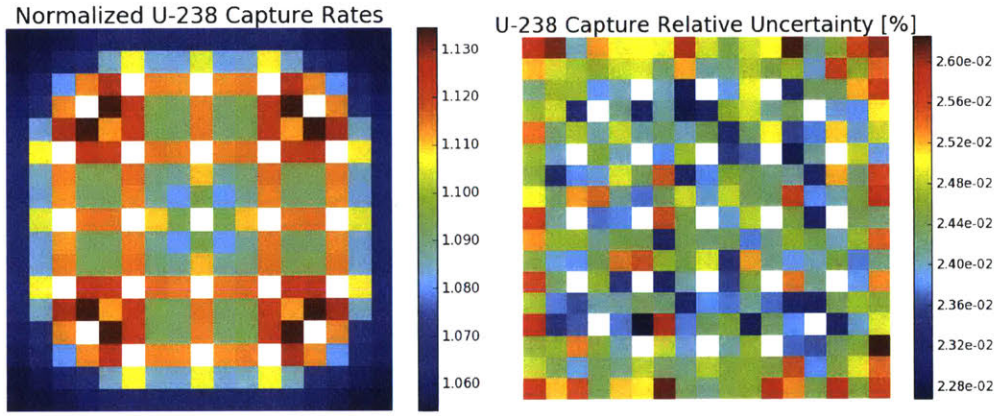
The impacts of CRGTs, BPs, reflectors and vacuum BCs on the U-238 capture rates are similar to those observed for the fission rates, though there are some noticeable differences. For example, the U-238 capture rates in the individual fuel assemblies appear to be more sensitive than the fission rates to the spatial self-shielding induced by moderation in CRGTs. In addition, the U-238 capture rates peak in the 1.6% enriched fuel assemblies in the 2×2 colorset without a reflector (Fig. 7-18b), while the fission rates peak in the 3.1% enriched fuel assemblies (Fig. 7-12b). The key reason for this is that there is a higher concentration of U-238 in the 1.6% enriched fuel than the 3.1% enriched fuel, which leads to a lower ratio of fission to U-238 capture rates. In addition, the U-238 capture rates in the reflected 2×2 benchmark (Fig. 7-19) are more smoothly varying at the inter-assembly interface than the fission rates (Fig. 7-13).

Similar to the fission rates, the U-238 capture rate distribution for the quarter core



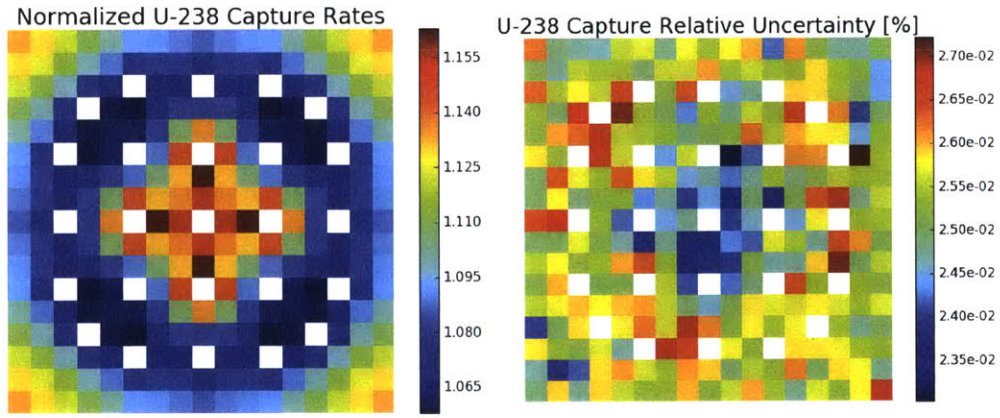
(a) (b)

Figure 7-15: U-238 capture rates for a 1.6% enriched assembly.



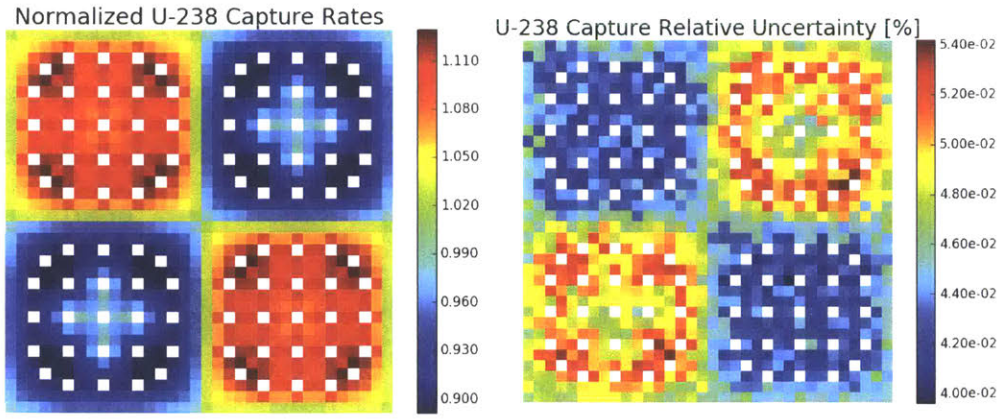
(a) (b)

Figure 7-16: U-238 capture rates for a 3.1% enriched assembly.



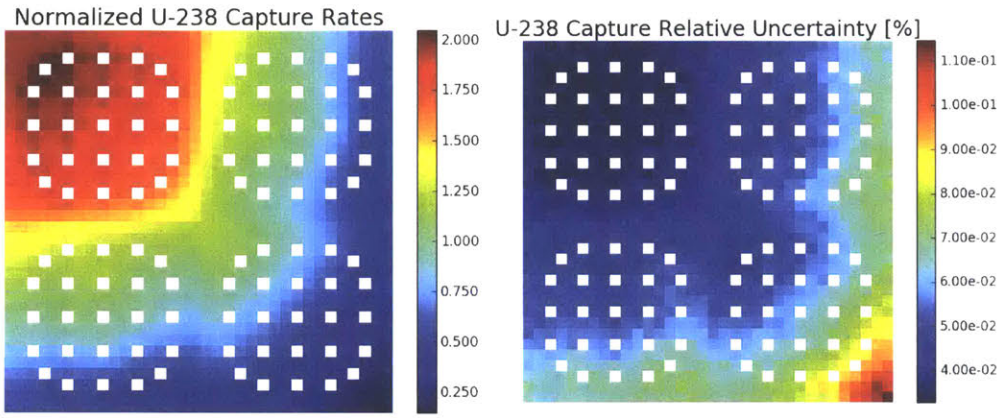
(a) (b)

Figure 7-17: U-238 capture rates for a 3.1% enriched assembly with 20 BPs.



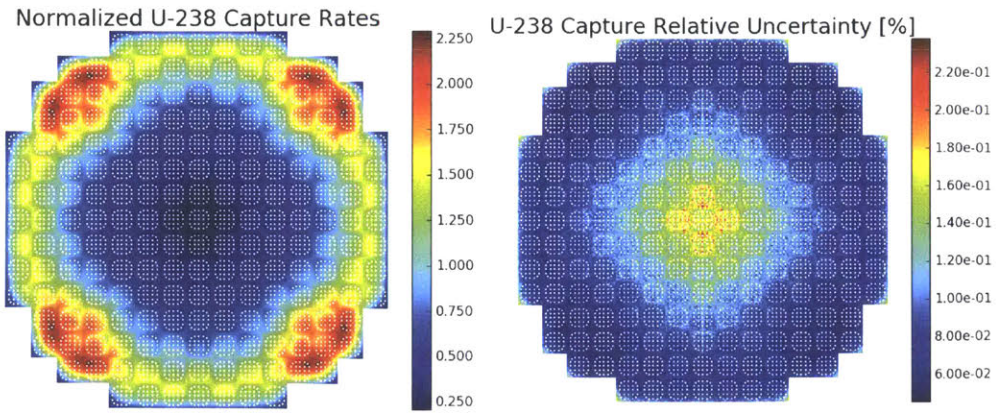
(a) (b)

Figure 7-18: U-238 capture rates for a 2×2 colorset.



(a) (b)

Figure 7-19: U-238 capture rates for a 2×2 colorset with a reflector.



(a) (b)

Figure 7-20: U-238 capture rates for the 2D quarter core BEAVRS model.

BEAVRS model is highly skewed due to the isotropic in lab scattering approximation. This can be seen from comparison of Fig. 7-20a with the true capture rate distribution calculated with normal anisotropic scattering in Fig. B-1b. Like fission, the U-238 capture distribution is highly sensitive to anisotropic scattering due to radial leakage out of the core, and is more peaked near the corner reflectors when the isotropic approximation is made. Although isotropic scattering is unphysical, it allows direct comparisons of OpenMC and OpenMOC results in latter chapters to quantify the impact of various approaches to spatially homogenize MGXS without conflicting effects due to the isotropic scattering model used in OpenMOC.

Highlights

- A series of six 2D heterogeneous benchmark models were derived from the full core BEAVRS model to explore spatial self-shielding effects on MGXS.
- The benchmarks include individual fuel assemblies with different CRGT and BP configurations, 2×2 fuel assembly colorsets with and without a water reflector, and the quarter core BEAVRS model.
- The Shannon entropy was computed to determine the number of inactive batches needed when modeling each benchmark with OpenMC.
- Reference results for the eigenvalues, pin-wise fission rates and pin-wise U-238 capture rates were computed using OpenMC.
- The benchmarks and reference results are used in the following chapters to validate the use of statistical clustering methods to capture spatial self-shielding effects in MGXS generated by OpenMC for OpenMOC.

Chapter 8

Quantification of Spatial Self-Shielding Effects

8.1 Overview

The preceding chapter introduced six heterogeneous 2D PWR benchmarks derived from the BEAVRS model, along with reference metrics tallied by OpenMC. This chapter applies multi-group transport calculations to model the same benchmarks in OpenMOC with MGXS generated by OpenMC. The objective is to identify the bias between OpenMC and OpenMOC for MGXS libraries which account for spatial self-shielding effects[38] to varying degrees. In particular, this chapter quantifies the difference in the approximation error between simulations in which the same MGXS are used in each unique fuel pin (*e.g.*, each fuel enrichment) and those in which unique MGXS are used in each and every pin. The former case does little if anything to model spatial self-shielding effects, whereas the latter case "fully" resolves these effects, albeit at the expense of very large MGXS libraries. This difference in approximation error motivates the development of a novel methodology in the following chapters which uses statistical clustering to capture spatial self-shielding effects in MGXS.

Three different schemes for spatial homogenization of pin-wise MGXS are introduced

[38]The effects of neighboring pins, burnable poisons, reflectors and the core baffle, barrel and vessel are all of interest in the context of spatial self-shielding in this and subsequent chapters.

in Sec. 8.2 and are referred to as *infinite*, *null* and *degenerate* spatial homogenization, respectively. The discretized models and runtime parameters used in the OpenMOC simulations are detailed in Sec. 8.3. The bias between the OpenMOC simulations and the reference OpenMC results – including eigenvalues, pin-wise fission rates and pin-wise U-238 capture rates – are presented in Sec. 8.4. The need for a new, more flexible and specialized approach to spatial homogenization which appropriately captures spatial self-shielding effects with minimal computational expense is discussed in Sec. 8.5.

8.2 Pin-wise Spatial Homogenization Schemes

This chapter employs three different spatial homogenization schemes to model spatial self-shielding effects in MGXS. Although all spatial zones may experience spatial self-shielding, this chapter only models the impact of spatial self-shielding on MGXS in fissile regions. The infinite, null and degenerate spatial homogenization schemes are introduced in Secs. 8.2.1 to 8.2.3. These schemes model spatial self-shielding for each fuel pin with increasing granularity and complexity. The total number of materials (*i.e.*, MGXS) used to model each benchmark with each homogenization scheme is given in Tab. 8.1. A fuel assembly, 2×2 colorset and part of the quarter core BEAVRS model are color-coded by material and illustrated in Fig. 8-1 for each homogenization scheme.

The `openmc.mgxs` module (see Sec. 4.2.4) was used to compute 70-group MGXS with OpenMC for each of the six heterogeneous benchmarks introduced in Chap. 7. The tallied MGXS data was condensed to coarse 2-group and 8-group structures with downstream data processing as necessary. The OpenMC simulations were performed with 1000 batches with 10^6 particle histories per batch for each benchmark. This was only one tenth of the 10^7 histories per batch used to tally the reference results in Chap. 7 for practical computational reasons[39]. Stationarity of the fission source was obtained with 200 inactive batches for the quarter core BEAVRS model, while 100 inactive batches were

[39]The total runtime consumed by OpenMC scales with the number of tallied quantities. The number of tallies used to compute MGXS was much larger than the three used to compute the reference solutions in Chap. 7. As a result, the simulation time per history was prohibitively slow to generate MGXS with the same number of histories as was used to compute the reference solution.

employed for the other five benchmarks (see Sec. 7.3.1). OpenMC's "iso-in-lab" feature (see Sec. 4.2.3) was employed to enable consistent comparisons between OpenMC's reference results and OpenMOC's calculations with an isotropic in lab scattering source.

Table 8.1: Number of materials modeled with unique MGXS in each heterogeneous benchmark for each spatial homogenization scheme.

Benchmark	# Fuel Materials		
	Infinite	Null	Degenerate
1.6% Assm	1	1	264
3.1% Assm	1	1	264
3.1% Assm w/ 20 BPs	1	1	264
2×2 Colorset	2	2	1,056
2×2 Colorset w/ Reflector	2	2	1,056
BEAVRS Quarter Core	3	3	12,993

8.2.1 Infinite Lattice Homogenization

The *infinite* spatial homogenization scheme is most reminiscent of the traditional multi-level schemes used to generate MGXS (see Sec. 2.5.2), and is the simplest approach to model spatial self-shielding effects considered by this thesis. The infinite scheme employs multiple OpenMC simulations to compute MGXS for each heterogeneous benchmark. The MGXS for each type of fuel (*e.g.*, enrichment) are generated by OpenMC simulations of each fuel pin type in an infinite, repeating array[40]. The MGXS for all other materials – including borated water, zircaloy, helium, etc. – are generated from OpenMC simulations of each heterogeneous benchmark where the reaction rates and fluxes are averaged across each geometry.

The infinite scheme is designed to quantify the impact of using the "true" Monte Carlo flux from an infinite lattice calculation, rather than the "true" MC flux from the true heterogeneous geometry, to collapse MGXS in fissile zones. The scheme employs a single MGXS in each instance of a material zone, such as a fuel pin replicated many

[40] An infinite, repeating array of fuel pins is modeled by a single fuel pin with reflective boundary conditions.

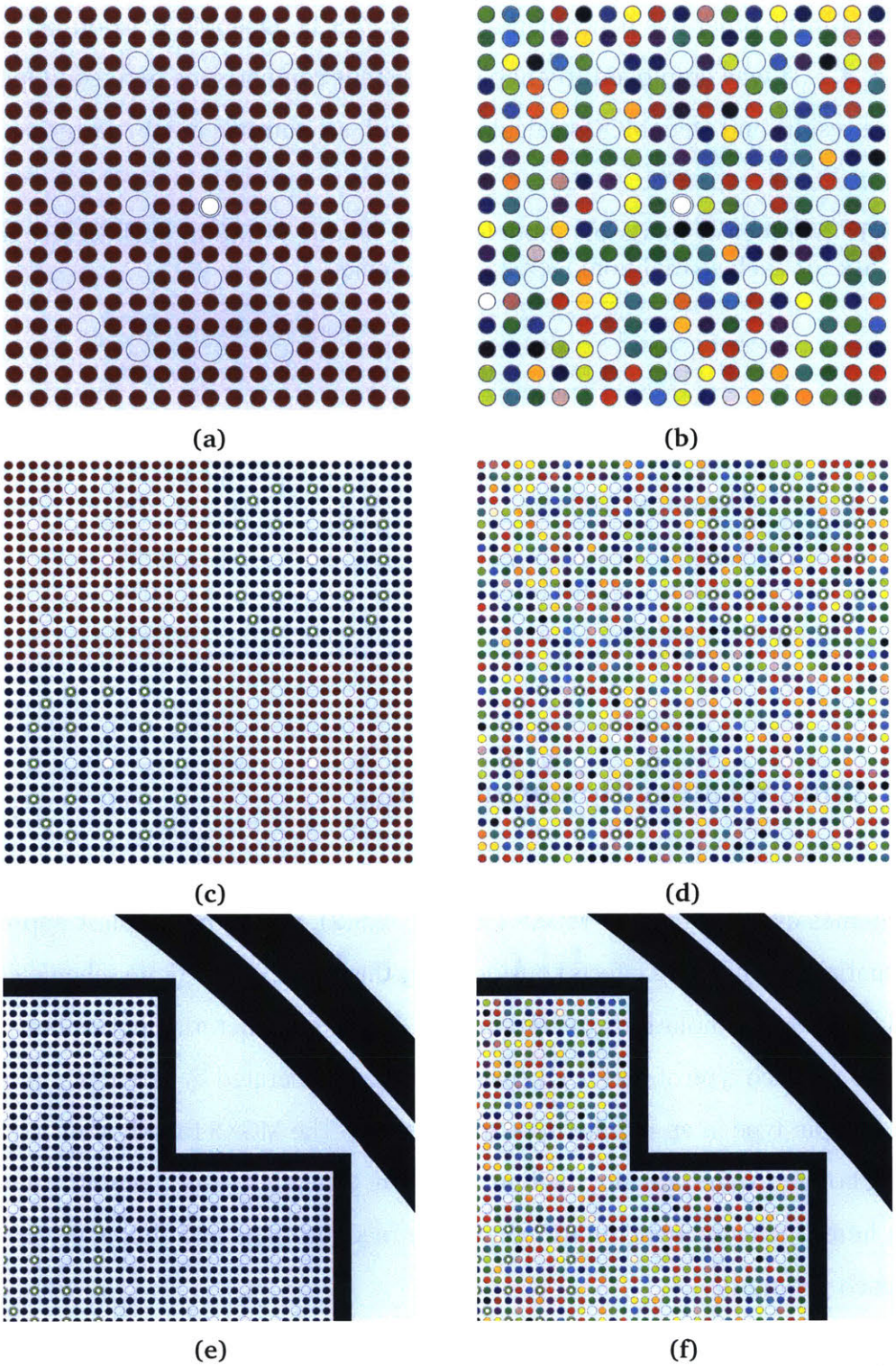


Figure 8-1: OpenMOC materials for a single fuel assembly, a 2×2 colorset and part of the 2D quarter core BEAVRS model. The materials for the infinite and null schemes are depicted in (a), (c) and (e), and for the degenerate scheme in (b), (d) and (f), respectively. Each uniquely colored material represents a unique set of MGXS.

times throughout a benchmark geometry. The MGXS for each fissile zone is generated from an infinite lattice calculation with OpenMC. The MGXS for all non-fissile zones are generated using the "true" flux distribution in space and energy for each of the heterogeneous benchmarks. The scheme does not account for spatial self-shielding effects experienced by different non-fissile spatial zones filled by the same material, and instead averages these effects across the entire geometry for each material.

8.2.2 Null Homogenization

The *null* spatial homogenization scheme builds upon the infinite scheme, but uses the "true" heterogeneous flux to collapse MGXS for fissile as well as non-fissile materials. The null scheme eliminates the infinite lattice calculation to generate MGXS for fissile zones, and instead uses a single Monte Carlo calculation of the complete heterogeneous geometry to generate MGXS for *all* materials. In this way, the null scheme fully abandons the multi-level approach used by the infinite scheme and most traditional approaches to generate MGXS. Unlike the infinite scheme, the spatially self-shielded flux is used to collapse the cross sections in the fuel. However, the null scheme does not account for spatial self-shielding effects experienced by different fuel pins filled by the same type of fuel, and instead averages these effects across the entire geometry. As with the infinite scheme, a single MGXS is employed in each instance of a material zone, such as a fuel pin replicated many times throughout a benchmark geometry.

8.2.3 Degenerate Homogenization

Unlike the infinite and null spatial homogenization schemes, the *degenerate* scheme accounts for the different spatial self-shielding effects experienced by each instance of each fuel pin throughout a heterogeneous geometry. Like the null scheme, a single MC calculation of the complete heterogeneous geometry is used to generate MGXS for all materials. Unlike the null scheme, the MGXS are tallied separately for each instance of fissile material zones. For example, if a heterogeneous benchmark includes N fuel pins, then N collections of MGXS are separately tabulated for each fuel pin instance. The

degenerate scheme tallies different MGXS even if the isotopic compositions in the fuel pin instances are identical (*e.g.*, fresh fuel at the beginning of life) since each instance may experience different spatial self-shielding effects and hence have different MGXS.

Multi-group transport calculations with MGXS generated using infinite/null and degenerate schemes may be compared to quantify the impact of modeling spatial self-shielding effects in MGXS for fissile zones in heterogeneous geometries. The degenerate scheme applies the finest granularity to pin-wise spatial homogenization of any of the schemes considered in this thesis since it best captures different spatial self-shielding effects in each fuel pin. As a result, the degenerate scheme is used to benchmark the efficacy of the new methodology for spatial homogenization based on statistical clustering developed in the following chapters. Like both the infinite and null schemes, the spatial self-shielding effects experienced by different non-fissile spatial zones are averaged across the entire geometry for each non-fissile material.

The degenerate scheme generates MGXS for each fuel pin instance using OpenMC's distributed cell tallies (see Sec. 4.2.2). The OpenCG region differentiation algorithm (see Sec. 4.4.3) is used to build a new OpenMOC geometry with unique cells and materials for each fuel pin. The MGXS are appropriately selected from OpenMC's distributed cell tallies to populate the MGXS in the OpenMOC materials.

The infinite, null and degenerate spatial homogenization schemes are used to quantify approximation errors made when neglecting spatial self-shielding due to neighboring pins, reflectors, etc. in heterogeneous PWR benchmarks.

8.3 OpenMOC Runtime Parameters

The infinite, null and spatial homogenization schemes were used to prepare MGXS libraries for OpenMOC simulations of each of the six heterogeneous PWR benchmarks introduced in Chap. 7. This section briefly outlines the energy group structures and angular, spatial and CMFD meshes used in the OpenMOC simulations. The total number of flat source regions, and MOC tracks and segments are summarized for each benchmark

in Tab. 8.2. It was crucial to use an adequate discretization to accurately compare simulation results between the three spatial homogenization schemes, as well as to the reference OpenMC results.

Table 8.2: The number of MOC FSRs, tracks and segments modeled in each benchmark.

Benchmark	# FSRs	# Tracks	# Segments
1.6% Assm	28,376	34,976	7,945,952
3.1% Assm	28,376	34,976	7,945,952
3.1% Assm w/ 20 BPs	29,496	34,976	8,110,192
2×2 Colorset	115,744	69,892	32,097,936
2×2 Colorset w/ Reflector	203,220	104,788	51,122,228
BEAVRS Quarter Core	1,718,368	201,620	38,3785,515

Each simulation was converged to 10^{-5} on the root mean square of the energy-integrated fission source in each flat source region (FSR). It should be noted that a convergence criterion of 10^{-7} was employed for the OpenMOC simulations of simple benchmarks without CMFD acceleration (see Sec. 4.3.5) in Chap. 5, but a looser convergence criterion of 10^{-5} may be used for calculations with CMFD[41].

8.3.1 Energy Group Structures

Each assembly, colorset and quarter core benchmark was modeled with MGXS in 2, 8 and 70 energy groups. The energy group structures were the same as those used in Chaps. 5 and 6 and are tabulated in App. A. The MOC energy group structures were collapsed onto coarser group structures for CMFD acceleration. The mapping from MOC to CMFD group structures is listed in Tab. 8.3. The coarse CMFD structures were derived to best approximate equal lethargy spacing between coarse CMFD groups. The coarse CMFD structures significantly improved the speed of the 70-group OpenMOC calculations.

[41] The convergence criterion measures the residual between successive iterations, but should actually measure convergence to the asymptotic solution, which depends on both the residual between successive iterations and the dominance ratio. CMFD significantly reduces the residual between successive iterations such that sufficient convergence to the asymptotic solution is achieved at a much lower successive iteration residual.

Table 8.3: The coarse CMFD group structures for each of the fine MOC group structures.

# MOC Groups	# CMFD Groups	CMFD Fine-to-Coarse Group Mapping
2	2	[1], [2]
8	4	[1–2], [3], [4–5], [6–8]
70	14	[1–2], [3–6], [7–9], [10–12], [13–16], [17–19], [20–21], [22–24], [25–27], [28–33], [34–53], [54–61], [62–68], [69–70]

8.3.2 Angular Discretization

The OpenMOC simulations of each of the six heterogeneous benchmarks used the same MOC angular discretization with 128 azimuthal angles and 0.05 cm track spacing. These parameters were selected since they were previously demonstrated to converge the solution for simple heterogeneous PWR benchmark models [65]. Although the eigenvalues in Chap. 5 were only shown to converge for a 1D slab and 2D fuel pin with 0.01 cm track spacing, a coarser spacing of 0.05 cm was employed for the heterogeneous benchmarks for practical reasons[42]. The total number of tracks and segments resulting from the selected angular discretization are itemized in Tab. 8.2.

8.3.3 FSR Discretization

Flat source region spatial discretization meshes were applied to each of the six heterogeneous benchmarks in the OpenMOC simulations. The total number of FSRs is itemized in Tab. 8.2 for each benchmark analyzed in this thesis. The FSR meshes applied to the fuel pin, control rod guide tube, instrument tube and burnable poison pin cells are shown in Fig. 8-2. As shown in the figures, eight equal angle subdivisions were used in all material zones. The UO_2 fuel was further subdivided into five equal volume radial rings, while ten radial rings were employed in the water-filled CRGTs and instrument tubes. The borosilicate glass and borated water material zones filling the BPs were each discretized into five equal volume radial rings. Finally, five equally spaced rings were

[42]The computational expense of solving the MOC equations scales linearly with the number of track segments. In addition, OpenMOC's memory footprint is dominated by track segments which can make some problems intractable except on high memory nodes. Finally, the time spent ray tracing across a combinatorial geometry is non-negligible if not prohibitive for fine track spacings in large geometries.

used in the moderator zones surrounding each pin.

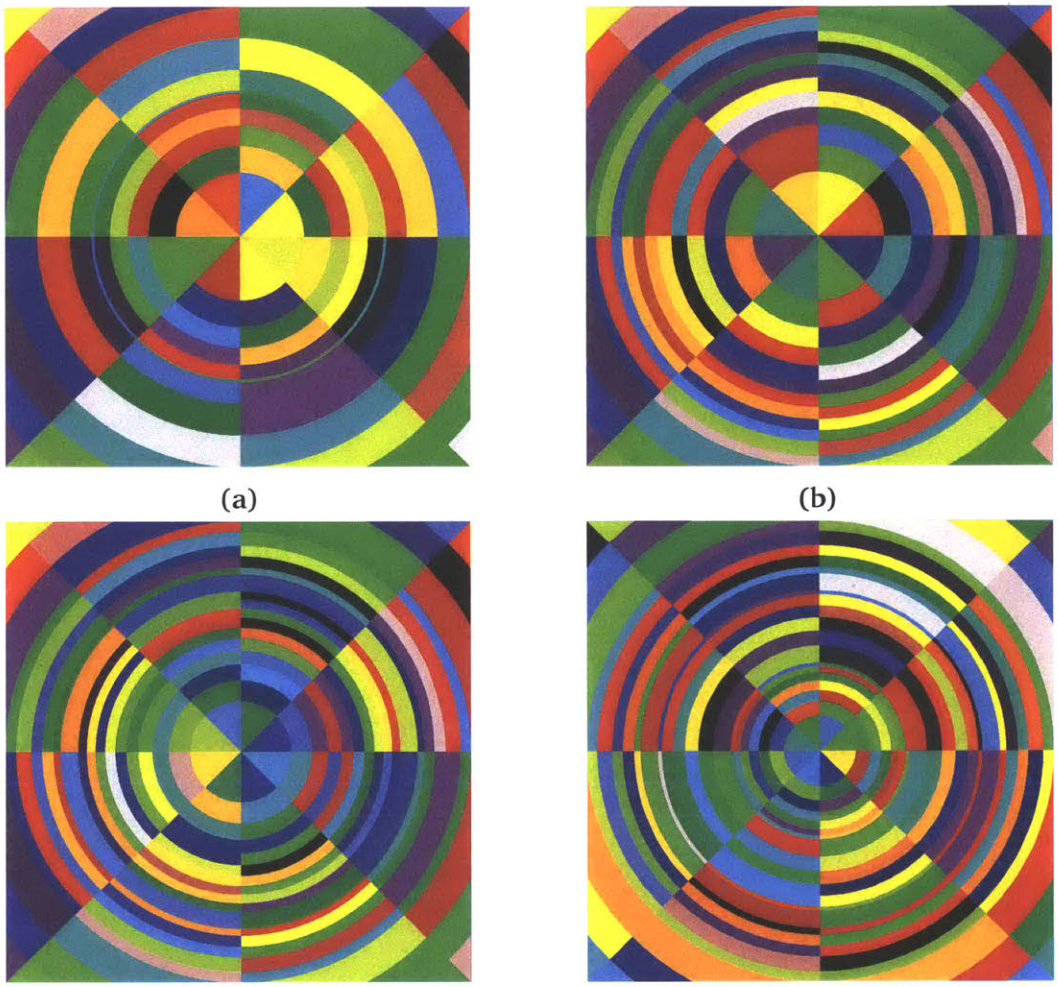


Figure 8-2: The FSR spatial mesh used for fuel pins (a), control rod guide tubes (b), instrument tubes (c) and burnable poisons (d). Diagrams of each pin type color-coded by material are shown in Fig. 7-1.

The spatially discretized pin cells were used in each of the six heterogeneous benchmarks. The FSR discretizations for an individual fuel assembly, 2×2 colorset with a water reflector, and the upper right quadrant of the BEAVRS core are depicted in Fig. 8-3. The water reflector in the 2×2 colorset was discretized using a mesh shown to converge the solution for the similarly designed C5G7 benchmark [81] in [65]. In particular, the 13.85824 cm (equivalent to 11 pin cells) of water reflector nearest the fuel assemblies was discretized in a 0.125984 cm × 0.125984 cm rectilinear mesh, equivalent to a 10×10 mesh in each pin. The 7.55904 cm (equivalent to 6 pin cells) of water reflector furthest

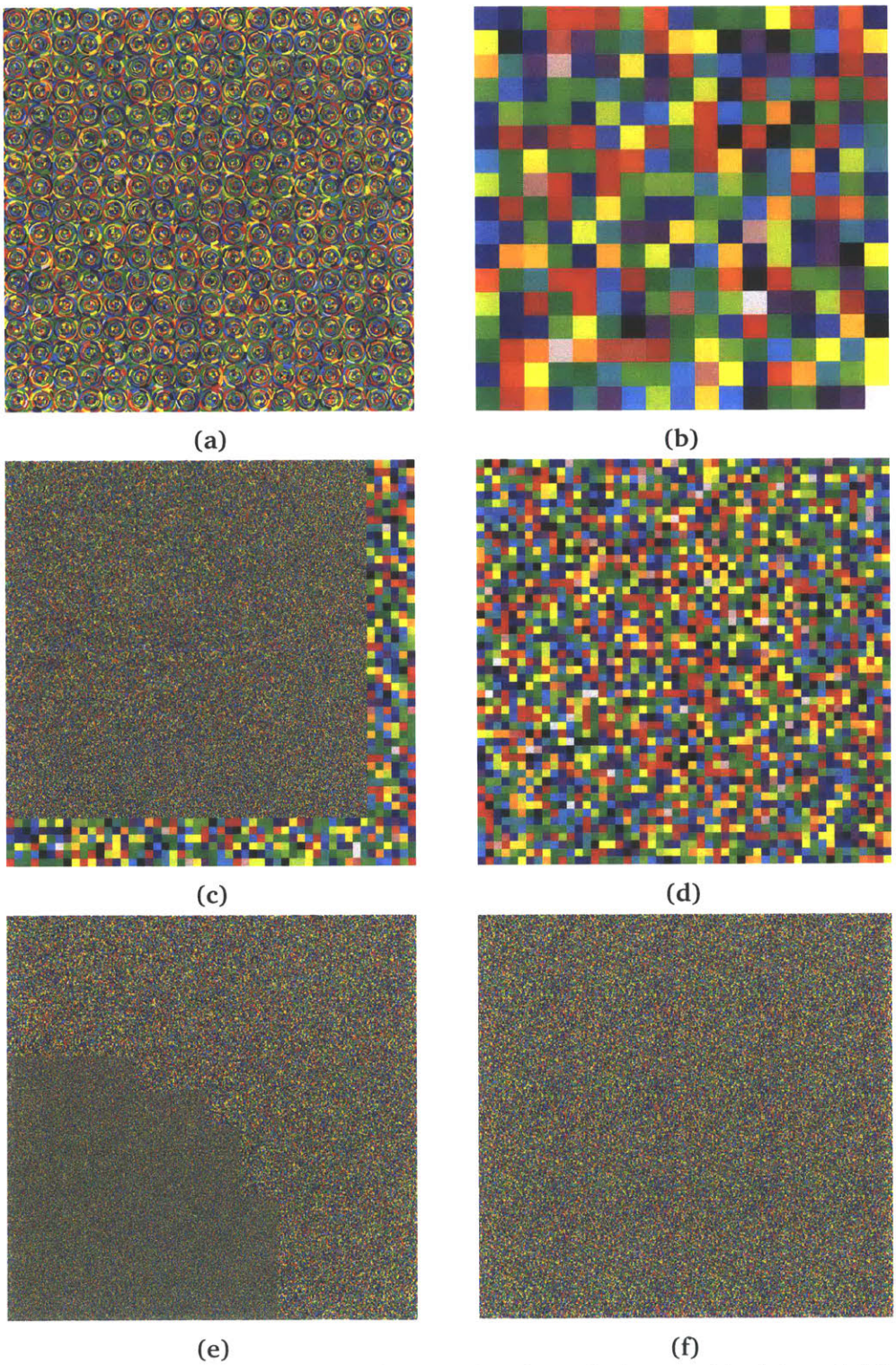


Figure 8-3: FSR (left) and CMFD (right) meshes for a fuel assembly (a) and (b), a 2×2 colorset with reflector (c) and (d), and quarter core BEAVRS benchmark (e) and (f).

from the fuel assemblies was discretized in a 1.25984 cm × 1.25984 cm pin-wise mesh.

Each type of pin cell was discretized in the same way for the quarter core BEAVRS model. A 0.31496 cm × 0.31496 cm mesh (equivalent to a 4×4 mesh in each pin) was applied in the water reflector. The mesh in the reflector was coarser than the mesh used in the reflector for the 2×2 colorset since a finer mesh was not computationally practical for the quarter core BEAVRS model, nor needed due to th presence of the steel baffle. A quarter pin-wise CMFD mesh (*i.e.*, 0.62992 cm × 0.62992 cm) was employed for the BEAVRS model, which discretized the core barrel and vessel into a corresponding FSR mesh.

8.4 Analysis of Multi-Group Results

Each of the six benchmarks was modeled with OpenMOC using MGXS generated by the infinite, null and degenerate spatial homogenization schemes. The eigenvalues and pin-wise fission and U-238 capture rates computed by OpenMOC are compared to the reference OpenMC solutions in Secs. 8.4.1, 8.4.2 and 8.4.3, respectively.

8.4.1 Eigenvalues

The OpenMOC eigenvalues were compared to the reference OpenMC eigenvalues from Tab. 7.1. The eigenvalue bias $\Delta\rho$ was computed from Eqn. 5.1 in units of pcm. The bias is listed for each benchmark, energy group structure and spatial homogenization scheme in Tab. 8.4. It should be recalled that isotropic in lab scattering is used by OpenMC to compute both the reference solution and the MGXS. If anisotropic scattering were employed in OpenMC, one would expect quite different biases without a robust implementation of a higher order scattering kernel in OpenMOC.

As expected, the eigenvalue bias is highly dependent on energy group structure. In general, a positive bias is exhibited for 2 groups with an increasingly negative trend with more groups. The bias is remarkably consistent between -100 and -250 pcm for 70 groups for all of the benchmarks. This slightly negative bias is reminiscent of the -200 pcm bias observed for the 1D slab and 2D pin cell PWR benchmarks when modeled

with 70 groups in Chap. 5. The introduction of a reflector region in the 2×2 colorset increases the bias to over 1700 pcm for all homogenization schemes for 2 groups, but the bias is greatly reduced when modeled with more groups.

Table 8.4: OpenMOC eigenvalue bias $\Delta\rho$ for heterogeneous benchmarks with varying spatial homogenization schemes and energy group structures.

Benchmark	MGXS Scheme	$\Delta\rho$ [pcm]		
		2-Group	8-Group	70-Group
1.6% Assm	Infinite	-132	-68	31
	Null	60	-72	-161
	Degenerate	62	-72	-161
3.1% Assm	Infinite	-188	-98	29
	Null	95	-79	-202
	Degenerate	98	-80	-202
3.1% Assm w/ 20 BPs	Infinite	392	20	-76
	Null	-136	-163	-252
	Degenerate	-160	-161	-248
2×2 Colorset	Infinite	267	-149	-16
	Null	27	-93	-196
	Degenerate	11	-93	-193
2×2 Colorset w/ Reflector	Infinite	2103	267	46
	Null	1818	478	-142
	Degenerate	1765	487	-132
BEAVRS Full Core	Infinite	2000	296	103
	Null	2181	406	-128
	Degenerate	2179	408	-125

The eigenvalue bias is also dependent on the spatial homogenization scheme used to compute MGXS in the fuel. It should first be noted that the eigenvalues are consistent for the null and degenerate schemes to within 10 pcm with 8 or more groups for all benchmarks. This is expected since the MGXS for each scheme is homogenized from the same flux and should preserve globally-integrated reaction rates. In contrast, the infinite scheme's eigenvalues differ by up to 300 pcm from the null/degenerate schemes, with no marked trend across benchmarks and energy group structures.

A consistent eigenvalue bias between OpenMC and OpenMOC of less than 250 pcm is observed for all benchmarks and homogenization schemes with 70 groups. The eigenvalues for the null and degenerate schemes are consistent to within 10 pcm for eight or more groups due to reaction rate preservation.

8.4.2 Fission Rate Distributions

The OpenMOC energy-integrated pin-wise fission rates were compared to the reference OpenMC fission rates from Figs. 7-9 to 7-14. The percent relative errors for each pin's fission rates were computed and the maximum and mean errors are listed for each benchmark, energy group structure and spatial homogenization scheme in Tabs. 8.5 and 8.6, respectively. In particular, the maximum errors are the maximum of the absolute values of the errors along with the appropriate sign, while the mean errors are the averages of the absolute error magnitudes.

As was the case for the eigenvalues, the fission rate errors are highly dependent on energy group structure. As expected, the maximum and mean errors are substantially reduced with finer energy group structures for all benchmarks and spatial homogenization schemes. The maximum fission rates are 2 – 11% in 2 groups for the individual fuel assembly and 2×2 colorset benchmarks, respectively, but decrease to less than 1% when modeled with 70 groups. The mean fission rate errors likewise decrease with finer energy group structures, and are less than 0.2% in magnitude for the assembly and colorset benchmarks for all homogenization schemes with 70 groups.

The fission rate errors are somewhat dependent on the spatial homogenization scheme used to compute MGXS in the fuel. In particular, the degenerate scheme produces slightly smaller maximum and mean errors than the null and infinite schemes. The error reduction is largest in 2 groups, and is less significant for more groups. The null and infinite schemes do not exhibit a systematic difference in their fission rate errors.

The spatial distributions of fission rate errors are plotted as heatmaps for each benchmark in Figs. 8-4 to 8-9. These figures illustrate the fission rate errors for 8 and

Table 8.5: OpenMOC maximum fission rate percent relative errors for heterogeneous benchmarks with varying spatial homogenization schemes and energy group structures.

Benchmark	MGXS Scheme	Max Error [%]		
		2-Group	8-Group	70-Group
1.6% Assm	Infinite	2.387	0.643	0.375
	Null	2.379	0.638	0.380
	Degenerate	1.903	0.726	0.315
3.1% Assm	Infinite	2.779	0.729	0.433
	Null	2.748	0.719	0.437
	Degenerate	2.151	0.832	0.372
3.1% Assm w/ 20 BPs	Infinite	-3.090	-0.707	0.372
	Null	-3.139	-0.701	0.380
	Degenerate	1.915	-0.689	0.331
2×2 Colorset	Infinite	-5.964	-1.425	0.418
	Null	-6.059	-1.428	0.463
	Degenerate	-5.339	-1.442	0.427
2×2 Colorset w/ Reflector	Infinite	11.024	2.773	0.670
	Null	-16.330	-2.855	0.764
	Degenerate	-11.158	-2.821	0.602
BEAVRS Quarter Core	Infinite	-79.994	-31.934	2.103
	Null	-85.893	-32.357	1.829
	Degenerate	-85.596	-32.382	1.728

70 energy group structures for the null and degenerate schemes; the reaction rate errors for 2, 8, and 70 groups for all three homogenization schemes are compared in Figs. C-1 to C-6 in App. C.1. The heatmaps illustrate systematic trends in the pin-wise fission errors which correlate with spatial heterogeneities in each benchmark. In particular, the fission rates are generally underpredicted for pins near CRGTs, but overpredicted for pins near BPs and in pins removed from CRGTs, such as those in the corners of each fuel assembly. The effects due to heterogeneities are further enhanced when differentiating pins which are facially adjacent to CRGTs, facially and corner adjacent to two CRGTs, etc. In addition, the error magnitudes are large for pins along the inter-assembly and assembly-reflector interfaces for the 2×2 colorset benchmarks. For the PWR benchmarks modeled here, the moderation provided by neighboring CRGTs and reflectors softens

Table 8.6: OpenMOC mean absolute fission rate percent relative errors for heterogeneous benchmarks with varying spatial homogenization schemes and energy group structures.

Benchmark	MGXS Scheme	Mean Error [%]		
		2-Group	8-Group	70-Group
1.6% Assm	Infinite	0.951	0.231	0.073
	Null	0.943	0.229	0.074
	Degenerate	0.687	0.240	0.079
3.1% Assm	Infinite	1.177	0.285	0.080
	Null	1.155	0.279	0.081
	Degenerate	0.808	0.290	0.087
3.1% Assm w/ 20 BPs	Infinite	0.927	0.194	0.095
	Null	0.937	0.193	0.098
	Degenerate	0.692	0.213	0.086
2×2 Colorset	Infinite	3.152	0.690	0.108
	Null	3.132	0.690	0.120
	Degenerate	2.918	0.702	0.120
2×2 Colorset w/ Reflector	Infinite	4.964	1.029	0.147
	Null	5.471	1.080	0.178
	Degenerate	4.731	1.065	0.138
BEAVRS Quarter Core	Infinite	34.831	10.391	0.492
	Null	39.355	10.541	0.323
	Degenerate	39.520	10.474	0.336

the flux for nearby fuel pins and should be modeled when collapsing pin-wise MGXS for high-fidelity multi-group transport calculations.

The use of degenerate spatial homogenization "smooths" the spatial distribution of pin-wise fission rate errors for 2-group MGXS. However, the fission rates computed with MGXS generated from the infinite, null and degenerate homogenization schemes are very similar with fine 70 group structures.

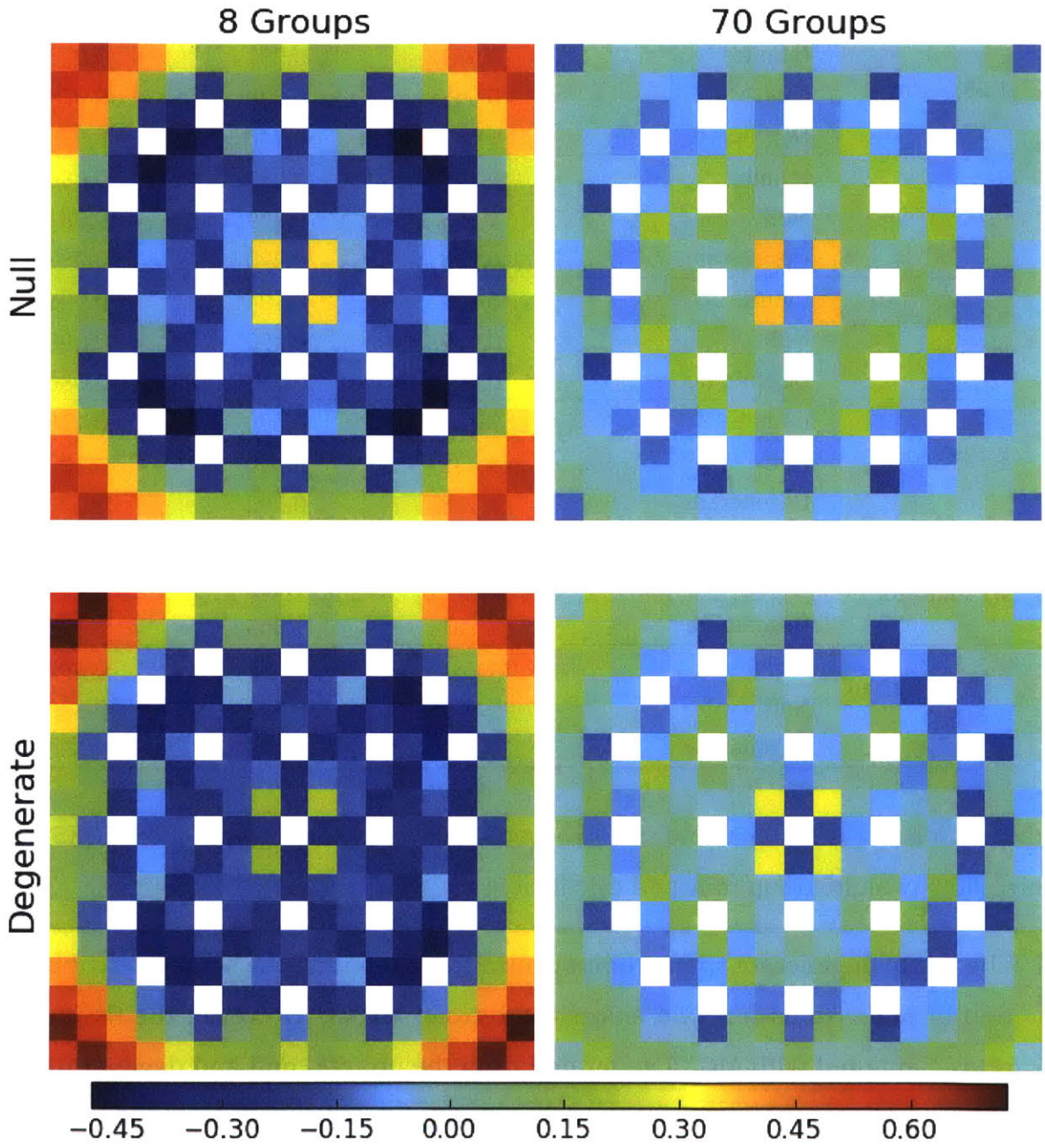


Figure 8-4: Fission rate percent relative errors for a 1.6% enriched assembly corresponding to the reference in Fig. 7-9.

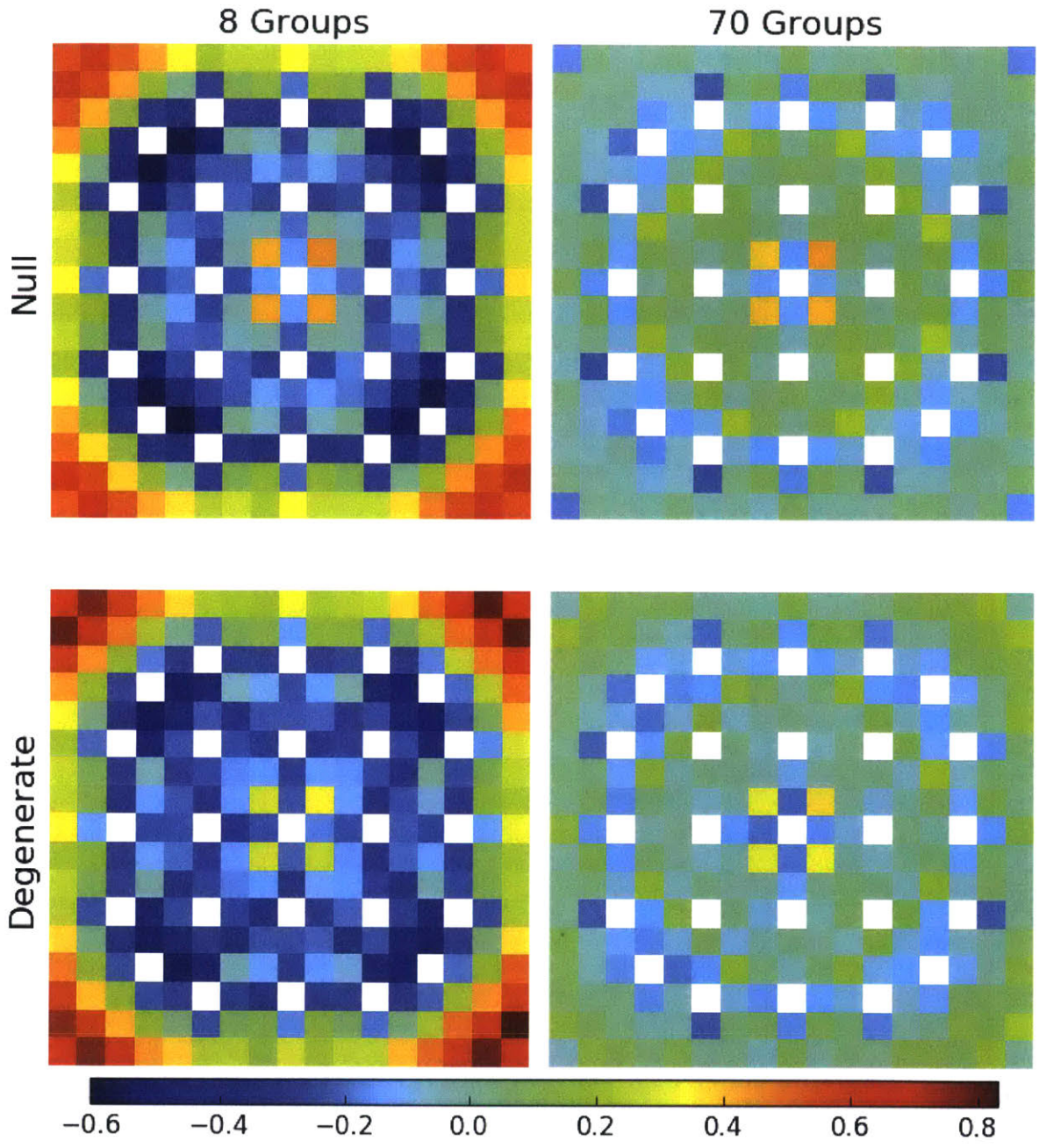


Figure 8-5: Fission rate percent relative errors for a 3.1% enriched assembly corresponding to the reference in Fig. 7-10.

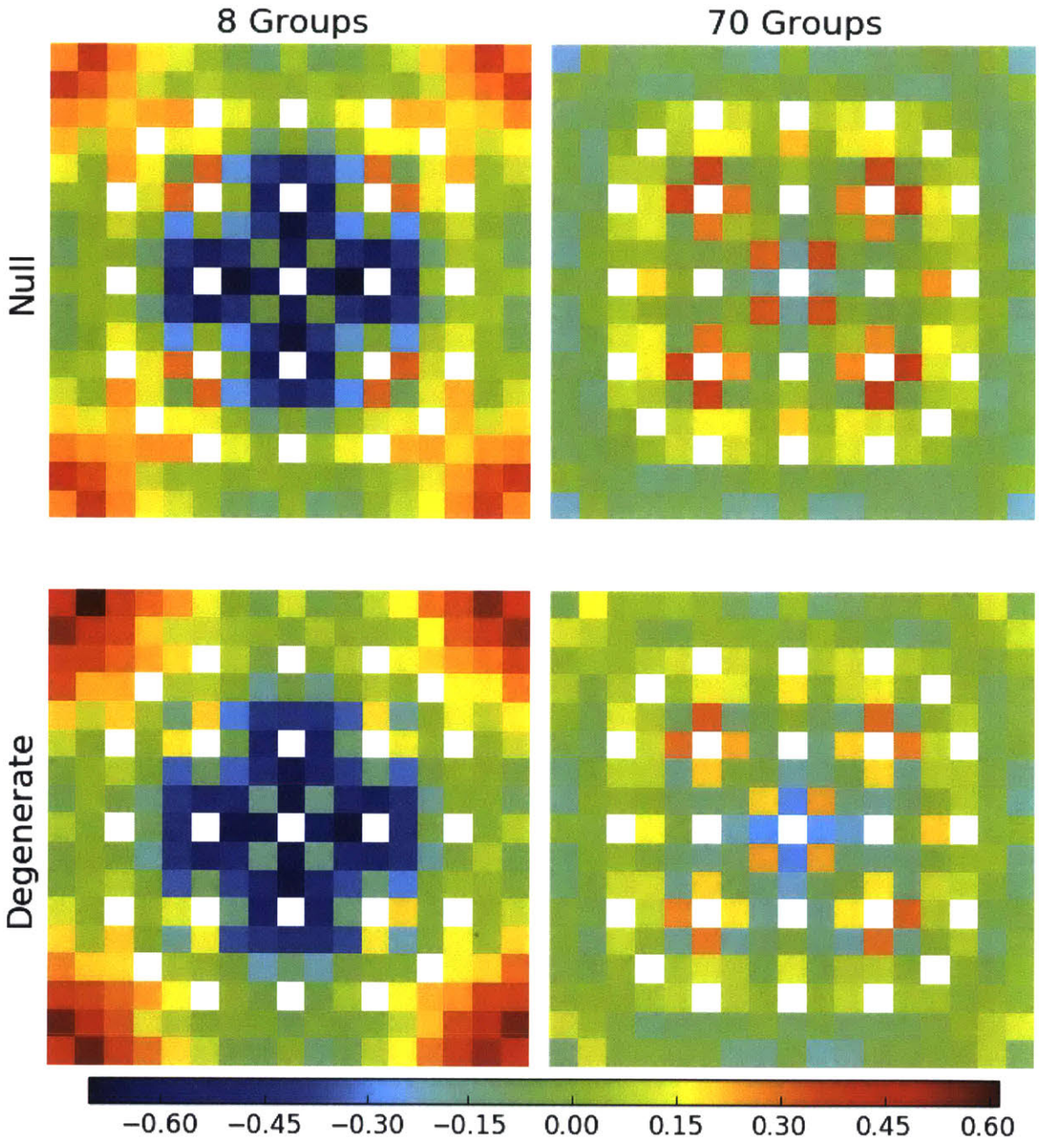


Figure 8-6: Fission rate percent relative errors for a 3.1% enriched assembly with 20 BPs corresponding to the reference in Fig. 7-11.

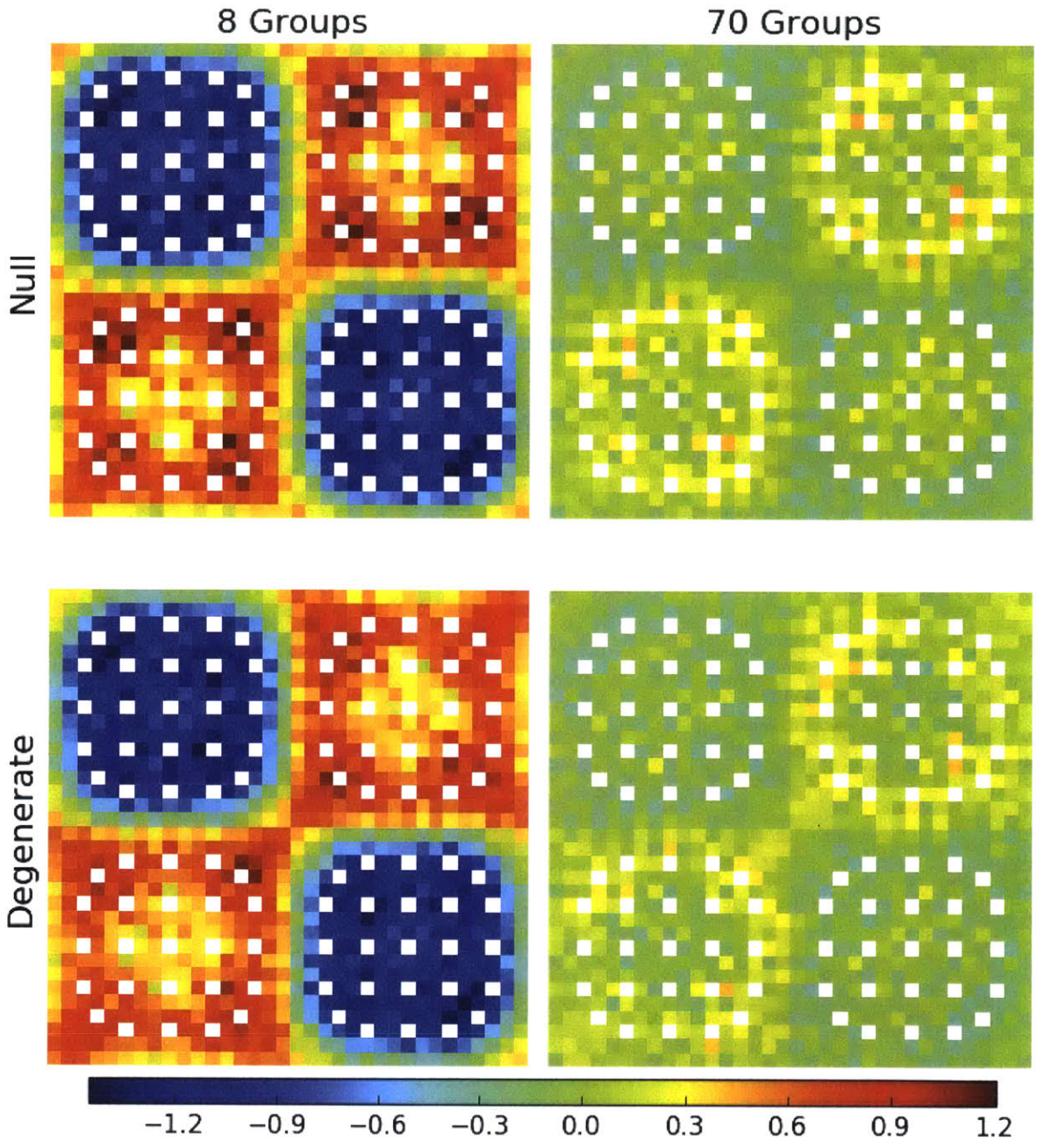


Figure 8-7: Fission rate percent relative errors for a 2×2 colorset corresponding to the reference in Fig. 7-12.

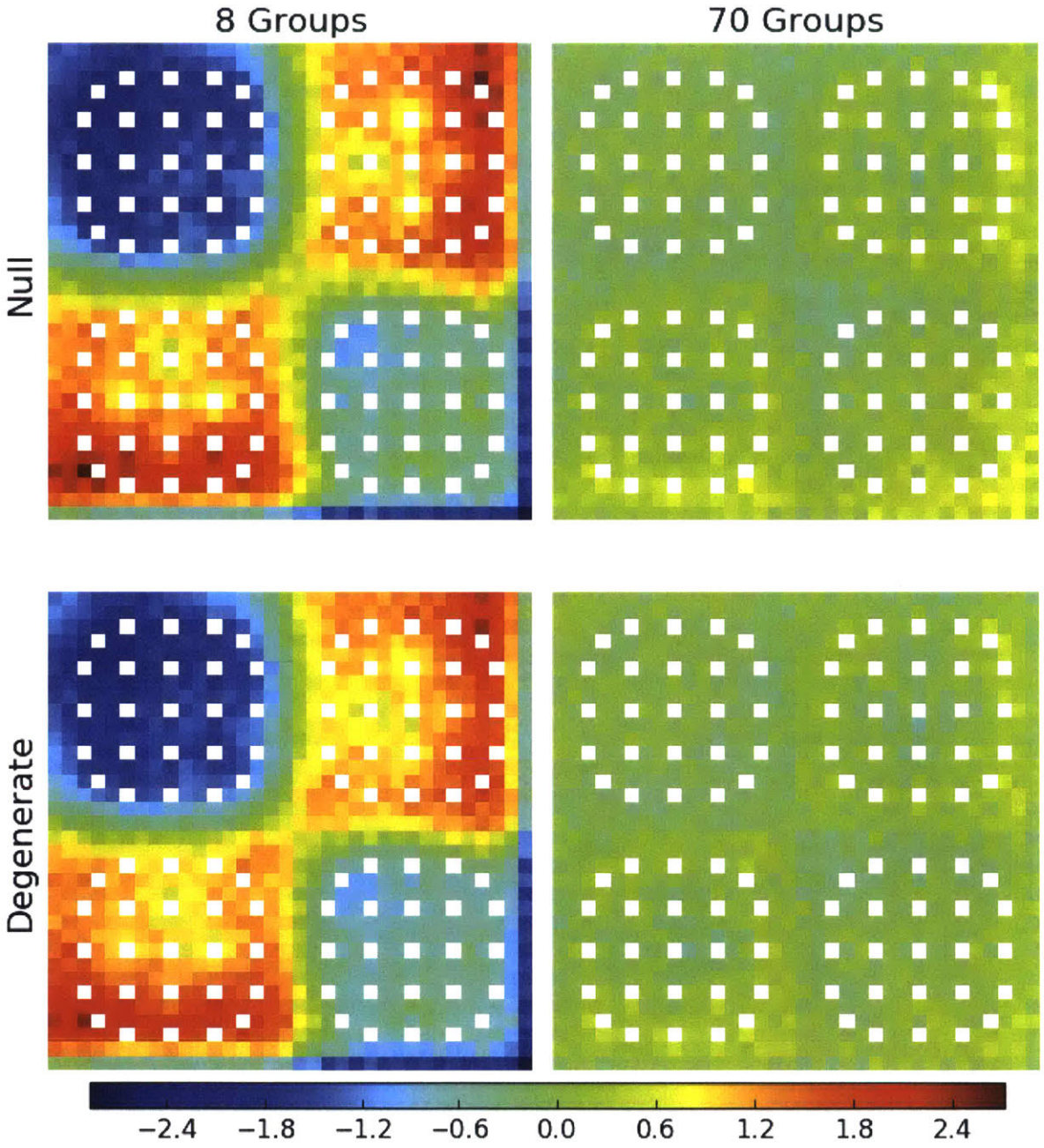
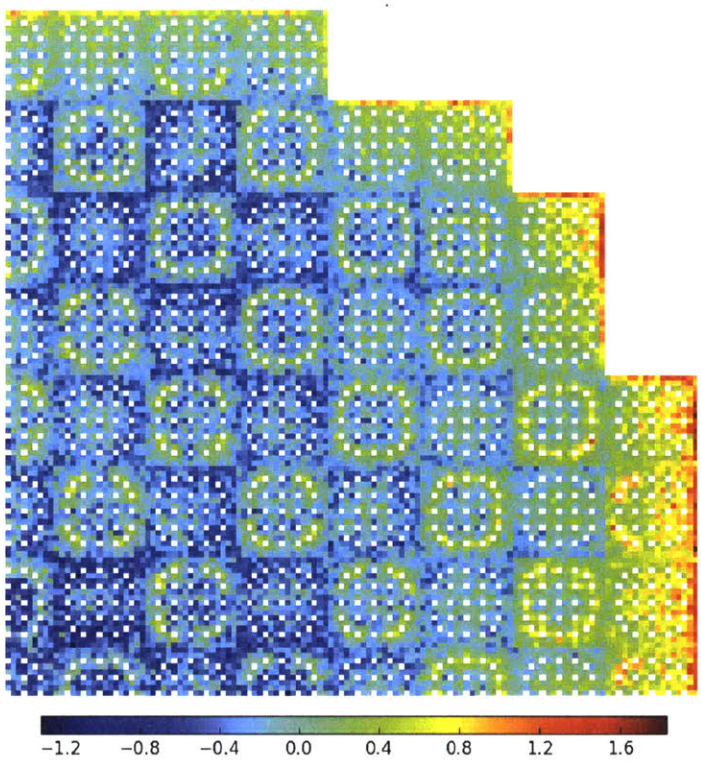
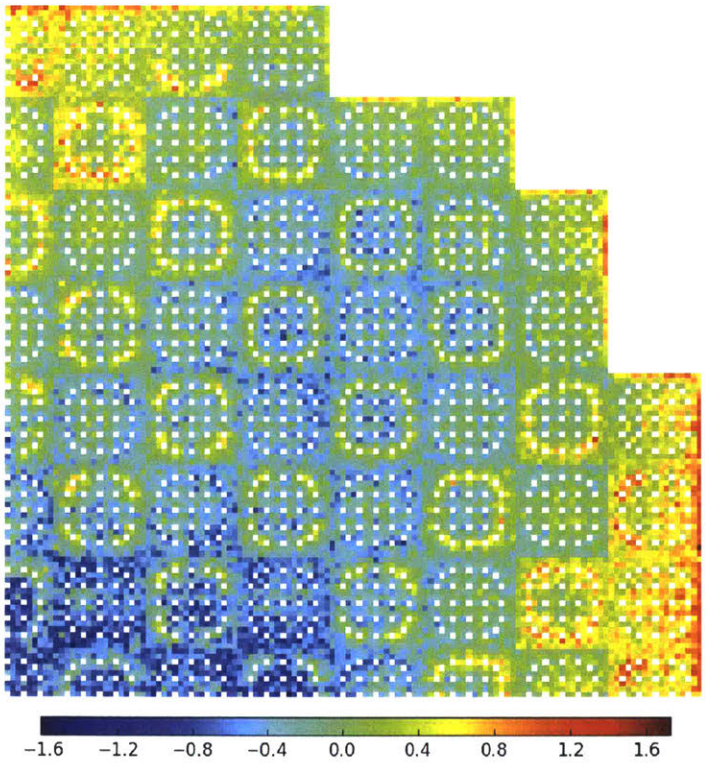


Figure 8-8: Fission rate percent relative errors for a 2×2 colorset with a reflector corresponding to the reference in Fig. 7-13.



(a)



(b)

Figure 8-9: Fission rate percent relative errors for the quarter core BEAVRS model with null (a) and degenerate (b) homogenization corresponding to the reference in Fig. 7-13.

8.4.3 U-238 Capture Rate Distributions

The OpenMOC energy-integrated U-238 capture rates were compared to the reference OpenMC fission rates from Figs. 7-15 to 7-20. The percent relative errors for each pin's capture rates were computed and the maximum and mean errors are listed for each benchmark, energy group structure and spatial homogenization scheme in Tabs. 8.7 and 8.8, respectively. In particular, the maximum errors are the maximum of the absolute values of the errors along with the appropriate sign, while the mean errors are the averages of the absolute error magnitudes.

Table 8.7: Maximum absolute U-238 capture rate percent relative errors for varying spatial homogenization schemes and energy group structures.

Benchmark	MGXS Scheme	Max Error [%]		
		2-Group	8-Group	70-Group
1.6% Assm	Infinite	-2.644	-1.480	-1.102
	Null	-2.629	-1.475	-1.101
	Degenerate	1.075	0.484	0.386
3.1% Assm	Infinite	-2.959	-1.678	-1.262
	Null	-2.934	-1.670	-1.262
	Degenerate	0.941	0.437	0.326
3.1% Assm w/ 20 BPs	Infinite	3.245	-1.278	-0.978
	Null	3.193	-1.253	-0.946
	Degenerate	1.635	0.571	0.311
2×2 Colorset	Infinite	-5.001	-1.743	-1.307
	Null	-4.591	-1.598	-1.305
	Degenerate	-2.983	-0.954	0.615
2×2 Colorset w/ Reflector	Infinite	12.010	3.618	-1.889
	Null	11.100	3.372	-1.969
	Degenerate	8.260	2.787	-0.783
BEAVRS Quarter Core	Infinite	-79.033	-31.419	-3.655
	Null	-84.954	-31.773	-3.645
	Degenerate	-84.624	-31.392	-2.067

As was the case for the other metrics, the U-238 capture rate errors are highly dependent on energy group structure. The maximum and mean errors are substantially reduced with finer energy group structures for all benchmarks and spatial homoge-

Table 8.8: Mean absolute U-238 capture rate percent relative errors for varying spatial homogenization schemes and energy group structures.

Benchmark	MGXS Scheme	Mean Error [%]		
		2-Group	8-Group	70-Group
1.6% Assm	Infinite	1.252	0.643	0.480
	Null	1.247	0.641	0.479
	Degenerate	0.367	0.091	0.086
3.1% Assm	Infinite	1.371	0.718	0.543
	Null	1.361	0.715	0.543
	Degenerate	0.326	0.106	0.087
3.1% Assm w/ 20 BPs	Infinite	1.278	0.551	0.424
	Null	1.261	0.543	0.414
	Degenerate	0.443	0.150	0.089
2×2 Colorset	Infinite	2.300	0.611	0.451
	Null	2.007	0.616	0.446
	Degenerate	1.482	0.170	0.154
2×2 Colorset w/ Reflector	Infinite	3.878	0.847	0.480
	Null	3.708	0.780	0.478
	Degenerate	3.302	0.655	0.165
BEAVRS Quarter Core	Infinite	34.706	10.005	0.556
	Null	39.137	10.164	0.474
	Degenerate	39.370	10.137	0.345

nization schemes. The maximum capture rate errors are 2 – 12% in 2 groups for the individual fuel assembly and 2×2 colorset benchmarks, respectively, but decrease to 0.4 – 1.7% when modeled with 70 groups. The mean capture rate errors likewise decrease with finer energy group structures, and are less than 0.6% in magnitude for the assembly and colorset benchmarks for all homogenization schemes with 70 groups.

The capture rate errors are more dependent on the spatial homogenization scheme used to compute MGXS in the fuel than are the fission rate errors. In particular, the degenerate scheme produces much smaller maximum and mean errors than the null and infinite schemes. The maximum error is greater than 1% for all benchmarks with the infinite and null schemes even with 70 energy groups. The maximum error is reduced by 2 – 5× for all benchmarks with the use of degenerate homogenization. In addition, the

degenerate scheme consistently reduces the error for all energy group structures, with the most substantial improvement over the infinite and null schemes observed for 70 groups. This stands in contrast to the fission rate errors where the degenerate scheme was only significantly better for 2-group MGXS. As was the case for the fission rate errors, the null and infinite schemes exhibit a negligible difference in their maximum and mean capture rate errors, with no systematic trend in the data.

The spatial distributions of capture rate errors are plotted as heatmaps for each benchmark in Figs. 8-10 to 8-15. These figures illustrate the capture rate errors for 8 and 70 energy group structures for the null and degenerate schemes; the capture rate errors for 2, 8, and 70 groups for all three homogenization schemes are compared in Figs. C-7 to C-12 in App. C.2. In addition, the U-238 capture absolute errors for the benchmarks with vacuum BCs are illustrated in App. C.3.

The heatmaps illustrate systematic error trends in the pin-wise capture errors which correlate with spatial heterogeneities in each benchmark. As is observed for the fission rates, the capture rates are generally underpredicted for pins near CRGTs, but overpredicted for pins removed from CRGTs, such as those in the corners of the fuel assembly. Unlike the fission rates, the capture rates are also underpredicted for pins near BPs. The effects due to heterogeneities are further enhanced when differentiating pins which are facially adjacent to CRGTs, facially and corner adjacent to two CRGTs, etc. In addition, the error magnitudes are large for pins along the inter-assembly and assembly-reflector interfaces for the 2×2 colorset benchmarks. For the PWR benchmarks modeled here, the moderation provided by neighboring CRGTs and/or reflectors softens the local flux for nearby fuel pins and should be modeled when collapsing pin-wise MGXS for high-fidelity multi-group transport calculations.

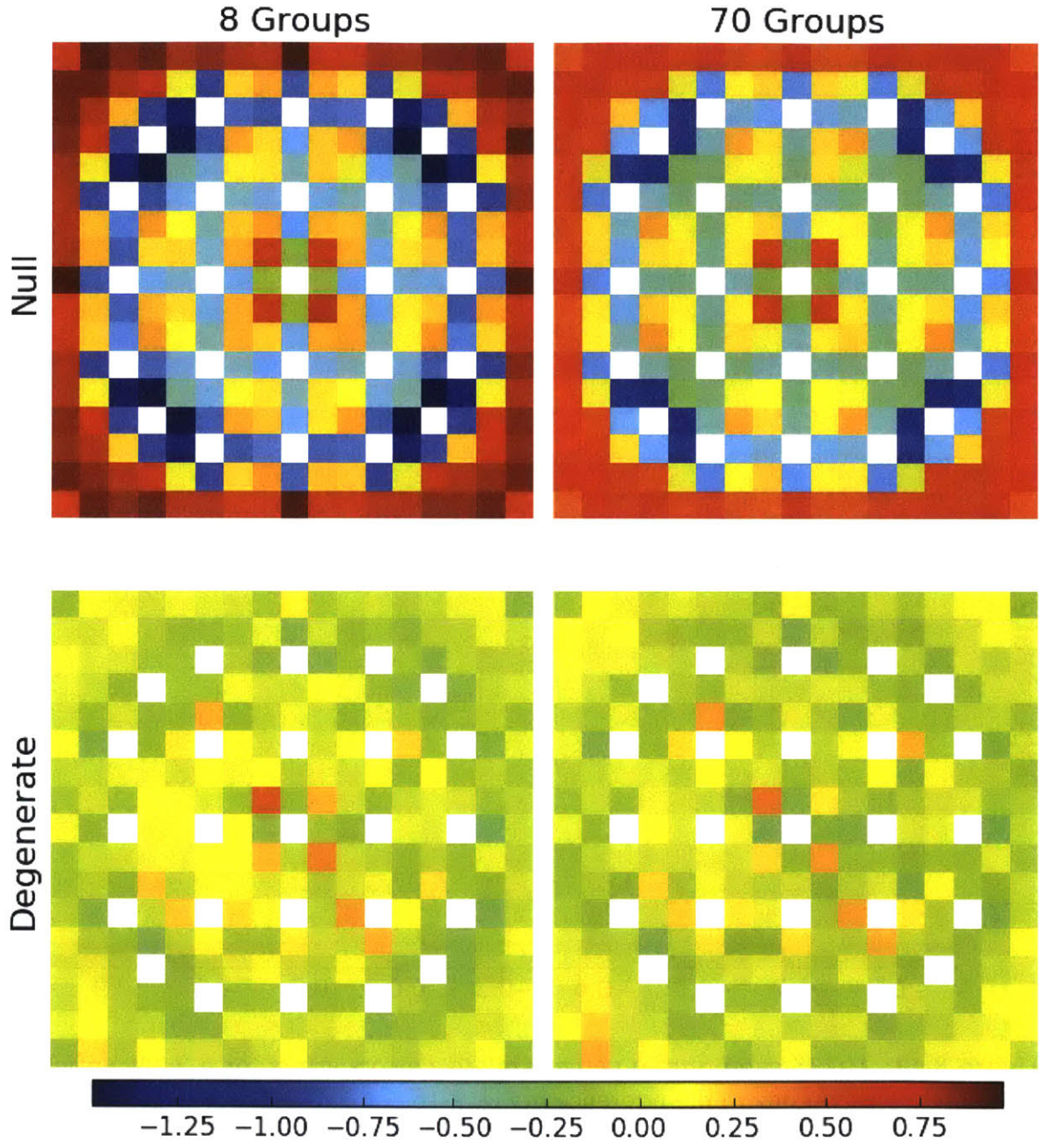


Figure 8-10: U-238 capture rate percent relative errors errors for a 1.6% enriched assembly corresponding to the reference in Fig. 7-15.

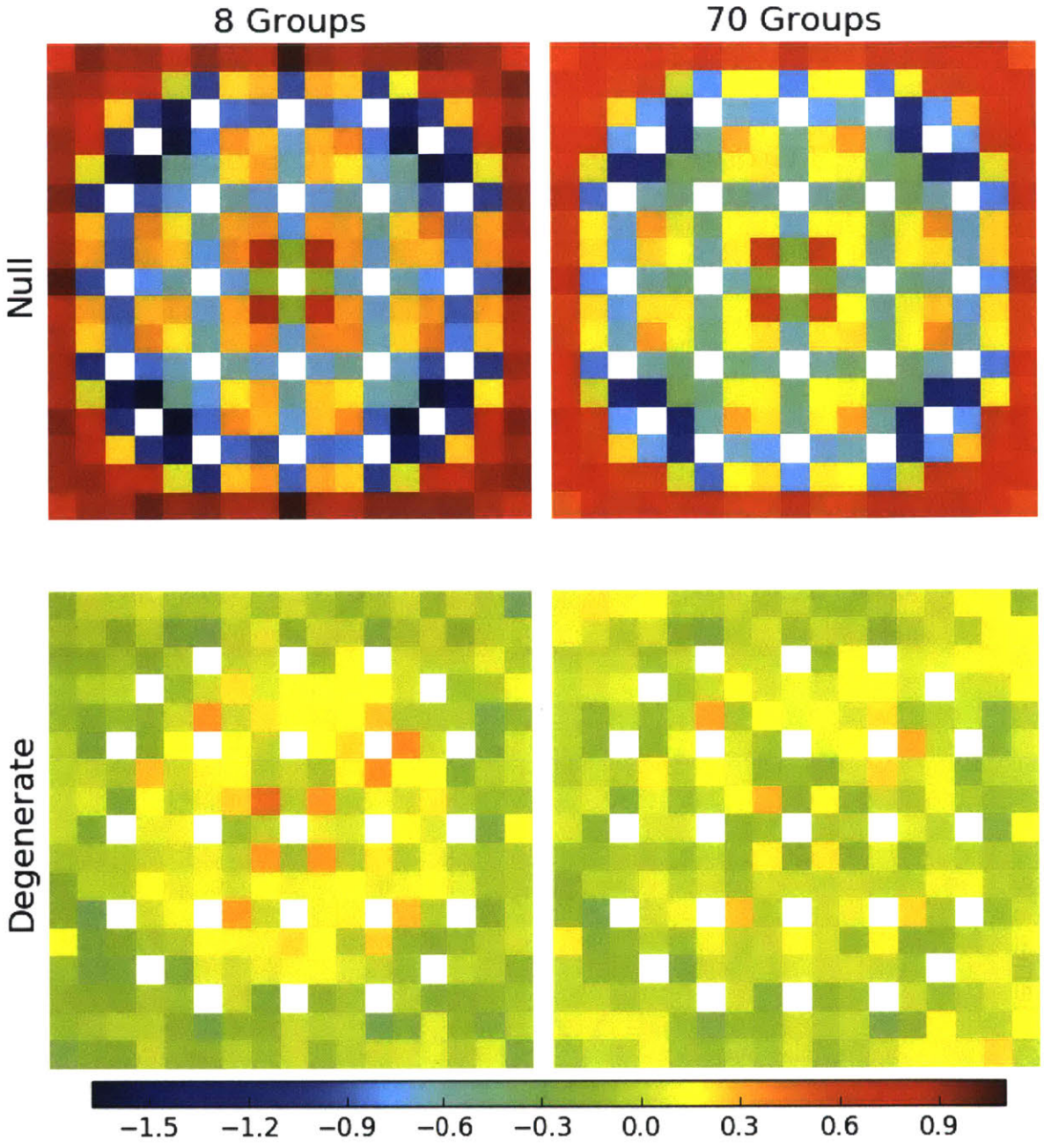


Figure 8-11: U-238 capture rate percent relative errors errors for a 3.1% enriched assembly corresponding to the reference in Fig. 7-16.

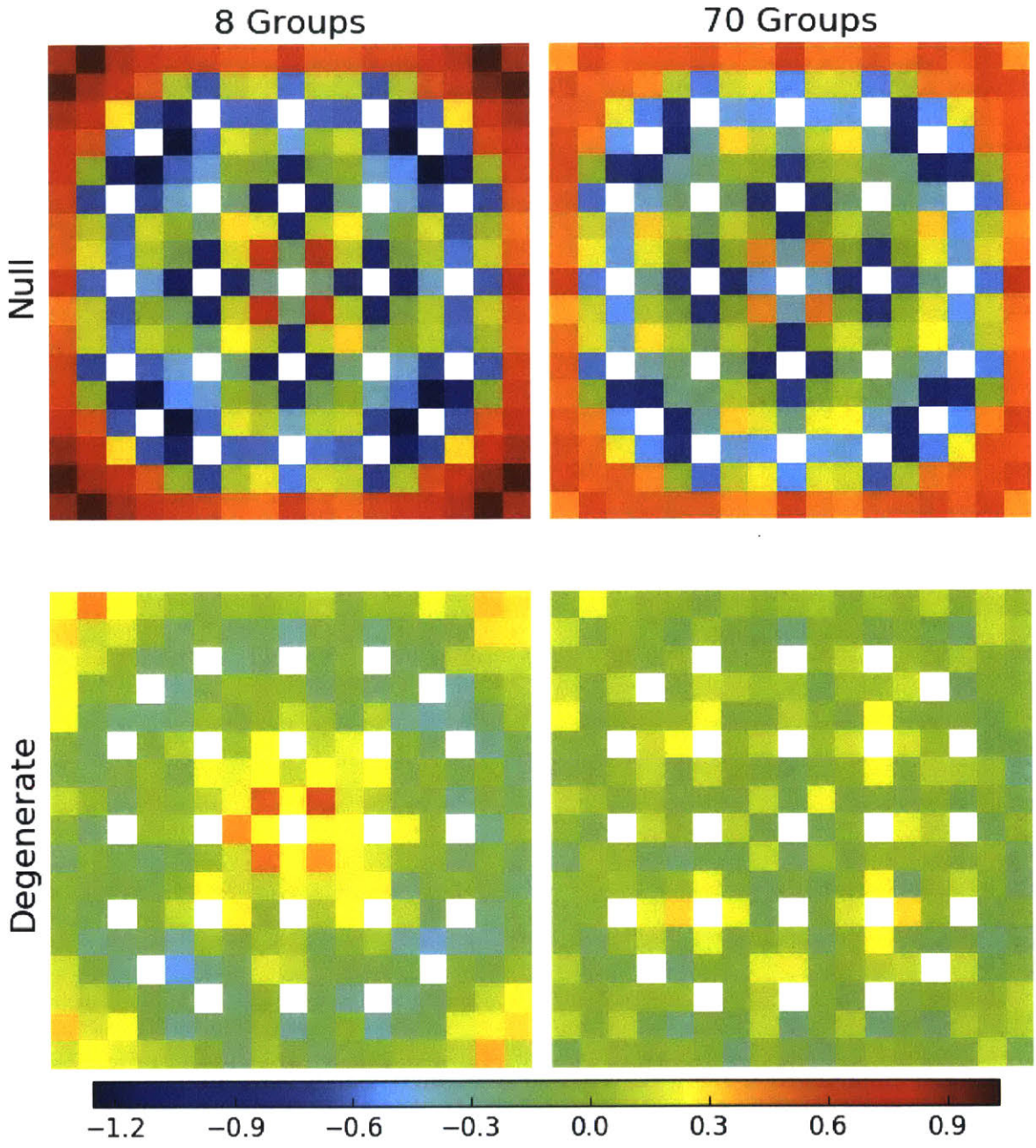


Figure 8-12: U-238 capture rate percent relative errors errors for a 3.1% enriched assembly with 20 BPs corresponding to the reference in Fig. 7-17.

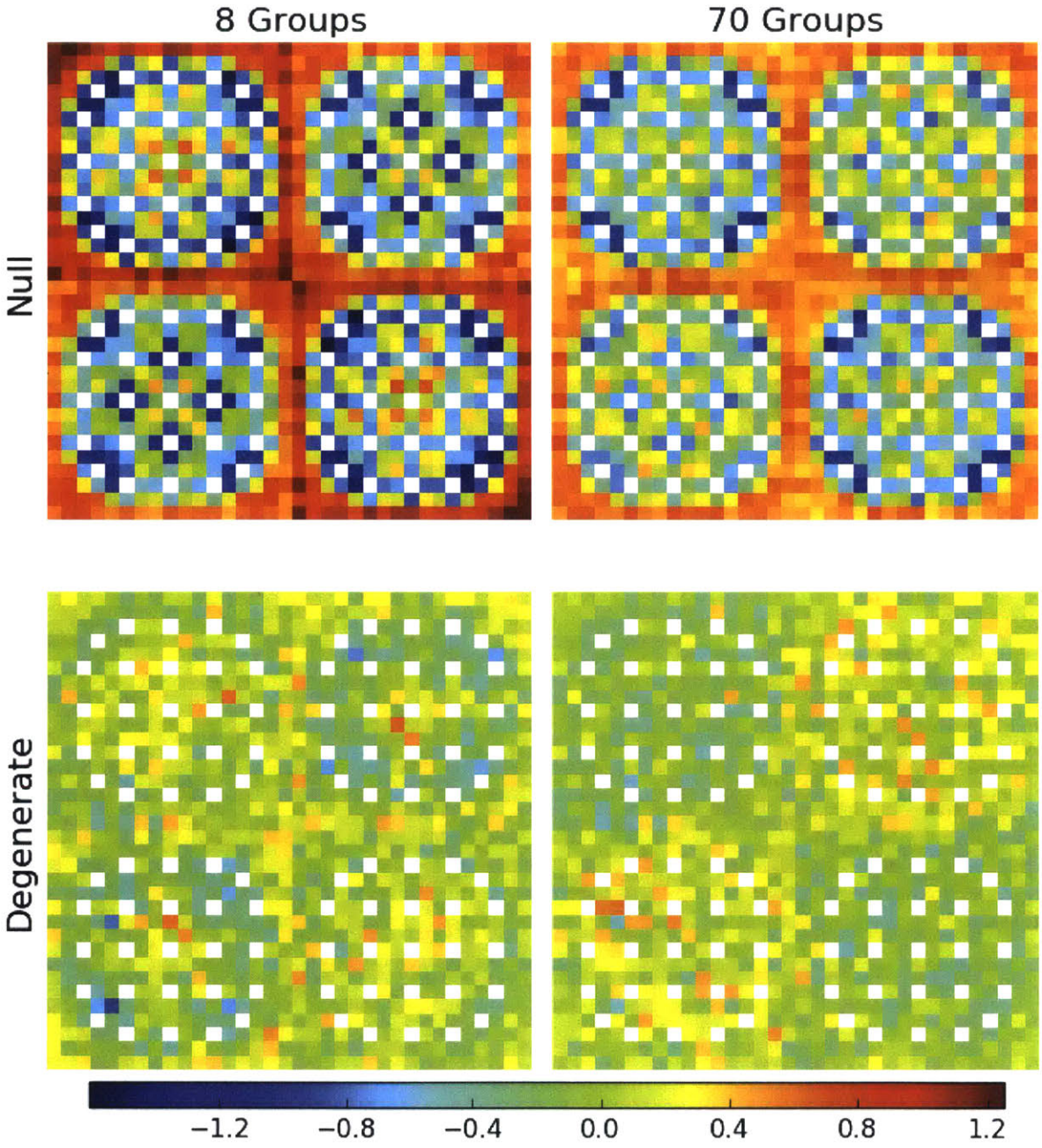


Figure 8-13: U-238 capture rate percent relative errors errors for a 2×2 colorset corresponding to the reference in Fig. 7-18.

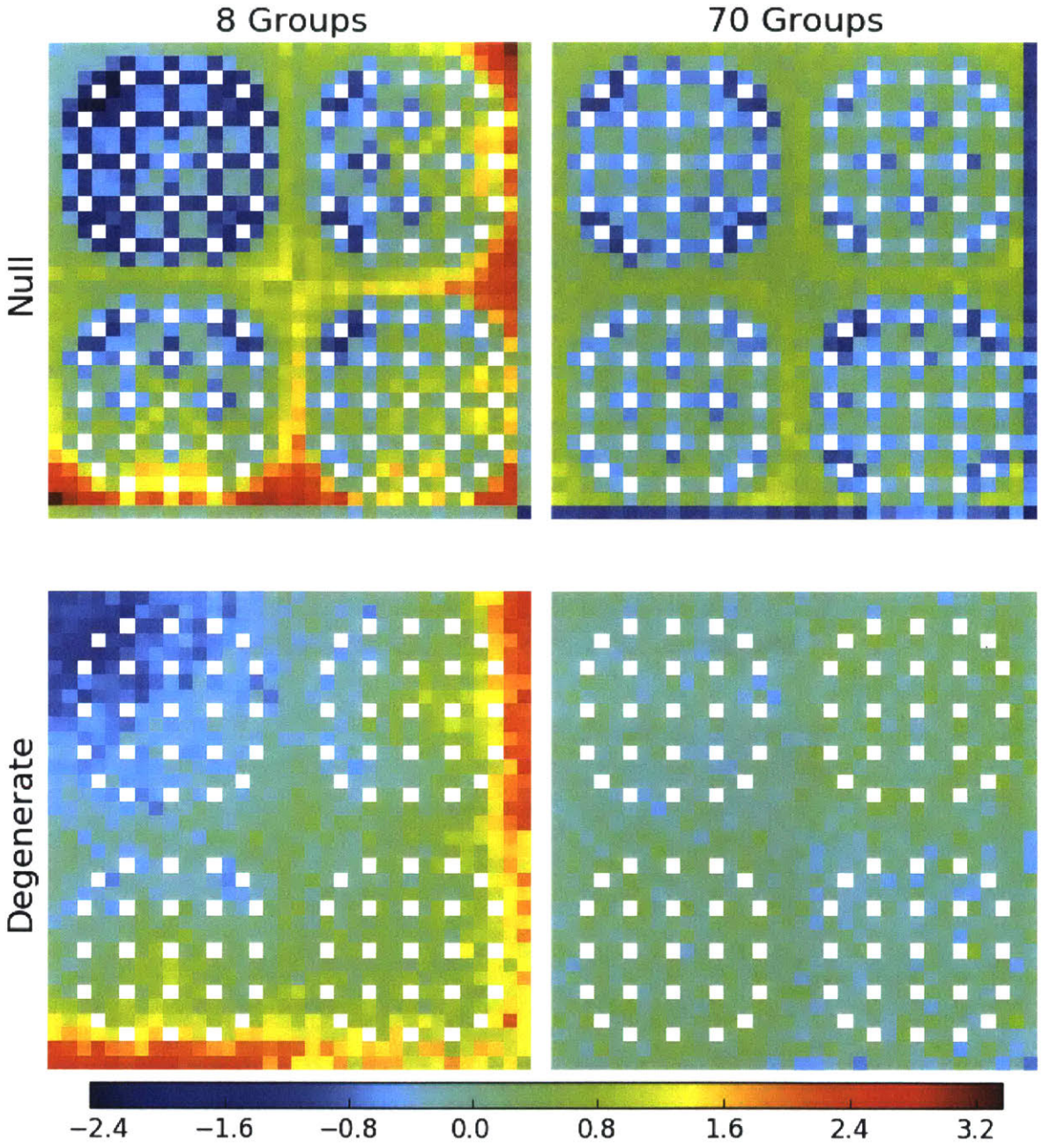
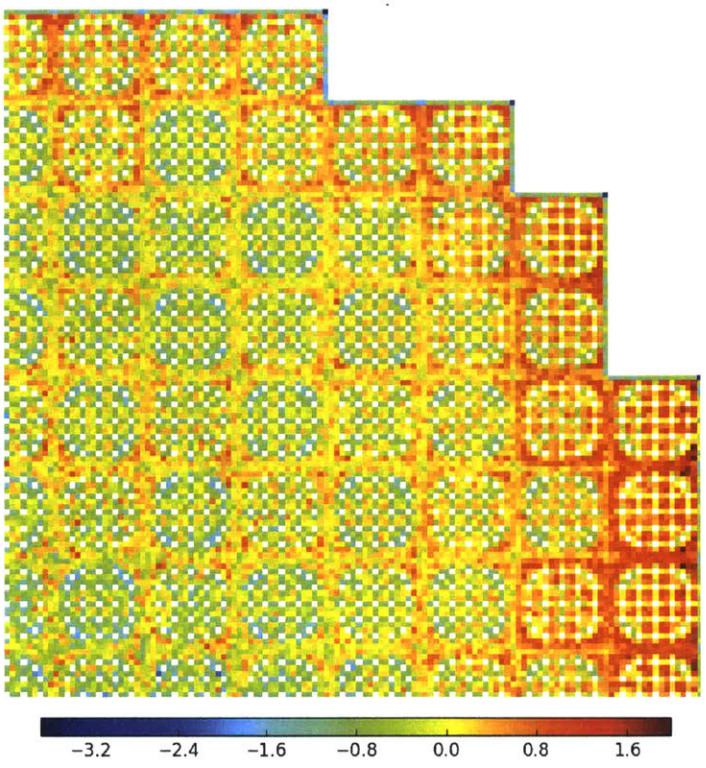
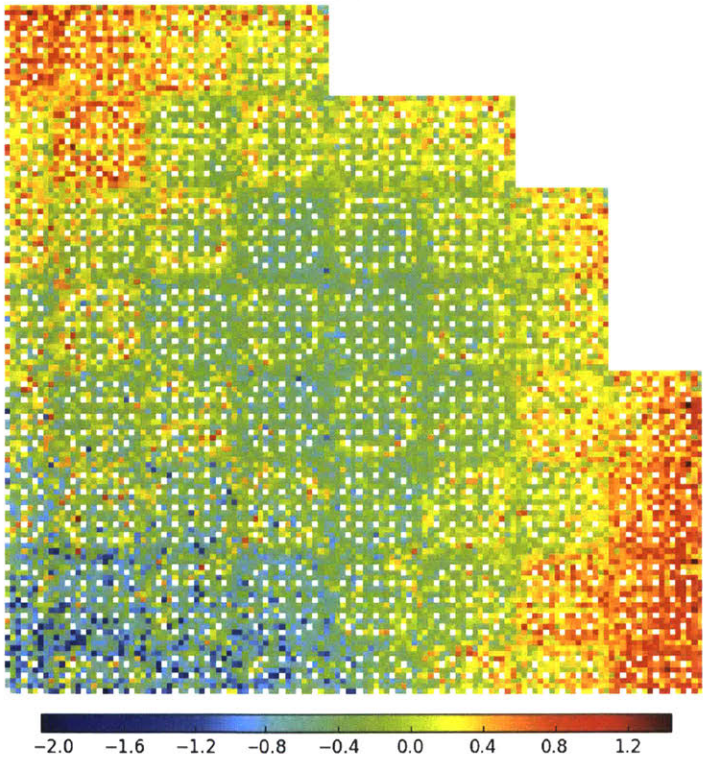


Figure 8-14: U-238 capture percent relative errors rate errors for a 2×2 colorset with a reflector corresponding to the reference in Fig. 7-19.



(a)



(b)

Figure 8-15: U-238 capture rate percent relative errors for the quarter core BEAVRS model with null (a) and degenerate (b) homogenization corresponding to the reference in Fig. 7-20.

As noted previously, the energy group structure has a large impact on the error distributions as can be seen from the figures. In addition, the heatmaps illustrate how the degenerate spatial homogenization scheme effectively "smooths" the pin-wise errors as compared to the infinite and null schemes. Although this effect appears most pronounced for the 2-group plots in App. C.2, the "smoothing" trend persists for 8 and 70 groups as well (unlike the fission rate errors). In particular, the differential of the errors for pins near CRGTs, BPs, and assembly and reflector interfaces is substantially reduced when degenerate homogenization is applied. The difference in the spatial distributions of errors for the infinite and null schemes is hardly noticeable except for pins near the assembly and reflector interfaces for the 2×2 colorset benchmarks.

The use of degenerate spatial homogenization "smooths" the spatial distribution of pin-wise U-238 capture rate errors. The improvement in accuracy with degenerate homogenization increases with more energy groups. This underscores the importance of accounting for spatial heterogeneities – such as the added moderation from CRGTs and reflectors – when generating MGXS to predict U-238 capture and Pu-239 production in LWRs.

8.5 Motivation for a New Spatial Homogenization Scheme

The results presented in the preceding section quantify and illustrate the impact of capturing heterogeneous spatial self-shielding effects in MGXS for fuel pins in PWR geometries. The pin-wise U-238 capture rates, and to a lesser extent, the pin-wise fission rates, are better predicted when these effects are incorporated into MGXS used in high-fidelity multi-group transport calculations. Alternatively, non-negligible systematic approximation errors in the reaction rates arise when using MGXS collapsed with the flux from an infinite lattice calculation which does not differentiate pins based on neighboring spatial heterogeneities[43]. It should be noted that other approximation errors due to spatial, angular and energy discretization in MOC are still present in these calculations,

[43]This is the case for many traditional approaches to self-shielding for MGXS generation.

and may complicate the generality of the conclusions drawn in this section.

The use of degenerate spatial homogenization reduces reaction rate errors by directly modeling perturbations in the flux within each fuel pin due to local heterogeneities such as neighboring CRGTs and BPs. Some reactions may be more or less sensitive to local spatial self-shielding effects. In the PWR benchmarks presented here, degenerate homogenization was most beneficial for predicting U-238 capture rates since it explicitly accounts for moderation from neighboring CRGTs and reflectors. Although degenerate spatial homogenization is advantageous for accuracy, it is not practical for routine reactor analysis due to computational resource limitations.

As shown in Tab. 8.1, the number of unique materials in the BEAVRS model is $\mathcal{O}(10^4)$ for degenerate homogenization[44], but only $\mathcal{O}(10)$ for the infinite and null homogenization schemes. As a result of the fine-grained spatial tally mesh employed by degenerate homogenization, far more particle histories are needed to converge the MGXS tallies to obtain the same statistical uncertainties as with the simpler schemes. In particular, the particle track density across each of the degenerate tally zones (*e.g*, fuel pins) is roughly one ten thousandth of that in the infinite/null schemes. Therefore, a factor of roughly 10,000 more particle histories are required to obtain the same particle track density within each of the degenerate tally zones[45] and converge the statistical uncertainties to the same level as would be obtained with infinite/null homogenization.

However, the spatial distribution of errors motivate the potential for a new homogenization scheme. In particular, the errors in the infinite/null homogenization schemes exhibit marked patterns which correlate with the local heterogeneities in the geometry. For example, the error of the predicted reaction rates for all pins facially adjacent to a CRGT is quite similar. More generally, the observed reaction rate errors are similar for groups of pins with similar neighboring local heterogeneities. This key observation implies that the reaction rate errors may be minimized if an appropriate set of spatially self-shielded MGXS are defined for each grouping of pins with similar flux profiles due

[44]The number of materials may be $\mathcal{O}(10^6)$ for 3D PWR models with axial enrichment zoning.

[45]For eigenvalue calculations in high dominance ratio reactor cores, the additional number of particle histories needed to converge MGXS tallies for degenerate homogenization may be even greater due to the highly uneven fission neutron source distribution.

to neighboring heterogeneities. In particular, can a library of MGXS be homogenized across a greatly reduced set of pins from the degenerate case which approaches both:

- the **accuracy** of the degenerate scheme
- the **convergence** of the infinite/null schemes

with far fewer particle histories than the degenerate scheme? This question is investigated with an in-depth analysis of clustering of pin-wise MGXS in the following chapter.

Highlights

- Each of the six heterogeneous benchmarks introduced in Chap. 7 is modeled in OpenMOC with MGXS generated by OpenMC.
- Three spatial homogenization schemes enable a direct quantification of spatial self-shielding effects from local heterogeneities in MGXS:
  - ***Infinite homogenization*** tallies MGXS for each unique fuel pin type using the MC flux in an infinite lattice calculation.
  - ***Null homogenization*** tallies MGXS for each unique fuel pin type using the MC flux from the complete heterogeneous geometry.
  - ***Degenerate homogenization*** tallies MGXS for each unique fuel pin instance using the MC flux from the complete heterogeneous geometry.
- The OpenMOC eigenvalues match OpenMC to within nearly 250 pcm for all benchmarks and homogenization schemes with 70-group MGXS.
- Degenerate homogenization best predicts reaction rates most sensitive to spatial self-shielding since it incorporates perturbations to the flux due to heterogeneities such as CRGTs and BPs.
  - Fission rate errors do not improve significantly
  - **U-238 capture rate errors are reduced by 2 – 4×**
- Degenerate homogenization requires far more particle histories to converge MC MGXS tallies than the simpler infinite and null schemes.
- The pin-wise reaction rate error distributions motivate the development of a new spatial homogenization scheme to identify groups of fuel pins which experience similar spatial self-shielding effects and have similar MGXS.

Chapter 9

Clustering of Pin-Wise MGXS

The preceding chapter quantified the benefit of using degenerate spatial homogenization to predict pin-wise U-238 capture rates as accurately as fission rates in high-fidelity multi-group transport methods. However, it was also noted that the degenerate scheme requires far more Monte Carlo particle histories to converge MGXS tallies than are necessary for the simpler null or infinite schemes. In addition, orders of magnitude more memory is needed to store MGXS libraries produced from degenerate homogenization. These observations motivate the need for a more sophisticated approach to pin-wise spatial homogenization which can simultaneously achieve nearly the accuracy of degenerate homogenization and the convergence of null homogenization.

This chapter seeks to accomplish this by leveraging the finding from Chap. 8 that pins with similar neighboring heterogeneities generally have similar reaction rate errors. For example, null homogenization led to a structured spatial distribution of errors with systematically similar errors in pins facially adjacent to one or two CRGTs, BPs, along inter-assembly and assembly-reflector interfaces, and so on. Since degenerate homogenization largely erased this structural error distribution, it follows that pins with similar errors likely experience similar spatial self-shielding effects due to neighboring heterogeneities. As a result, this and the following chapters develop the hypothesis that **pins with similar neighboring heterogeneities have similar microscopic MGXS**. If pins with similar microscopic MGXS can be identified, the MGXS tallied in these pin instances may be *homogenized* to compute an estimate which is nearly as accurate as the

MGXS from degenerate homogenization, and nearly as converged as the MGXS from null homogenization.

This chapter investigates this hypothesis by analyzing the pin-wise MGXS tallied with OpenMC to identify patterns – namely, clustering – for pins with similar neighbors. Furthermore, this chapter develops and quantifies a new spatial homogenization technique which analyzes a core geometry to predict which fuel pin instances have similar microscopic MGXS due to neighboring heterogeneities. This technique applies OpenCG's Local Neighbor Symmetry algorithm to generate a "geometric template" of fuel pins and averages the MGXS for pins across a core geometry with the same LNS identifiers. The impact of using LNS homogenization is compared to the null and degenerate schemes with respect to both general predictive accuracy as well as convergence. The results presented for LNS homogenization underscore the promise for an approach which combines the benefits of the null and degenerate schemes. However, the results also highlight some notable shortcomings to LNS which motivates the need for an unsupervised approach to MGXS clustering, as developed in the following chapter.

This chapter begins by analyzing patterns in visualizations of pin-wise MGXS in Sec. 9.1. This includes a case study of the statistical uncertainties and population variance of pin-wise MGXS in Secs. 9.1.1 and 9.1.2, and an analysis of the distributions of U-235 fission and U-238 capture MGXS for each of the six heterogeneous benchmarks with histograms and quantile-quantile plots in Secs. 9.1.3 and 9.1.4, respectively. A new spatial homogenization scheme based upon OpenCG's LNS algorithm is introduced in Sec. 9.2. The OpenMOC eigenvalues and pin-wise fission and U-238 capture rates with LNS spatial homogenization are presented in Sec. 9.3. Finally, the statistical uncertainties and convergence rate for MGXS generated with the null, degenerate and LNS schemes are compared in Sec. 9.4.

9.1 Clustering of Pin-Wise MGXS

This section investigates the clustering of pin-wise MGXS due to spatial self-shielding effects induced by neighboring heterogeneities. As a thought experiment, consider the

pin-wise MGXS in an infinite lattice[46]. Although the pin-wise MGXS in such a uniform geometry are necessarily identical, due to the stochastic nature of MC, the population of pin-wise MGXS estimates from MC will have a non-zero variance and represent random variates drawn from a normal distribution. When heterogeneities such as CRGTs and BPs are introduced into the geometry, they will induce local spatial self-shielding effects on nearby pins. As a result, the population of pin-wise MGXS will no longer be drawn from a simple normal distribution, but rather some mixture of potentially more complicated distributions. As shown in this section, such deviations may be observed from the population of pin-wise microscopic MGXS tallied with MC.

It is important to note that self-shielding effects induce clustering of microscopic rather than macroscopic MGXS. The pin-wise macroscopic MGXS cluster (or disperse) due to the different densities of nuclides in each fuel pin and give no indication of the similarity of the spectra experienced by each fuel pin. Since all of the benchmarks in this thesis use an isotopic vector for fresh PWR fuel, the clustering effects investigated here are necessarily the same for both micro and macro MGXS. However, the spatial homogenization methods developed in this thesis are intended to be used for more general applications with fuels of various burnups. Hence this section analyzes the microscopic MGXS since this is most appropriate for general purpose MGXS generation.

This section quantifies and visualizes the impact of heterogeneities on pin-wise MGXS. The MGXS data analyzed in this section was produced from the OpenMC simulations used to compute MGXS for Chap. 8 for degenerate homogenization. Sec. 9.1.1 begins by briefly quantifying the statistical uncertainties for the tallied MGXS datasets. Sec. 9.1.2 investigates the population variance of MGXS for each of the heterogeneous benchmarks and compares it to data for an infinite lattice[47]. Secs. 9.1.3 and 9.1.4 analyze histograms and quantile-quantile plots of the distribution of pin-wise MGXS for each of the benchmarks, respectively. Although the preceding chapter identified U-238 capture rates as being more sensitive to the pin-wise spatial homogenization model than the fission rates, this section explores both U-238 capture and U-235 fission MGXS data. As observed in

[46] An infinitely repeating lattice of identical fuel pins.

[47] A 17×17 assembly of identical fuel pins with reflective boundary conditions. The MGXS are tallied separately in each of the 289 identical fuel pins.

the following chapters, the clustering of one nuclide or reaction type's MGXS may not have a sizable impact on the corresponding reaction rate distribution (*e.g.*, fission), but may still reflect spatial self-shielding effects which can be leveraged to better predict other reaction rate distributions.

9.1.1 Pin-Wise MGXS Statistical Uncertainty

Before investigating MGXS clustering, it is important to first quantify the statistical uncertainties of the tallied pin-wise MGXS for each benchmark. The max and mean relative uncertainties[48] of the pin-wise U-238 capture and U-235 fission MGXS for an infinite lattice and each of the heterogeneous benchmarks are presented in Tab. 9.1. The standard deviations were computed by propagating the standard deviations of the reaction rate and flux tallies (Eqn. 3.9) as discussed in Sec. 3.2.2. The standard deviations were computed for all of the pins in each benchmark, and the max and mean values were selected for the table. The table highlights the standard deviations for both 1.6% and 3.1% enriched fuel pins[49]. The MGXS data was computed in two energy groups. The U-238 capture MGXS is analyzed for the first (fast) energy group since it encompasses the resonance region which is most sensitive to spatial self-shielding effects. The U-235 fission MGXS is analyzed for the second (thermal) energy group since it encompasses thermal energies which drives the majority of fission in LWRs.

There are a few key observations which can be made from the statistical uncertainties. First, the statistical uncertainties for both MGXS are relatively small with respect to the means. In particular, the relative uncertainties are 0.1 – 0.75% and 0.05 – 0.25% for the U-238 capture and U-235 fission MGXS, respectively. Furthermore, the reported standard deviations are likely overestimates since the reaction rate and flux tallies used to compute each MGXS are positively correlated (see Sec. 9.1). Therefore, the clustering of MGXS demonstrated in the following sections results from variation in MC sampling distributions with spatial heterogeneities rather than poor MC sample statistics.

[48]The *relative uncertainty* as used here is defined as the standard deviation of the sample mean divided by the sample mean.

[49]Although the BEAVRS model includes 2.4% enriched fuel pins, they were not included in this analysis.

Table 9.1: The percent relative uncertainties (1-sigma) for pin-wise U-235 fission and U-238 capture MGXS[50].

Fuel Enrichment	Benchmark	Metric	$\text{Var}\left[\hat{\sigma}_{c,1}^{238}\right]^{1/2}$ [%]	$\text{Var}\left[\hat{\sigma}_{f,2}^{235}\right]^{1/2}$ [%]
1.6%	Infinite Lattice	Max	0.10	0.05
		Mean	0.10	0.05
	Assm. (no BPs)	Max	0.10	0.05
		Mean	0.10	0.04
	2×2 Colorset	Max	0.20	0.10
		Mean	0.19	0.09
	2×2 Colorset w/ Reflector	Max	0.71	0.23
		Mean	0.24	0.11
	BEAVRS Quarter Core	Max	0.73	0.33
		Mean	0.28	0.13
3.1%	Infinite Lattice	Max	0.10	0.06
		Mean	0.10	0.06
	Assm. (no BPs)	Max	0.10	0.06
		Mean	0.10	0.05
	Assm. (20 BPs)	Max	0.10	0.06
		Mean	0.10	0.06
	2×2 Colorset	Max	0.20	0.12
		Mean	0.19	0.11
	2×2 Colorset w/ Reflector	Max	0.34	0.16
		Mean	0.22	0.12
	BEAVRS Quarter Core	Max	0.50	0.24
		Mean	0.19	0.11

In addition, the relative uncertainty increases with both the number of pins (and the corresponding decrease in particle track density in each tally volume) as well as the complexity of the heterogeneities in each benchmark. For example, there are 264 and 528 pins of a particular enrichment in the individual assembly benchmarks with CRGT and BPs, respectively. The uncertainties for the pins in the 2×2 colorset without a reflector are nearly 2× greater than those for the individual assemblies. The 1.6% and

[50]The reported uncertainties for the quarter core model may be underestimated due to correlations between fission source sites [40, 41].

3.1% enriched pins in the reflected colorset have uncertainties which are approximately 3× and 1.7× greater than those in the periodic colorset, respectively. This demonstrates the uneven distribution of particle track densities across the tally volumes in the reflected colorset, as evidenced by the power distributions in Figs. 7-12 and 7-13.

9.1.2 Pin-Wise MGXS Population Variance

The population variance is a useful metric to quantify the degree of dispersion of a probability distribution. The metric is used in this section to compare the dispersive effects of heterogeneities on pin-wise MGXS. The population variance σ^2 for a population of N random variates x_i drawn from a distribution with mean μ[51] is computed as follows:

$$\sigma^2 = \frac{\sum_{i=1}^{N}(x_i - \mu)^2}{N} \tag{9.1}$$

The population variance of U-238 capture and U-235 fission pin-wise MGXS, normalized to the mean of the population, are presented in Tab. 9.2 for an infinite lattice and each of the heterogeneous benchmarks. Each of the random samples in the variance calculation is the MGXS in one fuel pin instance in the respective benchmark. The table highlights the population variance for both 1.6% and 3.1% enriched fuel pins. The MGXS data was computed in two energy groups. The U-238 capture MGXS is analyzed for the first group since it encompasses the resonance region which is most sensitive to spatial self-shielding effects. The U-235 fission MGXS is analyzed for the second group since it encompasses thermal energies which drives the majority of fissions in LWRs.

A number of key trends emerge from the population variance data which support the overarching premise of this chapter – that spatial self-shielding effects from core heterogeneities induce clustering of pin-wise MGXS. First, the variance is over two orders of magnitude larger for each of the six benchmarks as compared to an infinite lattice, clearly indicating that spatial self-shielding from local heterogeneities disperses pin-wise MGXS. This dispersion is observed even for the individual fuel assemblies with

[51] If the mean μ is unknown, it may be empirically estimated from the sample mean $\bar{x}$.

Table 9.2: The population variance normalized to the population mean for pin-wise U-235 fission and U-238 capture MGXS.

Fuel Enrichment	Benchmark	$\text{Var}\left[\sigma_{c,1}^{238}\right]$ [%]	$\text{Var}\left[\sigma_{f,2}^{235}\right]$ [%]
1.6%	Infinite Lattice	0.10	0.02
	Assm. (no BPs)	1.55	0.37
	2×2 Colorset	1.66	0.94
	2×2 Colorset w/ Reflector	1.75	1.20
	BEAVRS Quarter Core	1.31	0.68
3.1%	Infinite Lattice	0.10	0.02
	Assm. (no BPs)	1.54	0.59
	Assm. (20 BPs)	1.44	0.99
	2×2 Colorset	1.53	1.36
	2×2 Colorset w/ Reflector	1.69	2.01
	BEAVRS Quarter Core	1.80	1.18

only a mixture of fuel pins and CRGTs. The introduction of BPs slightly reduces the variance for U-238 capture, but increases the variance by nearly 2× for U-235 fission MGXS. Inter-assembly effects in the 2×2 colorset increase the dispersion of U-238 capture MGXS by 10 – 20%, but dramatically increases the variance for the fission MGXS by up to 2.5×. The introduction of a water reflector to the colorset further increases the dispersion by 5 – 15% and 65 – 120% for U-238 capture and U-235 fission in the 1.6% and 3.1% enriched pins, respectively. Interestingly, the most of the population variances decrease by $1/3$ – $1/2$ for the BEAVRS quarter core with respect to the 2×2 colorset with a reflector. This is likely due to the smaller ratio of fuel pins along the assembly-reflector interface – which experience a dramatically softer flux spectrum than pins in the interior of the core – to total number of pins for the BEAVRS model.

Lastly, the variance of the U-235 fission MGXS is notably dependent on the enrichment. For example, the variance is nearly 2× larger for the assembly with 3.1% enriched fuel pins and CRGTs than the same assembly with 1.6% enriched fuel. This trend remains true for the larger, more complicated benchmarks. This observation is interesting in light of the results in Chap. 8 which demonstrated little improvement in the fission rate spatial distributions with degenerate homogenization. Notwithstanding, these results indicate

that the sensitivity of U-235 fission MGXS to spatial self-shielding may be leveraged to improve U-238 capture rate predictions.

Core heterogeneities such as CRGTs, BPs and reflectors increase the population variance of pin-wise MGXS due to spatial self-shielding effects. The magnitude of the dispersion varies by nuclide and reaction type, and is generally larger for thermal U-235 fission than fast/epithermal U-238 capture MGXS.

9.1.3 Histograms of Pin-Wise MGXS

This section expounds upon the preceding quantitative analysis of the dispersive effect of heterogeneities with a visual examination of the distributions of pin-wise MGXS. In particular, the following two sections present histograms of the pin-wise U-238 capture and U-235 fission MGXS to illustrate clustering due to spatial self-shielding effects. In addition, rug plots – which draw vertical ticks along the x-axis of each histogram – are used to further discern clustering within the distributions. The visualizations are presented for an infinite lattice as well as the six heterogeneous benchmarks for both 1.6% and 3.1% enriched fuel. The trends observed in the plots can be attributed to the presence of CRGTs, BPs, assembly-assembly and/or assembly-reflector interfaces or fuel enrichment.

As in Sec. 9.1.2, the random samples in each visualization (*e.g.*, each green tick in the rug plots) correspond to a single fuel pin instance in the corresponding benchmark model. Although this analysis could plausibly be performed for a variety of nuclides, reaction rates and/or energy groups, only U-238 capture and U-235 fission MGXS were selected due to their importance for reactor performance. It is also worth noting that the two group constants were selected since their tallied MC uncertainties are smaller than those for the 8- and 70-group MGXS. As a result, it is simpler to identify clustering trends in the 2-group MGXS data.

9.1.3.1 U-238 Capture MGXS

The pin-wise microscopic U-238 capture MGXS for 1.6% and 3.1% enriched fuel pins are illustrated with histograms and rug plots in Figs. 9-1 and 9-2, respectively. As expected based on the population variances, the distributions of pin-wise MGXS for the infinite lattices in Figs. 9-1a and 9-2a are narrow and symmetric – unlike the distributions for each of the heterogeneous benchmarks. The introduction of CRGTs to the individual fuel assemblies induces four clearly discernible clusters of MGXS in Figs. 9-1b and 9-2b which are separated by approximately 0.01 – 0.015 barns (or 1 – 2%) for both fuel enrichments. There may be further sub-clusters as evidenced by the dispersion of the rug plot ticks between the two lowest-lying clusters. These clusters are attributed to the softening of the flux due to moderation from neighboring CRGTs in increasing order for the following four groupings of fuel pins: (a) pins not adjacent to a CRGT, (b) pins corner adjacent to a CRGT, (c) pins facially adjacent to a CRGT and (d) pins facially and corner adjacent to separate CRGTs. The addition of BPs to the 3.1% enriched fuel assembly "smears" the four clusters of U-238 capture MGXS in Fig. 9-2c. In addition, the MGXS are shifted downwards by ~0.005 barns (or ~0.5%) with respect to the assembly without BPs.

The inter-assembly interfaces in the 2×2 colorset benchmark results in a further "smearing" of the clusters for both enrichments in Figs. 9-1c and 9-2d. It should be noted that there are twice as many samples (*e.g.*, 528 fuel pin instances for each enrichment) in the colorsets than the individual fuel assemblies. Hence, it is more challenging to distinguish clusters from the rug plots for the larger colorset benchmarks. Nonetheless, the histograms indicate the continuing presence of four distinct clusters for both enrichments, though they are easier to identify for the 1.6% enriched fuel pins (in assemblies without BPs). The MGXS are generally shifted up by ~0.01 barns for the 1.6% enriched pins, and down by ~0.0075 barns for the 3.1% enriched pins, with respect to the individual fuel assemblies.

The introduction of a reflector to the 2×2 colorset leads to very different distributions for the two fuel enrichments in Figs. 9-1d and 9-2e. The reason for this is that periodicity

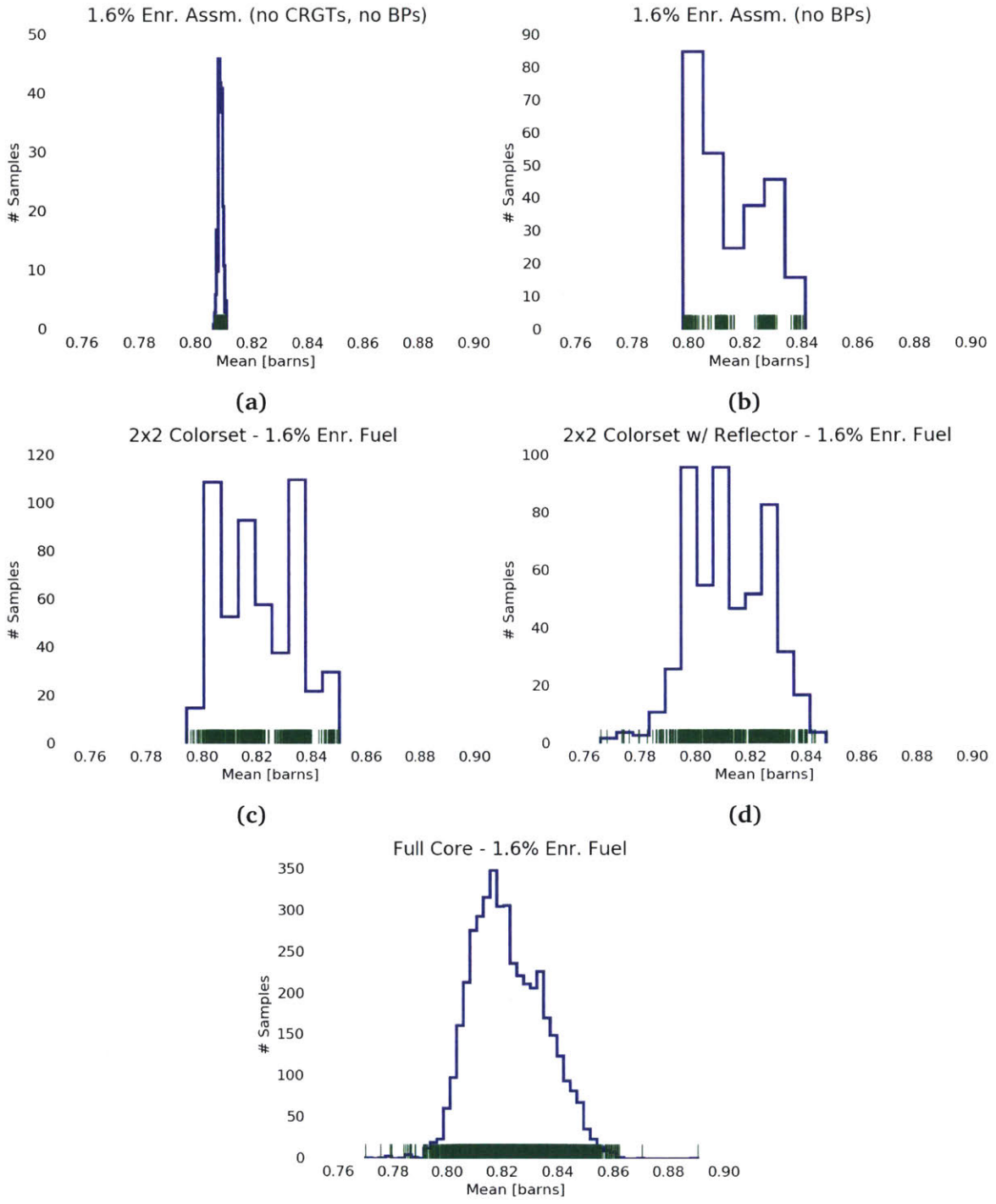


Figure 9-1: Histograms of U-238 capture MGXS (group 1 of 2) for 1.6% enriched fuel.

along the x and y axes is broken with the inclusion of the reflector. In particular, both of the 3.1% enriched assemblies are symmetrically adjacent to the reflector, and therefore have identical pin-wise MGXS. However, one of the 1.6% enriched assemblies is corner

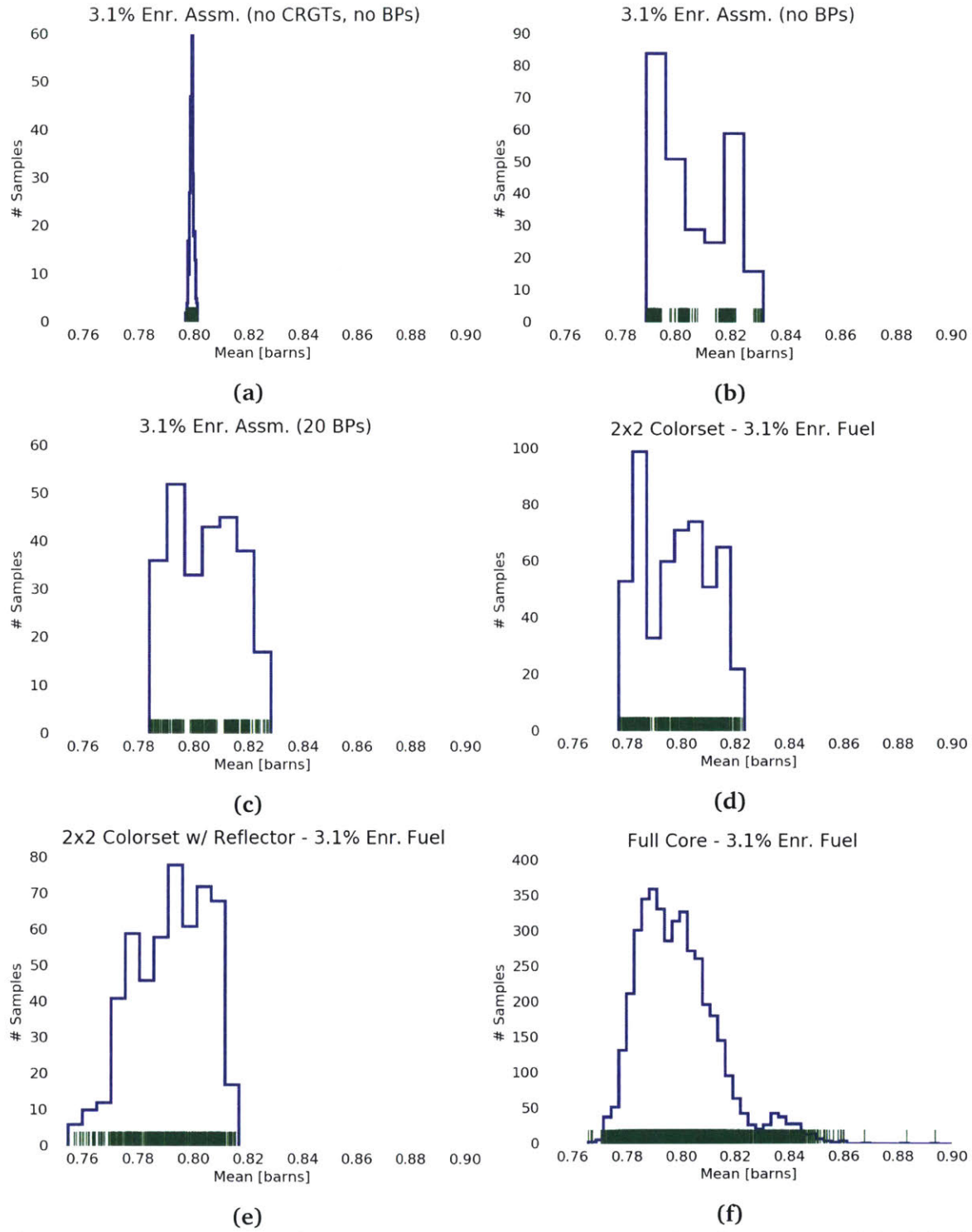


Figure 9-2: Histograms of U-238 capture MGXS (group 1 of 2) for 3.1% enriched fuel.

adjacent to the reflector while the other is located within the interior of the colorset. Nevertheless, some general observations can be made from the data. First, the MGXS

are generally even more "smeared" for both enrichments than is the case for the colorset without a reflector. The histogram indicates the existence of at least three clusters for the 1.6% enriched pins, with a similar but less concentrated three-way clustering for the 3.1% enriched pins. Finally, the lowest pin-wise MGXS for both enrichments are ~0.025 barns (or ~3%) less than those in the colorset without a reflector. This may be due to a harder spectrum in fuel pins along the left and top boundaries which have periodic BCs in the colorset without a reflector, but reflective BCs for the colorset with a water reflector.

The quarter core BEAVRS model has the smoothest varying distributions of any of the benchmarks in Figs. 9-1e and 9-2f. This is due in part to the fact that there are over 8× more samples for each fuel enrichment than are present for the colorsets[52]. The most notable observation is that the distributions appear to first order to be bimodal with large peaks centered at approximately 0.815 and 0.83 barns, and 0.785 and 0.805 barns, for the 1.6% and 3.1% enriched pins, respectively. The distribution for the 3.1% enriched fuel pins has an additional smaller shoulder centered at 0.83 barns. Additionally, a few 3.1% enriched pins have significantly greater MGXS between 0.86 and 0.9 barns. This is likely due to the fact that the 3.1% enriched assemblies surround the exterior of the core configuration. As a result, the flux is more strongly shielded in the U-238 resonance energies for 3.1% enriched pins adjacent to the baffle/reflector than the 1.6% enriched pins in the interior, resulting in some larger "outlier" pin-wise MGXS.

Finally, it should be noted that the microscopic U-238 capture MGXS are generally 0.01 – 0.02 barns (or 1.25 – 2.5%) larger for the 1.6% than the 3.1% enriched fuel pins for each respective benchmark, including the infinite lattice. These results indicate that the flux is less shielded at resonance energies for the lesser enriched fuel, perhaps due to a slightly larger fraction of neutrons up-scattering from group two to one due to the smaller probability of fission. This variation of U-238 capture MGXS with fuel enrichment remains true even for the quarter core BEAVRS model. However, the single largest U-238 capture MGXS is in a 3.1% enriched fuel pin corner adjacent to the baffle, which likely has the softest spectrum of any pin due to the nearby reflector.

[52]There are 4,332 1.6%, 4,260 2.4% and 4,236 3.1% enriched fuel pins in the quarter core model.

The dispersion of pin-wise U-238 capture MGXS is highly uneven and structured. Core heterogeneities such as CRGTs, BPs and reflectors induce clustering due to spatial self-shielding effects which are observed from distributions of the pin-wise MGXS.

9.1.3.2 U-235 Fission MGXS

The pin-wise microscopic U-235 fission MGXS for 1.6% and 3.1% enriched fuel pins are illustrated with histograms and rug plots in Figs. 9-3 and 9-4, respectively. It should first be noted that the U-235 fission MGXS are over 300× larger than the U-238 capture MGXS illustrated in the preceding section. However, it should be recalled that the U-238 nuclide density is over 30 – 60× larger than U-235 in the 1.6% and 3.1% enriched fuel, respectively. As a consequence, the clustering of either microscopic U-238 capture or the U-235 fission MGXS is important since it will have a similarly sized impact on the corresponding macroscopic MGXS and reaction rates.

As was observed for U-238 capture, the distributions of pin-wise MGXS for the infinite lattices in Figs. 9-3a and 9-4a are narrow and symmetric unlike the distributions for each of the heterogeneous benchmarks. Unlike the situation for U-238 capture, the introduction of CRGTs induces a more intricate clustering of the U-235 fission MGXS in Figs. 9-3b and 9-4b. However, the total range of U-235 fission MGXS in the fuel assemblies with CRGTs varies by only 1 – 2% as compared to more than 5% for U-238 capture. The histograms indicate the presence of roughly three clusters of MGXS which are separated by 1 – 2 barns for both fuel enrichments. The rug plots illustrate a more complicated dispersion, however, with a significantly more distinct concentration into approximately eight clusters for the 3.1% enriched fuel pins. These observations indicate that U-235 fission may be more sensitive than U-238 capture to spatial self-shielding effects from the added moderation of CRGTs, even though the impact on the resultant MGXS is of a smaller relative magnitude. This is the result of the $1/v$ profile of the U-235 fission cross section at thermal energies which is particularly sensitive to thermalization when weighted by the flux to compute MGXS.

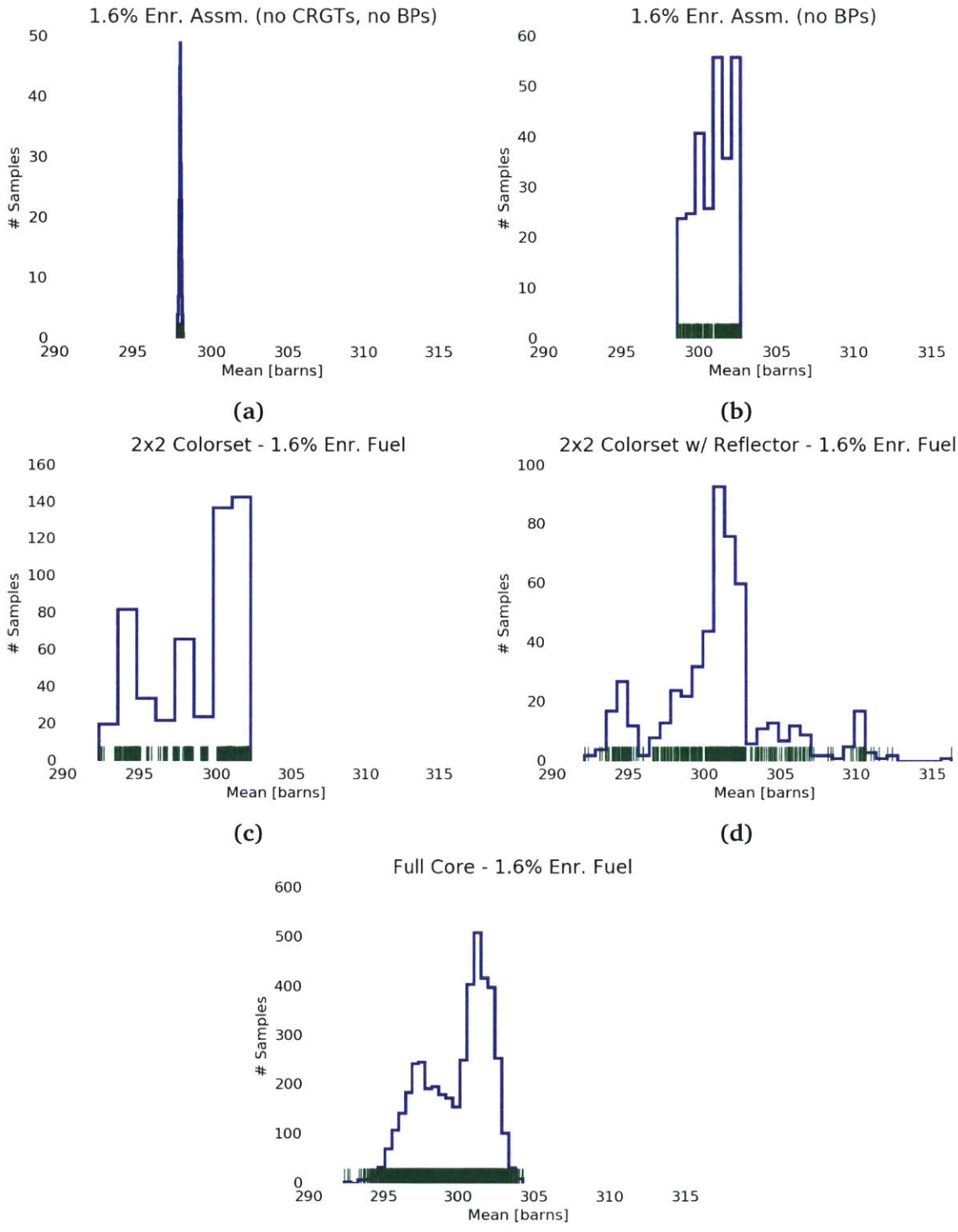


Figure 9-3: Histograms of U-235 fission MGXS (group 2 of 2) for 1.6% enriched fuel.

The further addition of BPs to the 3.1% enriched fuel assembly results in a more distinct set of isolated clusters of U-235 fission MGXS in Fig. 9-4c. In particular, there

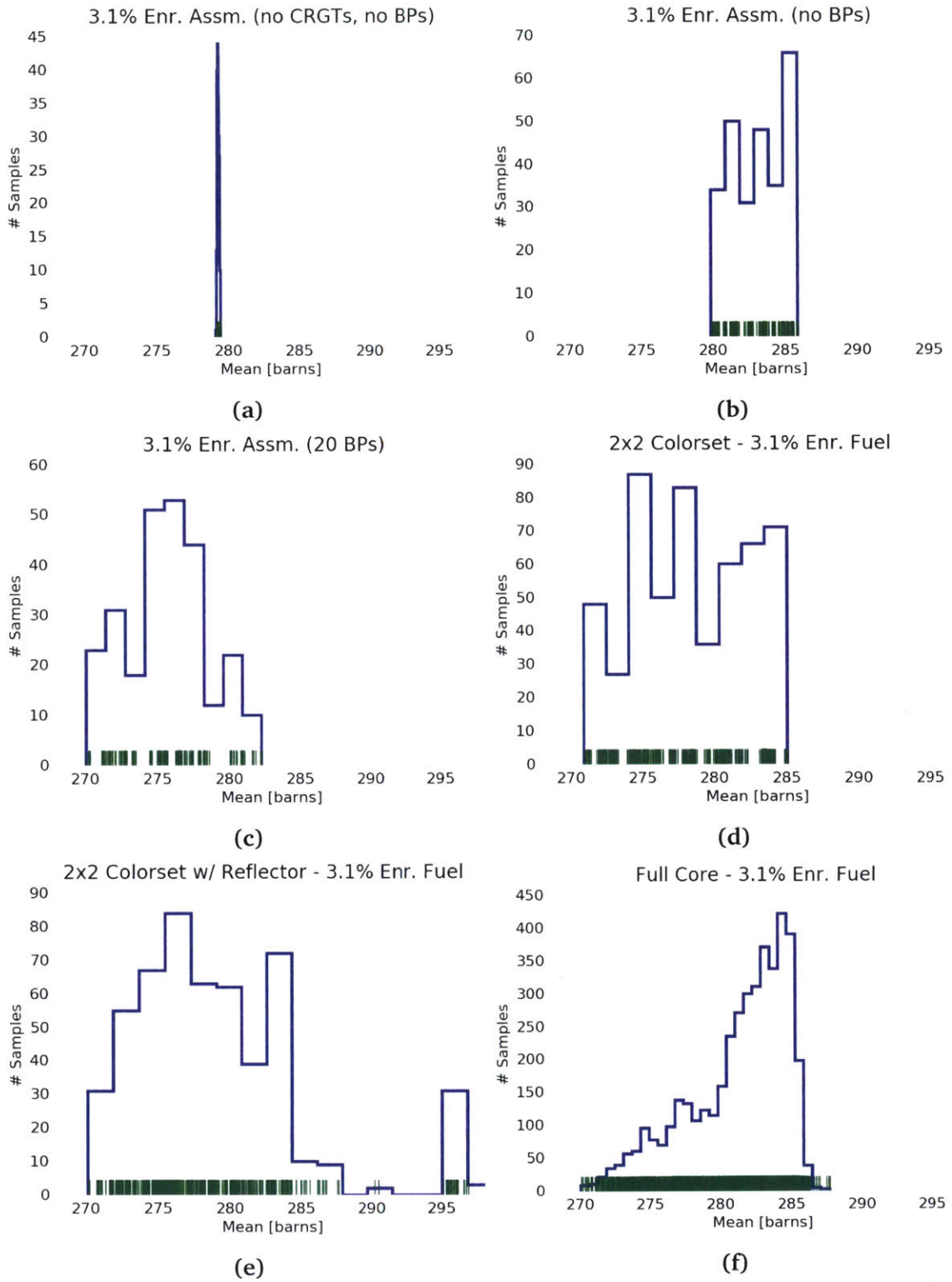


Figure 9-4: Histograms of U-235 fission MGXS (group 2 of 2) for 3.1% enriched fuel.

appear to be three overall clusters centered at approximately 272, 276 and 300 barns each with a number of sub-clusters. This stands in contrast to the case for U-238 capture

where the presence of BPs resulted in a "smearing" of the clusters. In addition, the MGXS are generally shifted downwards by 5 – 10 barns (or 1.5 – 3%) with respect to the assembly without BPs. The inter-assembly interfaces in the 2×2 colorset benchmark have a similarly concentrating effect for both enrichments in Figs. 9-3c and 9-4d. The histograms reveal three clusters centered at approximately 294, 298 and 301 barns for the 1.6% enriched pins, and four clusters centered at 271, 275, 278 and 283 barns for the 3.1% enriched pins. Furthermore, the rug plots indicate the presence of more clusters which are easier to distinguish than they are for the individual fuel assemblies, or for the U-238 capture MGXS in the 2×2 colorset benchmark. Finally, the MGXS in the colorset are generally shifted downwards by 3 – 6 barns (or 1 – 2%) with respect to the individual fuel assemblies.

The introduction of a reflector to the 2×2 colorset leads to very different distributions for the two fuel enrichments in Figs. 9-3d and 9-4e. As was noted in Sec. 9.1.3.1, the reason for this is that symmetry is broken with the inclusion of the reflector. A first observation is that the distributions of U-235 fission MGXS for the two enrichments differ much more profoundly than they do for U-238 capture. The histogram for the 1.6% enriched pins indicate perhaps six primary clusters centered at 294, 298, 301, 304, 306 and 310 barns, though the rug plot certainly indicates the presence of even more sub-clusters. The histogram for the 3.1% enriched pins indicate perhaps four clusters roughly centered at 276, 284, 290 and 296 barns, though the intervals between these apparent clusters contain many additional samples, complicating the analysis. Finally, the largest U-235 fission MGXS are approximately 315 and 300 barns for the 1.6% and 3.1% enriched pins in the colorset, respectively, about 15 barns (or 5%) greater than the largest MGXS for the colorset without a reflector. This is due to the softer flux spectrum experienced by those pins adjacent to the reflector.

As was noted for the U-238 capture MGXS, the quarter core BEAVRS model has the smoothest varying distributions of any of the benchmarks in Figs. 9-3e and 9-4f. The most notable observation is that the distributions appear to first order to be bimodal and trimodal with large peaks centered at approximately 297 and 302 barns, and 278, 283 and 284 barns, for the 1.6% and 3.1% enriched pins, respectively. The two peaks

are more clearly discernible for the 1.6% enriched fuel pins. This effect may be due to an increase in moderation for those 1.6% enriched assemblies that are only one assembly removed from the reflector as compared to those in the interior of the core. Furthermore, the range of MGXS in the quarter core model only spans 10 – 15 barns (or 5%) as compared to the 20 – 25 barns for the 2× colorset benchmark. This is likely due to the presence of a stainless steel baffle separating the assemblies on the exterior of the BEAVRS model from the water reflector. The baffle dampens the additional moderation experienced by those pins nearest the reflector in the BEAVRS model with respect to those in the 2×2 colorset with a reflector but no baffle.

Finally, it should be noted that the microscopic U-235 fission MGXS are generally 18 – 22 barns (or 6+%) larger for the 1.6% than the 3.1% enriched fuel pins for each respective benchmark, including the infinite lattice. These results indicate that the flux is more strongly moderated for the lesser enriched fuel, perhaps due to the smaller probability of fission. This variation of U-235 fission MGXS with fuel enrichment remains true even for the quarter core BEAVRS model.

The presence of core heterogeneities induce more clearly defined clustering of pin-wise U-235 fission MGXS than is observed for U-238 capture (in 2-group data). However, the population of pin-wise U-235 fission MGXS only vary by up to 1% about the mean, while the U-238 capture MGXS variation is ~2.5%.

9.1.4 Quantile-Quantile Plots of Pin-Wise MGXS

This section illustrates the deviation from normality of pin-wise MGXS with Quantile-Quantile (Q-Q) plots. A Q-Q plot is used to compare two datasets drawn from different probability distributions. In particular, the quantiles from one dataset are plotted against the quantiles of a second dataset in an (x, y) scatter plot[53]. If the two distributions are similar, the data points will lie along the $y = x$ reference line. Departures from $y = x$ indicate dis-similarities between the distributions, such as shifts in location or scale,

[53] A *quantile* is the point at which some fraction of the data falls below a given value. For example, the 10% quantile is the point at which 10% of the data is below the point and 90% is above it.

changes in symmetry, and outliers.

In this section, the theoretical quantiles for a normal distribution are plotted against empirical quantiles found from the empirical distribution of pin-wise U-238 capture and U-235 fission MGXS. These Q-Q plots illustrate the deviation from normality for the data as heterogeneities are introduced to the benchmark models. It should be noted that the MGXS data is first standardized so that it can be compared to theoretical quantiles from a standard normal distribution $\mathcal{N}(0,1)$ in the Q-Q plots. The visualizations are presented for an infinite lattice as well as the six heterogeneous benchmarks for both 1.6% and 3.1% enriched fuel. The random samples in each visualization (*e.g.*, each blue data point) correspond to a single fuel pin instance in the corresponding benchmark model. In addition, each plot highlights the p-value[54] for the Shapiro-Wilk test of normality [82] computed using the Python `scipy.stats` package [59][55].

9.1.4.1 U-238 Capture MGXS

The pin-wise microscopic U-238 capture MGXS for 1.6% and 3.1% enriched fuel pins are illustrated with histograms and rug plots in Fig. 9-5 and 9-6, respectively. As was hypothesized earlier, the datasets for the infinite lattices in Figs. 9-5a and 9-6a appear to be from a normal distribution since they lie very close to the $y = x$ reference line. In addition, the p-values are greater than 0.7, well above commonly used significance levels; thus one would not reject the null hypothesis that the data is from a normal distribution. The addition of CRGTs leads to a clear deviation from normality in Figs. 9-5b and 9-6b, with four "shoulders" representing the four clusters highlighted in Sec. 9.1.3 for the single assembly benchmarks. The presence of BPs seems to result in more shoulders in Fig. 9-6c corresponding to the various clusters identified in Fig. 9-2c.

At first glance, the larger number of data points (*e.g.*, fuel pin instances) in the Q-Q

[54]The p-value is used in null hypothesis significance testing and measures the likelihood of an observation as or more extreme than the given one. In this section, the null hypothesis is that pin-wise MGXS data is drawn from a normal distribution. To test this hypothesis, a significance level α is chosen (*e.g.*, 1%) and compared with the p-value of the Shapiro-Wilk normality test. If $p < \alpha$ then the null hypothesis is rejected (*i.e.*, the dataset was not drawn from a normal distribution); otherwise, it cannot be rejected.

[55]The implementation of the Shapiro-Wilk test in `scipy.stats` is stated to be accurate up to at least 5,000 data points. This is suitable for the datasets considered here which are largest for the quarter core BEAVRS model with ~4,500 points for the each fuel enrichment.

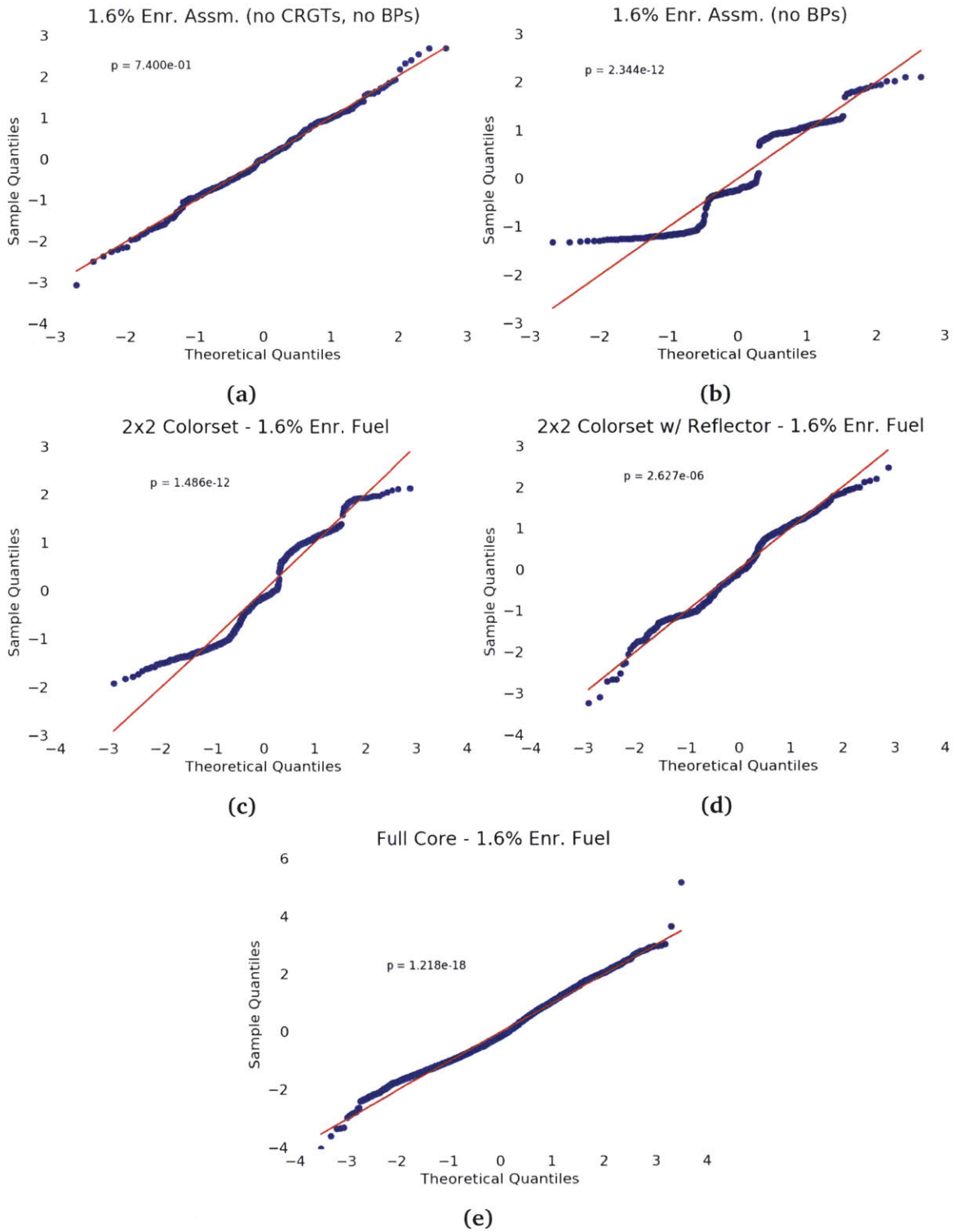


Figure 9-5: Q-Q plots of U-238 capture MGXS (group 1 of 2) for 1.6% enriched fuel.

plots for the larger colorset and quarter core benchmarks appear to hide any underlying structure. Although the data points may appear to lie closer to the $y = x$ line for the

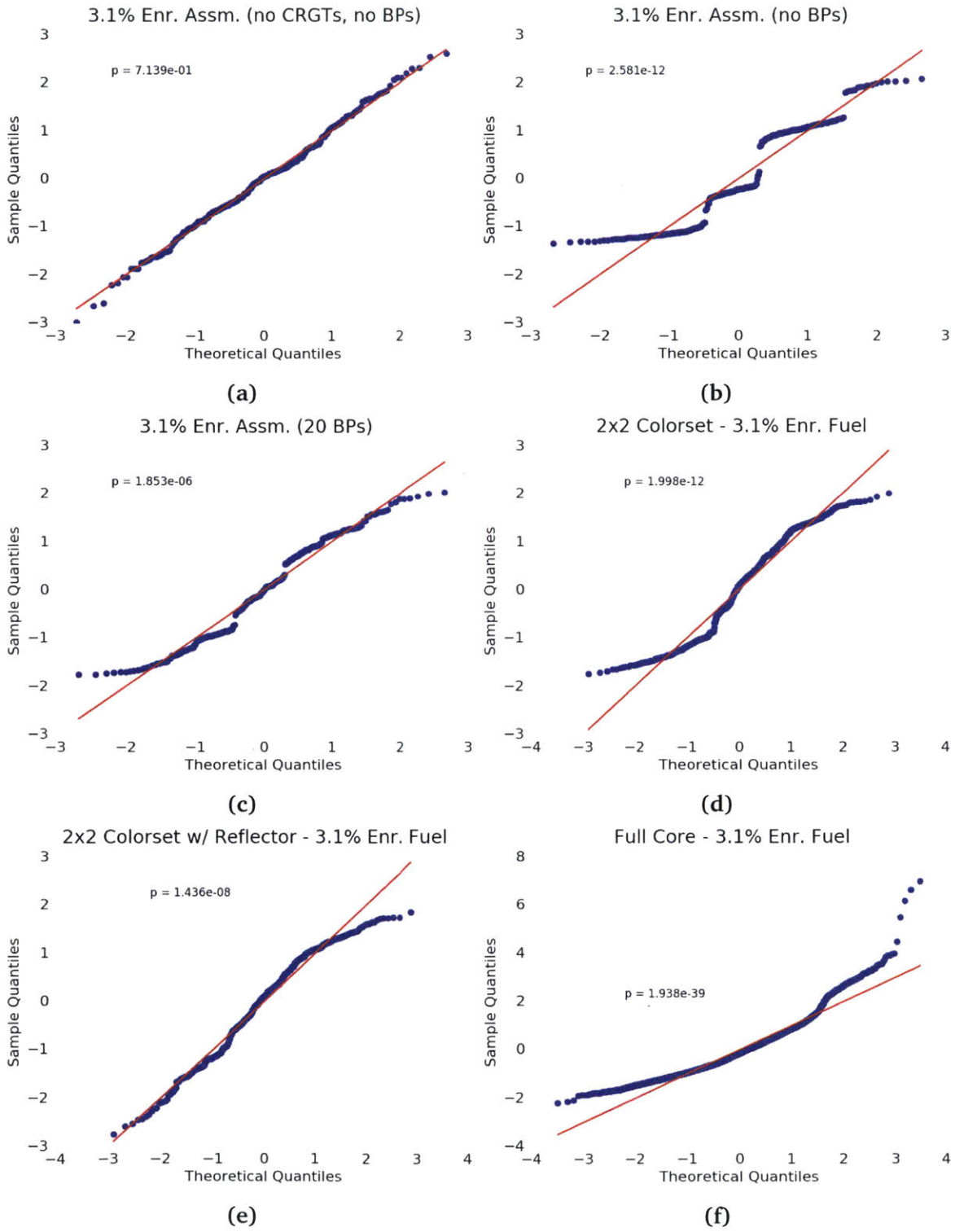


Figure 9-6: Q-Q plots of U-238 capture MGXS (group 1 of 2) for 3.1% enriched fuel.

larger benchmarks, this does not necessarily indicate that the data is more likely to have been drawn from a normal distribution. Indeed, smoothly varying deviations from

$y = x$ exhibited in the larger datasets results in smaller p-values for the colorset and quarter core models. Of particular note, p-values on the order of 10^{-18} and 10^{-39} arise from the data for the 1.6% and 3.1% enriched fuel pins, respectively, for the BEAVRS quarter core model. These results suggest that it is highly unlikely that the data arose from a normally distributed stochastic process.

The clustering of U-238 capture MGXS manifests itself as "shoulders" in Q-Q plots. The Shapiro-Wilks test rejects the null hypothesis that the MGXS data is drawn from a normal distribution for all six heterogeneous benchmarks.

9.1.4.2 U-235 Fission MGXS

The pin-wise microscopic U-235 fission MGXS for 1.6% and 3.1% enriched fuel pins are illustrated with histograms and rug plots in Fig. 9-7 and 9-8, respectively. As was observed in the preceding section for U-238 capture, the datasets for the infinite lattices in Figs. 9-7a and 9-8a appear to be from a normal distribution since they lie very close to the $y = x$ reference line. In addition, the p-values are greater than 0.3, well above commonly used significance levels; thus one would not reject the null hypothesis that the data is from a normal distribution.

The addition of CRGTs leads to a clear deviation from normality in Figs. 9-7b and 9-8b. However, the structure is substantially more complex than the four shoulders exhibited in the U-238 capture MGXS data. As was noted in Sec. 9.1.3.2, the presence of BPs results in many distinct clusters which appear as a highly fragmented, step-like profile in the Q-Q plot in Fig. 9-6c. The high degree of clustering exhibited in the histograms and rug plots for the larger colorset and quarter core benchmarks similarly appears in the corresponding Q-Q plots.

As noted for the U-238 capture MGXS, the p-values for the Shapiro-Wilks tests of the pin-wise U-235 fission MGXS data affirm the deviation from normality seen in the Q-Q plots. Of particular note, p-values on the order of 10^{-38} and 10^{-43} arise from the data for the 1.6% and 3.1% enriched fuel pins, respectively, for the BEAVRS quarter core model. These results suggest that it is highly unlikely that the data arose from a

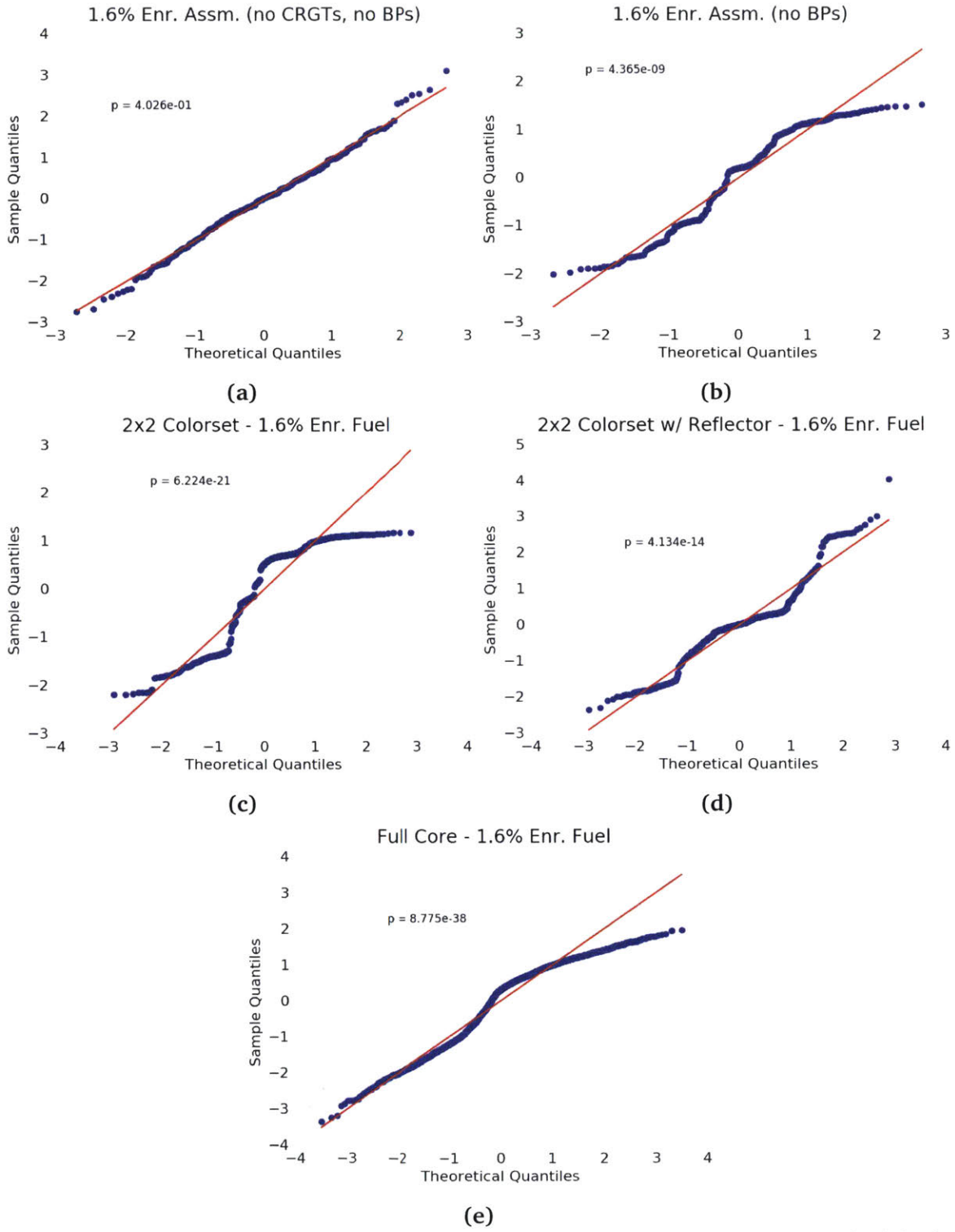


Figure 9-7: Q-Q plots of U-235 fission MGXS (group 2 of 2) for 1.6% enriched fuel.

normally distributed stochastic process.

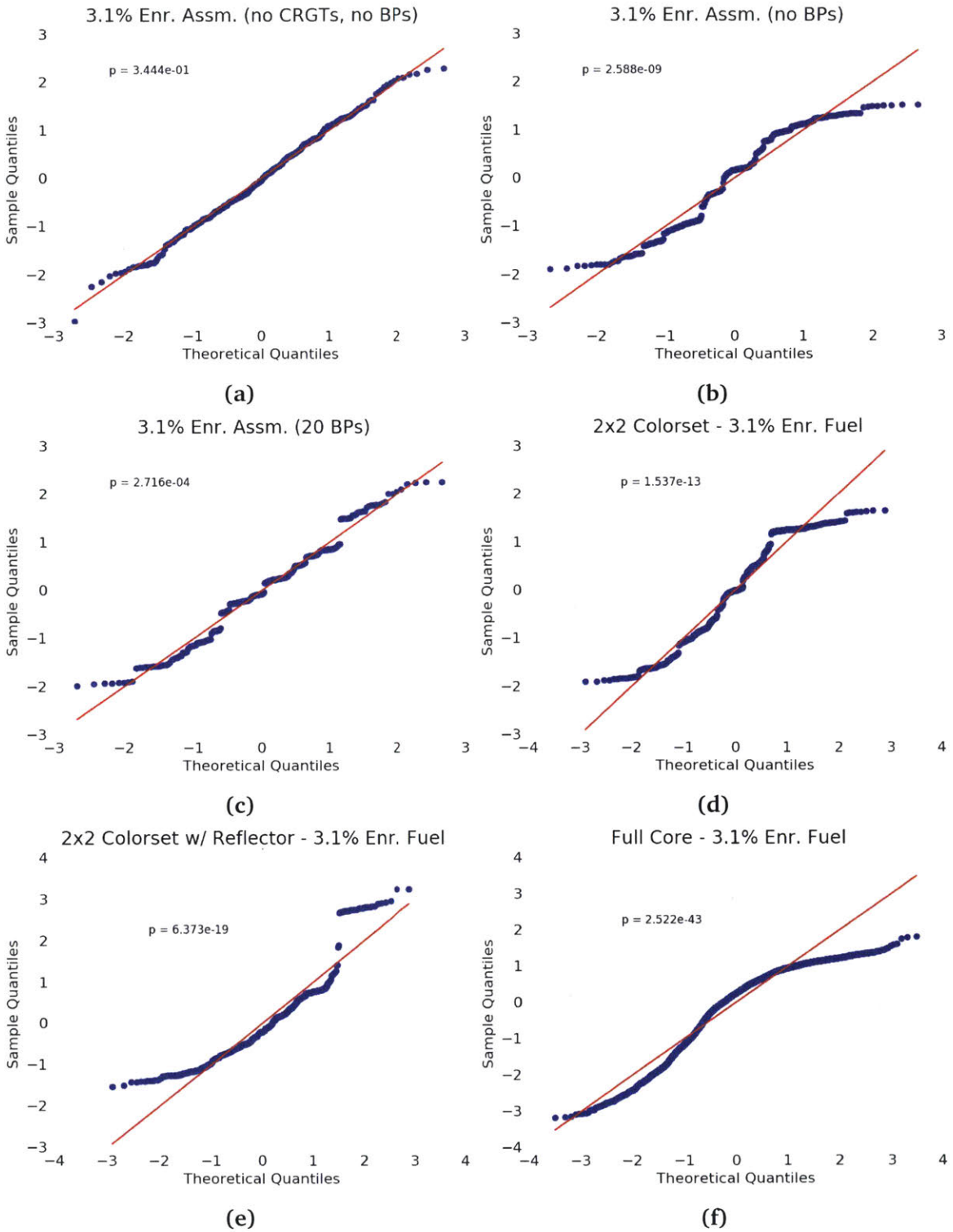


Figure 9-8: Q-Q plots of U-235 fission MGXS (group 2 of 2) for 3.1% enriched fuel.

The higher degree of clustering of pin-wise U-235 fission MGXS than U-238 capture MGXS results in highly fragmented step-like Q-Q plots. The Shapiro-Wilks test once again rejects the null hypothesis that the MGXS data is drawn from a normal distribution for all six heterogeneous benchmarks.

9.2 LNS Spatial Homogenization

The preceding sections quantified and visualized the dispersion and structural clustering of pin-wise MGXS due to spatial self-shielding effects. The degenerate spatial homogenization scheme evaluated in Chap. 8 is able to model the clustering of MGXS by assigning a unique set of MGXS to each fuel pin instance in a core geometry. However, the infinite and null schemes fail to account for MGXS clustering since they each assign a single MGXS to all instances of the same fuel pin type. As evidenced by the results in Chap. 8, it is important to adequately model clusters of pin-wise MGXS in order to accurately predict U-238 capture rate spatial distributions.

This section introduces a new spatial homogenization scheme which uses a deterministic approach to cluster pin-wise MGXS based on an analysis of the core geometry. The approach developed here is akin to geometric templates employed by some commonly used lattice physics codes, such as CASMO [77], to predict which groupings of pins are likely to experience similar spatial self-shielding effects and hence have similar microscopic MGXS. The new scheme is termed *LNS homogenization* since it is predicated upon OpenCG's Local Neighbor Symmetry (LNS) algorithm (see Sec. 4.4.2). The LNS algorithm analyzes the combinatorial geometry (CG) used to represent each benchmark model and groups pins together based on their neighboring spatial zones. The goal of LNS homogenization is to achieve degenerate homogenization's predictive accuracy by representing MGXS clustering, and approach null homogenization's tally convergence by homogenizing MGXS across many pin instances.

9.2.1 Overview

Like the degenerate spatial homogenization scheme (Sec. 8.2.3), a single MC calculation of the complete heterogeneous geometry is used to generate MGXS for all materials. The MGXS are tallied separately for each instance of fissile material zones using OpenMC's distributed cell tallies (see Sec. 4.2.2). The OpenCG LNS algorithm assigns an integral LNS identifier to each unique pin instance in the CG based on an analysis of each pin's neighbors at each level of the CG hierarchy. Pins with like neighboring pins, within assemblies with like neighboring assemblies, will receive the same LNS identifier[56]. The MGXS are averaged across all pin instances with the same unique LNS identifier. For example, all pins adjacent to a single CRGT on one face and fuel pins on all other faces are assigned the same MGXS averaged across the OpenMC distributed cell tallies for all of those pin instances. The OpenCG region differentiation algorithm (see Sec. 4.4.3) is used to build an OpenMOC geometry with unique cells and materials for each fuel pin which mirrors the LNS representation of the CG model. Like the infinite, null and degenerate schemes, spatial self-shielding effects experienced by different non-fissile spatial zones are averaged across the entire geometry for each non-fissile material.

The total number of materials (*i.e.*, MGXS) used to model each benchmark with the LNS homogenization scheme is given in Tab. 9-9. The fuel assemblies with CRGTs and BPs and 2×2 colorset benchmark models are color-coded by material and illustrated in Fig. 9-9 for the LNS homogenization scheme. Likewise, the materials for the quarter core BEAVRS model with LNS homogenization is highlighted in Fig. 9-10.

As quantified in the Tab. 9.3, 9 – 10 unique fissile materials are used for each fuel assembly with LNS, far fewer than the 264 in degenerate homogenization. Consequently, the MC tallies for LNS homogenization will converge more quickly than those for degenerate homogenization. In addition, as indicated by the figures, fuel pin types with neighboring heterogeneities are assigned unique MGXS which generally reflects the clustering of MGXS due to spatial self-shielding effects. As a result, LNS homogenization would be expected to enable more accurate predictions of reaction rate distributions than is possible with the null scheme.

[56]LNS hashes a data structure representing a rotationally invariant form of each pin's neighbors.

Table 9.3: Number of materials modeled with unique MGXS in each heterogeneous benchmark for LNS spatial homogenization.

Benchmark	# Fuel Materials		
	Null/Infinite	LNS	Degenerate
1.6% Assm	1	9	264
3.1% Assm	1	9	264
3.1% Assm w/ 20 BPs	1	10	264
2×2 Colorset	2	19	1,056
2×2 Colorset w/ Reflector	2	29	1,056
BEAVRS Quarter Core	3	495	12,993

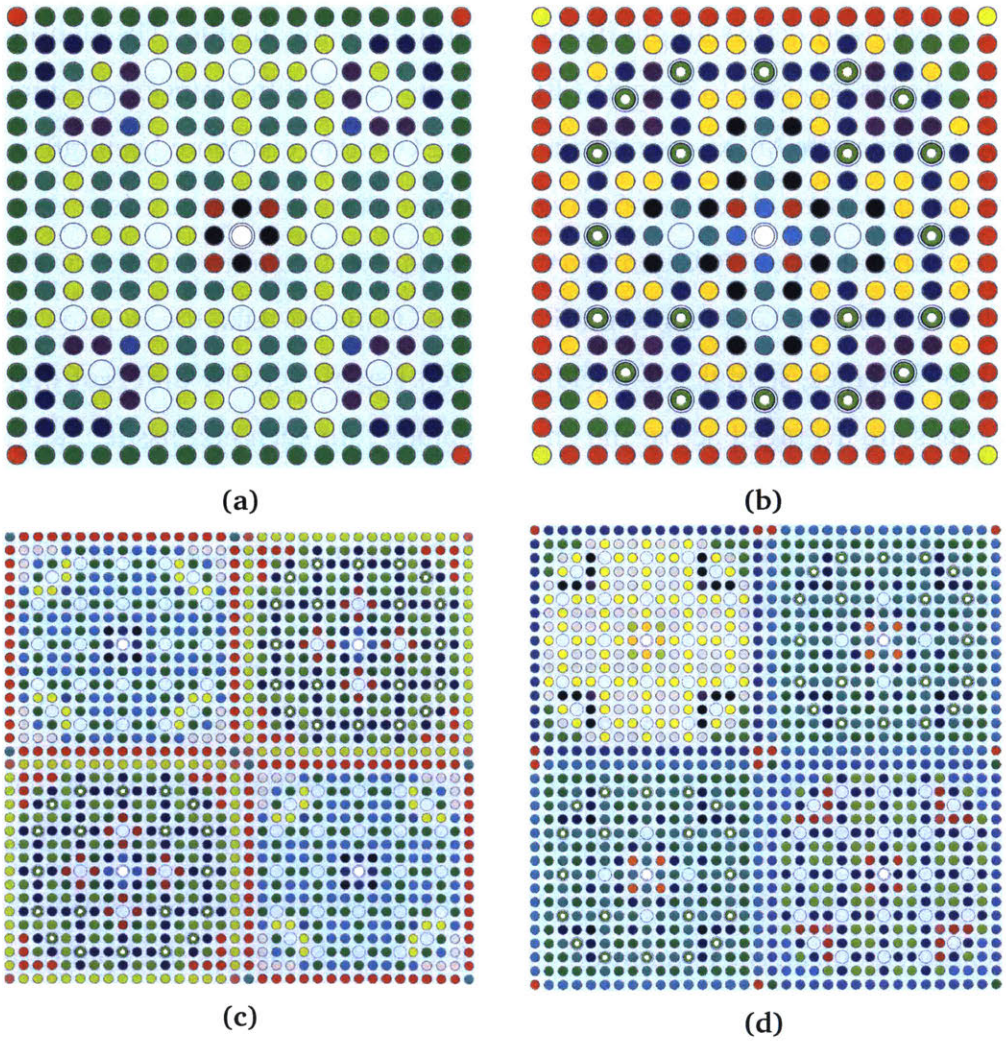


Figure 9-9: OpenMOC materials with LNS spatial homogenization for an assembly with CRGTs (a), an assembly with 20 BPs (b), a 2×2 colorset without (c) and with (d) a reflector. Each uniquely colored material represents a unique set of MGXS.

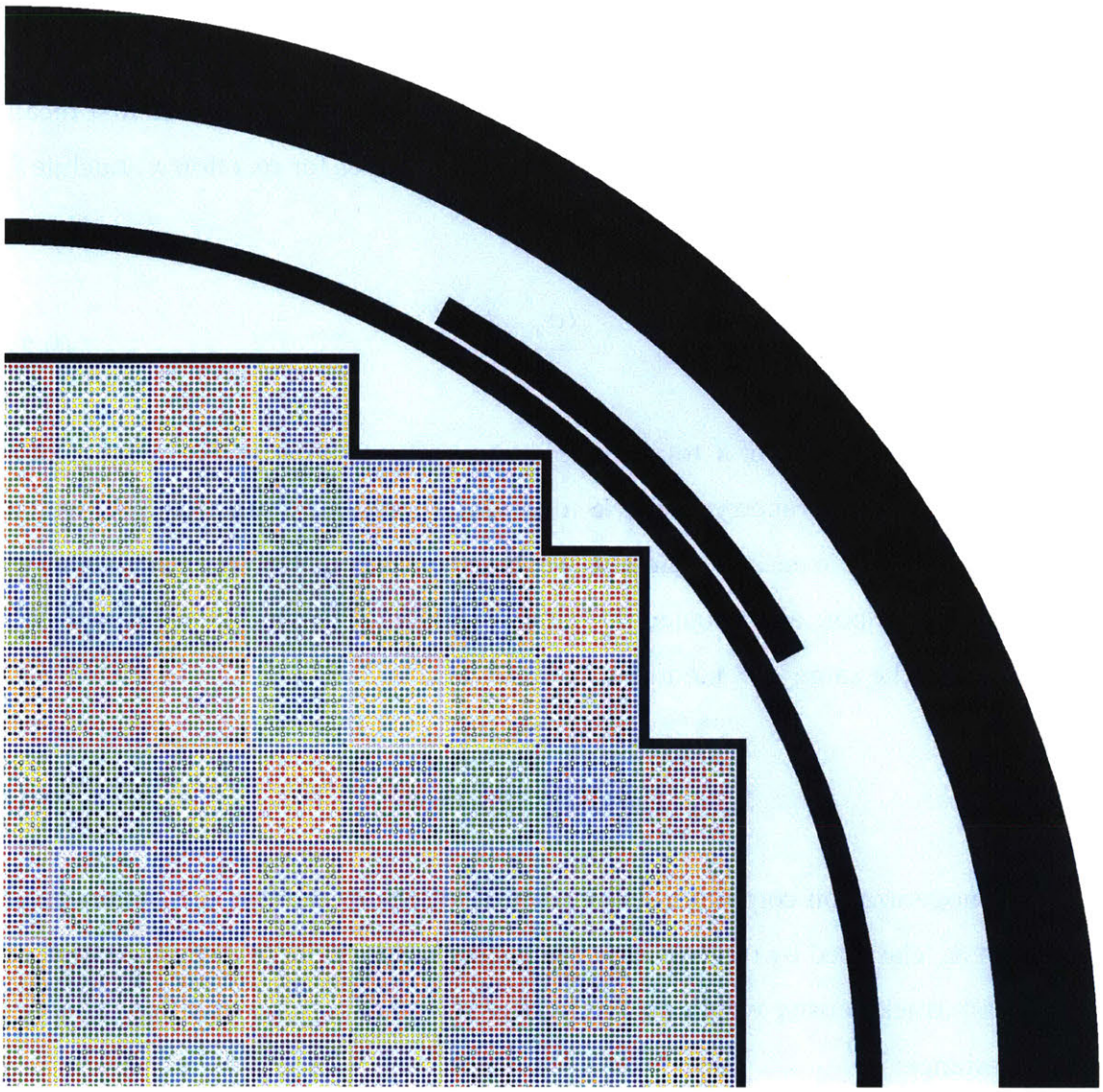


Figure 9-10: OpenMOC materials with LNS spatial homogenization for the 2D quarter core BEAVRS model. Each uniquely colored material represents a unique set of MGXS.

9.2.2 Track Density-Weighted MGXS

The LNS spatially-homogenized MGXS for each group of pin instances are not simply computed as the geometric average of the MGXS in each pin instance. Instead, the total reaction rates and fluxes in each pin instance are first summed together and then divided to compute an average MGXS which is effectively weighted by the relative particle track density in each fuel pin instance. The track density-weighted average MGXS preserve

global reaction rates[57] and are equivalent to defining specialized OpenMC "cell" reaction rate and flux tallies for each group of pin instances with like LNS identifiers. In order to formally express the track density weighted-average approach, it is useful to first recall Eqn. 3.14 for a microscopic MGXS estimated from MC tallies for reaction x, nuclide i, spatial zone k and energy group g:

$$\hat{\sigma}_{x,i,k,g} = \frac{\langle \sigma_{x,i}, \psi \rangle_{k,g}^{t\ell}}{\langle \psi \rangle_{k,g}^{t\ell}} \tag{9.2}$$

In this context, the index k of K total spatial zones refers to a particular instance of a fuel pin within a core geometry. The LNS algorithm represents a function $S(k)$ which assigns an identifier m to each fuel pin instance based on its neighbors (*i.e.*, pin instances with the same neighbors are assigned the same identifier). The set $\mathbb{S}_m$ encapsulates all instances k with the same LNS identifier:

$$\mathbb{S}_m = \{1 \leq k \leq K : S(k) = m\} \tag{9.3}$$

LNS homogenization computes a single set of MGXS for the fuel pin instances in each set $k \in \mathbb{S}_m$ classified by the LNS algorithm. This is equivalent to a specialization of Eqn. 9.2 with track density-weighted averages of the reaction rates and flux tallies in each pin instance:

$$\hat{\sigma}_{x,i,m,g} = \frac{\sum_{k=1}^{K} \mathbb{1}_{\mathbb{S}_m}(k) \langle \sigma_{x,i}, \psi \rangle_{k,g}^{t\ell}}{\sum_{k=1}^{K} \mathbb{1}_{\mathbb{S}_m}(k) \langle \psi \rangle_{k,g}^{t\ell}} \tag{9.4}$$

where the indicator function $\mathbb{1}_{\mathbb{S}_m}(k)$ is equal to 1 if $k \in \mathbb{S}_m$ and 0 otherwise. The track density-weighted average is similarly applied to the MC tallies for each type of MGXS, including scattering matrices and the fission spectrum.

[57] A simple geometric average of MGXS across fuel pins will not preserve global reaction rates.

9.2.3 Potential Shortcomings

Upon further inspection, it is clear from Figs. 9-9 and 9-10 that there are some notable shortcomings to the LNS scheme. For example, the pins along the inter-assembly and assembly-reflector interfaces in the 2×2 colorset and quarter core BEAVRS models in Fig. 9-9c to 9-9d and 9-10 are treated the same (*i.e.*, with the same MGXS). As a result, LNS may result in poor reaction rate predictions for these pins, as will be quantified in Sec. 9.3. It is possible that the LNS algorithm could be specialized in various ways to differentiate between the pin types on the outer edge of each assembly. However, such customizations would not be reactor agnostic and would be challenging to implement and generalize.

Furthermore, the relative number of LNS materials does not scale with the total number of fuel pins. In particular, there are 26 – 29× fewer materials with the LNS scheme as compared to the degenerate scheme for the individual fuel assembly benchmarks as well as the quarter core BEAVRS model[58]. It is likely that LNS uses more materials than necessary to capture MGXS clustering for large geometries, which will diminish its relative accelerated convergence with respect to degenerate homogenization.

LNS spatial homogenization applies a "geometric template" to homogenize MGXS for pins with similar neighboring spatial zones. The scheme aims to accelerate the MC tally convergence while capturing spatial self-shielding effects in clustered MGXS. However, the number of materials scales poorly with the size and complexity of the core geometry. Furthermore, the algorithm fails to distinguish pins with very different spatial self-shielding effects at inter-assembly and assembly-reflector interfaces.

[58]This reflects the fact that neighboring assemblies are accounted for in the LNS algorithm. The fuel assemblies in the quarter core BEAVRS model do not exhibit any neighboring symmetries, and as a result, no two assemblies are assigned the same set of pin-wise MGXS.

9.3 Multi-Group Results with LNS

Each of the six benchmarks was modeled with OpenMOC using MGXS generated by the LNS spatial homogenization scheme. Each of the six heterogeneous benchmarks was modeled with 2-, 8- and 70-group MGXS using the same OpenMOC runtime parameters as those used in Chap. 8 for infinite, null and degenerate homogenization. The eigenvalues and pin-wise fission and U-238 capture rates computed by OpenMOC are compared to the reference OpenMC solutions in Secs. 9.3.1, 9.3.2 and 9.3.3, respectively.

9.3.1 Eigenvalues

The OpenMOC eigenvalues were compared to the reference OpenMC eigenvalues from Tab. 7.1. The eigenvalue bias $\Delta\rho$ was computed from Eqn. 5.1 in units of pcm. The bias is listed for each benchmark and energy group structure in Tab. 9.4. The same trends highlighted in Sec. 8.4.1 observed from the null and degenerate biases in Tab. 8.4 remain true for LNS spatial homogenization. In fact, the LNS eigenvalues are within 10 pcm of those computed with both null and degenerate homogenization with 8 or more groups for all benchmarks. As previously noted in Sec. 8.4.1, this is expected since the MGXS for the null, degenerate and LNS schemes are homogenized from the same flux and should preserve globally-integrated reaction rates. Hence, LNS homogenization is not expected to improve OpenMOC's eigenvalue predictions.

Table 9.4: OpenMOC eigenvalue bias $\Delta\rho$ for heterogeneous benchmarks with LNS homogenization and varying energy group structures.

Benchmark	$\Delta\rho$ [pcm]		
	2-Group	8-Group	70-Group
1.6% Assm	62	-72	-161
3.1% Assm	98	-80	-202
3.1% Assm w/ 20 BPs	-158	-161	-248
2×2 Colorset	12	-93	-194
2×2 Colorset w/ Reflector	1797	481	-138
BEAVRS Full Core	2168	401	-129

The OpenMOC eigenvalues for LNS homogenization are consistent with the null and degenerate schemes to within 10 pcm for eight or more groups due to global reaction rate preservation.

9.3.2 Fission Rates

The OpenMOC energy-integrated pin-wise fission rates were compared to the reference OpenMC fission rates for LNS homogenization. The percent relative errors for each pin's fission rates were computed and the maximum and mean errors are listed for each benchmark and energy group structure in Tab. 9.5, respectively. In particular, the maximum errors are the maximum of the absolute values of the errors along with the appropriate sign, while the mean errors are the averages of the absolute error magnitudes. The results in Tab. 9.5 can be compared to the corresponding data for infinite, null and degenerate homogenization in Tabs. 8.5 and 8.6. The 70-group degenerate results are reproduced in the Tab. 9.5 to simplify comparison with LNS. No heatmaps for the fission rate errors are presented since the spatial homogenization scheme has little visible impact on the spatial distribution of errors.

One of the key findings in Chap. 8 was that degenerate homogenization did not result in a substantial reduction in the spatial distribution of fission rate errors with respect to OpenMC. This result indicates that an accurate model of MGXS clustering *is not* needed to accurately predict fission rate spatial distributions. Hence, LNS spatial homogenization would be expected to produce similar results to degenerate homogenization. Indeed, the fission rate errors for 8 and 70 groups with LNS homogenization are very nearly the same as those for degenerate homogenization for the three individual fuel assemblies and the 2×2 colorset. Although the errors are 0.1 – 0.2% worse for the 2×2 colorset with a reflector and the quarter core BEAVRS model, they remain slightly below those for null homogenization. This is due to the fact that the MGXS for the fuel pins near the assembly-reflector interface are homogenized along with those at the inter-assembly interfaces for LNS spatial homogenization, rather than separately homogenized due to

the additional moderation provided by the reflector.

Table 9.5: OpenMOC fission rate percent relative errors for heterogeneous benchmarks with LNS spatial homogenization and varying energy group structures.

Benchmark	Metric	Error [%]			
		LNS			Degenerate
		2-Group	8-Group	70-Group	70-Group
1.6% Assm	Max	2.095	0.728	0.314	0.315
	Mean	0.713	0.240	0.078	0.079
3.1% Assm	Max	2.382	0.827	0.371	0.372
	Mean	0.834	0.288	0.086	0.087
3.1% Assm w/ 20 BPs	Max	-2.030	-0.685	0.320	0.331
	Mean	0.724	0.215	0.085	0.086
2×2 Colorset	Max	-5.499	-1.412	0.405	0.427
	Mean	2.941	0.701	0.118	0.120
2×2 Colorset w/ Reflector	Max	-15.785	-2.976	0.709	0.602
	Mean	5.169	1.076	0.155	0.138
BEAVRS Full Core	Max	-85.579	-32.106	1.805	1.728
	Mean	39.542	10.382	0.296	0.336

LNS spatial homogenization performs as well as or slightly better than degenerate homogenization for simple benchmarks, but fails to model the impact of spatial self-shielding effects on pin-wise fission rates in more complicated geometries with inter-assembly and assembly-reflector interfaces.

9.3.3 U-238 Capture Rate Distributions

The OpenMOC energy-integrated pin-wise U-238 capture rates were compared to the reference OpenMC capture rates for LNS homogenization. The percent relative errors for each pin's capture rates were computed and the maximum and mean errors are listed for each benchmark and energy group structure in Tab. 9.6, respectively. In particular, the maximum errors are the maximum of the absolute values of the errors along with the appropriate sign, while the mean errors are the averages of the absolute error magnitudes. The results in Tab. 9.6 can be compared to the corresponding data

for infinite, null and degenerate homogenization in Tabs. 8.7 and 8.8. The 70-group degenerate results are reproduced in Tab. 9.6 to simplify comparison with LNS.

Table 9.6: OpenMOC U-238 capture rate percent relative errors for heterogeneous benchmarks with LNS spatial homogenization and varying energy group structures.

Benchmark	Metric	Error [%]			
		LNS			Degenerate
		2-Group	8-Group	70-Group	70-Group
1.6% Assm	Max	1.091	0.372	0.290	0.386
	Mean	0.390	0.084	0.076	0.086
3.1% Assm	Max	0.969	0.375	0.228	0.326
	Mean	0.351	0.090	0.076	0.087
3.1% Assm w/ 20 BPs	Max	2.005	0.548	0.249	0.311
	Mean	0.509	0.148	0.073	0.089
2×2 Colorset	Max	-2.753	-0.832	0.439	0.615
	Mean	1.516	0.136	0.115	0.154
2×2 Colorset w/ Reflector	Max	10.201	3.031	-1.964	-0.783
	Mean	3.482	0.604	0.236	0.165
BEAVRS Quarter Core	Max	-84.556	-30.671	-2.854	-2.067
	Mean	39.308	10.017	0.304	0.345

One of the key findings in Chap. 8 was that degenerate homogenization enabled substantial reductions of the spatial distribution of U-238 capture rate errors with respect to OpenMC. In contrast to the fission rates, this result indicates that accurate model of MGXS clustering *is* needed to accurately predict U-238 capture rate spatial distributions. Hence, LNS spatial homogenization would be expected to enable more accurate predictions than null homogenization, and to potentially approach the accuracy of degenerate homogenization. Upon investigation, the max and mean U-238 capture rate errors for 70 groups with LNS homogenization are 0.1 – 0.2% and 0.01% less, respectively, than degenerate homogenization for the three individual fuel assemblies and the 2×2 colorset (relative reductions of 10 – 30%).

However, the errors for the 2×2 colorset with a reflector are very nearly the same as those for null homogenization (about 2.5× larger errors than degenerate homogenization). As was previously noted for the fission rate errors, the MGXS for the fuel pins

near the assembly-reflector interface are not separately homogenized and thus fail to account for the additional moderation from the reflector. Perhaps the most surprising observation is that the max and mean errors for the quarter core BEAVRS model are actually 1.5% and 0.1% less, respectively than those for both null and degenerate homogenization. This is notable since degenerate homogenization actually has a slightly larger maximum error (8.296%) than null homogenization (8.076%), most likely due to the relatively larger tally uncertainties of the pin-wise MGXS for the quarter core model as compared to the five smaller benchmarks. Based on these results, it appears that LNS homogenization performs as intended and "de-noises" the statistical uncertainties by averaging the MGXS across clusters with similar spatial self-shielding effects.

The spatial distributions of capture rate errors are plotted as heatmaps for each benchmark in Figs. 9-11 to 9-18. These figures illustrate the capture rate errors for 8 and 70 energy group structures for the assembly and colorset benchmarks, and errors for 70 groups for the quarter core BEAVRS model. The figures are corollaries to those comparing the infinite, null and degenerate schemes in Figs. 8-10 to 8-15. In addition, the U-238 capture absolute errors for the 2×2 colorsets are illustrated in Figs. C-13 and C-14, and for the full core in Fig. C-15 to C-17 in App. C.3.

The heatmaps illustrate very nearly the same error distributions for degenerate and LNS spatial homogenization for the individual fuel assemblies and 2×2 colorset without a reflector, but systematic deviations for the reflected colorset and quarter core BEAVRS model. As previously noted in Sec. 8.4.3, the null scheme exhibited the largest errors for pins near CRGTs and along the inter-assembly and assembly-reflector interfaces, while the degenerate scheme produced a very nearly even error distribution across the pins in the reflected colorset (in 70 groups). The LNS scheme largely "smooths" the error distribution for pins, but exhibits large errors for the single outermost row of pins adjacent to the reflector, and to a lesser extent, the pins along the inter-assembly interfaces. More specifically, the U-238 capture rates are under-predicted near the reflector and over-predicted for interior pins. This result is indicative of the collective homogenization of all pins along the exterior of each assembly irregardless of the neighboring spatial zones. The flux at U-238 capture resonance energies is more shielded for pins adjacent

to the reflector than for interior pins, resulting in larger U-238 capture MGXS. LNS homogenizes the larger MGXS for pins along the reflector with the smaller capture MGXS for interior pins, which leads to the respective under- and over-prediction for the outermost and interior pins.

A couple of key conclusions can be drawn from these results. First, LNS effectively models MGXS clustering for pins in the interior of each fuel assembly (*e.g.*, pins adjacent to CRGTs and/or BPs). However, the pins along the outer edges of each assembly are homogenized in a way that does not appropriately account for the spatial self-shielding effects of the neighboring materials zones. Nevertheless, track density-weighted spatial homogenization does ensure that the errors improve the most for those pins with the largest reaction rates, as illustrated by the absolute errors in App. C.3.

LNS spatial homogenization reduces the U-238 capture rate errors by 10 – 30% with respect to degenerate homogenization for the single assembly and periodic colorset benchmarks. However, the scheme does not improve the errors for the reflected colorset with respect to null homogenization since it fails to model the impact of inter-assembly and assembly-reflector interfaces.

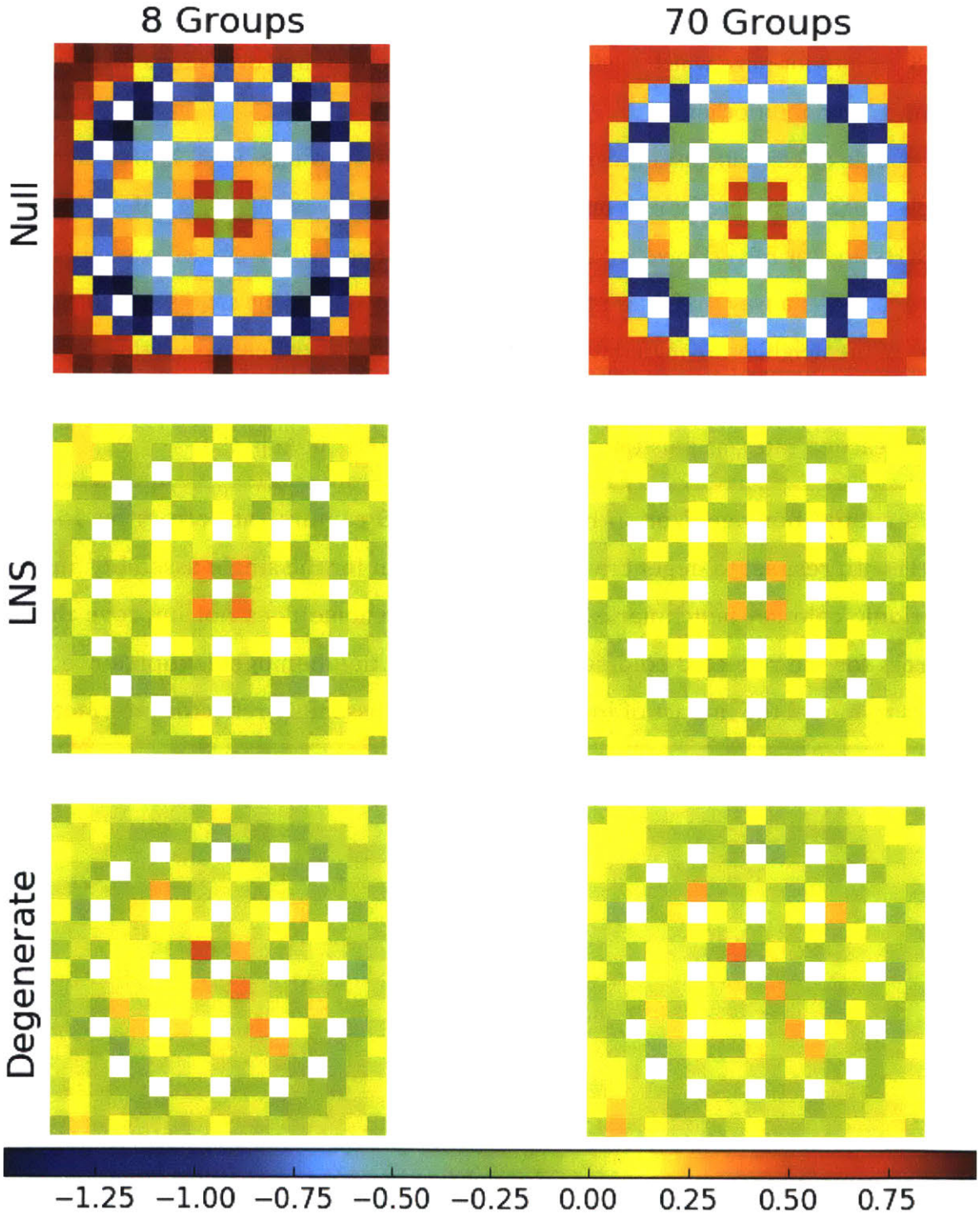


Figure 9-11: U-238 capture rate percent relative errors errors for a 1.6% enriched assembly with null, LNS and degenerate spatial homogenization.

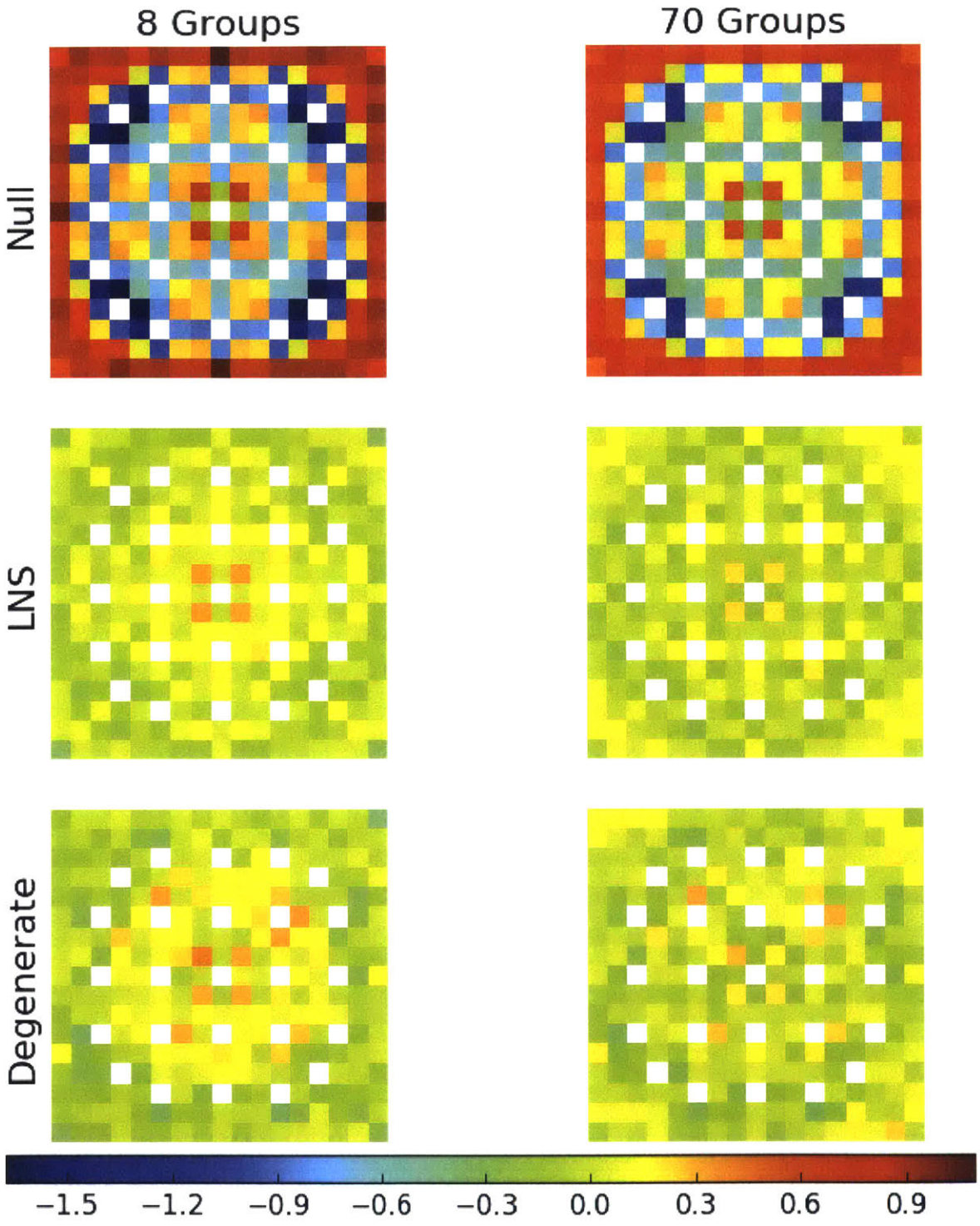


Figure 9-12: U-238 capture rate percent relative errors errors for a 3.1% enriched assembly with null, LNS and degenerate spatial homogenization.

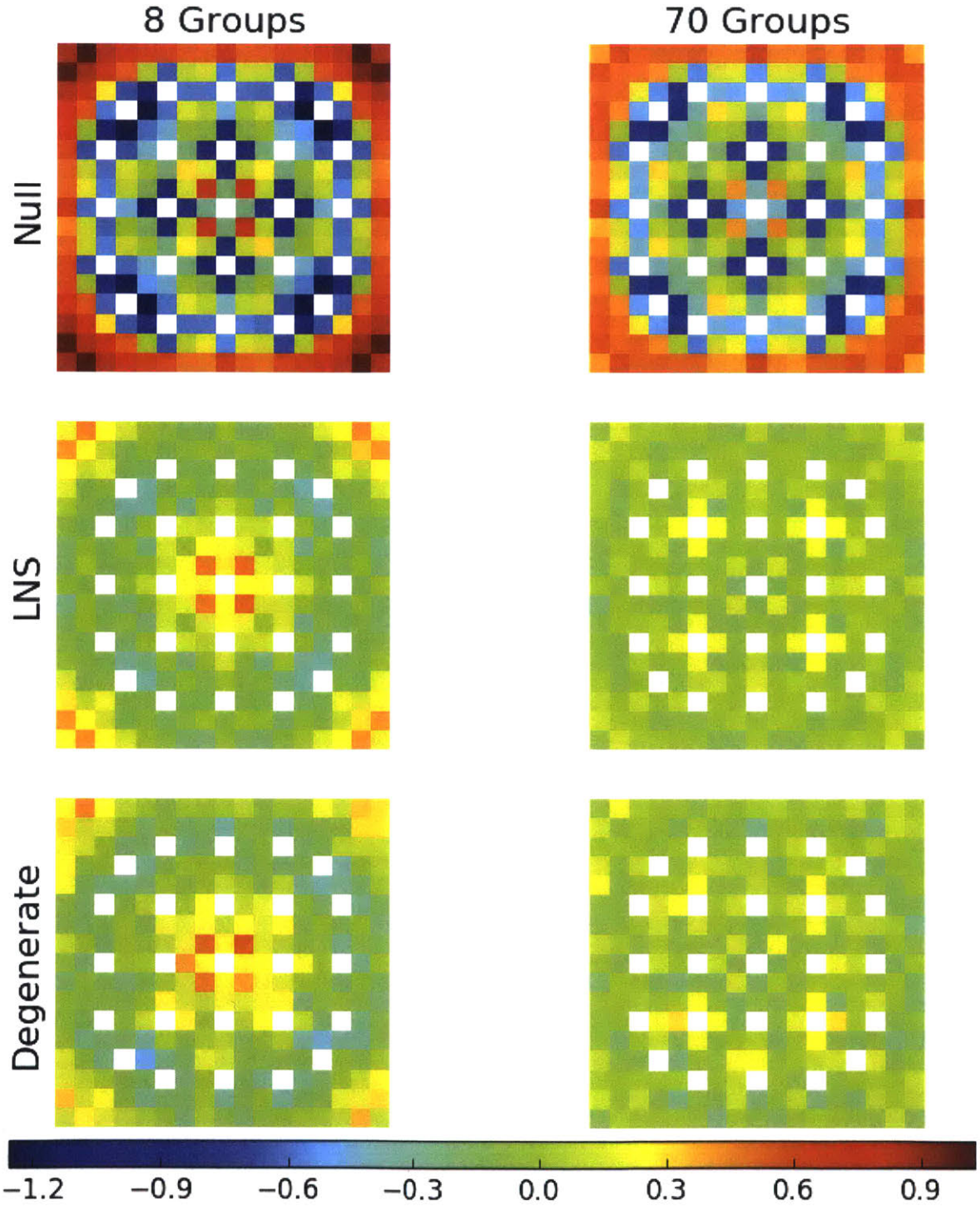


Figure 9-13: U-238 capture rate percent relative errors errors for a 3.1% enriched assembly with 20 BPs with null, LNS and degenerate spatial homogenization.

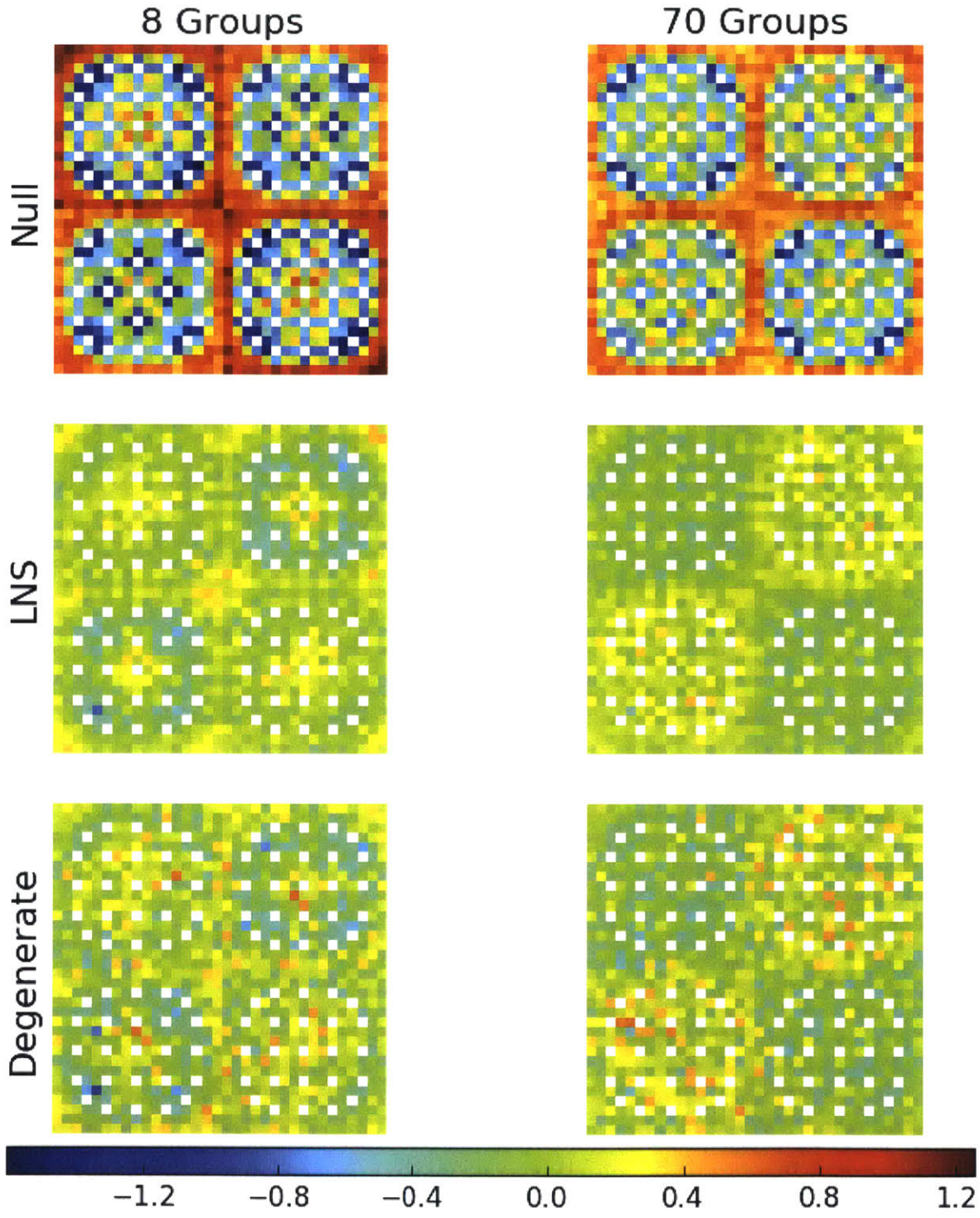


Figure 9-14: U-238 capture rate percent relative errors errors for a 2×2 colorset with null, LNS and degenerate spatial homogenization.

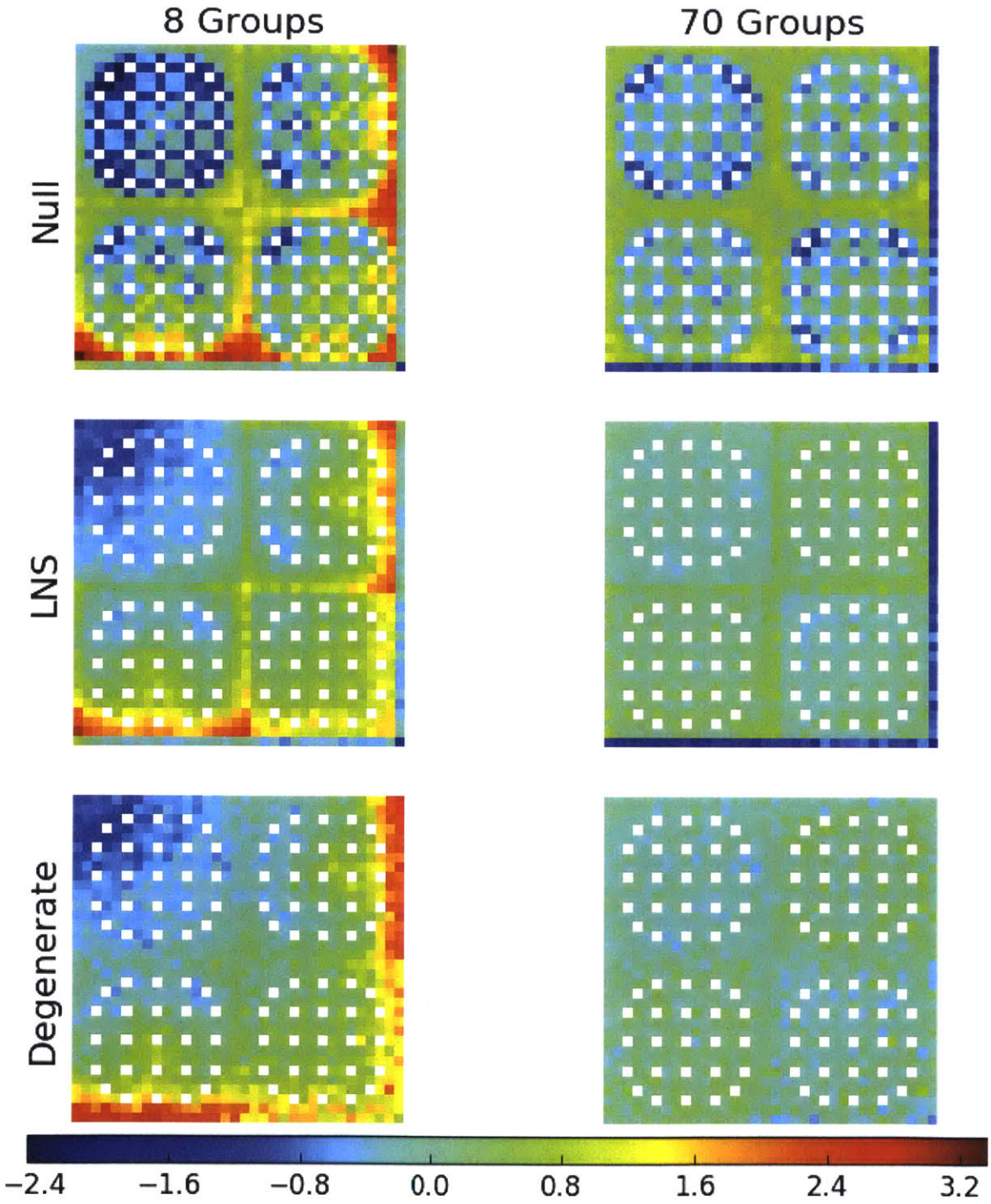


Figure 9-15: U-238 capture percent relative errors rate errors for a 2×2 colorset with null, LNS and degenerate spatial homogenization.

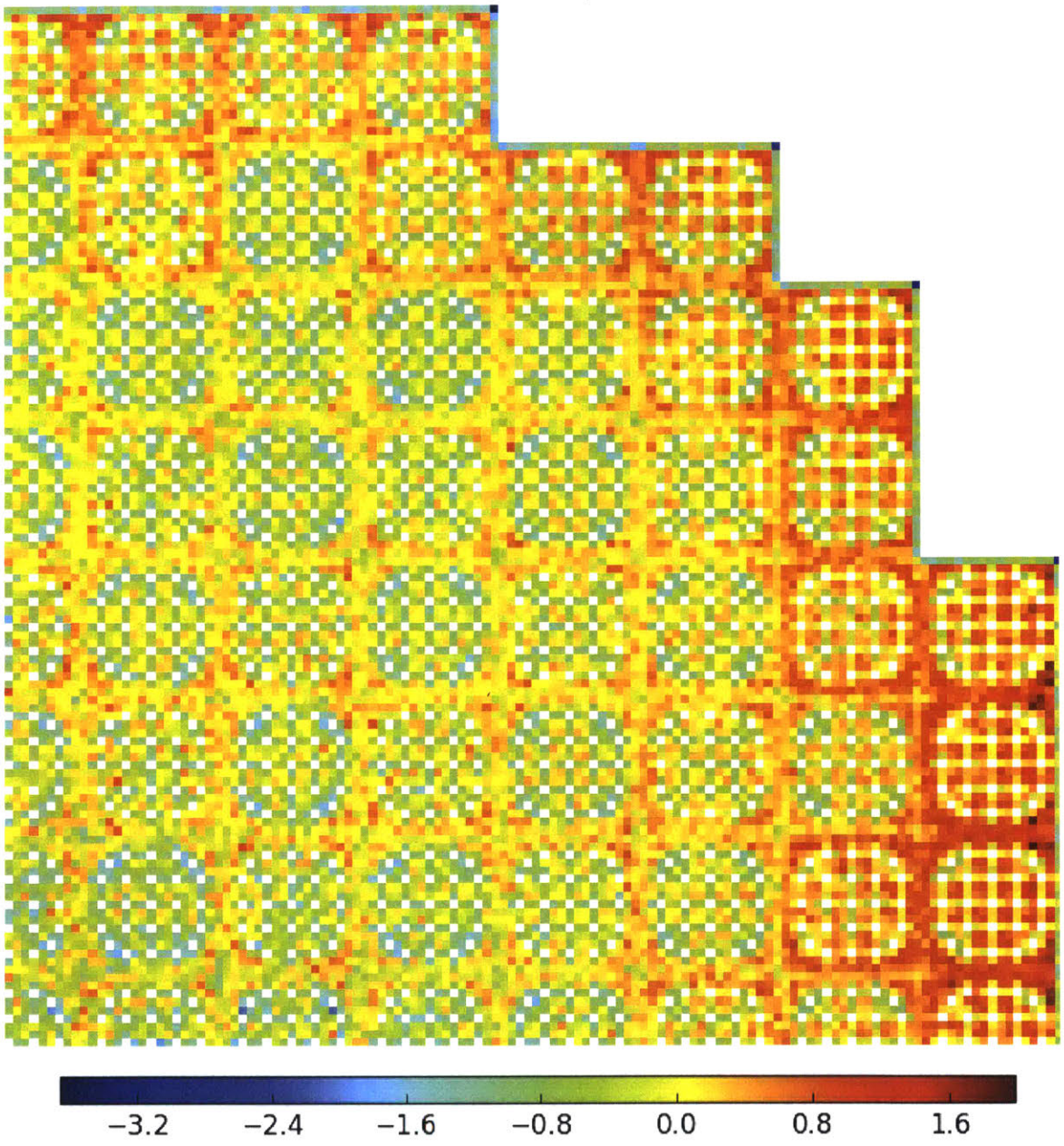


Figure 9-16: U-238 capture rate percent relative errors for the 2D quarter core BEAVRS model with null spatial homogenization.

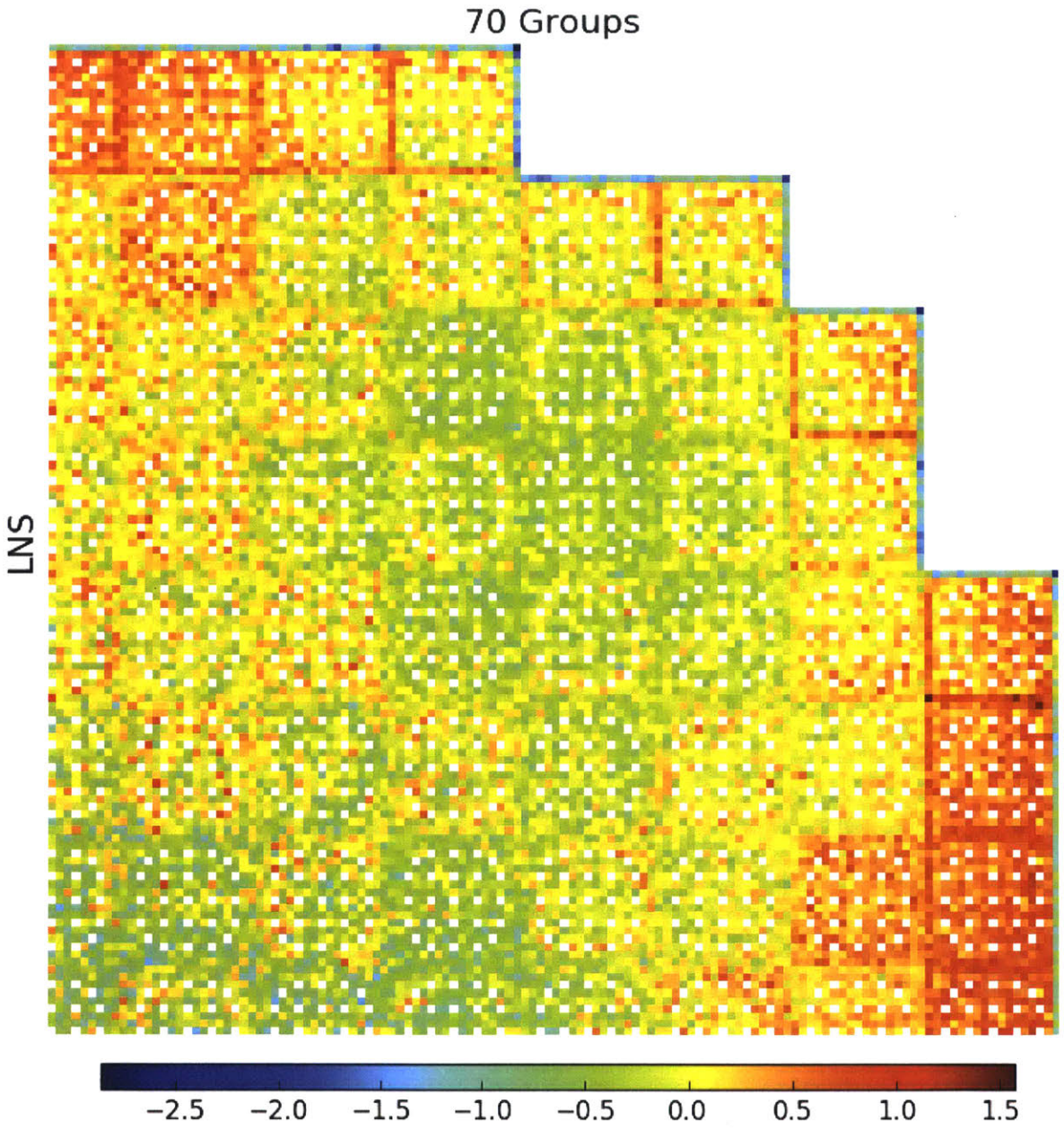


Figure 9-17: U-238 capture rate percent relative errors for the 2D quarter core BEAVRS model with LNS spatial homogenization.

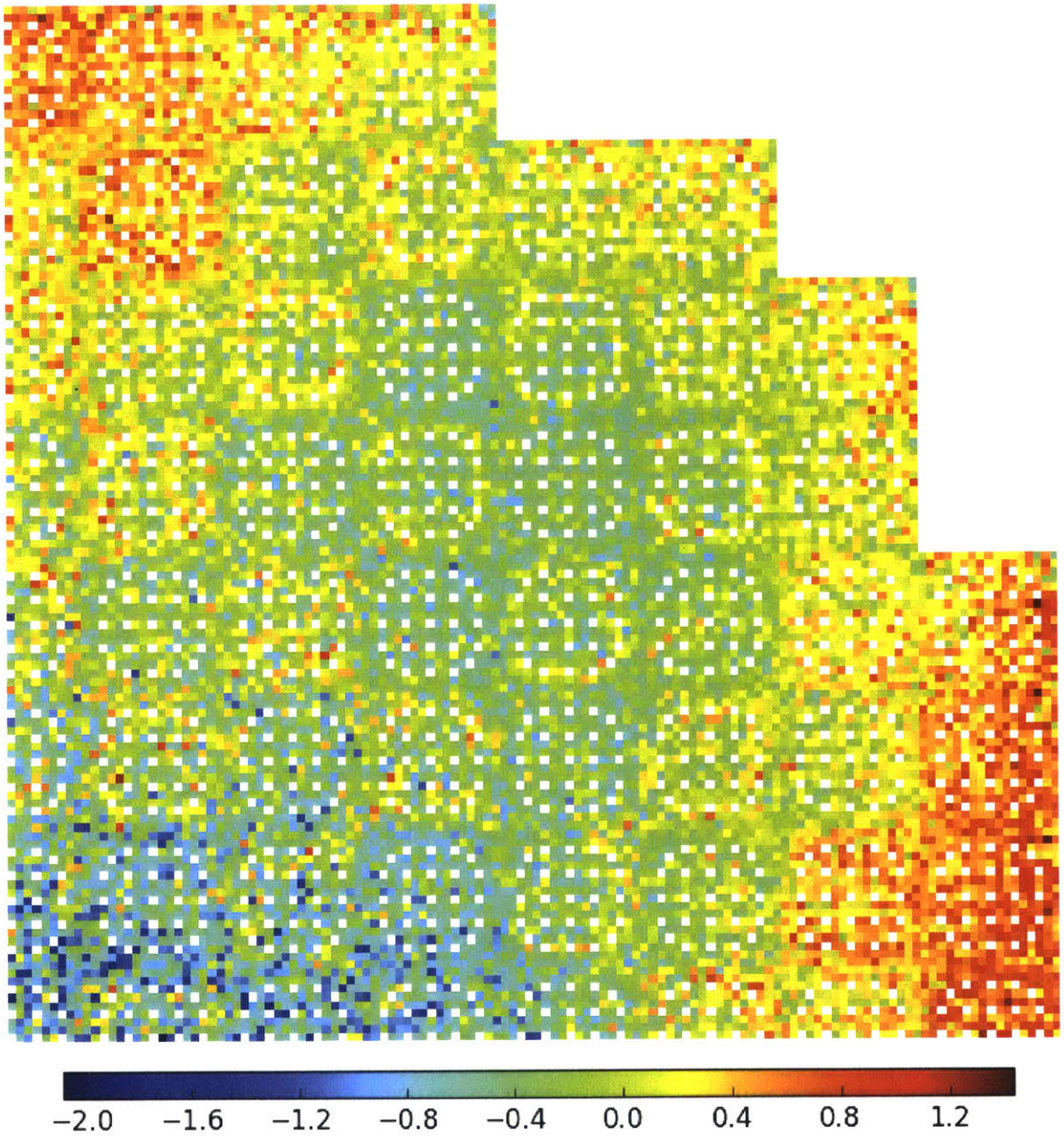


Figure 9-18: U-238 capture rate percent relative errors for the 2D quarter core BEAVRS model with degenerate spatial homogenization.

9.4 MGXS Uncertainties and Convergence Rates

As discussed at the beginning of this chapter, there are two key objectives to MGXS clustering – to simultaneously approach the accuracy of degenerate homogenization and the convergence of null homogenization. The former objective was previously quantified in the context of LNS spatial homogenization; the latter objective is the subject of this section. In particular, the particle track density within each spatially-homogenized tally volume must be increased in order to converge the statistical uncertainties faster than is possible with degenerate homogenization.

LNS homogenization attempts to accomplish this by predicting which fuel pins have similar MGXS and homogenizing their tally volumes into a single set of MGXS for each set of pins with unique LNS identifiers. This section quantifies the accelerated convergence for the LNS scheme and contrasts it with the null and degenerate schemes. A series of OpenMC simulations were performed for each benchmark with 10,000 batches of 100,000 particles per batch for a total of 10^9 particle histories, with cumulative tally results stored every 20 batches. Stationarity of the fission source was obtained with 100 inactive batches for the assembly and colorset benchmarks; 200 inactive batches were used for the quarter core BEAVRS model. Sec. 9.4.1 discusses the expected difference in statistical uncertainties and convergence rates for MGXS generated with null, degenerate and LNS spatial homogenization. Secs. 9.4.2 and 9.4.3 present the empirical batchwise evolution of the statistical uncertainties and percent deviation for pin-wise MGXS computed using null, degenerate and LNS homogenization, respectively[59].

9.4.1 Theoretical Considerations

This section highlights a few theoretical considerations which should be understood when analyzing the empirical data presented in the following sections. Sec. 9.4.1.1 introduces a model to predict the reduction of statistical uncertainty that can be expected with the use of LNS spatial homogenization. Sec. 9.4.1.2 makes a few comments about

[59]The interested reader is referred to the work by Nelson [37] which performs a more detailed and systematic evaluation of the convergence of MGXS – most notably, scattering and fission production matrices – which is the beyond the scope of this thesis.

the ideal and expected convergence rate of MGXS generated from MC tallies.

9.4.1.1 Relative Statistical Uncertainties

This section introduces a simple mathematical relation to predict the relative uncertainties for degenerate and LNS spatial homogenization for a given number of MC particle histories[60]. It should be recalled from Tab. 9.3 that the LNS scheme reduced the number of materials by >25× with respect to degenerate homogenization. As a result, the particle track density within each LNS tally volume increased according to the number of fuel pins homogenized for each unique LNS identifier m, respectively. The expected (or ideal) reduction of the statistical uncertainties with LNS homogenization may be derived from this with a few key assumptions.

First, assume the microscopic MGXS of the fuel pins $k \in \mathbb{S}_m$ with LNS identifier m are independent and identically distributed (i.i.d.) samples, each drawn from a corresponding normal distribution $\mathcal{N}(\mu_k, \sigma_m^2)$[61]. The sampling distributions for the MGXS in each fuel pin instance k may have different means μ_k, but are assumed to have the same standard deviation σ_m for each LNS identifier[62]. The standard deviations $\sigma_{\hat{\sigma}_{x,i,m,g}}$ of the homogenized microscopic MGXS $\hat{\sigma}_{x,i,m,g}$ (Eqn. 9.4) for each LNS set $\mathbb{S}_m$ are then inversely proportional to the square root of the number of pins in each set:

$$\sigma_{\hat{\sigma}_{x,i,m,g}} \approx \frac{\sigma_m}{\sqrt{|\mathbb{S}_m|}} \tag{9.5}$$

The factor $\sigma_m/\sqrt{|\mathbb{S}_m|}$ is the expected downward "shift" of the standard deviation for LNS homogenized MGXS with respect to the degenerate MGXS. The magnitude of the shift depends on the standard deviation σ_m of the tallied MGXS for each pin comprising the LNS set $\mathbb{S}_m$, as well as the number of fuel pins $|\mathbb{S}_m|$ assigned to the set. The former

[60] This analysis is relevant for any scheme which uses track density-weighting to homogenize MGXS tallied over distinct spatial volumes (*e.g.*, fuel pins). This approach will be used to evaluate the unsupervised statistical clustering methodology for spatial homogenization introduced in Chaps. 10 and 11.

[61] The symbol σ for the standard deviation should not be misconstrued for the microscopic cross section.

[62] The standard deviations are only the same if the track density within each fuel pin instance i for each volume of phase space $(\mathbf{r}, \mathbf{\Omega}, E)$ is identical. In this case, the standard deviations of the tallied pin-wise MGXS would be identical. While this approximation is never true in practice, Tab. 9.1 indicates that the standard deviations for pin-wise MGXS are only within a few multiples of one another for each of the six benchmarks (*i.e.*, the standard deviations are within a narrowly dispersed "band").

is roughly equivalent to the tally uncertainties for the pin-wise MGXS for degenerate homogenization and is controlled by the number of particle histories simulated for a given benchmark. The latter depends on the number of fuel pins assigned to each LNS set, which may not be evenly distributed. For example, the individual fuel assembly with CRGTs (see Fig. 9-9a) has $4 \leq |\mathbb{S}_m| \leq 64$; in other words, the number of fuel pins assigned unique LNS identifiers varies widely. As a result, the downward shift for the standard deviations of the MGXS for each LNS cluster will vary accordingly. For example, the standard deviations of those pins facially adjacent to a single CRGTs in the assembly benchmarks without BPs ($|\mathbb{S}_m| = 64$) should be reduced by approximately $1/8$ with respect to the standard deviations of the MGXS for each of the individual pins. The standard deviations of the pins adjacent to the central instrument tube ($|\mathbb{S}_m| = 4$) will only be reduced by approximately $1/2$ since there are far fewer pins within this LNS set.

As a result of the uneven assignment of pins to LNS sets, the relation in Eqn. 9.5 is only useful to compare the relative uncertainties for the MGXS in a single LNS set to those of the individual pins which comprise the corresponding set. It should not be used to predict the relative uncertainties for the population of MGXS generated from LNS homogenization, as these cannot be generally characterized as an ensemble. Lastly, the relation may be applied to predict the relative uncertainties between null and degenerate homogenization, as is done in Sec. 9.4.2 and benchmarked to empirical data.

9.4.1.2 MGXS Convergence Rates

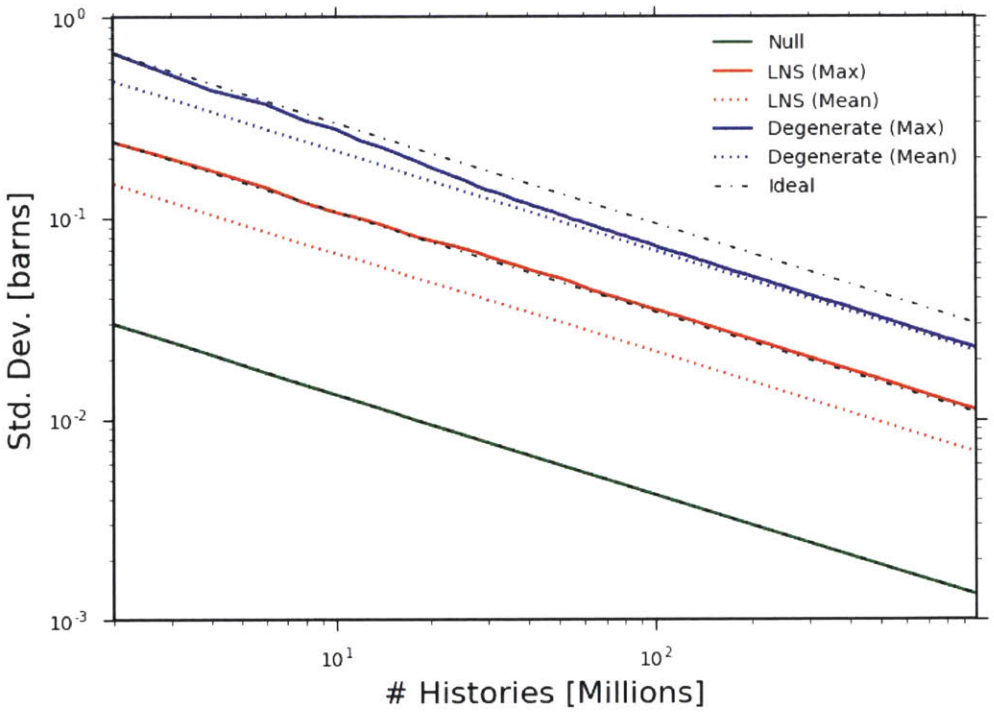
The MGXS "convergence rate" as defined here refers to the reduction in the statistical uncertainties with the number of independent simulated realizations (*i.e.*, MC batches). By definition, the empirical MC tally standard deviations will converge as $1/\sqrt{N}$ for N batches (see Sec. 3.1.2). However, this convergence rate only holds true for MC particle transport simulations if the tally realizations for each batch are i.i.d. Although this is the case for fixed source MC simulations, it is not the case for eigenvalue simulations for reactor physics analysis. In particular, the correlation between the fission source distributions for successive batches is correlated in criticality calculations which breaks the independence assumption and dampens the convergence rate [40, 41]. The auto-

correlation coefficient between successive batches is required to account for correlations of the fission source sites in the estimated standard deviation of the mean. In general, the coefficient cannot be quantified without an ensemble of simulations, and therefore is not used here to estimate the MGXS standard deviations. As a result, the reported standard deviations and convergence rates are likely to be under-estimated and over-estimated, respectively, especially for the high dominance ratio quarter core BEAVRS model.

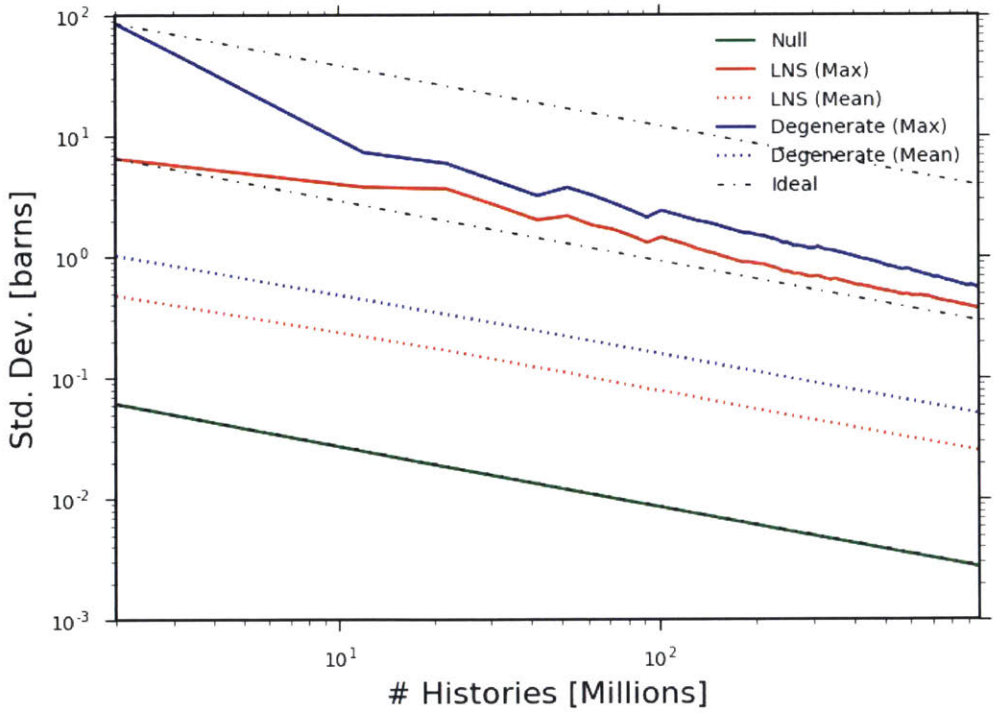
9.4.2 Batchwise Statistical Uncertainties

The statistical uncertainties for the pin-wise MGXS were reported for the "fully converged" MGXS in Sec. 9.1.1. The MGXS uncertainties were computed with uncertainty propagation (see Sec. 3.2.2) of the standard deviations of the reaction rate and flux tally means (see Eqn. 3.9). This section presents the evolution of those uncertainties with the number of particles simulated by OpenMC. The standard deviation of the pin-wise MGXS for null, degenerate and LNS homogenization were computed every 20 batches for U-235 fission and U-238 capture MGXS. The standard deviations are plotted for the 1.6% enriched assembly and the quarter core BEAVRS models in Figs. 9-19 to 9-21. The evolution is shown for both the maximum and mean standard deviations of the population of MGXS for LNS and degenerate homogenization.

The relation in Eqn. 9.5 may be used to understand the relative difference (*i.e.*, vertical offset) in statistical uncertainties for the three homogenization schemes. First, the relation predicts a $\sim$16$\times$ reduction of the uncertainties for the null homogenized MGXS of the 264 pins in the 1.6% enriched assembly with respect to the degenerate MGXS. This compares well to the empirically observed reduction of 18 – 22$\times$ for the the different MGXS types. Similarly, the relation predicts a $\sim$23$\times$ reduction in the uncertainties for the null homogenized MGXS of the 549 1.6% enriched fuel pins in the quarter core BEAVRS model with respect to the degenerate MGXS. The empirically observed reduction of 18 – 20$\times$ is smaller, most likely due to the larger disparity in the standard deviations of the MGXS for each fuel pin, which breaks the assumption made in Sec. 9.4.1.1 that the standard deviations are identical for all samples.

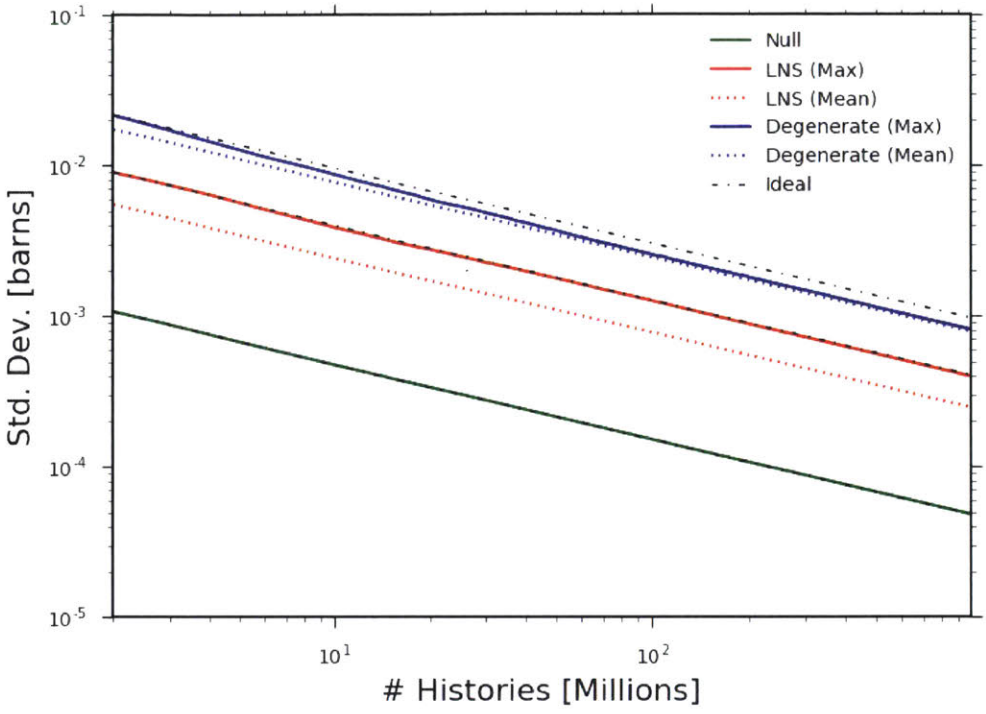


(a)

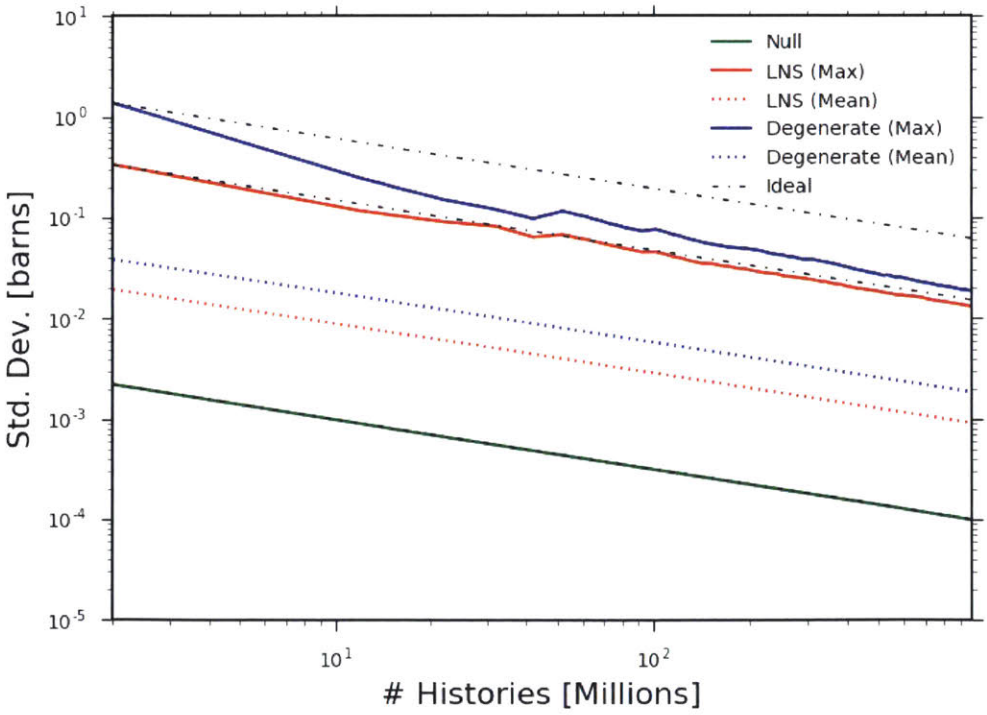


(b)

Figure 9-19: The standard deviation of pin-wise U-238 capture MGXS (group 27 of 70) for 1.6% enriched fuel pins in a single assembly (a) and the quarter core BEAVRS model (b). The thin black dashed lines indicate the theoretical $1/\sqrt{N}$ convergence for the maximum standard deviations of each homogenization scheme.

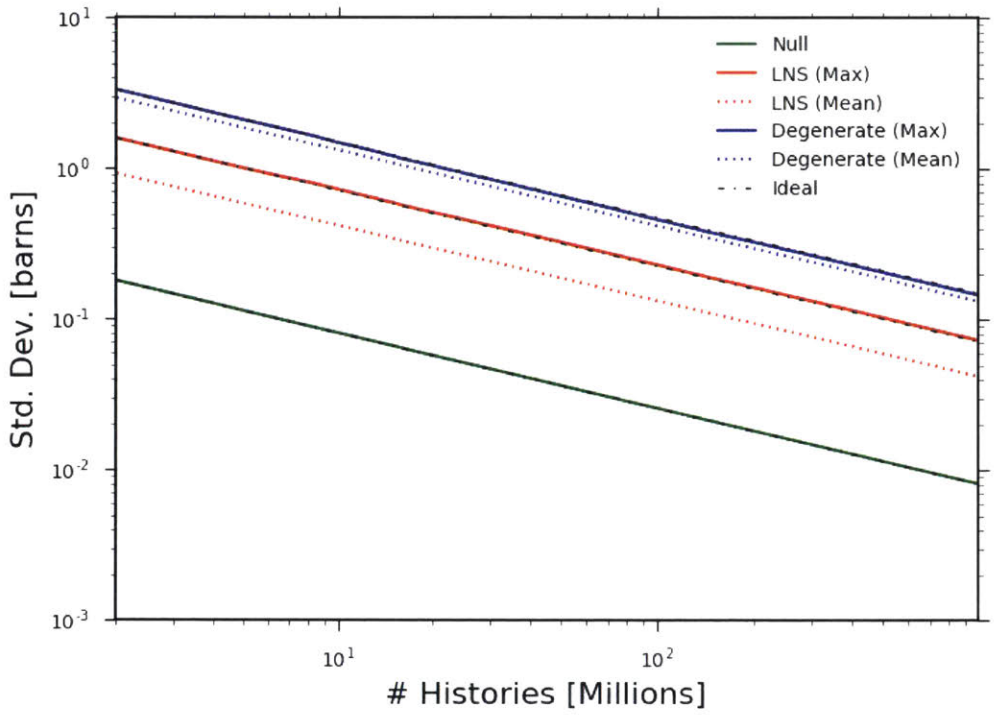


(a)

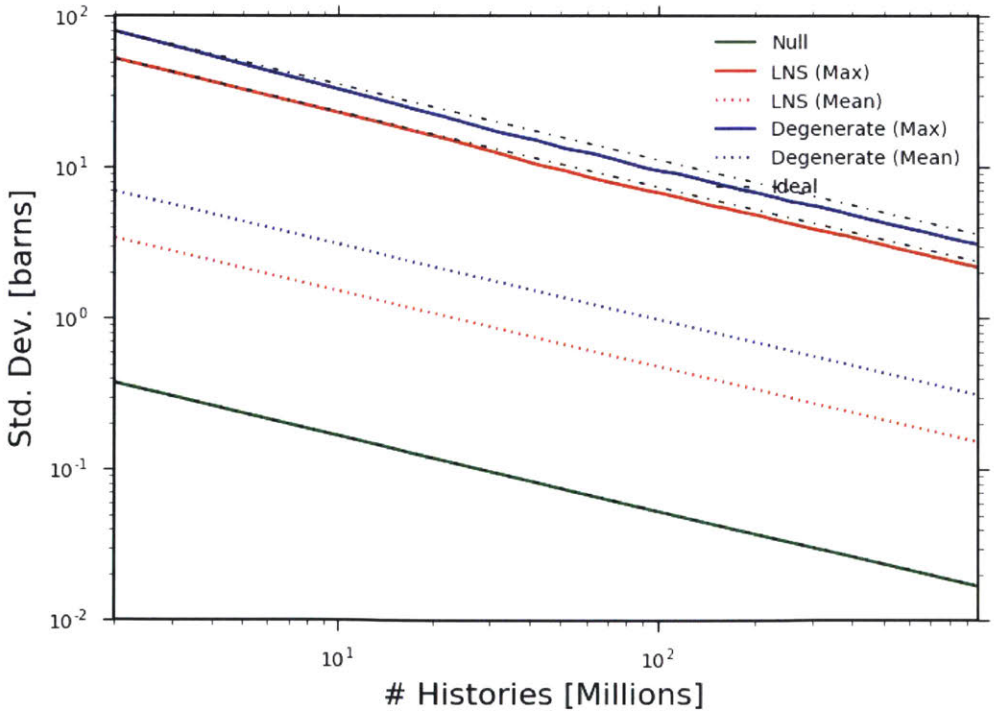


(b)

Figure 9-20: The standard deviation of pin-wise U-238 capture MGXS (group 1 of 2) for 1.6% enriched fuel pins in a single assembly (a) and the quarter core BEAVRS model (b). The thin black dashed lines indicate the theoretical $1/\sqrt{N}$ convergence for the maximum standard deviations of each homogenization scheme.



(a)



(b)

Figure 9-21: The standard deviation of pin-wise U-235 fission MGXS (group 2 of 2) for 1.6% enriched fuel pins in a single assembly (a) and the quarter core BEAVRS model (b). The thin black dashed lines indicate the theoretical $1/\sqrt{N}$ convergence for the maximum standard deviations of each homogenization scheme.

In addition, the gap between max and mean uncertainties for the degenerate and LNS schemes vary between the assembly and quarter core benchmarks. In particular, the max and mean uncertainties for the degenerate scheme differ by less than 10% for the assembly, but by ~2.5× for the quarter core model (this can also be seen in Tab. 9.1). This is caused by the highly uneven fission source distribution in the quarter core model, which results in a more uneven distribution of particle track densities across the fuel pin tally volumes than in the assembly. In contrast, the max and mean uncertainties for the LNS scheme differ by 60 – 70% for the assembly but by 13 – 16× for the quarter core model. The disparity between max and mean uncertainties for LNS homogenization is more complicated since it reflects the number of fuel pins assigned to each spatial homogenization set in addition to the uncertainties of the MGXS for each fuel pin.

The MGXS statistical uncertainties converge at the same rate for null, degenerate and LNS homogenization schemes. The uncertainties for null and LNS homogenization are smaller than those for the degenerate scheme according to the number of fuel pins assigned to each spatial homogenization set.

9.4.3 Batchwise Relative Percent Deviations

The statistical uncertainties presented in the preceding section estimate the amount by which MGXS may vary from their tallied means. However, the uncertainties do little to indicate the degree by which one may expect the solutions from a multi-group calculation to change from batch-to-batch of the MC simulation. This section seeks to quantify the impact of MGXS convergence on downstream codes by presenting the batchwise relative percentage deviation for MGXS with null, degenerate and LNS spatial homogenization.

The batchwise relative percentage deviation quantifies the change in cumulative MGXS estimates between two batches. The absolute batchwise relative deviation as employed here is computed for the cumulatively estimated microscopic MGXS $\hat{\sigma}^{b}_{x,i,k,g}$ and $\hat{\sigma}^{b+\Delta b}_{x,i,k,g}$ for batches b and $b + \Delta b$ as follows:

$$\Delta\hat{\sigma}_{x,i,k,g}[\%] = \left| \frac{\hat{\sigma}^{b}_{x,i,k,g} - \hat{\sigma}^{b+\Delta b}_{x,i,k,g}}{\hat{\sigma}^{b}_{x,i,k,g}} \right| \times 100 \tag{9.6}$$

The batchwise relative deviation indicates the degree by which the MGXS estimates change with the number of MC particle histories simulated. The MGXS estimates from batch-to-batch follow a random walk with steps which are increasingly attenuated with the number of batches. The degree of change between batches depends on the particle track density within each tally volume, and therefore should be ordered from least to greatest for null, LNS and degenerate homogenization.

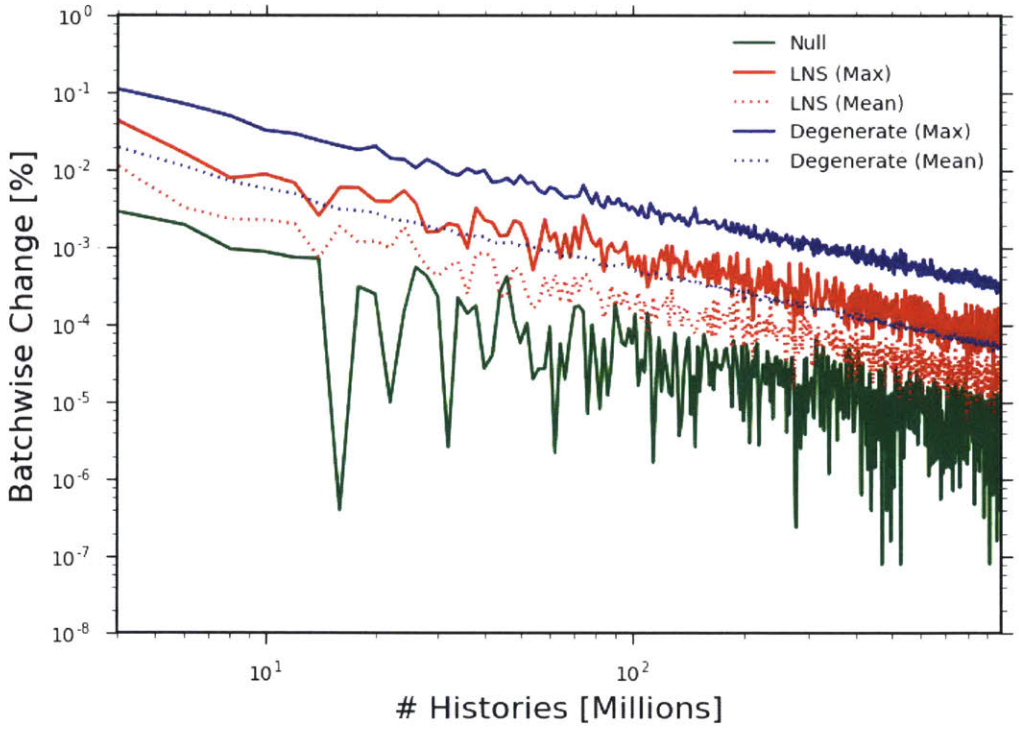
Due to computational resource constraints, the batchwise deviations were computed every $\Delta b = 20$ batches for each MC simulation. The absolute relative batchwise percentage deviations are plotted for the 1.6% enriched assembly and quarter core BEAVRS models in Figs. 9-22 to 9-24. The evolution is shown for both the maximum and mean deviations of the population of MGXS for LNS and degenerate homogenization.

A couple of observations can be made from the figures. First, the convergence rate (or log-log slope) of the batchwise deviations is very nearly the same for all three homogenization schemes. More importantly, however, the deviations are smallest for null homogenization and greatest for degenerate homogenization, as expected. This is manifested as a downward shift in the deviations for a given number of particle histories, and is due to the differential particle track density within each tally volume for the different homogenization schemes. The magnitude of the shift depends on the number of pins being averaged across, and is hence the greatest for the null and degenerate schemes for the quarter core BEAVRS model, where nearly 4,500 pins of each enrichment are averaged together for the null case (as compared to only 264 pins for the assembly). In contrast, the degenerate and LNS schemes are shifted by nearly the same amount for both benchmark models since the number of fuel pins averaged across for each LNS set is the same. These observations confirm that the more fuel pins that are homogenized together for MGXS clustering, the faster the resultant MGXS tally estimates will converge for use in multi-group codes.

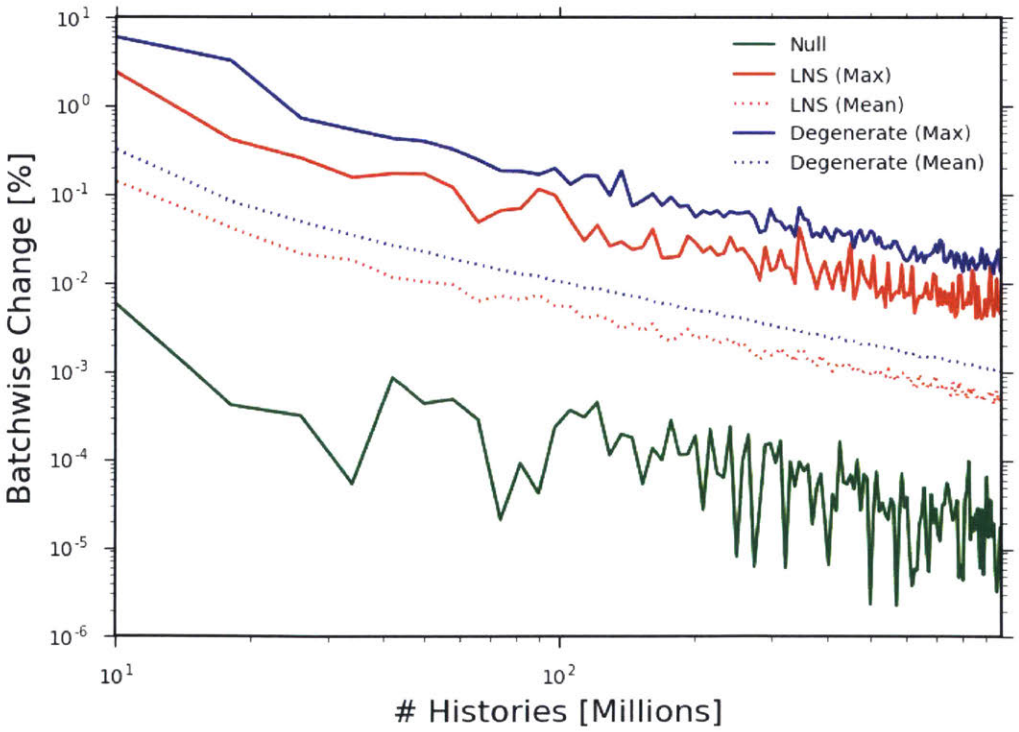
These results point to the need for a scheme that can effectively model MGXS

clustering in a more scalable fashion than LNS homogenization. LNS homogenization achieves little relative improvement in convergence of the batchwise deviations relative to the null and degenerate schemes since the pins in each fuel assembly are homogenized separately. The analysis of MGXS clustering in Sec. 9.1 indicated that fuel pins in many different assemblies may be effectively homogenized together while still accounting for clustering due to spatial self-shielding effects. This would increase the number of pins within each homogenization set and improve the convergence rate beyond what is achievable with LNS. The following chapter introduces a novel methodology based on unsupervised statistical clustering algorithms which seeks to outperform both the accuracy and the convergence of LNS homogenization.

As with the statistical uncertainties, the MGXS batchwise deviations converge at the same rate for null, degenerate and LNS homogenization schemes. The relative difference in deviations between schemes is reduced according to the number of fuel pins assigned to each spatial homogenization set.

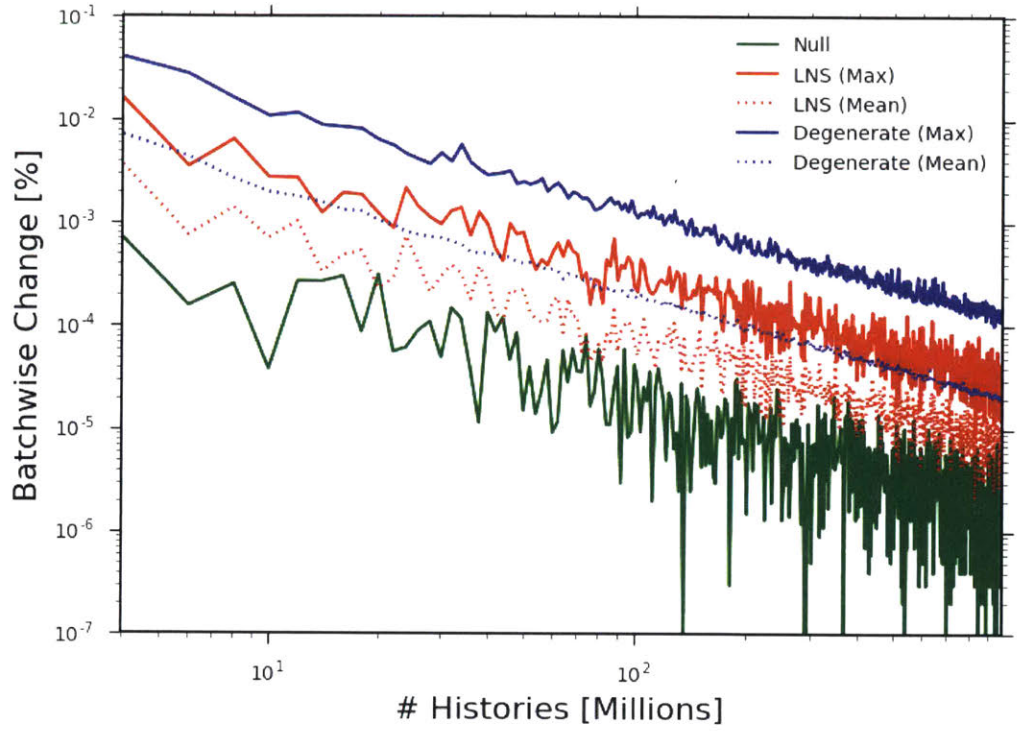


(a)

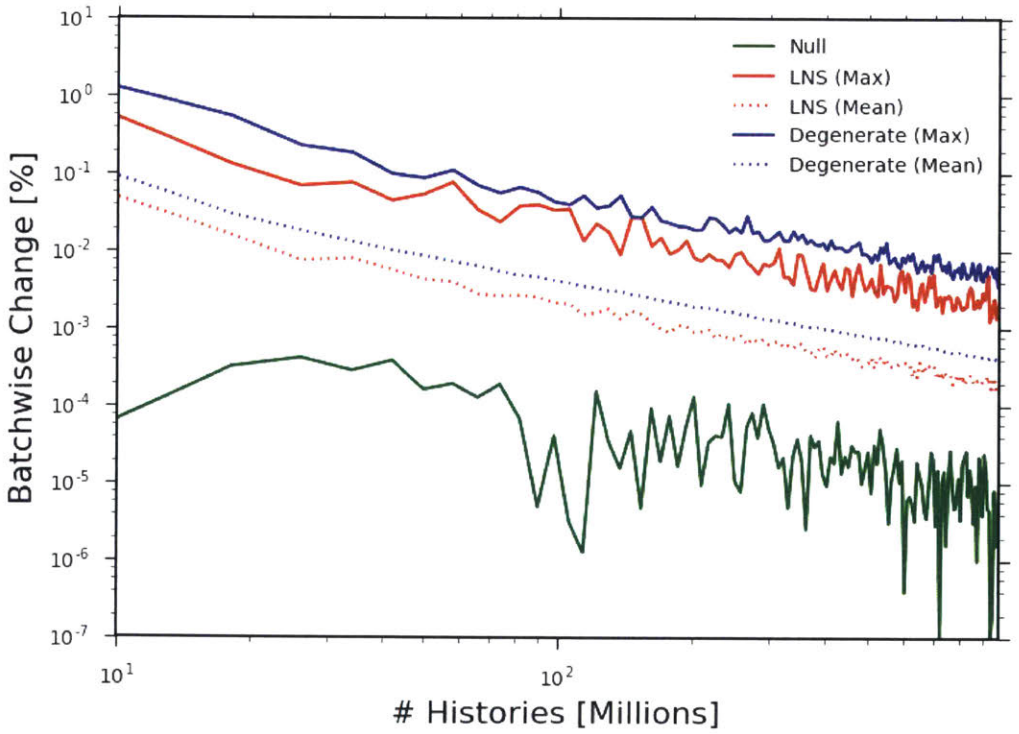


(b)

Figure 9-22: The absolute relative batchwise percentage deviation of pin-wise U-238 capture MGXS (group 27 of 70) for 1.6% enriched fuel pins in a single assembly (a) and the quarter core BEAVRS model (b).

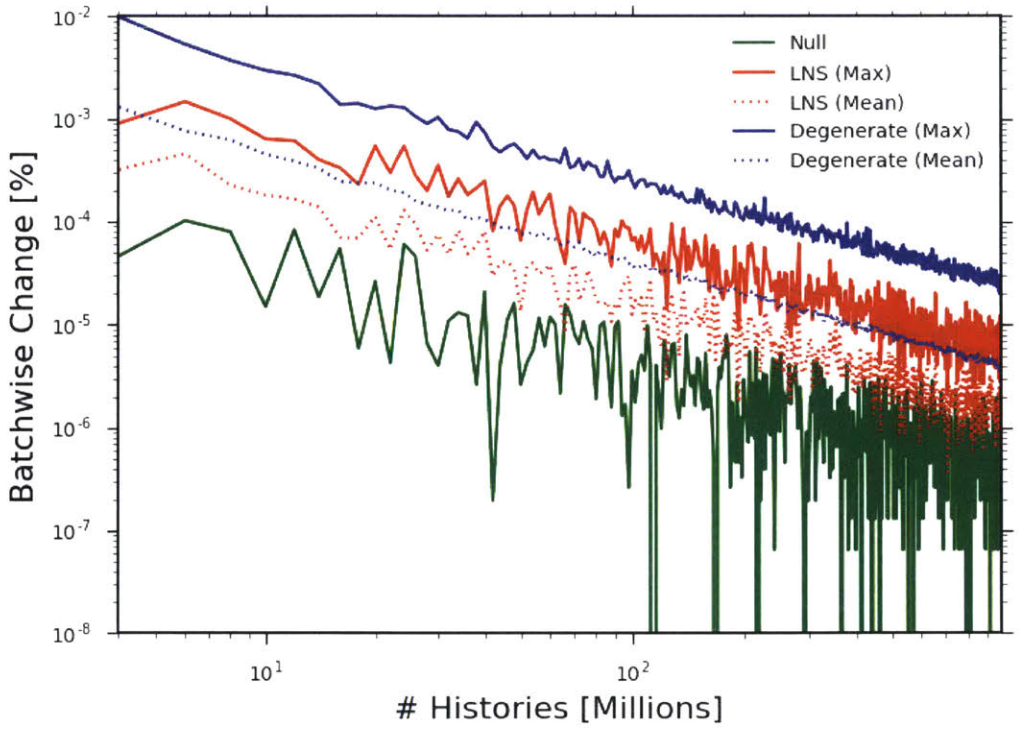


(a)

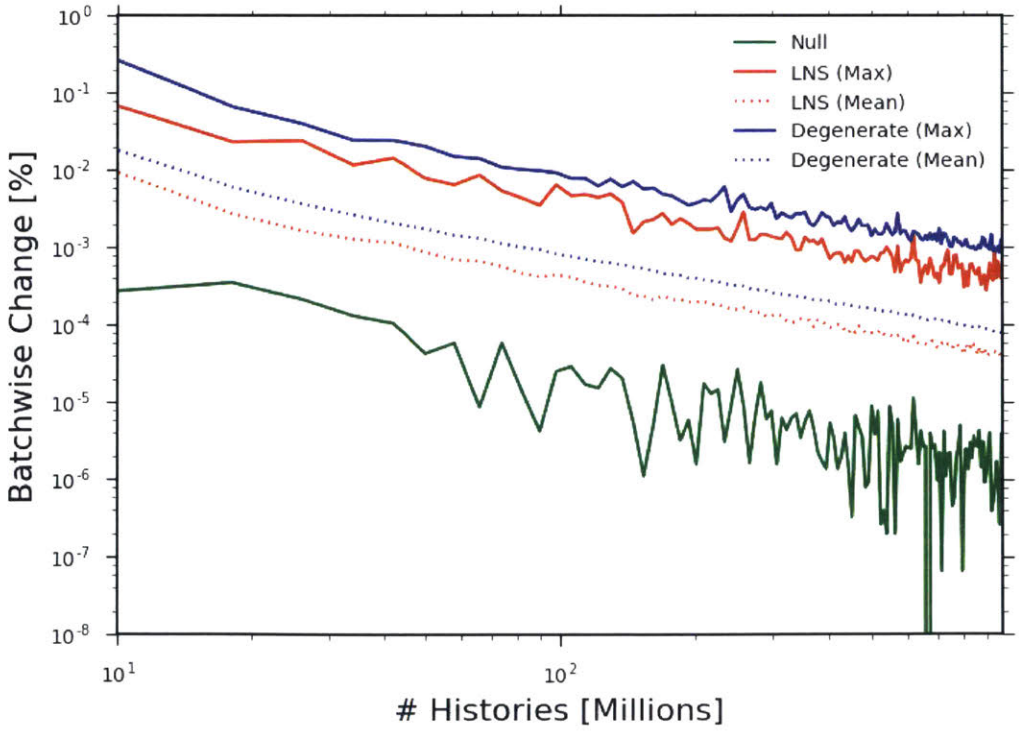


(b)

Figure 9-23: The absolute relative batchwise percentage deviation of pin-wise U-235 fission MGXS (group 1 of 2) for 1.6% enriched fuel pins in a single assembly (a) and the quarter core BEAVRS model (b).



(a)



(b)

Figure 9-24: The absolute relative batchwise percentage deviation of pin-wise U-235 fission MGXS (group 2 of 2) for 1.6% enriched fuel pins in a single assembly (a) and the quarter core BEAVRS model (b).

Highlights

- MGXS data for each of the six heterogeneous benchmarks was analyzed to show that spatial heterogeneities induce MGXS *dispersion* and *clustering*.
- Degenerate homogenization accounts for MGXS clustering, but requires a large number of MC particle histories to converge MGXS.
- The LNS spatial homogenization scheme is introduced which assigns fuel pins with similar neighbors to the same spatial homogenization set.
- The six benchmarks are modeled with MGXS generated by the LNS scheme:
  - *Eigenvalues* – LNS has no systematic impact on the eigenvalues due to the preservation of global reactivity.
  - *Fission Rates* – LNS performs slightly better than degenerate homogenization, which is only marginally better than the null scheme, since the fission rate predictions are largely insensitive to MGXS clustering.
  - ***U-238 Capture Rates* – LNS reduces the error by 10 – 30% as compared to degenerate homogenization** for assembly and periodic colorset benchmarks. However, LNS fails to distinguish between pins at inter-assembly and assembly-reflector interfaces which results in a systematically large reaction rate error for these fuel pins.
- A series of case studies illustrate the need to homogenize across many fuel pins to reduce the MGXS statistical uncertainties and batchwise deviations.
- An improved methodology is needed which can:
  - **Model arbitrary MGXS clustering** from all spatial self-shielding effects.
  - **Minimize the # of materials** to improve MGXS statistical uncertainties.

Chapter 10

The *i*MGXS Spatial Homogenization Scheme

The preceding chapter illustrated the clustering of pin-wise MGXS as a result of spatially self-shielded spectral effects. It was shown that MGXS clustering must be appropriately modeled to accurately resolve pin-wise U-238 capture rates. The LNS spatial homogenization scheme was introduced to predict MGXS clustering with a geometric template-like approach. The LNS scheme was shown to achieve the same level of accuracy as degenerate homogenization while simultaneously accelerating the MC tally convergence rate for simple benchmark problems. However, the LNS scheme suffered from its inability to predict MGXS clustering in geometries with water reflectors and steel baffles. In addition, the LNS scheme did not scale well with the complexity of the core geometry, resulting in a large number of materials and thus an under-accelerated convergence rate. This chapter introduces an adaptable and scalable alternative to LNS which uses unsupervised statistical learning methods to predict MGXS clustering.

The novel methodology presented here – referred to as *inferential MGXS* (*i*MGXS) spatial homogenization – uses algorithms developed by the machine learning community to infer MGXS clusters directly from MC tally data rather than predict clustering from an analysis of the core geometry. The *i*MGXS scheme can flexibly accommodate arbitrary core heterogeneities better than LNS or other geometric-based approaches which must be extensively customized for particular core geometries. In addition, the *i*MGXS scheme

aims to greatly accelerate the convergence rate of MGXS tallied with MC with respect to the degenerate and LNS schemes as shown in Fig. 10-1.

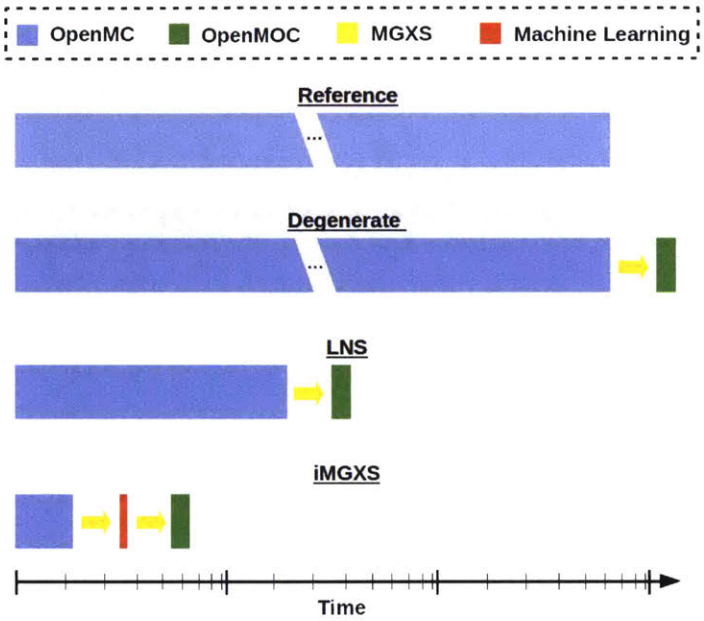


Figure 10-1: The expected relative runtime for the degenerate, LNS and *i*MGXS spatial homogenization schemes with respect to a reference MC calculation.

This chapter begins with an overview of the *i*MGXS scheme in Sec. 10.1, including in-depth presentations of each stage of tally data pre-processing, including feature extraction (Sec. 10.2) and selection (Sec. 10.3), and dimensionality reduction (Sec. 10.4). Sec. 10.5 highlights a few statistical clustering algorithms which may be interchangeably utilized within the code framework for iMGXS implemented for this thesis. A few heuristics for clustering model selection are discussed in Sec. 10.6. Sec. 10.7 discusses the track density-weighting used to spatially homogenize the pin-wise MGXS. Finally, Sec. 10.8 illustrates the material configurations produced by the *i*MGXS scheme for the heterogeneous PWR benchmarks studied in this thesis. The eigenvalues and pin-wise fission and U-238 capture rates produced with the *i*MGXS scheme are evaluated in the following chapter.

10.1 Overview of the *i*MGXS Data Processing Pipeline

The *i*MGXS spatial homogenization scheme is a multi-stage data processing *pipeline*. The objective of the scheme is to infer the optimal assignment of cluster labels to fuel pin instances directly from MC tally data. The cluster labels output by *i*MGXS are then used to generate track density-weighted MGXS (see Sec. 9.2.2) for each cluster of fuel pin instances using the same process as that employed by LNS homogenization (Eqn. 9.4). The *i*MGXS methodology differs from the LNS scheme in that it makes no consideration of the geometry and materials configuration and only examines tallied MC tally data when assigning cluster labels to fuel pin instances.

A high-level overview of the various stages of the *i*MGXS data processing pipeline is illustrated in Fig. 10-2. The *i*MGXS pipeline may be configured in various ways and this thesis makes no presumption that the incarnation presented in Fig. 10-2 is the best or most reliable version. Future work may develop schemes which improve upon the particular formulation of *i*MGXS presented in this chapter. Irregardless of the particular configuration, the overarching concept is that the *i*MGXS pipeline provides a *mapping* between MC tally data and cluster labels for each fuel pin[63]. Each of the six stages in Fig. 10-2 is detailed in the following sections of this chapter.

10.2 Feature Extraction

The *feature extraction* stage of the data processing pipeline in Fig. 10-2 builds *features* from MC tally data. In machine learning, features are measured quantities which serve as inputs to a predictive model. Features may be engineered based upon prior domain knowledge or inferred from automated feature learning algorithms such as neural networks. In the context of *i*MGXS, features are restricted to tallies, or combinations of tallies, from MC simulations which provide information about which fuel pin instances experience similar spatial self-shielding effects. For example, the pin-wise microscopic MGXS may be used as features since they exhibit the very clustering effects which

[63] Similarly, the LNS scheme defines a mapping between the combinatorial geometry and the cluster labels.

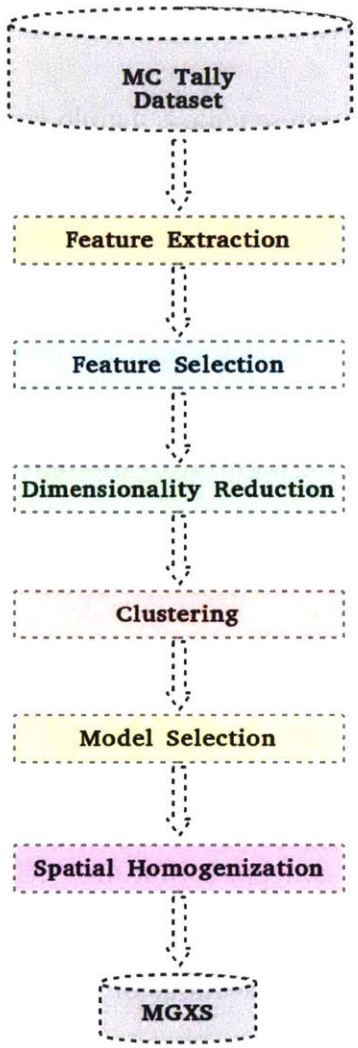


Figure 10-2: The multi-stage *i*MGXS data processing pipeline.

the *i*MGXS scheme attempts to predict[64]. Other tallied quantities may also be used as features for predicting which fuel pin instances have similarly self-shielded MGXS. The goal of extracting *i*MGXS features is to enable machine learning algorithms to identify MGXS clusters as quickly as possible from "noisy" or unconverged tally data.

The *i*MGXS scheme splits the MC tally dataset up into *samples* for each particular instance k of a fuel pin as illustrated in Fig. 10-3. A *sample* is a random realization of a *feature vector* corresponding to a particular instance k of a fuel pin. A sample may be comprised of features derived from MC tally data for one or multiple nuclides, energy

[64]Using the pin-wise MGXS as the only feature(s) for unsupervised clustering would render the clustering problem equivalent to specifying boundaries between the samples illustrated in the rug plots in Sec. 9.1.3.

groups and/or reaction types. Ideally, the features should maximize the *separation distance* in feature space $\{\hat{f}_{1,\cdot}, \hat{f}_{2,\cdot}, \ldots, \hat{f}_{J,\cdot}\}$ between samples in different clusters. Since features in *i*MGXS are tallied quantities from MC simulations, some samples may include "outlier" feature realizations which take on values far removed from the "true" value of the feature[65]. Outliers may result in the incorrect assignment of cluster labels to fuel pin instances. The predictive models used in *i*MGXS are trained with the complete vector of features in order to mitigate the sensitivity of the models to outlier features and minimize the frequency of mis-labeled cluster assignments.

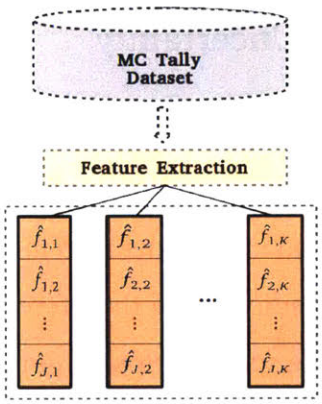


Figure 10-3: *i*MGXS extracts feature vectors for each sample (fuel pin instance).

In addition, the features do not necessarily have to be defined for the same energy group structure as the MGXS one wishes to cluster. For example, some or all features may be tallied on a relatively coarse energy group structure in order to minimize their MC statistical uncertainties. It may in fact be beneficial to tally features in few groups in order to identify clusters with fewer MC particle histories than would needed to distinguish structure from "noisy" fine group features. These coarse group features may be input to a clustering algorithm for spatial homogenization of pin-wise MGXS defined on a fine(r) energy group structure. For example, the following sections and chapters cluster features defined for a coarse 2-group structure for spatial homogenization of the "fine" 70-group pin-wise MGXS that are later used in downstream MOC transport calculations.

The following sections introduce the features employed by *i*MGXS in this and the

[65]The "true" values are the expected values for each feature.

following chapter. These include the pin-wise MGXS and their statistical uncertainties (Sec. 10.2.1), as well as features referred to as fractional reactivities (Sec. 10.2.2), spectral indices (Sec. 10.2.3) and reaction fractions (Sec. 10.2.4). Each fuel pin instance carries an instance of each feature, some of which may be specific to one or more nuclides, energy groups and/or reaction types. Each feature is presented with accompanying illustrations generated using a custom-built tool to visualize *i*MGXS features [83], and which correspond to the 1.6% enriched fuel assembly benchmark.

10.2.1 MGXS Statistical Uncertainty

The statistical uncertainty for each pin-wise MGXS may be a useful feature to indicate clustering effects. In particular, the standard deviation of the sample mean is easily obtained from OpenMC tallies and can be included in the feature vectors. The intuition behind this is that the track densities – and therefore, the statistical uncertainties – in different fuel pins may reflect the spatial self-shielding effects experienced by different types of fuel pins. For example, the thermal flux track density will vary for fuel pins with different levels of moderation from CRGTs, reflectors, etc., which may present itself through a systematic clustering of the statistical uncertainties. Similarly, the track density will vary for fuel pins which experience varying degrees of spatial self-shielding in U-238 resonance groups – which greatly impacts the resultant MGXS in those groups – and may be identifiable in the MGXS uncertainties.

The statistical uncertainty feature is illustrated with scatter plots in Figs. 10-4, 10-5 and 10-6 for 2-group U-235 fission and U-238 capture MGXS data. The scatter plots include a single data point for each of the 264 fuel pins in the 1.6% enriched fuel assembly benchmark. The x and y coordinates correspond to the tallied MGXS means $\hat{\sigma}_{x,i,k,g}$ and standard deviations $\sigma_{\hat{\sigma}_{x,i,k,g}}$ in units of barns, respectively. The complete datasets are illustrated in Figs. 10-4b and 10-5b. The interactive *i*MGXS visualization tool was used to select clusters of MGXS and plot the geometry to indicate the associated fuel pins, as displayed in Figs. 10-4c to 10-4f and 10-5c to 10-5f for the fission and capture MGXS, respectively. The figures illustrate that the U-235 MGXS uncertainties are

smallest for the fuel pins with the most moderation (*i.e.*, pins facially adjacent to CRGTs) and a relatively larger thermal flux track density. In contrast, the U-238 capture MGXS uncertainties are largest for those pins with the most moderation due to the smaller fast-to-thermal flux ratio in these pins.

Notwithstanding the motivation to use the statistical uncertainties, there are also potential challenges associated with this approach. First, it would be possible for two different fuel pin instances to experience the same flux shape – and therefore have the same MGXS – but very different flux magnitudes. Although the MGXS uncertainties for the two pins would be very different, this would not accurately reflect the fact that the two pins should ideally be assigned to the same MGXS cluster. Furthermore, error propagation theory only approximates the MGXS uncertainties since it neglects the covariance between reaction rate and flux tallies. This approximation overestimates the MGXS uncertainties since the reaction rate and flux tallies are highly correlated. As a result, the uncertainties may not exhibit systematic trends with poorly converged MC data since the estimated uncertainties will be larger than the "true" values. However, the MGXS may be consistently overestimated for all fuel pins, since the correlation between the reaction rate and flux is likely similar across fuel pins instances. Finally, it should be noted that these are not reasons to eliminate the uncertainties as a possible feature; rather, it incites the need for robust feature selection as discussed in Sec. 10.3.

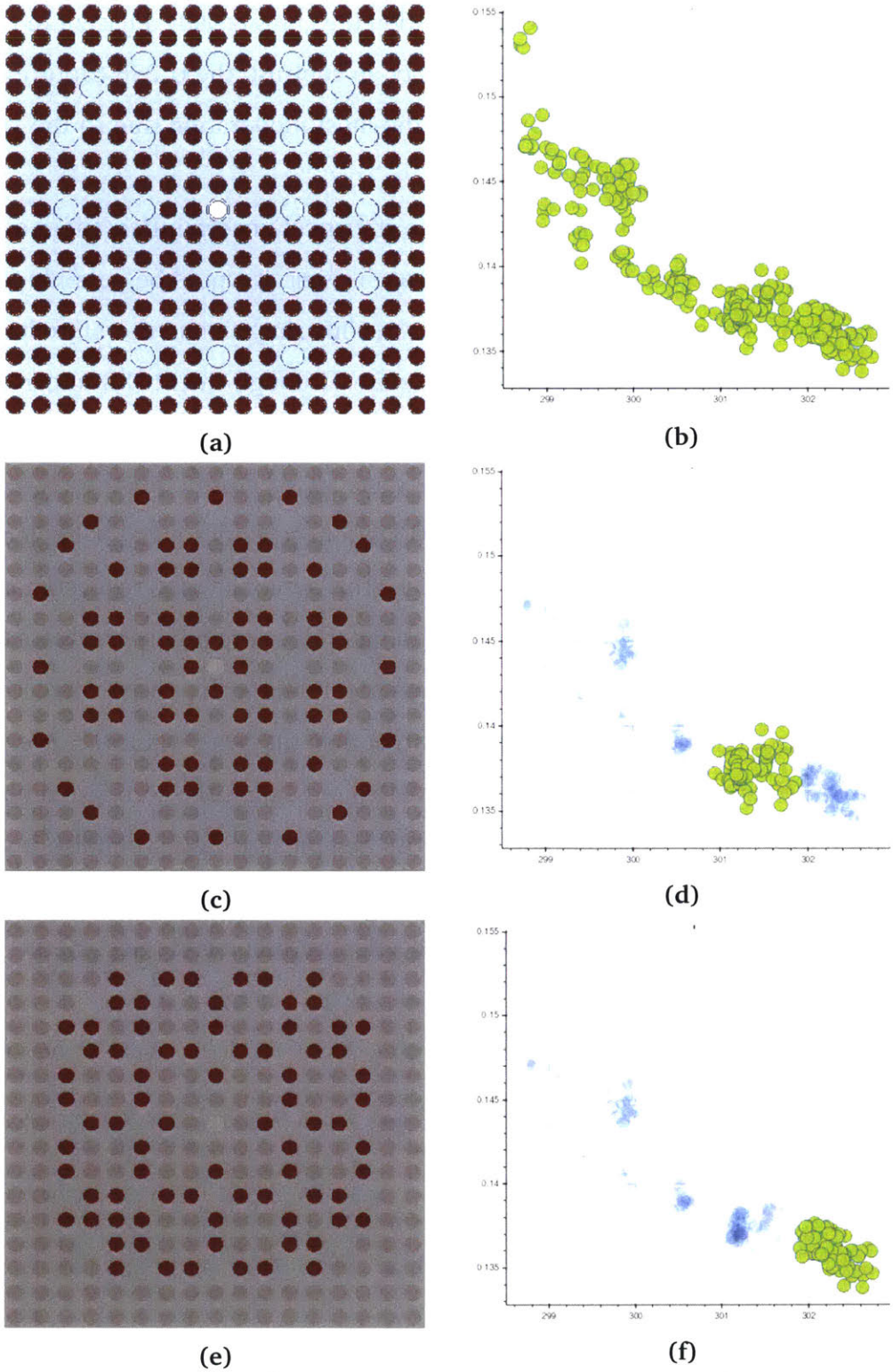


Figure 10-4: Scatter plots of the pin-wise U-235 fission (group 2 of 2) MGXS means (x) and percent relative uncertainties (y) for the 1.6% enriched fuel assembly.

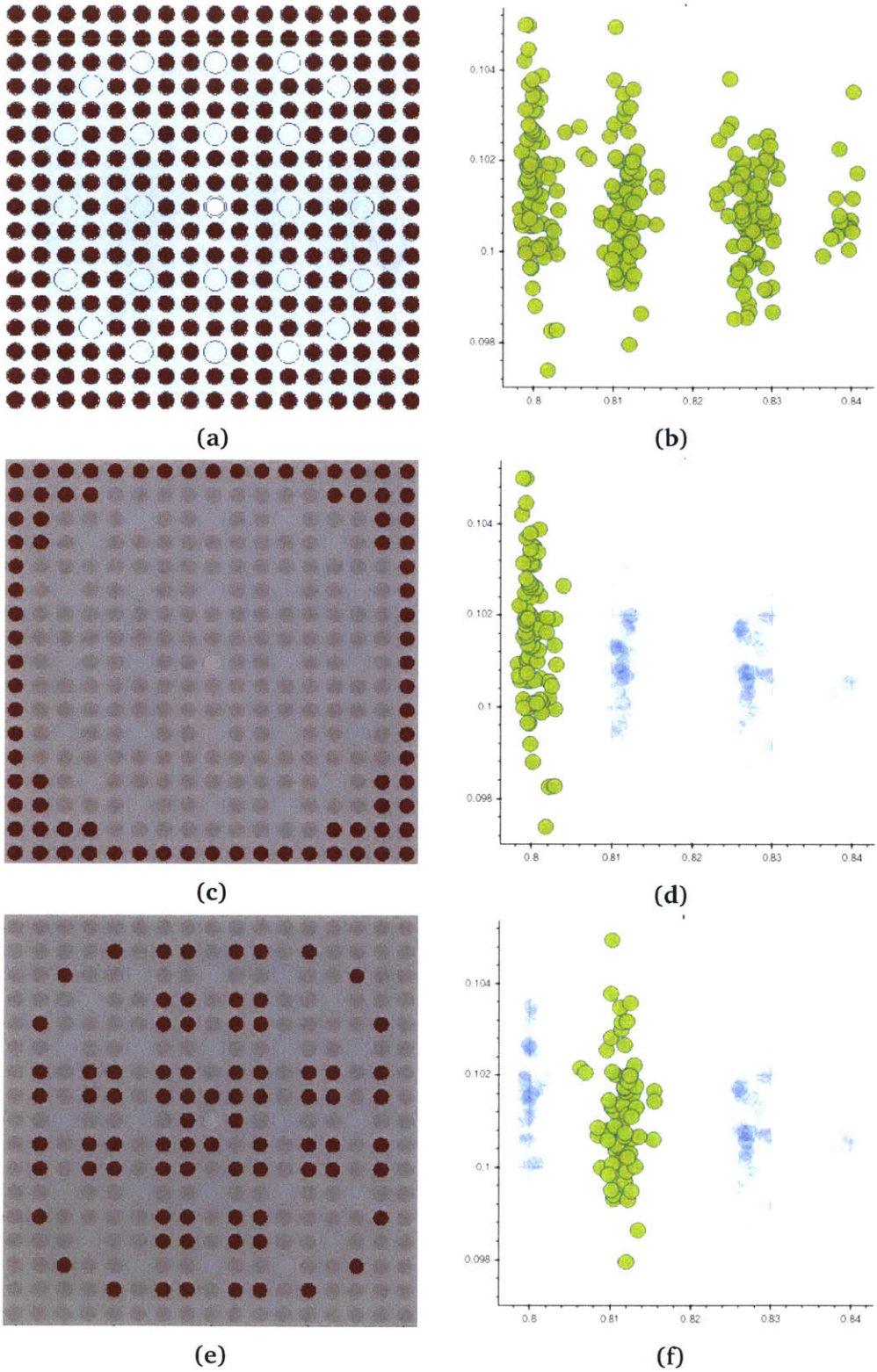


Figure 10-5: Scatter plots of the pin-wise U-238 capture (group 1 of 2) MGXS means (x) and percent relative uncertainties (y) for the 1.6% enriched fuel assembly.

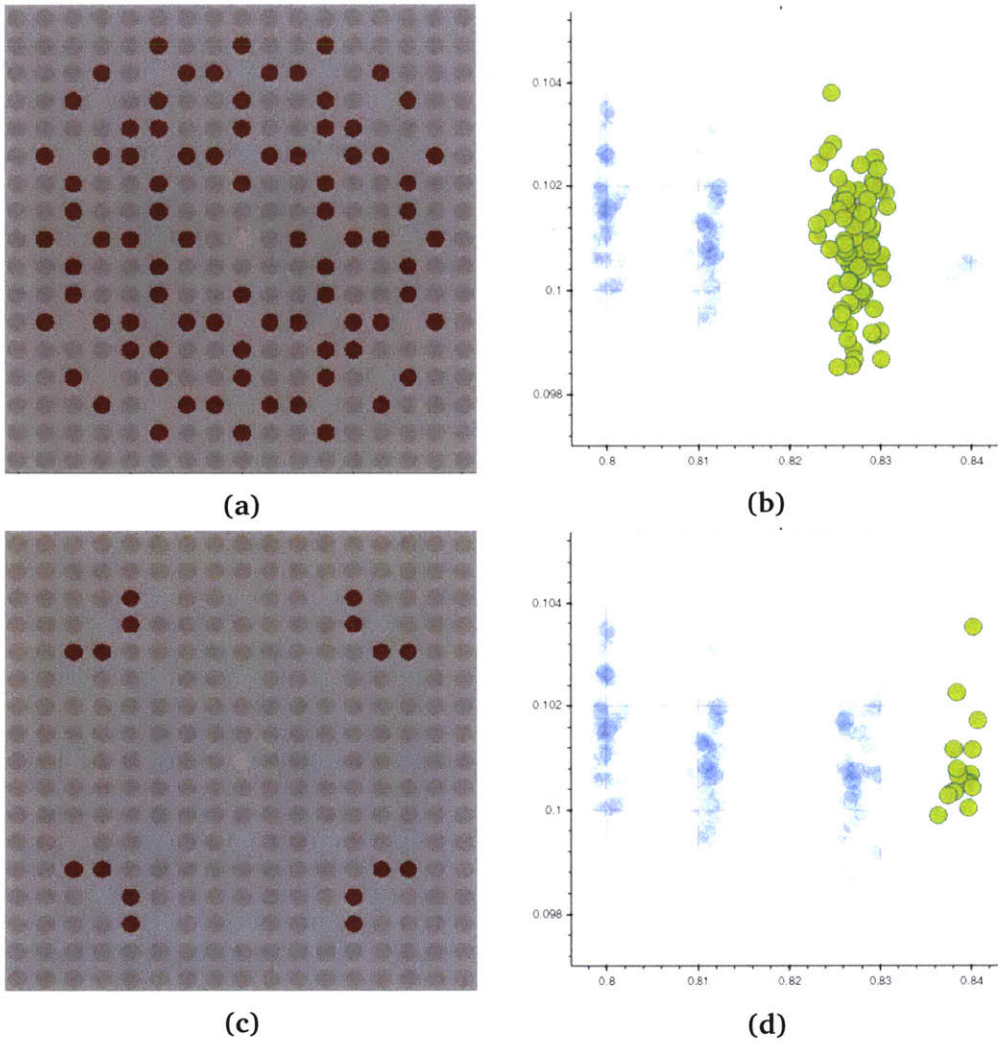


Figure 10-6: Scatter plots of the pin-wise U-238 capture (group 1 of 2) MGXS means (x) and percent relative uncertainties (y) for the 1.6% enriched fuel assembly.

10.2.2 Fractional Reactivity

The *fractional reactivity* compares the fission reaction rates for the population of fuel pins in a core geometry. This feature hypothesizes that fuel pins with similar spatial self-shielding effects have similar fission rates. This may be a valid assumption for simple geometries such as fuel assemblies and colorsets with all reflective/periodic BCs, but may not be true for benchmarks such as BEAVRS with large globally-varying power tilts due to leakage. The fission rate statistical uncertainties will necessarily be smaller than those for the microscopic MGXS since they are energy-integrated and summed across nuclides. As a result, clustering may emerge from "noisy" MC fission rate tallies more quickly than is possible for tallies for single energy groups and/or nuclides.

Although the pin-wise fission rates may be used directly as a feature, the *i*MGXS implementation in this thesis normalizes the pin-wise fission rates to the globally-integrated absorption rate in the fuel as follows:

$$\hat{\alpha}_k = \frac{\sum_{g=1}^{G} \nu\hat{\Sigma}_{f,i,k,g}\hat{\phi}_{k,g}}{\sum_{k=1}^{K}\sum_{g=1}^{G} \hat{\Sigma}_{a,i,k,g}\hat{\phi}_{k,g}} \times 10^5 \tag{10.1}$$

The normalized fractional reactivity $\hat{\alpha}_k$ is multiplied by 10^5 so that it may be reported in the familiar reactivity units of per cent mille (pcm)[66]. It is important to note that the denominator in Eqn. 10.1 encompasses all fuel pins of all compositions (*e.g.*, enrichments) in the core geometry, but does not include absorption in the moderator, clad, BPs, etc. As a result, the fractional reactivity as defined by this thesis is not indicative of the global reactivity, but rather the reactivity restricted to the fuel.

The fractional reactivity feature is illustrated with scatter plots in Figs. 10-7 and 10-8 for 2-group U-235 fission and U-238 capture MGXS data, respectively. The scatter plots include a single data point for each of the 264 fuel pins in the 1.6% enriched fuel assembly benchmark. The x and y coordinates correspond to the tallied MGXS means

[66]It might be preferable to equivalently normalize the fission rates to the mean. The normalization factor is a matter of analyst preference and does not impact the predictions made by unsupervised clustering algorithms if the features are standardized as discussed in Sec. 10.2.5.

$\hat{\sigma}_{x,i,k,g}$ and normalized fractional reactivities $\hat{\alpha}_k$ in units of barns and pcm, respectively. The complete datasets are illustrated in Figs. 10-7b and 10-8b. The interactive *i*MGXS visualization tool was used to select clusters of MGXS and plot the geometry to indicate the associated fuel pins, as displayed in Figs. 10-7c to 10-7f and 10-8c to 10-8f for the fission and capture MGXS, respectively.

The scatter plots illustrate the highly linear relationship between the U-235 fission MGXS and fractional reactivities: those pins facially and corner adjacent to distinct CRGTs have both the largest fission MGXS and fission rates due to the moderation of the neighboring CRGTs (at least for this benchmark). The clustering of U-238 capture MGXS generally indicates a similar proportional relationship with fractional reactivity. Perhaps more importantly, Fig. 10-8 distinguishes "sub-clusters" within each of the four primary clusters of U-238 capture MGXS. Although the sub-clusters within a primary cluster have similar MGXS[67], there is some variation due to subtle differences in the spatial self-shielding experienced by the pins in each primary cluster.

The visualizations highlight the *hierarchical* nature of spatial self-shielding effects: some factors are more important and induce primary clusters (*i.e.*, whether a fuel pin is adjacent to zero, one or two CRGTs), while other factors are less consequential yet still induce sub-clusters (*i.e.*, the type adjacency of neighboring CRGTs). The hierarchy of spatial self-shielding effects will grow increasingly intricate with the size and complexity of the reactor configuration, further motivating the importance for an unsupervised methodology like *i*MGXS to identify MGXS clustering for arbitrary core geometries.

[67]The track density-weighted average MGXS for the sub-clusters would be relatively similar.

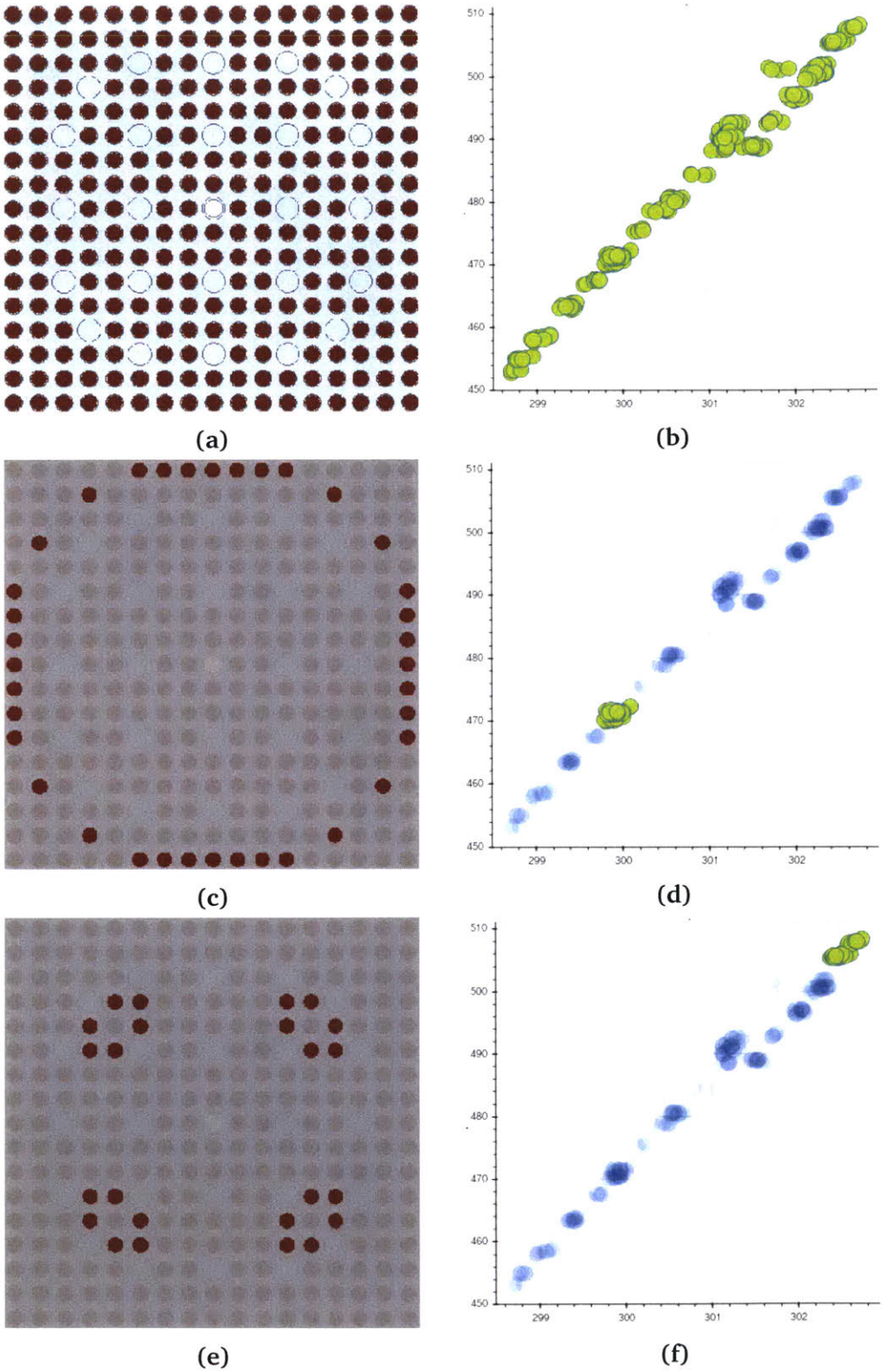


Figure 10-7: Scatter plots of the pin-wise U-235 fission (group 2 of 2) MGXS means (x) and fractional reactivities (y) for the 1.6% enriched fuel assembly.

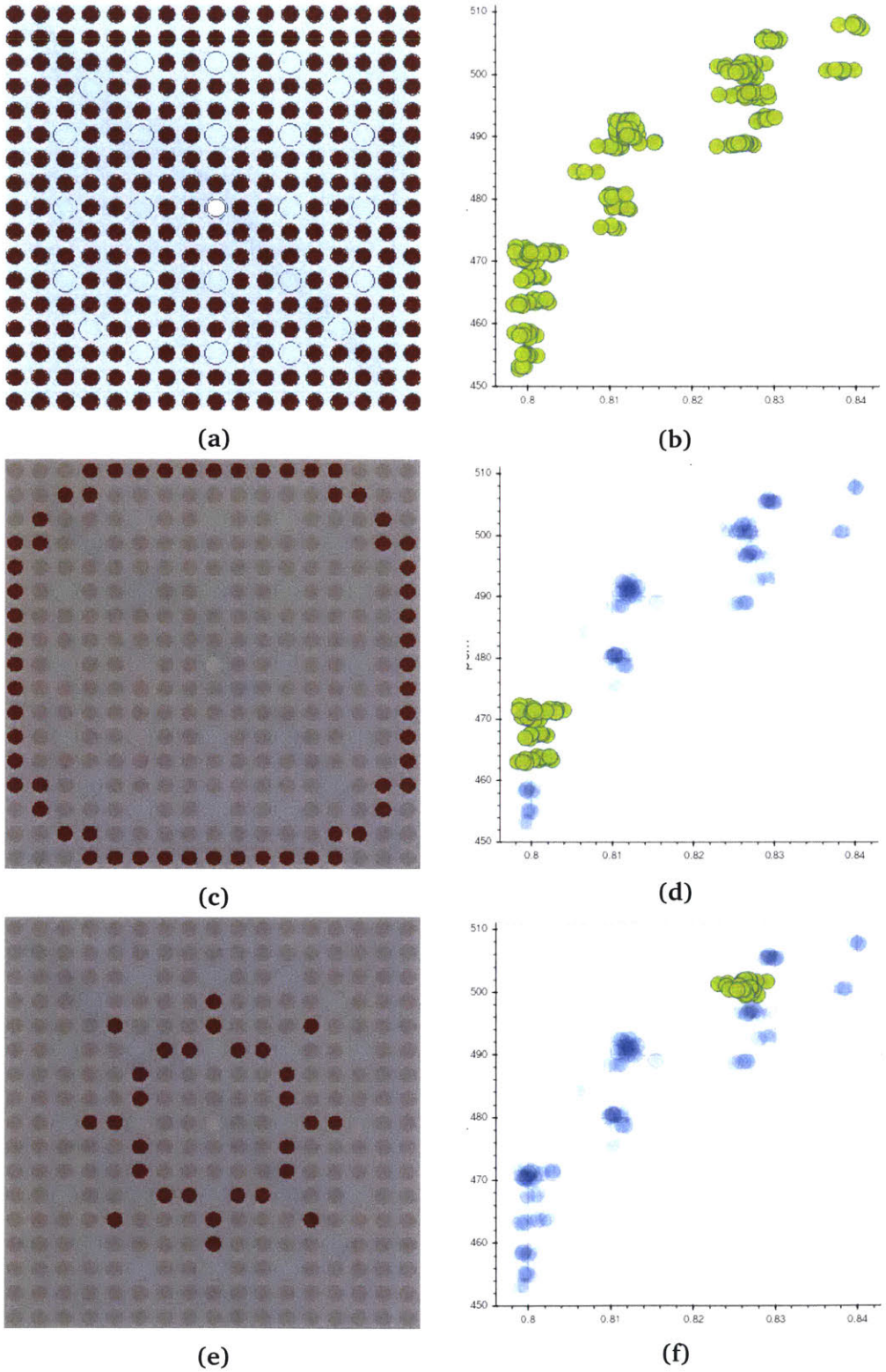


Figure 10-8: Scatter plots of the pin-wise U-238 capture (group 1 of 2) MGXS means (x) and fractional reactivities (y) for the 1.6% enriched fuel assembly.

10.2.3 Spectral Index

The *spectral index* compares the ratio of energy-integrated U-238 capture to U-235 fission reaction rates in each fuel pin. The *i*MGXS implementation in this thesis computes spectral indices as follows:

$$\hat{\beta}_k = \frac{\sum_{g=1}^{G} \hat{\sigma}^{238}_{\gamma,k,g} \hat{\phi}_{k,g}}{\sum_{g=1}^{G} \hat{\sigma}^{235}_{f,k,g} \hat{\phi}_{k,g}} \tag{10.2}$$

where $\hat{\sigma}^{238}_{\gamma,k,g}$ and $\hat{\sigma}^{235}_{f,k,g}$ are the microscopic U-238 radiative capture and U-235 fission production MGXS for fuel pin instance k and energy group g, respectively[68]. This feature postulates that the capture-to-fission ratio will significantly vary for fuel pins and energy groups with different spatial self-shielding effects. The spectral index is especially relevant since it accounts for relative differences in the U-238 capture rates, which previous chapters showed can only be accurately computed if an appropriate pin-wise spatial homogenization scheme is used. Unlike the fractional reactivity, the spectral index may identify pins with similar flux shapes but very different flux magnitudes since it is the ratio of two pin-wise reaction rates.

The spectral index feature is illustrated with scatter plots in Figs. 10-9 and 10-10 for 2-group U-235 fission and U-238 capture MGXS data, respectively. The scatter plots include a single data point for each of the 264 fuel pins in the 1.6% enriched fuel assembly benchmark. The x and y coordinates correspond to the tallied MGXS means $\hat{\sigma}_{x,i,k,g}$ and spectral indices $\hat{\beta}_k$, respectively. The complete datasets are illustrated in Figs. 10-9b and 10-10b. The interactive *i*MGXS visualization tool was used to select clusters of MGXS and plot the geometry to indicate the associated fuel pins, as displayed in Figs. 10-9c to 10-9f and 10-10c to 10-10f for the fission and capture MGXS, respectively.

The figures illustrate a complex relationship between the U-235 fission MGXS and the spectral indices. In contrast to the fractional reactivity, both the fission and capture MGXS

[68]The U-235 fission and fission production MGXS may be used interchangeably in Eqn. 10.2 with no impact on the predictions made by unsupervised clustering algorithms.

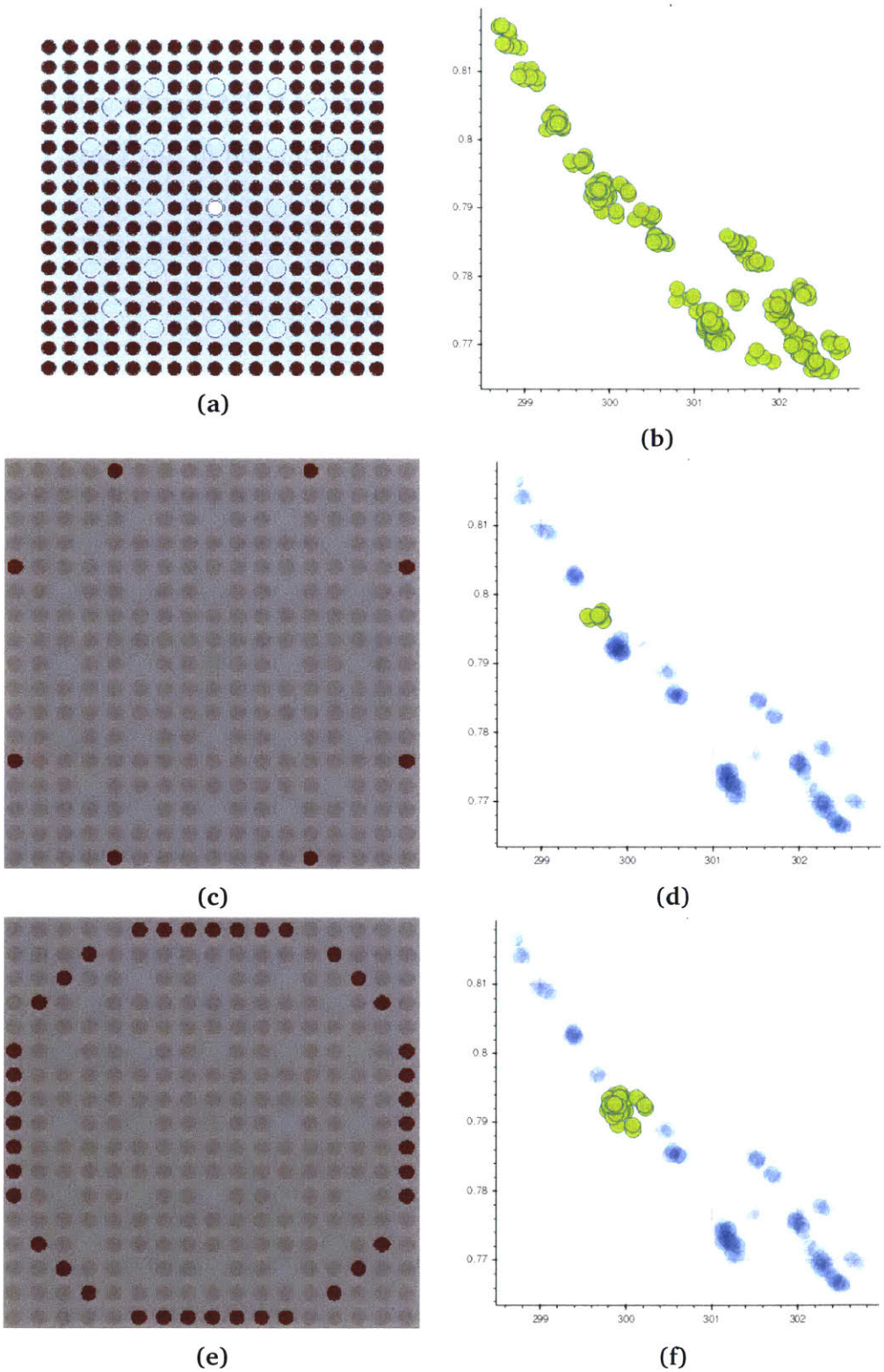


Figure 10-9: Scatter plots of the pin-wise U-235 fission (group 2 of 2) MGXS means (x) and spectral indices (y) for the 1.6% enriched fuel assembly.

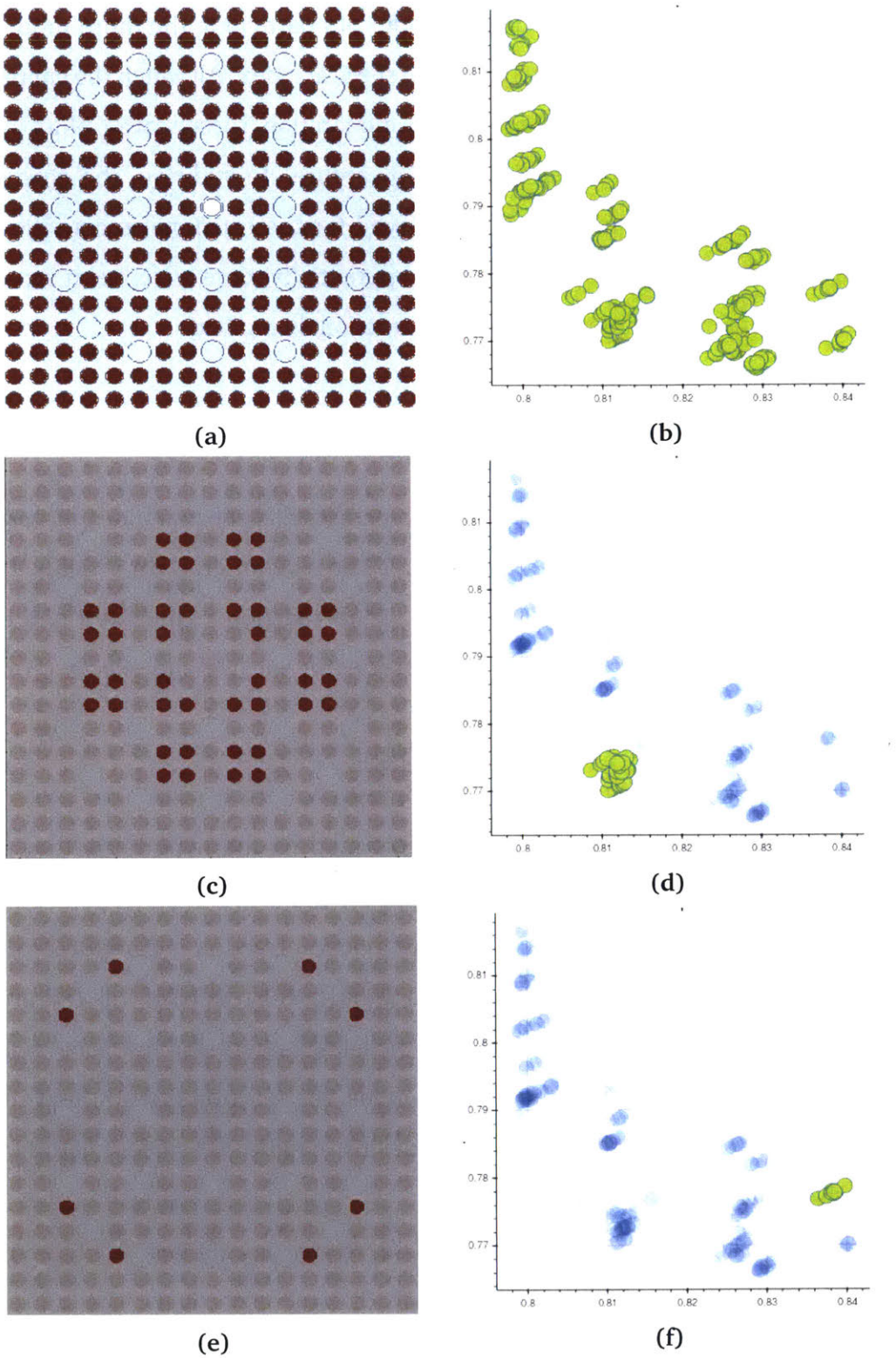


Figure 10-10: Scatter plots of the pin-wise U-238 capture (group 1 of 2) MGXS means (x) and spectral indices (y) for the 1.6% enriched fuel assembly.

are negatively correlated with the spectral index. In particular, the more moderation provided by neighboring CRGTs, the smaller the spectral index. Although the capture and fission rates both increase with the amount of moderation, these scatter plots indicate that U-235 fission increases *faster* than U-238 capture. In addition, the clusters are more spread out than the linear trends observed for the fractional reactivities. The scatter plots indicate that the spectral index is potentially more effective at separating sub-clusters within each primary cluster than fractional reactivity.

10.2.4 Reaction Fraction

In contrast to the energy-integrated fractional reactivity and spectral index features, the *reaction fraction* is specific to each energy group. The reaction fraction is defined as the ratio of each pin-wise microscopic MGXS $\hat{\sigma}_{x,i,k,g}$ for reaction x, nuclide i and energy group g to the total MGXS $\hat{\sigma}_{t,i,k,g}$ for the corresponding nuclide and energy group:

$$\hat{\tau}_{x,i,k,g} = \frac{\hat{\sigma}_{x,i,k,g}}{\hat{\sigma}_{t,i,k,g}} \tag{10.3}$$

The reaction fraction feature $\hat{\tau}_{x,i,k,g}$ is motivated by the hypothesis that spatial self-shielding effects may disproportionately impact certain reaction types (*e.g.*, U-238 capture) more than the total MGXS. Since the reaction fraction is computed for each energy group, it may be more challenging to identify clusters with this feature with "noisy" MC tally data than may be the case for the energy-integrated fractional reactivity and spectral index features. Like the spectral index, the reaction fraction may identify pins with similar flux shapes but very different flux magnitudes since it is the ratio of two pin-wise reaction rates.

The reaction fraction feature is illustrated with scatter plots in Figs. 10-11 and 10-12 for 2-group U-235 fission and U-238 capture MGXS data, respectively. The scatter plots include a single data point for each of the 264 fuel pins in the 1.6% enriched fuel assembly benchmark. The x and y coordinates correspond to the tallied MGXS means $\hat{\sigma}_{x,i,k,g}$ and reaction fractions β_k, respectively. The complete datasets are illustrated in Figs. 10-11b and 10-12b. The interactive *i*MGXS visualization tool was used to

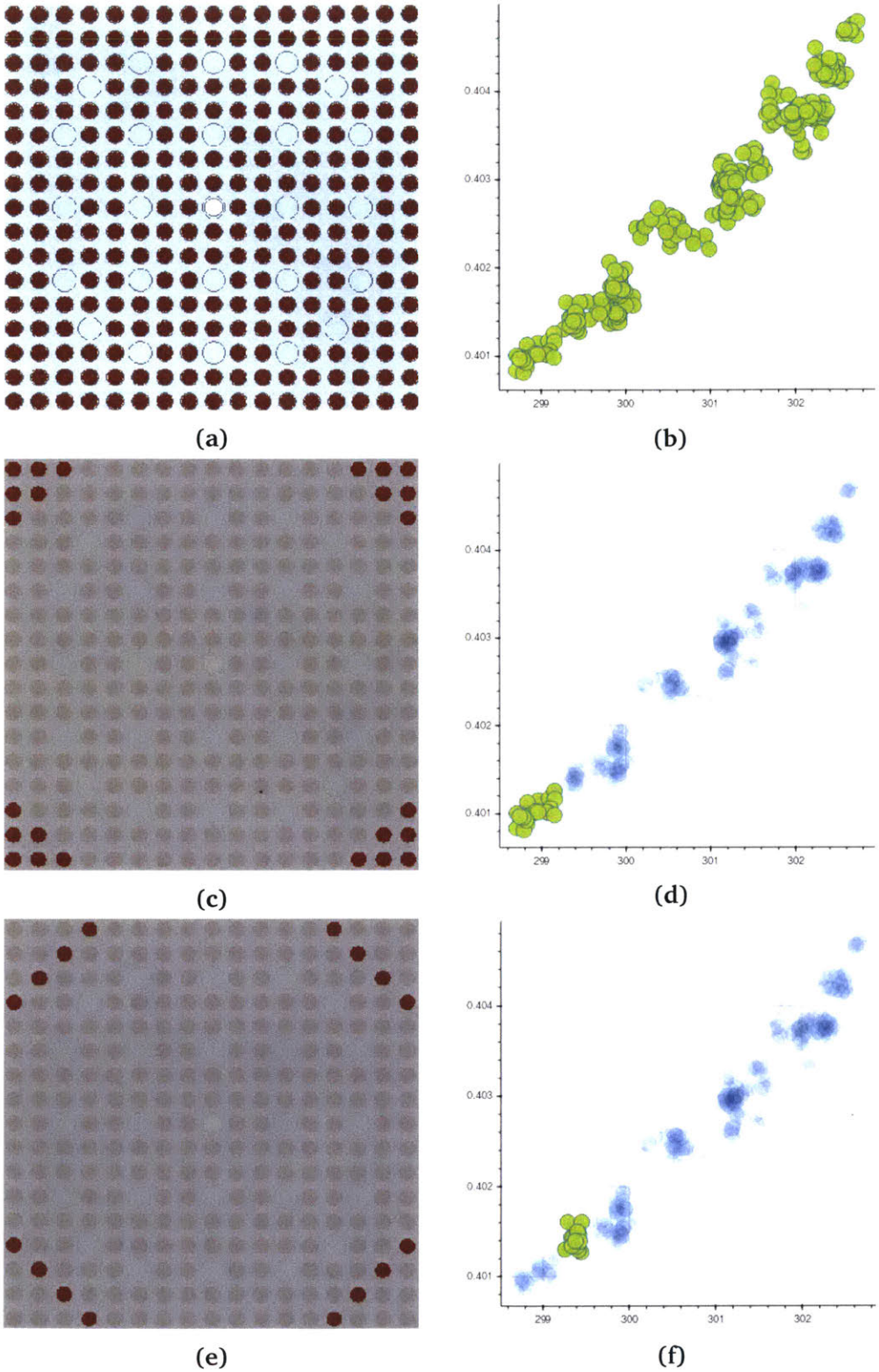


Figure 10-11: Scatter plots of the pin-wise U-235 fission (group 2 of 2) MGXS means (x) and reaction fractions (y) for the 1.6% enriched fuel assembly.

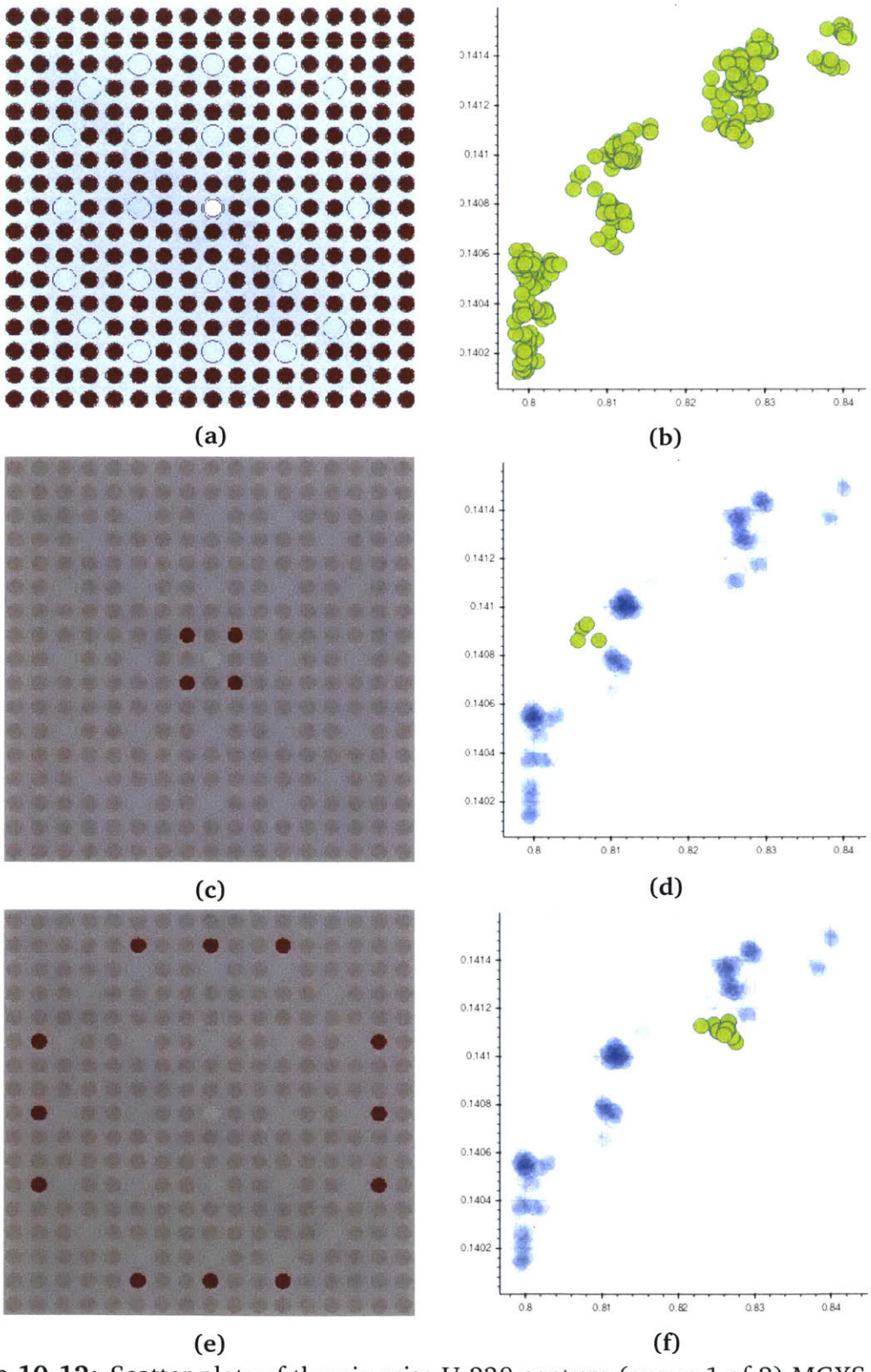


Figure 10-12: Scatter plots of the pin-wise U-238 capture (group 1 of 2) MGXS means (x) and reaction fractions (y) for the 1.6% enriched fuel assembly.

select clusters of MGXS and plot the geometry to indicate the associated fuel pins, as displayed in Figs. 10-11c to 10-11f and 10-12c to 10-12f for the fission and capture MGXS, respectively.

The figures illustrate a positive correlation between both the U-235 fission and U-238 capture MGXS and the reaction fractions (at least for this benchmark). In particular, the more moderation provided by neighboring CRGTs, the larger the reaction fraction. Although the fission, capture and total rates all increase with moderation, these scatter plots indicate that U-235 fission and U-238 capture increase *faster* than the corresponding total reaction rates with U-235 and U-238. The clusters are more spread out than the linear trends observed for the fractional reactivities, but this is likely due to the relatively larger statistical uncertainty for the energy-specific reaction fractions than the energy-integrated fractional reactivities.

10.2.5 Feature Standardization

Each of the features described in the preceding sections is assembled into feature vectors for each sample as illustrated in Fig. 10-13. From a practical standpoint, the *i*MGXS implementation in this thesis represents the sample feature vectors with a 2D `DataFrame` object from the Pandas module [58] for data processing in Python. Each column in the `DataFrame` corresponds to a feature, and each row corresponds to a sample. This `DataFrame` is the input to the feature selection stage of the *i*MGXS pipeline. However, each feature must first be *standardized* before used as an input to a predictive model.

Feature standardization – also known as *mean removal* and/or *variance scaling* – is necessary since many machine learning estimators assume that all features are centered about zero with near unit variance. In the context of *i*MGXS, the distance between samples (or clusters) in feature space will be skewed if the features are not standardized. In particular, the distance will be dominated by features with the largest magnitudes (*e.g.*, thermal U-235 fission MGXS), and conversely, insensitive to features with small magnitudes (*e.g.*, MGXS standard deviations). The canonical form of standardization *translates* each sample by subtracting the mean $\overline{\hat{f}_{j,\cdot}}$, and *scales* each sample by dividing

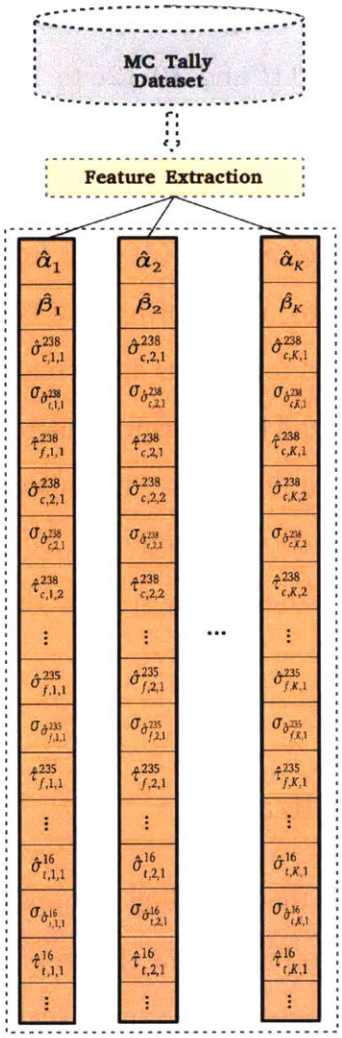


Figure 10-13: *i*MGXS feature extraction computes fractional reactivities $\hat{\alpha}_k$, spectral indices $\hat{\beta}_k$, reaction fractions $\hat{\tau}_{x,i,k,g}$, MGXS means $\hat{\sigma}_{x,i,k,g}$ and standard deviations $\sigma_{\hat{\sigma}_{x,i,k,g}}$.

by the standard deviation $\sigma_{\hat{f}_{j,\cdot}}$ of feature j:

$$\hat{f}^*_{j,k} = \frac{\hat{f}_{j,k} - \overline{\hat{f}_j}}{\sigma_{\hat{f}_{j,\cdot}}} \tag{10.4}$$

where $\hat{f}^*_{j,k}$ is the standardized feature j for fuel pin instance k. Several standardization schemes are provided by the `sklearn.preprocessing` module of the `scikit-learn` Python package for machine learning [61]. The *i*MGXS implementation in this thesis uses the `RobustScaler` to subtract the median and divide by the interquartile range

(IQR)[69] to diminish the influence of outliers on the standardization process:

$$\hat{f}^{*}_{j,k} = \frac{\hat{f}_{j,k} - \text{median}\left\{\hat{f}_{j,1}, \hat{f}_{j,2}, \ldots, \hat{f}_{j,K}\right\}}{\text{IQR}_{\hat{f}_{j,\cdot}}} \tag{10.5}$$

Features are random variables derived from MC tally data which may provide information about MGXS clustering. Feature vectors are constructed for each fuel pin instance and used as inputs for predictive clustering models.

10.3 Feature Selection

The *feature selection* stage of the data processing pipeline in Fig. 10-2 determines which features to use when training[70] a clustering model. The feature extraction stage provides many different features, some of which may be poor predictors of MGXS clusters. In addition, some features may be highly correlated and redundant as input variables to a predictive model. Feature selection attempts to identify the smallest possible subset of features necessary to achieve the desired predictive accuracy. Feature selection plays an important role in balancing the canonical *bias–variance tradeoff* in statistics and machine learning by reducing model variance, while potentially increasing model bias. In particular, feature selection is used to form simple(r) machine learning models for the following three reasons:

- **Reduce Model Complexity** – Simpler models are easier to intuit and explain
- **Reduce Generalization Error** – Simpler models reduce the risk of over-fitting
- **Reduce Training Time** – Simpler models are computationally efficient to train

Automated feature selection methods are often classified into three main categories: *filter*, *wrapper*, and *embedded* methods. *Filter methods* are agnostic to the family of

[69] The interquartile range is the difference between the first and third quartiles: $\text{IQR} = Q_3 - Q_1$.

[70] In machine learning, *training* a model refers to the process of assigning numerical values to the parameters of the model to best fit an empirical dataset.

predictive models one wishes to train and select features which provide the most information about a target variable. Correlation Feature Selection [84] is an example of a filter method which searches for the smallest subset of features which are highly correlated with the target variable, but uncorrelated with each other. In contrast to filter methods, *wrapper methods* evaluate subsets of features to determine their collective ability to minimize the generalization error of some specific family of predictive models. Wrapper methods may be computationally expensive since it entails a search of the space of possible feature subsets. In addition, wrapper methods may be at greater risk of over-fitting than filter methods since the feature selection criteria is based on model accuracy. Finally, *embedded methods* select features as an integral part of the model training procedure. Some examples of embedded methods include ℓ_1-regularization techniques and Recursive Feature Elimination [85]. The interested reader is referred to [86] or the plethora of other sources in the literature for more detailed information about automated feature selection methods.

This thesis makes a fairly limited use of feature selection techniques in its implementation of *i*MGXS. In particular, a number of relatively simple filter and wrapper methods are presented in the following sections as options to select the best features. This process is illustrated as part of the *i*MGXS data processing pipeline in Fig. 10-14. Sec. 10.3.1 presents *litmus tests* to filter the features and choose those pairings of nuclides and reaction types which are most likely to exhibit clustering. Sec. 10.3.2 introduces variance thresholding, while Sec. 10.3.3 highlights a collection of filter methods which select the highest scoring features. Sec. 10.3.4 discusses a wrapper method which selects features based on their importances for decision tree or ensemble regressors. Finally, while domain knowledge is not an automated approach to feature down-selection, it is employed by the case studies in the following chapter and is discussed in Sec. 10.3.5.

It is important to note that the methods and metrics highlighted here should not be considered an exhaustive list of options for feature selection for *i*MGXS. Furthermore, while all of the options discussed are supported in the *i*MGXS implementation for this thesis, out of practical necessity only a few of the methods are evaluated by the empirical cases studies in this and the following chapter. Future work may aim to provide a more

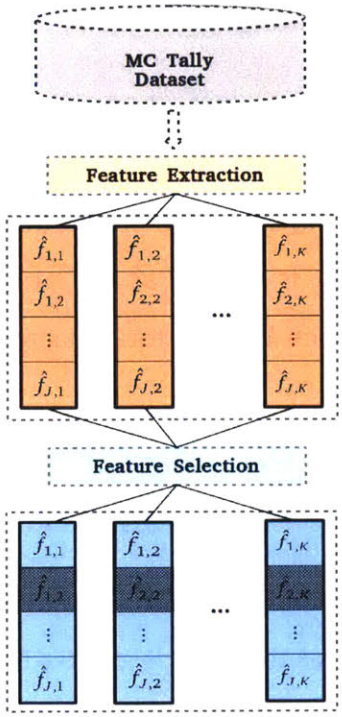


Figure 10-14: Feature selection chooses the features used to train a clustering model.

systematic appraisal of various feature selection methods for the *i*MGXS scheme.

10.3.1 Litmus Tests

Litmus tests are filter methods specifically designed to select data for particular nuclides and reaction types for clustering. The litmus tests presented here are used to down-select the MGXS mean and standard deviation features by predicting which nuclides and reaction types are most likely to exhibit clustering effects. The litmus tests are independently applied to each nuclide, energy group, and reaction type grouping. Sec. 10.3.1.1 introduces total fraction thresholding to select a nuclide for a given energy group, while Sec. 10.3.1.2 introduces reaction fraction thresholding to select a reaction type for a given nuclide and energy group. Sec. 10.3.1.3 discusses how normality tests can be used to reject features which may have been drawn from a common normal distribution.

10.3.1.1 Total Fraction Thresholding

Total fraction thresholding is a filter method used to select MGXS-based features[71] for one or more nuclides for a particular energy group g and reaction type x. The total fraction gives an indication of how much a particular nuclide contributes to the total MGXS. The *total fraction* $\hat{\omega}_{x,i,k,g}$ is defined as the ratio of a macroscopic MGXS $\hat{\Sigma}_{x,i,k,g}$ for nuclide i and fuel pin instance k to the total macroscopic MGXS for all nuclides:

$$\hat{\omega}_{x,i,k,g} = \frac{\hat{\Sigma}_{x,i,k,g}}{\sum_{i=1}^{I} \hat{\Sigma}_{t,i,k,g}} \tag{10.6}$$

The *i*MGXS implementation in this thesis simply chooses MGXS-based features for nuclide i if the mean $\overline{\hat{\omega}}_{x,i,g}$ across fuel pins is greater than some user-defined threshold. This litmus test assumes that nuclide(s) which contribute little to the total MGXS are not important to cluster. This assumption may not always be valid if some small MGXS are highly sensitive to spatial self-shielding effects. For example, if the flux is appreciably depressed in narrow resonance groups for nuclides and reaction types which are especially sensitive to spatial self-shielding (*e.g.*, U-238 capture), then the MGXS may likewise be depressed such that the total fraction drops below the threshold and the feature(s) are neglected. A relatively conservative value of 0.1 may be used for the threshold such that MGXS-based features for trace nuclides (*e.g.*, O-17, U-234) are neglected while those for U-235 and U-238 are always selected.

10.3.1.2 Reaction Fraction Thresholding

Reaction fraction thresholding is a filter method used to select MGXS-based features for one or more reaction types for a particular nuclide i and energy group g pairing. The reaction fraction $\hat{\tau}_{x,i,k,g}$ (Eqn. 10.3) gives an indication of how much a particular reaction contributes to a nuclide's total MGXS. This litmus test is employed in a similar fashion to total fraction thresholding. In particular, the *i*MGXS implementation in this

[71] MGXS-based features include the microscopic MGXS means $\hat{\sigma}_{x,i,k,g}$ and standard deviations $\sigma_{\hat{\sigma}_{x,i,k,g}}$, as well as the reaction fraction $\hat{\tau}_{x,i,k,g}$.

thesis simply chooses MGXS-based features for nuclide i and reaction x if the mean $\overline{\hat{\tau}}_{x,i,g}$ across fuel pins is greater than some user-defined threshold.

The reaction fraction thresholding litmus test assumes that reaction(s) which contribute little to a nuclide's total MGXS are not important to cluster. As was noted for total fraction thresholding, this assumption may not always be valid if some small MGXS are highly sensitive to spatial self-shielding effects. A relatively conservative value of 0.3 may be used for the threshold such that MGXS-based features for U-235 fission and U-238 capture are always selected for thermal energy groups.

10.3.1.3 Normality Tests

Normality tests select features which fail to reject the null hypothesis that the data is drawn from a normal distribution. Normality tests are a filter method for feature selection motivated by the thought experiment posed in Chap. 9 and investigated in Sec. 9.1.4 with Q-Q plots. This litmus test assumes that the realizations of a given feature are normally distributed if they are insensitive to spatial self-shielding effects. If this assumption is valid, then those features which pass a normality test do not provide any useful information about MGXS clustering and can be neglected.

The *i*MGXS implementation in this thesis uses the Shapiro-Wilk test of normality [82] computed using the Python `scipy.stats` package [59]. The Shapiro-Wilk test is applied to all samples for a given feature to compute a p-value to quantify the null hypothesis that the population came from a normal distribution. A user-defined significance level α is chosen (*e.g.*, 0.001) and compared with the Shapiro-Wilk p-value. If $p > \alpha$ then the null hypothesis that the feature is normally distributed is accepted and the feature is dropped from the dataset. Alternatively, if $p < \alpha$, then the null hypothesis is rejected and the feature is selected as a potentially good candidate to indicate MGXS clustering.

10.3.2 Variance Thresholding

Variance thresholding is a filter method which removes low variance features. The *i*MGXS implementation in this thesis uses the `VarianceThreshold` class from the

`sklearn.feature_selection` module to administer variance thresholding. It is important to note that variance thresholding must be applied to the features *prior* to standardization (see Sec. 10.2.5).

Variance thresholding presents an opportunity for a user to guide the *i*MGXS feature selection process. A user may define specialized variance thresholds for one or more features based on some prior understanding of the reactor physics relevant to the problem at hand. For example, a user may know that it is only important to model MGXS clustering if the variance of one, some or all features is greater than some collection of corresponding thresholds. If the feature variances are below the threshold, the features will be rejected and the data may be treated with a "coarse" clustering model with more data points assigned to fewer clusters, with improved statistics for the resultant track density-weighted MGXS (see Sec. 9.4.1.1).

10.3.3 Univariate Feature Selection

Univariate feature selection is a filter method which selects features based on univariate statistical tests. This thesis uses the `GenericUnivariateSelect` class from the `sklearn.feature_selection` module to administer univariate feature selection. This method individually ranks each feature according to some metric and selects some user-specified number or percentage of the highest scoring features. Univariate feature selection does not consider correlations between features, and as a result, may choose a collection of highly redundant features.

The typical statistical tests used for univariate feature selection quantify a mapping between the features and some target variable(s). The *i*MGXS implementation in this thesis uses the MGXS means $\hat{\sigma}_{x,i,k,g}$ as the target variables and builds a sequence of linear regression models based on the non-MGXS features (*e.g.*, fractional reactivities, spectral indices, etc.). Each linear regressor is scored with a p-value from an F-test which assesses the strength of the linear dependence between the non-MGXS feature and MGXS target variable(s). An alternative but equally valid approach would use the mutual information between the non-MGXS features and MGXS target variable(s) to

individually score each feature.

10.3.4 Select-from-Model Feature Importance Ranking

The *select-from-model* approach is a wrapper method for feature selection. This thesis uses the `SelectFromModel` class from the `sklearn.feature_selection` module to administer select-from-model feature selection. This approach uses a *meta-transformer* to select features based on a ranking of their importance weights. Similar to univariate feature selection, the meta-transformer defines a mapping between the features and some target variable(s). The *i*MGXS implementation in this thesis uses the MGXS means $\hat{\sigma}_{x,i,k,g}$ as the target variables, and trains multi-dimensional piece-wise constant regression models from the non-MGXS features (*e.g.*, fractional reactivities, spectral indices, etc.). For example, the `DecisionTreeRegressor` from the `sklearn.tree` module may be used to train a decision tree [87] and the best features selected by identifying the decision tree nodes which provide the greatest reduction in variance of the target variables. Alternatively, the `RandomForestRegressor` from `sklearn.ensemble` module may be used to train a random forest [88] of multiple decision tree regressors on subsets of the dataset. The random forest regressor better controls over-fitting than a single decision tree, and improves the generalization of the select-from-model feature selection method.

10.3.5 Pinch Feature Selection

Each of the preceding feature selection techniques automatically selects or rejects features based on some test criterion. An obvious alternative is to base feature selection entirely on user domain knowledge. The *i*MGXS implementation in this thesis allows users to employ *pinch feature selection* to specify a single nuclide, energy group and reaction type grouping from which to derive features. For example, the case studies in the following chapter employ the pinch method to select only those features derived from U-238 capture MGXS in group 1 of 2. This is based on the knowledge acquired from Chaps. 8 and 9 that a detailed model of MGXS clustering is necessary for accurate predictions of pin-wise U-238 capture reaction rates. Alternatively, a user may specify a nuclide,

energy group, and reaction type grouping which indicates the clustering of other MGXS which may be more important to cluster for accurate reaction rate predictions. For example, clustering of U-235 fission MGXS may be a good (*e.g.*, low uncertainty) signal of U-238 capture MGXS clustering, even if pin-wise fission rates are less sensitive to MGXS clustering than U-238 capture rates.

Feature selection is used to reduce the feature space to minimize clustering model complexity. *i*MGXS uses custom *litmus tests* to select the most promising candidate nuclides and reaction types, as well as generic techniques such as variance thresholding and select-from-model feature importance ranking.

10.4 Dimensionality Reduction

The *dimensionality reduction* stage in the data processing pipeline in Fig. 10-2 aims to reduce the number of random variables used to train a machine learning model, and is closely related to feature selection[72]. A dimensionality reduction method is a transformation $P : \mathbb{R}^J \rightarrow \mathbb{R}^T$ from the original J-dimensional feature space to a T-dimensional space with an equivalent descriptive power of the structure, or the relationships between samples, in the dataset. In general, dimensionality reduction is used in statistical clustering analysis for the following reasons:

- Reduce dataset storage requirements and model training time
- Improve model performance and reduce sensitivity to outliers
- Make datasets and/or models easier to visualize in 2D or 3D

Many common dimensionality reduction techniques perform a decomposition of the dataset matrix. These methods attempt to identify *latent* or hidden variables which adequately represent the variation of and correlation between features in a lower-dimensional vector space than the original feature space. The mapping can be represented as a basis of T functions g_t of the feature vectors $\hat{f}_k$ as illustrated in Fig. 10-15.

[72]Feature transformation can be used in place of *or* in conjunction with feature selection techniques.

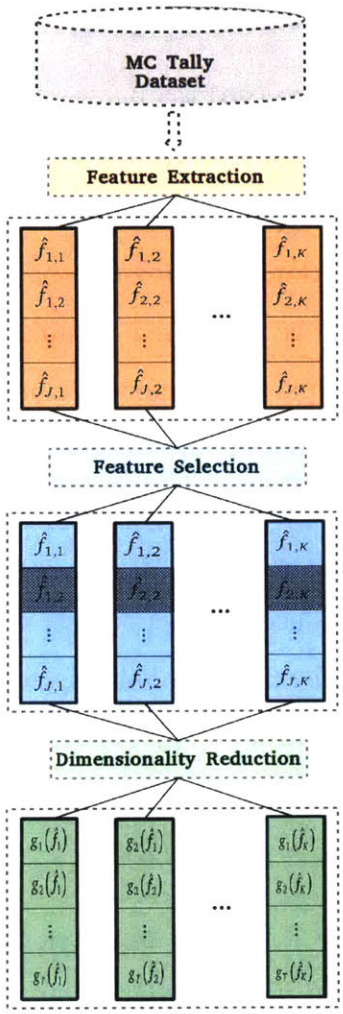


Figure 10-15: Dimensionality reduction maps the features into a new vector space.

Three matrix decomposition dimensionality reduction methods were implemented as optional elements of the *i*MGXS data processing pipeline. The Principal Component Analysis and Independent Component Analysis methods are highlighted in Sec. 10.4.1 and 10.4.2, respectively. The Factor Analysis method is summarized in Sec. 10.4.3. None of these methods are employed in the case studies of the *i*MGXS scheme applied to heterogeneous PWR benchmarks in the following chapter. Nonetheless, they are mentioned here since it is the author's opinion that they present future opportunities to improve the predictive accuracy and reduce the training time of the *i*MGXS scheme.

10.4.1 Principal Component Analysis

Principal Component Analysis (PCA) [89] is a matrix decomposition of the dataset that is perhaps the most common technique for dimensionality reduction. PCA is a *variance-centric* approach which seeks to reproduce the total variable variance in a dataset with an orthogonal linear transformation of the feature space. In particular, PCA projects the features – which may or may not be correlated – onto a linearly uncorrelated basis set of *principal components*. The principal components are linear combinations of the features that are rank-ordered by the amount of variance they account for in the original dataset. The first few principal components may be selected as the new feature space to reduce the dimensionality of the dataset while preserving most of the variance.

There are many methods to decompose a dataset into its principal components, but the most common is known as the Singular-Value Decomposition (SVD) [90]. The SVD decomposes the $K \times J$ dataset matrix $\hat{\boldsymbol{F}} = \left[\hat{\boldsymbol{f}}_1, \hat{\boldsymbol{f}}_2, \ldots, \hat{\boldsymbol{f}}_K\right]^T$ of feature vectors as follows:

$$\hat{\boldsymbol{F}} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{W}^T \tag{10.7}$$

The $\boldsymbol{U}$ and $\boldsymbol{W}$ matrices are $K \times K$ unitary matrices composed of orthogonal vectors called the *left singular vectors* and *right singular vectors*, respectively. PCA designates the right singular vectors as the principal components. The $\boldsymbol{\Sigma}$ matrix is a $K \times J$ rectangular diagonal matrix of the positive *singular values* (square roots of the eigenvalues of $\hat{\boldsymbol{F}}$) whose magnitudes are proportional to the amount of variance explained by the corresponding singular vector. Each sample $\hat{\boldsymbol{f}}_k$ in the original feature space is mapped to $\boldsymbol{t}_k$ in the vector space defined by the principal components as follows:

$$\boldsymbol{t}_k = \hat{\boldsymbol{f}}_k^T \boldsymbol{W} \approx \hat{\boldsymbol{f}}_k^T \boldsymbol{W}_L \tag{10.8}$$

The SVD defines a loss-less transformation of the dataset. However, dimensionality reduction uses the truncated $K \times L$ matrix $\boldsymbol{W}_L$ of the L largest singular values and their respective singular vectors to map the data into a lower-dimensional feature space.

This thesis implemented an option to use the `PCA` class in the `sklearn.decomposition` Python module for dimensionality reduction in the *i*MGXS pipeline. This option was

motivated by the potential for PCA to increase the signal-to-noise ratio – the effects from spatial self-shielding convolved with MC tally uncertainties – hidden within MGXS datasets by concentrating the sample variance within only a few principal components, with the remaining components dominated by the noise from MC[73]. A variant of PCA known as *kernel PCA* [91] applies the "kernel trick" with radial basis functions (RBF) or other kernels to define a non-linear projection to the principal component space. Kernel PCA may improve the performance of clustering algorithms if the clusters are defined on complex (*e.g.*, non-convex) manifolds, such as those observed in some MGXS datasets. Although the use of PCA is not further explored in this thesis, kernel PCA was shown to induce significantly different clustering model predictions than those models which were trained on datasets in the original feature space.

10.4.2 Independent Component Analysis

Independent Component Analysis (ICA) [92] is a matrix decomposition method inspired by signal processing and commonly employed in machine learning applications. ICA attempts to find *maximally independent* components to represent the relationships between samples within feature space. In particular, ICA assumes that the data can be represented using non-Gaussian and statistically independent components. In contrast to PCA which maximizes the remaining sample variance explained by each subsequent principal component, ICA maximizes the statistical independence of the components. Though ICA is designed to separate superimposed signals, the independent components may be selected as the new feature space to reduce the dimensionality of a dataset.

ICA represents the observed feature vector $\hat{f}_k$ as a linear superposition of a J^*-dimensional independent component vector s_k plus some additive white Gaussian noise:

$$\hat{f}_k = W^{-1} s_k + \epsilon \tag{10.9}$$

where it is assumed that $J^* < K$ for dimensionality reduction (*i.e.*, there are fewer components than samples). The $K \times J^*$ matrix W^{-1} is referred to as the *mixing matrix*,

[73] PCA increases the signal-to-noise ratio if the features are distributed with i.i.d. white Gaussian noise.

and ϵ is assumed to be a zero-mean and uncorrelated Gaussian noise vector. Each sample is projected into the vector space defined by the independent components as follows:

$$s_k \approx W \hat{f}_k \tag{10.10}$$

The problem of ICA is to estimate both the constant mixing matrix and the random component vectors for each sample by maximizing the mutual independence of each component. There are a variety of methods to solve for the ICA transformation, such as "Fast ICA" [93] which uses a fixed-point iteration scheme to optimize the kurtosis as the cost function that measures component independence. This thesis implemented an option to use the `FastICA` class in the `sklearn.decomposition` Python module for dimensionality reduction in the *i*MGXS pipeline. This option was motivated by the potential for ICA to identify spatial self-shielding effects (*e.g.*, adjacency to CRGTs, neighboring assembly, or baffle/reflector) as the independent components in a linear model of the clustering of pin-wise MGXS. Like PCA, the use of ICA is not further explored in this thesis, though it was shown to induce significantly different clustering model predictions than those models which were trained on datasets in the original feature space.

10.4.3 Factor Analysis

Factor Analysis (FA) [94] is a matrix decomposition method that uses a probabilistic model to describe variability among correlated features in terms of *latent* or hidden "factor" variables. The observed feature vectors are modeled as a linear combination of the latent factors along with some error or noise term(s). FA often produces factors similar to the principal components identified by PCA. However, FA can be advantageous since it does not assume that the variance within each feature is equal across the range of values taken on by other features (*i.e.*, heteroscedastic noise), which is likely not the case for MGXS features. Like PCA and ICA, the factors found by FA may be selected as the new feature space to reduce the dimensionality of a dataset.

FA introduces the continuous, random K-dimensional factor vectors $\boldsymbol{h}_k$ for each sample, and a *factor loading matrix* $\boldsymbol{W}$, to represent each feature vector $\hat{\boldsymbol{f}}_k$ as follows:

$$\hat{\boldsymbol{f}}_k = \boldsymbol{W}\boldsymbol{h}_k + \boldsymbol{\mu} + \boldsymbol{\epsilon} \tag{10.11}$$

where the vector $\boldsymbol{\mu} = \left[\overline{\hat{f}}_{1,\cdot}, \overline{\hat{f}}_{2,\cdot}, \ldots, \overline{\hat{f}}_{J,\cdot}\right]^T$ is comprised of the feature means. The noise term $\boldsymbol{\epsilon}$ is normally distributed with zero mean and a diagonal covariance matrix $\boldsymbol{\Sigma}$ such that the conditional probability of observing the sample $\hat{\boldsymbol{f}}_k$ is given by:

$$p\left(\hat{\boldsymbol{f}}_k | \boldsymbol{h}_k\right) = \mathcal{N}(\boldsymbol{W}\boldsymbol{h}_k + \boldsymbol{\mu}, \boldsymbol{\Sigma}) \tag{10.12}$$

Each sample is projected into the vector space defined by the latent factors as follows:

$$\boldsymbol{h}_k = \boldsymbol{W}^{-1}\left(\hat{\boldsymbol{f}}_k - \boldsymbol{\mu}\right) \tag{10.13}$$

A variety of algorithms may be employed to estimate the factor loading matrix, including the Expectation-Maximization (EM) algorithm [95] (see Sec. 10.5.4) which finds the maximum likelihood estimators of the multi-variate Gaussian distribution in Eqn. 10.12. This thesis implemented an option to use the `FactorAnalysis` class in the `sklearn.decomposition` Python module for dimensionality reduction in the *i*MGXS scheme. Similar to ICA, this option was motivated by the potential for FA to identify spatial self-shielding effects (*e.g.*, adjacency to CRGTs, neighboring assembly, or baffle/reflector) as the factors in a linear model of the clustering of pin-wise MGXS. Like PCA and ICA, the use of FA is not further explored in this thesis, though it was shown to induce significantly different clustering model predictions than those models which were trained on datasets in the original feature space.

Dimensionality reduction represents samples in a lower-dimensional vector space with the same descriptive power as the original features to reduce training time and dataset storage requirements, and improve clustering model performance. Three matrix decomposition methods for dimensionality reduction – PCA, ICA and FA – are presented for optional (future) use in the *i*MGXS scheme, but are not employed in any of the case studies in this thesis.

10.5 Training a Predictor

The *predictor training* stage of the data processing pipeline in Fig. 10-2 builds a predictive model and assigns cluster labels ℓ_k to each fuel pin instance. An unsupervised clustering algorithm is used to infer structure from the dataset of feature vectors generated by the feature extraction, feature selection and dimensionality reduction stages. This process attempts to identify which fuel pin instances should be grouped together with the same MGXS in order to best predict reaction rate spatial distributions.

Although machine learning researchers have developed many different clustering algorithms over the last few decades, only one thing is agreed upon by the community: "clustering is in the eye of the beholder" [96]. Perhaps the best way to differentiate between algorithms for cluster analysis is their implicit or explicit assumption(s) of what constitutes a "cluster." In general, all clustering algorithms aim to group samples such that those assigned to a cluster are more similar than they are to the samples in other clusters. Beyond that, no precise definition of what makes a "cluster" is broadly agreed upon. Instead, each clustering algorithm is designed to solve an optimization problem which may minimize the distances between samples within each cluster, identify high density regions of feature space, fit a statistical distribution to a given dataset, among other strategies. For practical reasons, only a select few clustering algorithms are evaluated within the *i*MGXS implementation developed for this thesis.

Clustering algorithms are often categorized according to the *(multi-)objective function* they seek to minimize and the *assumptions* they make. This thesis evaluates four different

clustering algorithms within the *i*MGXS data processing pipeline. These four algorithms are associated within one of the following three categories of clustering algorithms:

- **Centroid models** partition feature space using cluster central vectors
- **Connectivity models** build dendrograms to relate neighboring samples
- **Distribution models** build probability distributions to fit the data

Centroid models make assumptions about the size and convexity of each cluster and optimize the choice of central vectors used to represent the center of each cluster. *Connectivity models* assume that the similarity between samples is inversely related to the distance separating them and build hierarchical models to locally optimize sample connectivity. *Distribution models* assume that the observed dataset was generated from a well-defined probability distribution and optimally select parameters to fit a distribution model to best predict the data. It is important to note that the various algorithms within each category may differ according to how they associate each sample with a cluster. Some algorithms use *hard assignments* which designate each sample to one particular cluster, while other algorithms use *soft assignments* to indicate the likelihood of each sample belonging to each cluster.

Depending on the clustering algorithm, many different predictors may be built for different model parameters – including the number of clusters to find, a sample distance or similarity function, a density threshold, etc. – as is illustrated in the context of the *i*MGXS pipeline in Fig. 10-16. The "best" predictor is identified using model selection techniques as discussed in Sec. 10.6. The following sections provide synopses of the four clustering algorithms evaluated in this thesis, including the assumptions they make and the objective functions they optimize. Sec. 10.5.1 describes the k-means algorithm, a commonly used centroid model which is fast and scalable for large datasets. Sec. 10.5.2 outlines the algorithm for agglomerative clustering and Sec. 10.5.3 presents the BIRCH algorithm – two connectivity models which are widely used to model data that may lie on manifolds with with irregular (*i.e.*, non-convex) shapes. Sec. 10.5.4 highlights the Gaussian Mixture Model (GMM), a distribution model which fits the data to a mixture of one or more Gaussian probability distributions. Each of these algorithms

is implemented in the `scikit-learn` Python package and employed as needed in the *i*MGXS data processing pipeline.

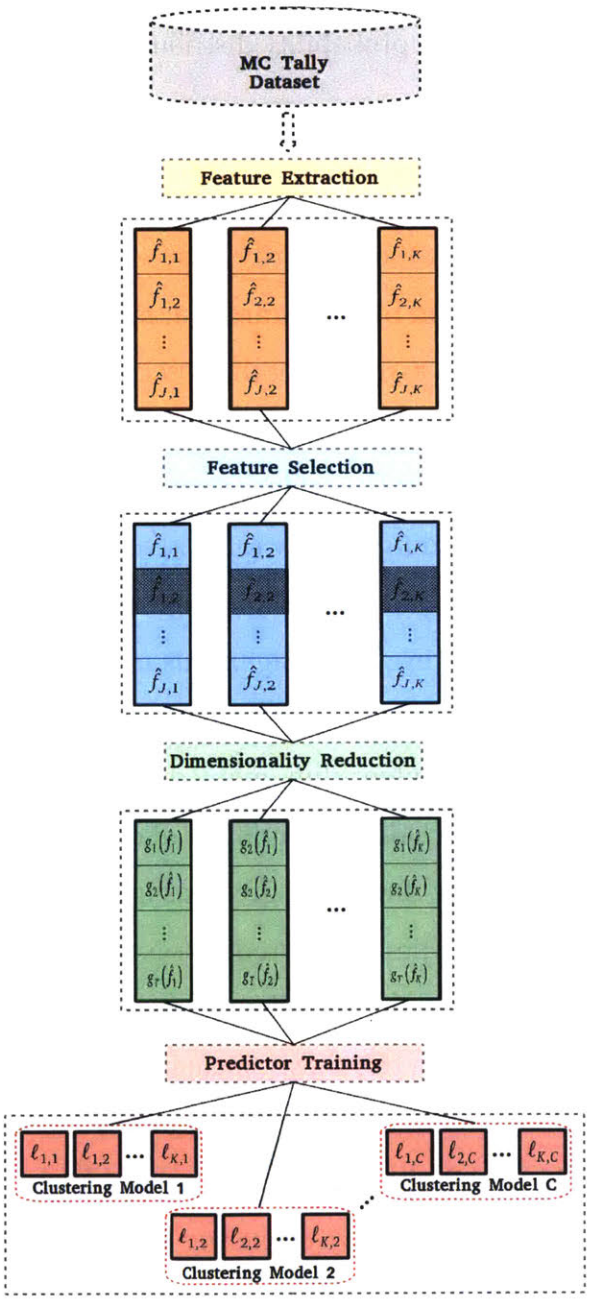


Figure 10-16: Each clustering model assigns labels to each sample.

10.5.1 k-means Clustering

The k-means algorithm [97, 98] constructs a Voronoi tesselation of feature space composed of k Voronoi cells[74]. A Voronoi tesselation is a partitioning of feature space into convex regions based on the distance to a discrete set of "seed" points. In the context of the k-means, each seed is the centroid of a cluster. Each sample is assigned a cluster label according to the Voronoi cell that contains it. The k-means algorithm attempts to minimize the within-cluster sum-of-squares, also known as the *inertia*:

$$\sum_{k=0}^{K} \min_{\mu_m \in \mathbb{C}} \|\hat{f}_k - \mu_m\|^2 \tag{10.14}$$

where μ_m and $\hat{f}_k$ are the J-dimensional centroid of cluster m and feature vector for fuel pin instance k in feature space, respectively. Each of the cluster centroids is stored in set $\mathbb{C} = \{\mu_1, \mu_2, \ldots, \mu_M\}$, where M is used in place of k to denote the total number of clusters to avoid confusion with the k used to index the fuel pin instances.

The key steps of the k-means algorithm are summarized in Alg. 10-1. The first step initializes M cluster centroids in feature space. A variety of techniques exist to initialize the centroids, including random selection of M samples as centroids, or the k-means++ method [99] which uniformly disperses the centroids across high density regions of feature space. The algorithm then iterates over *assignment* and *update* steps. The assignment step loops over each sample and assigns it to the nearest cluster centroid:

$$\mathbb{S}_m^{(t)} = \left\{ \hat{f}_k \ : \ \|\hat{f}_k - \mu_m^{(t)}\|^2 \leq \|\hat{f}_k - \mu_n^{(t)}\|^2 \quad \forall n,\ 1 \leq n \leq M \right\} \tag{10.15}$$

where $\mathbb{S}_m^{(t)}$ represents cluster m at iteration t as the set of nearby samples. The update step moves each cluster centroid to the mean of those samples assigned to it:

$$\mu_m^{(t+1)} = \frac{1}{|\mathbb{S}_m^{(t)}|} \sum_{\hat{f}_k \in \mathbb{S}_m^{(t)}} \hat{f}_k \tag{10.16}$$

The k-means iteration proceeds until the cluster centroids stop moving as defined by

[74] It should be noted that the "k" in k-means should not be confused for the k commonly used throughout this thesis to index fuel pin instances.

some criterion, such as the maximum square distance between successive centroids:

$$res = \max_{1 \leq m \leq M} \|\boldsymbol{\mu}_m^{(t+1)} - \boldsymbol{\mu}_m^{(t)}\|^2 \tag{10.17}$$

Algorithm 10-1: k-means Clustering Algorithm

1: Initialize M centroids $\{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_M\}$
2: **while** centroid residual is not converged **do**
3: Assign each sample $\hat{\boldsymbol{f}}_k$ to nearest centroid $\boldsymbol{\mu}_m$ ▷ Eqn. 10.15
4: Update centroids to mean of samples assigned to each centroid ▷ Eqn. 10.16
5: Compute centroid residuals ▷ Eqn. 10.17
6: **end while**

This thesis employs the k-means algorithm implemented in the `KMeans` class of the `sklearn.cluster` module in the `scikit-learn` Python package. The Euclidean distance metric is used to compare the similarity of samples in feature space, and the k-means++ algorithm is used to initialize cluster centroids. Two attractive features of k-means are its simplicity and speed. It has a linear computational complexity of $\mathcal{O}(K)$ for K fuel pin instances. The algorithm is relatively easy to parallelize since it primarily consists of independent calculations of the distances between each sample and cluster centroid. In addition, out-of-core variants of the algorithm exist for very big datasets such as those that might be realized if the *i*MGXS scheme were used to model a full core 3D PWR. For example, mini-batch k-means [100] is an out-of-core version of the algorithm and is implemented as `MiniBatchKMeans` in the `sklearn.cluster` module.

The k-means algorithm makes a number of assumptions about what constitutes a "cluster" that have consequences for the types of clusters that it finds. For example, k-means optimizes the inertia which implicitly assumes that clusters are convex and isotropic, and in particular, that clusters are spherical and of similar size in feature space. As a result, k-means is not well-suited to identify clusters with irregular shapes without special distance metrics and/or dimensionality reduction techniques. Another complication is that while k-means is provably guaranteed to converge[75], it only finds local minima of the inertia. Furthermore, the performance is especially sensitive to the placement of the initial centroids. The k-means++ algorithm is designed to mitigate

[75]This is only true if the Euclidean distance metric is used.

this by ensuring that the centroids are uniformly and randomly dispersed across the important regions of feature space. Nonetheless, since the algorithm is fast, it is common to run k-means many times with different initial cluster centroids and choose the solution with the smallest inertia. Finally, k-means requires a user to define the number of clusters M to find within the dataset. In practice, the optimal value of M is unknown *a priori*, and model selection techniques must be used to select from a sequence of cluster models trained for different values of M, as discussed in Sec. 10.6.

10.5.2 Agglomerative Clustering

The agglomerative clustering algorithm [101] constructs a dendrogram[76] representing the similarities between samples. Agglomerative clustering is a bottom-up method from the more general class of *hierarchical clustering* algorithms, which also includes the top-down *divisive* clustering method. In contrast to k-means which attempts to optimize the global inertia metric, agglomerative clustering is a greedy algorithm which seeks to *locally* minimize a linkage criterion between pairs of samples at each step. In particular, agglomerative clustering builds a dendrogram to represent a dataset by successively *merging* pairs of data points into clusters.

The key steps of the agglomerative clustering algorithm are summarized in Alg. 10-2. The first step initializes a collection $\mathbb{C}$ of clusters $\mathbb{S}_k$, where each sample constitutes a singleton cluster. The algorithm then iterates to construct a dendrogram representing M clusters. On each iteration, the *affinities* or *linkage criteria* between each pair of clusters is computed. Many different metrics for cluster affinities exist, including the *Ward linkage criterion* [102] given by $d_{a,b}$:

$$d_{a,b} = d(\mathbb{S}_a, \mathbb{S}_b) = \sum_{\hat{f}_k \in \mathbb{S}_{a+b}} \|\hat{f}_k - \mu_{a+b}\|^2 \tag{10.18}$$

The Ward criterion in Eqn. 10.18 measures the similarity between clusters a and b by considering the cluster $\mathbb{S}_{a+b}$ that would be formed if clusters $\mathbb{S}_a$ and $\mathbb{S}_b$ were combined. In particular, the Ward criterion computes the variance of the samples $\hat{f}_k$ in the combined

[76] A tree-like diagram used to represent hierarchical relationships between objects.

Algorithm 10-2: Agglomerative Clustering Algorithm

1: Initialize collection $\mathbb{C}$ of K singleton clusters: $\mathbb{S}_k = \{\hat{f}_k\}$
2: **while** there are too many clusters: $|\mathbb{C}| > M$ **do**
3: Compute pairwise affinities between clusters ▷ Eqn. 10.18
4: Find pair of clusters $\mathbb{S}_a$ and $\mathbb{S}_b$ that are closest
5: Merge clusters $\mathbb{S}_a$ and $\mathbb{S}_b$ into new cluster $\mathbb{S}_{a+b}$
6: Remove $\mathbb{S}_a$ and $\mathbb{S}_b$ from $\mathbb{C}$
7: Add $\mathbb{S}_{a+b}$ to $\mathbb{C}$
8: **end while**

cluster about its centroid $\boldsymbol{\mu}_{a+b}$:

$$\boldsymbol{\mu}_{a+b} = \frac{1}{|\mathbb{S}_{a+b}|} \sum_{\hat{f}_k \in \mathbb{S}_{a+b}} \hat{f}_k \tag{10.19}$$

The Ward criterion minimizes the sum-of-squared differences within all clusters, similar to the variance minimization performed by the k-means algorithm. Other common linkage criteria include the *complete-linkage*, *single-linkage*, and *average-linkage* criteria. The distance metric used to compare the separation between clusters is specified independent of the linkage criterion and may include the Manhattan distance (ℓ_1 norm), Euclidean distance (ℓ_2 norm), among others. This thesis employs the `AgglomerativeClustering` class of the `sklearn.cluster` module in the `scikit-learn` Python package. The Ward linkage criterion and Euclidean distance metric are used to evaluate cluster affinities at each step of the algorithm.

The agglomerative clustering algorithm makes no explicit assumption about the shape or size of clusters. As a result, it may outperform k-means for datasets with clusters with non-convex shapes. The linkage criterion as well as the distance metric have a large impact on the shapes of clusters found by the algorithm[77]. In contrast to k-means, agglomerative clustering does not depend on any starting conditions. However, these advantages come at a cost since the computational complexity of the algorithm is $\mathcal{O}(K^2 \log(K))$ which is too slow for large datasets. *Connectivity constraints* may be employed to improve the performance given some domain knowledge about the local

[77] Agglomerative clustering is known to exhibit a "rich get richer" behavior with highly uneven cluster sizes, which may be exacerbated for certain linkage criteria.

structure, such as a nearest neighbors graph. Finally, similar to the k-means algorithm, agglomerative clustering requires a user to define the number of clusters M to find within the dataset. In practice, the optimal value of M is unknown *a priori*, and model selection techniques must be used to select from a sequence of cluster models trained for different values of M, as discussed in Sec. 10.6.

10.5.3 BIRCH

The Balanced Iterative Reducing and Clustering Using Hierarchies (BIRCH) clustering algorithm [103] is an out-of-core hierarchical clustering algorithm designed for large datasets. BIRCH reduces the dataset to a series of *subclusters* contained within the CF leaf nodes, which are themselves clustered using a standard clustering algorithm such as k-means or agglomerative clustering. BIRCH is often considered a *data reduction* method which represents a dataset as a height-balanced Characteristic Feature Tree (CFT) composed of Characteristic Feature (CF) nodes. This approach is advantageous for "big data" settings since it does not require the equal inspection of all data points to make clustering decisions at each step as do many other algorithms (*i.e.*, k-means, agglomerative clustering). Instead, BIRCH makes local clustering decisions when inserting a sample into the CFT. After each node's Characteristic Feature has been updated and the CFT rebalanced, the sample may be discarded (*i.e.*, no longer stored in main memory).

The CFT is governed by two user-defined parameters: the threshold T representing the maximum *radius* of a subcluster, and the branching factor B specifying the maximum number of *children* per CF node[78]. The radius R_{a_i} of the i^{th} subcluster $\mathbb{S}_{a_i}$ contained by CF node $\mathbb{N}_a$ is defined as the average squared distance of each sample $\hat{\boldsymbol{f}}_k$ assigned to its i^{th} subcluster from its centroid $\boldsymbol{\mu}_{\mathbb{N}_{a_i}}$:

$$R_{a_i} = \left(\frac{1}{|\mathbb{S}_{a_i}|} \sum_{\hat{\boldsymbol{f}}_k \in \mathbb{S}_{a_i}} \left(\hat{\boldsymbol{f}}_k - \boldsymbol{\mu}_{a_i} \right)^2 \right)^{\frac{1}{2}} \tag{10.20}$$

[78] A CF node's *children* are the CF nodes that it points to at the next level in the Characteristic Feature Tree.

where the subcluster centroid $\boldsymbol{\mu}_{\mathbb{N}_{a_i}}$ is computed by Eqn. 10.16. The branching and threshold factors collectively govern the height-balancing of the tree, with the branching factor limiting the growth rate of new CF nodes and the threshold upper bounding the length scale of the subclusters found by the algorithm.

The Characteristic Feature nodes can be distinguished between terminal (leaf) and non-terminal nodes. Each CF node represents a sequence of at most B subclusters, and holds the information necessary to represent the subclusters without explicitly storing the samples assigned to each subcluster. Each CF node $\mathbb{N}_a$ contains a Characteristic Feature vector CF_{a_i} for its i^{th} subcluster:

$$CF_{a_i} = \left[|\mathbb{S}_{a_i}|, LS_{a_i}, SS_{a_i} \right]^T \tag{10.21}$$

The first element is the number of samples within the subcluster. The second and third elements are referred to as the *linear sum* LS_{a_i} and *squared sum* SS_{a_i} of the samples assigned to the i^{th} subcluster, and are expressed as follows:

$$LS_{a_i} = \sum_{\hat{\boldsymbol{f}}_k \in \mathbb{S}_{a_i}} \hat{\boldsymbol{f}}_k \tag{10.22}$$

$$SS_{a_i} = \sum_{\hat{\boldsymbol{f}}_k \in \mathbb{N}_{a_i}} \hat{\boldsymbol{f}}_k^2 \tag{10.23}$$

The linear and squared sums are all that is needed to uniquely and exactly determine the nearest subcluster to a sample, and to compute the subcluster radius. Each non-terminal CF node contains a list (at most B long) of pointers $\mathbb{N}_{a_i}$ and the CF vectors CF_{a_i} for each of its children. Each terminal CF node a contains a list (at most B long) with the CF vectors CF_{a_i} of each of its subclusters, each of which must have a radius R_{a_i} less than the threshold.

The key steps of the BIRCH clustering algorithm are summarized in Alg. 10-3. BIRCH consists of a primary loop over all samples to build the Characteristic Feature Tree, and typically only needs to make a single pass through a dataset. For each sample, BIRCH starts at the root node and proceeds to traverse the CFT, inserting the sample into each

Algorithm 10-3: BIRCH Clustering Algorithm

Initialize an empty root node $\mathbb{N}_r$
for each sample k **do**

Insert sample into CFT
Initialize node: $\mathbb{N} \leftarrow \mathbb{N}_r$
for each level of CFT (root-to-leaf) **do**
Compute radii for each subcluster if merged with sample ▷ Eqn. 10.20
Find subcluster $\mathbb{S}_{a_i}$ with smallest post-merged radius
if $\mathbb{N}$ is a terminal node **then**
Add sample to $\mathbb{S}_{a_i}$ and update CF_{a_i} ▷ Eqn. 10.24
else
Move to child $\mathbb{N}_{a_i}$ of subcluster: $\mathbb{N} \leftarrow \mathbb{N}_{a_i}$
end if
end for

Balance terminal node
if $R_{a_i} > T$ **then** ▷ Eqn. 10.20
Split subcluster $\mathbb{S}_{a_i}$ into $\mathbb{S}_{a_{i,1}}$ and $\mathbb{S}_{a_{i,2}}$
Add subclusters $\mathbb{S}_{a_{i,1}}$ and $\mathbb{S}_{a_{i,2}}$ to node $\mathbb{N}_{a_i}$
Remove subcluster $\mathbb{S}_{a_i}$ from node
if $|\mathbb{N}| > B$ **then**
Split node $\mathbb{N}$ into nodes $\mathbb{N}_1$ and $\mathbb{N}_2$
end if
end if

Rebalance CFT
for each level of CFT (leaf-to-root) **do**
if a child node was split **then**
Remove child and insert two new children
if $|\mathbb{N}| > B$ **then**
Split node $\mathbb{N}$ into nodes $\mathbb{N}_1$ and $\mathbb{N}_2$
end if
else
Update CF vector for child with sample ▷ Eqn. 10.24
end if
Move to parent $\mathbb{N}_p$: $\mathbb{N} \leftarrow \mathbb{N}_p$
end for

end for

Input terminal subclusters to standard clustering algorithm ▷ Algs. 10-1, 10-2

CF node and/or height-balancing the tree by creating new nodes as dictated by the threshold and branching factor. The BIRCH algorithm constructs the CFT by leveraging the *Characteristic Feature Additivity Theorem* [103], which allows subcluster $\mathbb{S}_{a_i}$ to be merged with the singleton subcluster $\mathbb{S}_k$ for sample k without loss of information:

$$CF_{a_i+k} = CF_{a_i} + CF_k = \left[|\mathbb{S}_{a_i}|+|\mathbb{S}_k|, LS_{a_i} + LS_k, SS_{a_i} + SS_k\right]^T \tag{10.24}$$

This thesis employs the BIRCH clustering algorithm implemented in the `Birch` class of the `sklearn.cluster` module in the `scikit-learn` Python package. The terminal subclusters were clustered using the `AgglomerativeClustering` from the `scikit-learn` package, with the Ward linkage criterion and Euclidean distance metric (see Sec. 10.5.2). A branching factor of 50 and threshold of 0.5 were used; future work may seek to identify the most appropriate values for B and T to cluster MGXS datasets.

Like agglomerative clustering, the BIRCH clustering algorithm makes no explicit assumption about the shape or size of clusters. As a result, it may out-perform k-means for datasets with clusters with non-convex shapes. In contrast to k-means and agglomerative clustering, BIRCH is local and does not require an explicit scan of all data points and existing clusters to make clustering decisions. As a result, BIRCH has a computational complexity that approaches $\mathcal{O}(K)$ for large datasets. Furthermore, unlike k-means, BIRCH does not depend on any starting conditions. In addition, the user does not necessarily need to specify the number of clusters to find since the terminal subclusters can simply be read from the final CFT. However, the commonly used variant of BIRCH employed by this thesis does require the number of clusters M to be specified for use in a final step which clusters the terminal subclusters in the CFT. As a result, model selection techniques must be used to select from a sequence of cluster models trained for different values of M, as discussed in Sec. 10.6.

It should be noted that the datasets clustered in this thesis may be stored in memory on most modern computing systems, and do not require the use of an out-of-core algorithm like BIRCH. Nevertheless, BIRCH may be useful if future efforts use *i*MGXS to cluster MGXS datasets for full core 3D reactor benchmarks with orders of magnitude

more samples (*e.g.*, fuel pin instances with axial subdivisions). In addition, the size of the MGXS datasets may substantially grow if new features are employed by *i*MGXS in the future. Finally, BIRCH may be used to cluster MGXS datasets on memory-limited high-performance hardware such as Graphical Processing Units (GPUs).

10.5.4 Gaussian Mixture Models

Gaussian Mixture Models (GMMs) are a commonly used distribution model approach to clustering [104]. The GMM is a machine learning algorithm from the broader class of *generative models* which attempt to describe the probability distribution from which a dataset was generated (assuming one exists). The GMM may be thought of as a generalization of the k-means algorithm to include information about the covariance between features. The distribution model approach assumes that each sample was i.i.d. generated from one of a finite number of component Gaussian distributions which comprise the mixture. The model assumes that each sample contains a *latent variable*[79] that indicates the identity of the component distribution from which it was drawn. In the context of *i*MGXS, the latent variables are the cluster labels ℓ_k for each fuel pin instance k and must be inferred by fitting a GMM to the MGXS dataset.

The generic GMM is parameterized by the number of components (or clusters) M, as well as the mixture weights π_m, means $\boldsymbol{\mu}_m$ and covariance matrices $\boldsymbol{\Sigma}_m$ for each component $m \in \{1, 2, \ldots, M\}$, and is expressed as:

$$p\left(\hat{\boldsymbol{f}}_k|\boldsymbol{\theta}\right) = \sum_{m=1}^{M} \pi_m \mathcal{N}\left(\hat{\boldsymbol{f}}_k; \boldsymbol{\theta}\right) = \sum_{m=1}^{M} \pi_m \mathcal{N}\left(\hat{\boldsymbol{f}}_k; \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m\right) \tag{10.25}$$

where each J-dimensional Gaussian component distribution is given by:

$$\mathcal{N}\left(\hat{\boldsymbol{f}}_k; \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m\right) = \frac{1}{(2\pi)^{J/2}|\boldsymbol{\Sigma}_m|^{1/2}} \exp\left\{-\frac{1}{2}\left(\hat{\boldsymbol{f}}_k - \boldsymbol{\mu}_m\right)^T \boldsymbol{\Sigma}_m^{-1}\left(\hat{\boldsymbol{f}}_k - \boldsymbol{\mu}_m\right)\right\} \tag{10.26}$$

The generic GMM is illustrated in Fig. 10-17 with the *plate notation* commonly used

[79] A hidden variable which is not observed within a feature vector.

by the machine learning community to describe complex probability distributions [105]. In summary, small rectangles represent unknown model parameters, unshaded circles describe unobserved random variables, and shaded circles signify observed random variables. The dimensionality of non-scalar variables are indicated in brackets inside each square and circle. Large rectangular *plates* group subgraphs of variables which repeat together according to the repetition numbers in the bottom right corners of each plate (*e.g.*, M and K). The inter-dependences of each model parameter and random variable within the graph are indicated by directed arrows. Categorical variables, such as the cluster labels ℓ_k, are related using a squiggly arrow with a T junction.

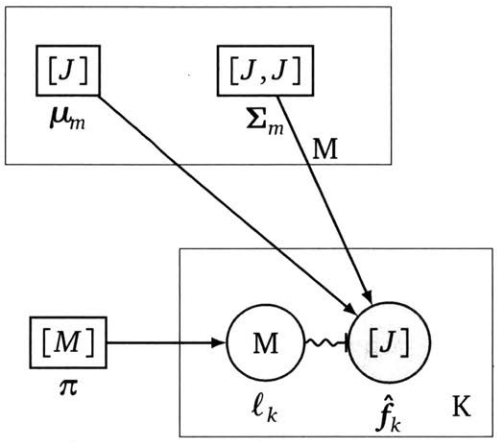


Figure 10-17: Plate notation for a Gaussian Mixture Model.

Algorithm 10-4: Expectation-Maximization Algorithm for GMMs

1: Initialize means $\left\{\boldsymbol{\mu}_1^{(0)}, \boldsymbol{\mu}_2^{(0)}, \ldots, \boldsymbol{\mu}_m^{(0)}\right\}$ ▷ Alg. 10-1
2: Initialize soft weights $\left\{h_{1,1}^{(1)}, h_{1,2}^{(1)}, \ldots, h_{M,K}^{(1)}\right\}$
3: Initialize component weights $\left\{\pi_1^{(1)}, \pi_2^{(1)}, \ldots, \pi_m^{(1)}\right\}$ ▷ Eqn. 10.27
4: Initialize component covariances $\left\{\boldsymbol{\Sigma}_1^{(1)}, \boldsymbol{\Sigma}_2^{(1)}, \ldots, \boldsymbol{\Sigma}_m^{(1)}\right\}$ ▷ Eqn. 10.28
5: **while** model parameters are not converged **do**
6: Expectation Step: Update soft weights ▷ Eqn. 10.29
7: Maximization Step: Update mixture parameters ▷ Eqns. (10.27) to (10.28)
8: **end while**
9: Compute cluster labels from MAP estimators[80] ▷ Eqn. 10.31

The number of mixture components M may either be specified by the user or inferred using Variational Bayesian methods. The means $\boldsymbol{\mu}_m$ and variances $\boldsymbol{\Sigma}_m$ are *a priori*

[80]The maximum a posteriori (MAP) estimate is the mode of the GMM distribution.

unknown and must be inferred from the data. These parameters cannot be solved for directly and must be approximated by an iterative algorithm such as *Expectation-Maximization* (EM) [95] as summarized in Alg. 10-4. The algorithm introduces *soft weights* $h_{m,k}$ which signify the probability that sample k was drawn from mixture component m, and are normalized to unity for each sample. The EM algorithm finds the maximum likelihood parameters for $\boldsymbol{\mu}_m$ and $\boldsymbol{\Sigma}_m$ on each iteration t and repeats until convergence.

The first step of the EM algorithm initializes the means $\boldsymbol{\mu}_m$ for the M mixture components. A variety of approaches exist to initialize the means; it is common to use the k-means algorithm (Sec. 10.5.1) and extract the cluster centroids for this purpose. Next, each sample k is associated with the nearest component mean $\boldsymbol{\mu}_m$ by setting soft weight $h_{m,k}$ to unity and all others to zero. Given the initial component means and soft weights, the mixture weights π_m may be computed for each component m from the following normalization:

$$\pi_m^{(t+1)} = \frac{1}{K} \sum_{k=1}^{K} h_{m,k}^{(t)} \tag{10.27}$$

The covariance matrices $\boldsymbol{\Sigma}_m$ for mixture component m are then computed from the component means and soft weights as follows:

$$\boldsymbol{\Sigma}_m^{(t+1)} = \frac{\sum_{k=1}^{K} h_{m,k}^{(t)} \left[\hat{\boldsymbol{f}}_k - \boldsymbol{\mu}_m^{(t+1)}\right]\left[\hat{\boldsymbol{f}}_k - \boldsymbol{\mu}_m^{(t+1)}\right]^T}{\sum_{k=1}^{K} h_{m,k}^{(t)}} \tag{10.28}$$

The EM algorithm then proceeds to iterate between *expectation* and *maximization* steps. The *expectation step* updates the soft weights $h_{m,k}$ with the *posterior probabilities* that each sample k was generated from mixture component m:

$$h_{m,k}^{(t)} = \frac{\pi_m^{(t)} \mathcal{N}\left(\hat{\boldsymbol{f}}_k; \boldsymbol{\mu}_m^{(t)}, \boldsymbol{\Sigma}_m^{(t)}\right)}{\sum_{m=1}^{M} \pi_m^{(t)} \mathcal{N}\left(\hat{\boldsymbol{f}}_k; \boldsymbol{\mu}_m^{(t)}, \boldsymbol{\Sigma}_m^{(t)}\right)} \tag{10.29}$$

The *maximization step* updates the mixture parameters by first computing the new component means as the average of all samples biased by the soft weights:

$$\boldsymbol{\mu}_m^{(t+1)} = \frac{\sum_{k=1}^{K} h_{m,k}^{(t)} \hat{\boldsymbol{f}}_k}{\sum_{k=1}^{K} h_{m,k}^{(t)}} \tag{10.30}$$

The component weights and covariances are updated according to Eqns. 10.27 and 10.28, respectively. The iteration terminates once the GMM parameters converge to within a user-defined tolerance. The mixture or cluster labels ℓ_k for each sample are inferred using *maximum a posteriori* probability (MAP) estimation from the final soft weights:

$$\ell_k = \underset{m}{\text{argmax}}\ h_{m,k}^{(T)} \tag{10.31}$$

The EM algorithm is guaranteed to monotonically increase the likelihood function of the observed data, but may not necessarily find the global optimum. The algorithm is only guaranteed to converge to a local maximum of the likelihood function. The converged solution is highly dependent on the initial component means, though some heuristics, such as the use of k-means to initialize the means, may make the algorithm more robust against converging to spurious local maxima. Nevertheless, it is common to run EM many times with different initial component means and choose the solution with the maximum likelihood. This thesis employs GMMs fit with the EM algorithm implemented in the `GaussianMixture` class of the `sklearn.mixture` module in the `scikit-learn` Python package. A full or dense covariance matrix is fit during the process; approximations such as diagonal or spherical covariance matrices may alternatively be used to tractably model very large MGXS datasets.

Although the GMM does not make assumptions about the size or shape of clusters, it does assume that each cluster is normally distributed. In addition, the EM algorithm employed by this thesis does require the number of clusters M to be specified. As a result, model selection techniques must be used to select from a sequence of cluster models trained for different values of M, as discussed in Sec. 10.6. Alternatively, Variational

Bayesian estimation methods developed for GMMs [106, 107] may be used to automatically infer the number of mixture components using a Dirichlet Process prior over the parameters for infinite GMMs [108]. These methods are broadly available, including the `BayesianGaussianMixture` implementation in the `sklearn.mixture` module.

Clustering algorithms make different assumptions about what constitutes a "cluster," and optimize different objective functions to find clusters in data. The k-means, agglomerative, BIRCH and Gaussian Mixture Model clustering algorithms were each used within the *i*MGXS scheme in this thesis.

10.6 Model Selection

The *model selection* stage of the data processing pipeline in Fig. 10-2 chooses the "best" predictive model for MGXS clustering. The four clustering algorithms discussed in Sec. 10.5 require the number of clusters to be specified *a priori* by the user. Model selection techniques are used to compare multiple clustering models (*i.e*, trained for different numbers of clusters) and choose the most appropriate one for a particular application. In the context of the *i*MGXS scheme, the number of MGXS clusters is unknown, nor is it clear that there will always be a well-defined number of clusters in highly heterogeneous core configurations. In this case, model selection aims to determine the *fewest* number of MGXS clusters needed to meet the requirements for predictive accuracy and enable the fastest possible MC convergence rate. More specifically, model selection must identify the optimal number of clusters such that adding another cluster would only marginally improve the predictive accuracy of multi-group transport codes such as OpenMOC.

The model selection stage of the *i*MGXS data processing pipeline is described in Fig. 10-18. Since *i*MGXS is an unsupervised scheme for spatial homogenization, a numerical criterion must be used to quantify the "quality" of each clustering model. The criterion may be subjected to thresholding or other techniques to automate model selection and obviate the need for human intervention. When choosing the appropriate

number of clusters, model selection must strike a balance between:

- **Maximum Compression** – Assign all samples to a single cluster
- **Maximum Accuracy** – Assign each sample to its own cluster

In the context of pin-wise spatial homogenization, the former is null homogenization (Sec. 8.2.2) and the latter is degenerate homogenization (Sec. 8.2.3).

The machine learning community has developed a plethora of model selection techniques for statistical clustering analysis. The techniques are often segmented into the following two categories:

- **External Evaluation** – Validate model data with the *ground truth labels*
- **Internal Evaluation** – Score model on its intra- and inter-cluster *similarities*

External evaluation criteria are useful in supervised learning settings where some or all of the dataset may include the ground truth cluster labels. In this case, some of the dataset may be left out of the training phase and used to validate the model trained on the rest of the data. In the context of *i*MGXS, none of the samples are labeled and all of the cluster labels are unknown. As a result, *internal evaluation* criteria must be used to optimize cluster models based upon the unlabeled dataset with which each was trained. Internal evaluation criteria generally attempt to identify cluster models with high *intra-cluster* similarities and low *inter-cluster* similarities[81]. Alternatively, some internal evaluation criteria act as *regularizers* to balance model complexity with generality.

The following sections describe several model selection criteria that are use to evaluate the clustering models produce by the *i*MGXS scheme. Secs. 10.6.1 to 10.6.4 introduce the Davies-Bouldin, Dunn and Calinski-Harabaz indices and silhouette coefficients. These criteria use different metrics to quantify intra- and inter-cluster similarities to quantify the quality of a clustering model. Sec. 10.6.5 presents the Bayesian Information Criterion, a technique with a strong theoretical formulation that may be used to evaluate GMMs. Finally, Sec. 10.6.6 discusses cross-validation and how it may be used in conjunction with any of these model selection criteria to prevent over-fitting.

[81] Conversely, some criteria minimize and maximize the intra and inter-cluster *dissimiliarities*, respectively.

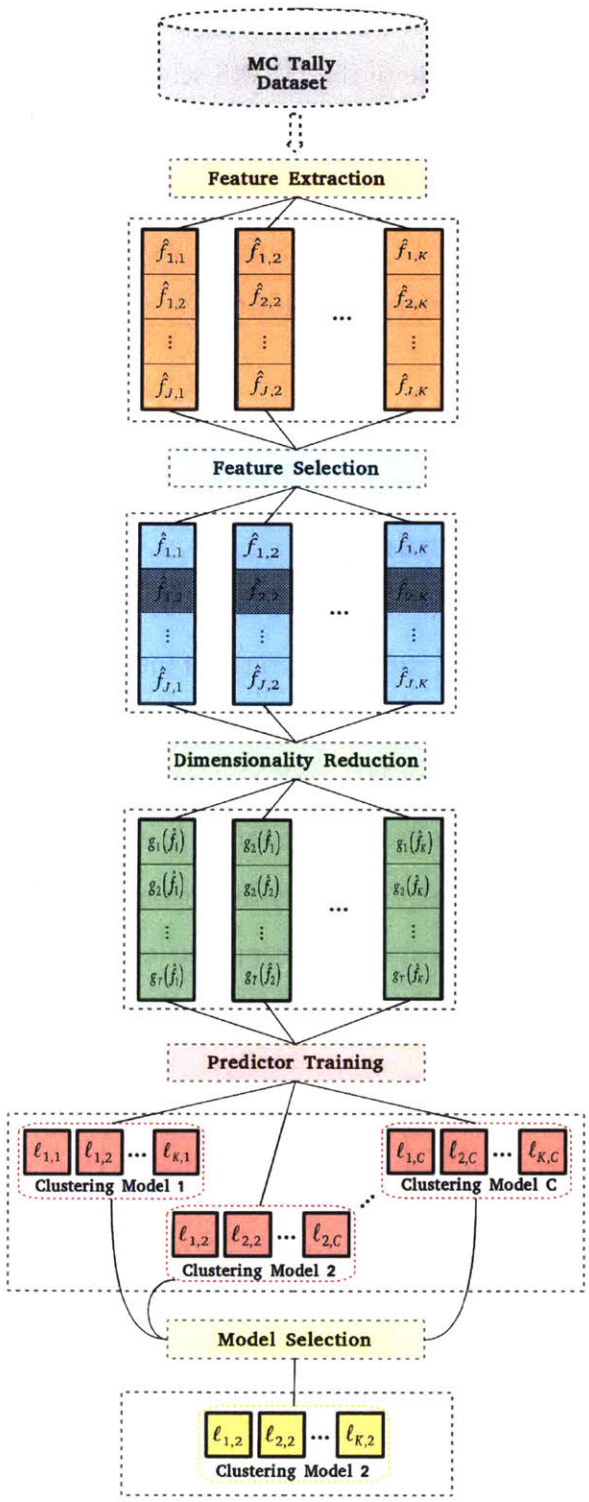


Figure 10-18: Model selection evaluates the clustering models.

It is important to note that model selection may *only* examine the clustering models. The clustered MGXS may *not* be employed in a multi-group transport simulation and

then compared with respect to some reference MC solution. Although such an exercise is important as a proof of principle of the *i*MGXS scheme, it is unacceptable as a model selection technique since it is supervised. Furthermore, it would nullify the over-arching goal of this thesis – to obtain MC quality solutions with computationally efficient deterministic multi-group transport codes – since it would require the computationally expensive calculation of a MC reference solution.

10.6.1 Davies-Bouldin Index

The Davies-Bouldin (DB) index [109] is the ratio of intra-cluster to inter-cluster dissimilarity, or equivalently, the ratio of within-cluster scatter to between-cluster separation. The DB index penalizes cluster models with high *intra-cluster distances* (*i.e.*, low intra-cluster similarities) and low *inter-cluster distances* (*i.e.*, high inter-cluster similarities). The inter-custer distance $d(\boldsymbol{\mu}_m, \boldsymbol{\mu}_n)$ is simply the distance between cluster centroids $\boldsymbol{\mu}_m$ and $\boldsymbol{\mu}_n$ in feature space. The most common metric for the intra-cluster distance R_m is the average distance of all samples in cluster $\mathbb{S}_m$ to its centroid $\boldsymbol{\mu}_m$:

$$R_m = \left(\frac{1}{|\mathbb{S}_m|} \sum_{\hat{f}_k \in \mathbb{S}_m} \left(\hat{f}_k - \boldsymbol{\mu}_m \right)^2 \right)^{\frac{1}{2}} \tag{10.32}$$

This formulation of the intra-cluster distance is equivalent to the subcluster radius used by the BIRCH clustering algorithm (Eqn. 10.20). The DB index is then the similarity between each cluster $\mathbb{S}_m$ and its most similar one $\mathbb{S}_n$, averaged over all of the clusters:

$$\text{DB} = \frac{1}{K} \sum_{m=1}^{M} \max_{m \neq n} \left(\frac{\sigma_m + \sigma_n}{d(\boldsymbol{\mu}_m, \boldsymbol{\mu}_n)} \right) \tag{10.33}$$

The "best" clustering model is the one which *minimizes* the DB index. The DB index is used to evaluate the clustering models produced by *i*MGXS in the following chapter.

10.6.2 Dunn Index

The Dunn index [110] is the ratio of the minimum *inter-cluster distance* to the maximum *intra-cluster distance*, or equivalently, the ratio between the distance separating the two most similar clusters and the radius of the most dissimilar cluster. The Dunn index seeks to identify clustering models with high-density (*i.e.*, high intra-cluster similarities) and well-separated clusters (*i.e.*, low inter-cluster similarities). In contrast to the DB index which is an average across clusters, the Dunn index is indicative of the most poorly defined cluster(s) in the model. Various formulations may be used for both the intra- and inter-cluster distances. This thesis used the same formulations for the intra-cluster distance R_m (Eqn. 10.20) and inter-cluster distance $d(\boldsymbol{\mu}_m, \boldsymbol{\mu}_n)$ (Eqn. 10.33) as those used for the DB index. The Dunn index is then given by the following expression:

$$\mathrm{D} = \frac{\min\limits_{1 \le m \le n \le M} d(\boldsymbol{\mu}_m, \boldsymbol{\mu}_n)}{\max\limits_{1 \le p \le M} R_p} \tag{10.34}$$

The "best" clustering model is the one which *maximizes* the Dunn index. The Dunn index is used to evaluate the clustering models produced by *i*MGXS in the following chapter.

10.6.3 Calinski-Harabaz Index

The Calinski-Harabaz (CH) index [111] is the ratio of between-cluster and within-cluster *dispersion*, and is often referred to as the *variance ratio criterion*. The CH index is formally defined as follows:

$$CH = \frac{\mathrm{Tr}(\boldsymbol{B})}{\mathrm{Tr}(\boldsymbol{W})} \times \frac{K - M}{M - 1} \tag{10.35}$$

where $\boldsymbol{B}$ and $\boldsymbol{W}$ are the between-group and within-group dispersion matrices given by:

$$\boldsymbol{B} = \sum_{m=1}^{M} |\mathbb{S}_m| \times \left(\hat{\boldsymbol{\mu}}_m - \overline{\hat{\boldsymbol{f}}}_k\right)\left(\hat{\boldsymbol{\mu}}_m - \overline{\hat{\boldsymbol{f}}}_k\right)^T \tag{10.36}$$

$$W = \sum_{m=1}^{M} \sum_{\hat{f}_k \in \mathbb{S}_m} (\hat{f}_k - \hat{\mu}_m)(\hat{f}_k - \hat{\mu}_m)^T \tag{10.37}$$

The B matrix depends on the mean $\overline{\hat{f}}_k$ of all samples in the dataset.

The higher the between-group variance and the lower the within-group variance, the more well-defined the clusters are for a given model. Hence, the "best" clustering model is the one which *maximizes* the Calinski-Harabaz index. The Calinski-Harabaz index is used to evaluate the clustering models produced by *i*MGXS in the following chapter.

10.6.4 Silhouette Coefficient

The silhouette coefficient [112] is defined for each sample, and compares the average distance between a sample and all other samples within its assigned cluster to the average distance to samples in all other clusters. Since the silhouette coefficient is defined for each sample, it provides a more granular metric to evaluate a clustering model on a sample-by-sample basis than the Davies-Bouldin, Dunn and Calinski-Harabaz indices. In particular, a high silhouette coefficient signifies a "well-clustered" sample, while a low silhouette coefficient may indicate that a sample is an outlier. The distribution of silhouette coefficients is often visualized with a *silhouette plot* to aid in identifying the quality of each cluster in a model. If many samples have a low silhouette coefficient, then it may indicate the inclusion of too few or too many clusters in the model.

The silhouette coefficient defines the dissimilarity a_k of sample $\hat{f}_k$ to its cluster $\mathbb{S}_m$ as the average distance from $\hat{f}_k$ to all points in $\mathbb{S}_m$:

$$a_k = \frac{1}{|\mathbb{S}_m|-1} \sum_{\substack{\hat{f}_j \in \mathbb{S}_m \\ j \neq k}} \|\hat{f}_j - \hat{f}_k\| \tag{10.38}$$

Similarly, the smallest dissimilarity b_k of sample $\hat{f}_k$ to all other clusters compares the sample to all samples in the "next best fit" or nearest neighboring cluster $\mathbb{S}_n$:

$$b_k = \frac{1}{|\mathbb{S}_n|} \sum_{\hat{f}_j \in \mathbb{S}_n} \|\hat{f}_j - \hat{f}_k\| \tag{10.39}$$

The silhouette coefficient for sample $\hat{f}_k$ is then given by the following:

$$s_k = \frac{b_k - a_k}{\max(a_k, b_k)} \tag{10.40}$$

The average silhouette coefficient over all samples $\bar{s}_k$ may be used to measure the overall quality of a clustering model:

$$\bar{s}_k = \frac{1}{K} \sum_{k=1}^{K} s_k \tag{10.41}$$

The silhouette coefficient is defined over the range $-1 \leq s_k \leq +1$ where a value near $+1$ indicates that the sample is well-matched to its assigned cluster. Likewise, a value near -1 indicates that sample $\hat{f}_k$ is ill-matched to its assigned cluster and may be more appropriately matched with the nearest neighboring cluster. Furthermore, a value near zero indicates that the sample is on the border of two equally valid (but poorly matched) clusters. Therefore, in general, higher average silhouette coefficient $\bar{s}_k$ indicate a better matching of samples to assigned clusters. Hence, the "best" clustering model is the one which *maximizes* the average silhouette coefficient. The silhouette coefficient is used to evaluate the clustering models produced by *i*MGXS in the following chapter.

10.6.5 Bayesian Information Criterion

The Bayesian Information Criterion (BIC) [113] may be used to optimize parameter selection for models with well-defined likelihood functions. As a result, of the four clustering algorithms presented in Sec. 10.5, the BIC may only be used to select the number of components for GMMs. In contrast to the previously defined model selection criterion, the BIC is based on an information theory analysis of model complexity and attempts to measure the "efficiency" with which a parametrized model predicts an observed dataset. The BIC is an asymptotic result that is only guaranteed to recover the

"true" number of mixture components in a GMM given an infinite number of samples which were drawn i.i.d. from a mixture of Gaussians. In practice, the asymptotic result is only valid if the sample size is much larger than the number of mixture components.

These assumptions are important to consider in the context of the *i*MGXS data processing pipeline. Pin-wise MGXS datasets are not necessarily distributed according to a GMM. The case studies in Chap. 9 indicate that the number of samples per component should increase with the size of the core geometry[82]. Hence, one might expect that the BIC would perform better for a full core geometry than for a single assembly.

The BIC is based upon the value of the *maximum likelihood function* $\hat{L}$[83]:

$$\hat{L} = \max_{\boldsymbol{\theta}} p\left(\{\hat{\boldsymbol{f}}_1, \hat{\boldsymbol{f}}_2, \ldots, \hat{\boldsymbol{f}}_K\} | \boldsymbol{\theta}\right) \tag{10.42}$$

where the model parameters are denoted by $\boldsymbol{\theta}$ (*i.e.*, the GMM mixture means $\boldsymbol{\mu}_m$ and covariance matrices $\boldsymbol{\Sigma}_m$). The parameters found by the expectation-maximization algorithm maximize the *log-likelihood* $\ln\left(\hat{L}\right)$ for the GMM defined by Eqns. 10.25 and 10.26:

$$\ln\left(\hat{L}\right) = \sum_{k=1}^{K} \log \sum_{m=1}^{M} \mathbb{1}_{\mathbb{S}_m(k)} \left\{ -\frac{1}{2}\ln(|\boldsymbol{\Sigma}_m|) - \frac{1}{2}\left(\hat{\boldsymbol{f}}_k - \boldsymbol{\mu}_m\right)^T \boldsymbol{\Sigma}_m^{-1} \left(\hat{\boldsymbol{f}}_k - \boldsymbol{\mu}_m\right) - \frac{J}{2}\ln(2\pi) \right\} \tag{10.43}$$

where the indicator function $\mathbb{1}_{\mathbb{S}_m}(k)$ is equal to 1 if $k \in \mathbb{S}_m$ and 0 otherwise. The BIC is then given by the following expression:

$$\text{BIC} = -2 \cdot \ln\left(\hat{L}\right) + |\boldsymbol{\theta}| \cdot \ln(K) \tag{10.44}$$

The first term in the BIC uses the maximum log-likelihood to measure the ability of the GMM to predict the observed MGXS dataset. The second term acts as a regularizer and penalizes the complexity of the model by comparing the number of model parameters $|\boldsymbol{\theta}|$ to the number of observed samples K. The "best" clustering model is the one which *minimizes* the Bayesian Information Criterion. The BIC is used to evaluate the GMMs produced by *i*MGXS in the following chapter.

[82] This assumes that the complexity of heterogeneities – each of which induce MGXS (sub)clusters – does not also scale with the size of the core geometry.

[83] The probability of observing a dataset given the model parameters $\boldsymbol{\theta}$.

10.6.6 Cross-Validation

Cross-validation is a technique for model validation [114], and is closely related to the *bootstrap* method commonly used to approximate the properties of statistical estimators [115]. Cross-validation is used to assess the generalization of a model by evaluating it with a dataset independent of the one used to train the model. In general, cross-validation is used to control model complexity and prevent over-fitting to outliers. The integration of the predictor training and model selection stages with cross-validation is highlighted in Alg. 10-5. Cross-validation aims to estimate a predictor's performance by averaging a model selection criterion across multiple models trained on disjoint subsets of a dataset. A model selection criterion averaged across multiple models will be more robust than a single model trained on the complete dataset. Although cross-validation was not evaluated as part of the *i*MGXS data processing pipeline for this thesis, it may be worthwhile to consider its value for model selection in the future.

Algorithm 10-5: Cross-Validation for Clustering Model Selection

```
1: Divide dataset into P mutually exclusive partitions
2: for each set of model parameters do
3:     for each partition p do
4:         Train model on data from the other P − 1 partitions
5:         Score model with criterion evaluated for partition p
6:     end for
7:     Average selection criterion across models
8: end for
9: Choose model with best model selection criteria
```

Model selection techniques are used to choose clustering model parameters, including the number of clusters. Various methods score clustering models based upon their intra- and inter-cluster similarities, or penalize model complexity using regularization. The Davies-Bouldin, Dunn and Calinski-Harabaz indices and silhouette coefficients are implemented for each clustering algorithm employed by the *i*MGXS scheme, and the Bayesian Information Criterion is implemented for Gaussian Mixture Models.

10.7 Spatial Homogenization

The final *spatial homogenization* stage in the data processing pipeline in Fig. 10-2 applies the selected clustering model to spatially homogenize pin-wise MGXS as illustrated in Fig. 10-19. Like the degenerate and LNS spatial homogenization schemes (Secs. 8.2.3 and 9.2.1), a single MC calculation of the complete heterogeneous geometry is used to generate MGXS for all materials. The MGXS are tallied separately for each instance of fissile material zones using OpenMC's distributed cell tallies (Sec. 4.2.2). The clustering model found from the preceding stages of the *i*MGXS data processing pipeline assigns an integral cluster label ℓ_k to each unique pin instance k in the combinatorial geometry (CG). *Track density-weighting* (Sec. 9.2.2) is used to average MGXS across all pin instances with the same cluster label in an identical fashion to LNS spatial homogenization. The OpenCG region differentiation algorithm (Sec. 4.4.3) is used to build an OpenMOC geometry with unique cells and materials for each fuel pin which mirrors the clustering model's representation of the CG model. Like the infinite, null, degenerate and LNS schemes, spatial self-shielding effects experienced by different non-fissile spatial zones are averaged across the entire geometry for each non-fissile material.

It is important to recall that *i*MGXS spatial homogenization aims to homogenize across as many fuel pin instances as possible while sacrificing little or no accuracy in downstream multi-group calculations. The observations made in Sec. 9.4 about the convergence rate for MGXS generated with MC are equally valid for *i*MGXS as they were

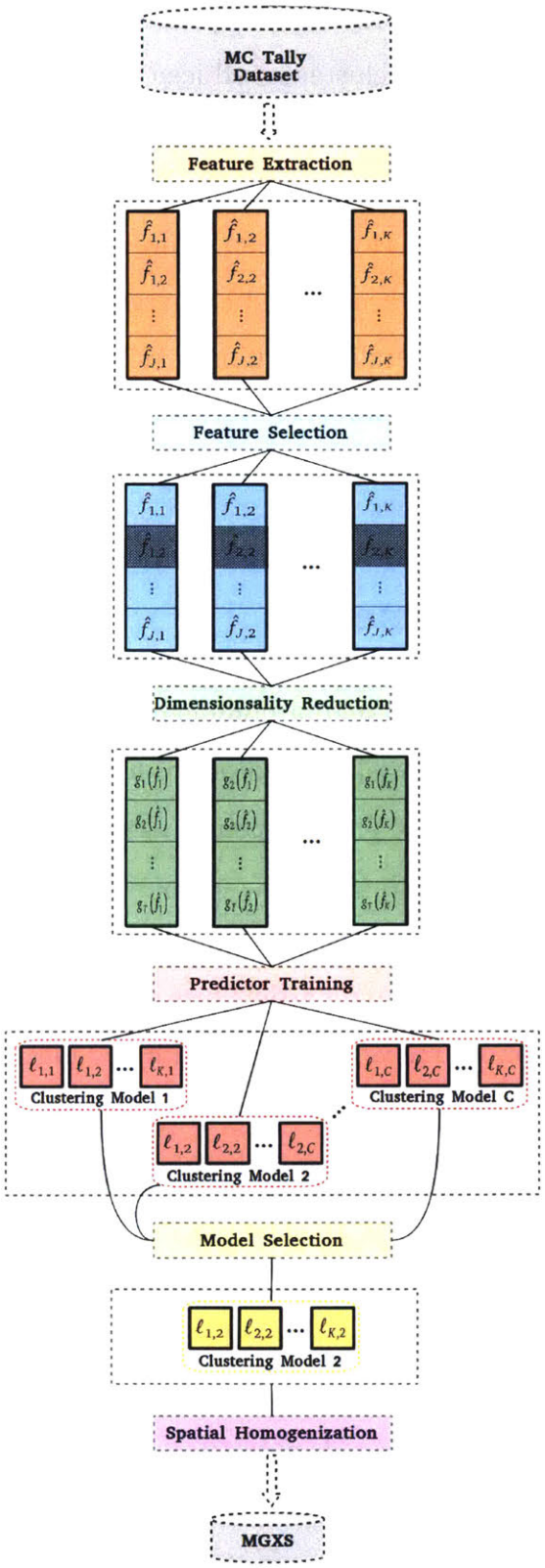


Figure 10-19: The detailed *i*MGXS data processing pipeline.

for the null, degenerate and LNS schemes. In particular, the standard deviation of the MGXS homogenized across each cluster is reduced by a factor inversely proportional to the number of pins in the clusters according to Eqn. 9.5. Therefore, the more fuel pin instances assigned to each cluster, the faster the MGXS will converge to within an acceptable level of uncertainty. However, an uneven assignment of pin instances to clusters will impact the relative accelerated convergence rates achieved for different clusters. In other words, the overall reduction in MGXS uncertainties – or conversely, the acceleration of the MGXS convergence rates – will be bounded by the cluster with the fewest fuel pin instances.

10.8 Clustered Geometries

This section employs the *i*MGXS spatial homogenization scheme and presents visualizations of the predicted material configurations for each of the six heterogeneous benchmarks introduced in Chap. 7. The `openmc.mgxs` module (Sec. 4.2.4) was used to compute 70-group MGXS with OpenMC, along with all other tallies necessary to compute each of the features described in Sec. 10.2. The OpenMC simulations were performed with 1000 batches of 10^6 particle histories per batch for each benchmark. Stationarity of the fission source was obtained with 200 inactive batches for the quarter core BEAVRS model, while 100 inactive batches were employed for the other five benchmarks (Sec. 7.3.1). OpenMC's "iso-in-lab" feature (Sec. 4.2.3) was employed to enable consistent comparisons between OpenMC's reference results and OpenMOC's calculations with an isotropic in lab scattering source in the following chapter.

The 70-group MC tally data was condensed to 2-groups for feature extraction within the *i*MGXS data processing pipeline. Two different approaches were separately evaluated for feature selection. Pinch feature selection (Sec. 10.3.5) was used to select features derived from U-238 capture MGXS in group 1 of 2 (encompassing the resonance region), as well as the fractional reactivity and spectral index features (Secs. 10.2.2 and 10.2.3). A second feature selection approach, referred to here as *litmus-only* feature selection, only used the reaction fraction thresholding litmus test (Sec. 10.3.1.2) to select the "best"

reaction type (*i.e*, total, capture or fission) for each pair of nuclides and energy groups, for each nuclide in the fuel (*i.e.*, U-234, U-235, U-238, O-16 and O-17) and both energy groups. All available features were used for each selected reaction type, nuclide and energy group. No dimensionality reduction techniques (Sec. 10.4) were applied to the selected features. The agglomerative clustering algorithm (Sec. 10.5.2) was used for each of the case studies presented here[84].

The geometries produced with 2, 4 and 8 clusters for the 1.6% enriched fuel assembly and 3.1% enriched fuel assembly with 20 BPs are presented in Figs. 10-20 and 10-21, respectively. Likewise, the geometries produced with 2, 4 and 8 clusters for the 2×2 colorset without and with a reflector are displayed in Figs. 10-22 and 10-23. Finally, the geometries produced with 2 – 64 clusters, spaced in powers of 2, for the quarter core BEAVRS model are depicted in Figs. 10-24 to 10-29. Each uniquely colored material represents a unique set of MGXS. In general, it is apparent that the *i*MGXS scheme is able to identify clusters of fuel pins which are related by the presence of similar neighboring heterogeneities which induce the clustering of samples in feature space[85]. As a result, *i*MGXS spatial homogenization would be expected to approach the accuracy of the degenerate scheme and the convergence rate of the null scheme.

A couple of key observations can be made from Figs. 10-20 to 10-23. First, the material configurations are highly dependent on the types of features used to train the clustering model, as evidenced by the very different clustering assignments found with pinch and litmus-only feature selection. The inclusion of features for many different nuclides and reaction types with litmus-only selection clearly impacts not only the separation distance, but more importantly, the relationships between samples in feature space. Pinch feature selection only considers the structure in features derived from U-238 capture MGXS, along with the spectral indices and fractional reactivities, while litmus-only feature selection also considers clustering in U-235 fission and other

[84]The four clustering algorithms highlighted in Sec. 10.5 were observed to produce different clustered geometries for each of the benchmarks. For the sake of brevity, only those geometries produced by agglomerative clustering are illustrated here.

[85]The clustering of samples in feature space does not necessarily reflect the clustering of MGXS. The two are only equivalent if the features cluster in the same way as the MGXS, and the clustering algorithm optimizes over the correct objective function governing the structure of the clusters.

tallied reactions. As was observed from the histograms and quantile-quantile plots in Secs. 9.1.3.2 and 9.1.4.2, the thermal U-235 MGXS are more sensitive to spatial self-shielding effects than U-238 resonance capture, even though clustering effects are more relevant to the prediction of U-238 capture rates. Although pinch and litmus-only feature selection lead to divergent clustering models, the two are not compared in this thesis with respect to their impact on the predictions of spatially distributed reaction rates for practical reasons.

In addition, it is noteworthy that the *i*MGXS scheme discriminates pins along assembly-assembly and reflector-assembly interfaces for the 2×2 colorset benchmarks and the quarter core BEAVRS model. This contrasts with the LNS scheme which fails to distinguish pins along the various borders of fuel assemblies. The ability of *i*MGXS to separate out fuel pins along interfaces without engineering prescriptions illustrates the key advantage of *i*MGXS over the LNS scheme. As a result, the *i*MGXS scheme is expected to outperform LNS spatial homogenization for arbitrary core geometries.

Interestingly, the models of 2 – 4 clusters for the quarter core BEAVRS model cluster the pins in the interior fuel assemblies in a way which resembles the clustered geometries for the individual fuel assemblies in Figs. 10-20 and 10-21. This indicates that the flux profile in the interior assemblies is similar to that in an infinite lattice. As more clusters are introduced, the pins along the assembly-assembly interfaces are resolved into their own clusters. However, it should be noted that more "outlier" pins (pins assigned to the "wrong" cluster) are observed as more clusters are introduced, although the predictions with litmus-only feature selection are more "robust" for more clusters. Ideally, the model selection criteria discussed in Sec. 10.6 may be used to determine the minimum number of clusters necessary without sacrificing accuracy due to the presence of outliers.

The interested reader is referred to App. D.1 for a presentation of the clustered geometries produced when non-MGXS features are removed during feature selection (*i.e.*, the clustering algorithm only considers the pin-wise MGXS means and uncertainties). This highlights the importance of using features such as the fractional reactivities, spectral indices and reaction fractions to best identify structures in MC tally data due to spatial self-shielding effects. Furthermore, App. D.2 illustrates the clustered geometries produced

when dimensionality reduction techniques such as PCA, ICA and FA are employed.

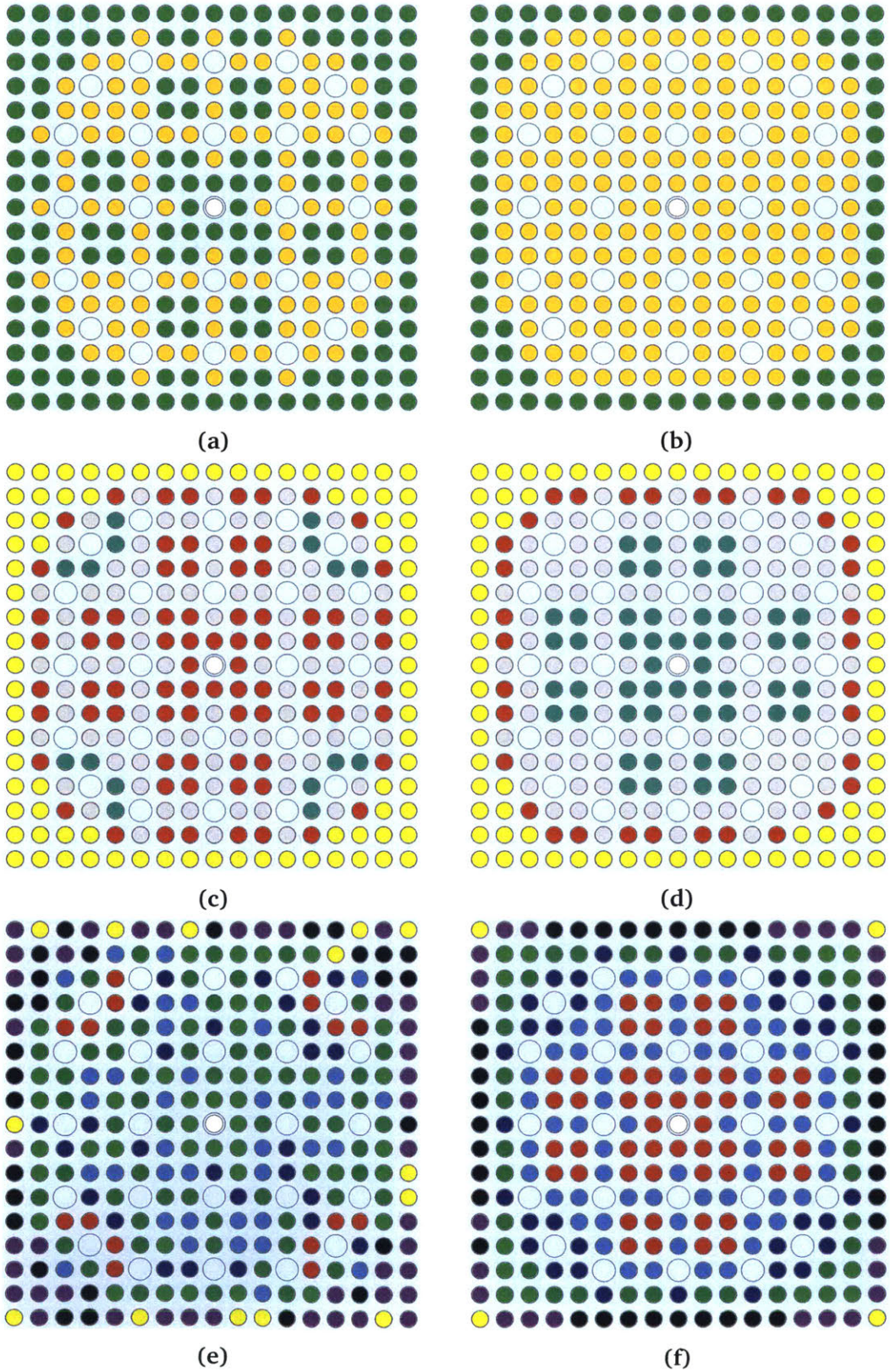


Figure 10-20: Materials for the 1.6% enriched fuel assembly with *i*MGXS spatial homogenization. The materials for 2, 4, and 8 clusters are illustrated in (a), (c) and (e) for pinch and in (b), (d) and (f) for litmus-only feature selection, respectively.

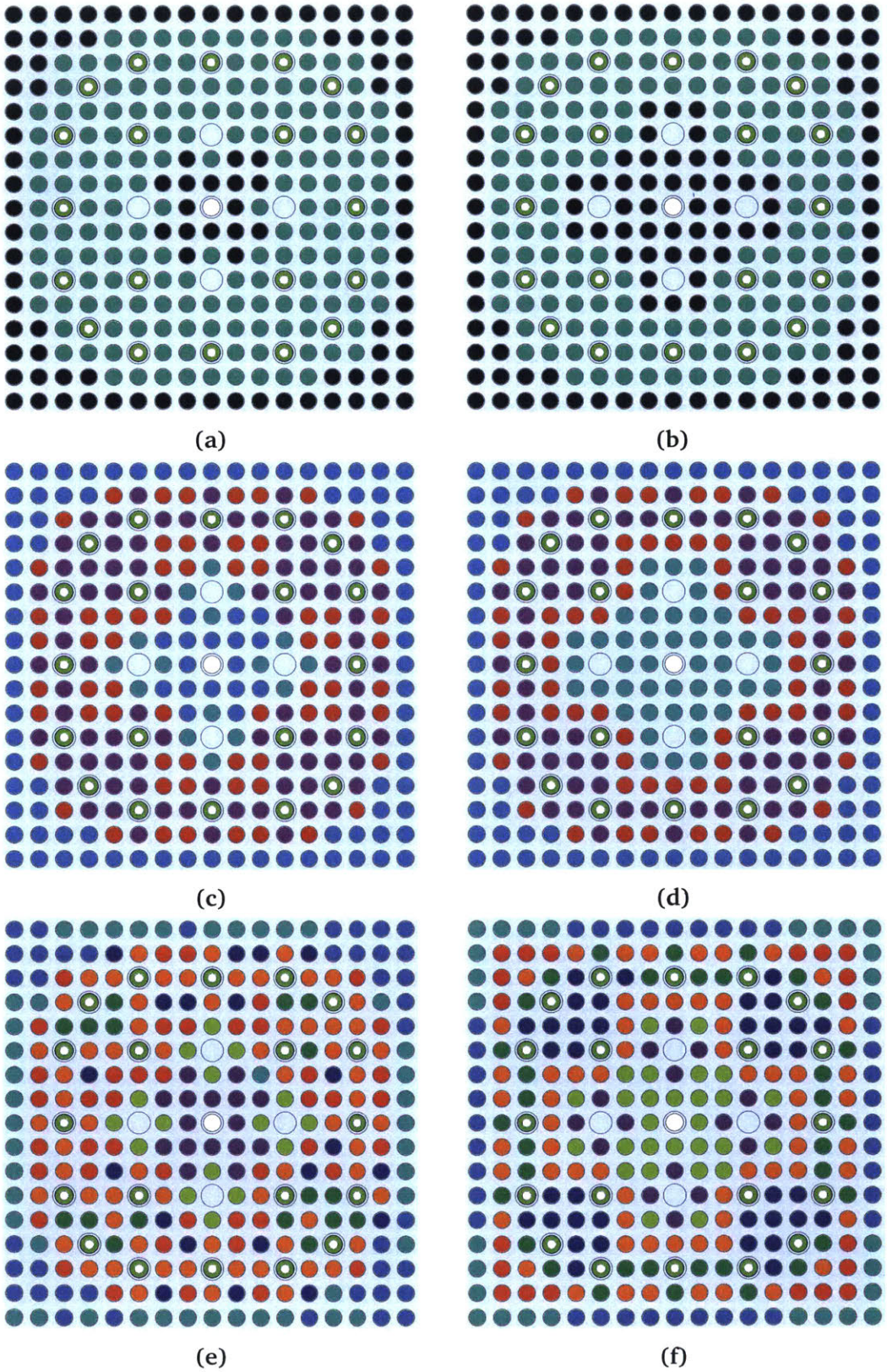


Figure 10-21: Materials for the 3.1% enriched fuel assembly with 20 BPs with *i*MGXS spatial homogenization. The materials for 2, 4, and 8 clusters are illustrated in (a), (c) and (e) for pinch and in (b), (d) and (f) for litmus-only feature selection, respectively.

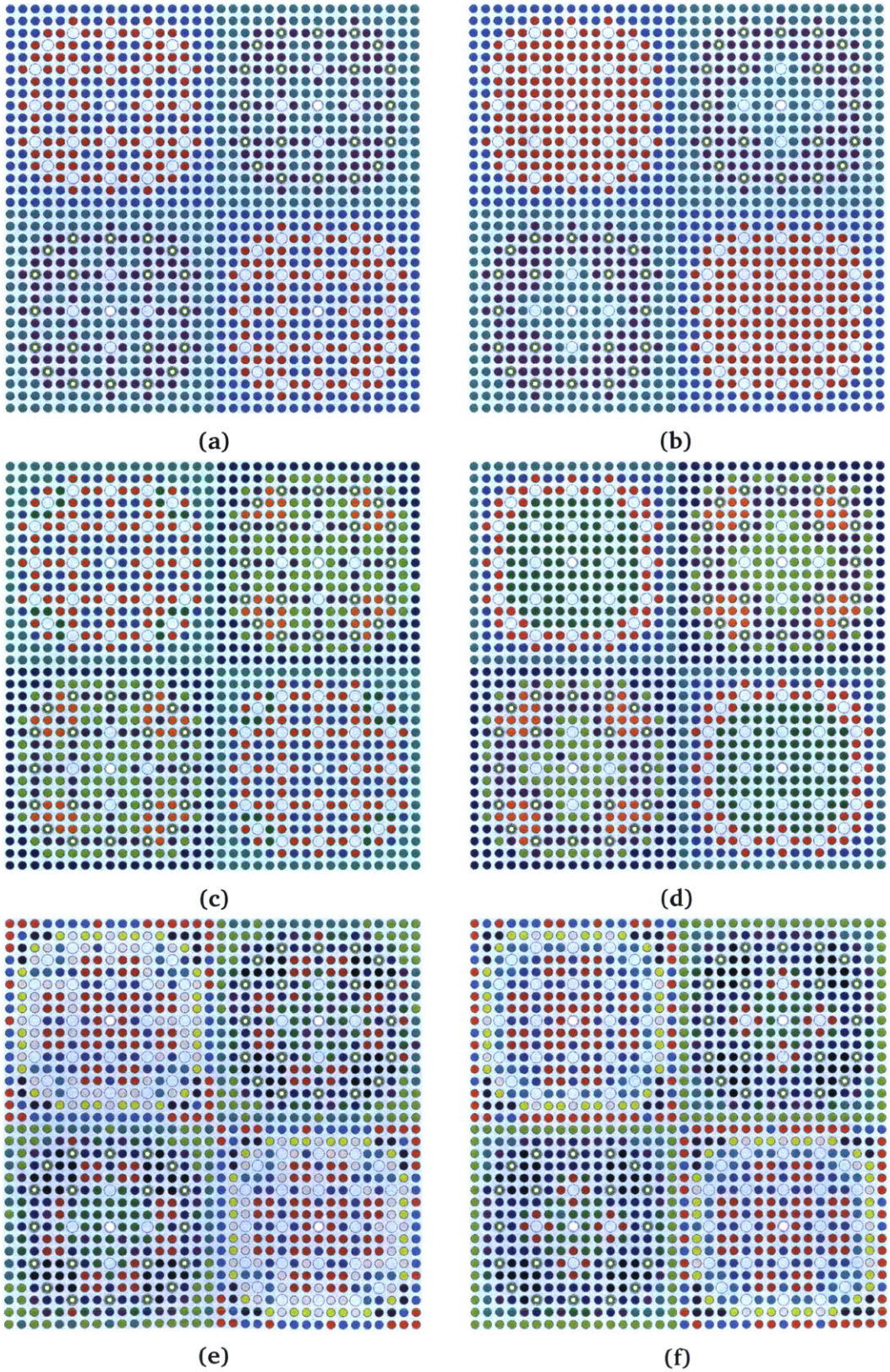


Figure 10-22: Materials for the 2×2 periodic colorset with *i*MGXS spatial homogenization. The materials for 2, 4, and 8 clusters are illustrated in (a), (c) and (e) for pinch and in (b), (d) and (f) for litmus-only feature selection, respectively.

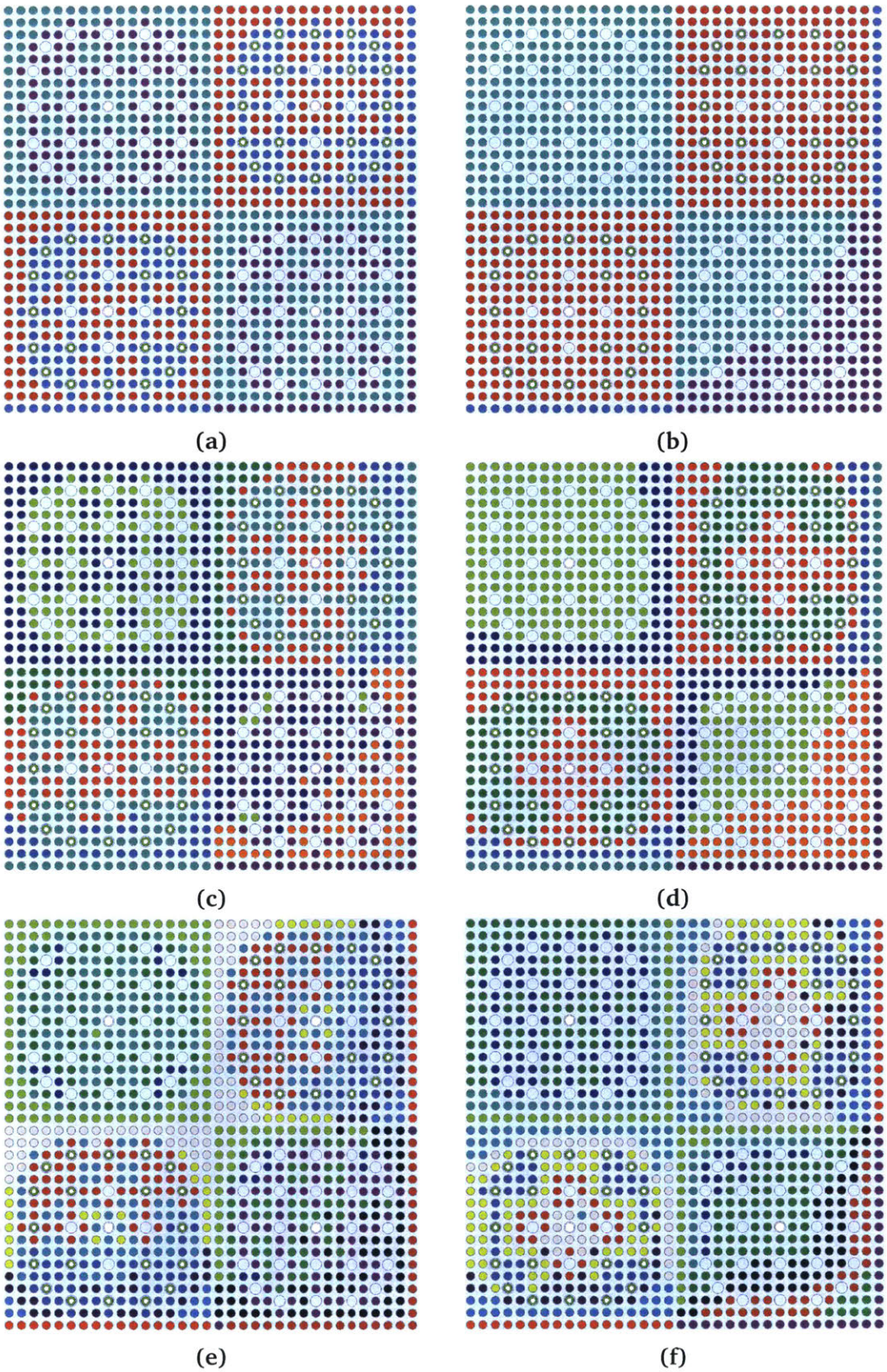
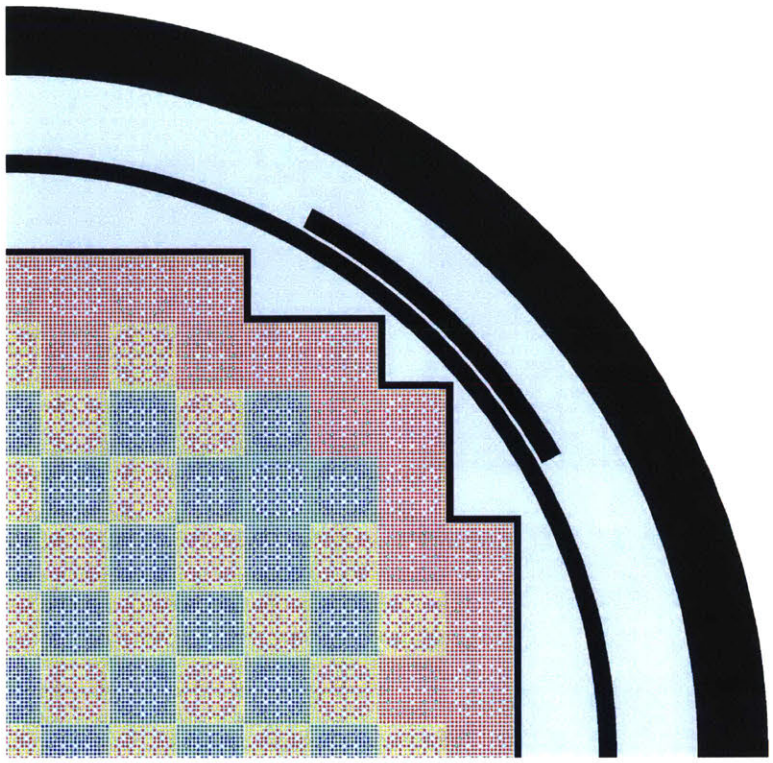
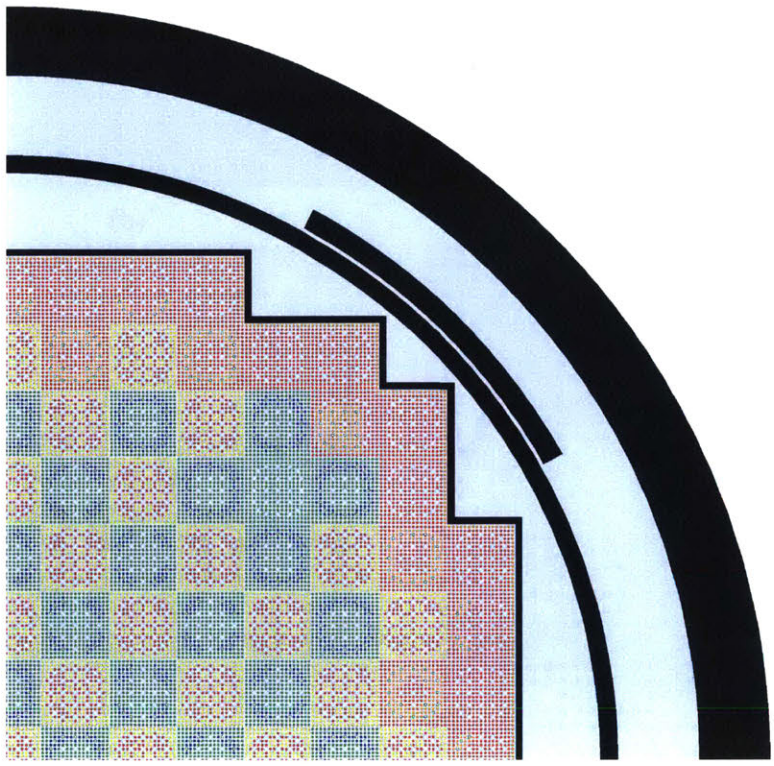


Figure 10-23: Materials for the 2×2 colorset with water reflector with *i*MGXS spatial homogenization. The materials for 2, 4, and 8 clusters are illustrated in (a), (c) and (e) for pinch and in (b), (d) and (f) for litmus-only feature selection, respectively.



(a)



(b)

Figure 10-24: Materials for the BEAVRS model with *i*MGXS spatial homogenization of 2 clusters with pinch feature selection (a) and litmus-only feature selection (b).

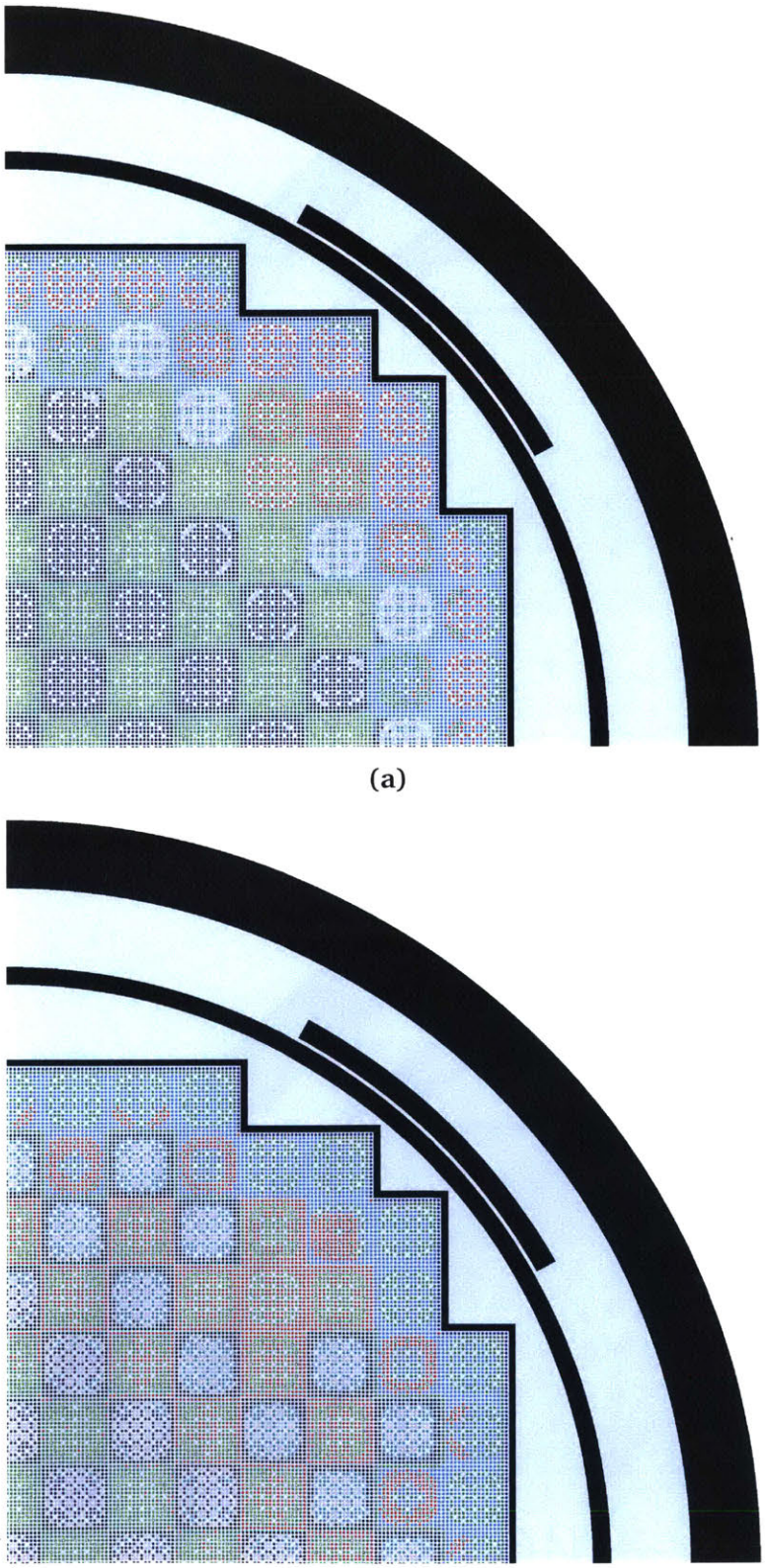
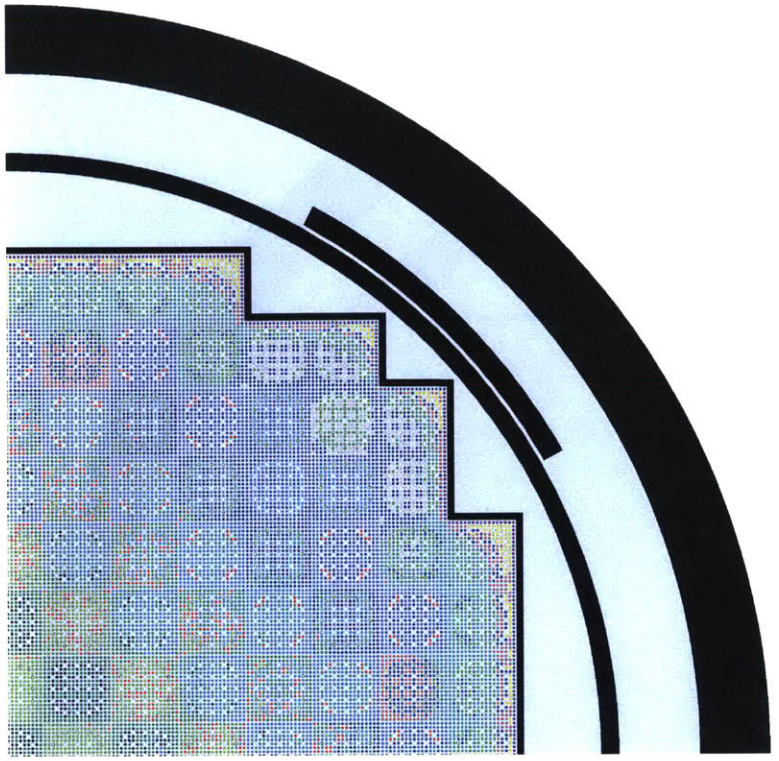
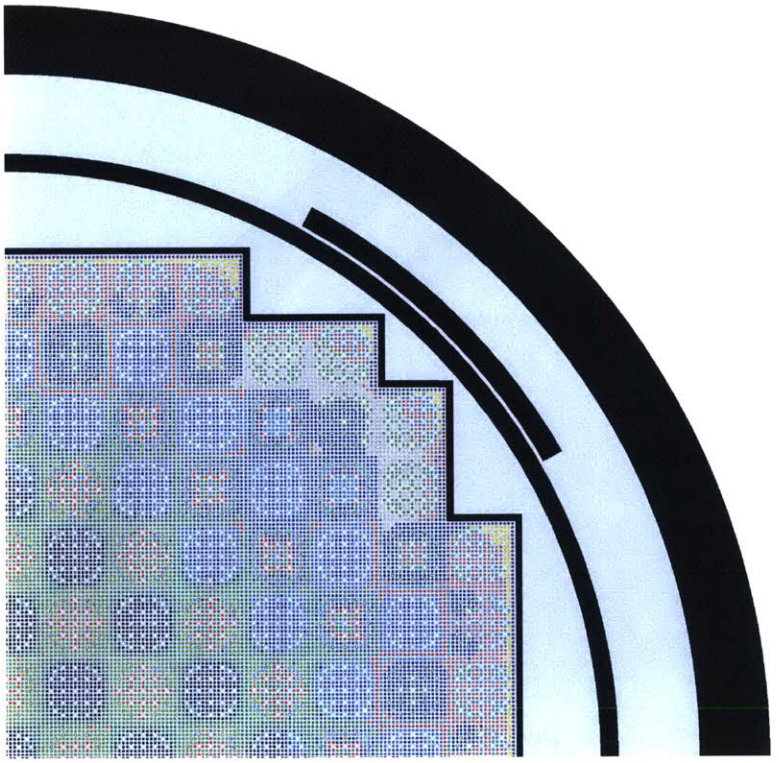


Figure 10-25: Materials for the BEAVRS model with *i*MGXS spatial homogenization of 4 clusters with pinch feature selection (a) and litmus-only feature selection (b).



(a)



(b)

Figure 10-26: Materials for the BEAVRS model with *i*MGXS spatial homogenization of 8 clusters with pinch feature selection (a) and litmus-only feature selection (b).

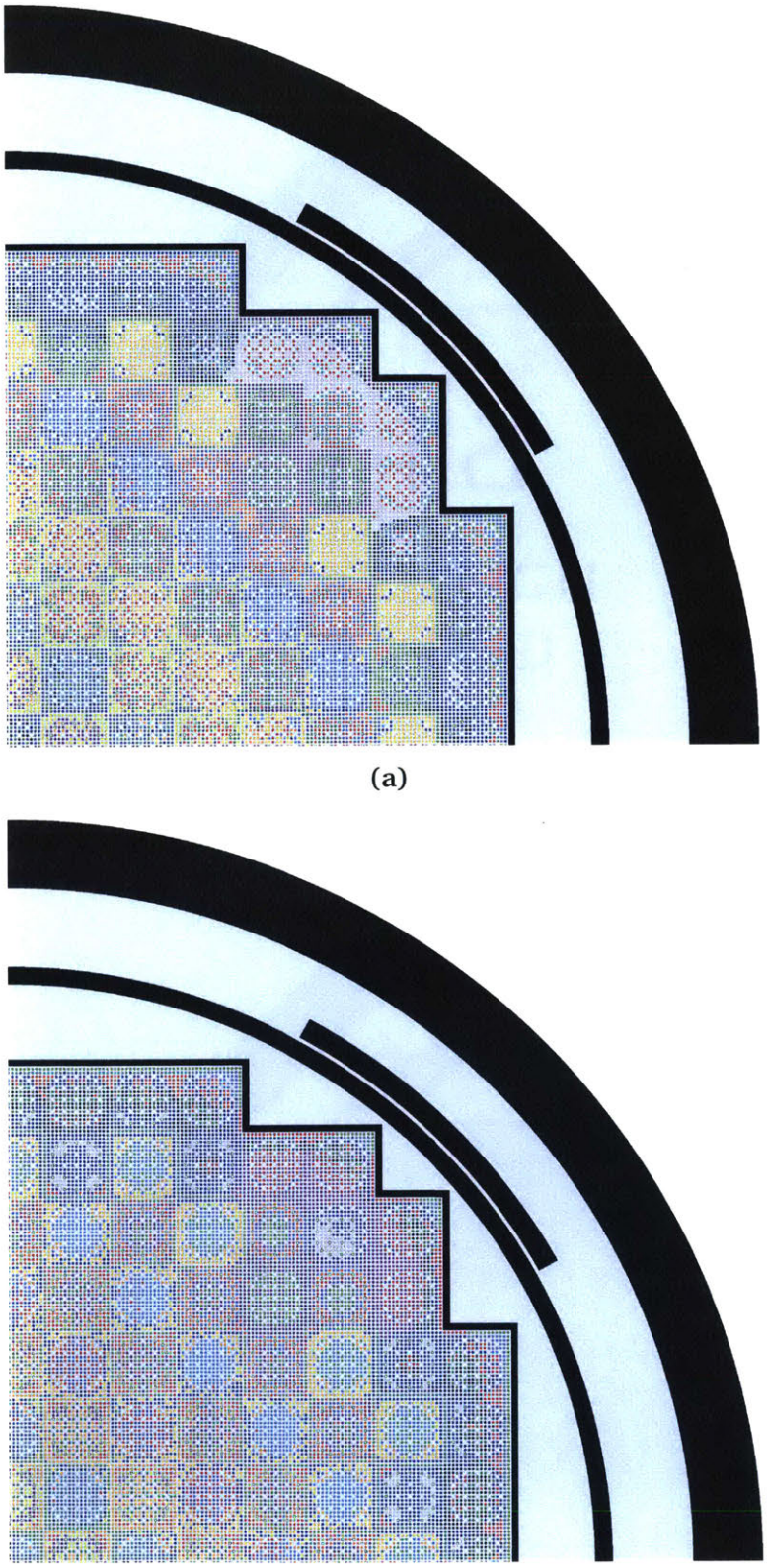
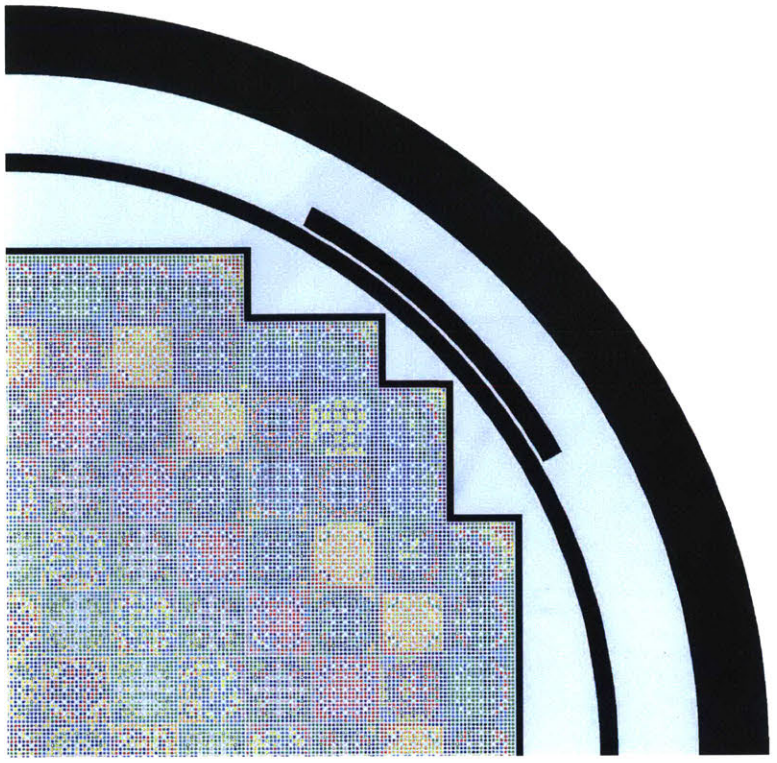
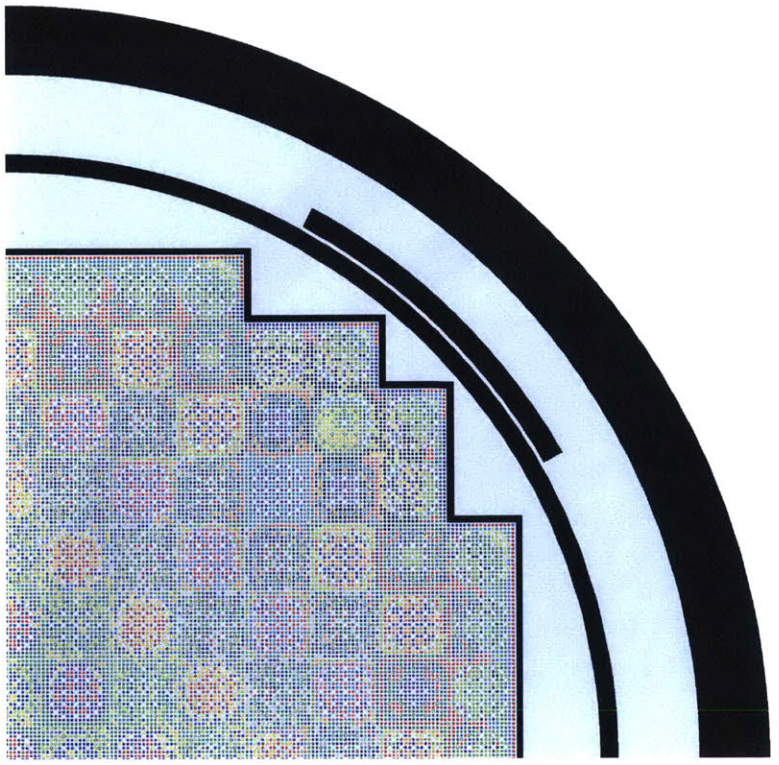


Figure 10-27: Materials for the BEAVRS model with *i*MGXS spatial homogenization of 16 clusters with pinch feature selection (a) and litmus-only feature selection (b).



(a)



(b)

Figure 10-28: Materials for the BEAVRS model with *i*MGXS spatial homogenization of 32 clusters with pinch feature selection (a) and litmus-only feature selection (b).

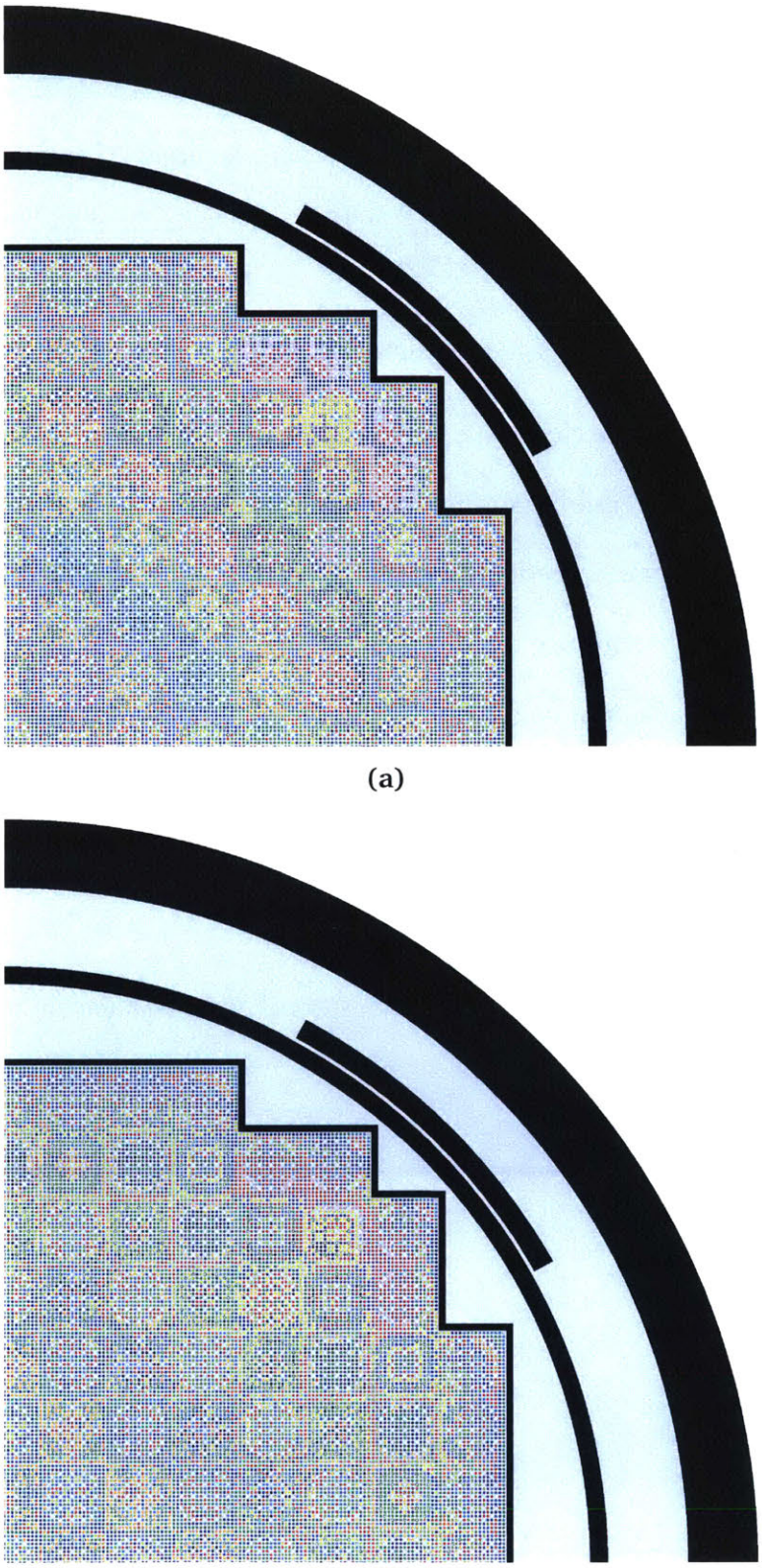


Figure 10-29: Materials for the BEAVRS model with *i*MGXS spatial homogenization of 64 clusters with pinch feature selection (a) and litmus-only feature selection (b).

Highlights

- The *inferential MGXS* (*i*MGXS) spatial homogenization scheme uses unsupervised clustering analysis of MC tally data to predict which fuel pins experience similar spatial self-shielding effects.
- The *i*MGXS data processing pipeline is composed of the following stages:
  - ***Feature extraction*** builds *features* from MC tallies for each fuel pin.
  - ***Feature selection*** identifies the features indicative of MGXS clustering.
  - ***Dimensionality reduction*** projects features into a reduced vector space.
  - ***Predictor training*** applies a clustering algorithm to the dataset.
  - ***Model selection*** chooses the best fit clustering model.
  - ***Spatial homogenization*** averages the MGXS of fuel pins in each cluster.
- The material configurations predicted by the *i*MGXS scheme for each of the six heterogeneous benchmarks are illustrated.
- The *i*MGXS scheme easily identifies clusters of fuel pins adjacent to CRGTs and BPs – and more importantly, along assembly-assembly and reflector-assembly interfaces – which the LNS scheme fails to do for arbitrary geometries.

Chapter 11

Evaluation of the *i*MGXS Scheme

The preceding chapter introduced a methodology for *inferential multi-group cross section* (*i*MGXS) spatial homogenization. The *i*MGXS scheme uses unsupervised clustering algorithms to infer the optimal assignment of fuel pin instances to spatial homogenization tally zones based on an analysis of "noisy" MC tally data. This chapter evaluates *i*MGXS homogenization with respect to the scheme's two primary objectives:

- Approach the **accuracy** of *degenerate* spatial homogenization (Sec. 8.2.3)
- Approach the MC **convergence** of *null* spatial homogenization (Sec. 8.2.2)

Furthermore, the *i*MGXS scheme seeks to address the shortcomings of the Local Neighbor Symmetry (LNS) spatial homogenization scheme (Sec. 9.2) – namely, its inability to flexibly distinguish MGXS clusters in arbitrary geometries (*e.g.*, pins along assembly-assembly and assembly-reflector interfaces) – and represents a pathway for reactor agnostic MGXS generation.

The results presented in this chapter reflect the key observations made in Chaps. 8 and 9 for the null, degenerate and LNS homogenization schemes. First, it was observed that pin-wise homogenization with track density-weighted MGXS (Sec. 8.4.2) has no impact on the resultant eigenvalue since all schemes preserve global reactivity. In addition, it was shown that pin-wise MGXS clustering must be accounted for to accurately predict pin-wise U-238 capture rates, but it is much less consequential for accurate fission rate predictions. These results inform the results and analysis presented in this chapter.

The accuracy of the *i*MGXS scheme is evaluated with respect to the null and degenerate schemes in Sec. 11.1. The analysis includes a presentation of the eigenvalue bias and the distributions of pin-wise U-238 capture rate errors for four clustering algorithms (Sec. 10.5) with varying numbers of clusters, for each of the six PWR benchmarks (Sec. 7.2). The convergence rate of the OpenMOC solutions with the null, degenerate and *i*MGXS schemes is presented in Sec. 11.2 for "noisy" MC tally data computed with varying numbers of particle histories. Sec. 11.3 examines the empirical results for model selection criteria (Sec. 10.6) which may be use to select the most appropriate number of clusters for the *i*MGXS scheme. Finally, Sec. 11.4 concludes with an analysis of the computational resource requirements necessary to reach a desired level of accuracy for pin-wise U-238 capture rates for a reference MC calculation in comparison to deterministic multi-group calculations with the degenerate and *i*MGXS schemes.

11.1 Multi-Group Results with *i*MGXS

The `openmc.mgxs` module (Sec. 4.2.4) was used to compute 70-group MGXS with OpenMC, along with all other tallies necessary to compute each of the features described in Sec. 10.2. In contrast to the MGXS generated with OpenMC in Chaps. 8 to 10, the MGXS used in this chapter were generated with 10× more batches and 10× fewer histories per batch. The use of so many active batches made it possible to evaluate the convergence rate of the OpenMOC solutions in Sec. 11.2[86]. In particular, the OpenMC simulations were performed with 10,000 batches of 10^5 particle histories per batch for the assembly and colorset benchmarks, while 10^6 histories per batch were used for the quarter core BEAVRS model. Stationarity of the fission source was obtained with 200 inactive batches for the BEAVRS model, while 100 inactive batches were employed for the other five benchmarks (Sec. 7.3.1). OpenMC's "iso-in-lab" feature (Sec. 4.2.3) was

[86] As a result of the different OpenMC runtime parameters used in this chapter, the OpenMOC eigenvalues and capture rate errors differ slightly from those reported in Chaps. 8 to 10. The preceding chapters use $8 - 9 \times 10^8$ active histories to generate MGXS, while this chapter use $9.8 - 9.9 \times 10^8$ active histories, since fewer histories are expended in the inactive cycles. Although so few histories per batch might risk under-sampling of the fission source distribution for a reference MC calculation, it is less of a concern here since it is assumed that under-sampling is not as problematic for MGXS generation.

employed to enable consistent comparisons between OpenMC's reference results and OpenMOC's calculations with an isotropic in lab scattering source.

The 70-group MC tally data was condensed to 2-groups for feature extraction within the *i*MGXS data processing pipeline. The litmus-only feature selection approach used the reaction fraction thresholding litmus test (Sec. 10.3.1.2) to select the "best" reaction type for each pair of nuclides and energy groups, for each nuclide in the fuel and both energy groups. All available features were used for each selected reaction type, nuclide and energy group. No dimensionality reduction techniques (Sec. 10.4) were applied to the selected features. The *k*-means, agglomerative, BIRCH and Gaussian Mixture Model (GMM) clustering algorithms were separately used to train predictors for varying numbers of clusters (Sec. 10.5) to inform pin-wise spatial homogenization.

Each of the six benchmarks was modeled with OpenMOC using MGXS generated by the *i*MGXS spatial homogenization scheme with the same runtime parameters as those used in Chap. 8 for infinite, null and degenerate homogenization, and in Chap. 9 for LNS homogenization. The eigenvalue bias is presented in Sec. 11.1.1, while the U-238 capture rate errors are analyzed in Sec. 11.1.2. The pin-wise fission rate errors for the *i*MGXS scheme are not considered here since MGXS clustering was previously shown to be largely inconsequential for fission rate predictions.

11.1.1 Eigenvalues

The OpenMOC eigenvalues were compared to the reference OpenMC eigenvalues from Tab. 7.1. The eigenvalue bias $\Delta\rho$ was computed from Eqn. 5.1 in units of pcm. The bias is listed in Tab. 11.1 for each benchmark, and varying numbers of clusters for each clustering algorithm. The same trends highlighted in Sec. 8.4.1 observed from the null and degenerate biases in Tab. 8.4 remain true for *i*MGXS spatial homogenization. The *i*MGXS eigenvalues are within 10 pcm of those computed with both null and degenerate homogenization for all benchmarks. As previously noted in Sec. 8.4.1, this is expected since the MGXS for the null, degenerate and *i*MGXS schemes are homogenized from the same flux and should preserve globally-integrated reaction rates. Hence, neither

the type of clustering algorithm nor the number of clusters is expected to systematically impact OpenMOC's eigenvalue predictions. Nevertheless, the consistent eigenvalue biases do support the conclusion that the *i*MGXS scheme is properly implemented.

Table 11.1: OpenMOC eigenvalue bias $\Delta\rho$ for *i*MGXS spatial homogenization.

Benchmark	Clustering Algorithm	# Clusters					
		Null	2	4	8	16	Degen.
1.6% Assm	Agglomerative	-168	-168	-168	-168	-168	-168
	BIRCH		-168	-168	-168	-168	
	GMM		-168	-168	-168	-168	
	k-means		-168	-168	-168	-168	
3.1% Assm	Agglomerative	-194	-194	-194	-194	-194	-194
	BIRCH		-194	-194	-194	-194	
	GMM		-194	-194	-194	-194	
	k-means		-194	-194	-194	-194	
3.1% Assm w/ 20 BPs	Agglomerative	-240	-237	-236	-235	-235	-235
	BIRCH		-237	-236	-235	-235	
	GMM		-236	-235	-236	-235	
	k-means		-236	-236	-236	-235	
2×2 Colorset	Agglomerative	-191	-189	-189	-189	-189	-188
	BIRCH		-189	-189	-189	-188	
	GMM		-188	-189	-188	-189	
	k-means		-188	-189	-188	-189	
2×2 Colorset w/ Reflector	Agglomerative	-141	-136	-136	-134	-129	-141
	BIRCH		-138	-137	-136	-132	
	GMM		-136	-136	-137	-132	
	k-means		-136	-136	-134	-132	
BEAVRS Quarter Core	Agglomerative	-122	-115	-117	-117	-117	-116
	BIRCH		-115	-116	-115	-117	
	GMM		-120	-116	-116	-117	
	k-means		-118	-115	-117	-117	

[85]Null homogenization is equivalent to *i*MGXS homogenization with a single cluster.
[86]Degenerate homogenization is equivalent to *i*MGXS homogenization with a cluster for each fuel pin.

The OpenMOC eigenvalues for *i*MGXS homogenization are consistent to within 10 pcm for all numbers of clusters due to the preservation of global reactivity.

11.1.2 U-238 Capture Rates

The OpenMOC energy-integrated pin-wise U-238 capture rates were compared to the reference OpenMC capture rates for *i*MGXS homogenization to compute the percent relative errors for each pin's capture rates. Sec. 11.1.2.1 investigates the dependence of the maximum and mean errors with the number of clusters, while Sec. 11.1.2.2 compares the errors between null, degenerate and *i*MGXS homogenization. Sec. 11.1.2.3 illustrates the spatial distributions of U-238 capture rate errors. Finally, Sec. 11.1.2.5 compares OpenMOC's U-238 capture rate predictions for different clustering models to understand the relative impact of clustered MGXS for different types of fuel pins.

11.1.2.1 Variation with the Number of Clusters

This section compares the U-238 capture rate for each of the four clustering algorithms as the number of clusters is varied. The simplest clustering model assigns all fuel pin instances to the same cluster, which is equivalent to null homogenization; conversely, the most complex clustering model assigns each fuel pin instance to its own unique cluster, which is equivalent to degenerate homogenization. Based on the results for LNS homogenization, it is expected that the U-238 capture rate error can approach that of degenerate homogenization with only perhaps 10 – 20 clusters for the assembly and colorset benchmarks. Beyond this point, it is expected that there are diminishing returns to increased model complexity.

Furthermore, the balance between model complexity and accuracy will depend on the degree of statistical uncertainty present in the MC tally data, as depicted in Fig. 11-1. In particular, the statistical uncertainties of the "noisy" MC tally data for X particle histories may preclude the *i*MGXS scheme from achieving the asymptotic accuracy of the degenerate scheme with fully converged MGXS. The inclusion of more clusters in *i*MGXS

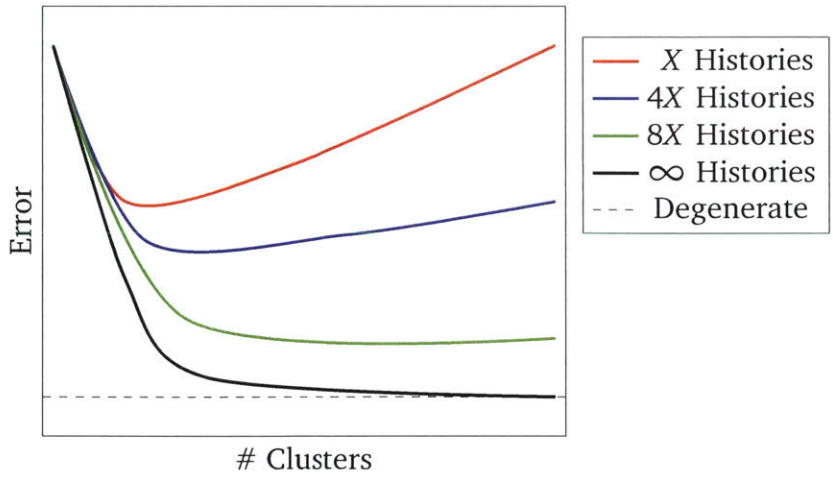


Figure 11-1: Characteristics of reaction rate error convergence with *i*MGXS in the limit of infinite particle histories to degenerate homogenization with fully converged MGXS.

scheme will enable it to approach the asymptotic accuracy of the degenerate scheme if enough particle histories have been simulated to reliably estimate the mean of each MGXS cluster. The optimal balance between accuracy and speed is achieved when the majority of the gap in accuracy between the null and degenerate schemes is eliminated with only a few clusters in *i*MGXS, which quickly converge with only a fraction of the particle histories needed to converge the MGXS for the degenerate scheme.

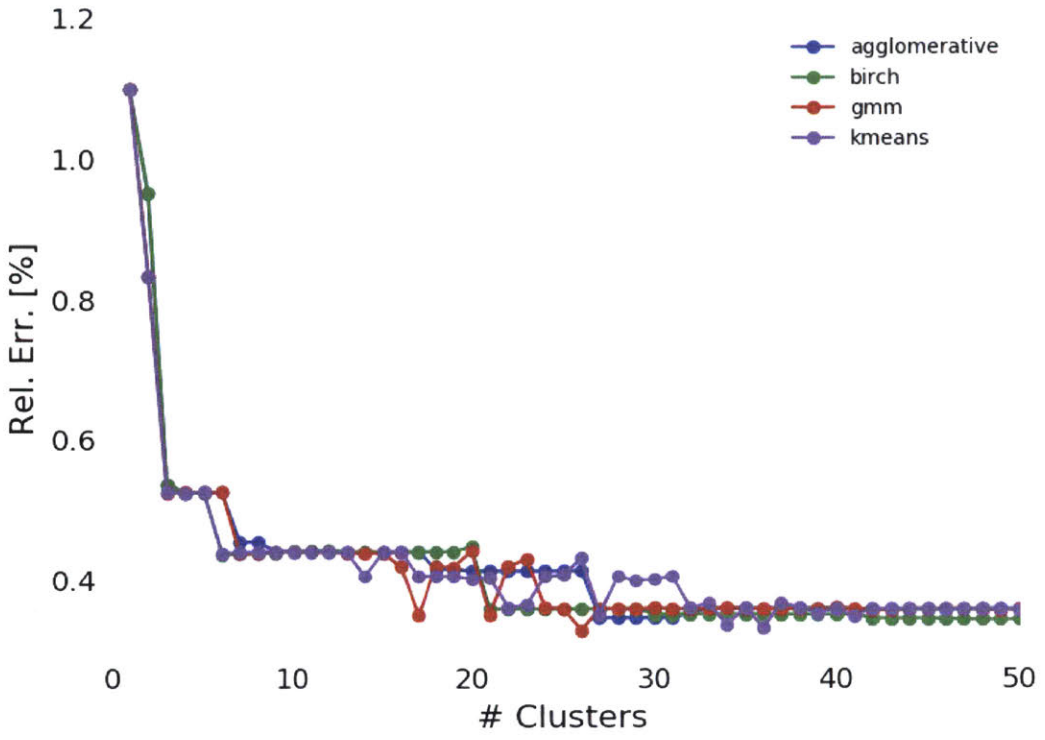
The maximum and mean errors for *i*MGXS with 1 – 50 clusters are illustrated in Figs. 11-2 to 11-7 for the individual assembly, 2×2 colorset and quarter core BEAVRS benchmark models. In particular, the maximum errors are the maximum of the absolute values of the errors, while the mean errors are the averages of the absolute error magnitudes. A couple of key observations can be made with regards to the errors' dependence on the number of clusters, as well as the clustering algorithm. First, it is clear from Figs. 11-2 to 11-4 that the errors for the individual assemblies is greatly reduced with only 4 – 8 clusters, with additional clusters having a marginal impact. As expected, the introduction of spatial heterogeneities requires more clusters to converge the error reduction. For example, both the max and mean errors are still trending downward even with 20 clusters for the 2×2 colorset with a water reflector.

The different clustering algorithms also vary in terms of their impact on the max and mean error. However, the figures do not clearly indicate which if any of the algorithms

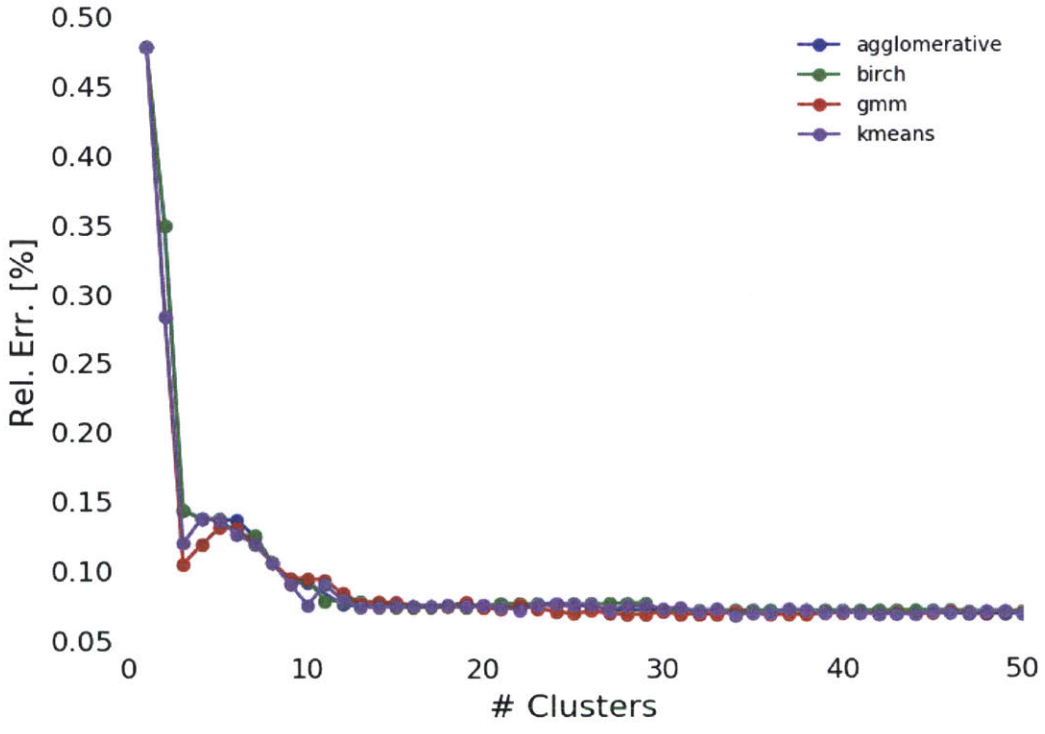
consistently under or over-performs the others for all benchmarks. All four algorithms perform similarly for the 1.6% and 3.1% enriched fuel assemblies without BPs for 1 – 20 clusters, as shown in Figs. 11-2 to 11-3. Although the mean errors are similar for up to 50 clusters for both assemblies, the max errors fluctuate somewhat erratically between 0.2 – 0.4% for 20 – 50 clusters. Similarly, the mean errors for the 3.1% enriched assembly with 20 BPs and the 2×2 colorsets exhibit a consistent downward trend, while the maximum errors separate and fluctuate wildly beyond 10 clusters for the four clustering algorithms. Although no algorithm can be clearly chosen as a "winner," the GMM algorithm seems to most consistently outperform the other clustering algorithms for 10 or more clusters.

Finally, it should be noted that while the mean error is generally a smooth, monotonically decreasing curve, the max error exhibits sharp discontinuities with the addition of clusters. The maximum error likely decreases in a step-like fashion when the addition of a new cluster discriminates those fuel pins with the largest error into their own unique cluster. Evidently, new clusters generally do not refine the MGXS for pins with the worst errors. This is likely due to the fact that the k-means, agglomerative and GMM clustering models optimize a global metric representing the cluster assignments of all points in a dataset, rather than the behavior of the most poorly defined cluster in the model. A revised clustering algorithm may be needed to achieve the maximum error reduction with as few clusters as possible, by successively refining the samples in the most poorly defined clusters. The BIRCH algorithm shows some promise with 10 or fewer clusters, which may be the result of its locally optimized assignment of samples to clusters.

Only a few MGXS clusters are needed to substantially reduce the U-238 capture rate errors, with diminishing returns for the inclusion of more clusters. None of the clustering algorithms is a clear winner, though BIRCH generally performs the best for <10 clusters, while GMMs do better with 10+ clusters.

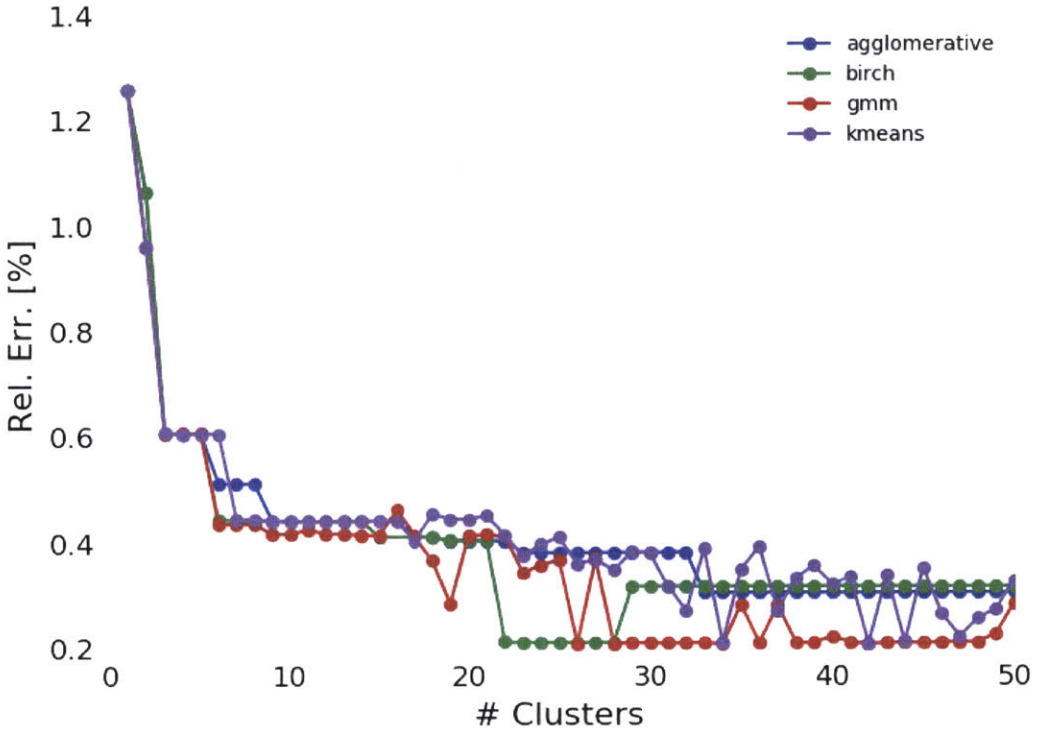


(a)

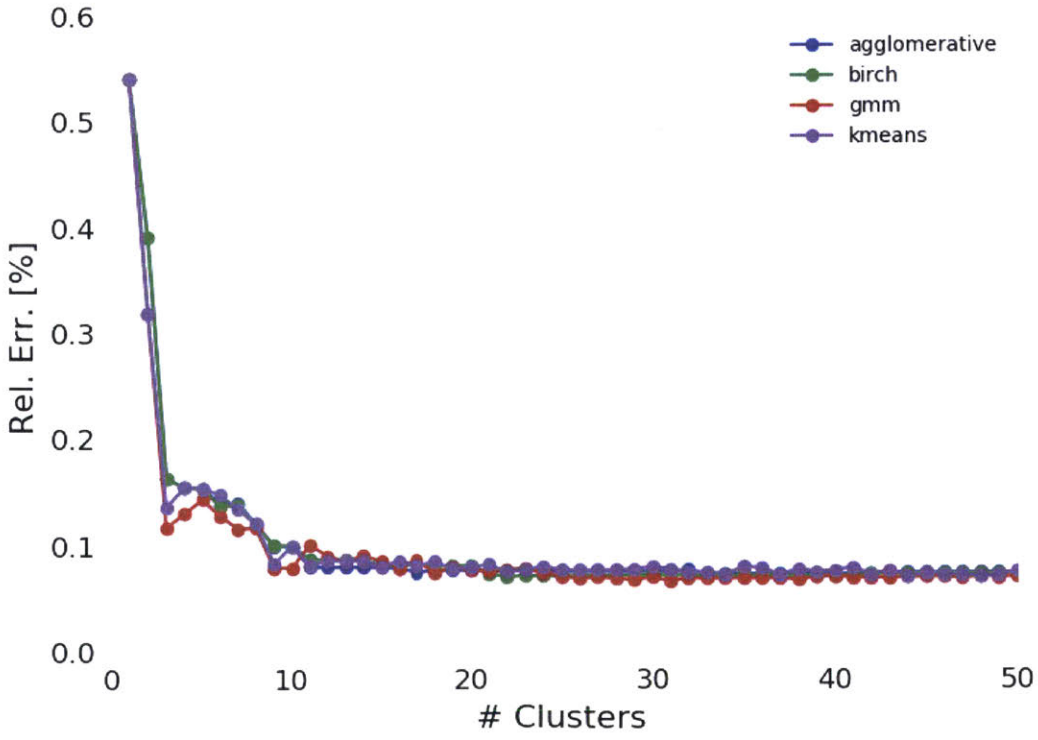


(b)

Figure 11-2: The max (a) and mean (b) U-238 capture rate errors for the 1.6% enriched assembly with *i*MGXS spatial homogenization.

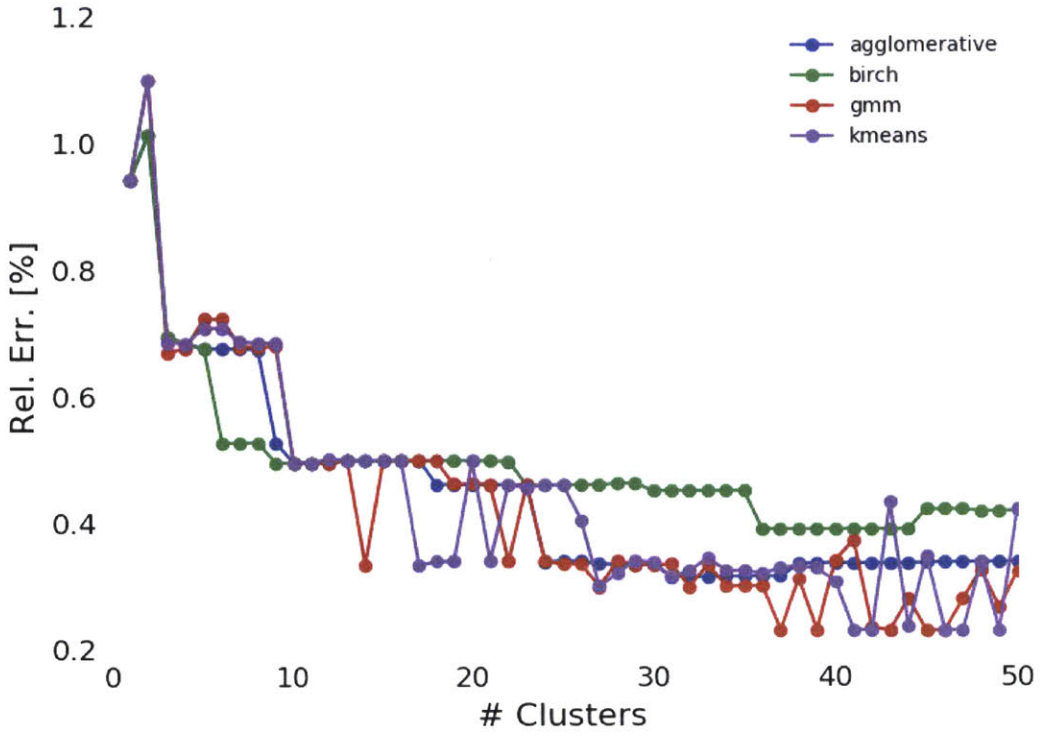


(a)

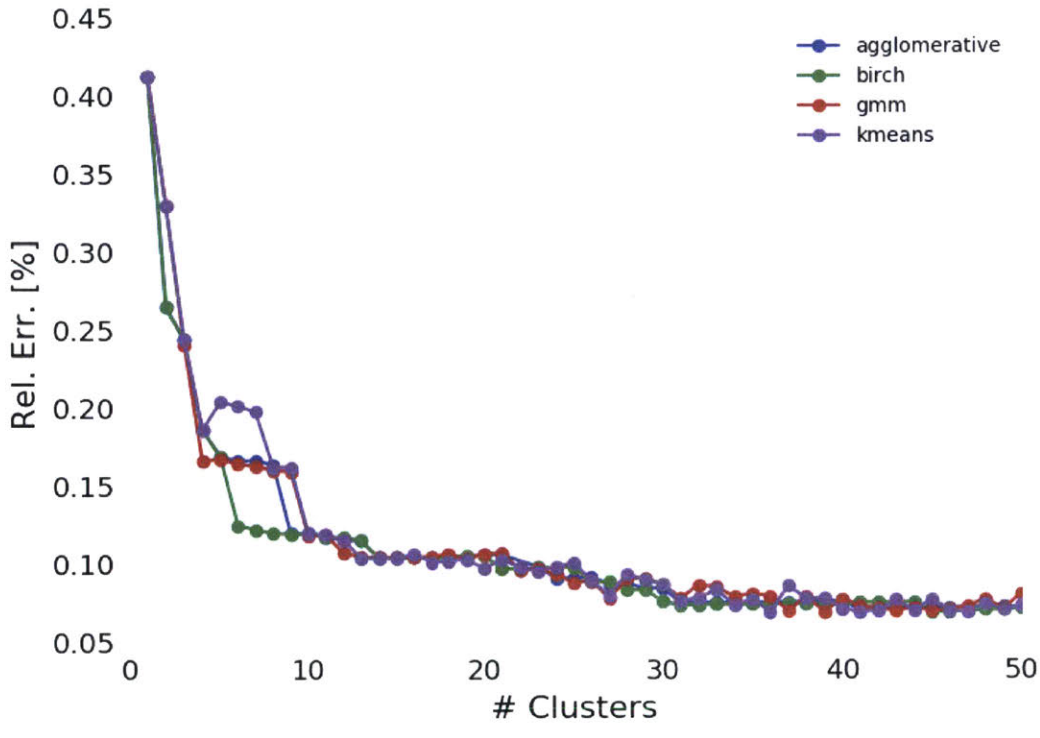


(b)

Figure 11-3: The max (a) and mean (b) U-238 capture rate errors for the 3.1% enriched assembly with *i*MGXS spatial homogenization.

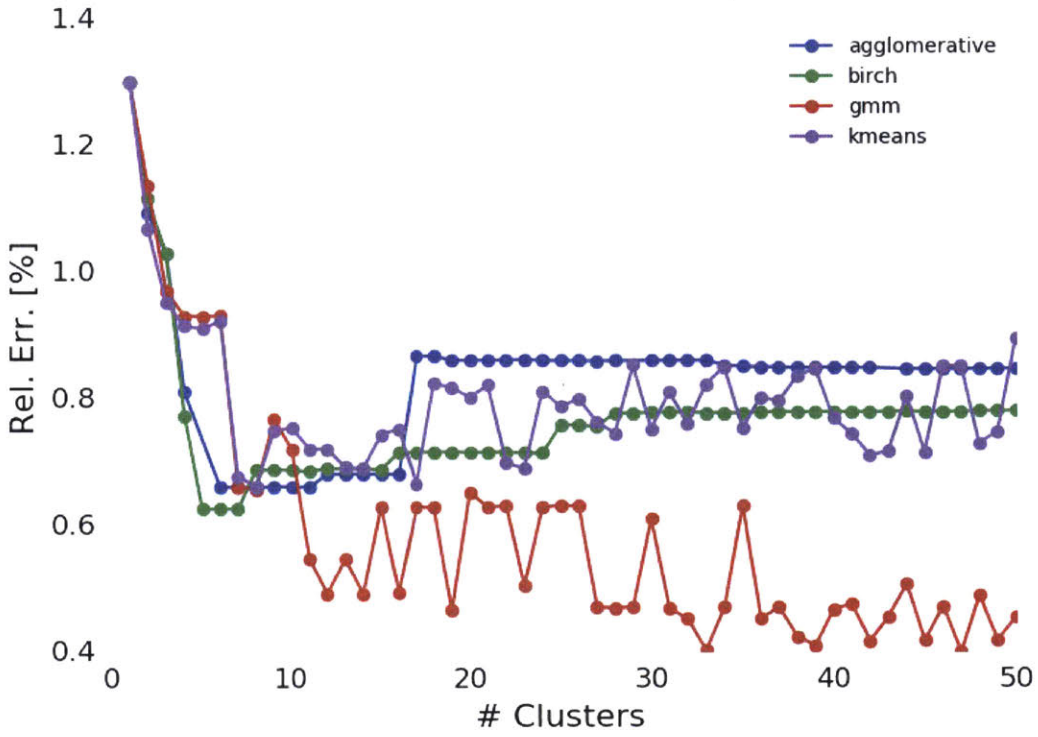


(a)

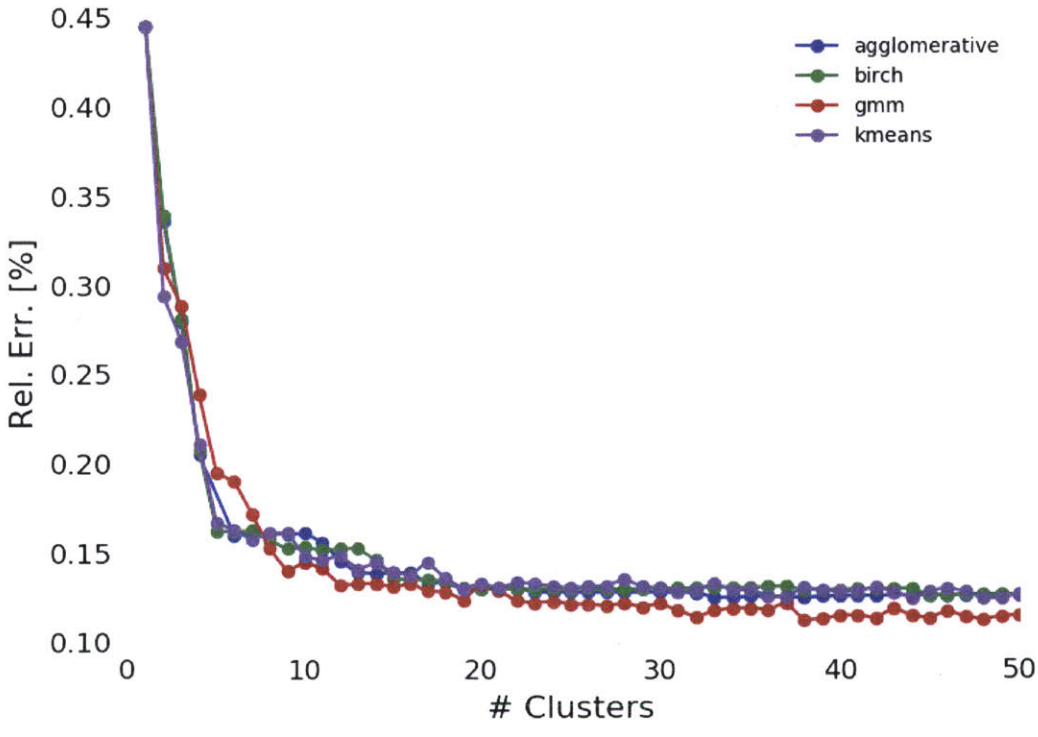


(b)

Figure 11-4: The max (a) and mean (b) U-238 capture rate errors for the 3.1% enriched assembly with 20 BPs with *i*MGXS spatial homogenization.

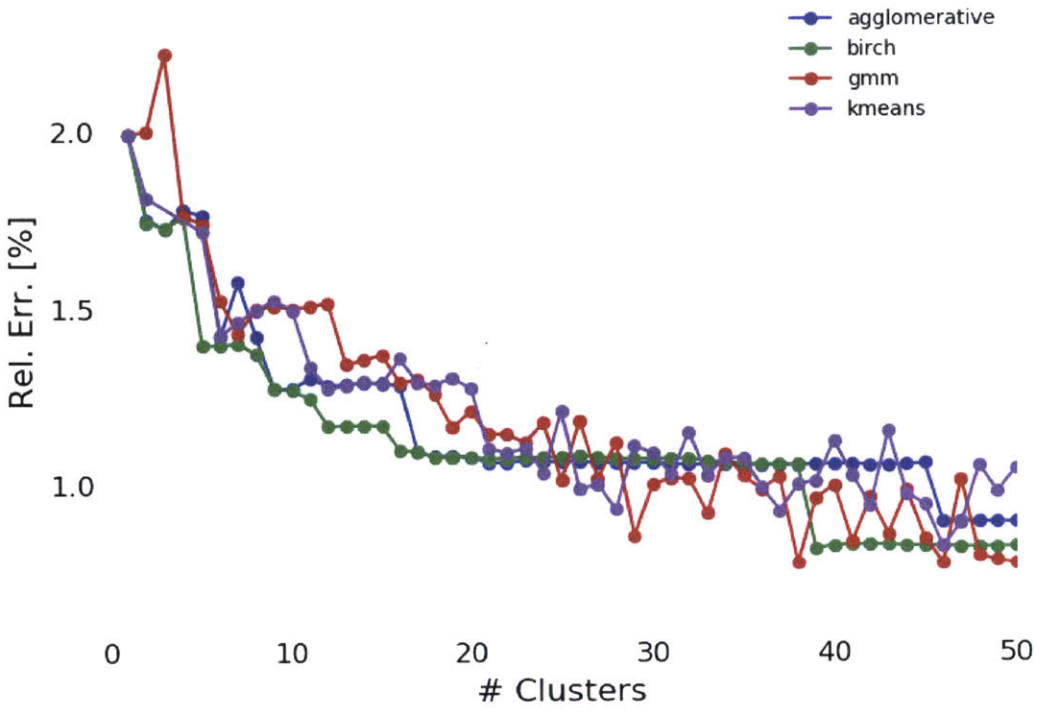


(a)

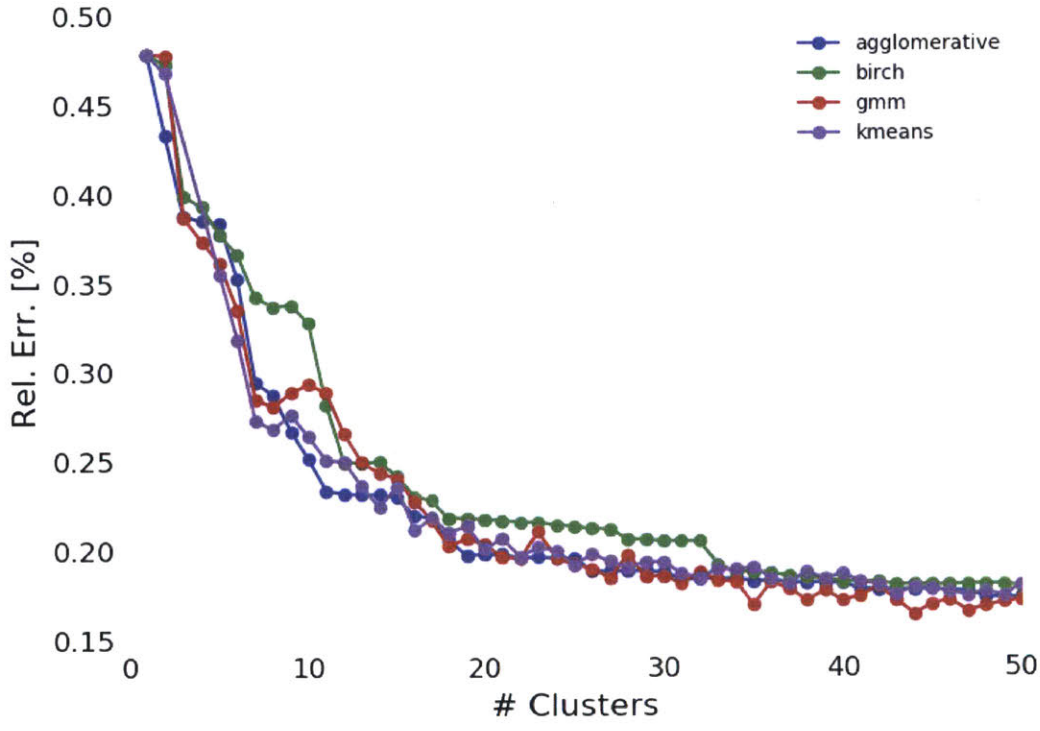


(b)

Figure 11-5: The max (a) and mean (b) U-238 capture rate error for the 2×2 colorset with *i*MGXS spatial homogenization.

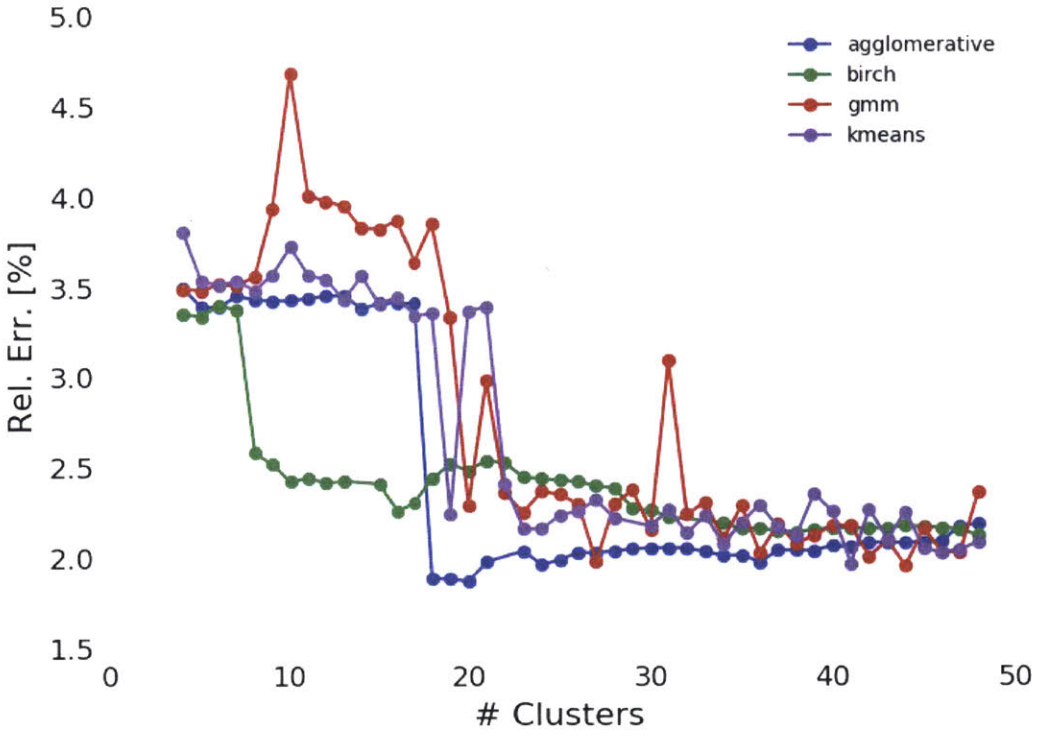


(a)

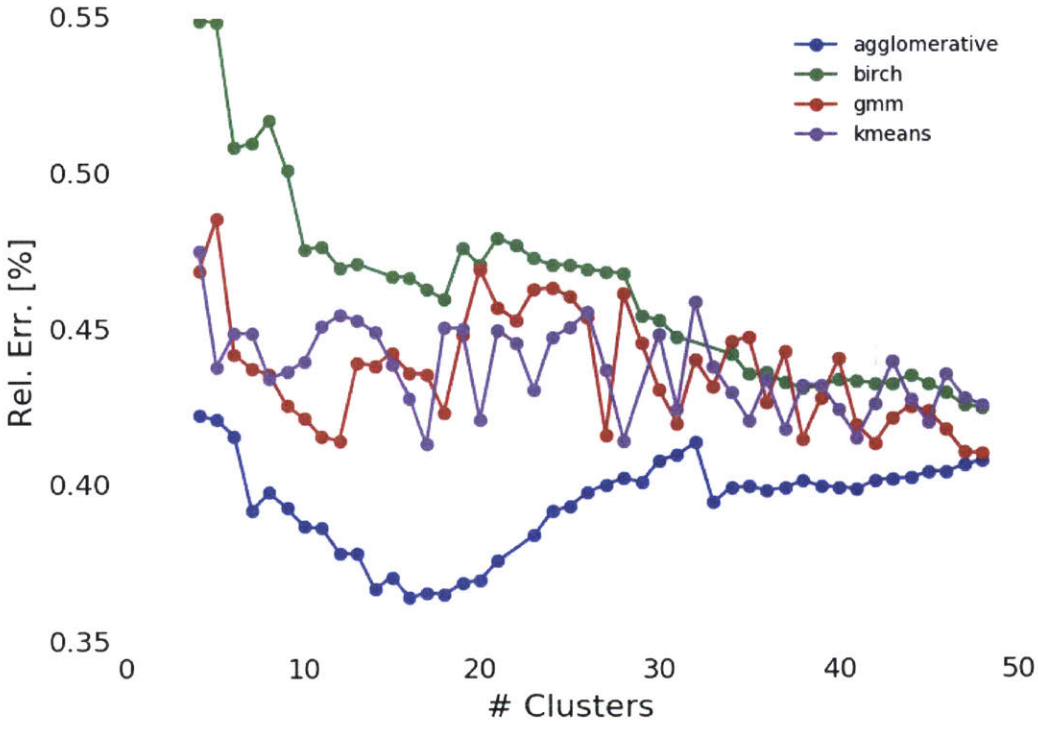


(b)

Figure 11-6: The max (a) and mean (b) U-238 capture rate error for the 2×2 colorset with water reflector with *i*MGXS spatial homogenization.



(a)



(b)

Figure 11-7: The max (a) and mean (b) U-238 capture rate error for the quarter core BEAVRS model with *i*MGXS spatial homogenization.

11.1.2.2 Benchmark with Null and Degenerate Schemes

The maximum and mean percent relative errors for each pin's U-238 capture rates are listed for each benchmark in Tabs. 11.2 and 11.3, respectively, for a variable number of clusters for each clustering algorithm. In particular, the maximum errors are the maximum of the absolute values of the errors along with the appropriate sign, while the mean errors are the averages of the absolute error magnitudes. The errors are tabulated for 2 – 16 clusters, along with null homogenization (one cluster for all pin instances) and degenerate homogenization (one cluster per pin instance).

Since the tables simply represent a subset of the data illustrated in Figs. 11-2 to 11-7, the observations made in Sec. 11.1.2.1 remain valid here. First, the majority of the error is reduced with only a few clusters. For example, 70 – 80% of the reduction in the maximum error enabled with degenerate homogenization can be achieved with only 4 clusters for the 1.6% and 3.1% enriched fuel assembly benchmarks without BPs. Although 16 clusters (or more) are needed to reduce the error to 70% for the 3.1% enriched assembly with 20 BPs, over 75% of the error reduction is achieved with only 4 clusters for the periodic 2×2 colorset. Likewise, the addition of a water reflector to the colorset necessitates the use of 16 or more clusters to reduce the error by 75% or more. In addition, the maximum errors exhibit greater variation across algorithms than the mean errors, as was previously noted in Sec. 11.1.2.1. Finally, the BIRCH and GMM algorithms generally achieve smaller errors than the k-means and agglomerative clustering algorithms for a fixed number of clusters.

Table 11.2: Maximum absolute U-238 capture rate percent relative errors for *i*MGXS spatial homogenization for each clustering algorithm.

Benchmark	Clustering Algorithm	Null	# Clusters 2	4	8	16	Degen.
1.6% Assm	Agglomerative	-1.10	0.95	-0.53	-0.46	-0.44	0.38
	BIRCH		0.95	-0.54	-0.44	-0.44	
	GMM		-0.84	-0.53	-0.44	-0.44	
	k-means		-0.84	-0.53	-0.44	-0.44	
3.1% Assm	Agglomerative	-1.26	1.07	-0.61	-0.52	0.45	-0.33
	BIRCH		1.07	-0.61	0.45	-0.42	
	GMM		-0.96	-0.61	-0.44	-0.42	
	k-means		-0.96	-0.61	-0.52	0.45	
3.1% Assm w/ 20 BPs	Agglomerative	-0.95	-1.02	0.69	0.68	-0.50	-0.30
	BIRCH		-1.02	0.69	-0.53	-0.50	
	GMM		-1.10	0.68	-0.53	-0.50	
	k-means		-1.10	0.69	-0.69	0.34	
2×2 Colorset	Agglomerative	-1.30	-1.09	0.81	0.66	0.68	-0.64
	BIRCH		-1.12	0.77	0.69	0.72	
	GMM		-1.14	0.93	0.77	0.65	
	k-means		-1.07	-0.92	-0.83	0.70	
2×2 Colorset w/ Reflector	Agglomerative	-2.00	-1.76	-1.78	-1.42	-1.28	-0.80
	BIRCH		-1.75	-1.76	-1.37	1.10	
	GMM		-2.00	-1.77	1.50	1.07	
	k-means		-1.82	-1.69	1.35	-1.26	
BEAVRS Quarter Core	Agglomerative	-4.78	-4.36	-3.51	-3.44	-3.43	-3.03
	BIRCH		-4.44	-3.37	-2.60	-2.27	
	GMM		-4.63	-3.50	-3.57	-3.88	
	k-means		-4.38	-3.81	-3.49	-3.46	

Table 11.3: Mean absolute U-238 capture rate percent relative errors for *i*MGXS spatial homogenization for each clustering algorithm.

Benchmark	Clustering Algorithm	Null	# Clusters 2	4	8	16	Degen.
1.6% Assm	Agglomerative	0.48	0.35	0.14	0.11	0.08	0.08
	BIRCH		0.35	0.15	0.11	0.08	
	GMM		0.29	0.12	0.11	0.08	
	k-means		0.29	0.14	0.11	0.08	
3.1% Assm	Agglomerative	0.54	0.39	0.16	0.12	0.08	0.09
	BIRCH		0.39	0.16	0.12	0.08	
	GMM		0.32	0.13	0.10	0.09	
	k-means		0.32	0.16	0.12	0.09	
3.1% Assm w/ 20 BPs	Agglomerative	0.41	0.27	0.19	0.17	0.11	0.09
	BIRCH		0.27	0.19	0.12	0.11	
	GMM		0.33	0.17	0.12	0.11	
	k-means		0.33	0.18	0.20	0.10	
2×2 Colorset	Agglomerative	0.45	0.34	0.21	0.16	0.14	0.15
	BIRCH		0.34	0.21	0.16	0.14	
	GMM		0.31	0.24	0.15	0.14	
	k-means		0.30	0.21	0.16	0.14	
2×2 Colorset w/ Reflector	Agglomerative	0.48	0.43	0.39	0.29	0.22	0.16
	BIRCH		0.47	0.39	0.34	0.23	
	GMM		0.48	0.37	0.29	0.24	
	k-means		0.47	0.37	0.27	0.22	
BEAVRS Quarter Core	Agglomerative	0.49	0.54	0.42	0.40	0.36	0.39
	BIRCH		0.53	0.55	0.52	0.47	
	GMM		0.56	0.47	0.44	0.44	
	k-means		0.53	0.48	0.43	0.43	

11.1.2.3 Spatial Distributions of U-238 Capture Rate Errors

The spatial distributions of capture rate errors are plotted as heatmaps for the assembly and colorset benchmarks in Figs. 11-8 to 11-12 with null, degenerate, LNS and *i*MGXS homogenization with BIRCH clustering of 2, 4 and 16 clusters. The capture rate errors for the full core are illustrated in Fig. 11-13 for null and degenerate homogenization, and in Fig. 11-14 for *i*MGXS homogenization with BIRCH clustering of 4 and 16 clusters.

A couple of key conclusions can be drawn from these results. First, the results from Sec. 11.1.2.2 indicated that the errors with *i*MGXS homogenization with 16 clusters exceed those for degenerate homogenization for both the 3.1% enriched fuel assembly with 20 BPs and the periodic 2×2 colorset. This is revealed in the Figs. 11-10 to 11-11 which illustrate a systematic bias which differs from that of the degenerate case. In particular, the pins facially and diagonally adjacent to the central instrument tube exhibit the smallest and largest errors in the assembly with 20 BPs, respectively. Likewise, the pins between the two rings of BPs in the 3.1% enriched fuel assembly exhibit the largest errors for the periodic colorset. These observations are particularly interesting since the LNS scheme outperforms *i*MGXS for these benchmarks while using substantially fewer materials (9 – 10 LNS materials as compared to 16 *i*MGXS materials per assembly). Although LNS necessarily discriminates those pins adjacent to the instrument tube and BPs into unique clusters, the *i*MGXS scheme only does so if the feature vectors for those fuel pins are well-separated from those for other fuel pins in feature space.

Unlike the LNS scheme, the *i*MGXS scheme excels at discriminating pins along assembly-assembly and assembly-reflector interfaces as shown in Figs. 11-11 to 11-12[87]. This is most obvious for the pins along the assembly-reflector interface which exhibit nearly the same error for both null and LNS homogenization. The *i*MGXS scheme discriminates these pins into unique clusters such that little if any residual bias is observed with 16 clusters. Indeed, the *i*MGXS scheme dedicates unique clusters to those pins along the interfaces *before* discriminating interior pins adjacent to CRGTs and BPs into clusters. For this reason, *i*MGXS may necessarily need to model more unique materials

[87]However, even with 16 clusters, there are some lingering residual errors for pins along the interfaces between the bottom right assembly and its neighbors.

than LNS to achieve the same reduction of the maximum pin-wise error.

Finally, it should be noted that the observations made here are specific to the configuration of the *i*MGXS data processing pipeline. As noted in Sec. 10.8, the choice of features and clustering algorithm has a large impact on the clustering model and the resultant spatial distribution of U-238 capture rate errors. In addition, dimensionality reduction adds another degree of freedom to the types of clustered geometries that may be found with the *i*MGXS scheme. Future work should systematically examine the spatial error distributions for different configurations of the *i*MGXS data processing pipeline to quantify the ability of each to minimize the error with as few clusters as possible.

The *i*MGXS scheme does a better job discriminating fuel pins along assembly-assembly and assembly-reflector interfaces than LNS, but requires more unique materials to achieve the same reduction in U-238 capture rate errors.

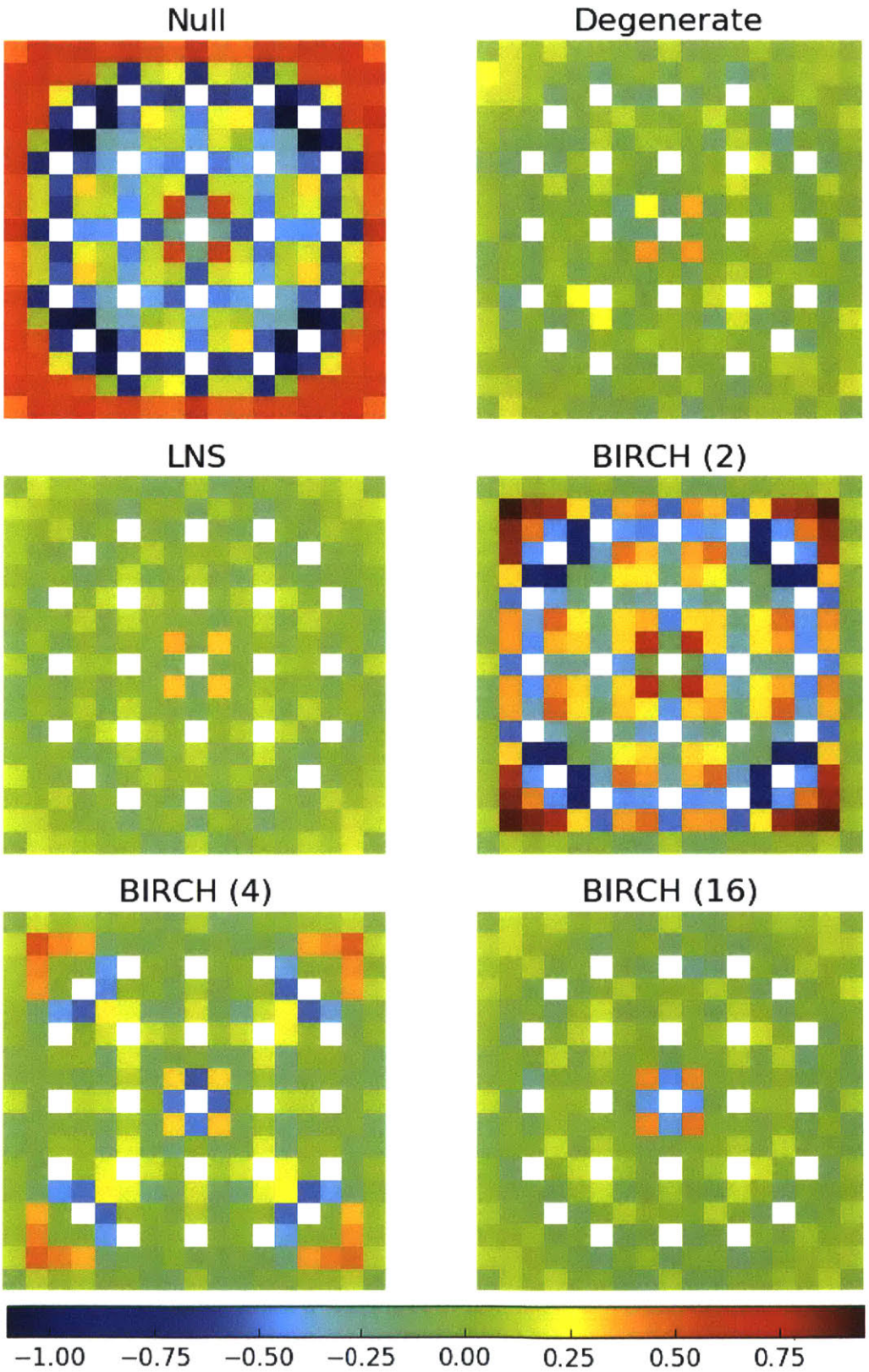


Figure 11-8: U-238 capture percent relative errors for the 1.6% enriched assembly with null, degenerate, LNS and *i*MGXS spatial homogenization.

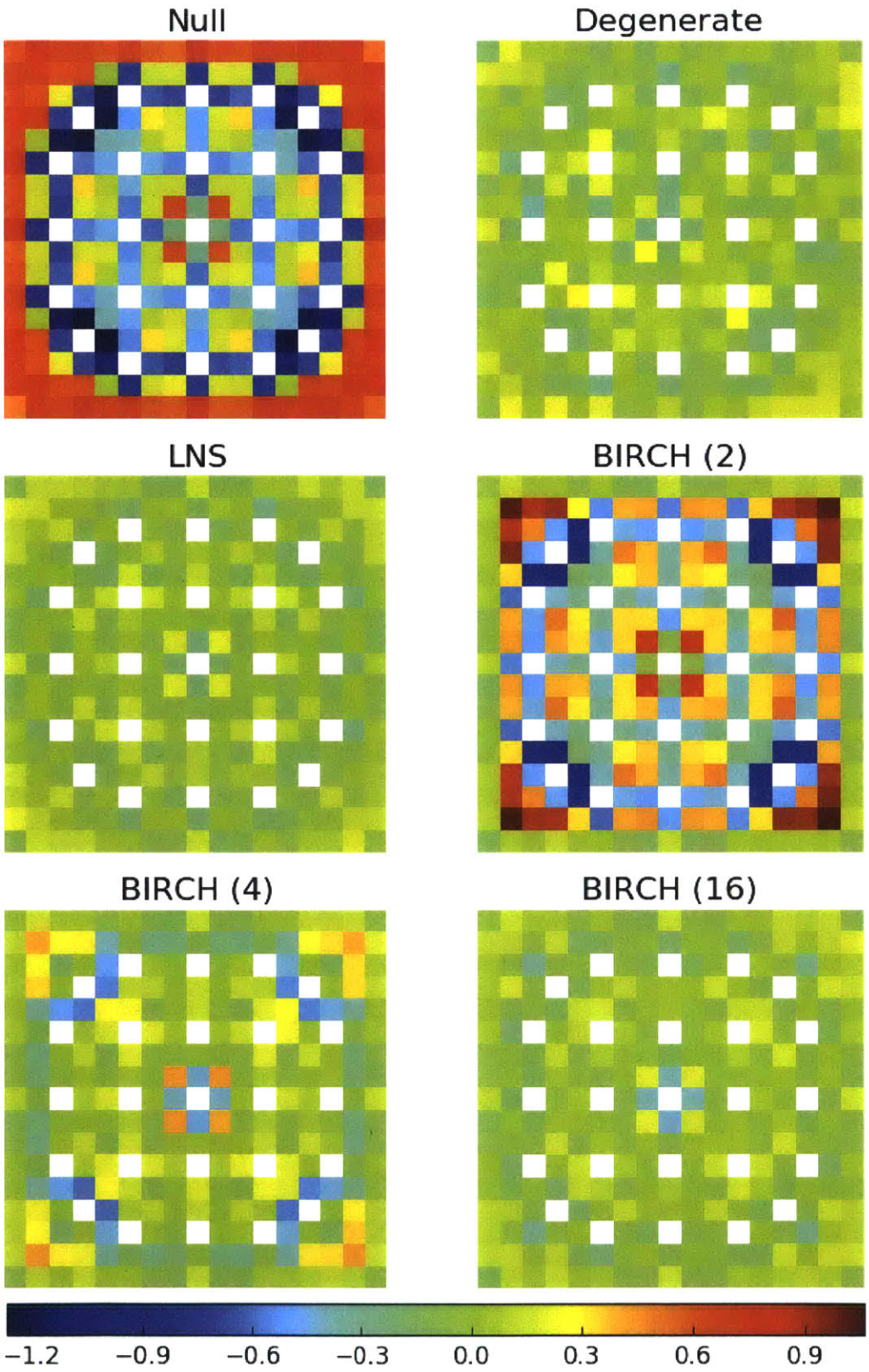


Figure 11-9: U-238 capture percent relative errors for the 3.1% enriched assembly with null, degenerate, LNS and *i*MGXS spatial homogenization.

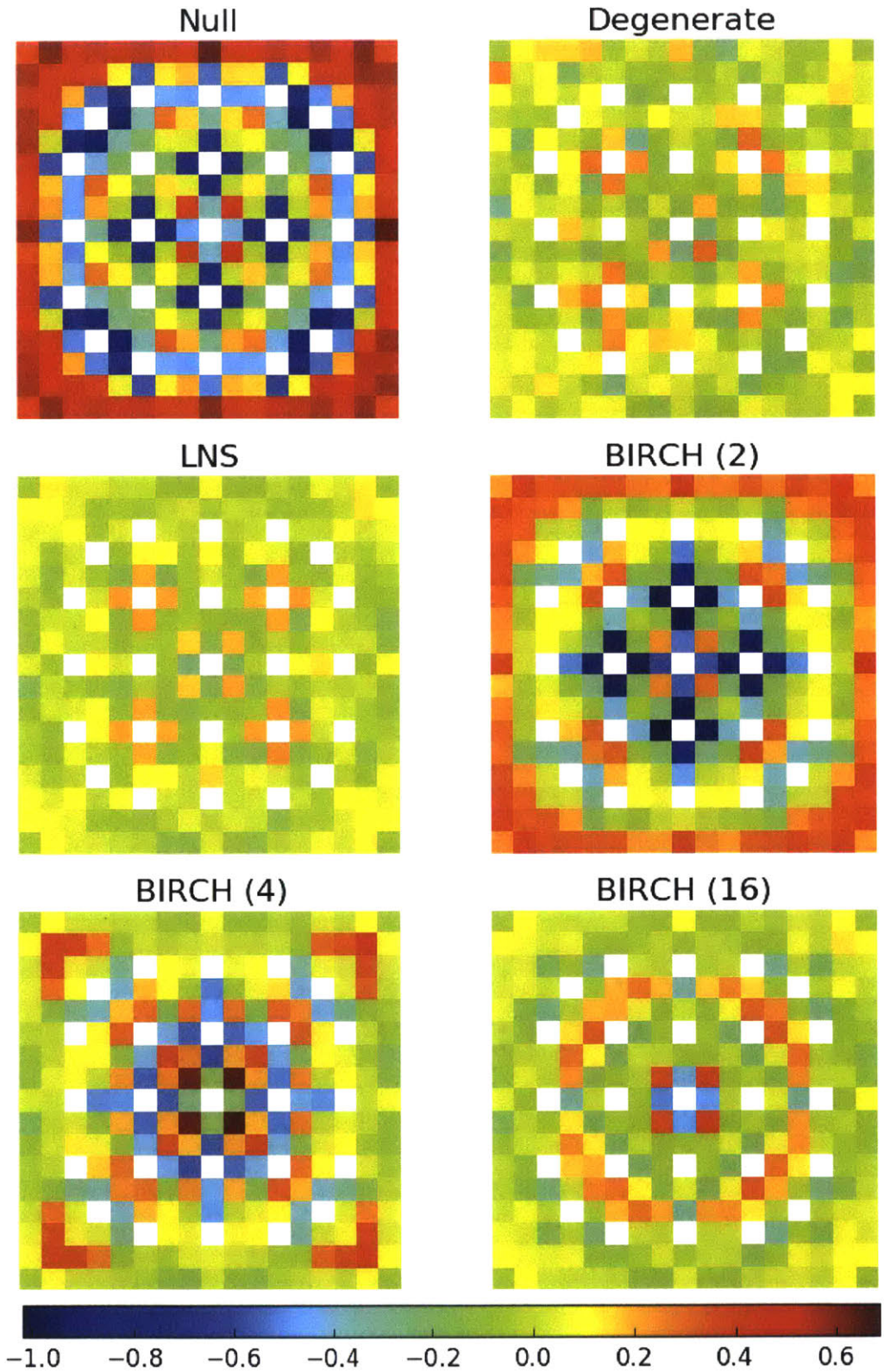


Figure 11-10: U-238 capture percent relative errors for the 3.1% enriched assembly with 20 BPs with null, degenerate, LNS and *i*MGXS spatial homogenization.

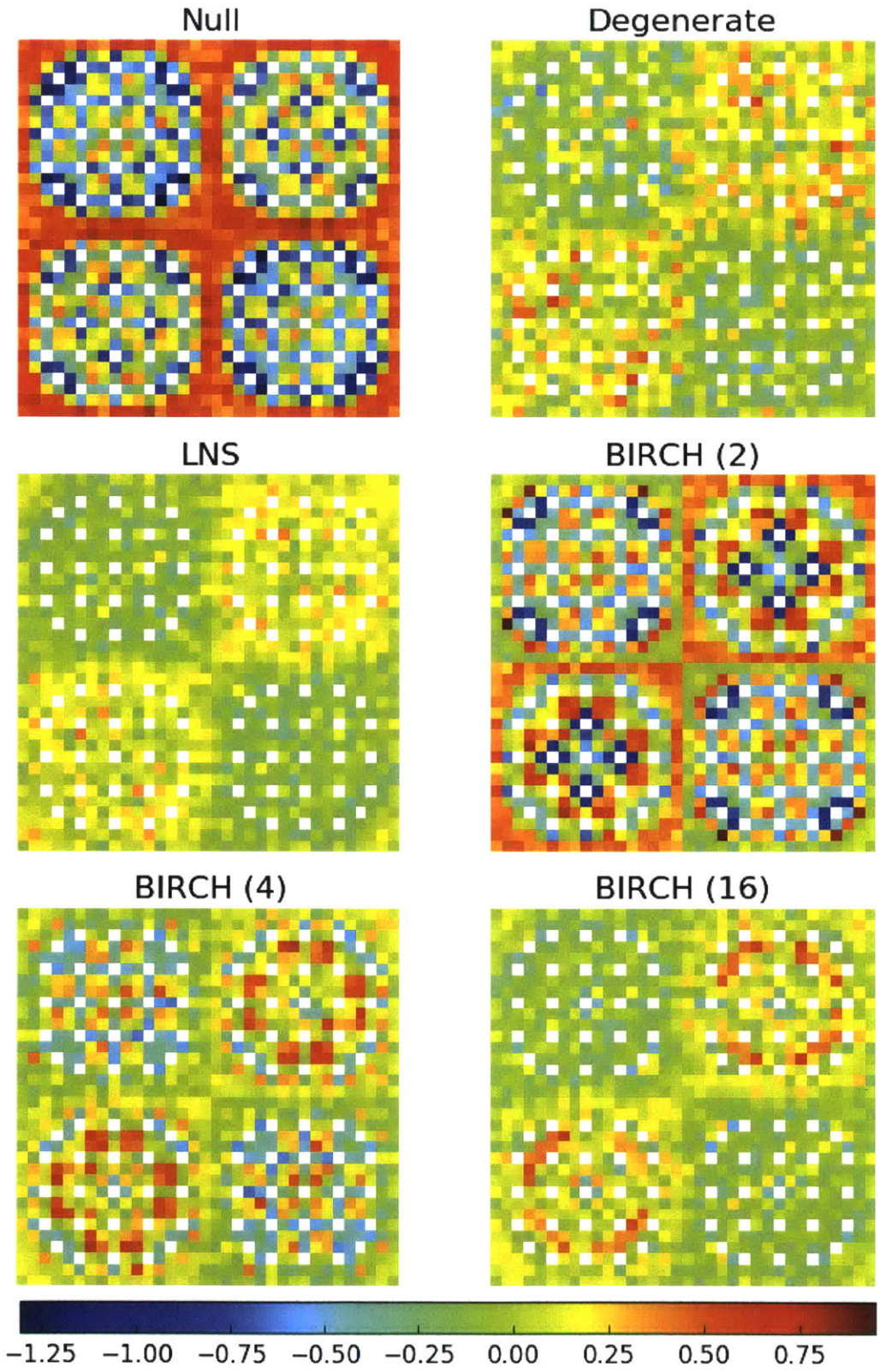


Figure 11-11: U-238 capture percent relative errors for the 2×2 colorset with null, degenerate, LNS and *i*MGXS spatial homogenization.

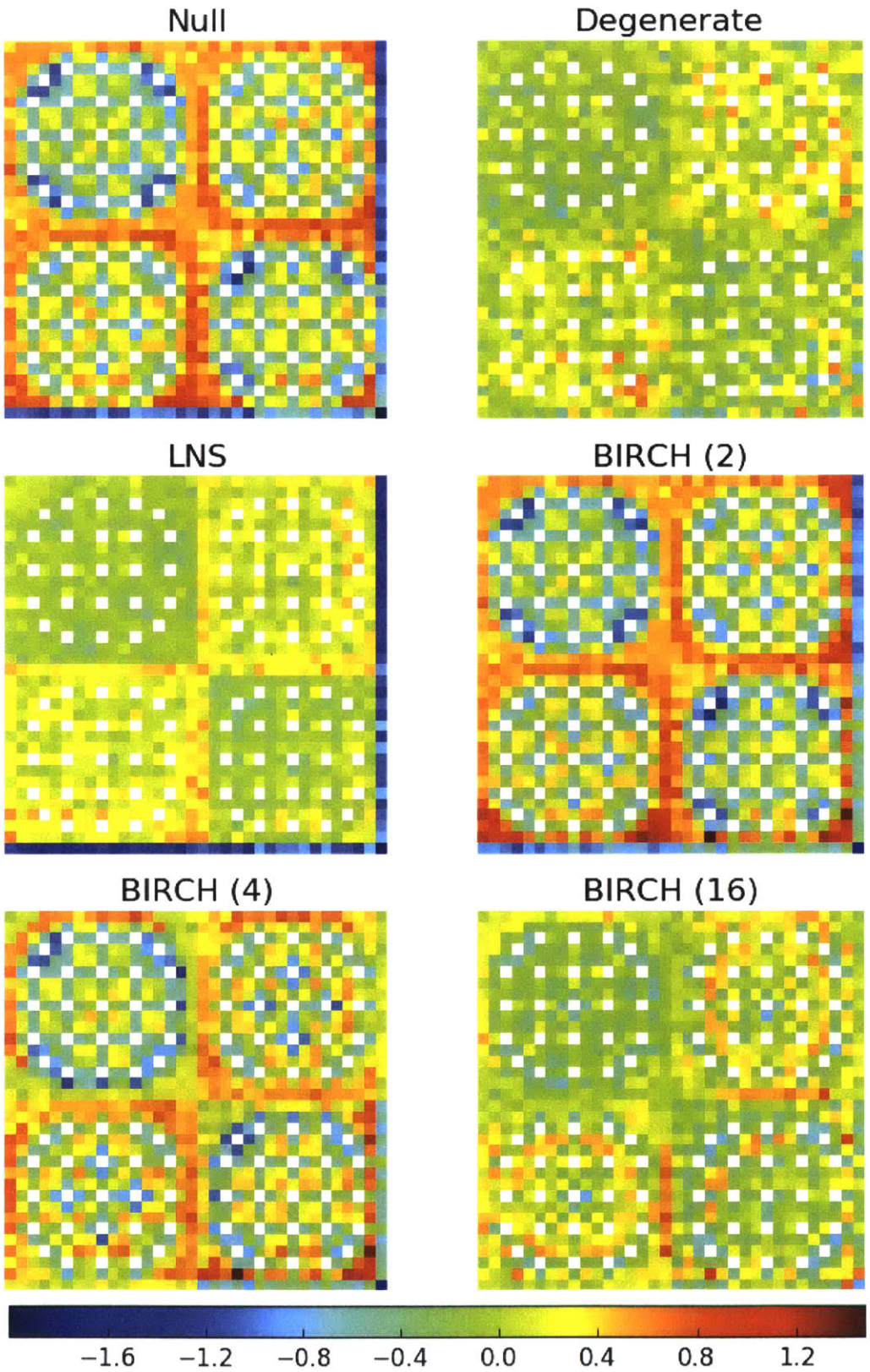
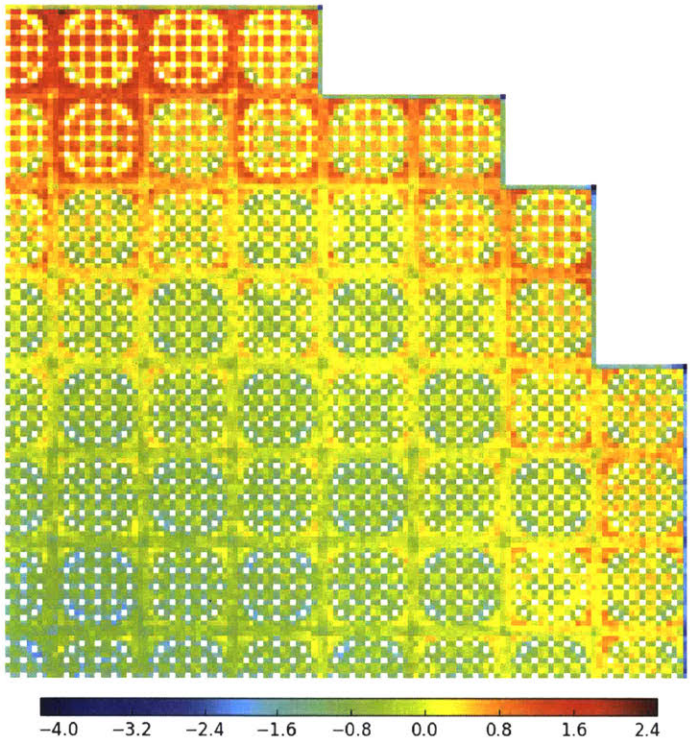
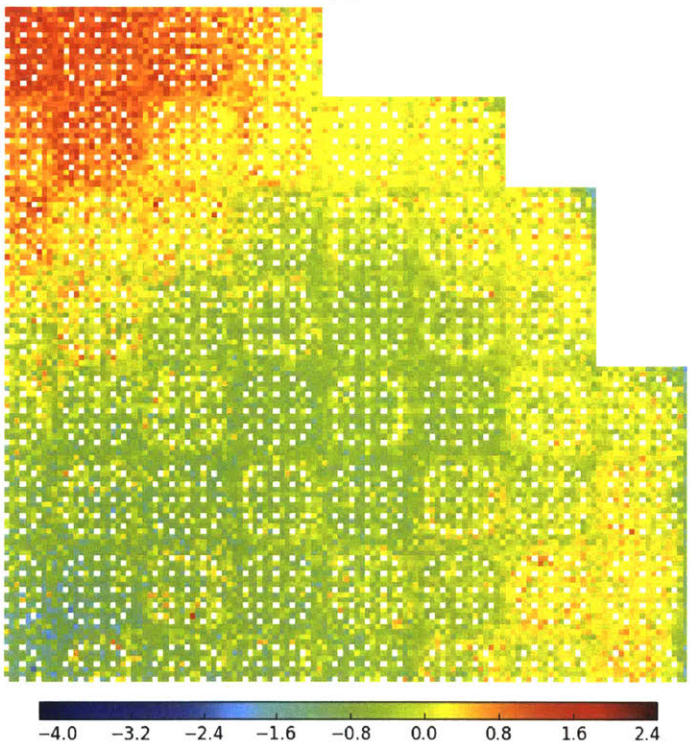


Figure 11-12: U-238 capture percent relative errors for the 2×2 colorset with a water reflector with null, degenerate, LNS and *i*MGXS spatial homogenization.

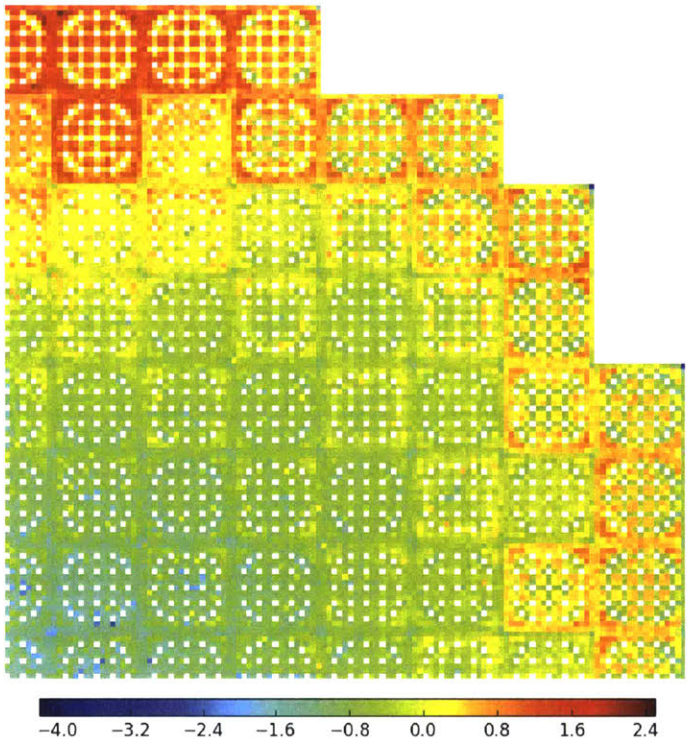


(a)

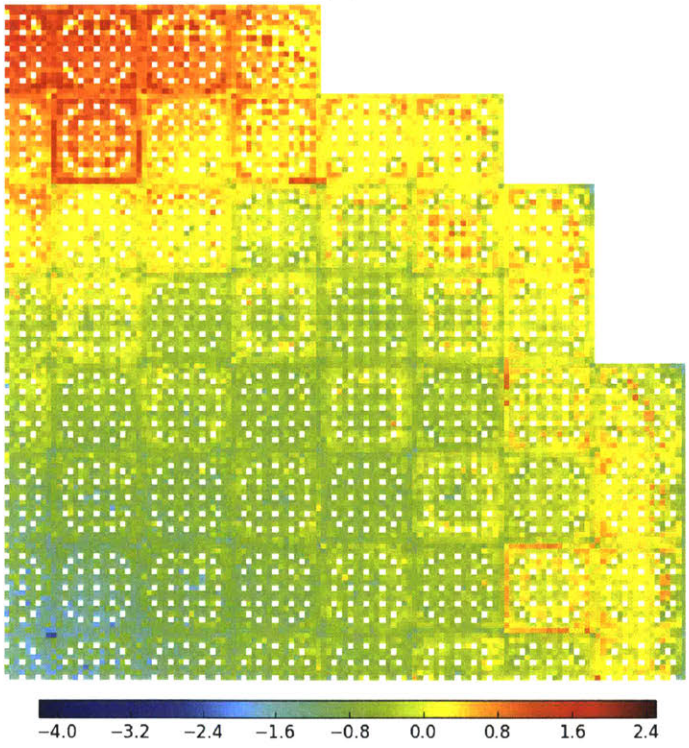


(b)

Figure 11-13: U-238 capture percent relative errors for the quarter core BEAVRS model with null (a) and degenerate (b) spatial homogenization.



(a)



(b)

Figure 11-14: U-238 capture percent relative errors for the quarter core BEAVRS model with *i*MGXS spatial homogenization with BIRCH clustering of 4 (a) and 20 (b) clusters.

11.1.2.4 Spatial Distributions of Capture-to-Fission Errors

The U-238 capture rate errors for the quarter core BEAVRS model shown in Figs. 11-13 and 11-14 exhibit a tilt which peaks near the top center of the core. It is unclear why this tilt is present in the OpenMOC solution, but it may be related to issues encountered with the Coarse Mesh Finite Difference (CMFD) acceleration scheme in OpenMOC for this benchmark[88]. The ratio of the energy-integrated U-238 capture rates to the total fission rates in each fuel pin is an alternative metric to compare OpenMC and OpenMOC solutions. The spatial distributions of the capture-to-fission ratio errors are plotted as heatmaps for the null homogenization scheme in Fig. 11-15, and in Figs.11-16 and 11-17 for the LNS, degenerate and *i*MGXS spatial homogenization schemes. The capture-to-fission ratio is a useful metric since it is insensitive to the magnitude of the flux in each fuel pin, which is distorted by the tilt observed for BEAVRS. Instead, the capture-to-fission ratio indicates whether the shape of the flux in energy is consistent between OpenMC and OpenMOC in each fuel pin[89]. Thus, the capture-to-fission ratio may signify whether localized spatial self-shielding spectral effects are adequately modeled with each pin-wise spatial homogenization scheme.

The figures clearly illustrate that the capture-to-fission ratio errors do not exhibit an asymmetric tilt as do the U-238 capture rate errors. The figures also clearly demonstrate that the errors are significantly reduced by modeling MGXS clustering. The errors for the null scheme are largest for the single row of fuel pins adjacent to the baffle and peak in magnitude at -4.5% for the four pins at the corners of the baffle (Fig. 11-15). In addition, the null scheme leads to systematically positive errors for fuel pins adjacent to CRGTs and along assembly-assembly interfaces. The degenerate scheme largely resolves the systematic errors along the inter-assembly and assembly-baffle interfaces (Fig. 11-16b), reducing the magnitude of the errors from -4.5% – 1.7% to -1.7% – 1.2%. The LNS scheme also eliminates much of the systematic error distribution (Fig. 11-16a), and

[88]The CMFD acceleration scheme exhibited some instabilities for the BEAVRS model, and as a result, it was necessary to use a quarter pin-wise CMFD mesh in OpenMOC (Sec. 8.3.3) in order to stabilize the scheme. Even still, it is possible that the CMFD scheme was insufficiently converged for the BEAVRS model, and the presence of a higher order eigenmode may have tilted the flux across the core.

[89]Most U-238 capture occurs in the resonance region, while most fission occurs at thermal energies.

the error distribution is more smoothly varying than the relatively "noisy" distribution observed for degenerate homogenization[90]. However, the fuel pins adjacent to the baffle still have errors as large as -3.5% with the LNS scheme since it neglects to distinguish their MGXS from those pins on the interior boundaries of each assembly. The use of only four clusters with *i*MGXS largely reduces the errors for many of the interior fuel pins, and greatly reduces the errors for the pins adjacent to the baffle, but fails to fully distinguish the fuel pins at the corners of the baffle (Fig. 11-17a). When the *i*MGXS scheme is employed with more (*i.e.*, 20) clusters, the fuel pins along the inter-assembly interfaces, and those pins at the corners of the baffle, are distinguished into unique clusters, reducing the errors for all pins to within -1.2% – 1.2% (Fig. 11-17b).

[90]The "noisy" error distribution in Fig. 11-16b is likely due in part to the non-negligible MC statistical uncertainties for the MGXS in each fuel pin for degenerate homogenization.

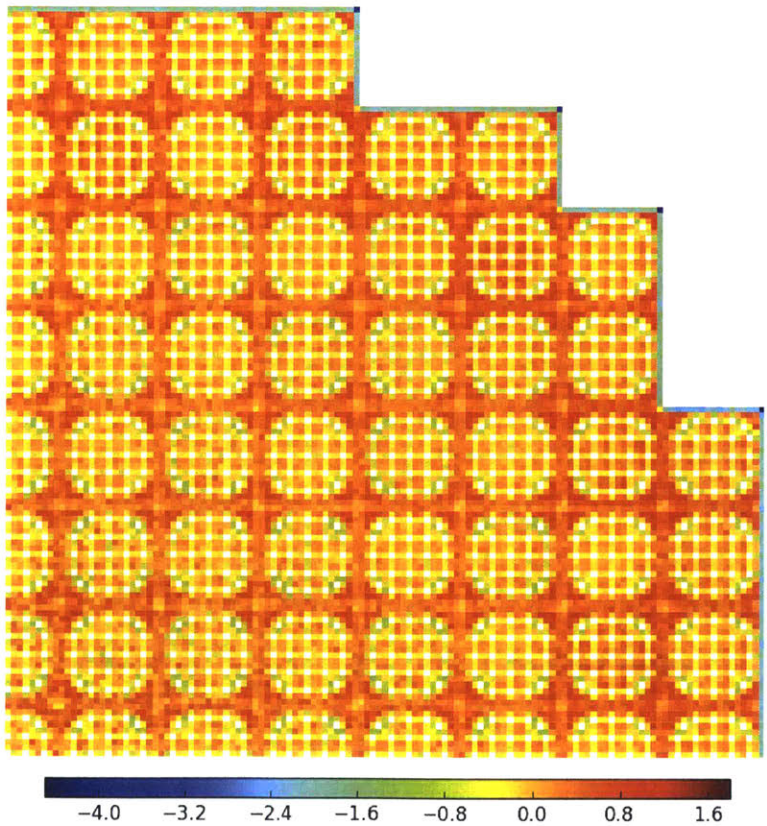
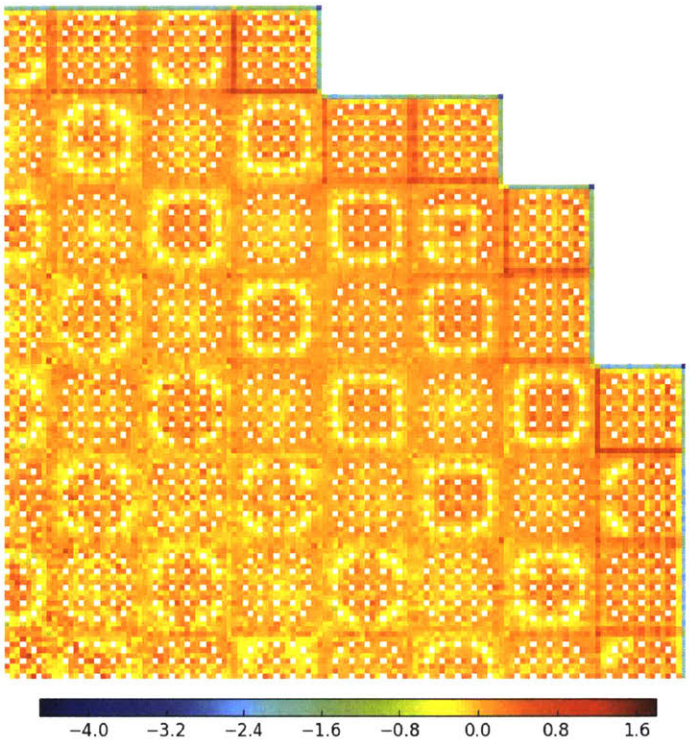
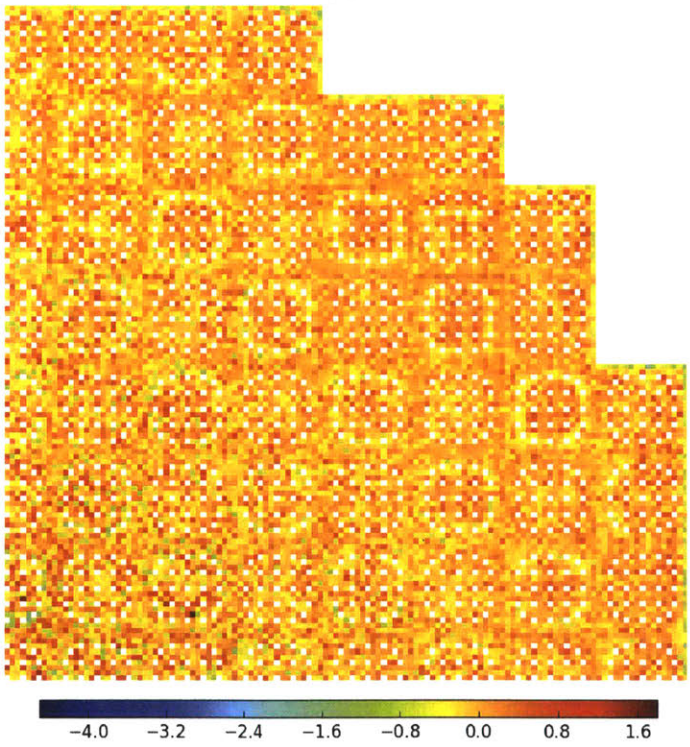


Figure 11-15: Percent relative errors on the U-238 capture to total fission rate ratio for the quarter core BEAVRS model with null spatial homogenization.

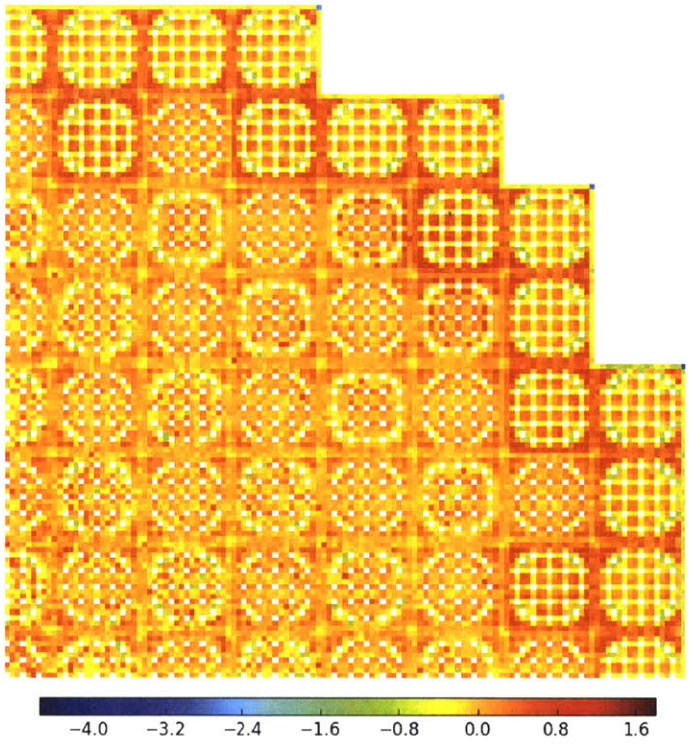


(a)

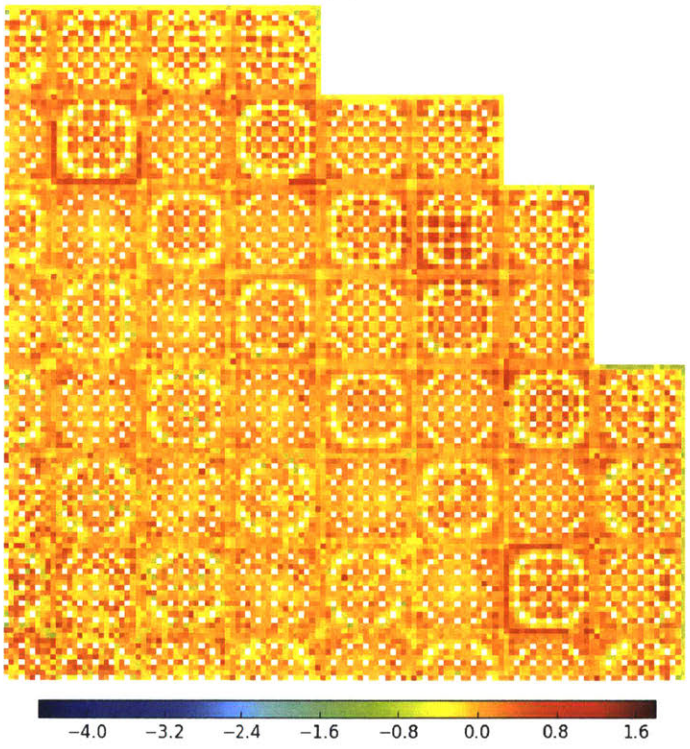


(b)

Figure 11-16: Percent relative errors on the U-238 capture to total fission rate ratio for the quarter core BEAVRS model with LNS (a) and degenerate (b) spatial homogenization.



(a)



(b)

Figure 11-17: Percent relative errors on the U-238 capture to total fission rate ratio for the quarter core BEAVRS model with *i*MGXS spatial homogenization with BIRCH clustering of 4 (a) and 20 (b) clusters.

11.1.2.5 A Comparison of OpenMOC Solutions

This section directly compares the reaction rates from two different OpenMOC simulations with null and *i*MGXS homogenization to better understand the impact of MGXS clustering on the spatial distribution of U-238 capture rate errors. In particular, the percent relative deviation Δ_k^{iMGXS} of the OpenMOC U-238 capture rates for fuel pin instance k between the null and *i*MGXS schemes is computed as follows:

$$\Delta_k^{iMGXS}\ [\%] = \frac{\left[\sum_{g=1}^{G} \hat{\Sigma}_{\gamma,k,g}^{238} \phi_{k,g}\right]^{iMGXS} - \left[\sum_{g=1}^{G} \hat{\Sigma}_{\gamma,k,g}^{238} \phi_{k,g}\right]^{Null}}{\left[\sum_{g=1}^{G} \hat{\Sigma}_{\gamma,k,g}^{238} \phi_{k,g}\right]^{Null}} \times 100 \tag{11.1}$$

Unlike the U-238 capture rate error distributions, the relative deviations are not sensitive to the statistical uncertainties of the OpenMC reference solutions. As a result, it is easier to visualize the impact of clusters on the U-238 capture rate predictions. Figs. 11-18 to 11-22 present the percent relative deviations for the individual assembly and 2×2 colorset benchmarks for BIRCH clustering of 2, 4, 8 and 16 clusters. Likewise, Figs. 11-23 to 11-25 illustrate the deviations for the quarter core BEAVRS model with k-means, GMM and BIRCH clustering of 4, 8, 16 and 20 clusters, respectively.

The figures illustrate the hierarchical nature of MGXS clustering by discriminating fuel pins with different spatial self-shielding effects that occur in different regimes. For example, the first 2 clusters used for the individual assemblies distinguishes fuel pins which are adjacent (or nearly adjacent) to CRGTs, BPs or instrument tubes from those which are only surrounded by other fuel pins. The introduction of more clusters further refines these clusters to identify fuel pins with neighbors which have similar spatial self-shielding effects. For example, as the *i*MGXS scheme transitions between 8 – 16 clusters for the 3.1% enriched fuel assembly with 20 BPs (Fig. 11-20), the fuel pins that are facially and diagonally adjacent to two CRGTs are separated into their own cluster. As a result, these eight fuel pins have the largest relative deviation from null homogenization since they experience the largest amount of moderation from the two

CRGTs. Similarly, the first 2 – 4 clusters in the colorset with a reflector (Fig. 11-22) discriminate pins along the assembly-assembly and assembly-reflector interfaces. As more clusters are introduced, they are increasingly customized to model the more local spatial self-shielding effects affecting the pins within the interior of each assembly due to the presence of CRGTs and BPs.

The percent relative deviations between OpenMOC solutions with null and *i*MGXS homogenization makes it easy to discern the hierarchical nature of MGXS clustering and its impact on U-238 capture rate predictions.

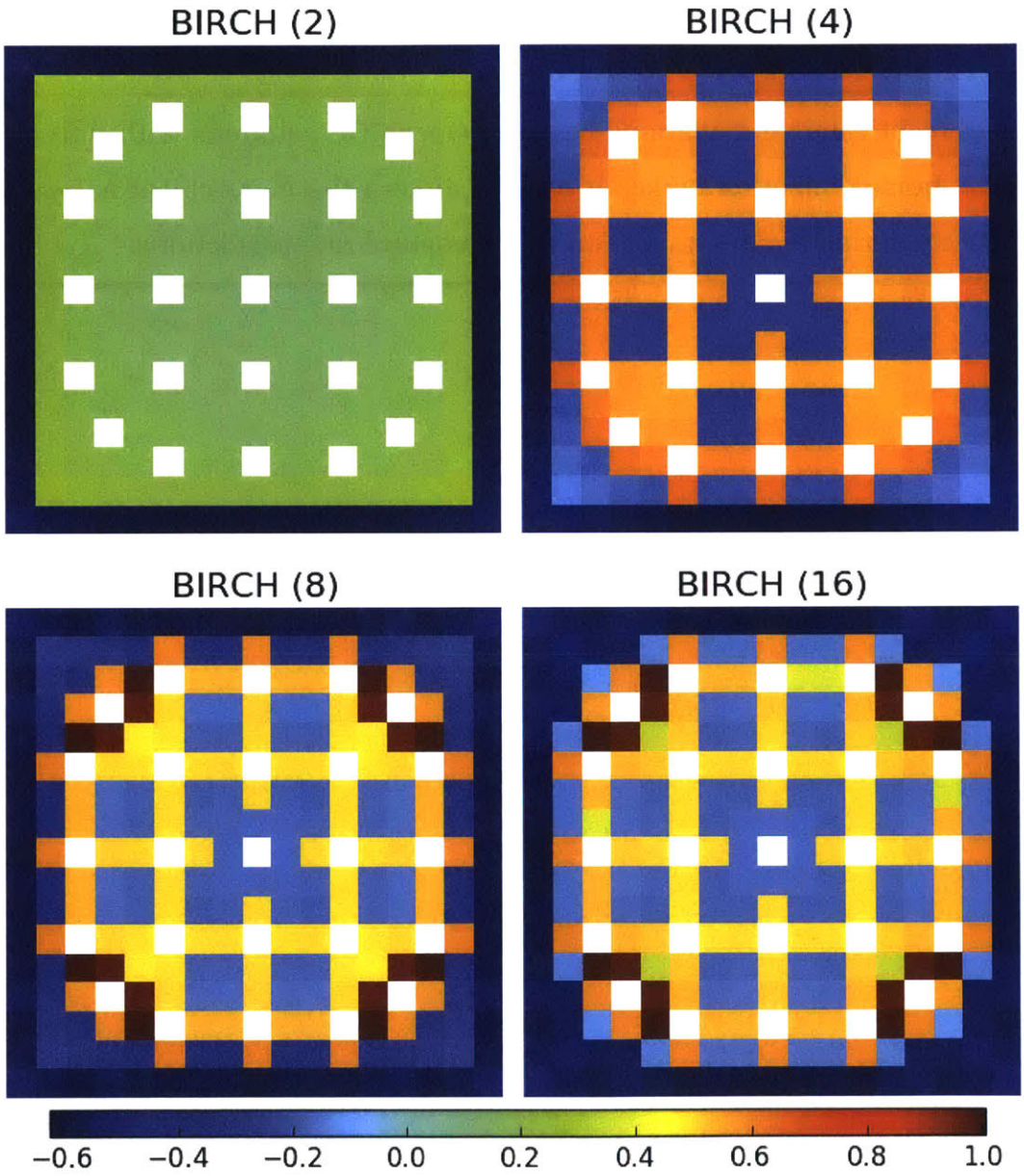


Figure 11-18: The percent relative deviation of U-238 capture rate spatial distributions for *i*MGXS and null homogenization for a 1.6% enriched assembly.

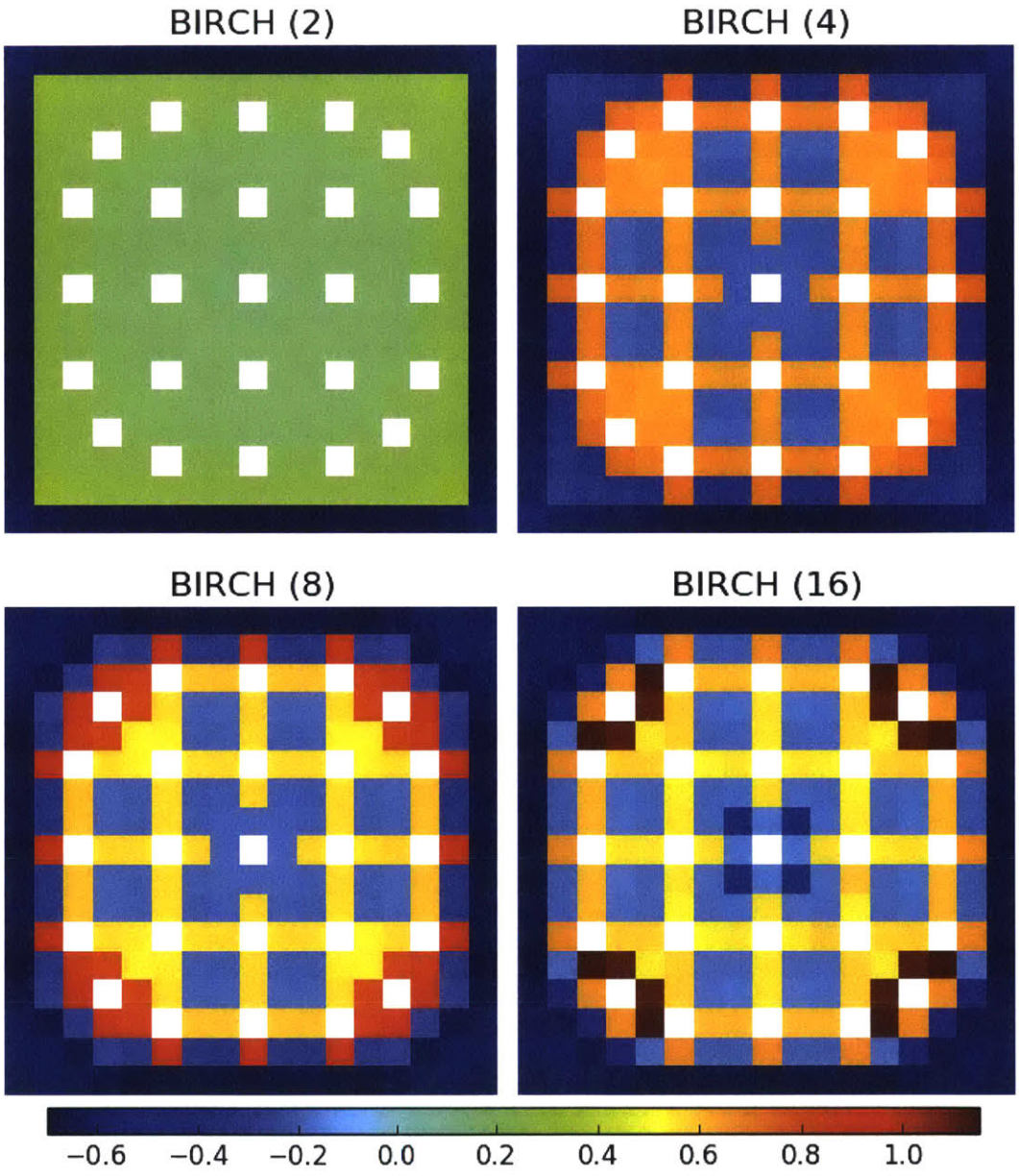


Figure 11-19: The percent relative deviation of U-238 capture rate spatial distributions for *i*MGXS and null homogenization for a 3.1% enriched assembly.

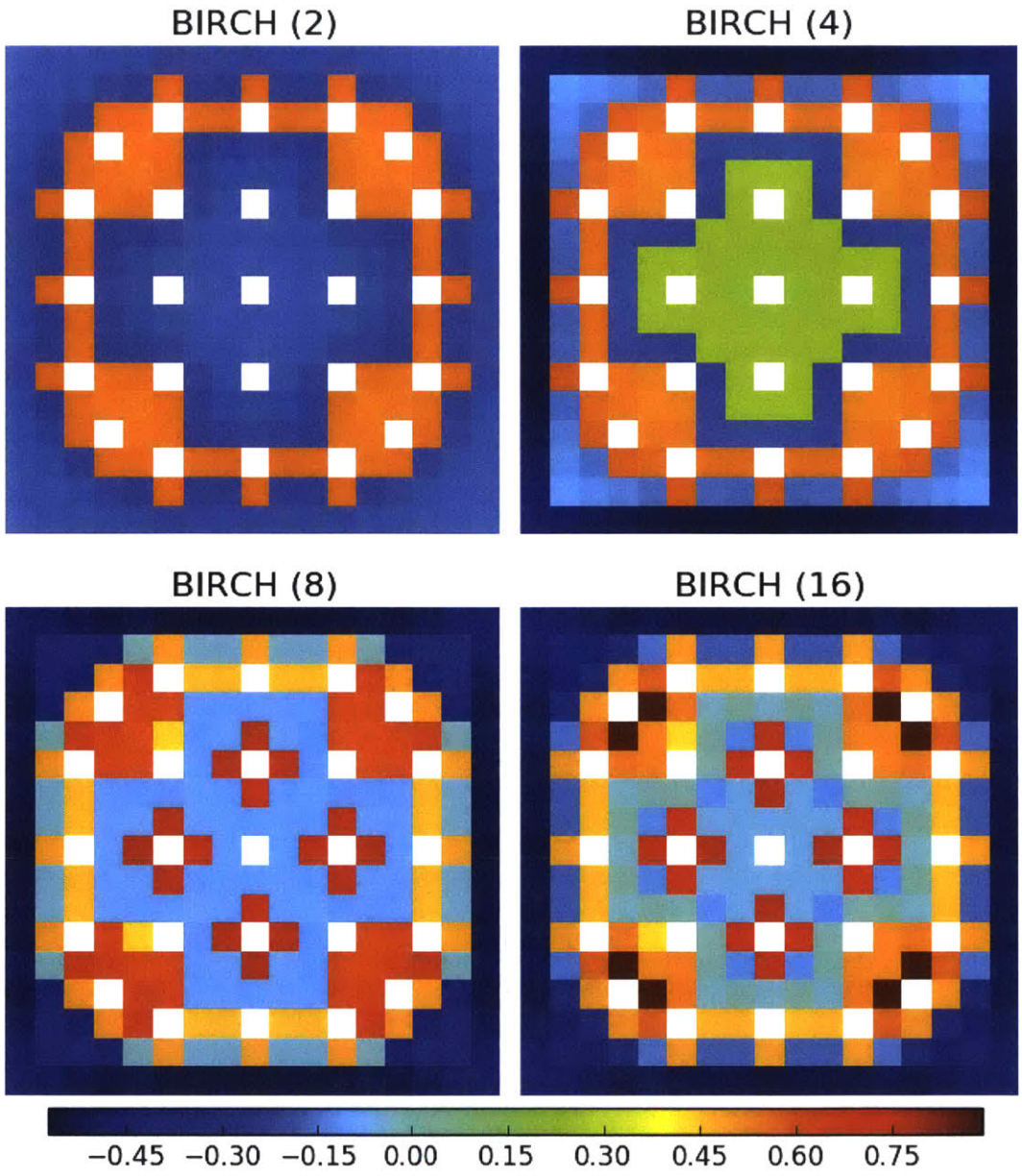


Figure 11-20: The percent relative deviation of U-238 capture rate spatial distributions for *i*MGXS and null homogenization for a 3.1% enriched assembly with 20 BPs.

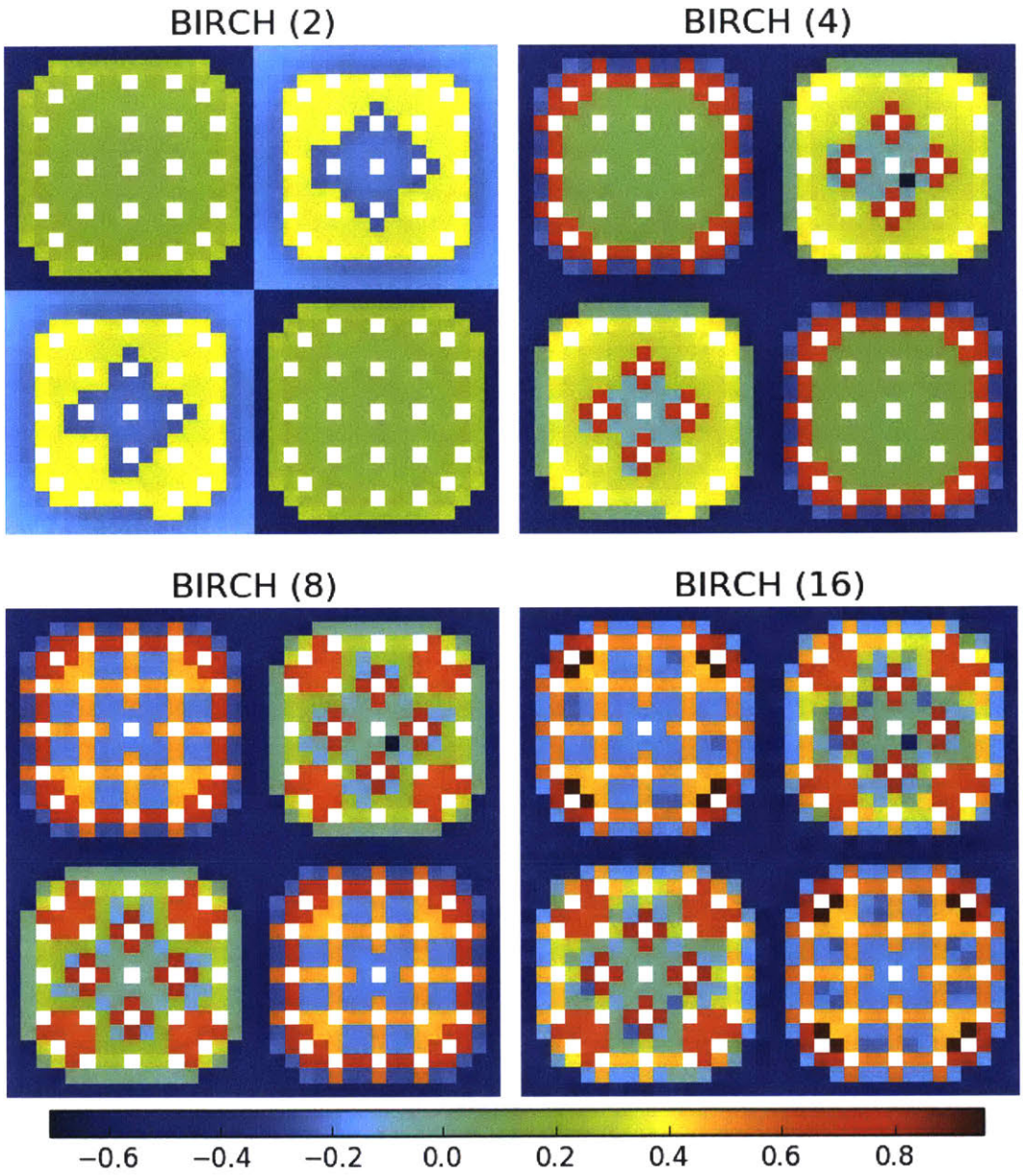


Figure 11-21: The percent relative deviation of U-238 capture rate spatial distributions for *i*MGXS and null homogenization for a 2×2 colorset.

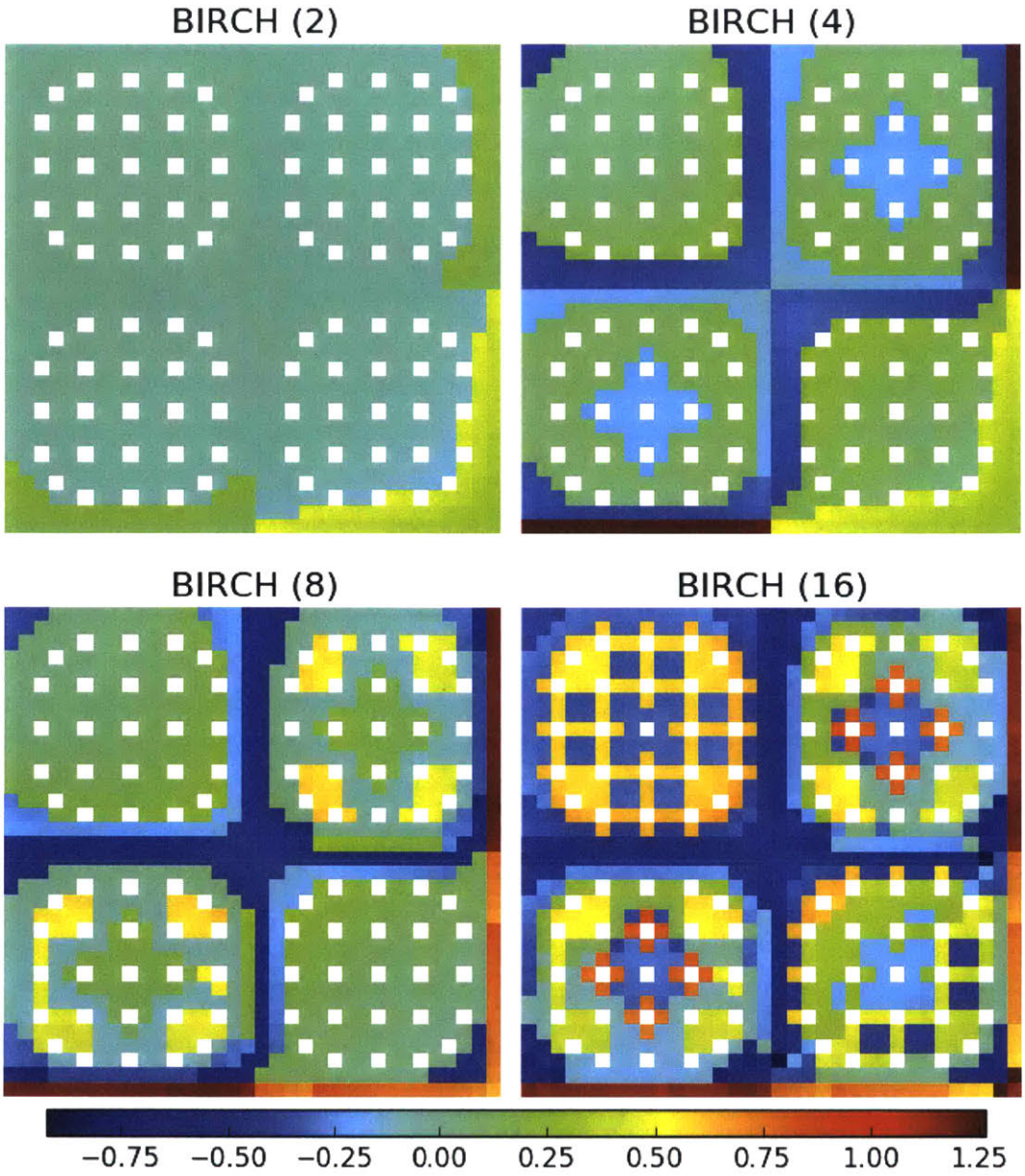


Figure 11-22: The percent relative deviation of U-238 capture rate spatial distributions for *i*MGXS and null homogenization for a 2×2 colorset with a water reflector.

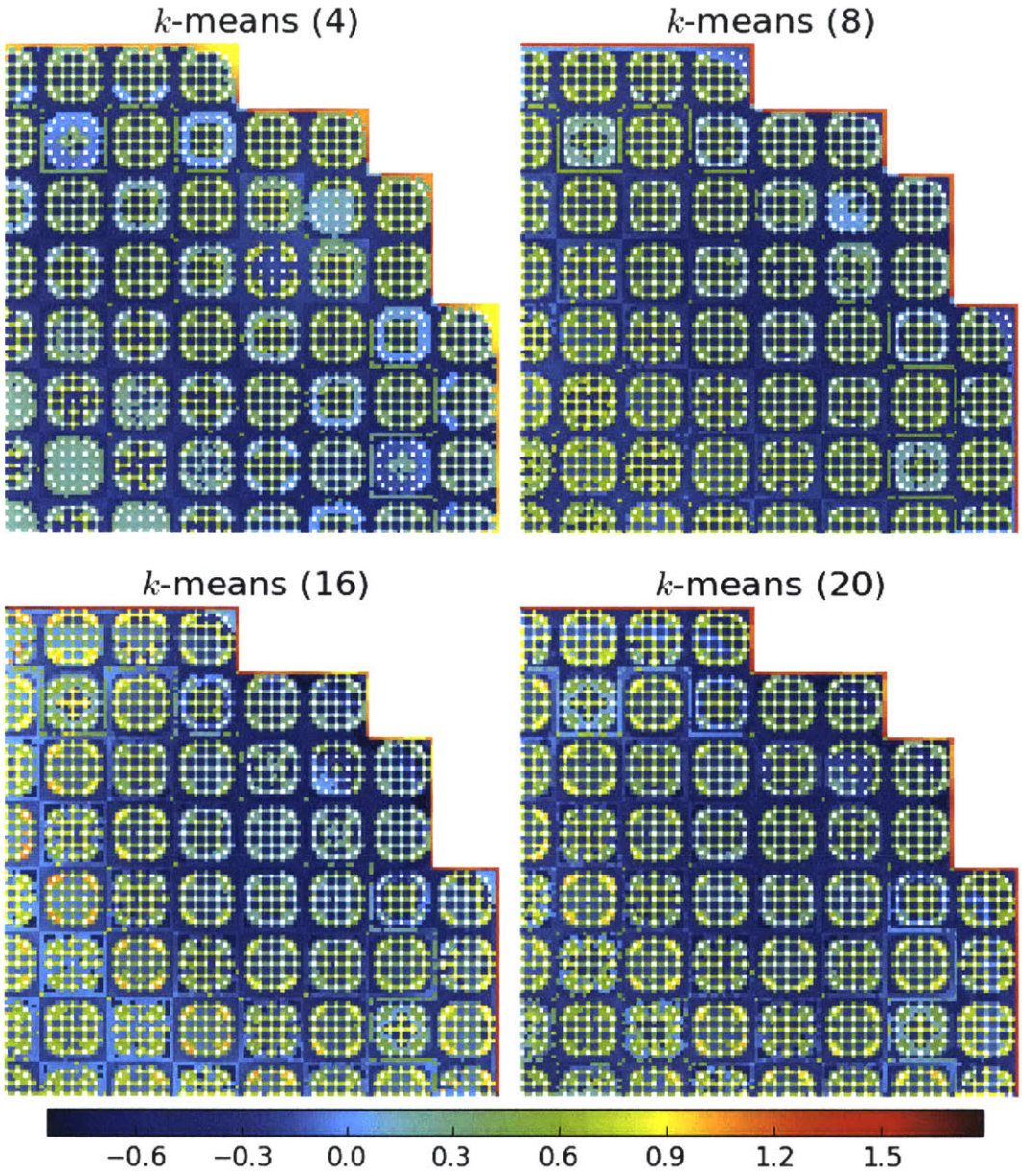


Figure 11-23: The percent relative deviation of U-238 capture rate spatial distributions for *i*MGXS (*with k-means*) and null homogenization for the quarter core BEAVRS model.

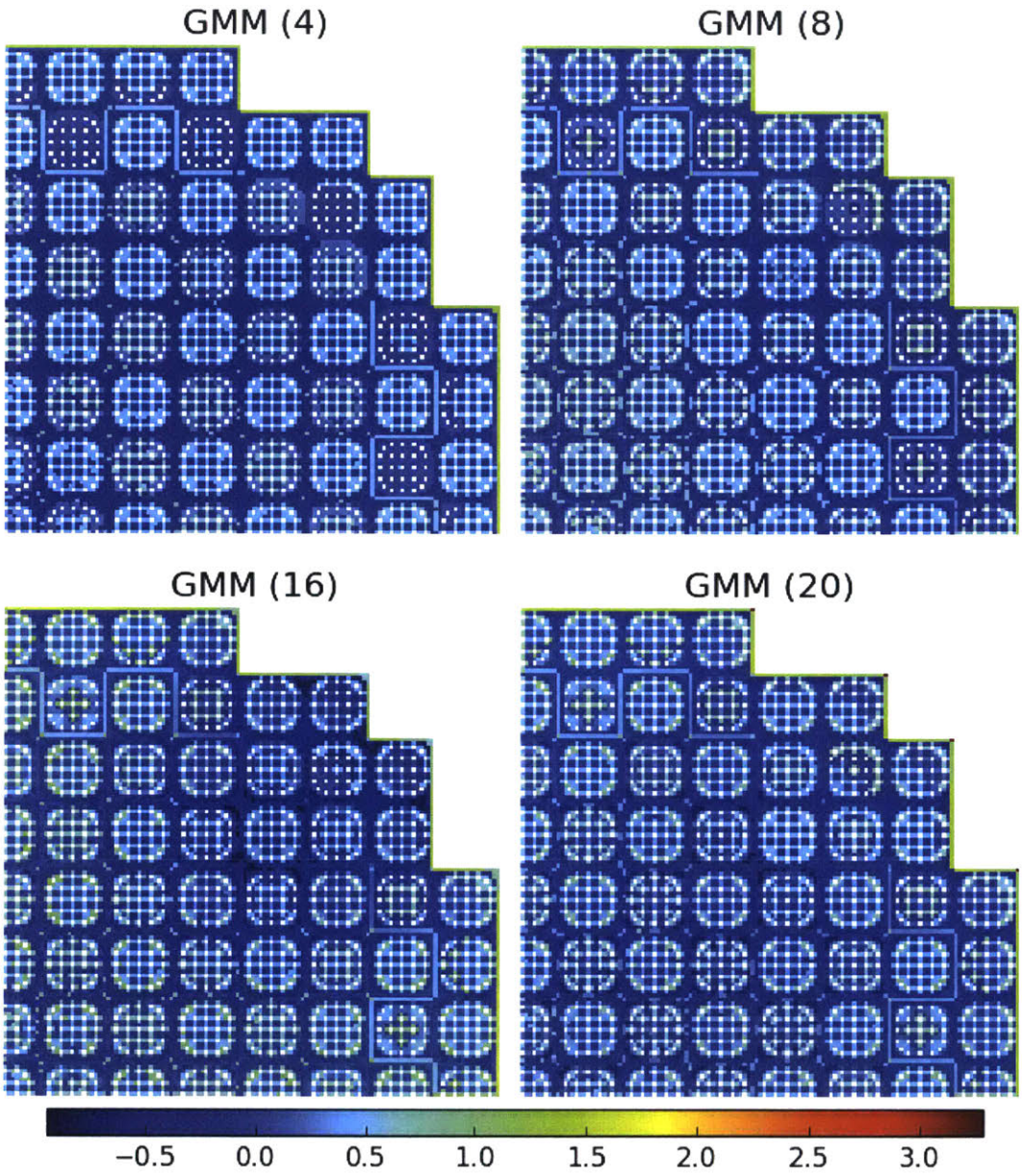


Figure 11-24: The percent relative deviation of U-238 capture rate spatial distributions for *i*MGXS (*with GMM*) and null homogenization for the quarter core BEAVRS model.

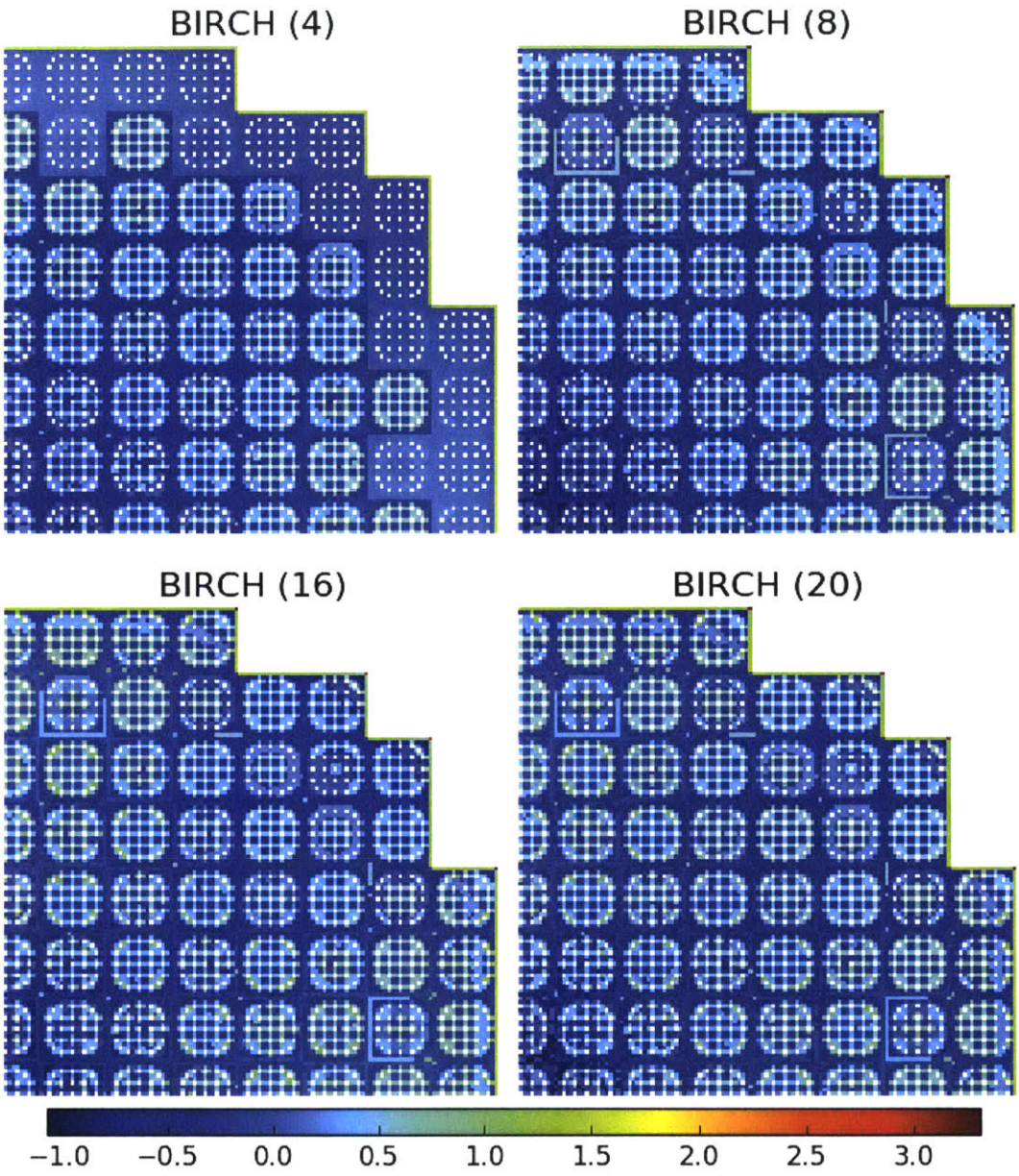


Figure 11-25: The percent relative deviation of U-238 capture rate spatial distributions for *i*MGXS (*with BIRCH*) and null homogenization for the quarter core BEAVRS model.

11.2 Convergence of OpenMOC Solutions

This section demonstrates how quickly multi-group solutions converge with number of MC particle histories simulated to generate MGXS. In particular, this section quantifies the number of histories needed to sufficiently converge MGXS for stable OpenMOC solutions with the null, degenerate and *i*MGXS spatial homogenization schemes for each of the six benchmarks. In addition, the convergence of the OpenMOC solutions are compared to the statistical uncertainties of the corresponding reference OpenMC calculation. This comparison is used to quantify how much faster a solution for a specified accuracy can be achieved with OpenMOC with MGXS generated by OpenMC.

This section uses the MC tally data generated by the same OpenMC simulations which generated MGXS for Sec. 11.1. However, the MGXS in this section are computed from MC tally data stored as OpenMC *statepoint files* for 100 different active batches. For the assembly and colorset benchmarks, 80% of the statepoints were recorded at a logarithmically-spaced interval for the first 1 – 1,000 active batches (10^5 – 10^8 histories), while the remaining 20% were recorded at a logarithmically-spaced interval for the final 1,000 – 9,800 active batches (10^8 – 10^9 histories). The OpenMC statepoints for the earliest batches contained MC tally data with the largest statistical uncertainties, while the tallies for the latter batches had smaller statistical uncertainties. The different statepoints enabled a thorough evaluation of the sensitivity of OpenMOC's solutions to the statistical uncertainties of the MGXS.

The clustering models were trained with the most highly converged MC tally data stored for the final active batch (*i.e.*, the same data used to train the clustering models employed in Sec. 11.1). These clustering models were then re-applied for spatial homogenization of the lesser converged MC tally data from earlier active batches. This methodology permitted a best case assessment of the OpenMOC accuracy one could hope to achieve if the "best" clustering model were accessible (*i.e.*. the clustering model one would find from "fully" converged MC tally data). It is important to note that this methodology is *not* representative of how *i*MGXS would be deployed for MGXS generation in a production setting. Nevertheless, it was a useful exercise to quantify

an upper bound on the acceleration achievable if an optimal *i*MGXS configuration were invented which could infer reliable cluster assignments from unconverged MC tally data.

The convergence rate of the eigenvalues generated with null, degenerate and *i*MGXS homogenization is presented in Sec. 11.2.1. Likewise, the convergence of the U-238 capture rate relative errors is examined in Sec. 11.2.2 for each scheme. The convergence analysis was performed for the single assembly and 2×2 colorset benchmarks, but was not performed for the quarter core BEAVRS model due to computational constraints.

11.2.1 Eigenvalue Convergence

The OpenMOC eigenvalues were compared to the reference OpenMC eigenvalues from Tab. 7.1 for each of the 100 active batches. The eigenvalue bias $\Delta\rho$ was computed from Eqn. 5.1 in units of pcm. The batch-by-batch bias is presented in Figs. 11-26 to 11-30 for the individual assembly and 2×2 colorset benchmarks. In particular, the bias is presented for null, degenerate and *i*MGXS homogenization with BIRCH clustering of 2, 4 and 10 clusters for the assemblies and periodic colorset, and for 2, 4 and 16 clusters for the colorset with a water reflector. The biases are highlighted for $10^5 - 10^9$ active histories for the assemblies and $10^6 - 10^9$ histories for the colorsets, respectively[91].

The biases converge to nearly the same value for all of the benchmarks and spatial homogenization schemes, as expected based on the converged eigenvalue biases tabulated in Sec. 11.1.1. This is consistent with the fact that globally-integrated reaction rates are preserved for all schemes irregardless of the number of clusters modeled. Of particular note is that the bias for the null and *i*MGXS schemes are quite consistent even with only 10^5 and 10^6 particle histories for the assembly and colorset benchmarks, respectively. The bias for the degenerate scheme deviates by up to 100 pcm, but eventually converges to the null and *i*MGXS schemes with enough particle histories. The bias fluctuates on the order of 500 pcm with fewer than 10^7 particle histories, but converges with 10^8 histories for all of the benchmarks and schemes.

A number of different factors may cause the eigenvalue to fluctuate and deviate

[91] Several hundred thousand histories were required to sufficiently converge the MGXS for the colorsets to compute stable solutions with OpenMOC.

between schemes for few particle histories. First, the statistical uncertainties of the MGXS must be minimized in order to converge the OpenMOC eigenvalues. In addition, it should be recalled from Sec. 3.2 that the MGXS are generated from OpenMC using a mixture of track-length and analog tally estimators. In particular, the total, fission and capture MGXS are computed with track-length estimators, while the scattering matrices and fission spectra are computed from analog estimators since they depend on the neutron's outgoing energy at each interaction. Although both track-length and analog estimators will converge to the same expected value, analog estimators will converge more slowly. As a result, neutron balance will not be preserved for MGXS computed from a mixture of track-length and analog estimators without a sufficient number of simulated histories. This issue was the a key motivating factor for Nelson's Nuclear Data PreProcessor (NDPP) [37] to enable track-length estimators for multi-group scattering matrices and fission spectra. Methods such as NDPP may be a fruitful area of future research to enable faster converging eigenvalues from MGXS generated with MC.

The null, degenerate and *i*MGXS schemes converge to the same eigenvalue bias with 10^8 particle histories for the assembly and colorset benchmarks.

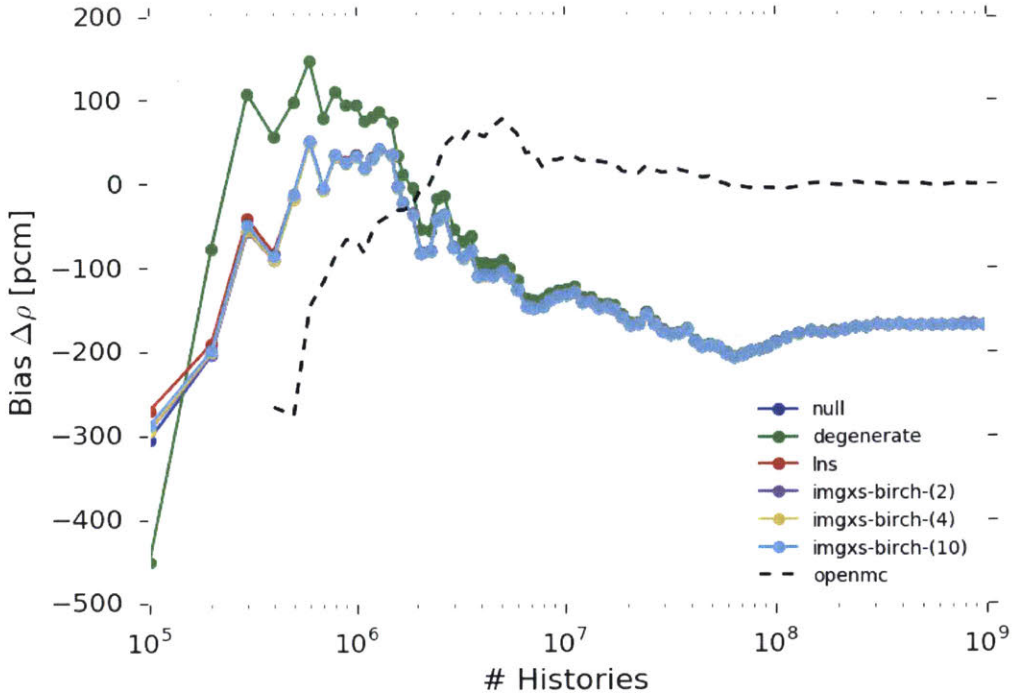


Figure 11-26: Convergence of the eigenvalue bias for the 1.6% enriched assembly.

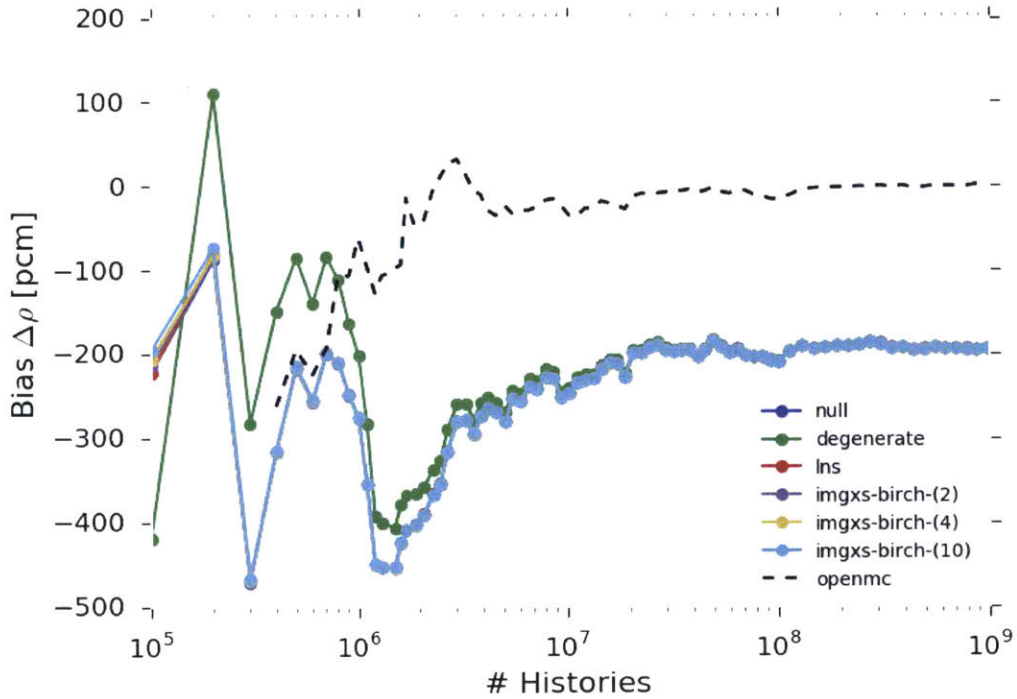


Figure 11-27: Convergence of the eigenvalue bias for the 3.1% enriched assembly.

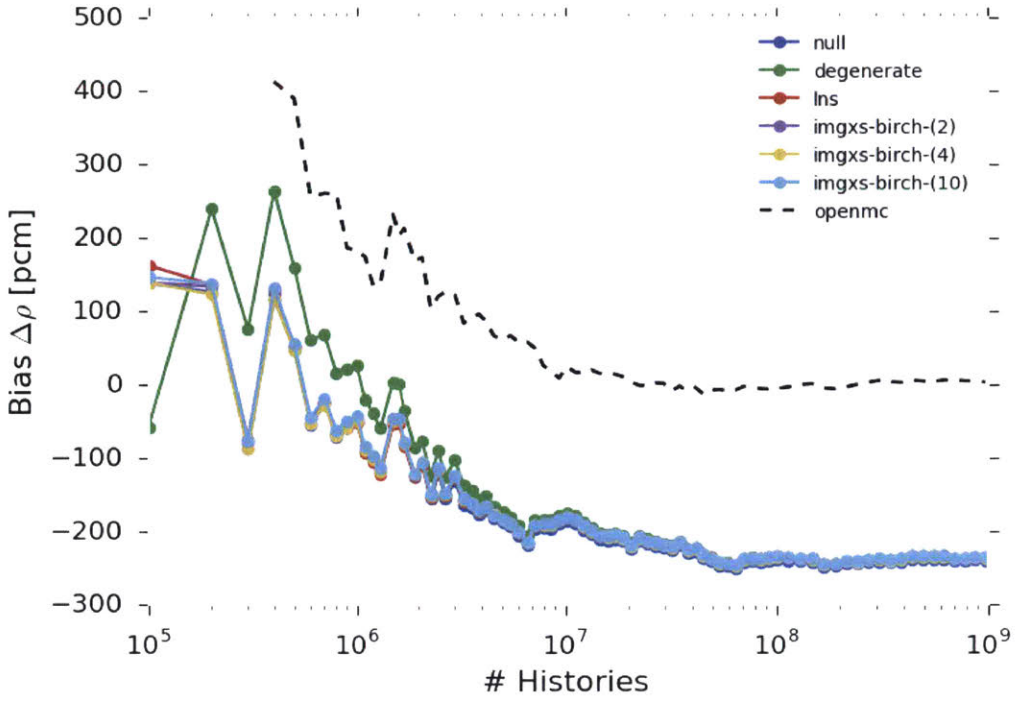


Figure 11-28: Convergence of the eigenvalue bias for the 3.1% enriched assembly with 20 BPs.

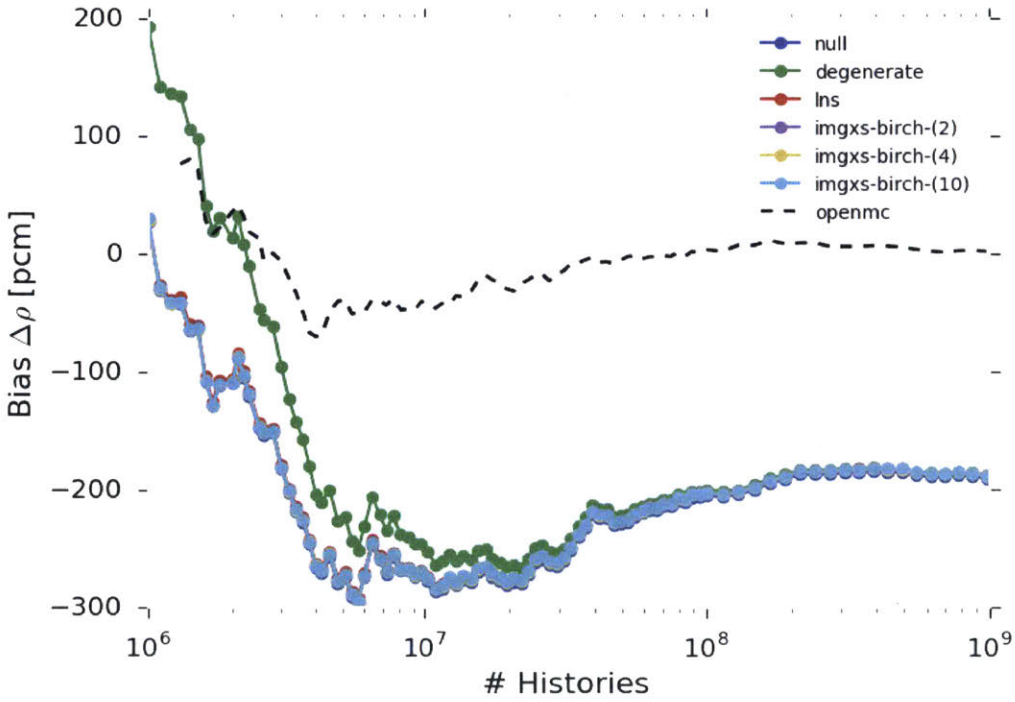


Figure 11-29: Convergence of the eigenvalue bias for the periodic colorset.

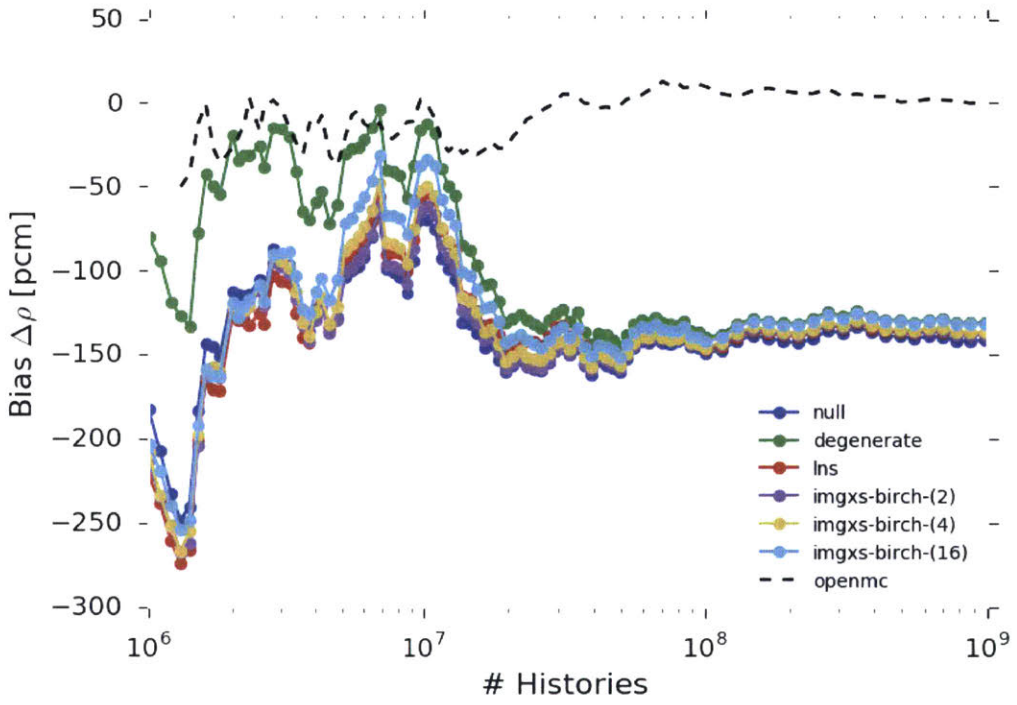


Figure 11-30: Convergence of the eigenvalue bias for the colorset with a water reflector.

11.2.2 U-238 Capture Rate Convergence

The OpenMOC energy-integrated pin-wise U-238 capture rates were compared to the reference OpenMC capture rates to compute the percent relative errors for each pin's capture rates for each of the 100 active batches. The batch-by-batch maximum and mean relative errors are presented in Figs. 11-31 to 11-35 for the individual assembly and 2×2 colorset benchmarks. In particular, the errors are presented for null, degenerate and *i*MGXS homogenization with BIRCH clustering of 2, 4 and 10 clusters for the assemblies and periodic colorset, and for 2, 4 and 16 clusters for the colorset with a water reflector. In addition, the maximum and mean statistical uncertainty (1-sigma) of the OpenMC estimate for the pin-wise U-238 capture rates is also highlighted in each figure[92]. The OpenMOC errors and OpenMC uncertainties are highlighted for $10^5 - 10^9$ active histories for the assemblies and $10^6 - 10^9$ histories for the colorsets, respectively[91].

A couple of key conclusions can be drawn from these results. First, the OpenMC statistical uncertainties converge at a rate similar to the OpenMOC errors for degenerate homogenization. The maximum OpenMC uncertainty is always less than the maximum degenerate error. However, the mean uncertainty is greater than the mean degenerate error for fewer than approximately 500,000,000 histories. With a sufficient number of histories, the uncertainties fall below the errors. The OpenMC uncertainties approach zero while the OpenMOC errors approach a non-zero bias dependent on the approximation error separating the OpenMOC and reference OpenMC solutions. The final converged errors illustrate the same behavior as discussed in Sec. 11.1.2. In particular, the converged *i*MGXS errors increasingly approach the degenerate errors as more clusters are introduced. While the degenerate scheme always outperforms *i*MGXS, the improvement is relatively marginal compared to the improvement that *i*MGXS achieves with respect to the null scheme.

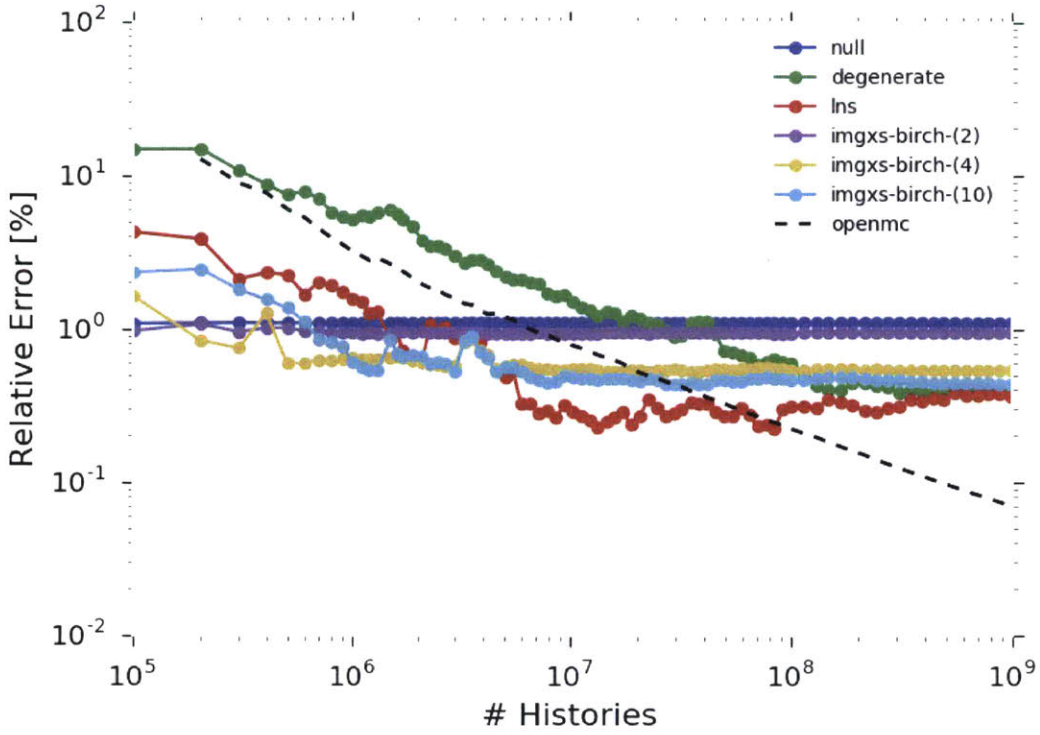
Both the maximum and mean errors for the null homogenization scheme are converged with only 10^5 and 10^6 histories for the assembly and colorset benchmarks,

[92] Although the OpenMOC relative error and OpenMC statistical uncertainty are not equivalent, it is useful to compare their convergence rates as a proxy for the relative runtimes of directly computing a reference solution with OpenMC, versus using OpenMC to generate sufficiently converged MGXS for OpenMOC.

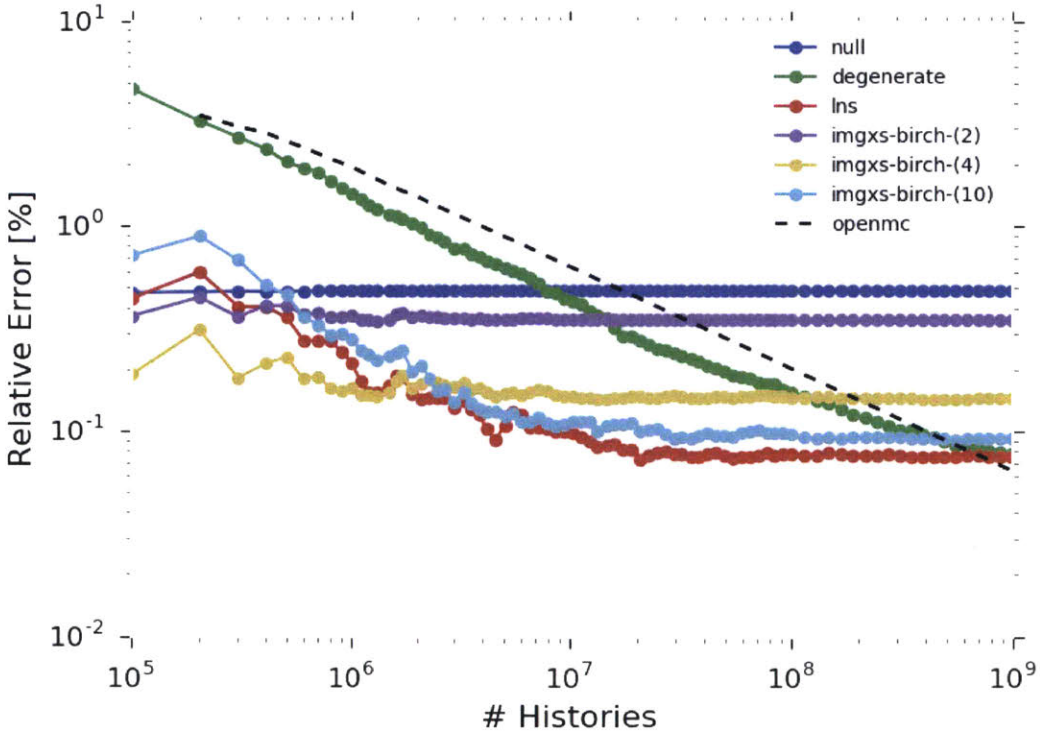
respectively. In contrast, the errors for the degenerate scheme are not fully converged even with 10^9 histories. This indicates that at least some fraction of the OpenMOC relative errors with the degenerate scheme is due to the lingering statistical uncertainties of the MGXS. The degenerate error would only converge if more particle histories were simulated to compute the MGXS. The convergence of the *i*MGXS scheme lies between that of the null and degenerate schemes, and requires more particle histories to converge as more clusters are identified. The convergence behavior is increasingly erratic with more histories, as is illustrated by the maximum error of the *i*MGXS scheme with 16 clusters for the colorset with water reflector in Fig. 11-35a.

The figures illustrate the potential for *i*MGXS to achieve nearly the same accuracy as the degenerate scheme with a factor of 10 or fewer particle histories. Furthermore, the results indicate that OpenMOC's errors converge at least 10× faster than OpenMC's statistical uncertainties converge to the same value.

The OpenMOC U-238 capture rate errors with null homogenization converge with fewer than 10^6 histories, while the errors with degenerate homogenization are not converged even with 10^9 histories. The convergence rate of the errors decreases with the number of clusters used in *i*MGXS spatial homogenization. The *i*MGXS scheme requires 10× fewer particle histories to converge than both the degenerate scheme and the OpenMC reference solution.

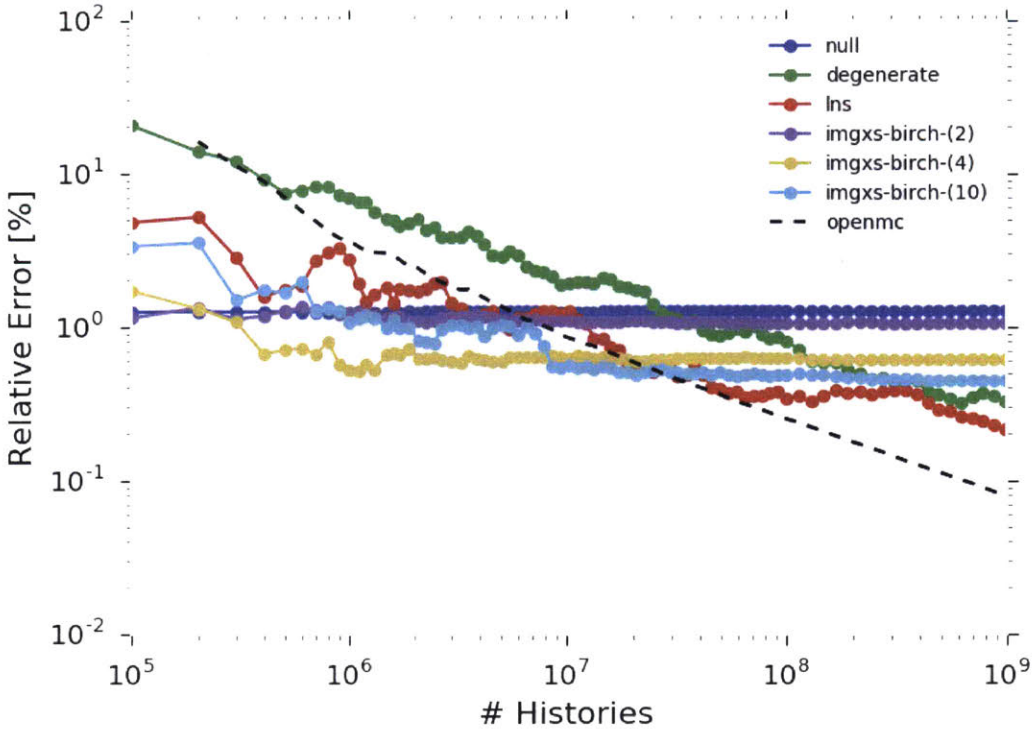


(a)

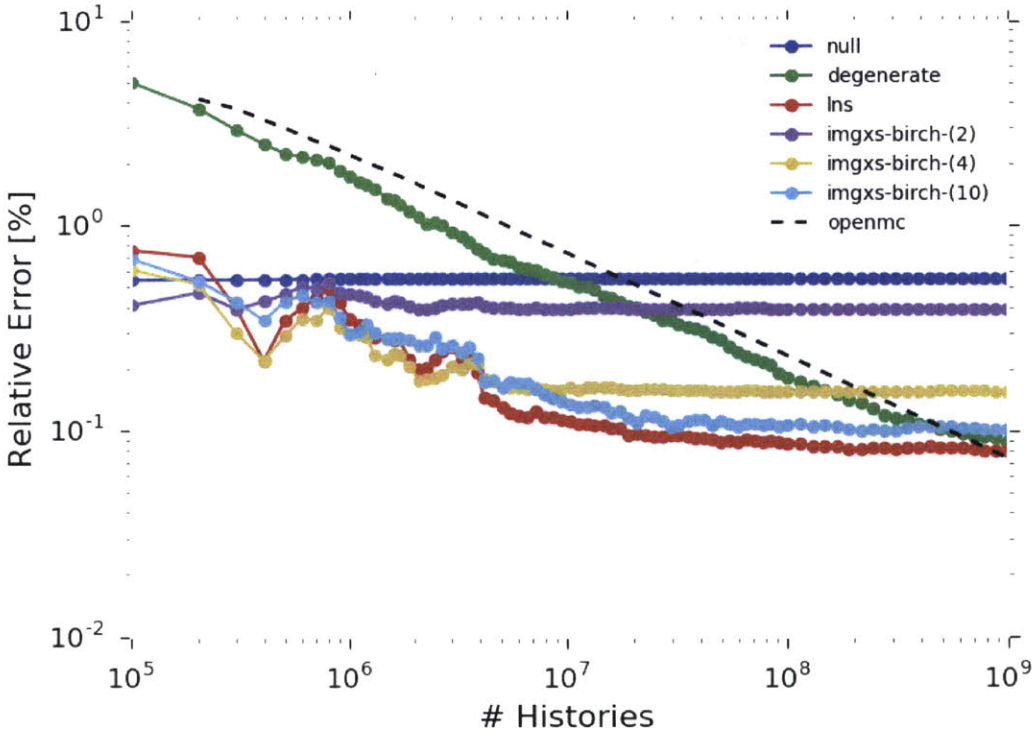


(b)

Figure 11-31: Convergence of the max (a) and mean (b) absolute U-238 capture rate percent relative errors for the 1.6% enriched assembly.

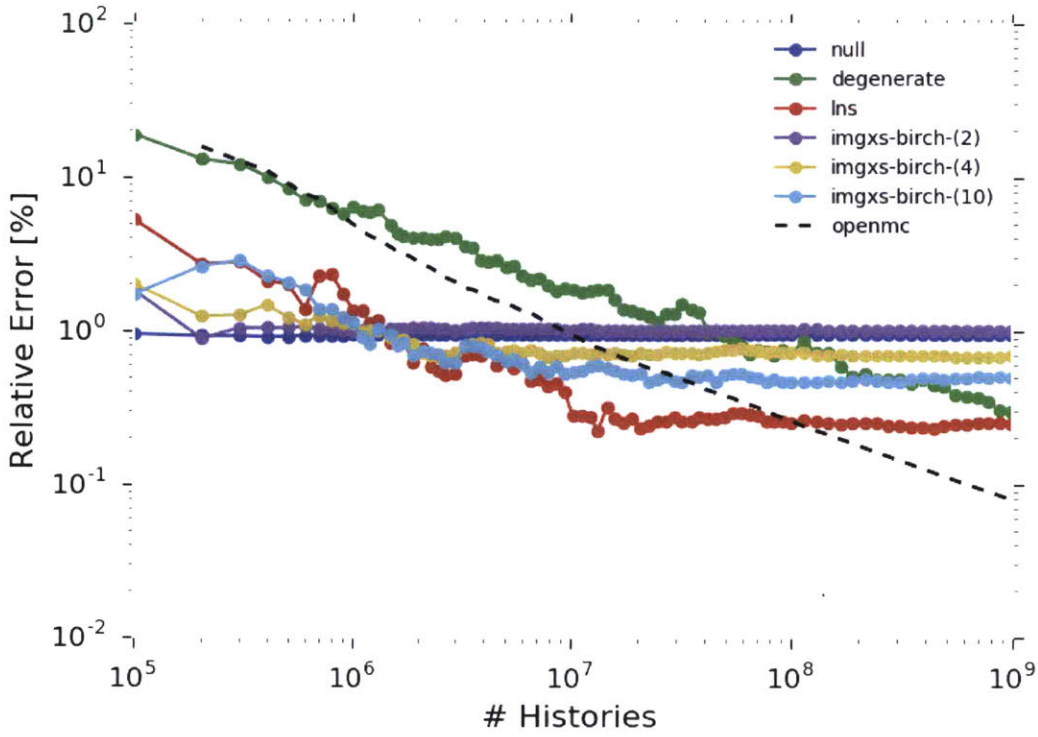


(a)

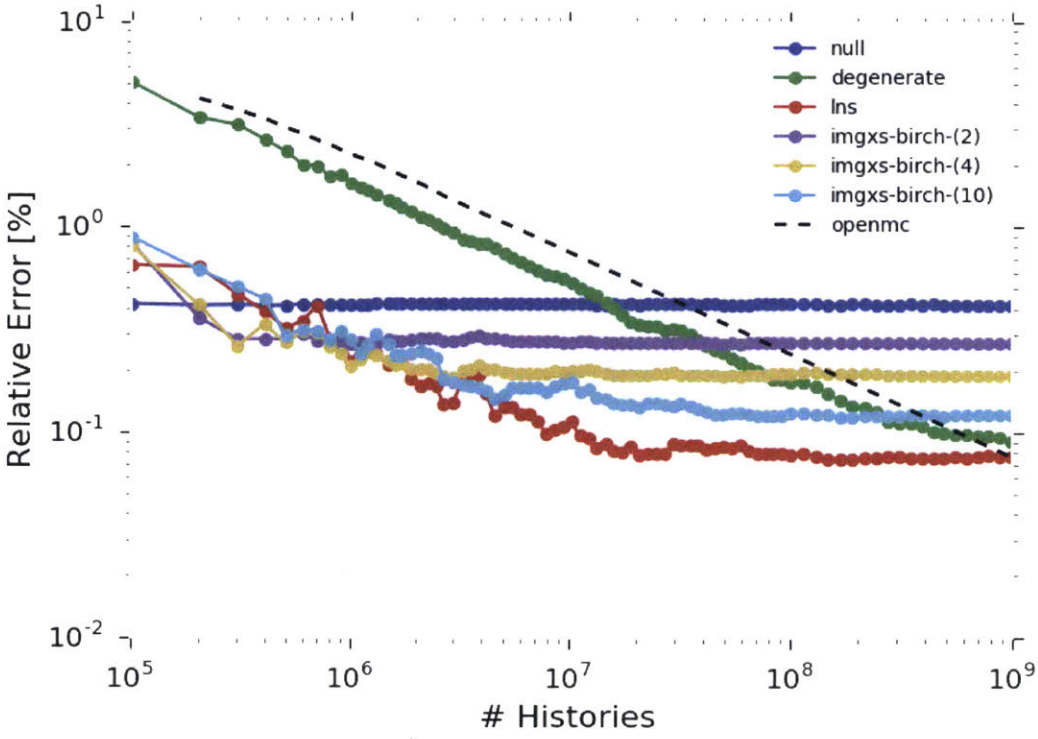


(b)

Figure 11-32: Convergence of the max (a) and mean (b) absolute U-238 capture rate percent relative errors for the 3.1% enriched assembly.

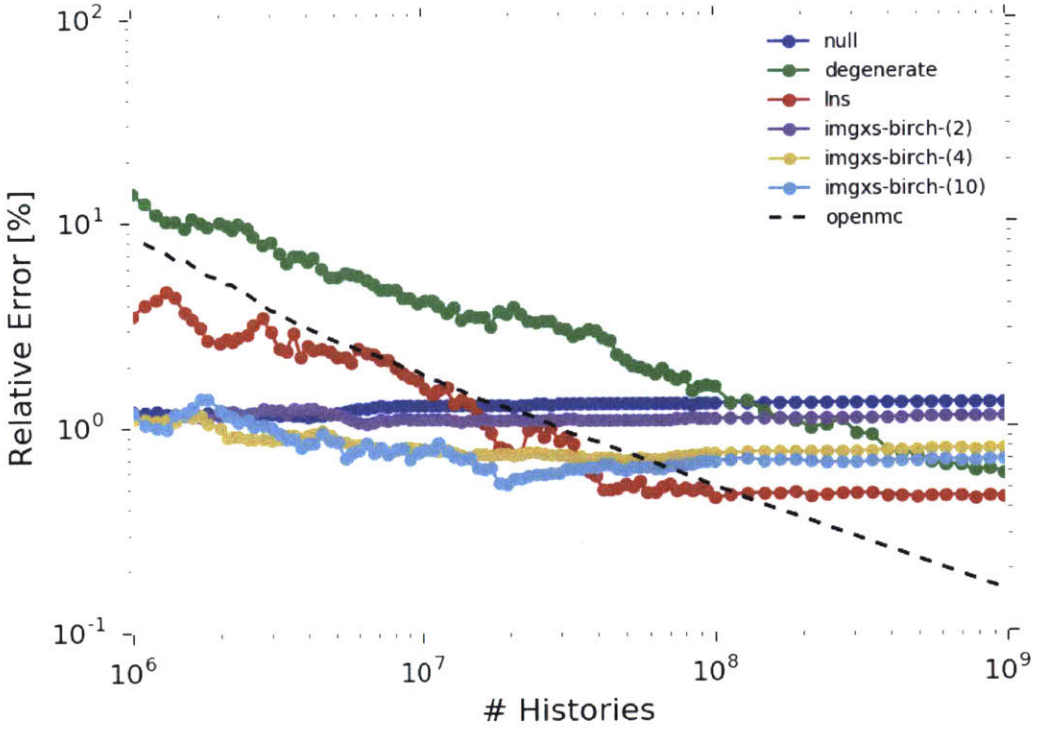


(a)

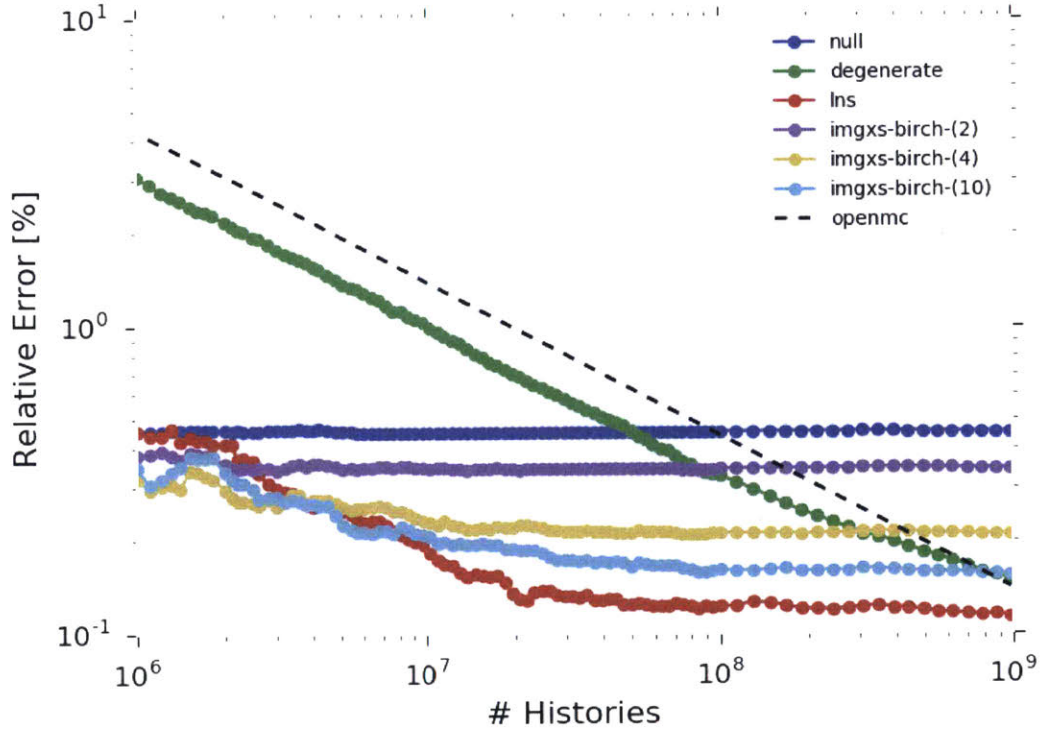


(b)

Figure 11-33: Convergence of the max (a) and mean (b) absolute U-238 capture rate percent relative errors for the 3.1% enriched assembly with 20 BPs.

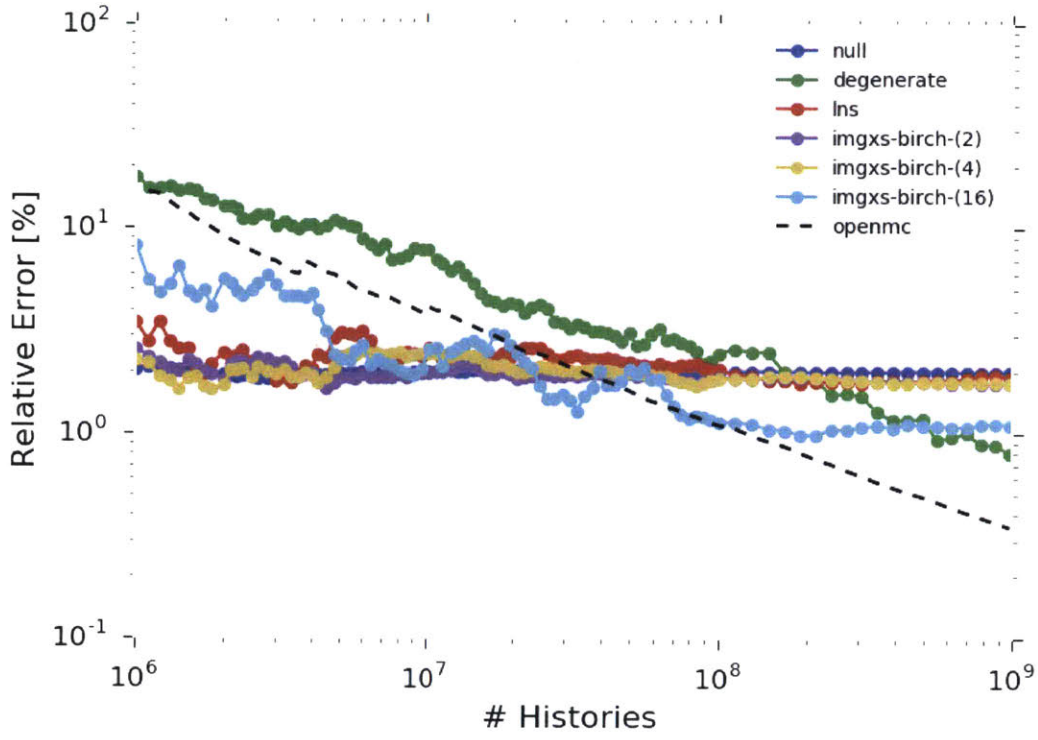


(a)

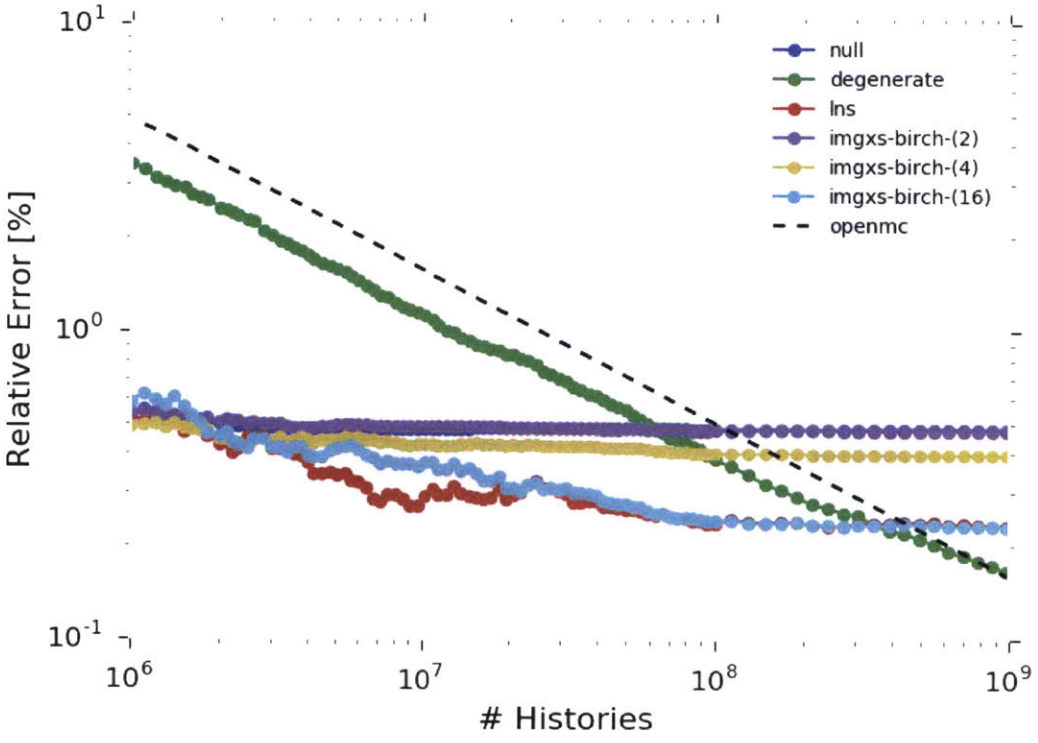


(b)

Figure 11-34: Convergence of the max (a) and mean (b) absolute U-238 capture rate percent relative errors for the periodic colorset.



(a)



(b)

Figure 11-35: Convergence of the max (a) and mean (b) absolute U-238 capture rate percent relative errors for the colorset with a water reflector.

11.3 Evaluation of Model Selection Techniques

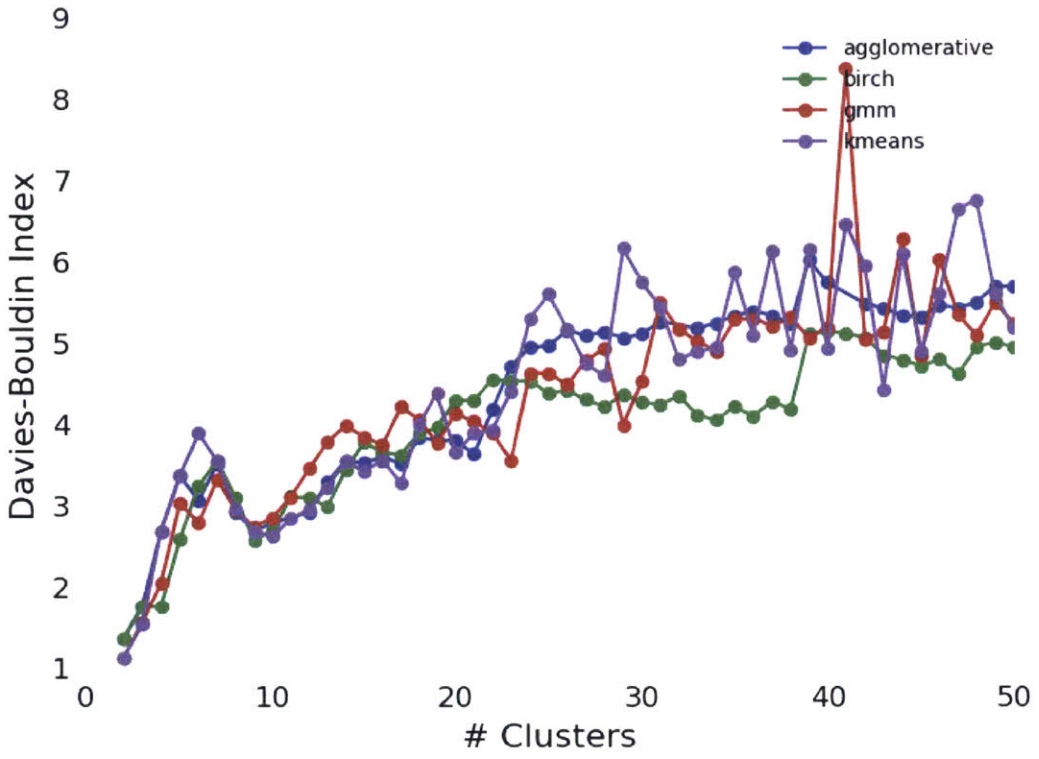
This section analyzes the model selection criteria introduced in Sec. 10.6. The various criteria are computed for the clustering models trained in Sec. 11.1.2.1. The model selection criteria are designed to quantitatively compare clustering models, and in particular, to help choose the optimal number of clusters. In a truly unsupervised (*i.e.*, production) setting, the OpenMOC eigenvalue bias and reaction rate errors with respect to a reference OpenMC solution would not be available to help choose the optimal number of clusters. Instead, the "best" clustering model would be chosen strictly from the model selection criteria.

Each criterion was computed for the clustering models built for the 1.6% enriched fuel assembly and 2×2 colorset with water reflector. The Davies-Bouldin, Dunn and Calinski-Harabaz indices are presented in Secs. 11.3.1 to 11.3.3, respectively. The silhouette coefficients are presented in Sec. 11.3.4. Finally, Sec. 11.3.5 highlights the Bayesian Information Criterion (BIC) for the GMM clustering algorithm. Cross-validation (Sec. 10.6.6) was not employed to compute the heuristics, but could be used in the future to compute more robust estimates for each criterion.

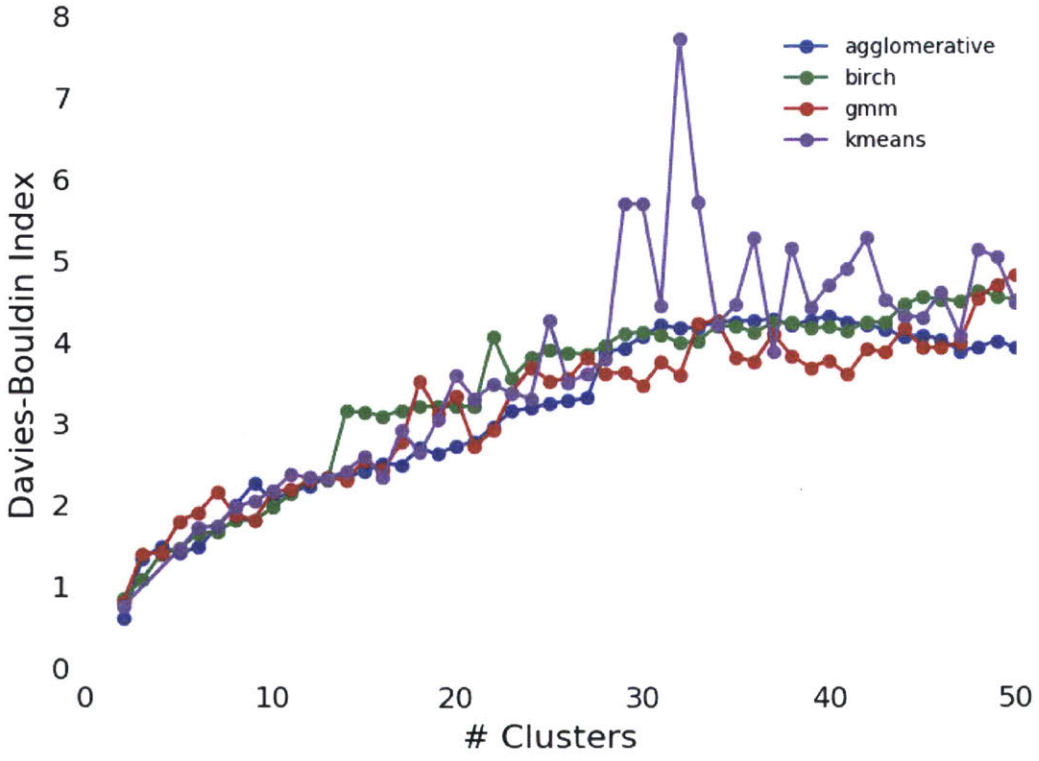
11.3.1 Davies-Bouldin Index

The Davies-Bouldin (DB) index is the ratio of within-cluster scatter to between-cluster separation (Sec. 10.6.1), and is presented in Fig. 11-36. In theory, the "best" clustering model should minimize the DB index. The figures illustrate that the smallest DB indices correspond to the clustering models with two clusters[93]. As more clusters are added to each model, the within-cluster scatter should decrease, while the between-cluster separation should increase. The increasing DB indices indicate that the within-cluster scatter decreases more slowly than the between-cluster separation increases for both benchmarks. It is unclear why the DB index behaves in this way. Furthermore, it is not clear that the DB index would be a useful metric for determining the optimal number of clusters since it would only choose two clusters for both benchmarks.

[93]The Davies-Bouldin index is undefined for a single cluster.



(a)



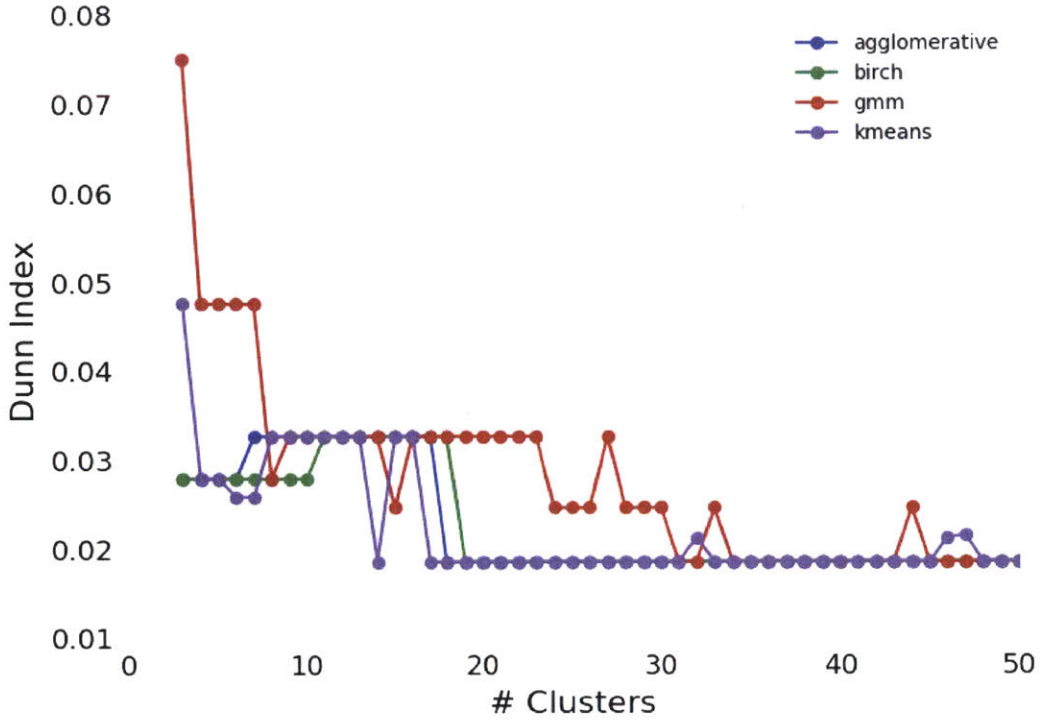
(b)

Figure 11-36: Davies-Bouldin indices for the 1.6% enriched assembly (a) and 2×2 colorset with a water reflector (b) for each of the four clustering algorithms.

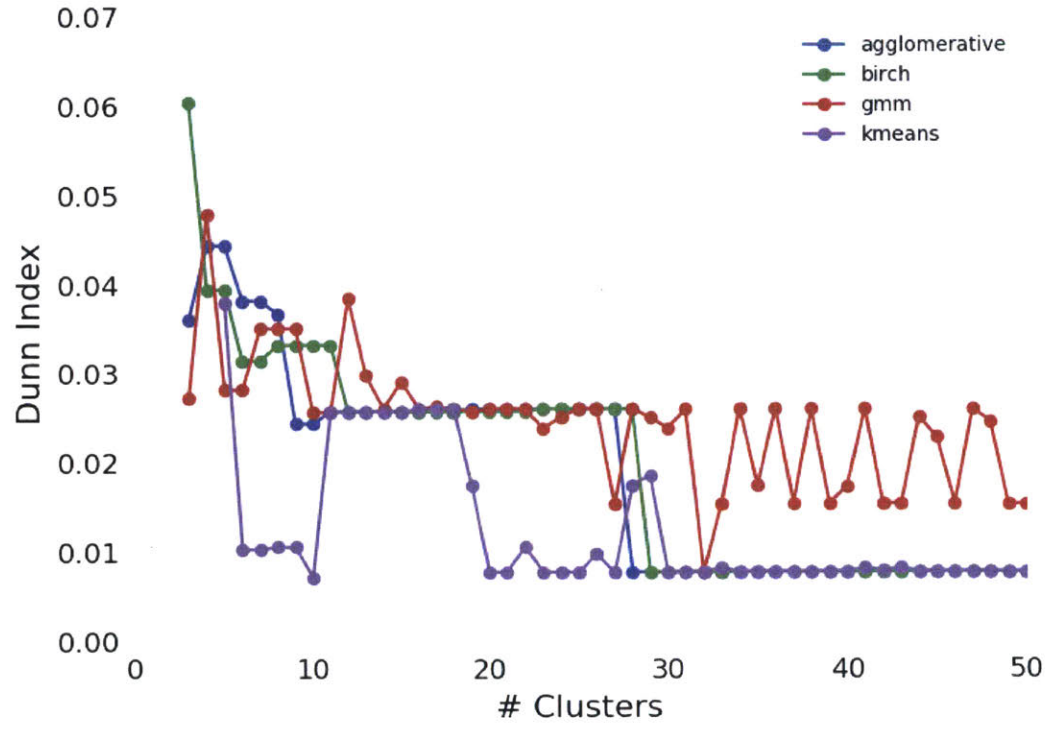
11.3.2 Dunn Index

The Dunn index is the ratio of the minimum inter-cluster distance to the maximum intra-cluster distance (Sec. 10.6.2), and is presented in Fig. 11-37. The Dunn indices are only highlighted for 3 – 50 clusters since the index was over five orders of magnitude larger for only two clusters, which would not be useful to plot alongside the indices for more clusters[94]. In theory, the "best" clustering model should maximize the Dunn index. The clustering models with only two clusters maximize the Dunn index. As more clusters are added to each model, both the minimum inter-cluster and maximum intra-cluster distances should decrease. The decreasing Dunn indices indicate that the minimum inter-cluster distance decreases more slowly than the maximum intra-cluster distance for both benchmarks. It is unclear why the Dunn index behaves in this way. Furthermore, it is not clear that the Dunn index would be a useful metric for determining the optimal number of clusters since it would only choose two clusters for both benchmarks. Nevertheless, it should be recalled that the Dunn index is designed to characterize the most poorly defined cluster(s) in the model. The Dunn index varies in a step-like fashion in the figures, indicating that the most poorly defined cluster(s) are not typically refined with each successive cluster added to the models.

[94]The Dunn index is undefined for a single cluster.



(a)



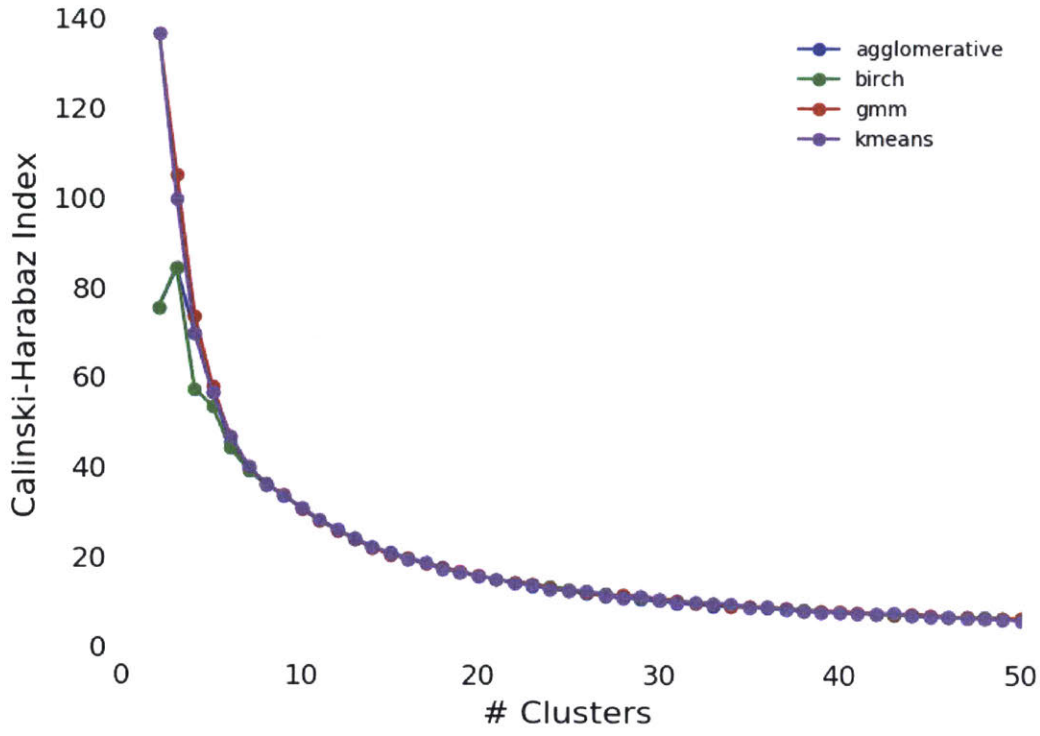
(b)

Figure 11-37: Dunn indices for the 1.6% enriched assembly (a) and 2×2 colorset with a water reflector (b) for each of the four clustering algorithms.

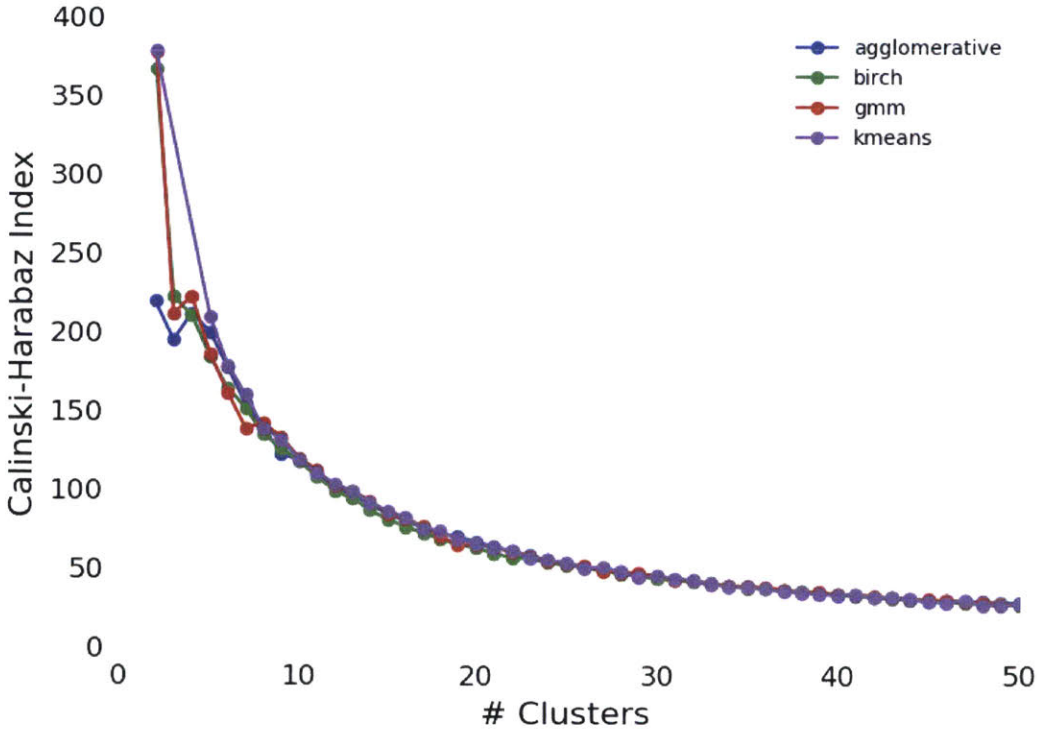
11.3.3 Calinski-Harabaz Index

The Calinski-Harabaz (CH) index is the ratio of between-cluster and within-cluster dispersion (Sec. 10.6.3) and is presented in Fig. 11-38. In theory, the "best" clustering model should maximize the CH index. The figures illustrate that the largest CH indices correspond to the clustering models with two clusters[95]. As more clusters are added to each model, the between-cluster dispersion should increase, while the within-cluster dispersion should decrease. The decreasing CH indices indicate that the between-cluster dispersion increases more slowly than the within-cluster dispersion decreases for both benchmarks. It is unclear why the CH index behaves in this way. Furthermore, it is not clear that the CH index would be a useful metric for determining the optimal number of clusters since it would only choose two clusters for both benchmarks. Nevertheless, unlike the DB and Dunn indices, the CH indices exhibit smoothly varying behavior with the number of clusters. In addition, the CH indices are very nearly the same for all four clustering algorithms. Hence, the CH index might be useful if future work designs features such that the index exhibited a maxima representing the optimal number of clusters.

[95]The Calinski-Harabaz index is undefined for a single cluster.



(a)



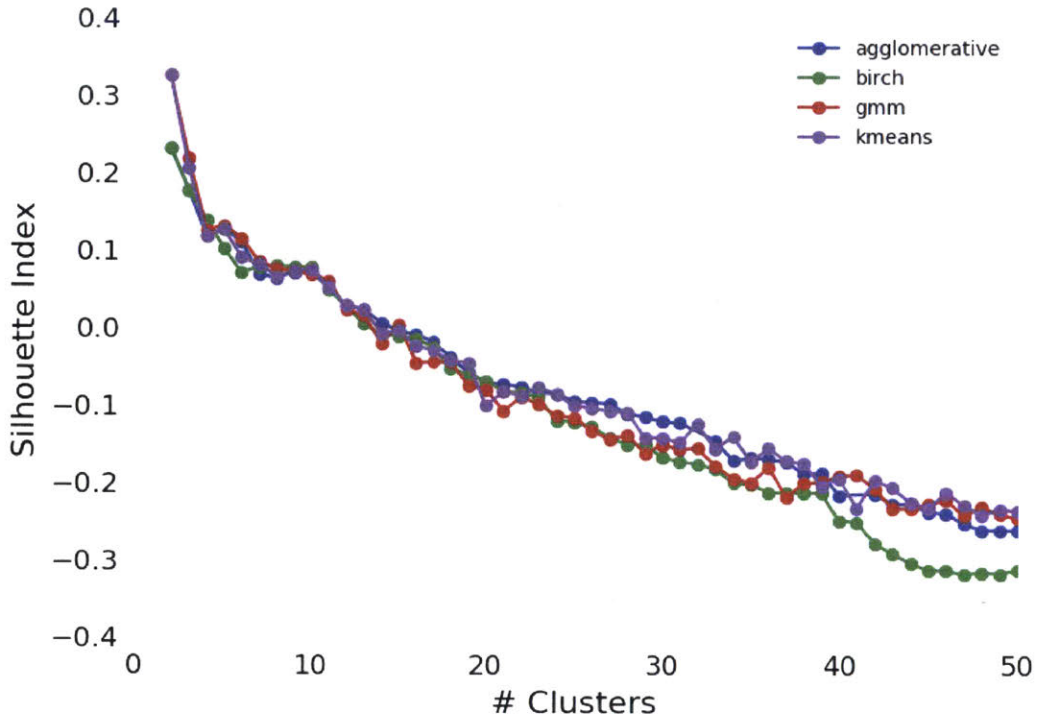
(b)

Figure 11-38: Calinski-Harabaz indices for the 1.6% enriched assembly (a) and 2×2 colorset with a water reflector (b) for each of the four clustering algorithms.

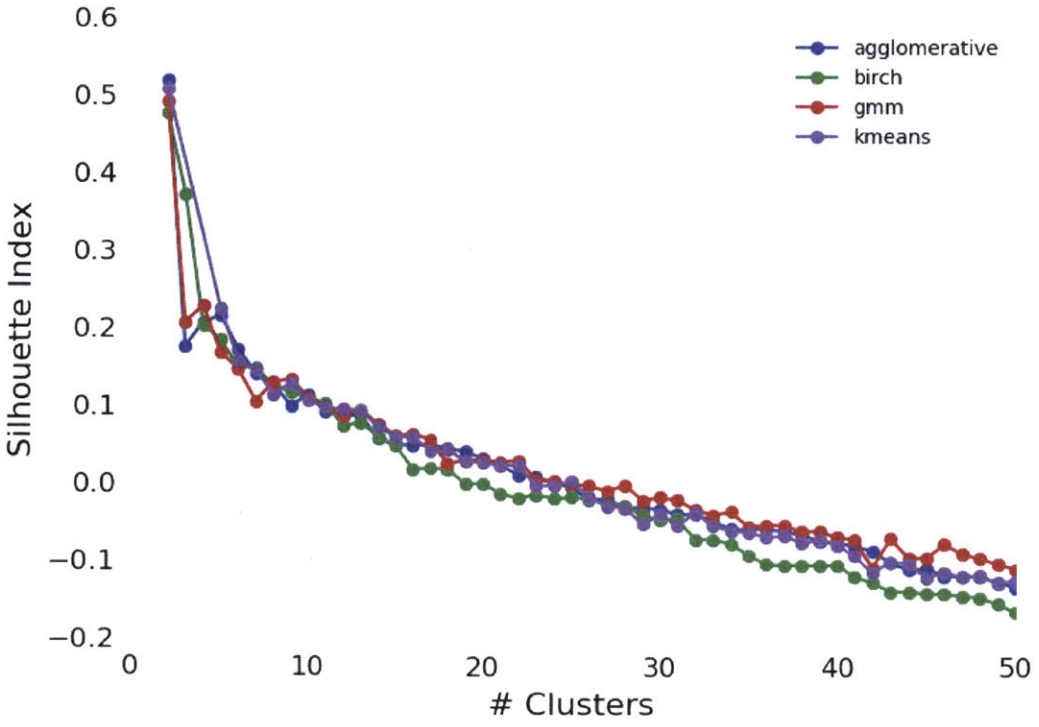
11.3.4 Silhouette Coefficient

The silhouette coefficient compares the average distance between a sample and all other samples within its assigned cluster to the average distance to samples in all other clusters (Sec. 10.6.4). The average silhouette coefficients are presented in Fig. 11-39. In theory, the "best" clustering model should maximize the silhouette coefficient. The figures illustrate that the largest silhouette coefficients correspond to the clustering models with only two clusters[96]. As more clusters are added to each model, the average distance a_k between a sample k and all other samples within its assigned cluster should decrease, while the average distance b_k between the sample and the samples in other clusters should increase. The monotonically decreasing silhouette coefficients indicate that in general, a_k decreases more quickly than b_k for both benchmarks. It is unclear why the silhouette coefficients behave in this way. Furthermore, it is not clear that the silhouette coefficient would be a useful metric for determining the optimal number of clusters since it would only choose two clusters for both benchmarks. Nevertheless, the silhouette coefficient exhibits more smoothly varying behavior with the number of clusters than the DB and Dunn indices, though it is less smooth than the CH indices. In addition, the silhouette coefficients are very nearly the same for all four clustering algorithms. Hence, the silhouette coefficient might be useful if future work designs features such that the coefficient exhibited a maxima representing the optimal number of clusters.

[96]The silhouette coefficient is undefined for a single cluster.



(a)



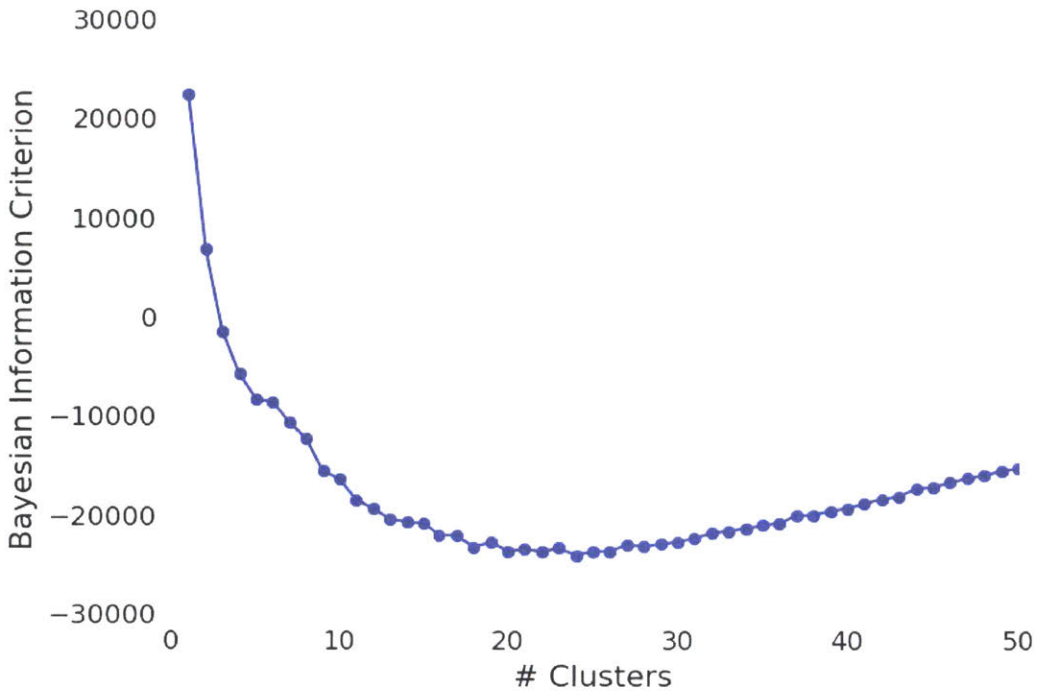
(b)

Figure 11-39: Silhouette coefficients for the 1.6% enriched assembly (a) and 2×2 colorset with a water reflector (b) for each of the four clustering algorithms.

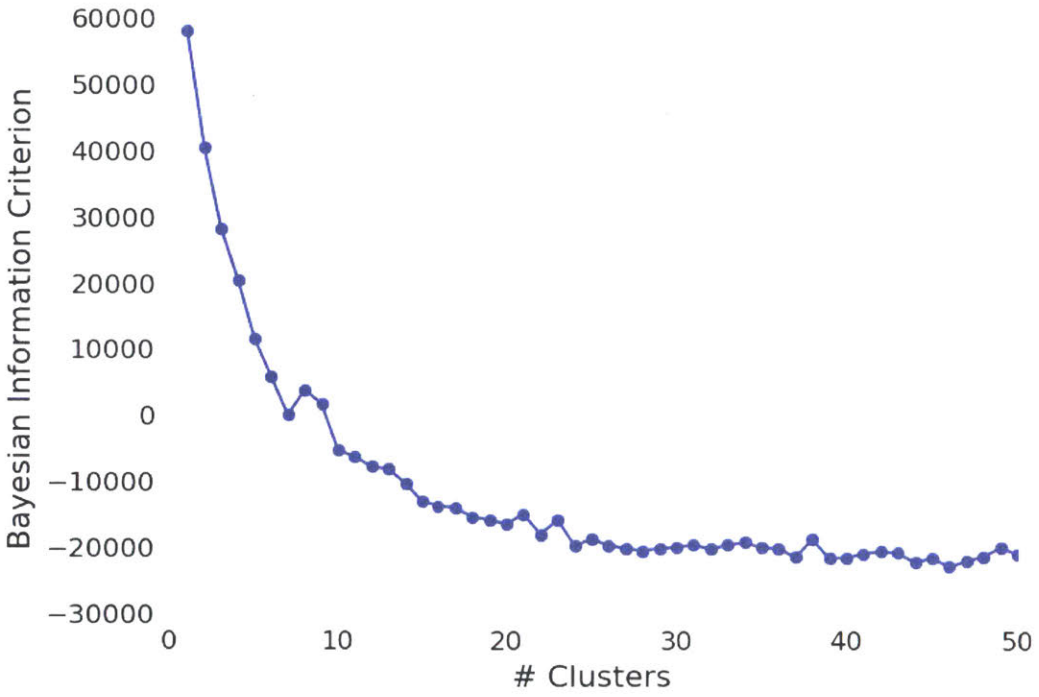
11.3.5 Bayesian Information Criterion

The Bayesian Information Criterion (BIC) balances a model's explanatory power, as quantified by the maximum log-likelihood, with a regularization penalty on model complexity. Of the four clustering algorithms considered here, only the GMM has a closed form solution for the BIC (Sec. 10.6.5). The criteria are presented in Fig. 11-40. In theory, the "best" clustering model should minimize the BIC. The BIC for the 1.6% enriched assembly (Fig. 11-40a) reaches a minimum with 20 – 24 clusters before the regularization penalty forces the BIC to increase with more clusters. Although the BIC for the colorset (Fig. 11-40b) converges to approximately -20,000 for 30 or more clusters, it does not start increasing even with 50 clusters. This indicates that even with 50 clusters, the inclusion of additional clusters continues to sufficiently reduce the maximum log-likelihood function of the GMM enough to balance the regularization penalty associated with introducing the new mixture components. It is expected that if enough clusters were included in the plot, the BIC would exhibit a clearly defined minimum as the regularization penalty begins to dominate the log-likelihood.

Of the model selection metrics evaluated in this thesis, the BIC is the only one which shows promise as a means to choose the "best" number of clusters for the *i*MGXS data processing pipeline. The BIC is a smoothly varying metric for which an unsupervised thresholding technique could potentially be applied to select the optimal number of clusters. In particular, future work may consider choosing the optimal number of clusters for the model with the minimum BIC. Furthermore, it is noteworthy for both benchmarks, that the BIC achieves over 80% its minimum value with only the first 10 clusters. This reflects the fact that only a few Gaussian mixture components are needed to model most of the clustering behavior, with diminishing returns achieved by more complex GMMs. An alternative approach might choose the optimal number of clusters by considering the fractional change of the BIC with each successive cluster (*i.e.*, choose the number of clusters which approaches 80% of the converged minimum). In effect, this would represent a variation of the BIC in which the regularization term is more heavily weighted than in Eqn. 10.44 in order to preferentially select models with fewer clusters.



(a)



(b)

Figure 11-40: Bayesian information criteria for the 1.6% enriched assembly (a) and 2×2 colorset with a water reflector (b) for each of the four clustering algorithms.

Model selection criteria are needed to determine the optimal number of clusters without human supervision. The BIC shows promise as a means to choose the optimal number of components for GMMs. It is not clear whether the Davies-Bouldin, Dunn and Calinski-Harabaz indices or silhouette coefficients may be useful model selection criteria for the *i*MGXS scheme.

11.4 Synthesis

As discussed in Chap. 1, various methods for reactor physics simulation – and in particular, pin-wise spatial homogenization – are distinguished by the tradeoffs each makes between accuracy and speed. The *i*MGXS scheme was developed to directly model all energy and spatial self-shielding effects to generate accurate MGXS for computationally efficient deterministic transport codes. Furthermore, as described in Fig. 10-1, the *i*MGXS scheme aims to make it possible to obtain accurate results from deterministic reactor physics calculations with MGXS generated by MC *faster* than would be possible from a direct calculation with MC. The following sections synthesize the performance and accuracy of the *i*MGXS scheme to derive some general conclusions about its computational efficiency relative to other simulation approaches. Sec. 11.4.1 compares the time required to converge the relative errors for the null, degenerate and *i*MGXS schemes, as well as the time required to converge the statistical uncertainties for the reference OpenMC simulations. Sec. 11.4.2 compares the traditional multi-level approach for MGXS generation to the spatial homogenization schemes introduced in this thesis which use the flux from the complete heterogeneous geometry for spatial homogenization.

11.4.1 A Comparison of Simulation Runtimes

This section examines the convergence rates for the U-238 capture rates presented in Sec. 11.2 to compare the computational resource requirements for the null, degenerate, LNS and *i*MGXS schemes relative to a reference MC calculation. It is challenging to construct a general figure of merit to compare the four different simulation workflows

since each one converges to a different solution. Instead, this section simply compares the simulation runtimes required for each workflow to converge the U-238 capture rates. In summary, the runtimes for the degenerate, LNS, *i*MGXS and reference MC simulation workflows were computed as follows:

1. **Count MC particle histories required to:**
   - *To converge degenerate and iMGXS U-238 capture rate errors*
   - *To converge OpenMC U-238 capture rate reference solution uncertainties*
2. **Compute MC runtime:**
   - *Multiply OpenMC particle tracking rate by number of histories*
3. **Compute total runtime:**
   - *Add OpenMC and OpenMOC runtimes*

Although the multi-step process to compute the OpenMC runtimes may seem straightforward, several subtle points influence the analysis at each step. First, the number of MC histories required for convergence was based on the mean rather than the maximum U-238 capture rate error since the mean error was shown to be more limiting for the OpenMC reference solutions in Sec. 11.2[97]. The first OpenMC statepoint for which each homogenization scheme converged or stabilized was determined based on a visual inspection of Figs. 11-31b to 11-35b, and the number of MC histories corresponding to that statepoint was recorded as the number of histories required to converge the error[98]. The numbers of histories required to converge the statistical uncertainties for the OpenMC reference solutions were determined from the statepoint at which the OpenMC uncertainties and the *i*MGXS errors crossed for each benchmark.

Second, the particle tracking rate is the number of particle histories simulated per second per CPU core. The tracking rate depends on the computational hardware employed to run the simulation, as well as the runtime parameters, including the number of histories per batch. Most importantly, the tracking rate scales with the number of

[97]The number of histories needed to converge the eigenvalues may be more limiting than that required to converge the U-238 capture rate errors. However, the convergence behavior of the eigenvalues is the result of a reaction rate imbalance due to the mixture of analog and track-length tally estimators used to generate MGXS (Sec. 11.2.1), which may be eliminated with advanced MC tally estimators [37].

[98]The number of particle histories was conservatively rounded up in each case.

tally *objects* in an OpenMC simulation. As shown in Tab. 11.4, the tracking rates are 3 – 4× faster when computing the reference solution than when generating MGXS[99]. The tracking rates without MGXS are employed to compute the OpenMC runtime for the reference solution, while the slower rates with MGXS are used to compute the OpenMC runtimes for the null, degenerate and *i*MGXS workflows.

Table 11.4: The particle tracking rates for OpenMC.

Benchmark	Active Histories / Core-Second	
	With MGXS	Without MGXS[100]
Assemblies	1100	3000
Colorsets	1100	3000
BEAVRS Quarter Core	600	2400

Third, the OpenMOC runtimes for the null, degenerate and *i*MGXS workflows were recorded and averaged for each benchmark. This is a reasonable approximation since the OpenMC runtimes are much larger than those for OpenMOC. Nevertheless, it should be noted that the OpenMOC runtimes for the degenerate scheme were on average 20% greater than those for the the null and *i*MGXS schemes. The cause for this slowdown is likely due to the overhead associated with processing the large number of distinct materials to tabulate MGXS for Coarse Mesh Finite Difference (CMFD) acceleration. Finally, this analysis neglects the time spent within the *i*MGXS data processing pipeline since this was negligible (*i.e.*, <0.1 core-hour) for all benchmarks and pipeline configurations (*e.g.*, clustering algorithms) that were tested.

The number of particle histories and OpenMC and OpenMOC runtimes are compiled in Tab. 11.5 for the single assembly and 2×2 colorset benchmarks. The runtimes for the *i*MGXS scheme correspond to BIRCH clustering of 10 clusters for the individual fuel assembly and periodic 2×2 colorset benchmarks, and 16 clusters for the colorset with a water reflector. As clearly observed from the data, the workflow for null homogenization is the fastest approach if the resulting error can be tolerated. However, the *i*MGXS scheme

[99]The OpenMC reference calculation required two mesh tally *objects* to compute pin-wise fission and U-238 capture rates. In contrast, 29 – 56 distinct tally *objects* were required to tally reaction rates and fluxes for each nuclide, energy group and reaction type to generate MGXS for each benchmark, even when using OpenMC's mergeable tally feature (Sec. 4.2.1.3).

[100]This corresponds to calculations which only tallied the pin-wise fission and U-238 capture rates.

outperforms both the degenerate and reference simulation workflows with respect to the prescribed relative errors on the mean U-238 capture rates. In particular, the *i*MGXS scheme requires 5 – 30× fewer MC particle histories relative to the reference OpenMC calculation for each of the benchmarks. Furthermore, the total runtime for the workflow with *i*MGXS homogenization is 4 – 30× faster than that for the degenerate scheme, and 2 – 10× faster than the reference calculation. The large variation in acceleration between the different benchmarks is largely due to the larger number of clusters required to minimize the error for the colorset with a water reflector as compared to the other benchmarks. In addition, the acceleration is less significant for the reflected colorset since the *i*MGXS scheme only reduces the error by approximately a factor of two, as compared to three or four for the other benchmarks. These results illustrate the potential for the *i*MGXS scheme to enable deterministic transport calculations to converge to a prescribed accuracy faster than a corresponding reference MC calculation.

11.4.2 A Comparison with a Multi-Level Approach

As a thought experiment, this section briefly considers the computational resources that would be required if one were to apply a traditional multi-level approach to generate MGXS with MC. It should be recalled from Sec. 2.5.2 – and specifically Fig. 2-3 – that multi-level schemes are commonly used to model energy and spatial self-shielding effects for increasingly complex geometric sub-components (*e.g.*, fuel pins, assemblies, etc.). The null, degenerate, LNS and *i*MGXS spatial homogenization schemes introduced in this thesis abandon the multi-level approach, and instead use MC to model the complete heterogeneous geometry in a single step. These schemes are more accurate since they use the "true" MC flux to collapse the cross sections. However, this increased accuracy comes at a cost – namely, the computational expense of modeling the complete heterogeneous geometry with MC. This section estimates the computational expense of employing MC to generate MGXS for the quarter core BEAVRS model with a multi-level scheme and compares it to the single-step schemes introduced in this thesis.

A multi-level MC scheme would employ *separate* MC simulations of each unique

Table 11.5: The computational resources required to converge the OpenMOC relative error or OpenMC statistical uncertainty on the mean pin-wise U-238 capture rates.

Benchmark	Workflow	Mean Rel. Err. [%]	Max Rel. Err. [%]	Histories in 1000s	Runtime [core-hours]		
					OpenMC	OpenMOC	Total
1.6% Assm	Null	0.50	-1.1	100	0.025	0.4	0.43
	LNS	0.10	0.29	20,000	5.1	0.4	5.5
	*i*MGXS	0.10	-0.44	20,000	5.1	0.4	5.5
	Degenerate	0.10	0.38	500,000	130	0.4	130
	Reference	0.10	0.37	650,000	60		60
3.1% Assm	Null	0.50	-1.3	100	0.025	0.4	0.43
	LNS	0.10	0.23	20,000	5.1	0.4	5.5
	*i*MGXS	0.10	-0.42	20,000	5.1	0.4	5.5
	Degenerate	0.10	-0.33	500,000	130	0.4	130
	Reference	0.10	0.26	650,000	60		60
3.1% Assm w/ 20 BPs	Null	0.41	-0.95	100	0.025	0.4	0.43
	LNS	0.13	0.25	20,000	5.1	0.4	5.5
	*i*MGXS	0.13	-0.50	20,000	5.1	0.4	5.5
	Degenerate	0.13	0.30	350,000	88	0.4	88
	Reference	0.13	0.24	450,000	42		42
2×2 Colorset	Null	0.45	-1.3	100	0.035	2.0	2.0
	LNS	0.16	0.44	40,000	10	2.0	12
	*i*MGXS	0.16	0.72	40,000	10	2.0	12
	Degenerate	0.16	-0.64	1,000,000	350	2.0	350
	Reference	0.16	0.45	1,000,000	93		93
2×2 Colorset w/ Reflector	Null	0.48	-2.0	100	0.035	5.0	5.0
	LNS	0.25	-2.0	80,000	20	5.0	25
	*i*MGXS	0.25	1.1	80,000	20	5.0	25
	Degenerate	0.25	-0.80	400,000	101	5.0	110
	Reference	0.25	1.9	550,000	51		51

sub-component in the BEAVRS model. The quarter core BEAVRS model is comprised of assemblies with three different fuel enrichments, each of which is composed of some configuration of CRGTs and BPs. Only two MC calculations would be required to calculate the MGXS for each enrichment with or without BPs, for a total of six single-assembly calculations. In addition, BEAVRS is composed of 3.1% enriched fuel assemblies that are facially adjacent to the baffle and reflector. This analysis assumes that the self-shielding effects from the baffle/reflector can be adequately modeled with a single MC calculation of an assembly adjacent to an assembly-equivalent area of baffle and water reflector[101].

[101]The total spatial domain size would be equivalent to two fuel assemblies.

Finally, there are four fuel assemblies in the quarter core BEAVRS model that are corner adjacent to the baffle/reflector. This analyis assumes that two MC calculations would be required to account for the unique self-shielding effects on these fuel assemblies. The first would model the three assemblies in an "L" configuration along the $y = x$ diagonal, along with the baffle, reflector, barrel and neutron shield pad[102]. The second would model the three fuel assemblies in an "L" configuration that are offset from the $y = x$ diagonal, along with the baffle, reflector and barrel[103].

The computational expense for a multi-level MC scheme can be estimated by summing the computational expense of generating MGXS for each sub-component. The results in Tab. 11.5 indicated that approximately 20,000,000 MC particle histories were required to converge the mean U-238 capture rate errors for *i*MGXS and LNS homogenization schemes for the individual fuel assemblies. As a simple approximation, the MC particle track density required to converge MGXS is assumed to scale linearly with the area of each sub-component, with 20,000,000 histories required for a single assembly. The estimated number of particle histories is then divided by the particle tracking rates in Tab. 11.4 to estimate the runtime for each sub-component. In this analysis, the particle tracking rate was assumed to be 1,100 histories per core-second.

The number of particle histories and runtimes for each of the assembly, colorset and corner assembly sub-components in the quarter core BEAVRS model are estimated in Tab. 11.6. According to this estimation approach, 420 million neutron histories would be required to converge the MGXS for each of the 9 sub-components with an estimated runtime of 110 core-hours. The final row of the table applies the same approach to estimate the number of histories to converge the MGXS for a single MC simulation of the quarter core BEAVRS geometry. In this case, there are $193/4$ fuel assemblies, each of which would require 20,000,000 histories, for a total of 970 million histories.

The estimated runtime to generate MGXS with a single MC simulation of the quarter core BEAVRS model is only about twice that required to generate MGXS for each sub-component. This is a relatively minor price to pay for the accuracy afforded by the MGXS

[102]The total spatial domain would be equivalent to 9 fuel assemblies in a 3×3 assembly configuration.
[103]The total spatial domain would be equivalent to 2 fuel assemblies in a 2×2 assembly configuration.

Table 11.6: The computational expense of a multi-level MGXS generation scheme.

Component	Number	Histories	Runtime [core-hours]
Multi-Level Approach			
Assemblies	6	120,000	30
Reflected Assembly	1	40,000,000	10
Corner Assemblies	2	260,000,000	90
TOTAL	9	420,000,000	110
Single MC Calculation Approach			
Quarter Core BEAVRS	1	970,000,000	250

collapsed with "true" MC flux from a single MC simulation of the complete heterogeneous geometry. Furthermore, the MGXS generated with a single MC calculation will accurately account for the effects of inter-assembly and and assembly-baffle/reflector interfaces and leakage on the flux. In addition, the software infrastructure required to generate and use MGXS for each of the sub-components in a core model would be greater than that needed to generate MGXS from a single MC simulation of the complete geometry. This analysis provides a rough estimate of the relative runtimes and is subject to several key assumptions. Nevertheless, it points to the potential for advanced pin-wise spatial homogenization schemes – such as those introduced in this thesis – to harness MC to efficiently generate accurate MGXS for deterministic transport codes.

Highlights

- The clustering algorithm and/or the number of clusters has no systematic impact on the eigenvalues due to the preservation of global reactivity.
- The U-238 capture rate errors are greatly reduced with only a few MGXS clusters, with diminishing returns for additional clusters.
- The *i*MGXS scheme requires more clusters to achieve the same error as the LNS scheme for all benchmarks except the colorset with a water reflector.
- *i*MGXS outperforms LNS for the reflected colorset since it distinguishes fuel pins along assembly-assembly and assembly-reflector interfaces.
- Approximately 10× fewer histories are required to converge the errors for *i*MGXS than for degenerate homogenization or a reference MC calculation.
- The total runtime for the entire simulation workflow with *i*MGXS homogenization is 5 – 10× faster than that for the degenerate scheme, and 1.5 – 4× faster than a reference MC calculation.
- The results in this chapter point to the potential for *i*MGXS as a means to efficiently generate MGXS with reactor agnostic MC calculations of the complete heterogeneous geometry in a single step.

Part V

Conclusions

Chapter 12

Conclusions

This research was motivated by a desire to obtain Monte Carlo (MC) quality solutions with computationally efficient deterministic neutron transport methods. This thesis approached this objective by employing continuous energy MC neutron transport simulations to generate accurate multi-group cross sections (MGXS) for fine-mesh deterministic transport codes. The new methods developed in this thesis were designed to accelerate the convergence of MGXS tallied on full-core, fine-spatial meshes by leveraging the phenomenon of pin-wise MGXS clustering. These methods reduce the computational burden of MC-based MGXS generation techniques, positioning it as a reactor agnostic alternative to today's deterministic methods which rely on engineering approximations.

This chapter concludes by evaluating the results presented in this thesis with respect to this over-arching objective, and by defining a roadmap of the milestones which must be addressed in the future to design a production-ready methodology for MC-based reactor agnostic fine-mesh MGXS generation. The key results demonstrated by this thesis are discussed in Sec. 12.1, the author's contributions to the field of reactor physics are outlined in Sec. 12.2, and opportunities for future research are summarized in Sec. 12.3.

12.1 Summary of Work

This thesis replaces the traditional multi-level framework used for MGXS generation with a single full-core MC calculation, as summarized in Sec. 12.1.1. Sec. 12.1.2 highlights

results which quantify different sources of approximation error in multi-group methods with MGXS generated by MC. The clustering of pin-wise MGXS due to spatial self-shielding effects is recapped in Sec. 12.1.3. Finally, Sec. 12.1.4 reviews the performance of new pin-wise spatial homogenization schemes which model MGXS clustering.

12.1.1 A Single-Step Approach for MGXS Generation

Today's state-of-the-art methods for MGXS generation use a multi-level approach in space, energy and angle to account for self-shielding effects while approximating the flux used to collapse cross sections (see Sec. 2.5.2). The flux approximations used in multi-level schemes are based on engineering prescriptions for specific reactor configurations and spectra and are not easily generalizable to new core designs. In addition, MC-based MGXS generation methods to date have retained the multi-level geometric framework to tabulate MGXS for individual reactor components for subsequent use in full-core multi-group calculations. Although the use of MC within a multi-level scheme eliminates the need to approximate the flux in energy, it does not account for spatial self-shielding effects throughout a reactor core. This thesis replaces the multi-level framework with a full-core MC calculation that simultaneously accounts for all energy and spatial effects with a single simulation of the complete heterogeneous geometry.

This work required the development of a "simulation triad" encompassing three primary simulation codes. The OpenMC Monte Carlo code was utilized to generate multi-group cross sections on high-spatial-fidelity tally meshes. Second, the OpenMOC code used MGXS for deterministic multi-group transport calculations. Finally, the OpenCG library enabled the processing and transfer of tally data on combinatorial geometry meshes between OpenMC and OpenMOC. The simulation triad directly modeled all energy and spatial self-shielding effects with a single full-core OpenMC calculation of the complete heterogeneous geometry to generate MGXS for use in OpenMOC.

12.1.2 Approximation Error in Multi-Group Methods

This thesis investigated approximation error present in multi-group solutions even when the "true" flux from MC is used to collapse cross sections for MGXS generation. The case studies presented in Chap. 5 quantified the eigenvalue bias between OpenMC and OpenMOC by separately varying the angular discretization, flat source region discretization and energy group structure while holding all other variables constant. A systematic bias on the order of -300 pcm was demonstrated for a representative 2D PWR fuel pin cell with converged angular and spatial discretization schemes and 70-group cross sections. The iso-in-lab scattering feature was employed in OpenMC to enable direct comparisons with OpenMOC which assumes an isotropic scattering source, but this only mitigated the bias by approximately 100 pcm. The remaining -200 pcm was shown to be caused by over-predictions of U-238 capture rates in resonance energy groups. As the energy group structures were refined, the errors were magnified for those energy groups encompassing the three lowest lying U-238 capture resonances.

In collaboration with Gibson [78], the U-238 capture rate errors and resulting eigenvalue bias were demonstrated to be the result of the flux separability approximation (Sec. 2.3.2) which permits the use of constant-in-angle MGXS. In particular, it was shown that the flux separability approximation is not generally valid for U-238 capture since the angular neutron flux leaving a fuel pin is much more self-shielded in resonance groups than it is for the neutron flux entering the fuel pin from the moderator. Since the angular dependence of MGXS is not typically modeled in deterministic multi-group transport codes, an equivalence scheme is needed to correct for the loss of angular information when cross sections are collapsed with the scalar rather than the angular flux.

Chap. 6 explored the use of SuPerHomogénéisation (SPH) factors as one possible equivalence scheme to enforce reaction rate preservation between OpenMC and OpenMOC. The SPH factor approach uses a reference fixed source to correct the total MGXS to preserve reaction rates between fine and coarse mesh methods. The SPH factors systematically eliminated the few percent reaction rate errors in U-238 resonance groups, and correspondingly reduced the eigenvalue bias from -200 pcm to approximately 10

pcm. In particular, the SPH factors reduced the total MGXS in resonance groups by 1 – 3% to resolve errors of the same magnitude in each of the lowest lying U-238 capture resonance groups. Notwithstanding these results, it is unclear if a generalizable scheme based upon SPH factors may be used to correct for the flux separability approximation. Future work should further investigate equivalence methods which adequately preserve reaction rates in fine-mesh transport methods with MC-generated MGXS.

12.1.3 Clustering of Pin-Wise MGXS

This thesis explored the dispersion and clustering of pin-wise MGXS due to the spatial self-shielding spectral effects in heterogeneous single assembly benchmarks, 2×2 assembly colorsets and a quarter core model of the BEAVRS PWR geometry. The results in Chap. 9 demonstrated that the population variance of pin-wise MGXS increases with the introduction of geometric heterogeneities such as control rod guide tubes (CRGTs), burnable poisons (BPs) and water reflectors. The magnitude of the dispersion depended on the sensitivity of each nuclide, reaction type and energy group to spatial self-shielding effects. Furthermore, the distributions of pin-wise MGXS plotted as histograms illustrated the clustering of MGXS in fuel pins with similar neighboring heterogeneities, and hence similar spatially self-shielded flux spectra. Although U-235 thermal fission MGXS exhibited more clearly defined clusters than U-238 capture MGXS, the U-238 capture MGXS varied by up to 1.8% about the population mean while U-235 fission MGXS varied by only 1.2% about the mean. These results indicate that MGXS clustering is more challenging to identify but more important to model for accurate U-238 capture rates than for U-235 fission rates.

This thesis performed a series of case studies to investigate the impact of modeling (or neglecting) the clustering of pin-wise MGXS due to spatial self-shielding effects[104]. The null and degenerate pin-wise spatial homogenization schemes were designed to quantify the magnitude of the approximation error in deterministic methods that can be resolved by accounting for MGXS clustering. Both schemes use a single MC calculation

[104]Although all spatial zones may experience spatial self-shielding, this thesis specifically modeled the impact of spatial self-shielding on MGXS in fissile materials.

of the complete heterogeneous geometry to collapse MGXS with the "true" flux. Null homogenization simply assigns all fuel pins to the same MGXS cluster and averages all spatial self-shielding effects across the entire geometry. The null scheme makes no effort to account for spatial self-shielding effects experienced by different fuel pins, and computes a single MGXS for each fuel enrichment. Degenerate spatial homogenization takes the opposite approach and assigns each fuel pin its own MGXS, and is equivalent to modeling each fuel pin as a unique MGXS cluster. The degenerate scheme accounts for all spatial self-shielding effects experienced by each instance of each fuel pin throughout a heterogeneous geometry. Although degenerate homogenization is more accurate than null homogenization, it requires many more MC particle histories to converge the uncertainties on the MGXS tallied separately for each fuel pin.

A series of OpenMOC simulations were performed in Chap. 8 with 70-group MGXS libraries prepared by the null and degenerate spatial homogenization schemes for six heterogeneous PWR benchmarks to measure the impact of MGXS clustering on deterministic calculations. In each case, the OpenMOC eigenvalue, pin-wise fission rate and U-238 capture rate predictions were compared with reference OpenMC results[105]. The OpenMOC eigenvalues were within 100 – 250 pcm of the reference OpenMC eigenvalues for all six benchmarks. More importantly, the effect of MGXS clustering had no impact on the eigenvalue predictions since null and degenerate homogenization use the same MC flux to collapse the cross sections and therefore preserve global reaction rates. The effects of MGXS clustering only marginally impacted the OpenMOC fission rates errors which were <2% for all six benchmarks[106].

The U-238 capture rate errors were much more sensitive to MGXS clustering than the fission rates. The degenerate homogenization scheme reduced the errors by 2 – 4× with respect to the null scheme for each of the single assembly and colorset benchmarks, approaching the same error magnitude as that observed for the fission rates. The errors for the null scheme were largest for fuel pins near control rode guide tubes and

[105]The case studies were performed for 2-, 8- and 70-group cross sections. The results cited in this chapter correspond to the 70-group calculations since the fine group structure was necessary to resolve OpenMOC's U-238 capture rate predictions to within 2% of the reference OpenMC results.

[106]The degenerate homogenization fission rate errors were 5 – 20% less than those for null homogenization.

along inter-assembly and assembly-reflector interfaces. The degenerate homogenization scheme reduced the errors for these pins since it accounts for spatial self-shielding spectral effects, such as the additional moderation provided by neighboring control rod guide tubes and/or water reflectors. These results indicate that the U-238 capture rate errors in deterministic multi-group calculations of PWRs are largely dominated by the approximation(s) made to model pin-wise MGXS clustering.

A large fraction of the fissions that occur in LWR fuel at the end-of-life occur in the Pu-239 that is bred from U-238 capture reactions. As a result, accurate, high-spatial-fidelity fission rate predictions are limited by the accuracy of the U-238 capture rate predictions in burn-up calculations. The results presented in this thesis underscore the importance of modeling MGXS clustering in order to predict Pu-239 production and therefore fission rates over time. This is important not only for LWRs, but also for advanced reactor concepts that are designed to convert sizable amounts of U-238 to Pu-239 for consumption. In general, MGXS clustering is important to model in reactors with a complex configuration of absorbing (*e.g.*, control) and/or moderating materials and relatively short neutron mean free paths (*i.e.*, thermal reactors) such that geometric heterogeneities shield the localized flux throughout the reactor.

12.1.4 Pin-Wise Spatial Homogenization Schemes

Although the degenerate scheme greatly reduces U-238 capture rate errors by accounting for MGXS clustering, it is computationally expensive to converge the MC tallies since the particle track densities in each spatial tally zone is quite small. This thesis developed two new pin-wise spatial homogenization schemes which aim to strike a balance between accuracy and computational efficiency by accounting for MGXS clustering to accelerate the convergence of the MGXS tallies in each fine-spatial-mesh tally zone. The two new schemes are motivated by the idea that fuel pins with similar neighboring heterogeneities will have similar microscopic MGXS. In particular, these schemes aim to approach the accuracy of the degenerate scheme by accounting for spatial self-shielding effects, while simultaneously approaching the MC convergence of the null scheme by homogenizing

over as many spatial tally zones (*i.e.*, fuel pins) as possible.

This thesis proposes both engineering-based clustering and statistical clustering methods to accelerate the convergence of full-core MC calculations for MGXS generation. Local Neighbor Symmetry (LNS) spatial homogenization (Chap. 9) is an engineering-based approach which clusters MGXS based on a nearest neighbor analysis of the fuel pins in a combinatorial geometry. The LNS scheme is akin to geometric templates employed by some lattice physics codes to predict which groupings of fuel pins are likely to experience similar spatial self-shielding effects and hence have similar MGXS. The MGXS are homogenized from "noisy" MC tally data across all pins within the same LNS grouping. Inferential MGXS (*i*MGXS) spatial homogenization (Chap. 10) uses statistical clustering algorithms to infer MGXS clusters directly from "noisy" MC tally data. Unlike the LNS scheme, the *i*MGXS has no knowledge of a reactor's geometric or materials configuration and instead relies on unsupervised machine learning techniques to determine which fuel pins to cluster in spatial homogenization zones. The *i*MGXS scheme is a multi-stage data processing pipeline and includes feature extraction, feature selection, dimensionality reduction, predictor training, model selection and spatial homogenization stages. Both LNS and *i*MGXS homogenization schemes attempt to model MGXS clustering with fewer materials than degenerate homogenization in order to accelerate the MC tally convergence rate by homogenizing MGXS across many fuel pins.

The efficacy of the LNS and *i*MGXS schemes to identify MGXS clusters was evaluated for each of the six heterogeneous PWR benchmarks. The "clustered geometries" for the LNS scheme illustrate the method's ability to distinguish fuel pins with neighboring CRGTs and/or BPs, but its inability to distinguish fuel pins at the interfaces with neighboring assemblies, reflectors or baffles. In contrast, the *i*MGXS scheme distinguished pins with similar neighboring CRGTs and/or BPs, and subsequently distinguished pins along interfaces into unique clusters. In general, the clustered geometries indicate that *i*MGXS scheme can flexibly accommodate arbitrary core heterogeneities better than heuristic approaches like LNS which must be customized for particular core geometries.

A series of OpenMOC simulations were performed with MGXS libraries prepared by

the LNS (Chap. 9) and *i*MGXS (Chap. 11) spatial homogenization schemes. As expected, the eigenvalues predicted by both schemes were nearly identical (to within 10 pcm) since both schemes preserve global reaction rates. Although fission rates are only marginally impacted by MGXS clustering, the predictions with the LNS and *i*MGXS schemes were very nearly as accurate as those for the degenerate scheme. Most importantly, the U-238 capture rate errors approached those for the degenerate scheme to varying degrees for both schemes. The LNS scheme performed as well the degenerate scheme for the single assembly benchmarks, but failed to systematically reduce the largest errors in the pins along inter-assembly and assembly-reflector interfaces. The *i*MGXS scheme largely reduced the U-238 capture rate errors with just a few clusters with relatively diminishing returns for more clusters. The *i*MGXS scheme required more materials (*i.e.*, clusters) than LNS to achieve the same accuracy for single assembly benchmarks. However, *i*MGXS outperformed LNS for those benchmarks with inter-assembly and assembly-reflector interfaces, since it assigned unique MGXS to the fuel pins along the interfaces to account for the local spatially self-shielded flux spectra.

In addition, the simulations were performed to quantify the number of MC particle histories required to sufficiently converge the MGXS for stable OpenMOC solutions with each pin-wise spatial homogenization scheme. The eigenvalues for the null, degenerate, LNS and *i*MGXS schemes converged to the same value with approximately 10^8 particle histories. As expected, the pin-wise U-238 capture rates converged in accordance with the number of clusters used in each pin-wise spatial homogenization scheme. In particular, the capture rates for the null scheme quickly converged with only 10^6 histories, but were still not yet fully converged for degenerate homogenization even with 10^9 histories. In contrast, the LNS and *i*MGXS schemes converged faster than the degenerate scheme since they averaged the MGXS tallied for the pins assigned to each cluster. Furthermore, both schemes converged faster than the statistical uncertainties of the corresponding reference OpenMC calculation. The magnitude of the acceleration achieved by both schemes depends on both the number of clusters required to meet the desired level

of accuracy, as well as the number of fuel pins assigned to each cluster[107]. The mean capture rate rate errors converged 5 – 20× faster than the degenerate scheme or the OpenMC reference solution uncertainties for the benchmarks considered in this thesis.

These results demonstrate a path forward to generate MGXS with reactor agnostic MC for computationally efficient deterministic transport codes. This thesis replaced traditional multi-level MGXS generation schemes with MC calculations of the complete heterogeneous geometry to generate reactor agnostic MGXS in a single step. Engineering-based and statistical clustering algorithms were developed to model the clustering of pin-wise MGXS to reduce the U-238 capture rate errors while simultaneously accelerating the convergence of the tallied MGXS. The LNS and *i*MGXS schemes enabled deterministic reactor physics simulations to produce accurate results from MGXS generated by MC faster than would be possible with a direct calculation with MC. Furthermore, the *i*MGXS scheme was shown to be advantageous over geometric heuristic approaches such as LNS which must be highly customized for specific types of core geometries.

[107]The error convergence is limited by the number of pins assigned to the smallest MGXS cluster. The pins adjacent to the baffle/reflector – which exhibited the largest U-238 capture rate errors – comprised the smallest clusters in each benchmark and were the most limiting to the overall convergence.

12.2 Contributions

- Generated MGXS with a single MC simulation of a complete heterogeneous reactor geometry for use in fine-mesh deterministic transport calculations.
- Implemented a simulation triad that generated MGXS with OpenMC, used MGXS in deterministic OpenMOC calculations, and transferred MGXS between codes and built clustered geometries with OpenCG.
- Explored SPH factors as an equivalence scheme between continuous energy MC and multi-group MOC methods and to correct for approximation error resulting from constant-in-angle MGXS.
- Developed pin-wise spatial homogenization schemes to model spatial self-shielding for each fuel pin with varying degrees of granularity and complexity while simultaneously accelerating MC tally convergence:
  - Local Neighbor Symmetry (LNS) homogenization uses a nearest neighbor-like analysis of a combinatorial geometry to predict which fuel pins will experience similar spatial self-shielding effects.
  - Inferential MGXS (*i*MGXS) homogenization uses unsupervised statistical clustering algorithms to predict MGXS clustering from "noisy" MC tally data without any knowledge of the reactor geometry.
- Quantified approximation error between reference OpenMC and deterministic OpenMOC calculations of the eigenvalues, pin-wise fission rates and U-238 capture rates for each homogenization scheme.
- Demonstrated that LNS and *i*MGXS spatial homogenization schemes are credible paths to generate very accurate MGXS and require fewer MC histories to generate MGXS and to converge deterministic calculations to a given accuracy than an equivalent MC calculation.

12.3 Future Work

This thesis identified several issues which must be investigated in the future to position single-step MC simulations as a practical alternative to traditional multi-level MGXS generation techniques. The following sections itemize the author's assessment of the most important outstanding issues and the order in which they should be addressed. Sec. 12.3.1 discusses case studies which should be performed to optimize and to evaluate the performance of the *i*MGXS spatial homogenization scheme. Sec. 12.3.2 highlights several key shortcomings in the present approach to generate MGXS with MC which must be confronted to resolve lingering sources of approximation error between continuous energy MC and deterministic multi-group methods.

12.3.1 Further Evaluation of the *i*MGXS Scheme

A number of challenges must be addressed in the future in order for *i*MGXS to be useful in a production code setting. Sec. 12.3.1.1 discusses the need for future studies to determine the number of particle histories required to accurately identify MGXS clusters from "noisy" MC tally data. Sec. 12.3.1.2 outlines the many remaining challenges to optimize the various stages of the *i*MGXS data processing pipeline. Sec. 12.3.1.3 highlights issues which must be resolved to enable fast and reliable calculations with the simulation triad of OpenMC, OpenMOC and OpenCG.

12.3.1.1 Evaluate the *i*MGXS Scheme with Noisy Tally Data

The case studies in Sec. 11.2 investigated the number of MC particle histories required to sufficiently converge MGXS tallied by OpenMC for stable deterministic solutions with OpenMOC. Each of the studies used "fully converged" MC tally data to train a clustering model, which was then repeatedly used to cluster fuel pins for spatial homogenization with "noisier" MC tally data. This was a useful exercise since it provided a lower bound on how quickly MGXS would converge if a formulation of the *i*MGXS pipeline is devised in the future that can accurately identify clusters from "noisy" MC tally data. However, these results did not adequately gauge the number of MC particle histories required to

stabilize the OpenMOC solutions with *i*MGXS in a production-like setting.

Future convergence studies should train clustering models with the "noisy" MC tally data for each statepoint from an OpenMC simulation to determine how quickly the clustering predictions converge. Based on this author's preliminary studies, a larger fraction of fuel pins are assigned to the "wrong" MGXS clusters when the clustering models are trained with "noisy" MC tally data. As a result, it is reasonable to expect that the approximation error between OpenMOC and the reference OpenMC solutions will be larger than that observed in Sec. 11.2 when relatively few particle histories are used to generate MGXS. However, given enough particle histories, the approximation errors should approach the converged values presented in Sec. 11.2. The results from future convergence studies should be used to improve the estimates for the expected runtimes for OpenMOC simulations with *i*MGXS in Sec. 11.4.1.

12.3.1.2 Optimize the *i*MGXS Data Processing Pipeline Configuration

There is a plethora of opportunities to optimize the performance of the *i*MGXS spatial homogenization scheme. This thesis tested only but a few of the many different possible machine learning algorithms that may be interchangeably used within the *i*MGXS data processing pipeline. Future work should systematically evaluate various configurations of the pipeline and score each by the number of MC histories needed to accurately identify MGXS clusters from "noisy" MC tally data. The feature extraction stage (Sec. 10.2) may be improved with the development of new MC tallies, or combinations of tallies, to be used as features which may indicate the existence of MGXS clustering with fewer particle histories than the features introduced in this thesis. New techniques should be evaluated for the feature selection stage (Sec. 10.3) to identify those features most useful for clustering analysis. The dimensionality reduction stage (Sec. 10.4) should be a key focal point of future work to determine how to extract latent information from highly correlated features and to reduce the number of features used in clustering analysis[108]. Future efforts may explore different clustering algorithms – and optimize the

[108]Although the clustered geometries produced with three different dimensionality reduction techniques are briefly compared in App. D.2, dimensionality reduction was not used by the *i*MGXS pipeline for the simulation results presented in Chap. 11.

parameters for each with cross-validation – for the predictor training stage (Sec. 10.5). The performance of each clustering algorithm should be scored according to the number of clusters required to reduce the U-238 capture rate errors in OpenMOC. Finally, future work should identify one or more heuristics to reliably determine the most appropriate number of clusters[109] for the model selection stage (Sec. 10.6).

12.3.1.3 Optimize the Computational Performance of the Simulation Triad

The analysis of the quarter core BEAVRS model in Chaps. 8 to 11 was limited by some computational bottlenecks that must be resolved to enable practical and reliable full-core calculations with the simulation triad (Chap. 4). First and foremost, a quarter pin mesh was required to guarantee stable solutions for the BEAVRS model with OpenMOC's implementation of Coarse Mesh Finite Difference (CMFD) acceleration, which greatly increased the runtime of each simulation. Future work should improve the stability of CMFD in OpenMOC for coarser meshes (*e.g.*, quarter assembly meshes) and optimize the computational efficiency of the CMFD implementation overall. A linear source approximation (instead of a flat source approximation) would greatly reduce the number of discretized spatial zones, likewise reducing both the runtime and memory footprint for OpenMOC simulations. Furthermore, the computational efficiency of the transport solver in OpenMOC would be greatly improved with the use of Single Instruction, Multiple Data (SIMD) vectorization over energy groups. In addition to improving OpenMOC's computational efficiency, the performance of the OpenMC code may be improved in the future. As noted in Tab. 11.4, the per-core particle tracking rates for OpenMC were 3 – 4× slower with tallies. This performance degradation may be mitigated in the future with spatial domain decomposition [54], tally servers [53], and/or SIMD vectorization of the tallying algorithms in OpenMC.

[109] None of the model selection techniques evaluated in Sec. 11.3 performed as expected except for the Bayesian Information Criterion, which showed some promise as a means to select the number of components for Gaussian Mixture Models.

12.3.2 Improve Methods to Generate MGXS with MC

This thesis identified several areas which must be addressed in order to make MC-based MGXS generation methods practical for fine-mesh transport codes. Sec. 12.3.2.1 presents the need for new statistical estimators to improve the tally efficiency of multi-group scattering matrices. As discussed in Sec. 12.3.2.2, new methods must be developed to compute transport-corrected MGXS with MC which account for anisotropic scattering, thereby eliminating the isotropic in lab scattering approximations used throughout this thesis. Sec. 12.3.2.3 motivates the need for an equivalence method to enforce reaction rate preservation between continuous energy MC and multi-group deterministic methods with scalar flux-weighted MGXS. Finally, future work should consider employing MC to generate MGXS for multi-physics applications as highlighted in Sec. 12.3.2.4.

12.3.2.1 Statistical Efficiencies of Tally Estimators

This thesis generated MGXS using a mixture of track-length and analog tally estimators as described in Sec. 3.2.1.7. Track-length estimators were used for reaction rate and flux tallies which depend on the incoming neutron energy (*e.g.*, total, fission), while less efficient analog estimators were necessary for tallies with an outgoing energy dependence, including those used to compute multi-group scattering matrices and fission emission spectra. Although the expectation of track-length and analog statistical estimates are identical, the variance is typically much larger for the analog estimates for a given number of particle histories. As a result, the MGXS tallied with a mixture of track-length and analog estimators will not preserve neutron balance without a sufficient number of particle histories. This was observed in Sec. 11.2.1 where the eigenvalue estimates for different pin-wise spatial homogenization schemes only converged to the same value after 10^8 particle histories were simulated. Future work may rectify this by designing a more statistically efficient tally estimator for multi-group scattering matrices and fission spectra. Nelson developed a theoretically rigorous methodology for this purpose in [37], though a simple heuristic to simultaneously tally to multiple outgoing energy bins in scattering collisions may be sufficient for hydrogenous reactor systems.

12.3.2.2 Tally Estimators for Transport-Corrected MGXS

This thesis implemented a feature to use isotropic in lab ("iso-in-lab") scattering in OpenMC to enable direct comparisons with OpenMOC which uses an isotropic scattering source. Isotropic scattering is generally not a valid approximation for nuclear reactors as is vividly evident by comparing the pin-wise fission and U-238 capture rates for BEAVRS modeled in OpenMC with iso-in-lab scattering (Figs. 7-14 and 7-20) and anisotropic scattering (Fig. B-1)[110]. Future work should account for anisotropic scattering in deterministic multi-group calculations with MGXS generated by MC. The most theoretically rigorous approach would model a truncated form of the Legendre moment expansion of the scattering kernel (Sec. 2.2.2) as is done in many deterministic transport codes today. This approach requires the use of multi-group scattering matrix Legendre moments which may be computed using the `openc.mgxs` module. However, a model of the scattering moments does increase the computational complexity for deterministic codes.

A common alternative is to use a transport correction to the total MGXS and scattering matrix (Sec. 2.2.3). Future work may aim to develop and validate methods to model transport corrections for MGXS generated from MC tallies. The work by Liu [25] to tally diffusion coefficients without approximation in MC may be extended to develop a reliable method to tally transport-corrected MGXS in MC. The development of an improved MC-based method to tally transport-corrected MGXS may also make it possible to maintain a high-level of accuracy with reduced energy group structures. The results in Chaps. 8 determined that 70 groups were required to sufficiently minimize the U-238 capture rate errors. Future work should employ transport-corrected MGXS in energy group condensation studies to determine if it is possible to maintain a high-leve of accuracy with fewer energy groups

[110]The fission and capture rates are much more highly peaked in the assemblies near the periphery of the core since more neutrons scatter back into the reactor from the reflector with the iso-in-lab approximation. The reaction rate distributions are flatter when modeled with anisotropic scattering since more neutons leak from the core.

12.3.2.3 Equivalence Methods for Angular-Dependent MGXS

The results in Chap. 5 demonstrated how constant-in-angle MGXS leads to U-238 capture rate errors in resonance groups, resulting in a few hundred pcm eigenvalue bias for a simple PWR pin cell problem[111]. Chap. 6 successfully used SPH factors as an equivalence scheme to correct for the approximation error resulting from the use of the angular rather than the scalar flux to collapse cross sections. However, the SPH factor approach suffers from a number of shortcomings (Sec. 6.4) and it is unclear whether it may be broadly applied to correct for the flux separability approximation in MGXS generated from MC. In particular, the SPH factor method is complicated by the need to compute a reference fixed source with MC for the spatial discretization mesh used in multi-group methods. In addition, SPH factors do not simply correct for the error due to the flux separability approximation, but instead indiscriminately correct for all sources of approximation error between MC and multi-group methods.

As a result, new equivalence schemes should be explored in the future to specifically correct for the flux separability approximation in arbitrarily discretized geometries. For example, the angular dependence of the total MGXS may be adequately embedded into the scattering kernel using the Consistent-P approximation [7] (also known as the BHS approximation). Alternatively, a coarse set of angular-dependent MGXS may mitigate most of the approximation error observed between continuous energy MC and deterministic multi-group methods. For example, a simple approximation might model two different total MGXS for neutrons entering or leaving a fuel pin. Although such a coarse angular scheme would not capture the high degree of angular variation of MGXS, it might capture enough to adequately reduce the integrated error[112].

[111] It is also likely that the flux separability approximation was a dominant factor in the <2% pin-wise fission and U-238 capture rate errors which remained even with the use of highly accurate degenerate spatial homogenization (Chap. 9), though this was not definitively proven in this thesis.

[112] Recent work by Gibson [78] showed that the use of finely discretized angular-dependent MGXS very nearly eliminated the error in resonance groups resulting from the flux separability approximation.

12.3.2.4 Multi-Physics Feedback in MGXS

This thesis performed steady-state calculations of PWR benchmarks with fresh fuel at hot zero power conditions. Future work should build upon the progress made in this thesis to generate MGXS with MC for multi-physics applications. Further development of the *i*MGXS scheme may account for thermal-hydraulic feedback in which fuel temperature and moderator density gradients are modeled in MC calculations for MGXS generation. For example, MC calculations performed using Doppler broadened cross sections sampled on-the-fly with the windowed multipole method in OpenMC [116] to generate MGXS which accurately reflect the distribution of fuel temperatures across a reactor core. The extensions of the *i*MGXS scheme would be most useful if developed in conjunction with an extension of the simulation framework from two to three dimensions. In addition, MC may be used to generate MGXS for deterministic codes at each burnup step in a depletion calculation with the appropriate isotopic vectors modeled in each fuel pin. Furthermore, future work should consider using the *i*MGXS scheme in calculations with thermal-hydraulic feedback and nuclide depletion where the moderator density, fuel temperature and burnup may be used as features to predict MGXS clustering. Indeed, it may be valuable to expand upon this thesis' dependence on clustering algorithms and instead use multivariate regression models (*e.g.*, decision tree regressors) to predict continuously varying MGXS based on features such as fuel temperature and burnup[113].

12.3.3 Inspiration for New Research Directions

The *i*MGXS scheme uses unsupervised statistical clustering methods to replace engineering-based approximations or heuristics with data-informed decision-making to cluster MGXS for pin-wise spatial homogenization. This thesis may inspire the future application of machine learning to closely related issues, such as automating the selection of energy group boundaries for MGXS generation. However, it is this author's opinion that MGXS clustering is only one of many potential applications for which machine learning may

[113] Regression models may pose a challenge since continuously varying MGXS predictions will not preserve global reactivity, which is guaranteed by simple(r) discrete predictions from clustering models used to compute track density-weighted average MGXS (Sec. 9.2.2) in this thesis.

be used to improve the accuracy and speed of numerical simulations. The scientific computing community has an increasingly vast array of machine learning algorithms at its disposal which may be used to automate the tedious and error prone process of parameter selection. Only the sky is the limit when it comes to harnessing the power of machine learning in the service of advanced computational physics simulation.

Appendices

Appendix A

Energy Group Structures

The energy group structures are from the CASMO-4 lattice physics code [75]. The author gratefully acknowledges Geoffrey Gunow for his assistance in obtaining the structures.

Table A.1: One group energy boundaries.

Group No.	Lower Bound [MeV]	Upper Bound [MeV]
1	0.0000E+00	2.0000E+01

Table A.2: Two group energy boundaries.

Group No.	Lower Bound [MeV]	Upper Bound [MeV]
2	0.0000E+00	6.2500E-07
1	6.2500E-07	2.0000E+01

Table A.3: Four group energy boundaries.

Group No.	Lower Bound [MeV]	Upper Bound [MeV]
4	0.0000E+00	6.2500E-07
3	6.2500E-07	5.5300E-03
2	5.5300E-03	8.2100E-01
1	8.2100E-01	2.0000E+01

Table A.4: Eight group energy boundaries.

Group No.	Lower Bound [MeV]	Upper Bound [MeV]
8	0.0000E+00	5.8000E-08
7	5.8000E-08	1.4000E-07
6	1.4000E-07	2.8000E-07
5	2.8000E-07	6.2500E-07
4	6.2500E-07	4.0000E-06
3	4.0000E-06	5.5300E-03
2	5.5300E-03	8.2100E-01
1	8.2100E-01	2.0000E+01

Table A.5: Sixteen group energy boundaries.

Group No.	Lower Bound [MeV]	Upper Bound [MeV]
16	0.0000E+00	3.0000E-08
15	3.0000E-08	5.8000E-08
14	5.8000E-08	1.4000E-07
13	1.4000E-07	2.8000E-07
12	2.8000E-07	3.5000E-07
11	3.5000E-07	6.2500E-07
10	6.2500E-07	8.5000E-07
9	8.5000E-07	9.7200E-07
8	9.7200E-07	1.0200E-06
7	1.0200E-06	1.0970E-06
6	1.0970E-06	1.1500E-06
5	1.1500E-06	1.3000E-06
4	1.3000E-06	4.0000E-06
3	4.0000E-06	5.5300E-03
2	5.5300E-03	8.2100E-01
1	8.2100E-01	2.0000E+01

Table A.6: Twenty-five group energy boundaries.

Group No.	Lower Bound [MeV]	Upper Bound [MeV]
25	0.0000E+00	3.0000E-08
24	3.0000E-08	5.8000E-08
23	5.8000E-08	1.4000E-07
22	1.4000E-07	2.8000E-07
21	2.8000E-07	3.5000E-07
20	3.5000E-07	6.2500E-07
19	6.2500E-07	9.7200E-07
18	9.7200E-07	1.0200E-06
17	1.0200E-06	1.0970E-06
16	1.0970E-06	1.1500E-06
15	1.1500E-06	1.8550E-06
14	1.8550E-06	4.0000E-06
13	4.0000E-06	9.8770E-06
12	9.8770E-06	1.5968E-05
11	1.5968E-05	1.4873E-04
10	1.4873E-04	5.5300E-03
9	5.5300E-03	9.1180E-03
8	9.1180E-03	1.1100E-01
7	1.1100E-01	5.0000E-01
6	5.0000E-01	8.2100E-01
5	8.2100E-01	1.3530E+00
4	1.3530E+00	2.2310E+00
3	2.2310E+00	3.6790E+00
2	3.6790E+00	6.0655E+00
1	6.0655E+00	2.0000E+01

Table A.7: Forty group energy boundaries.

Group No.	Lower Bound [MeV]	Upper Bound [MeV]
40	0.0000E+00	1.5000E-08
39	1.5000E-08	3.0000E-08
38	3.0000E-08	4.2000E-08
37	4.2000E-08	5.8000E-08
36	5.8000E-08	8.0000E-08
35	8.0000E-08	1.0000E-07
34	1.0000E-07	1.4000E-07
33	1.4000E-07	1.8000E-07
32	1.8000E-07	2.2000E-07
31	2.2000E-07	2.8000E-07
30	2.8000E-07	3.5000E-07
29	3.5000E-07	6.2500E-07
28	6.2500E-07	8.5000E-07
27	8.5000E-07	9.5000E-07
26	9.5000E-07	9.7200E-07
25	9.7200E-07	1.0200E-06
24	1.0200E-06	1.0970E-06
23	1.0970E-06	1.1500E-06
22	1.1500E-06	1.3000E-06
21	1.3000E-06	1.5000E-06
20	1.5000E-06	1.8550E-06
19	1.8550E-06	2.1000E-06
18	2.1000E-06	2.6000E-06
17	2.6000E-06	3.3000E-06
16	3.3000E-06	4.0000E-06
15	4.0000E-06	9.8770E-06
14	9.8770E-06	1.5968E-05
13	1.5968E-05	2.7700E-05
12	2.7700E-05	4.8052E-05
11	4.8052E-05	1.4873E-04
10	1.4873E-04	5.5300E-03
9	5.5300E-03	9.1180E-03
8	9.1180E-03	1.1100E-01
7	1.1100E-01	5.0000E-01
6	5.0000E-01	8.2100E-01
5	8.2100E-01	1.3530E+00
4	1.3530E+00	2.2310E+00
3	2.2310E+00	3.6790E+00
2	3.6790E+00	6.0655E+00
1	6.0655E+00	2.0000E+01

Table A.8: Seventy group energy boundaries.

Group No.	Lower Bound [MeV]	Upper Bound [MeV]
70	0.0000E+00	5.0000E-09
69	5.0000E-09	1.0000E-08
68	1.0000E-08	1.5000E-08
67	1.5000E-08	2.0000E-08
66	2.0000E-08	2.5000E-08
65	2.5000E-08	3.0000E-08
64	3.0000E-08	3.5000E-08
63	3.5000E-08	4.2000E-08
62	4.2000E-08	5.0000E-08
61	5.0000E-08	5.8000E-08
60	5.8000E-08	6.7000E-08
59	6.7000E-08	8.0000E-08
58	8.0000E-08	1.0000E-07
57	1.0000E-07	1.4000E-07
56	1.4000E-07	1.8000E-07
55	1.8000E-07	2.2000E-07
54	2.2000E-07	2.5000E-07
53	2.5000E-07	2.8000E-07
52	2.8000E-07	3.0000E-07
51	3.0000E-07	3.2000E-07
50	3.2000E-07	3.5000E-07
49	3.5000E-07	4.0000E-07
48	4.0000E-07	5.0000E-07
47	5.0000E-07	6.2500E-07
46	6.2500E-07	7.8000E-07
45	7.8000E-07	8.5000E-07
44	8.5000E-07	9.1000E-07
43	9.1000E-07	9.5000E-07
42	9.5000E-07	9.7200E-07
41	9.7200E-07	9.9600E-07
40	9.9600E-07	1.0200E-06
39	1.0200E-06	1.0450E-06
38	1.0450E-06	1.0710E-06
37	1.0710E-06	1.0970E-06
36	1.0970E-06	1.1230E-06
35	1.1230E-06	1.1500E-06
34	1.1500E-06	1.3000E-06
33	1.3000E-06	1.5000E-06
32	1.5000E-06	1.8550E-06
31	1.8550E-06	2.1000E-06
30	2.1000E-06	2.6000E-06
29	2.6000E-06	3.3000E-06
28	3.3000E-06	4.0000E-06
27	4.0000E-06	9.8770E-06
26	9.8770E-06	1.5968E-05

25	1.5968E-05	2.7700E-05
24	2.7700E-05	4.8052E-05
23	4.8052E-05	7.5501E-05
22	7.5501E-05	1.4873E-04
21	1.4873E-04	3.6726E-04
20	3.6726E-04	9.0690E-04
19	9.0690E-04	1.4251E-03
18	1.4251E-03	2.2395E-03
17	2.2395E-03	3.5191E-03
16	3.5191E-03	5.5300E-03
15	5.5300E-03	9.1180E-03
14	9.1180E-03	1.5030E-02
13	1.5030E-02	2.4780E-02
12	2.4780E-02	4.0850E-02
11	4.0850E-02	6.7340E-02
10	6.7340E-02	1.1100E-01
9	1.1100E-01	1.8300E-01
8	1.8300E-01	3.0250E-01
7	3.0250E-01	5.0000E-01
6	5.0000E-01	8.2100E-01
5	8.2100E-01	1.3530E+00
4	1.3530E+00	2.2310E+00
3	2.2310E+00	3.6790E+00
2	3.6790E+00	6.0655E+00
1	6.0655E+00	2.0000E+01

Appendix B

Heterogeneous BEAVRS Model Parameters

B.1 Material Isotopic Compositions

The isotopic compositions used in the PWR benchmark models in Part IV were identical to those in the BEAVRS PWR model v1.1.1 [62] and are reproduced in the tables below.

Table B.1: Composition of air (0.000616 g/cc).

Nuclide	Density [atom/b-cm]
C-12	6.7565e-09
C-13	7.3076e-11
O-16	5.2864e-06
O-17	2.0137e-09
N-14	1.9681e-05
N-15	7.1900e-08
Ar-36	7.9414e-10
Ar-38	1.4915e-10
Ar-40	2.3506e-07

Table B.2: Composition of borated water (0.740582 g/cc).

Nuclide	Density [atom/b-cm]
H-1	4.9457e-02
H-2	7.4196e-06
B-10	8.0042e-06
B-11	3.2218e-05
O-16	2.4672e-02
O-17	9.3982e-06

Table B.3: Composition of borosilicate glass (2.260000 g/cc).

Nuclide	Density [atom/b-cm]
B-10	9.6506e-04
B-11	3.9189e-03
O-16	4.6511e-02
O-17	1.7717e-05
Al-27	1.7352e-03
Si-28	1.6924e-02
Si-29	8.5977e-04
Si-30	5.6743e-04

Table B.4: Composition of 1.6% enriched UO_2 fuel (10.31341 g/cc).

Nuclide	Density [atom/b-cm]
U-234	3.0131E-06
U-235	3.7503E-04
U-238	2.2625e-02
O-16	4.5895e-02
O-17	1.7482e-05

Table B.5: Composition of 3.1% enriched UO_2 fuel (10.34115 g/cc).

Nuclide	Density [atom/b-cm]
U-234	5.7987e-06
U-235	7.2175e-04
U-238	2.2253e-02
O-16	4.5850e-02
O-17	1.7466e-05

Table B.6: Composition of helium (0.001598 g/cc).

Nuclide	Density [atom/b-cm]
He-4	2.4044e-04

Table B.7: Composition of SS304 stainless steel (8.03 g/cc).

Nuclide	Density [atom/b-cm]
Si-28	9.5274e-04
Si-29	4.8400e-05
Si-30	3.1943e-05
Cr-50	7.6778e-04
Cr-52	1.4806e-02
Cr-53	1.6789e-03
Cr-54	4.1791e-04
Mn-55	1.7604e-03
Fe-54	3.4620e-03
Fe-56	5.4345e-02
Fe-57	1.2551e-03
Fe-58	1.6703e-04
Ni-58	5.6089e-03
Ni-60	2.1605e-03
Ni-61	9.3917e-05
Ni-62	2.9945e-04
Ni-64	7.6261e-05

Table B.8: Composition of zircaloy 4 (6.55 g/cc).

Nuclide	Density [atom/b-cm]
O-16	3.0743e-04
O-17	1.1711e-07
Cr-50	3.2962e-06
Cr-52	6.3564e-05
Cr-53	7.2076e-06
Cr-54	1.7941e-06
Fe-54	8.6699e-06
Fe-56	1.3610e-04
Fe-57	3.1431e-06
Fe-58	4.1829e-07
Zr-90	2.1827e-02
Zr-91	4.7600e-03
Zr-92	7.2758e-03
Zr-94	7.3734e-03
Zr-96	1.1879e-03
Sn-112	4.6735e-06
Sn-114	3.1799e-06
Sn-115	1.6381e-06
Sn-116	7.0055e-05
Sn-117	3.7003e-05
Sn-118	1.1669e-04
Sn-119	4.1387e-05
Sn-120	1.5697e-04
Sn-122	2.2308e-05
Sn-124	2.7897e-05

B.2 Geometric Configuration

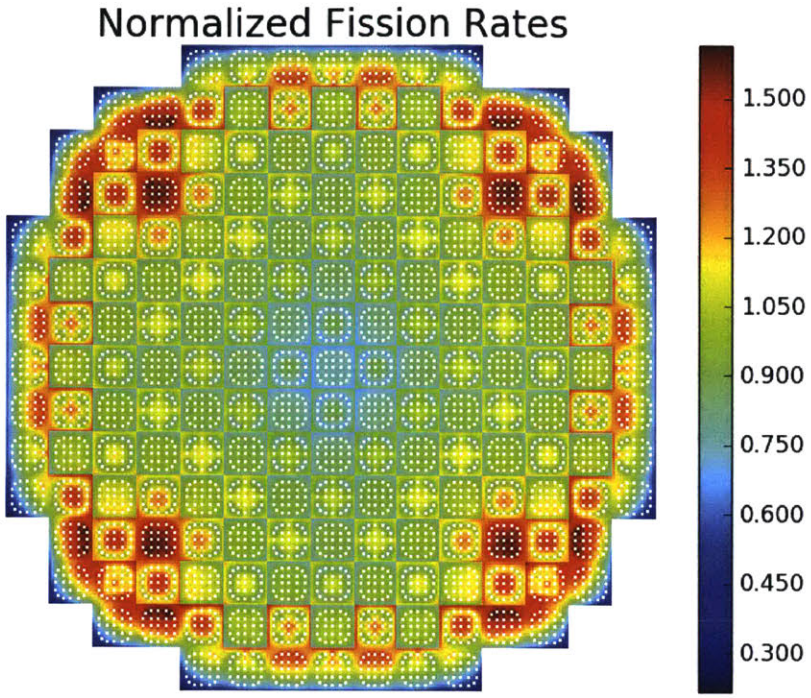
The geometric pin cell parameters for the PWR benchmark models in Part IV were identical to those in the BEAVRS PWR model v1.1.1 [62] at the core axial mid-plane. The pin cell radii are reproduced in the table below for clarity.

Table B.9: Pin cell radii for the BEAVRS model.

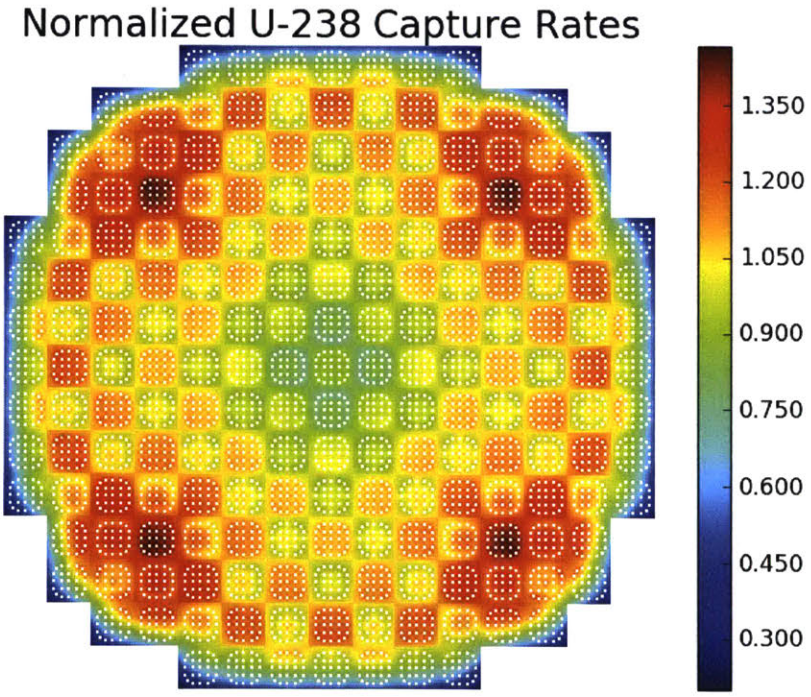
Material	Radius [cm]
Fuel Pin	
Fuel	0.39218
Helium	0.40005
Zircaloy	0.45720
Control Rod Guide Tube	
Borated Water	0.56134
Zircaloy	0.60198
Instrument Tube	
Air	0.43688
Zircaloy	0.48387
Borated Water	0.56134
Zircaloy	0.60198
Burnable Poison	
Air	0.21400
Stainless Steel	0.23051
Air	0.24130
Borosilicate Glass	0.42672
Air	0.43688
Stainless Steel	0.48387
Borated Water	0.56134
Zircaloy	0.60198

B.3 BEAVRS Reaction Rates

The BEAVRS reference reaction rate distributions used in Chap. 7.1 were computed using isotropic in lab scattering in OpenMC to enable comparisons between OpenMC and OpenMOC. The fission and U-238 capture rate spatial distributions are highly skewed with this approximation. In particular, the distributions are significantly more peaked in the assemblies near the corner reflectors since the isotropic-in-lab scattering approximation does not model the preferential streaming and leakage of neutrons through the reflector due to anisotropic scattering in water. The "true" fission and U-238 capture rate spatial distributions computed using normal anisotropic scattering in OpenMC are illustrated below for comparison purpose, and as a reminder of the necessity for accurate models of higher order scattering in high-fidelity full-core deterministic transport calculations.



(a)



(b)

Figure B-1: Fission rates (a) and U-238 capture rates (b) for the 2D quarter core BEAVRS model tallied using a pin-wise mesh in OpenMC with anisotropic scattering.

Appendix C

Quantification of Spatial Self-Shielding Effects

This section provides figures illustrating the impacts of the infinite, null and spatial homogenization as an addendum to Chap. 8. The percent relative fission and U-238 capture rate errors between OpenMC and OpenMOC for each homogenization scheme and benchmark for 2, 8 and 70 energy groups are illustrated in Sec. C.1 and Sec. C.2, respectively. Sec. C.3 illustrates the U-238 capture rate absolute errors for the 2×2 colorset with reflector and quarter core BEAVRS benchmarks.

C.1 Fission Rate Relative Errors

The percent relative fission rate errors between OpenMC and OpenMOC for each homogenization scheme and benchmark for 2, 8 and 70 energy groups are shown in the following figures. These heatmaps complement Figs. 8-4 to 8-9 by illustrating the error distributions for solutions computed by OpenMOC with 2-group MGXS.

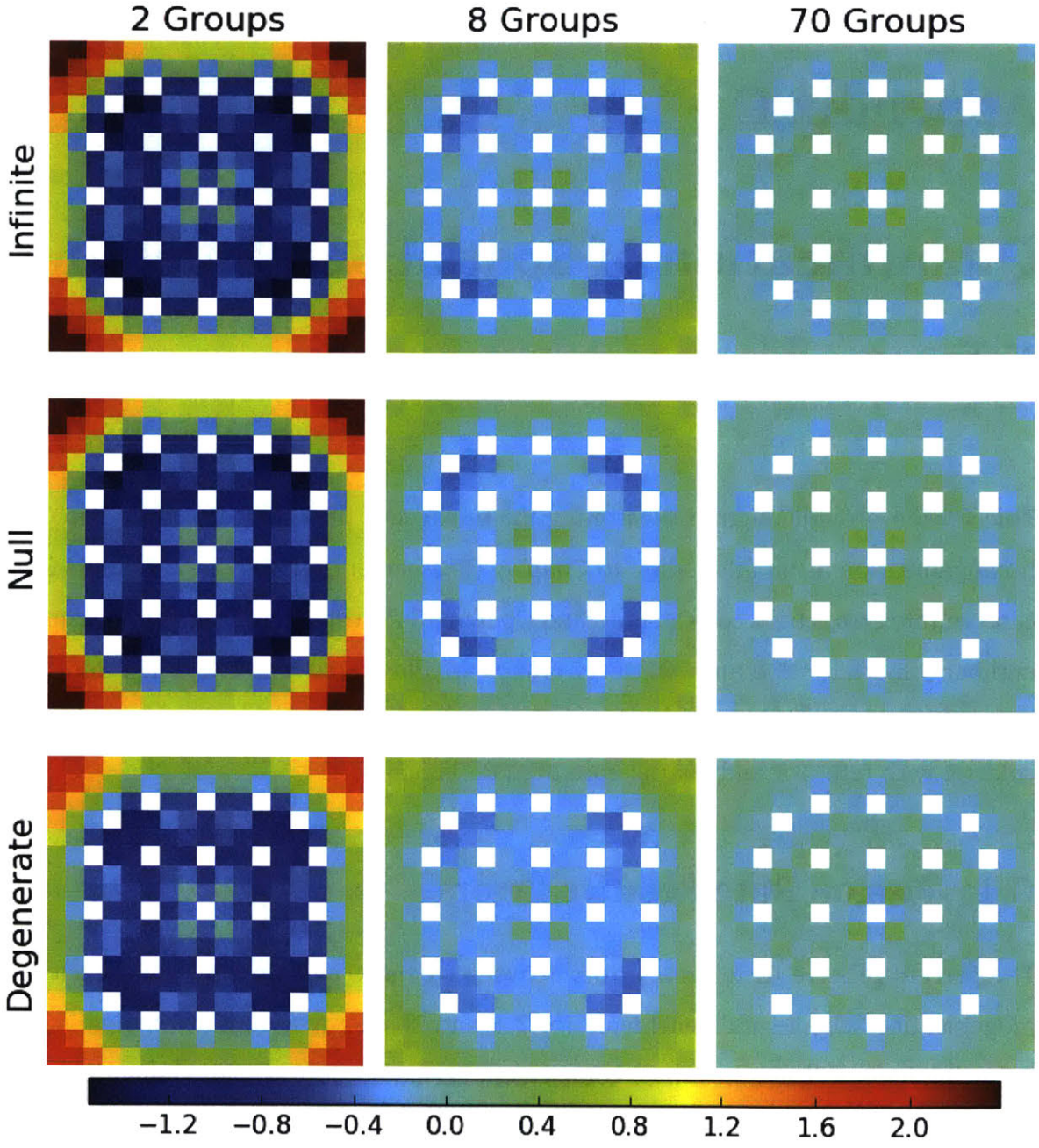


Figure C-1: Fission rate percent relative errors for a 1.6% enriched assembly corresponding to the reference in Fig. 7-9.

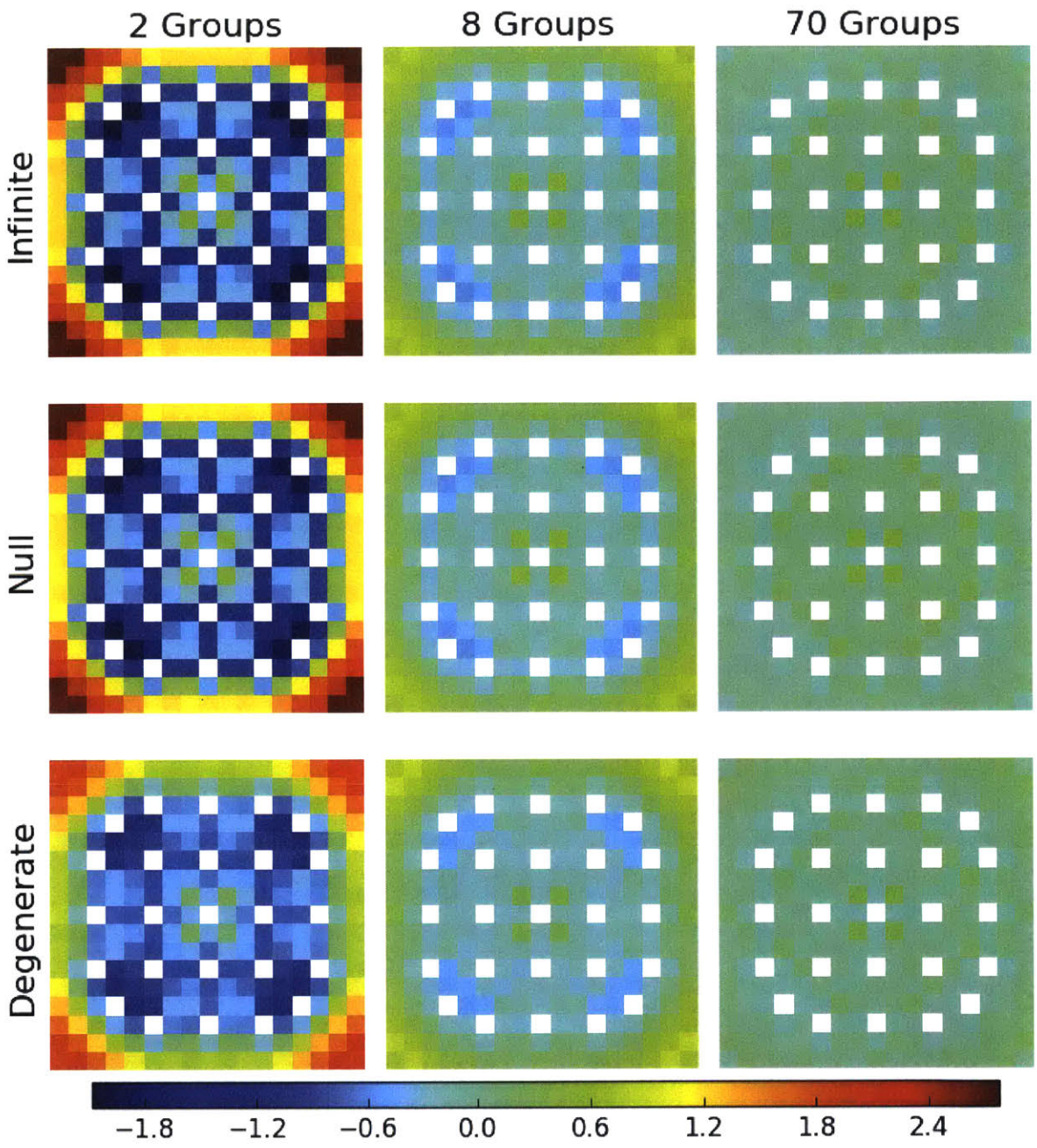


Figure C-2: Fission rate percent relative errors for a 3.1% enriched assembly corresponding to the reference in Fig. 7-10.

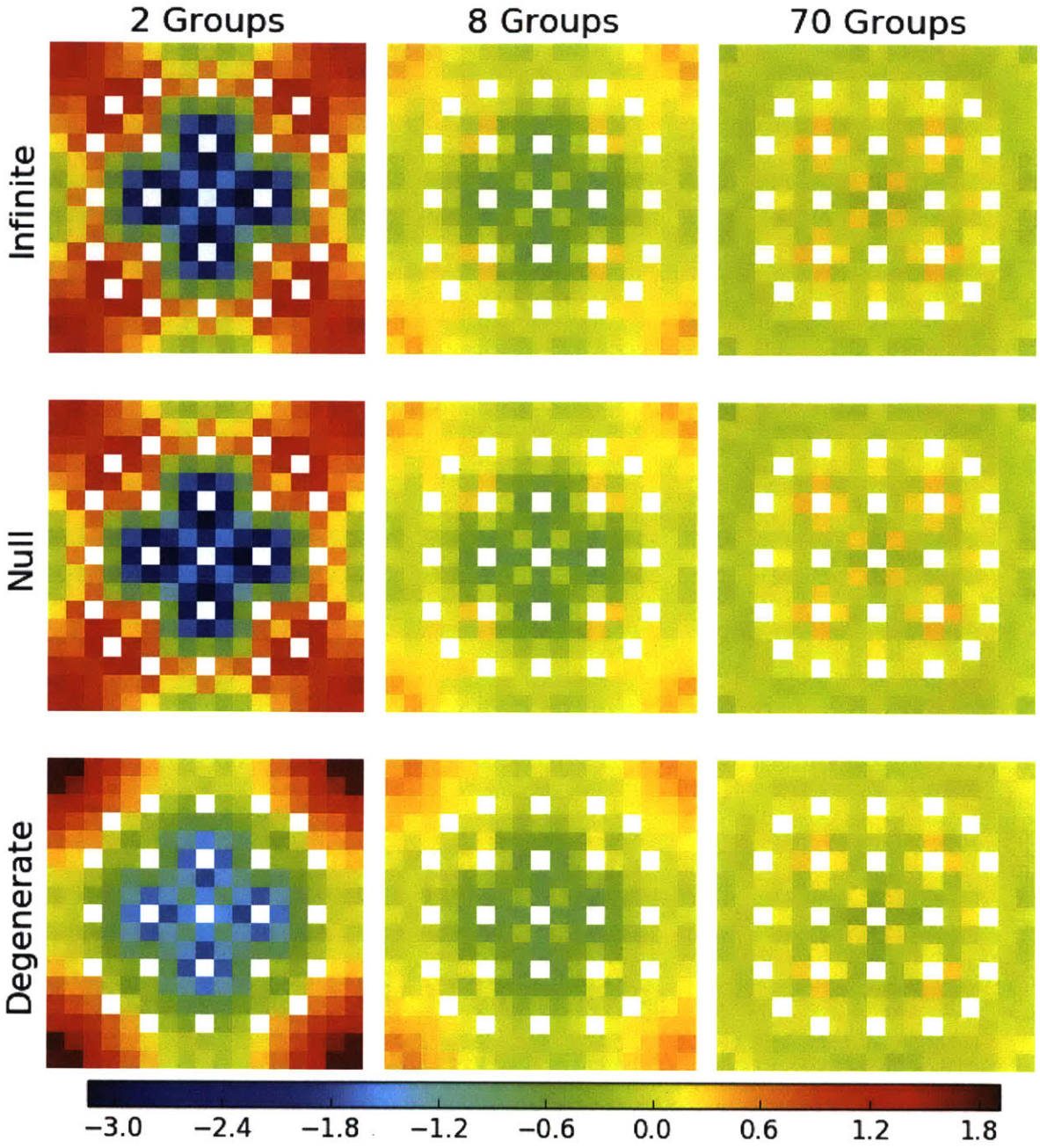


Figure C-3: Fission rate percent relative errors for a 3.1% enriched assembly with 20 BPs corresponding to the reference in Fig. 7-11.

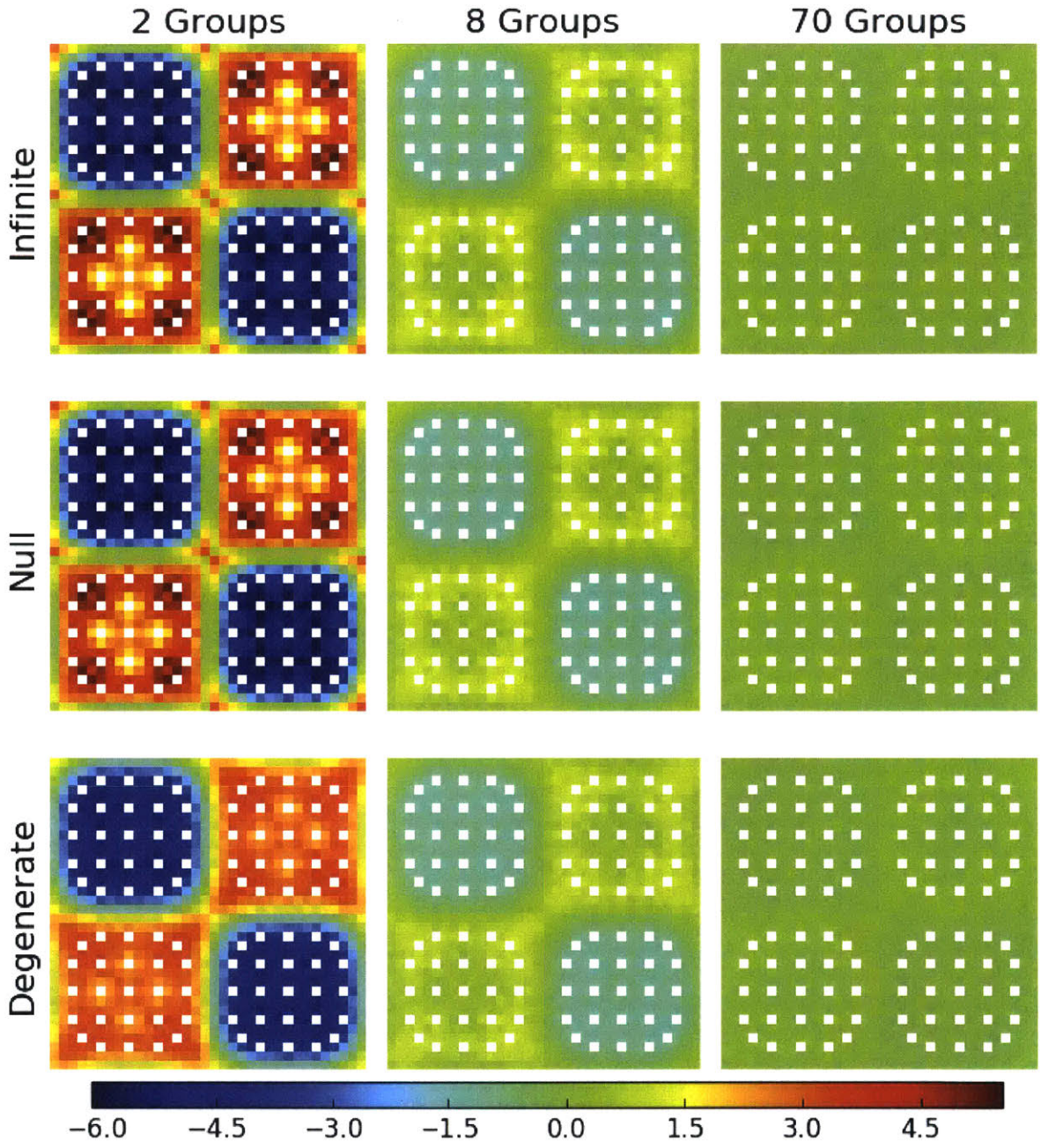


Figure C-4: Fission rate percent relative errors for a 2×2 colorset corresponding to the reference in Fig. 7-12.

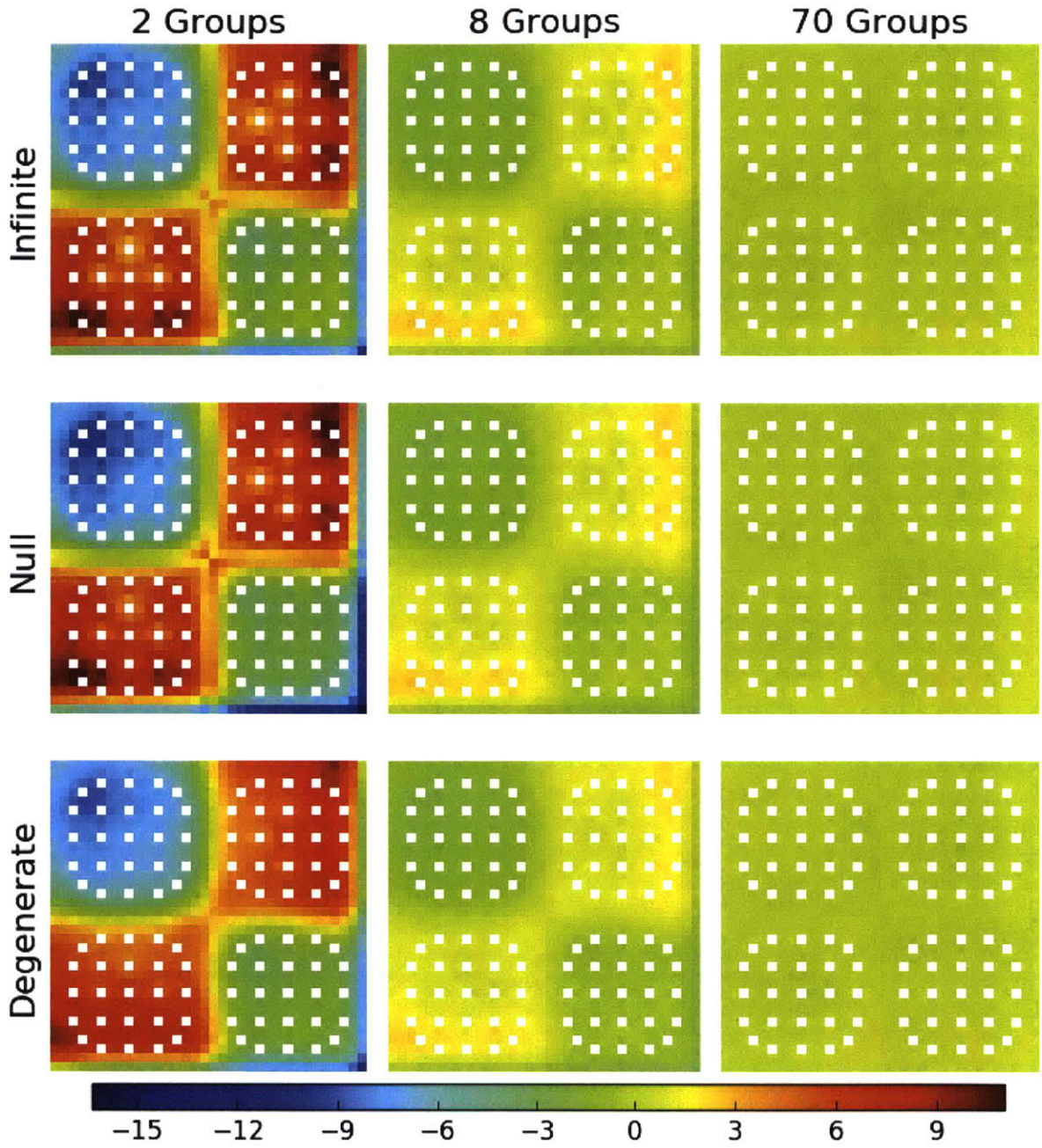


Figure C-5: Fission rate percent relative errors for a 2×2 colorset with a reflector corresponding to the reference in Fig. 7-13.

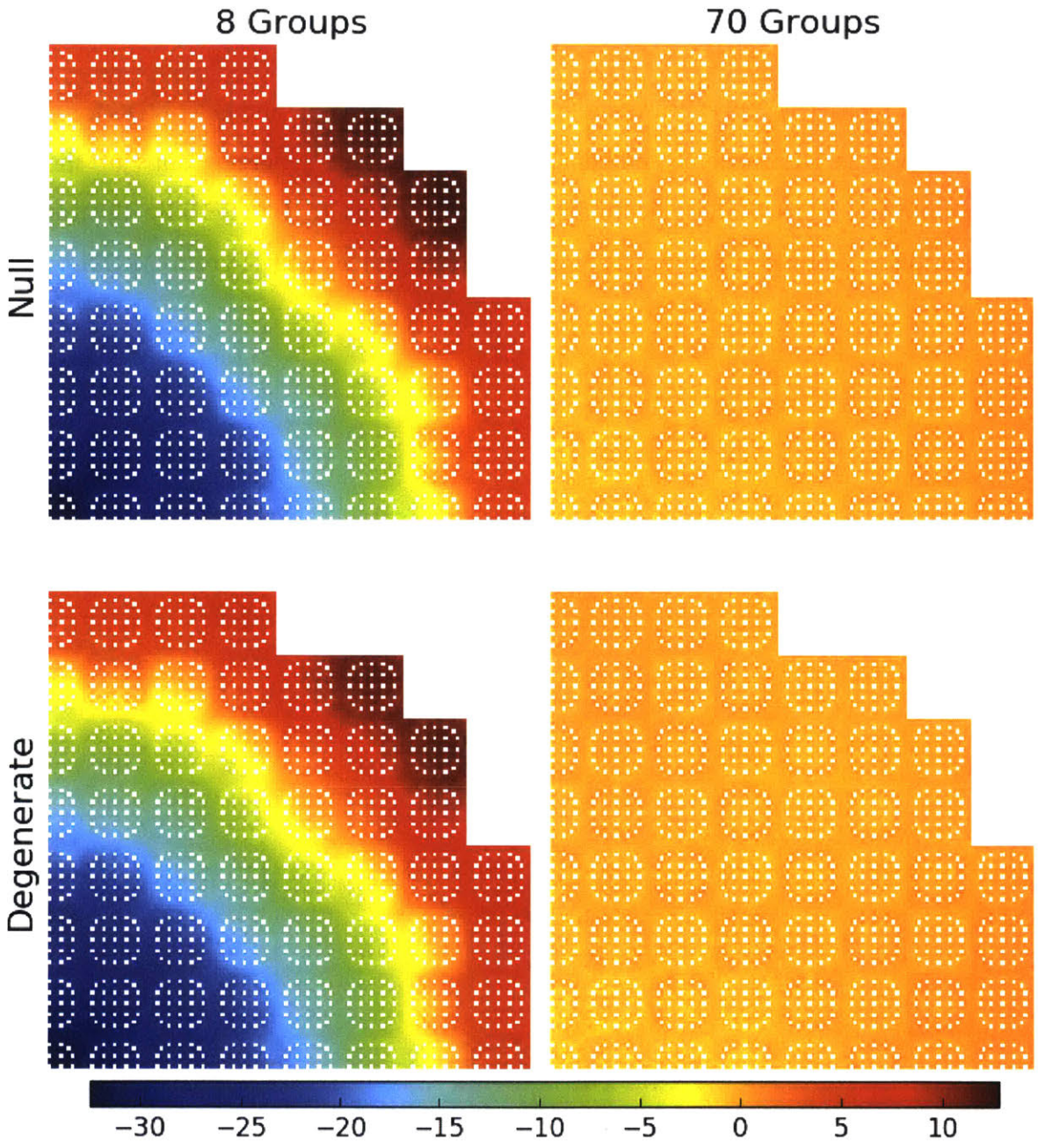


Figure C-6: Fission rate percent relative errors for the quarter core BEAVRS model corresponding to the reference in Fig. 7-14.

C.2 U-238 Capture Rate Relative Errors

The percent relative U-238 capture rate errors between OpenMC and OpenMOC for each spatial homogenization scheme and benchmark for 2, 8 and 70 energy groups are shown in the following figures. These heatmaps complement Figs. 8-10 to 8-15 by illustrating the error distributions for solutions computed by OpenMOC with 2-group MGXS.

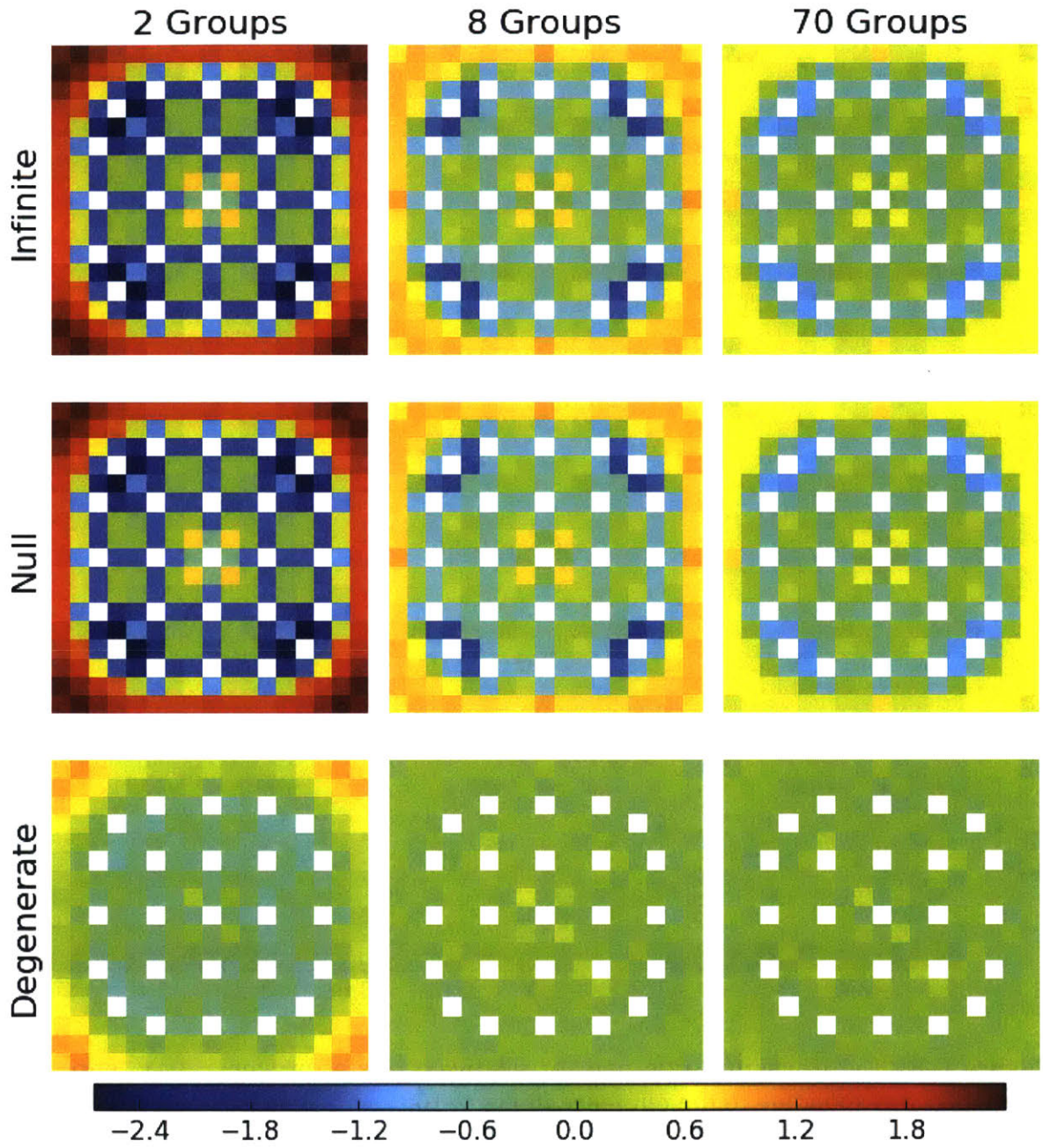


Figure C-7: U-238 capture rate percent relative errors errors for a 1.6% enriched assembly corresponding to the reference in Fig. 7-15.

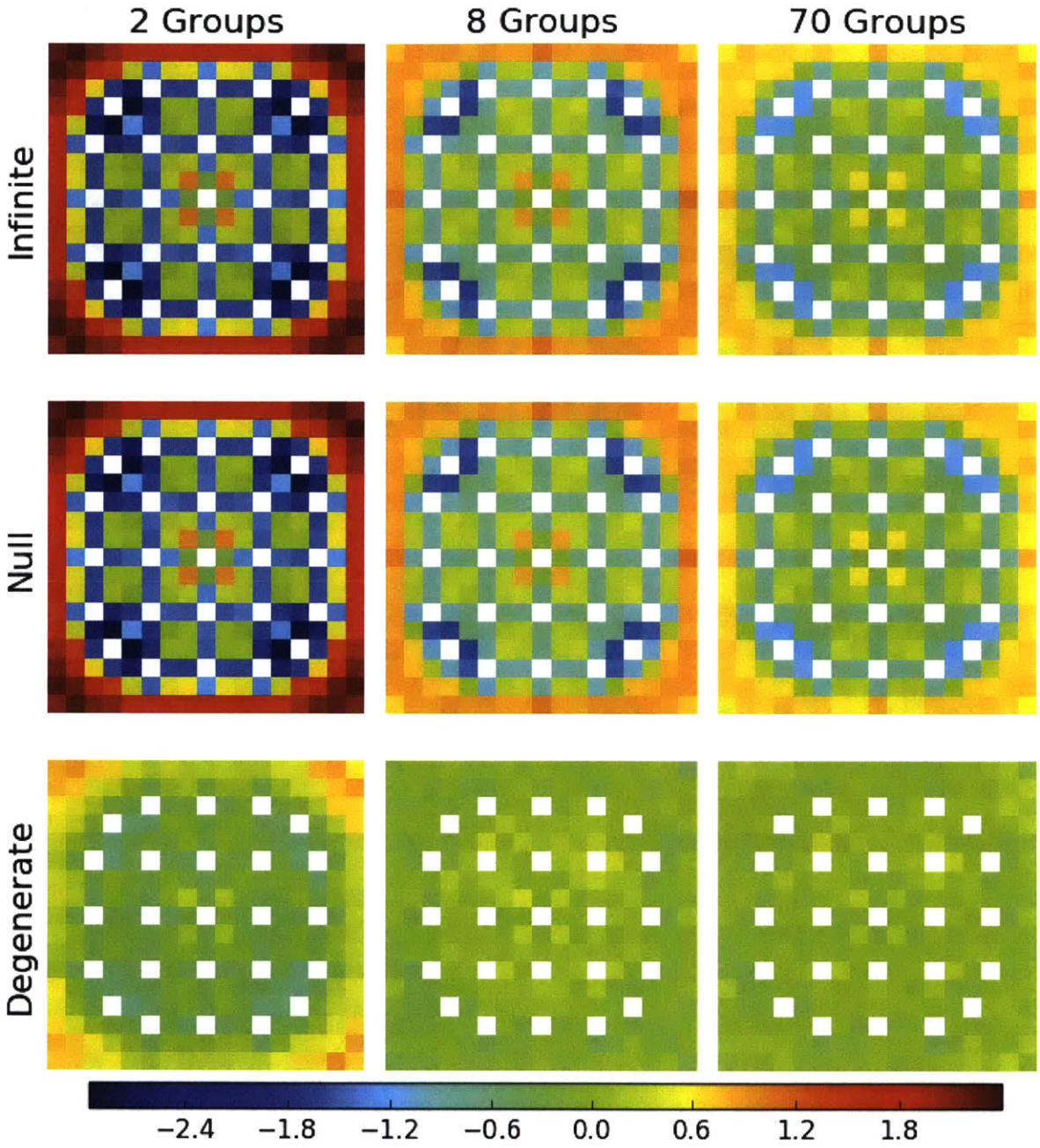


Figure C-8: U-238 capture rate percent relative errors errors for a 3.1% enriched assembly corresponding to the reference in Fig. 7-16.

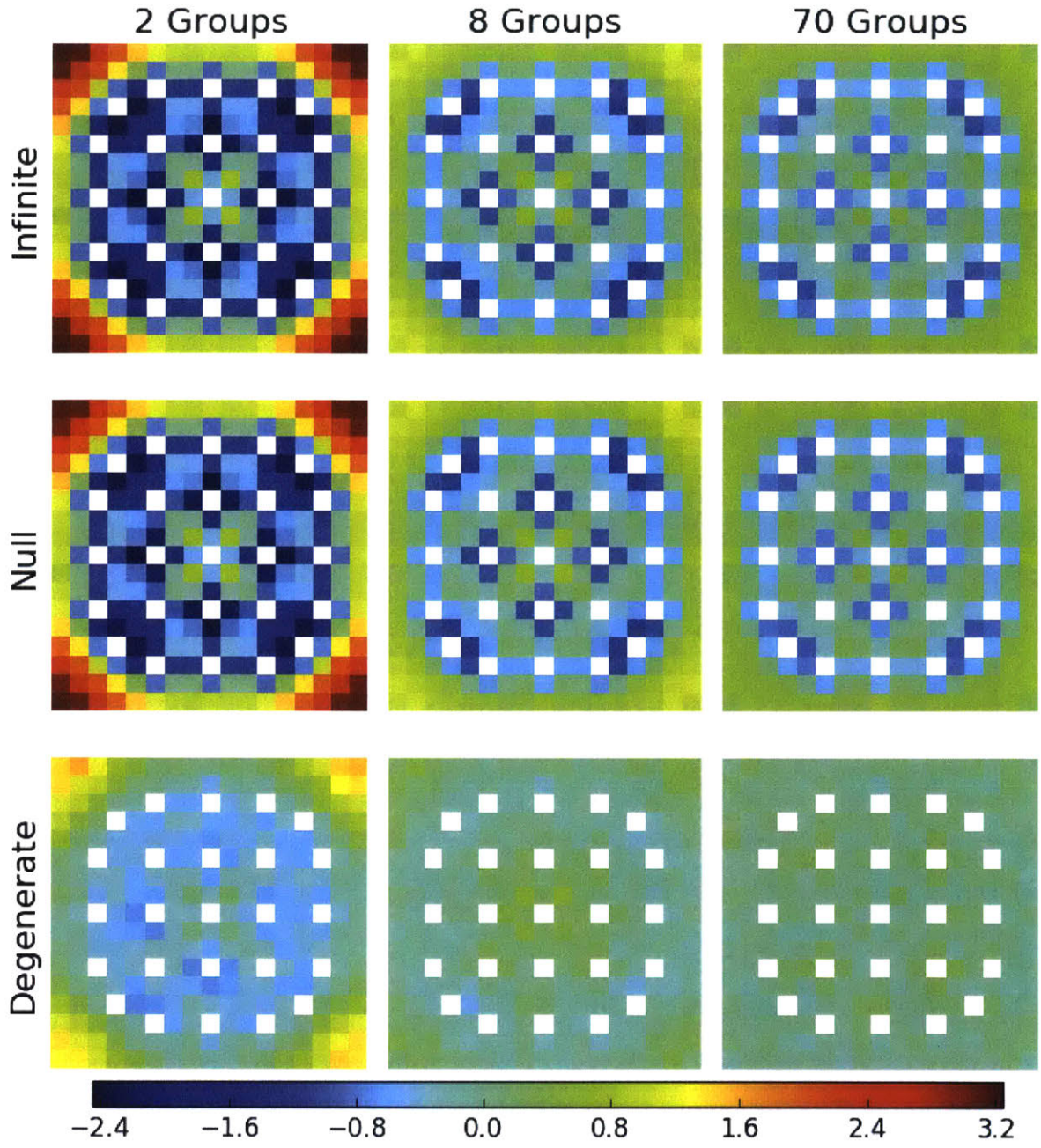


Figure C-9: U-238 capture rate percent relative errors errors for a 3.1% enriched assembly with 20 BPs corresponding to the reference in Fig. 7-17.

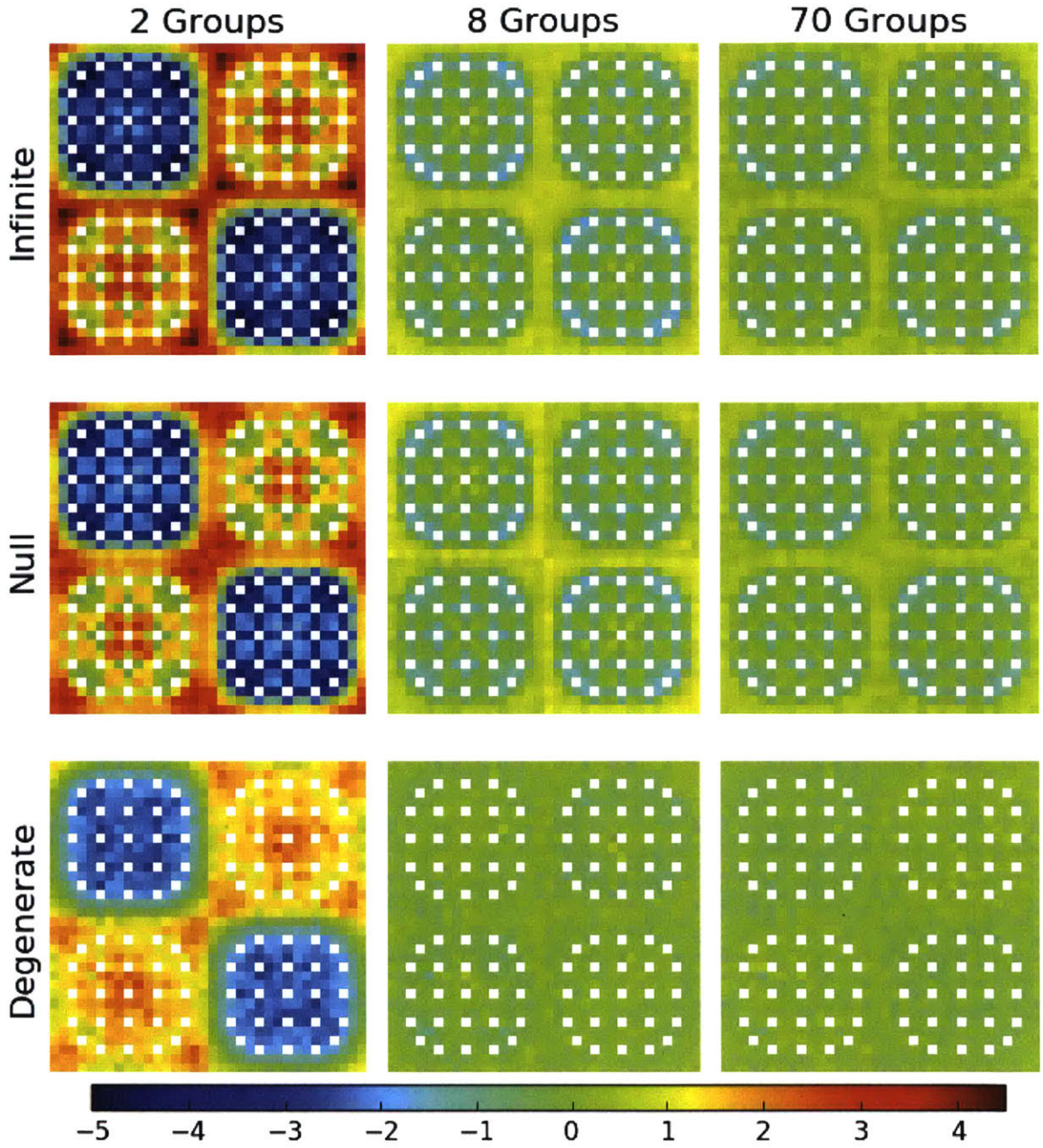


Figure C-10: U-238 capture rate percent relative errors errors for a 2×2 colorset corresponding to the reference in Fig. 7-18.

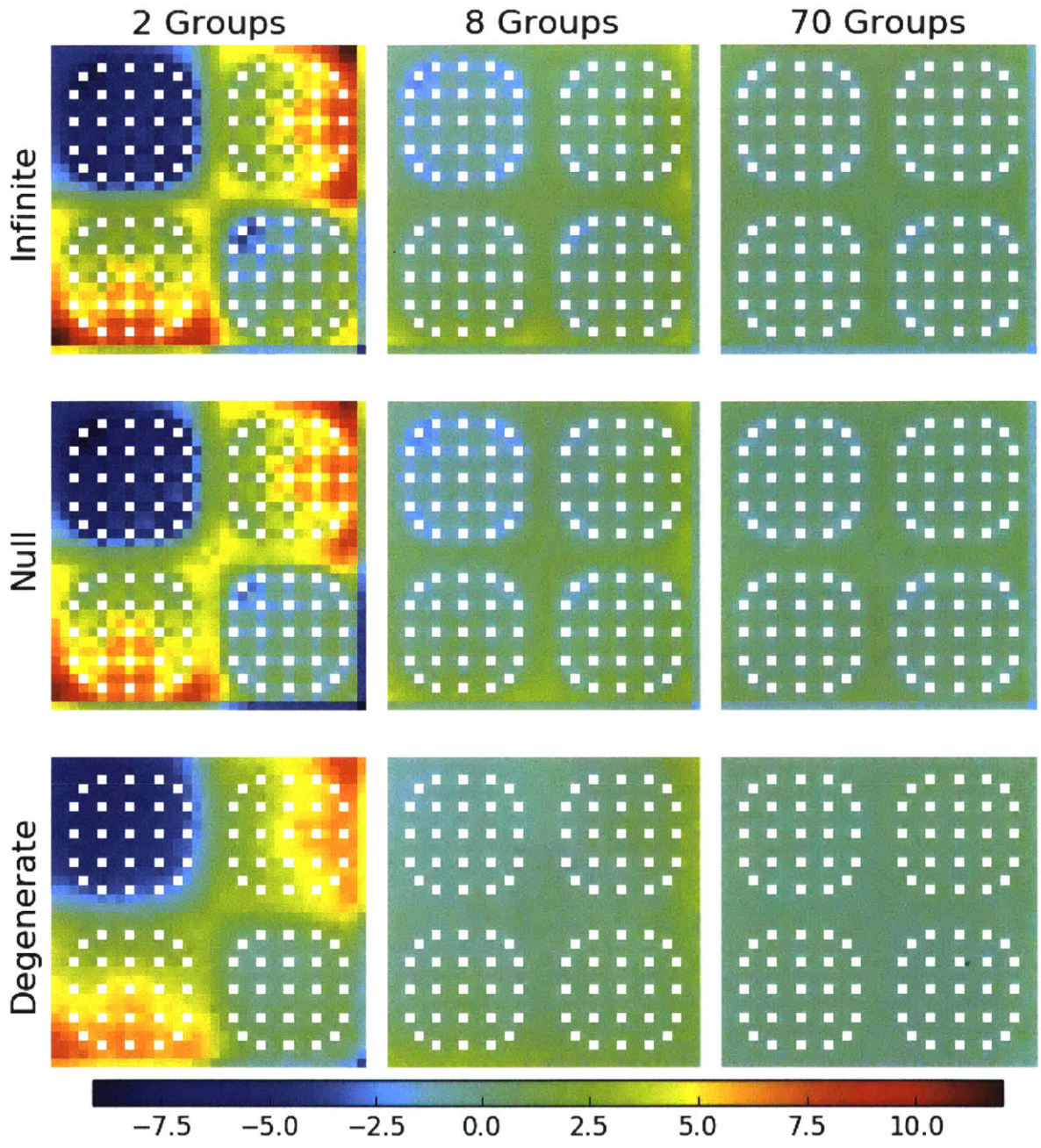


Figure C-11: U-238 capture percent relative errors rate errors for a 2×2 colorset with a reflector corresponding to the reference in Fig. 7-19.

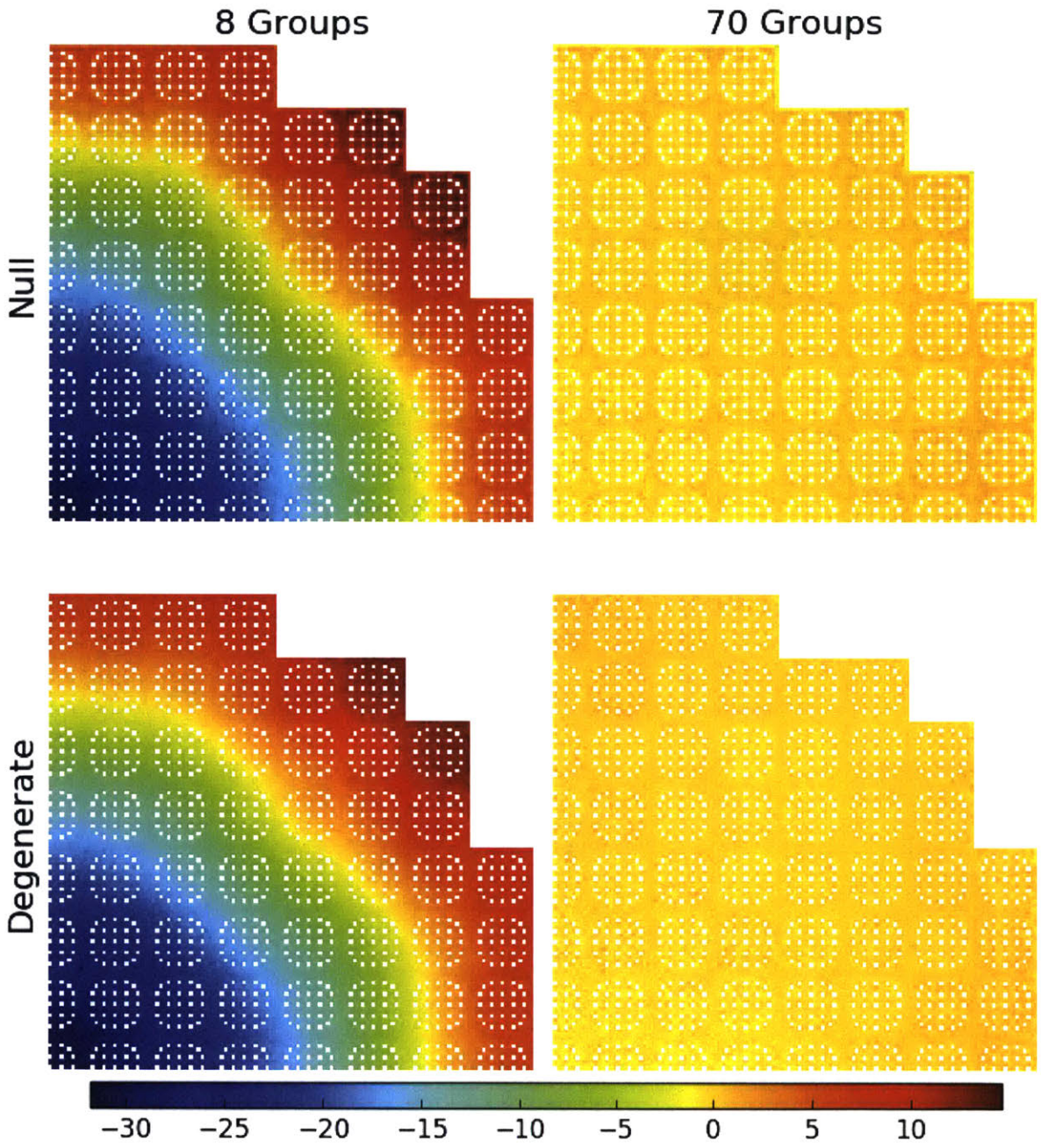


Figure C-12: U-238 capture percent relative errors rate errors for the 2D quarter core BEAVRS model corresponding to the reference in Fig. 7-20.

C.3 Capture Rate Absolute Errors

The fission and U-238 reaction rates vary much more across pins for benchmarks with leakage due to vacuum boundary conditions (BCs), such as the 2×2 colorset and quarter core BEAVRS models. In these models, it can be useful to investigate the absolute reaction rate error in addition to the percent relative error. The reason for this is that it is important to reduce the error in those pins in which the reaction rates are largest and hence the most limiting to reactor performance (*i.e.*, the hottest fuel pins). The absolute error spatial distributions better illustrate improved reaction rate predictions in the pins that matter most in these benchmark models.

Figs. C-13 and C-14 show the U-238 capture rate absolute errors between OpenMC and OpenMOC for the null and degenerate spatial homogenizations with 8 and 70 energy groups for the 2×2 colorset benchmarks. Figs. C-15 to C-17 show the U-238 capture rate absolute errors for the null, degenerate and LNS schemes for 70 energy groups. These heatmaps complement the percent relative errors in Figs. 8-14 to 8-15. The absolute errors are not shown for the benchmarks with all reflective BCs since they very closely mirror the percent relative error distributions.

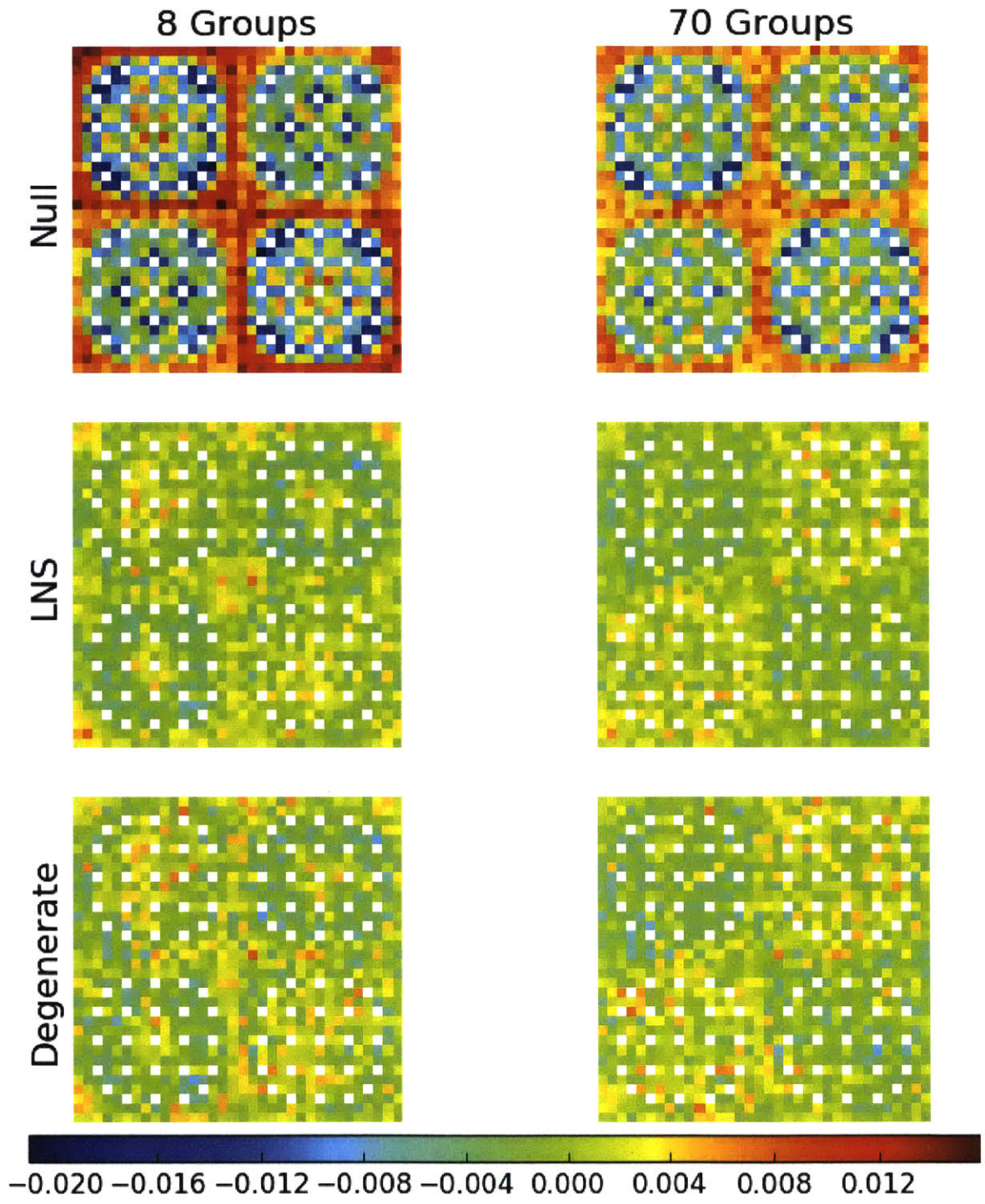


Figure C-13: U-238 capture rate absolute errors for a 2×2 colorset corresponding to the reference in Fig. 7-18.

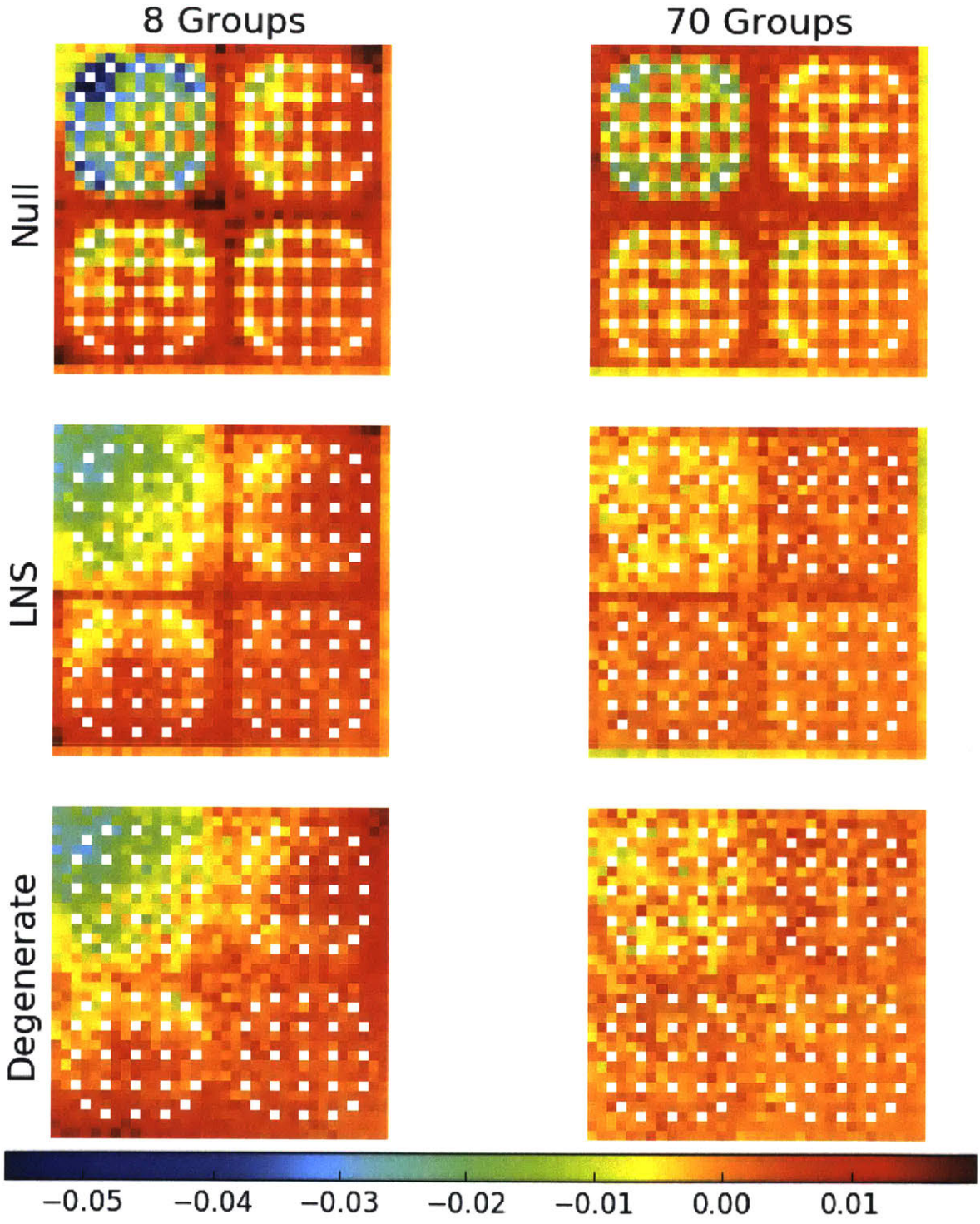


Figure C-14: U-238 capture rate absolute errors for a 2×2 colorset with a reflector corresponding to the reference in Fig. 7-19.

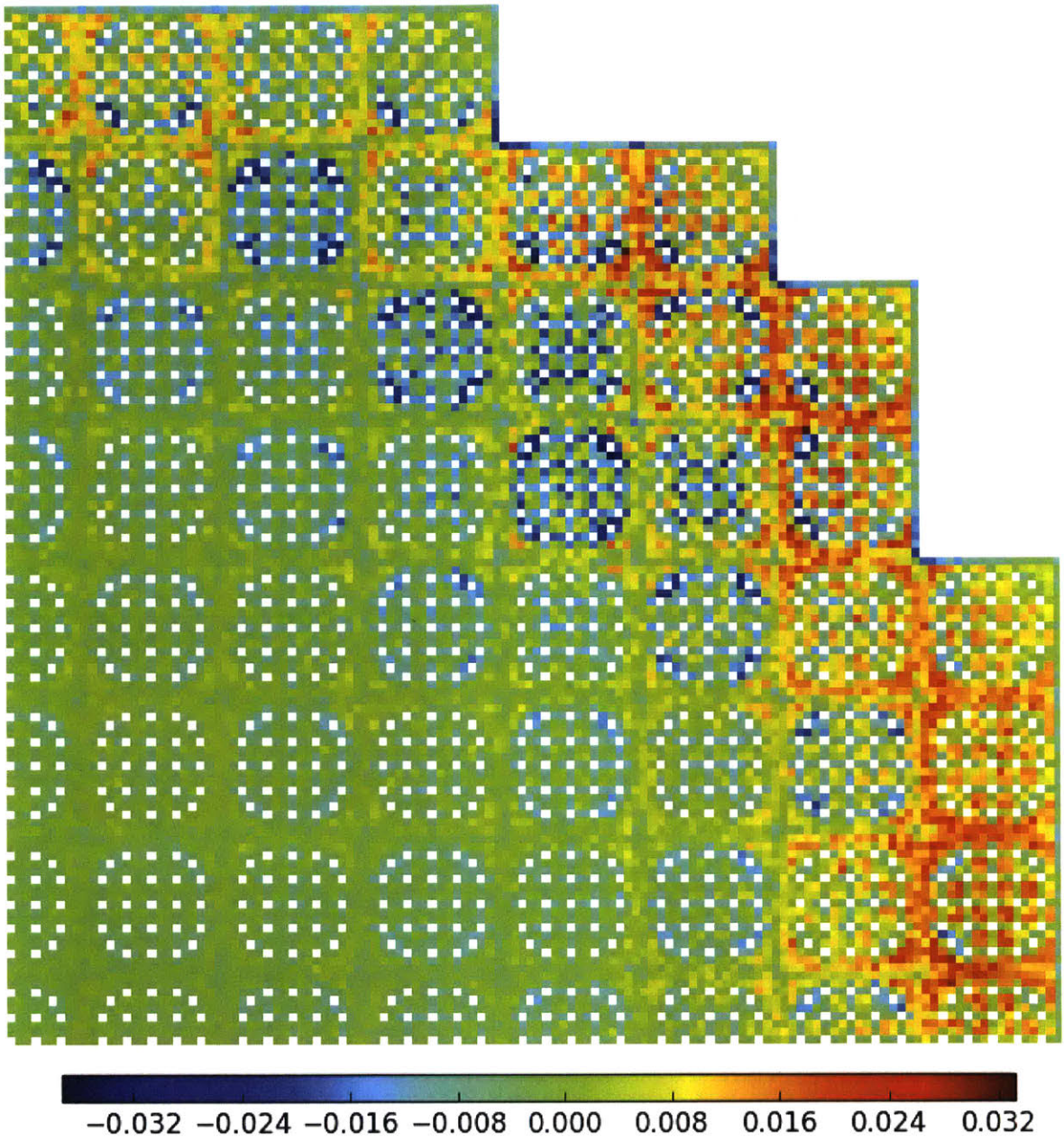


Figure C-15: U-238 capture rate absolute errors for the 2D quarter core BEAVRS model with null spatial homogenization corresponding to the reference in Fig. 7-20.

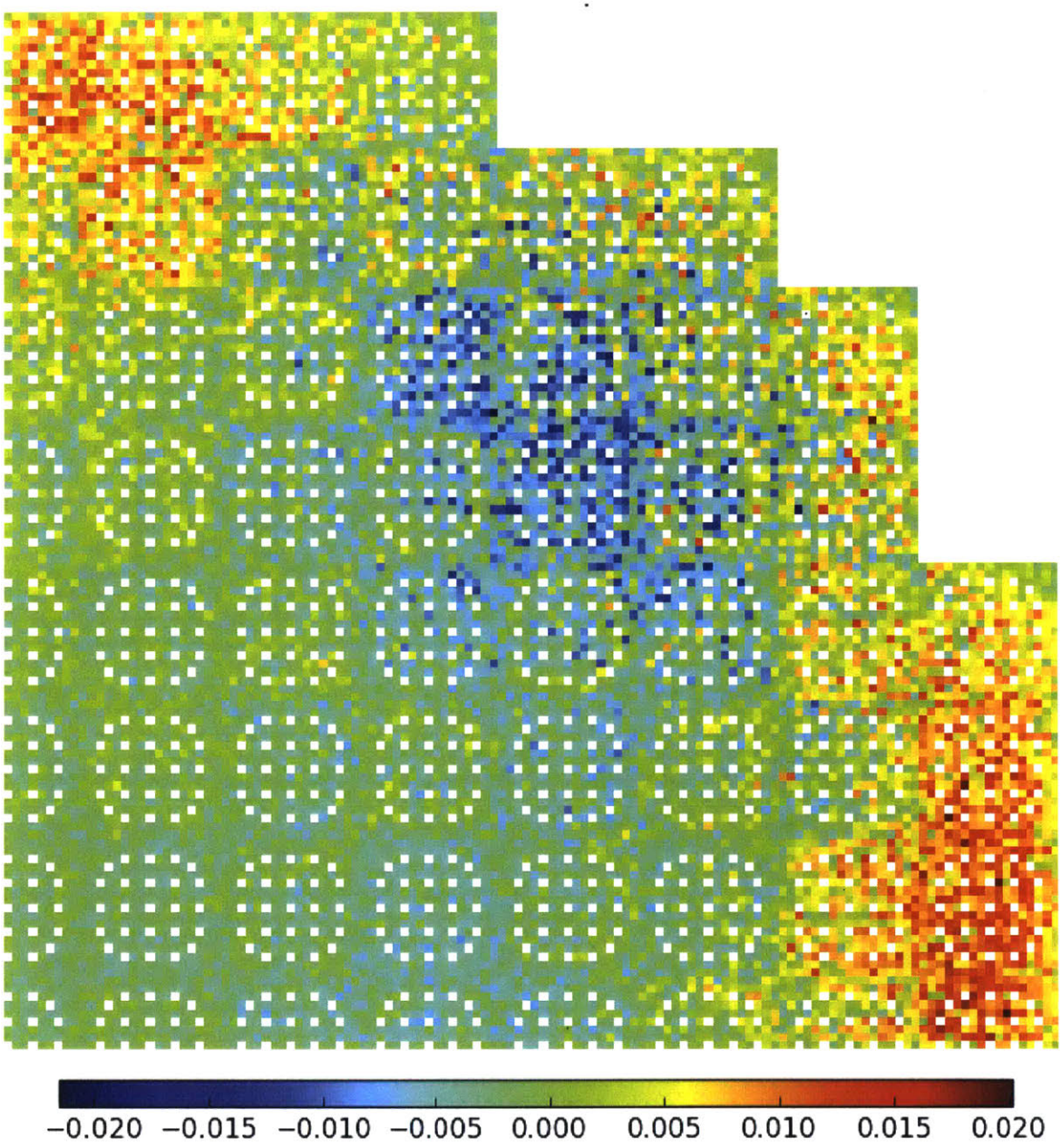


Figure C-16: U-238 capture rate absolute errors for the 2D quarter core BEAVRS model with degenerate spatial homogenization corresponding to the reference in Fig. 7-20.

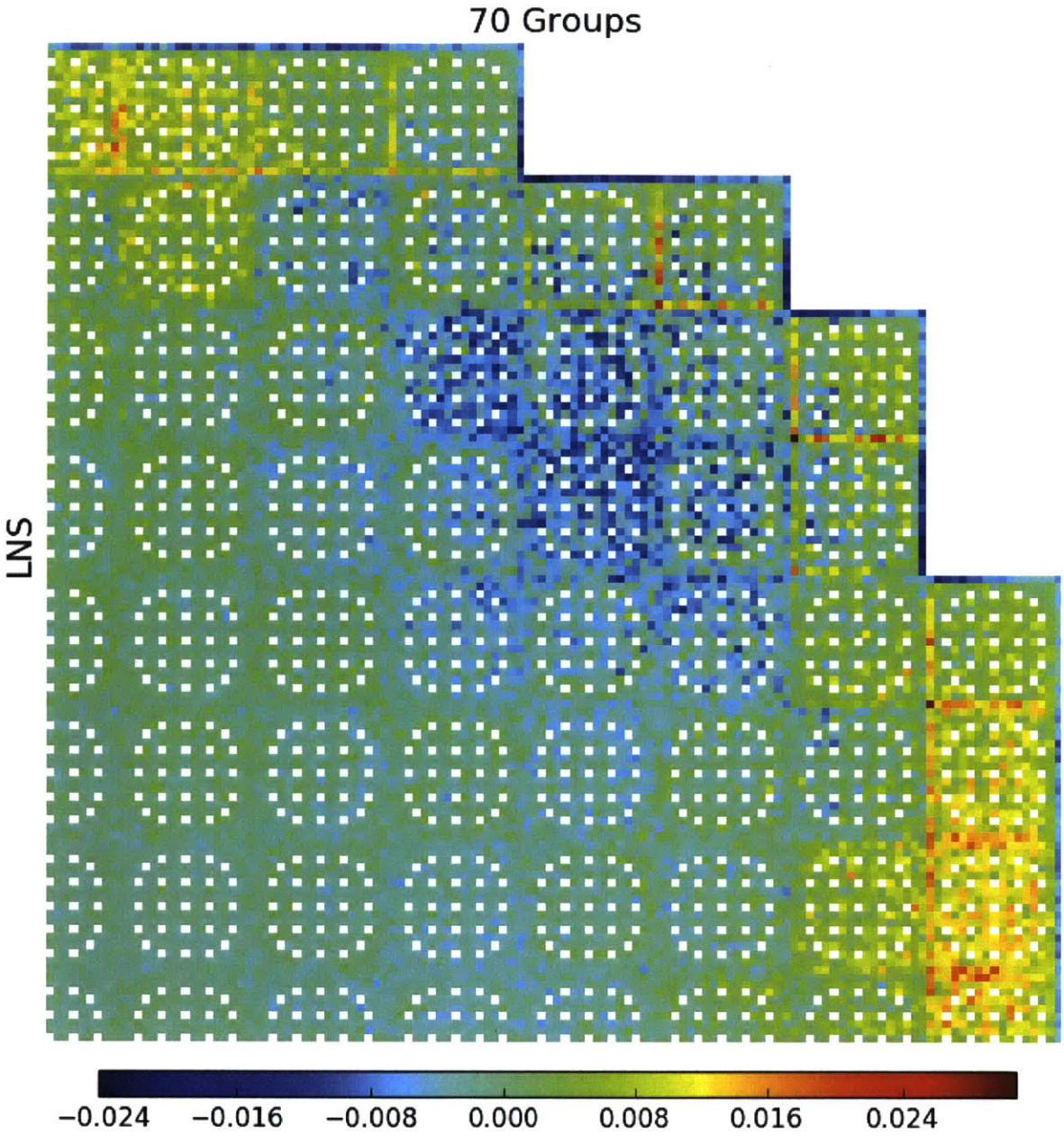


Figure C-17: U-238 capture rate absolute errors for the 2D quarter core BEAVRS model with LNS spatial homogenization corresponding to the reference in Fig. 7-20.

Appendix D

Clustered Geometries with *i*MGXS Spatial Homogenization

The *i*MGXS spatial homogenization scheme is presented in Chap. 10 and a number of example "clustered geometries" generated with the scheme are highlighted in Sec. 10.8. The *i*MGXS scheme is highly configurable, and various methods may be employed for feature extraction, feature selection and dimensionality reduction which greatly impact the resultant materials configuration found by a clustering algorithm. The geometries in Sec. 10.8 were generated using the agglomerative clustering algorithm (Sec. 10.5.2) with datasets filtered by pinch and litmus-only feature selection (Secs. 10.3.5 and 10.8). The importance of using non-MGXS features is illustrated in Sec. D.1, while Sec. D.2 demonstrates the impact of using dimensionality reduction techniques.

D.1 Clustering without Features

The different types of features considered in this thesis are presented in Sec. 10.2. Since the objective of using unsupervised clustering algorithms in the *i*MGXS scheme is to identify the clustering of pin-wise MGXS – and to apply this information in the spatial homogenization process – it may be useful to consider whether the non-MGXS

features[114] are necessary or whether the MGXS can simply be clustered by themselves. The clustered geometries presented in Sec. 10.8 were found by clustering both MGXS and non-MGXS features. This section illustrates the geometries that are found when only the pin-wise MGXS means and uncertainties are selected as features by the *i*MGXS data processing pipeline.

The clustered geometries for the individual fuel assembly and 2×2 colorset benchmarks are shown in Figs. D-1 to D-4. Likewise, Figs. D-5 to D-6 highlight the clustered geometries for the quarter core BEAVRS model. Similar to the analysis in Sec. 10.8, the agglomerative clustering algorithm (Sec. 10.5.2) is employed with the same 2-group MC tally datasets used in Sec. 10.8. Both pinch and litmus-only feature selection techniques were used to select the "best" reaction type for each pair of nuclides and energy groups, for each nuclide in the fuel and both energy groups. No dimensionality reduction techniques (Sec. 10.4) were applied to the selected features.

The most important insight that can be drawn from the figures is that the clustering models are much "noisier" than those found when the non-MGXS features are considered in Sec. 10.8. This illustrates the importance of using some features which may better reveal the effects of clustering in MC tally data due to spatial self-shielding effects than the pin-wise MGXS themselves. In addition, some of the non-MGXS features are energy-integrated and/or summed across nuclides (*e.g.*, fractional reactivities) and will necessarily have smaller statistical uncertainties than microscopic MGXS[115].

[114]The fractional reactivities (Sec. 10.2.2), spectral indices (Sec. 10.2.3) and reaction fractions (Sec. 10.2.4) are examples of non-MGXS features.

[115]Although this is true in general, it should not make a difference here since the MC tally data used in this analysis was very highly converged.

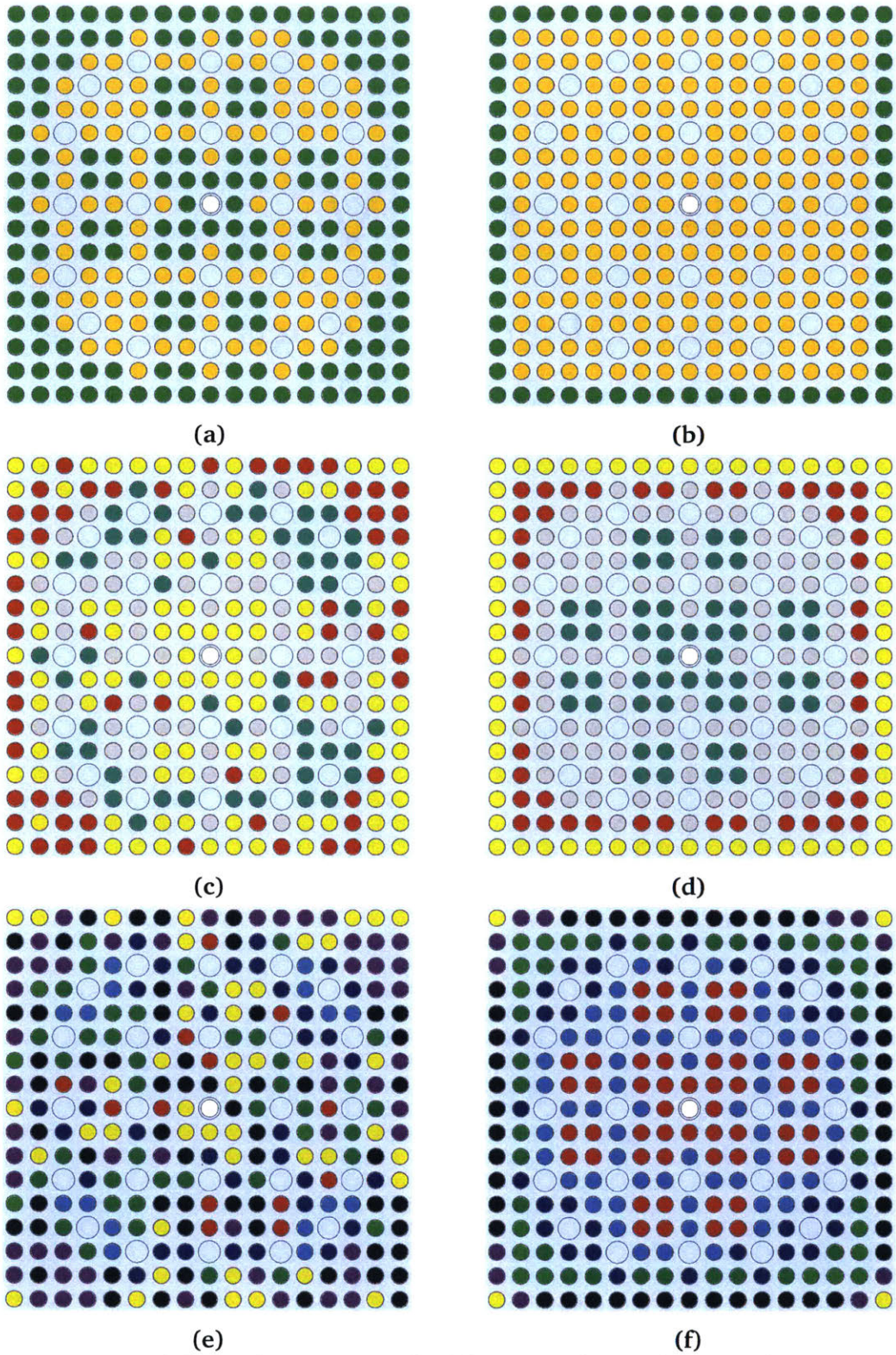


Figure D-1: Materials for the 1.6% enriched fuel assembly with *i*MGXS homogenization without non-MGXS features. The materials for 2, 4, and 8 clusters are illustrated in (a), (c) and (e) for pinch feature selection, and in (b), (d) and (f) for litmus-only feature selection, respectively.

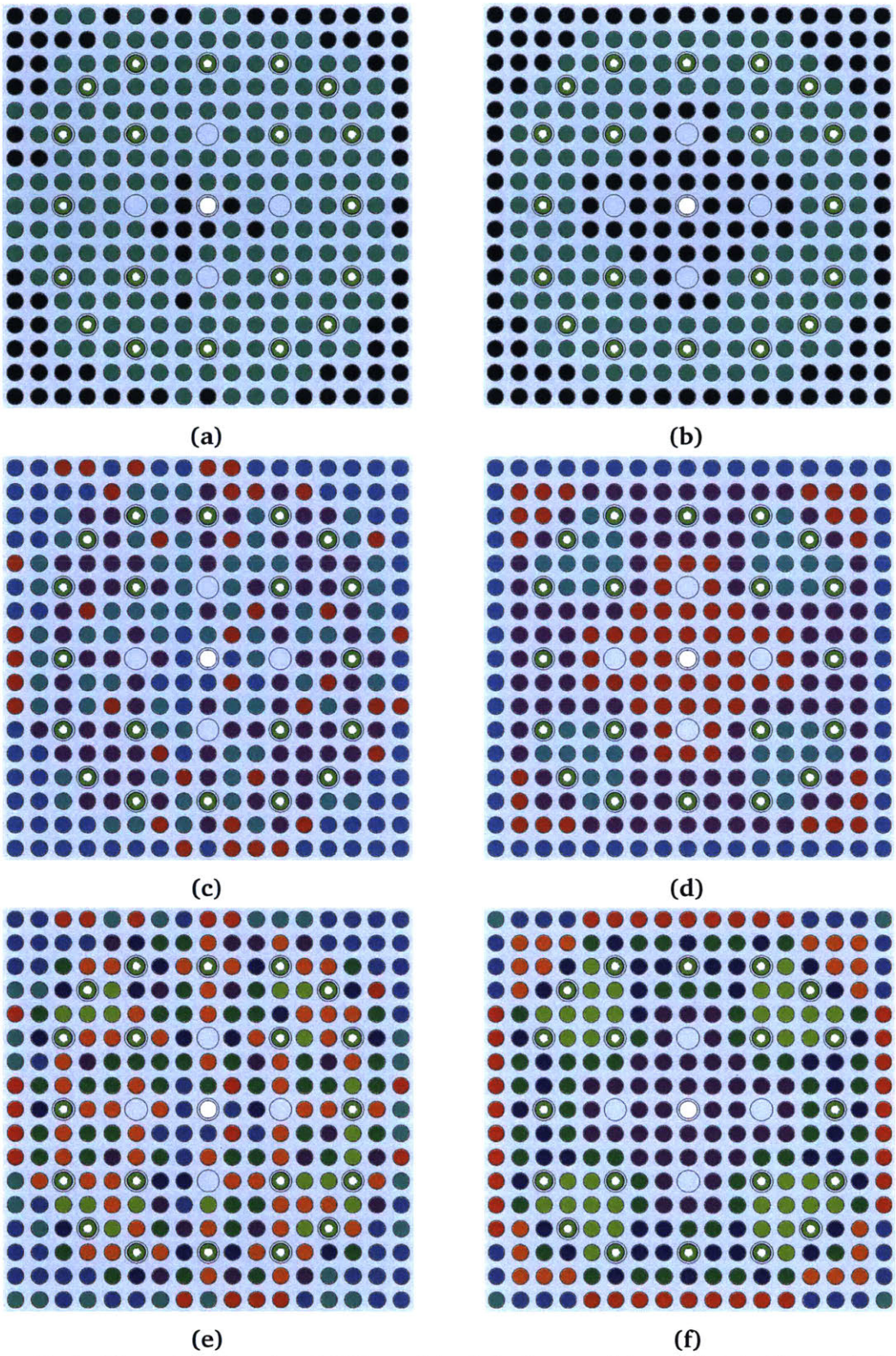


Figure D-2: Materials for the 3.1% enriched fuel assembly with 20 BPs with *i*MGXS homogenization without non-MGXS features. The materials for 2, 4, and 8 clusters are illustrated in (a), (c) and (e) for pinch feature selection, and in (b), (d) and (f) for litmus-only feature selection, respectively.

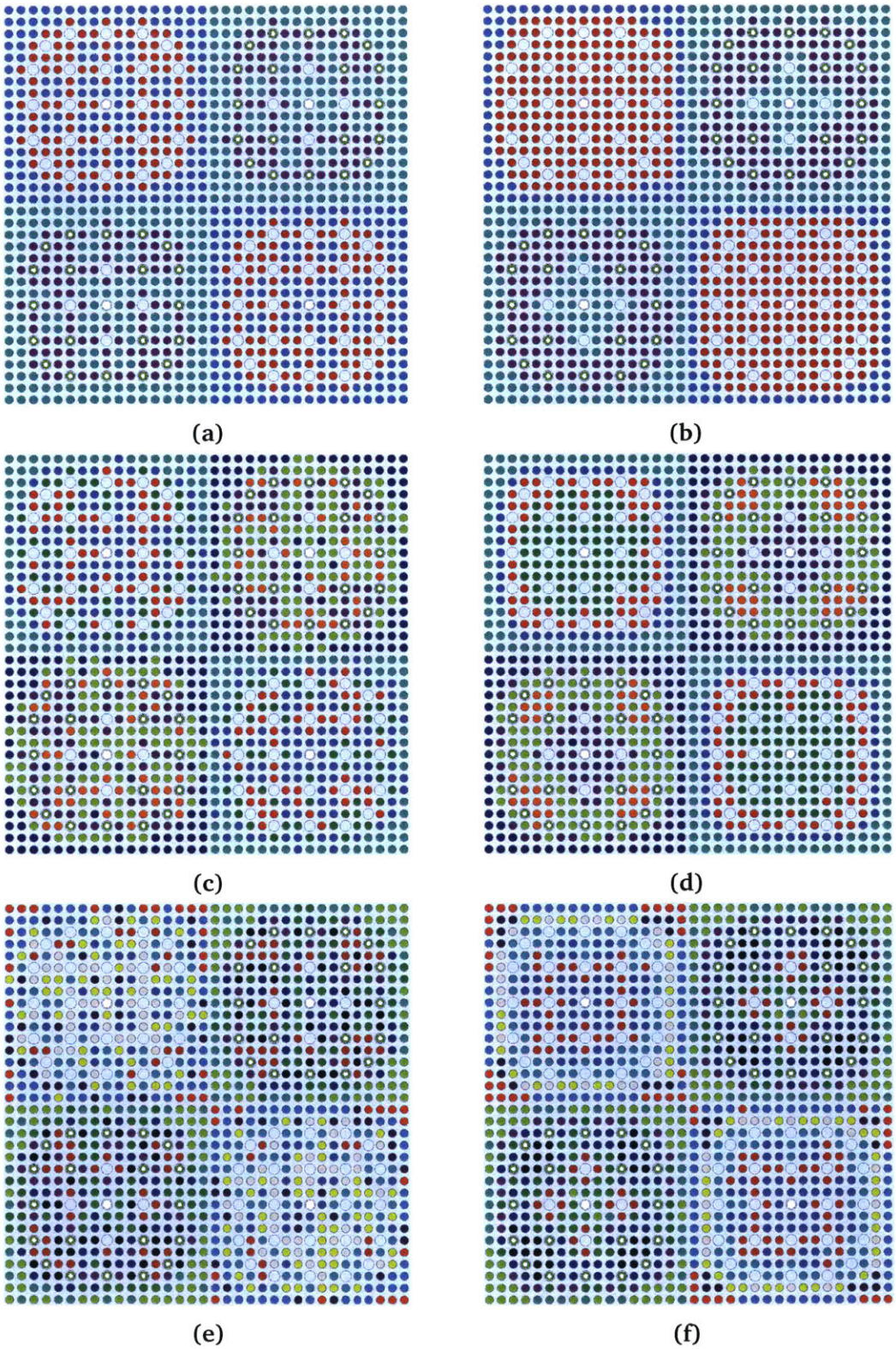


Figure D-3: Materials for the 2×2 colorset with *i*MGXS homogenization without non-MGXS features. The materials for 2, 4, and 8 clusters are illustrated in (a), (c) and (e) for pinch feature selection, and in (b), (d) and (f) for litmus-only feature selection, respectively.

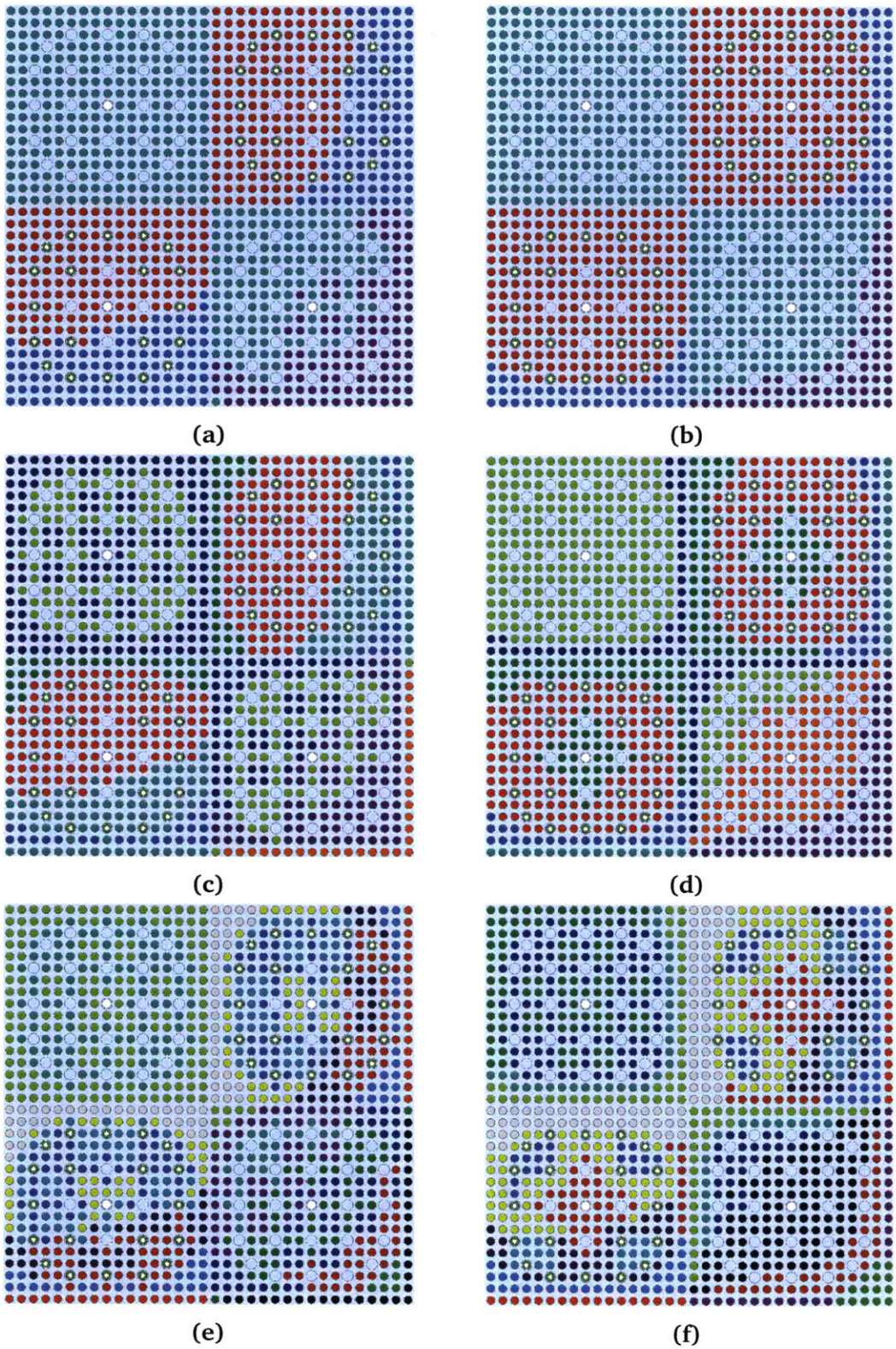


Figure D-4: Materials for the 2×2 colorset with water reflector with *i*MGXS homogenization without non-MGXS features. The materials for 2, 4, and 8 clusters are illustrated in (a), (c) and (e) for pinch feature selection, and in (b), (d) and (f) for litmus-only feature selection, respectively.

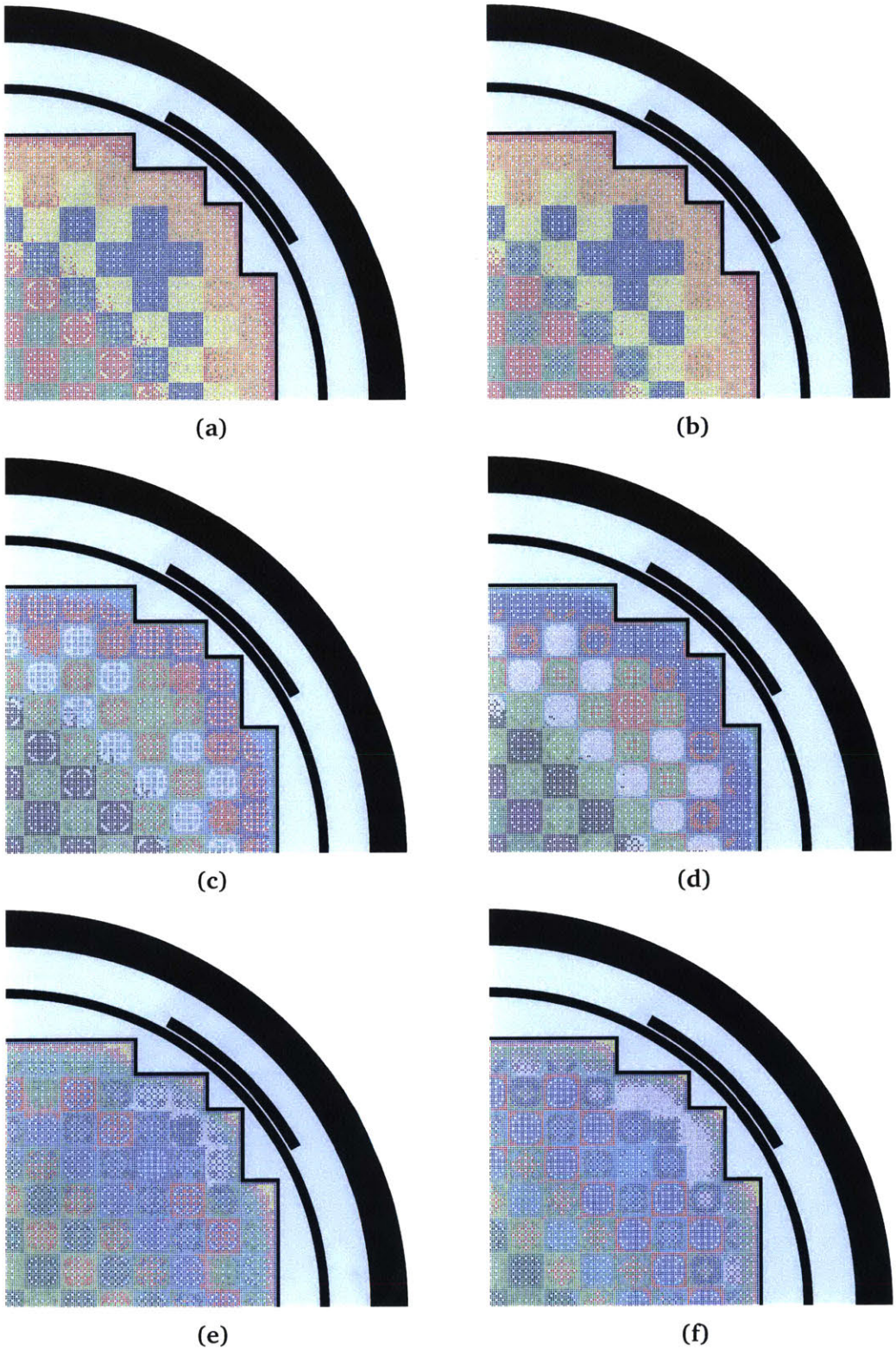


Figure D-5: Materials for the quarter core BEAVRS model with *i*MGXS homogenization without non-MGXS features. The materials for 2, 4, and 8 clusters are illustrated in (a), (c) and (e) for pinch feature selection, and in (b), (d) and (f) for litmus-only feature selection, respectively.

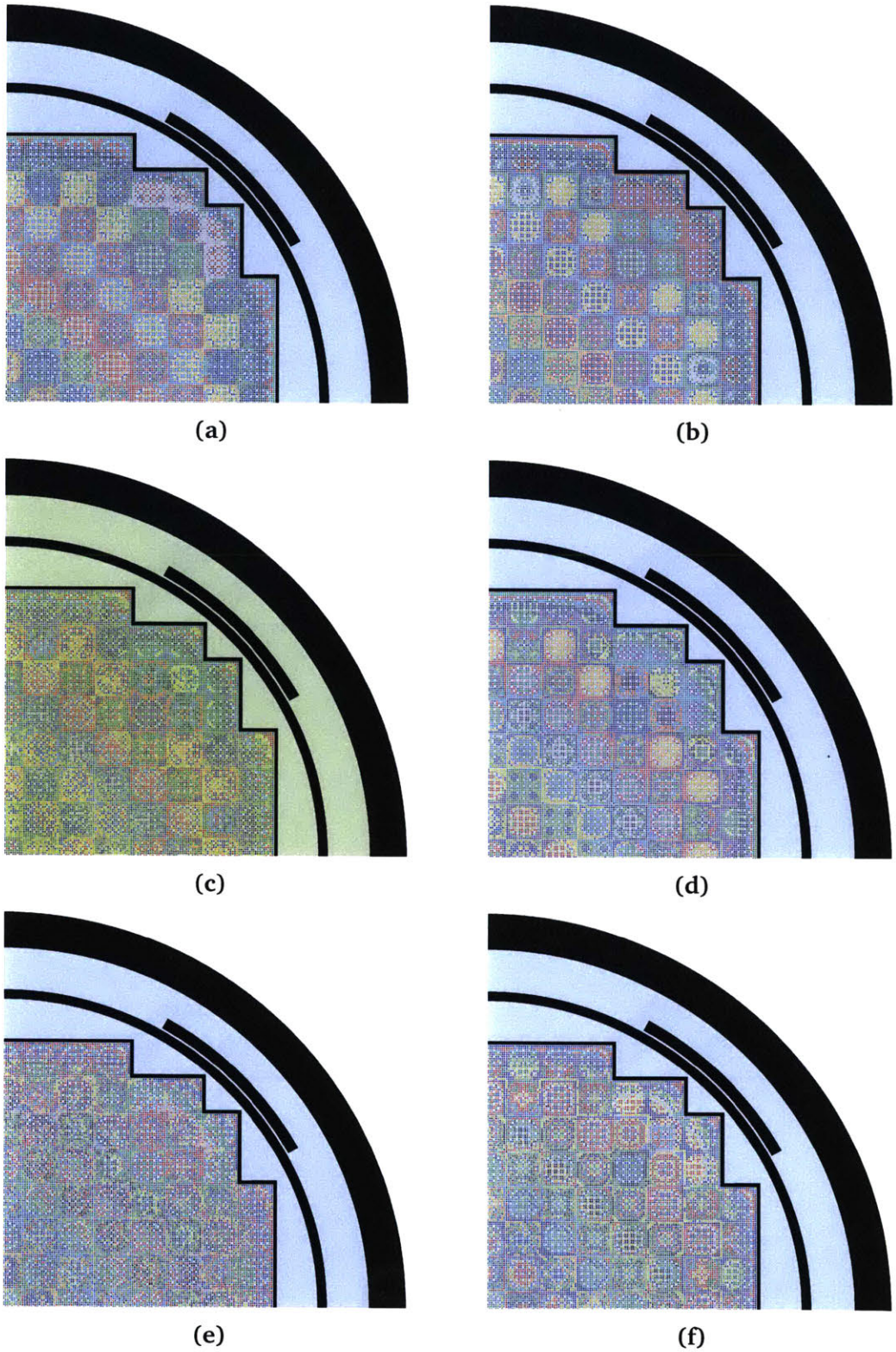


Figure D-6: Materials for the quarter core BEAVRS model with *i*MGXS homogenization without non-MGXS features. The materials for 16, 32 and 64 clusters are illustrated in (a), (c) and (e) for pinch feature selection, and in (b), (d) and (f) for litmus-only feature selection, respectively.

D.2 Clustering With Dimensionality Reduction

A variety of dimensionality reduction techniques such as Principal Component Analysis (PCA, Sec. 10.4.1), Independent Component Analysis (ICA, Sec. 10.4.1) and Factor Analysis (FA, Sec. 10.4.3) may be used by the *i*MGXS data processing pipeline. Dimensionality reduction extracts information from features into a low-dimensional vector space to reduce data storage and predictor training time, and to control model complexity. The clustered geometries presented in Sec. 10.8 were found by clustering features which were *not* subjected to dimensionality reduction. This section illustrates the geometries that are found when PCA, ICA or FA are employed.

The clustered geometries for the individual fuel assembly and 2×2 colorset benchmarks are shown in Figs. D-7 to D-10. Similar to the analysis in Sec. 10.8, the agglomerative clustering algorithm (Sec. 10.5.2) is applied to find four clusters from the same 2-group MC tally datasets used in Sec. 10.8. The litmus-only feature selection technique was used to select the "best" reaction type for each pair of nuclides and energy groups, for each nuclide in the fuel and both energy groups. In each case, PCA, ICA and FA are considered along with the base case of no dimensionality reduction. The PCA, ICA and FA methods were each used to identify two new features as linear combinations of the original features. The two new features were then used in place of the original features in the predictor training phase of the *i*MGXS data processing pipeline.

The most important insight that can be drawn from the figures is that dimensionality reduction techniques have a dramatic impact on the resultant cluster models. It is beyond the scope of this thesis to determine the fundamental reason(s) why each dimensionality reduction technique leads to unique cluster models. Nevertheless, it is clear that each technique's construction of a linear basis set to optimize different metrics (*i.e.*, sample variance and mutual independence for PCA and ICA, respectively) separates the samples in highly distinct structures from those observed in the original feature space. Although this thesis does not further consider the impact of dimensionality reduction on the efficacy of the *i*MGXS scheme, it is the author's opinion that it may be especially helpful to cluster "noisy" MGXS data, and may be a promising avenue for future research.

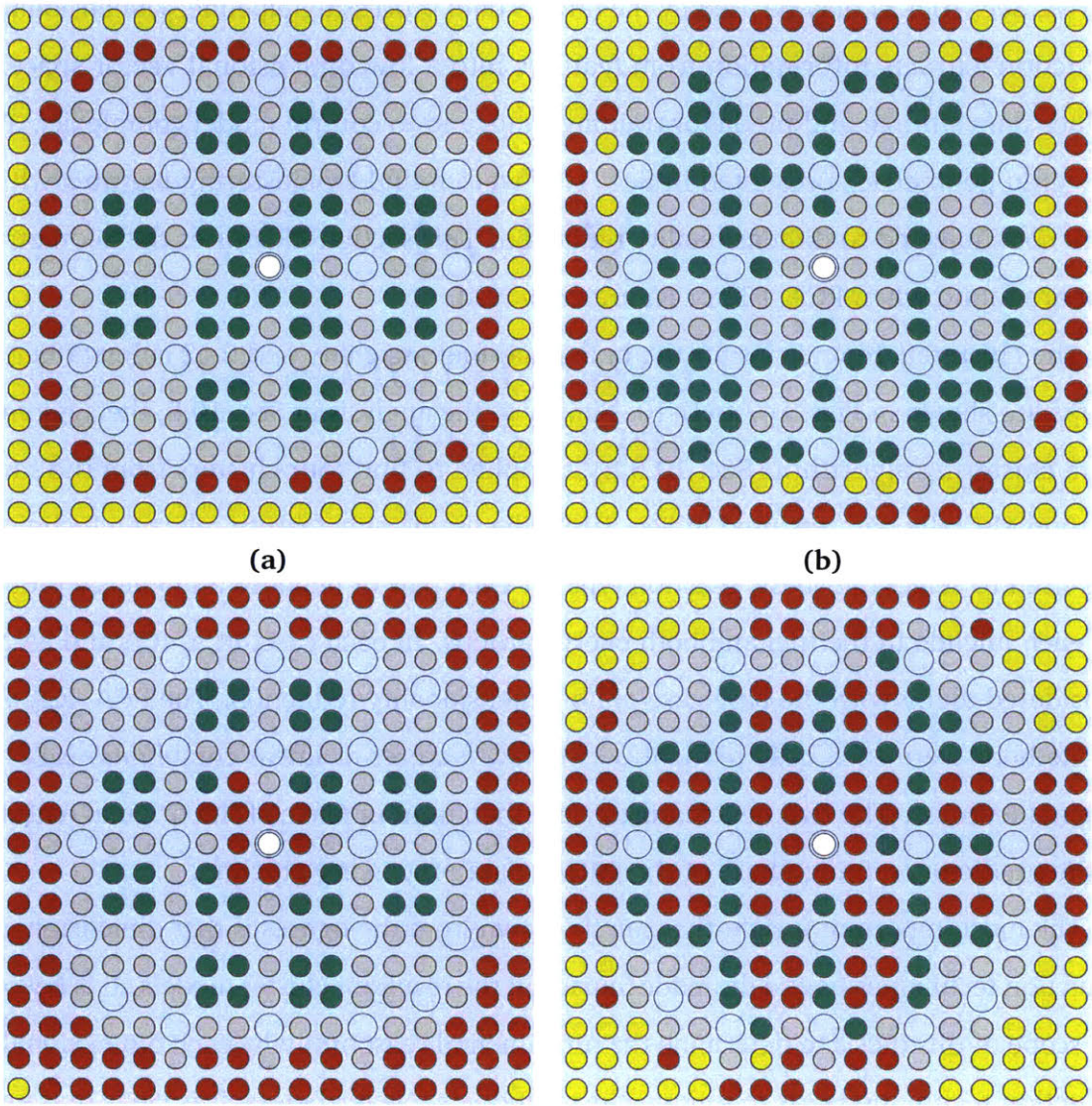


Figure D-7: Materials for the 1.6% enriched fuel assembly with *i*MGXS homogenization. The materials without dimensionality reduction (a) are compared to those identified when PCA (b), ICA (c) and factor analyis (d) are applied, respectively.

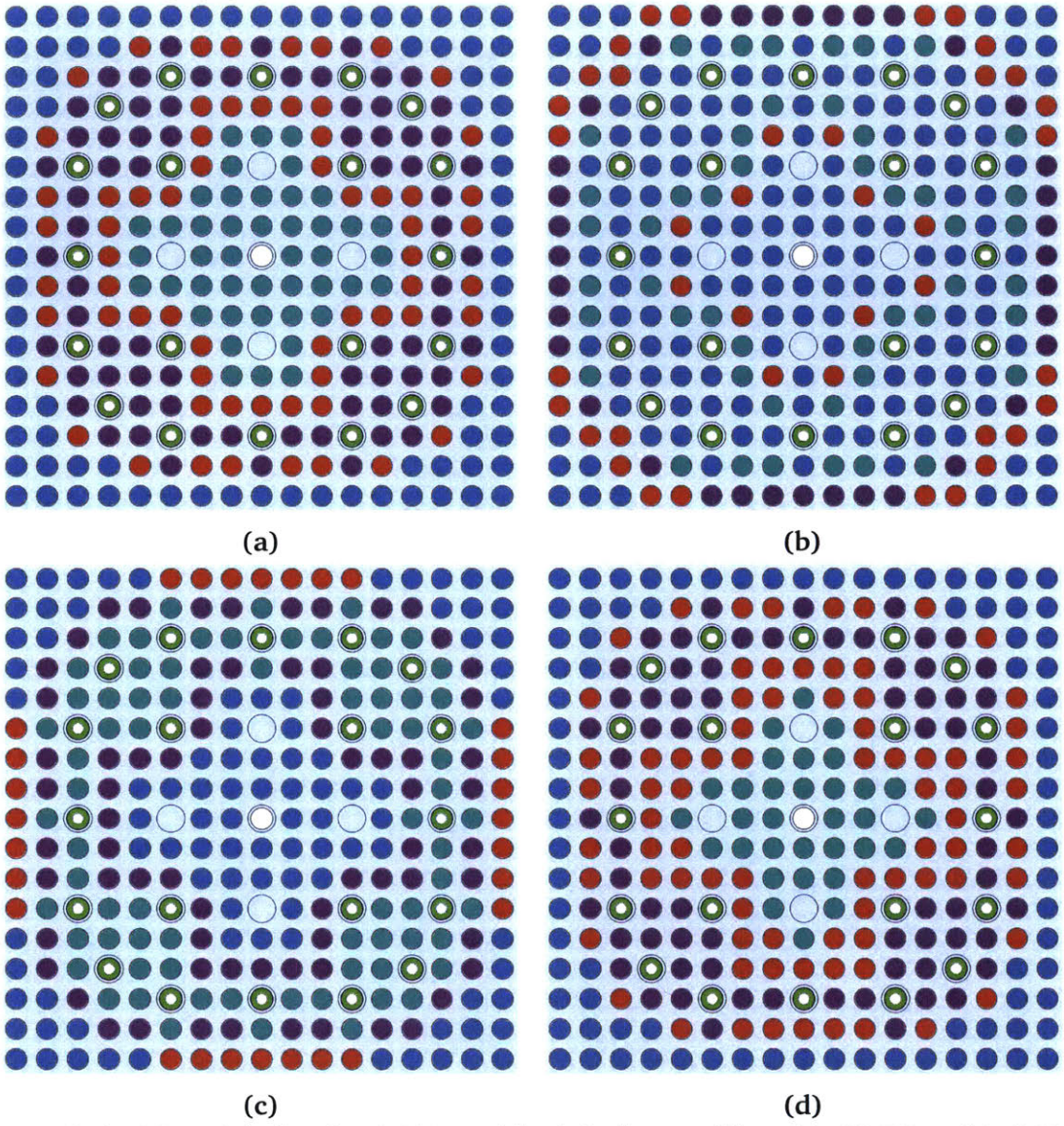


Figure D-8: Materials for the 3.1% enriched fuel assembly with 20 BPs with *i*MGXS homogenization. The materials without dimensionality reduction (a) are compared to those identified when PCA (b), ICA (c) and factor analyis (d) are applied, respectively.

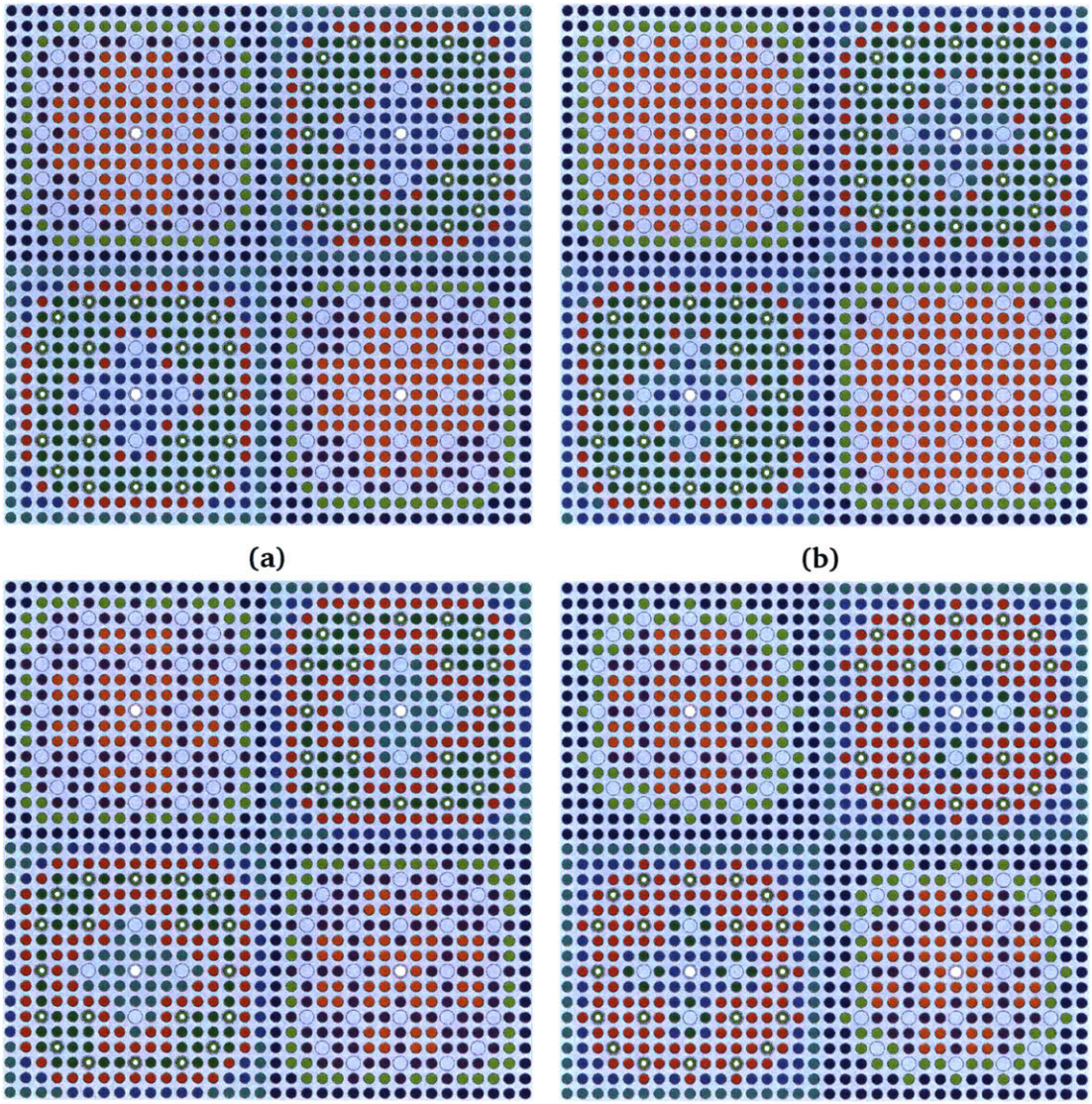


Figure D-9: Materials for the 2×2 colorset with *i*MGXS homogenization. The materials without dimensionality reduction (a) are compared to those identified when PCA (b), ICA (c) and factor analyis (d) are applied, respectively.

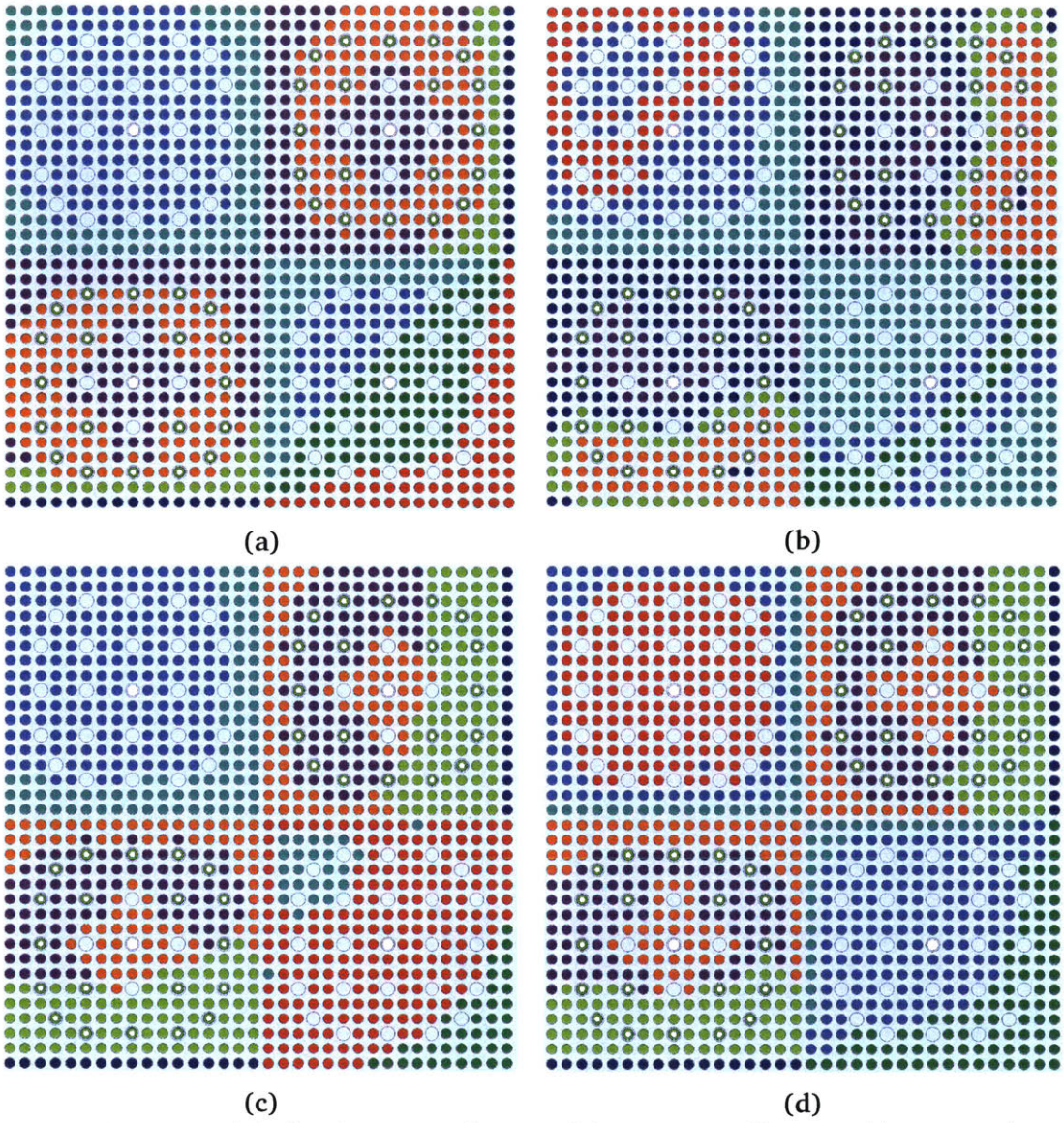


Figure D-10: Materials for the 2×2 colorset with a water reflector with *i*MGXS homogenization. The materials without dimensionality reduction (a) are compared to those identified when PCA (b), ICA (c) and factor analyis (d) are applied, respectively.

References

[1] J. LEPPÄNEN, *Serpent – A Continuous-Energy Monte Carlo Reactor Physics Burnup Calculation Code*, VTT Technical Research Centre of Finland, 2013.

[2] G. I. BELL and S. GLASSTONE, "Nuclear Reactor Theory," Division of Technical Information, US Atomic Energy Commission (1970).

[3] A. HÉBERT, *Applied Reactor Physics*, Presses inter Polytechnique (2009).

[4] D. G. CACUCI et al., *Handbook of Nuclear Engineering*, Springer New York (2010).

[5] R. MACFARLANE, "TRANSX 2: A Code for interfacing MATXS Cross-Section Libraries to Nuclear Transport Codes," Los Alamos National Laboratory (1993).

[6] R. MACFARLANE, "PSR-480/NJOY99.0: Code System for Producing Pointwise and Multigroup Neutron and Photon Cross Sections from ENDF/B Data," Los Alamos National Laboratory (2000).

[7] G. BELL, G. HANSEN, and H. SANDMEIER, "Multitable Treatments of Anisotropic Scattering in S N Multigroup Transport Calculations," *Nuclear Science and Engineering*, **28**, *3*, 376 (1967).

[8] A. YAMAMOTO, Y. KITAMURA, and Y. YAMANE, "Simplified Treatments of Anisotropic Scattering in LWR Core Calculations," *Journal of Nuclear Science and Technology*, **45**, *3*, 217 (2008).

[9] J. LEPPÄNEN et al., *Development of a New Monte Carlo Reactor Physics Code*, VTT Technical Research Centre of Finland (2007).

[10] E. FRIDMAN and J. LEPPÄNEN, "On the Use of the Serpent Monte Carlo Code for Few-Group Cross Section Generation," *Annals of Nuclear Energy*, **38**, *6*, 1399 (2011).

[11] J. LEPPÄNEN, M. PUSA, and E. FRIDMAN, "Overview of Methodology for Spatial Homogenization in the Serpent 2 Monte Carlo Code," *Annals of Nuclear Energy*, **96**, 126 (2016).

[12] E. DORVAL and J. LEPPäNEN, "Monte Carlo Current-Based Diffusion Coefficients: Application to Few-Group Constants Generation in Serpent," *Annals of Nuclear Energy*, **78**, 104 (2015).

[13] G. ILAS and F. RAHNEMA, "A Monte Carlo Based Nodal Diffusion Model for Criticality Analysis of Spent Fuel Storage Lattices," *Annals of Nuclear Energy*, **30**, *10*, 1089 (2003).

[14] J. M. POUNDERS, *Stochastically Generated Multigroup Diffusion Coefficients*, PhD thesis, Georgia Institute of Technology, 2006.

[15] J. M. POUNDERS and F. RAHNEMA, "On the Diffusion Coefficients for Reactor Physics Applications," *Nuclear Science and Engineering*, **163**, *3*, 243 (2009).

[16] J. M. POUNDERS, F. RAHNEMA, and K. J. CONNOLLY, "The History-Partitioning Method for Multigroup Stochastic Cross Section Generation," *Nuclear Engineering and Design*, **293**, 16 (2015).

[17] N. Z. CHO, S. YUN, and J. LEE, "Generation of Homogenized Nodal Parameters by Monte Carlo Method with Non-Zero Leakage Spectra in Global-Local Iteration Framework," *Transactions of the American Nuclear Society*, **101**, 707 (2009).

[18] S.-H. YUN and N.-Z. CHO, "Monte Carlo Depletion Under Leakage-Corrected Critical Spectrum via Albedo Search," *Nuclear Engineering and Technology*, **42**, *3*, 271 (2010).

[19] T. YAMAMOTO, "Monte Carlo Algorithm for Buckling Search and Neutron Leakage-Corrected Calculations," *Annals of Nuclear Energy*, **47**, 14 (2012).

[20] T. YAMAMOTO, "Monte Carlo Method with Complex Weights for Neutron Leakage-Corrected Calculations and Anisotropic Diffusion Coefficient Generations," *Annals of Nuclear Energy*, **50**, 141 (2012).

[21] H. J. SHIM, J. Y. CHO, J. S. SONG, and C. H. KIM, "Generation of Few Group Diffusion Theory Constants by Monte Carlo Code," *Transactions of the American Nuclear Society*, **99**, 343 (2008).

[22] H. J. PARK, H. G. JOO, H. J. SHIM, and C. H. KIM, "Assembly Depletion with Critical Spectrum in McCARD Monte Carlo Calculations and Comparison with HELIOS," *PHYSOR* (2010).

[23] H. J. PARK, H. J. SHIM, H. G. JOO, and C. H. KIM, "Generation of Few-Group Diffusion Theory Constants by Monte Carlo Code McCARD," *Nucl. Sci. and Eng.*, **172**, *1*, 66 (2012).

[24] B. R. HERMAN, B. FORGET, K. SMITH, and B. N. AVILES, "Improved Diffusion Coefficients Generated from Monte Carlo Codes," *Proc. Int'l Conf. on Mathematics and Computational Methods Applied to Nuclear Science & Engineering*, Sun Valley, ID, USA, 2013.

[25] L. ZHAOYUAN, S. KORD, and F. BENOIT, "A Cumulative Migration Method for Computing Rigorous Transport Cross Sections and Diffusion Coefficients for LWR Lattices with Monte Carlo," *Proc. PHYSOR*, Sun Valley, ID, USA, 2016.

[26] K. OKUMURA, T. MORI, M. NAKAGAWA, and K. KANEKO, "Validation of a Continuous-Energy Monte Carlo Burn-Up Code MVP-BURN and its Application to Analysis of Post Irradiation Experiment," *Journal of Nuclear Science and Technology*, **37**, *2*, 128 (2000).

[27] M. TOHJOH, M. WATANABE, and A. YAMAMOTO, "Application of Continuous-Energy Monte Carlo Code as a Cross-Section Generator of BWR Core Calculations," *Annals of Nuclear Energy*, **32**, *8*, 857 (2005).

[28] R. GAST, "Procedure for Obtaining Neutron Diffusion Coefficients From Neutron Transport Monte Carlo Calculations (AWBA Development Program)," Bettis Atomic Power Lab (1981).

[29] L. ONDIS II, L. TYBURSKI, and B. MOSKOWITZ, "RCP01–A Monte Carlo Program for Solving Neutron and Photon Transport Problems in Three-Dimensional Geometry with Detailed Energy Description and Depletion Capability," B-TM-1638. Bettis Atomic Power Laboratory (2000).

[30] "Status of the VIM Monte Carlo Neutron/Photon Transport Code,".

[31] E. L. REDMOND, *Multi-Group Cross Section Generation via Monte Carlo Methods*, PhD thesis, Massachusetts Institute of Technology, 1997.

[32] S. VAN DER MARCK, J. KUIJPER, and J. OPPE, "Homogenized Group Cross Section by Monte Carlo," *Proc. PHYSOR*, Vancouver, Canada, 2006.

[33] J. E. HOOGENBOOM, V. A. KHOTYLEV, and J. M. THOLAMMAKKIL, "Generation of Multi-Group Cross Sections and Scattering Matrices with the Monte Carlo Code MCNP5," *Proc. Joint Int'l Topical Meeting on Math. & Comp. and Supercomp. in Nucl. Appl.*, p. 15–19, Monterey, CA, 2007.

[34] K. YOSHIOKA and Y. ANDO, "Multi-Group Scattering Matrix Generation Method Using Weight-to-Flux Ratio Based on a Continuous Energy Monte Carlo Technique," *Journ. of Nucl. Sci. and Tech.*, **47**, *10*, 908 (2010).

[35] K. YOSHIOKA, Y. TAKEUCHI, T. KITAMURA, and S. SAKURAI, "Multi-Group Constants Generation System for 3D-Core Simulation Using a Continuous Energy Monte Carlo Technique," *Prog. in Nucl. Sci. and Tech.*, **2**, 334 (2011).

[36] L. CAI, *Condensation and Homogenization of Cross Sections for the Deterministic Transport Codes with Monte Carlo Method: Application to the GEN IV Fast Neutron Reactors*, PhD thesis, Université Paris Sud-Paris XI, 2014.

[37] A. NELSON, *Improved Convergence Rate of Multi-Group Scattering Moment Tallies for Monte Carlo Neutron Transport Codes*, PhD thesis, University of Michigan, 2014.

[38] X-5 MONTE CARLO TEAM, "MCNP-A General Monte Carlo N-Particle Transport Code, Version 5," Los Alamos National Laboratory (2003).

[39] P. ROMANO et al., "OpenMC User's Guide – Theory and Methodology," `http://mit-crpg.github.io/openmc/`, Accessed: 2016-06-14.

[40] B. R. HERMAN, B. FORGET, K. SMITH, P. K. ROMANO, T. M. SUTTON, D. J. KELLY, and B. N. AVILES, "Analysis of Tally Correlation in Large Light Water Reactors," *Proc. PHYSOR*, Kyoto, Japan, 2014.

[41] J. MIAO, B. FORGET, and K. SMITH, "Analysis of Correlations and Their Impact on Convergence Rates in Monte Carlo Eigenvalue Simulations," *Annals of Nuclear Energy*, **92**, 81 (2016).

[42] P. R. BEVINGTON and D. K. ROBINSON, "Data Reduction and Error Analysis," *McGraw-Hill* (2003).

[43] E. WOODCOCK, T. MURPHY, P. HEMMINGS, and S. LONGWORTH, "Techniques Used in the GEM Code for Monte Carlo Neutronics Calculations in Reactors and Other Systems of Complex Geometry," *Proc. Conf. Applications of Computing Methods to Reactor Problems*, volume 557, 1965.

[44] G. POMRANING, "Flux-Limited Diffusion Theory with Anisotropic Scattering," *Nuclear Science and Engineering*, **86**, *4*, 335 (1984).

[45] P. K. ROMANO and B. FORGET, "The OpenMC Monte Carlo Particle Transport Code," *Annals of Nuclear Energy*, **51**, 274 (2013).

[46] B. R. HERMAN, *Monte Carlo and Thermal Hydraulic Coupling Using Low-Order Nonlinear Diffusion Acceleration*, PhD thesis, Massachusetts Institute of Technology, 2014.

[47] J. BUCHOLZ, "SCALE: A Modular Code System for Performing Standardized Computer Analyses for Licensing Evaluation," Oak Ridge National Lab., TN (USA) (1982).

[48] W. BOYD, S. SHANER, L. LI, B. FORGET, and K. SMITH, "The OpenMOC Method of Characteristics Neutral Particle Transport Code," *Annals of Nuclear Energy*, **68**, 43 (2014).

[49] W. BOYD, B. FORGET, and K. SMITH, "OpenCG: A Combinatorial Geometry Modeling Tool for Data Processing and Code Verification," *Proc. Int'l Conf. on Mathematics and Computational Methods Applied to Nuclear Science & Engineering*, Nashville, TN, USA, 2015.

[50] P. K. ROMANO, *Parallel Algorithms for Monte Carlo Particle Transport Simulation on Exascale Computing Architectures*, PhD thesis, Massachusetts Institute of Technology, 2013.

[51] W. GROPP, E. LUSK, N. DOSS, and A. SKJELLUM, "A High-Performance, Portable Implementation of the MPI Message Passing Interface Standard," *Parallel computing*, **22**, *6*, 789 (1996).

[52] A. R. SIEGEL, K. SMITH, P. K. ROMANO, B. FORGET, and K. G. FELKER, "Multi-Core Performance Studies of a Monte Carlo Neutron Transport Code," *Int'l Journal of High Performance Computing Applications*, **28**, *1*, 87 (2014).

[53] P. K. ROMANO, A. R. SIEGEL, B. FORGET, and K. SMITH, "Data Decomposition of Monte Carlo Particle Transport Simulations via Tally Servers," *Journal of Computational Physics*, **252**, 20 (2013).

[54] N. HORELIK, A. SIEGEL, B. FORGET, and K. SMITH, "Monte Carlo Domain Decomposition for Robust Nuclear Reactor Analysis," *Parallel Computing*, **40**, *10*, 646 (2014).

[55] W. BOYD, P. K. ROMANO, and S. HARPER, "Equipping OpenMC for the Big Data Era," *Proc. PHYSOR*, Sun Valley, ID, USA, 2016.

[56] S. KORANNE, "Hierarchical Data Format 5: HDF5," in *Handbook of Open Source Tools*, p. 191–200, Springer US, 2011.

[57] S. WALT, S. C. COLBERT, and G. VAROQUAUX, "The NumPy Array: A Structure for Efficient Numerical Computation," *Computing in Science & Engineering*, **13**, *2*, 22 (2011).

[58] W. MCKINNEY, "Data Structures for Statistical Computing in Python," *Proc. 9th Python in Science Conference*, p. 51–56, 2010.

[59] E. JONES et al., "SciPy: Open Source Scientific Tools for Python," 2001–, [Online; accessed 2016-07-07].

[60] J. SEABOLD and J. PERKTOLD, "Statsmodels: Econometric and Statistical Modeling with Python," *Proc. 9th Python in Science Conference*, 2010.

[61] F. PEDREGOSA, G. VAROQUAUX, A. GRAMFORT, V. MICHEL, B. THIRION, O. GRISEL, M. BLONDEL, P. PRETTENHOFER, R. WEISS, V. DUBOURG, J. VANDERPLAS, A. PASSOS, D. COURNAPEAU, M. BRUCHER, M. PERROT, and E. DUCHESNAY, "Scikit-learn: Machine Learning in Python," *Journ. of Machine Learning Research*, **12**, 2825 (2011).

[62] N. HORELIK, B. HERMAN, B. FORGET, and K. SMITH, "Benchmark for Evaluation and Validation of Reactor Simulations (BEAVRS), v1.0.1," *Proc. Int. Conf. Math. and Comp. Methods Applied to Nuc. Sci. & Eng.*, Sun Valley, Idaho, USA, 2013.

[63] D. LAX, W. BOYD, and N. HORELIK, "An Algorithm for Identifying Unique Regions in Constructive Solid Geometries," *Proc. PHYSOR*, Kyoto, Japan, 2014.

[64] L. LI, "A Low Order Acceleration Scheme for Solving the Neutron Transport Equation," Master's thesis, Massachusetts Institute of Technology, 2013.

[65] W. R. D. BOYD III, "Massively Parallel Algorithms for Method of Characteristics Neutral Particle Transport on Shared Memory Computer Architectures," Master's thesis, Massachusetts Institute of Technology, 2014.

[66] S. C. SHANER, "Transient Method of Characteristics Via the Adiabatic, Theta, and Multigrid Amplitude Function Methods," Master's thesis, Massachusetts Institute of Technology, 2014.

[67] J. R. ASKEW, "A Characteristics Formulation of the Neutron Transport Equation in Complicated Geometries," AAEW-M 1108, UK Atomic Energy Establishment (1972).

[68] W. BOYD, A. SIEGEL, S. HE, B. FORGET, and K. SMITH, "Parallel Performance Results for the OpenMOC Neutron Transport Code on Multicore Platforms," *Int'l Journ. of High Performance Computing Applications* (2016).

[69] W. BOYD et al., "OpenMOC User's Guide – Theory and Methodology," `https://mit-crpg.github.io/OpenMOC/`, Accessed: 2016-07-11.

[70] D. M. BEAZLEY, "Automated Scientific Software Scripting with SWIG," *Future Generation Computer Systems*, **19**, *5*, 599 (2003).

[71] OpenMP Architecture Review Board, "OpenMP Application Program Interface Version 4.0," `http://www.openmp.org/mp-documents/OpenMP4.0.0.pdf`, 2013, [Online; accessed 12/22/2013].

[72] NVIDIA, "NVIDIA CUDA C Programming Guide," `http://docs.nvidia.com/cuda/cuda-c-programming-guide/`, 2013, [Online; accessed 12/22/2013].

[73] W. BOYD, K. SMITH, and B. FORGET, "A Massively Parallel Method of Characteristic Neutral Particle Transport Code for GPUs," *Proc. Int'l Conf. on Mathematics and Computational Methods Applied to Nuclear Science & Engineering*, Sun Valley, ID, USA, 2013.

[74] S. SHANER, B. FORGET, and K. SMITH, "K-Nearest Neighbor Flux Updating Scheme for CMFD Acceleration," *Annals of Nuclear Energy* (2016 (submitted)).

[75] M. EDENIUS, K. EKBERG, B. H. FORSSÉN, and D. KNOTT, "CASMO-4, A Fuel Assembly Burnup Program, User's Manual," *Studsvik0SOA-9501, Studsvik of America, Inc.* (1995).

[76] E. W. WEISSTEIN, "k-Partite Graph. From MathWorld—A Wolfram Web Resource," Last visited on 7/11/2016.

[77] J. RHODES, K. SMITH, and D. LEE, "CASMO-5 Development and Applications," *Proc. ANS Topical Meeting on Reactor Physics (PHYSOR)*, p. 10–14, 2006.

[78] N. A. GIBSON, *Novel Resonance Self-Shielding Methods for Nuclear Reactor Analysis*, PhD thesis, Massachusetts Institute of Technology, 2016.

[79] A. HÉBERT, "A Consistent Technique for the Pin-by-Pin Homogenization of a Pressurized Water Reactor Assembly," *Nuclear Science and Engineering*, **113**, *3*, 227 (1993).

[80] F. B. BROWN, "On the Use of Shannon Entropy of the Fission Distribution for Assessing Convergence of Monte Carlo Criticality Calculations," *Proc. PHYSOR*, p. 10–14, 2006.

[81] E. E. Lewis, G. Palmiotti, T. A. Taiwo, R. N. Blomquist, M. A. Smith and N. Tsoulfanidis, "Benchmark Specifications for Deterministic MOX Fuel Assembly Transport Calculations without Spatial Homogenization," Organisation for Economic Co-operation and Development's Nuclear Energy Agency (2003).

[82] S. S. SHAPIRO and M. B. WILK, "An Analysis of Variance Test for Normality (Complete Samples)," *Biometrika*, **52**, *3/4*, 591 (1965).

[83] L. ABEL, W. BOYD, B. FORGET, and K. SMITH, "Interactive Visualization of Multi-Group Cross Sections on High-Fidelity Spatial Meshes," *Transactions of the American Nuclear Society* (2016).

[84] M. A. HALL, *Correlation-Based Feature Selection for Machine Learning*, PhD thesis, The University of Waikato, 1999.

[85] I. GUYON, J. WESTON, S. BARNHILL, and V. VAPNIK, "Gene Selection for Cancer Classification Using Support Vector Machines," *Machine learning*, **46**, *1-3*, 389 (2002).

[86] I. GUYON and A. ELISSEEFF, "An Introduction to Variable and Feature Selection," *Journal of Machine Learning Research*, **3**, 1157 (2003).

[87] L. BREIMAN, J. FRIEDMAN, C. J. STONE, and R. A. OLSHEN, *Classification and Regression Trees*, CRC Press (1984).

[88] T. K. HO, "The Random Subspace Method for Constructing Decision Forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **20**, *8*, 832 (1998).

[89] I. JOLLIFFE, *Principal Component Analysis*, Wiley Online Library (2002).

[90] G. H. GOLUB and C. REINSCH, "Singular Value Decomposition and Least Squares Solutions," *Numerische Mathematik*, **14**, *5*, 403 (1970).

[91] B. SCHÖLKOPF, A. SMOLA, and K.-R. MÜLLER, "Kernel Principal Component Analysis," *Proc. International Conference on Artificial Neural Networks*, p. 583–588, Springer, 1997.

[92] A. HYVÄRINEN and E. OJA, "Independent Component Analysis: Algorithms and Applications," *Neural networks*, **13**, *4*, 411 (2000).

[93] A. HYVARINEN, "Fast and Robust Fixed-Point Algorithms for Independent Component Analysis," *IEEE Transactions on Neural Networks*, **10**, *3*, 626 (1999).

[94] H. H. HARMAN, "Modern Factor Analysis," (1960).

[95] A. P. DEMPSTER, N. M. LAIRD, and D. B. RUBIN, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journ. of the Royal Statistical Society. Series B (Methodological)* (1977).

[96] V. ESTIVILL-CASTRO, "Why So Many Clustering Algorithms: A Position Paper," *SIGKDD Explor. Newsl.*, **4**, *1*, 65 (2002).

[97] J. MACQUEEN et al., "Some Methods for Classification and Analysis of Multivariate Observations," *Proc. Proc. of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, p. 281–297, Oakland, CA, USA, 1967.

[98] S. LLOYD, "Least Squares Quantization in PCM," *IEEE Transactions on Information Theory*, **28**, *2*, 129 (1982).

[99] D. ARTHUR and S. VASSILVITSKII, "*k*-Means++: The Advantages of Careful Seeding," *Proc. Proc. of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, p. 1027–1035, Society for Industrial and Applied Mathematics, 2007.

[100] D. SCULLEY, "Web-Scale *k*-Means Clustering," *Proc. Proc. of the 19th International Conference on World Wide Web*, p. 1177–1178, ACM, 2010.

[101] S. C. JOHNSON, "Hierarchical Clustering Schemes," *Psychometrika*, **32**, *3*, 241 (1967).

[102] J. H. WARD JR, "Hierarchical Grouping to Optimize an Objective Function," *Journ. of the American Statistical Association*, **58**, *301*, 236 (1963).

[103] T. ZHANG, R. RAMAKRISHNAN, and M. LIVNY, "BIRCH: An Efficient Data Clustering Method for Very Large Databases," *Proc. ACM Sigmod Record*, volume 25, p. 103–114, 1996.

[104] G. J. MCLACHLAN and K. E. BASFORD, "Mixture Models. Inference and Applications to Clustering," *Statistics: Textbooks and Monographs, New York: Dekker, 1988*, **1** (1988).

[105] M. I. JORDAN, Z. GHAHRAMANI, T. S. JAAKKOLA, and L. K. SAUL, "An Introduction to Variational Methods for Graphical Models," *Machine Learning*, **37**, *2*, 183 (1999).

[106] H. ATTIAS, "A Variational Bayesian Framework for Graphical Models," *Advances in Neural Information Processing Systems*, **12**, *1-2*, 209 (2000).

[107] D. M. BLEI et al., "Variational Inference for Dirichlet Process Mixtures," *Bayesian Analysis*, **1**, *1*, 121 (2006).

[108] T. S. FERGUSON, "A Bayesian Analysis of Some Non-Parametric Problems," *The Annals of Statistics* (1973).

[109] D. L. DAVIES and D. W. BOULDIN, "A Cluster Separation Measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2**, 224 (1979).

[110] J. C. DUNN, "Well-Separated Clusters and Optimal Fuzzy Partitions," *Journal of Cybernetics*, **4**, *1*, 95 (1974).

[111] T. CALIŃSKI and J. HARABASZ, "A Dendrite Method for Cluster Analysis," *Communications in Statistics – Theory and Methods*, **3**, *1*, 1 (1974).

[112] P. J. ROUSSEEUW, "Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis," *Journal of Computational and Applied Mathematics*, **20**, 53 (1987).

[113] G. SCHWARZ et al., "Estimating the Dimension of a Model," *The Annals of Statistics*, **6**, *2*, 461 (1978).

[114] R. KOHAVI et al., "A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection," *Proc. Ijcai*, volume 14, p. 1137–1145, 1995.

[115] B. EFRON and R. J. TIBSHIRANI, *An Introduction to the Bootstrap*, CRC Press (1994).

[116] B. FORGET, S. XU, and K. SMITH, "Direct Doppler broadening in Monte Carlo simulations using the multipole representation," *Annals of Nuclear Energy*, **64**, 78 (2014).