# Automated Problem-Specific Nuclide-Transition Selection for Reduced Order Modeling

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Scott Michael Richards

August 2019

© by Scott Michael Richards, 2019
All Rights Reserved.

*In dedication to Alexandra, I never would have been able to do this without you.*

# Acknowledgments

There are many people that I would like to thank, without whom this dissertation would not be what it is today.

Foremost, I would like to thank my advisor Dr. Steven Skutnik, without whose help, advice, and guidance this would not have been possible. I would also like to thank my committee members Dr. Ondrej Chvala, Dr. Jim Ostrowski, and Dr. Ivan Maldonado for their support and assistance.

I would also like to thank the people at Oak Ridge National Laboratory that helped in the formalization of this project, development of the code base, and the theoretical validity of the approach: Dr. William Wieselquist, Dr. Shane Hart, and Dr. Mark Williams.

Finally, I would like to thank my family for their support and encouragement throughout the decade that encompassed my higher education. Without them this would not have been possible.

This material is based upon work supported by the Department of Energy Office of Nuclear Energy under award DE-NE0000737 and National Nuclear Security Administration under Award Number DE-NA0002576.

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

# Abstract

A method for automated library reduction for Origen nuclear depletion and decay code was developed for increased computational efficiency. The added fidelity of on-the-fly depletion and source term generation would be beneficial to many application types. For this to be practical though, the depletion problem needs to be restricted to only the most important nuclides for the application. The large standard library inventories result in relatively large memory requirements and runtimes that become burdensome within codes that require many depletion zones and/or depletion substeps per time step. However, the tracked nuclides do not equally contribute to the problem, and therefore a subset of the total nuclides can be removed from the system with little loss of accuracy. While there are current methods to select a limited nuclide inventory, this method does so in a generalized manner that requires consideration of the specific problem and metric for the application and preservation of the physical relationships between nuclides. To this end a number of metrics are available to measure library accuracy for a given problem, such as depletion inventory, total activity, gamma dose, decay heat, and individual nuclide inventory. Using these metrics, and their sensitivities to nuclide inventories, it is possible to reduce Origen's full inventory of thousands of nuclides to several hundred nuclides while only affecting the metric of interest by less than 1 pcm (per cent mille or $10^{-5}$). The method for this problem-specific reduction relies on maintaining the physical meaning of the transition system to the highest degree reasonable. This means maintaining the integrity of the subsystem in relation to its behavior within the full system. Of the various approaches studied to achieve this compression, the most successful method is one that takes a layered approach. This method makes an estimate of the final system through the cutting planes method then makes successive corrections to

that estimate in each layer to account for the physical behavior of truncating the transition system that is not present in standard system problems.

# Motivation

To meet the ever present need for more sophisticated and more accurate models for reactors and nuclear safeguards it is necessary to also include isotopic depletion and isotopic inventory tracking. The areas of study that are most seeing this demand are core modeling and neutronics in high fidelity modeling of existing reactors designs and advanced reactor designs [23], isotopic modeling in spent fuel pools to take advantage the decreased reactive worth of used fuel to increase storage capacity, and higher-fidelity source term models for development of the next generation of international safeguards techniques. On-the-fly depletion capabilities are also advantageous in fuel cycle modeling to predict the relative benefits of different fuel cycle options and better plan an intermediate and final repository design needing an accurate measure of decay heat production, reactive worth, and radiotoxicity over thousands to millions of years.

To facilitate this wide range of end uses, coupling of high fidelity depletion and isotopic tracking codes must not be limited to high performance computing applications. It is the goal of this research reduce the computational cost of including high fidelity depletion in the work flow of general user level codes that require a great many depletion steps, the types of codes that are run on standard workstations and laptops with limited resources. It is only through full integration of accurate depletion models can a source term be accurately modeled, whether that source term be a radiation signature, macroscopic nuclear property, or general nuclide concentrations.

# Table of Contents


**1 Introduction** **1**

1.1 Historic Depletion Approaches . . . . . 2

1.2 Alternative Depletion Methods . . . . . 4

1.3 Origen . . . . . 4

1.4 Depletion Solver Methodology . . . . . 6

1.5 Origen Validation . . . . . 10

1.6 Applications of Origen . . . . . 12

**2 Present Approaches to Reduced-Order Depletion Modeling** **14**

2.1 Transition Matrix Reduction Methods . . . . . 14

2.2 Approximate Subspace Basis Methods . . . . . 15

2.3 Topological Transition Matrix Approximation . . . . . 16

2.4 Library Reduction by Expert Judgment . . . . . 17

**3 Library Reduction Methods** **21**

3.1 Graph Translation of Transition Matrix . . . . . 21

3.2 Removals . . . . . 25

3.3 Determination of Removals . . . . . 28

3.3.1 Threshold-Truncation Method . . . . . 30

3.3.2 Graph Theory Methods . . . . . 32

3.4 Weighting and Problem Type Metrics . . . . . 40

3.4.1 Contribution Weighting Method . . . . . 42

3.4.2 Distributed Contribution Weighting Method . . . . . 44

3.4.3 Adjoint Weighting Method . . . 46
3.5 Subsystem Estimation and Correction Methods . . . 49
3.5.1 Competition Correction . . . 49
3.5.2 Path Correction . . . 50
3.5.3 Removal of Duplicate Nuclides . . . 51
3.5.4 Removal of Short-Lived Fission Products . . . 52

**4 Problem-Dependent Transition System Behavior 54**
4.1 Problem Type . . . 56
4.1.1 Depletion Problems . . . 58
4.1.2 Total Activity Problems . . . 59
4.1.3 Decay Heat Problems . . . 61
4.1.4 Energy-Dependent Radiation Source Problems . . . 63
4.2 Weighting Methods . . . 64
4.2.1 Contribution Weighting . . . 65
4.2.2 Adjoint Weighting . . . 71
4.3 Problem Definition . . . 76
4.3.1 Burnup Dependence . . . 77
4.3.2 Enrichment Dependence . . . 82
4.3.3 Irradiation Condition Dependence . . . 86
4.3.4 Reactor Type Dependence . . . 90

**5 Library Reduction Effects 94**
5.1 Threshold-Truncation Method . . . 95
5.2 Greedy Algorithms . . . 99
5.3 Cutting Planes Algorithms . . . 104
5.4 PYOMO Model . . . 108

**6 Reduced Library Results 112**
6.1 Reduction Convergence . . . 112
6.2 Reduced Library Robustness . . . 118

6.3 Computational Efficiency . . . 120

**7 Conclusions** **123**

**8 Future Work** **127**

**Bibliography** **128**

**Appendices** **134**

A Transition Systems . . . 135

B Fission Yields . . . 139

C System Determination Method Convergence . . . 140

C.1 Threshold Truncation Method . . . 140

C.2 Greedy Algorithms . . . 142

C.3 Cutting Planes . . . 144

C.4 Best Method Combinations . . . 146

C.5 Expected Reduction Errors . . . 148

**User's Manual** **150**

A Functions . . . 150

A.1 Constructors . . . 152

A.2 Reduction Functions . . . 152

A.3 Visualization Functions . . . 155

B Sample Problems . . . 158

B.1 Default Problem . . . 158

B.2 Setting a Simple Problem . . . 160

B.3 Setting a Detailed Problem and Reduction . . . 162

C All Code Descriptions . . . 166

**Vita** **171**

# List of Tables


3.1 Examples of candidate problem-based weighting factors for the given error to be minimized . . . 42

3.2 Sub-libary specification . . . 52

4.1 Shared contributing nuclides for depletion (DEPL), total activity (TACT), decay heat (DECH), and gamma source (GAMM) problem types for the given number of highest contributing nuclides . . . 57

6.1 Comparison of the effectiveness of the Competition-Correction methods . . . 116

6.2 Accuracy of reduced libraries for varying final burnups, relative to the Origen standard library. . . . 118

6.3 Accuracy of reduced libraries for varying initial enrichments, relative to the Origen standard library. . . . 120

6.4 Computational efficiency gains from N=2 cutting planes reduction for the depletion metric. . . . 121

# List of Figures

2.1 Subsection of the Transition System (Z=[43,62], A=[107,150]) with all nuclides and transitions except fission and byproduct production displayed. Absorption cross section is indicated by shade of red and transition type is indicated by arrow color, and whether the transition is a decay or a reaction is indicated by a dashed or solid line respectively . . . . . 20

3.1 Diagram of transition system graph classes. A solid arrow indicates ownership (strong pointer) and a dotted arrow indicates a reference (weak pointer). The reference arrow from transitions to nuclides points to a different nuclide that has ownership over the transition . . . . . 23

3.2 Addition of a nuclide to the transition system for cases where a parent nuclide doesn't exist and one where it does. Bold grey outline indicates the added components at that step and boxes indicate ownership of all components originating in the box by the respective nuclide . . . . . 26

3.3 Diagram of nuclide and transition structure in TransitionSystem, Nuclide has ownership of everything that originates inside the orange dashed lines. Removal from the system entails removing everything leading into and out of the box. . . . . 27

3.4 Diagram of a polytree representation of a simple transition system and the resultant system from using the threshold-truncation method to reduce the system size by 50%. The black nodes indicate source materials and the other nodes are depicted using a blue-red scale with red indicating a higher weight. 31

3.5 Diagram of a polytree representation of a simple transition system and the resultant system from using the greedy algorithm with nuclide weight to reduce the system size by 50%. The black nodes indicate source materials and the other nodes are depicted using a blue-red scale with red indicating a higher weight. . . . . 34
3.6 Diagram of a polytree representation of a simple transition system and the resultant system from using the cutting-planes algorithm to reduce the system size by 50%. The black nodes indicate source materials and the other nodes are depicted using a blue-red scale with red indicating a higher weight. . . . 37

4.1 The irradiation history and time-step weighting for the default problem. . . . 55
4.2 Nuclide chart of relative contributions in the sample problem for the depletion metric. . . . . 59
4.3 Nuclide chart of relative weights in the sample problem for the total activity metric. . . . . 60
4.4 Nuclide chart of relative weights in the sample problem for the decay heat metric. . . . . 62
4.5 Nuclide chart of relative weights in the sample problem for the gamma source metric, with energy bound of 0.5MeV and 1.5MeV. . . . . 64
4.6 Nuclide chart of depletion metric contribution weights of degree 3 for the sample problem. . . . . 66
4.7 Contribution weights for the sample problem depletion metric for different weighting degrees, including the nuclide only (N=0), the nuclide and products (N=0.5), the nuclide and all directly contributing species (N=1), and further up the burnup chain (N=2) . . . . . 67
4.8 Contribution weights for the sample problem total activity metric for different weighting degrees, including the nuclide only (N=0), the nuclide and products (N=0.5), the nuclide and all directly contributing species (N=1), and further up the burnup chain (N=2) . . . . . 68

4.9 Nuclide contribution weights for the decay heat metric for the sample problem for different weighting degrees, including the nuclide only (N=0), the nuclide and products (N=0.5), the nuclide and all directly contributing species (N=1), and further up the burnup chain (N=2) . . . . . 70
4.10 Nuclide contribution weights for the gamma metric for the sample problem with energy bounds of 0.5 and 1.5 MeV for different weighting degrees, including the nuclide only (N=0), the nuclide and products (N=0.5), the nuclide and all directly contributing species (N=1), and further up the burnup chain (N=2) . . . . . 71
4.11 Nuclide adjoint weights for the depletion metric for the sample problem for the product-corrected and uncorrected adjoint weighting. . . . . 73
4.12 Nuclide adjoint weights for the total activity metric for the sample problem for the product-corrected and uncorrected adjoint weighting. . . . . 74
4.13 Nuclide adjoint weights for the decay heat metric for the sample problem for the product-corrected and uncorrected adjoint weighting. . . . . 75
4.14 Nuclide adjoint weights for the gamma source metric for the sample problem for the product-corrected and uncorrected adjoint weighting. . . . . 76
4.15 Graph of the fewest nuclides shared between final burnups for a given metric type relative to the number of most highly contributing nuclides compared. . 79
4.16 Nuclide charts of the mass distributions for the sample problem with varying final burnups. . . . . 81
4.17 Graph of the fewest nuclides shared between initial enrichments of U-235 for a given metric type relative to the number of most highly contributing nuclides compared. . . . . 83
4.18 Nuclide charts of the mass distributions for the sample problem with varying initial enrichments of U-235. . . . . 85
4.19 Graph of the fewest nuclides shared between different levels of detail and different cycle structures for a given metric type relative to the number of most highly contributing nuclides compared. . . . . 88

4.20 Nuclide charts of the mass distributions for the sample problem with varying detail in cycle structure. . . . . 89
4.21 Graph of the fewest nuclides shared between reactor types (candu28_e07 and w15_e50) for a given metric type relative to the number of most highly contributing nuclides compared. . . . . 91
4.22 Nuclide charts of the mass distributions for the sample problem with varying reactor types and following restrictions associated with their designs. . . . . 93

5.1 Relative errors at a given library size reduced with the threshold-truncation method and the given weighting method. . . . . 96
5.2 Absolute difference between the actual error at a nuclide set size and the error estimated by the given weighting method. . . . . 97
5.3 Relative errors at a given library size reduced with the weight-based greedy algorithm method and the given weighting method. . . . . 100
5.4 Relative weight in the subsystems at a total given size, excluding the weight of source nuclides that are required to be in the subsystems. . . . . 101
5.5 Relative errors at a given library size reduced with the weight-based greedy algorithm or the transition-based greedy algorithm. . . . . 103
5.6 Absolute difference between the relative error at a nuclide set size and the error estimated by the given weighting method. . . . . 104
5.7 Nuclide chart of relative contributions in the standard problem for the depletion metric. . . . . 105
5.8 Nuclide chart of relative contributions in the standard problem for the depletion metric. . . . . 106
5.9 Relative errors at a given library size reduced with the cutting planes based algorithm and the given weighting method. . . . . 108
5.10 Relative errors at a given library size reduced based on the PYOMO integer programming model and the given weighting method. . . . . 110
5.11 Relative errors at a given library size reduced with the most suited weighting for each subsystem determination method. . . . . 111

6.1 Convergence of relative error for the three best combinations of weightings and subsystem determination methods. . . . . 117
6.2 Graph of the difference in highest contributing nuclides after reduction by truncation vs. the greedy algorithm (left) and the truncation vs. the cutting planes algorithm (right). . . . . 121

7.1 Partial reduced-transition system using direct contribution weighting and system truncation to an error of 1%. . . . . 124

A.1 Transition system for Westinghouse 15x15 PWR library with 5% initial enrichment (fission and byproduct transition are not illustrated). . . . . 135
A.2 Transition system for Westinghouse 15x15 PWR library with 5% initial enrichment reduced by N=2 contribution weighting truncation with a 0.01% threshold error (fission and byproduct transition are not illustrated). . . . . 136
A.3 Transition system for Westinghouse 15x15 PWR library with 5% initial enrichment reduced by corrected adjoint weighting and the greedy algorithm with a 0.01% threshold error (fission and byproduct transition are not illustrated). . . . . 137
A.4 Transition system for Westinghouse 15x15 PWR library with 5% initial enrichment reduced by N=2 contribution weighting and the cutting planes algorithm with a 0.01% threshold error (fission and byproduct transition are not illustrated). . . . . 138
B.1 Fission yield curves for fission of U-235 and Pu-239. [7] . . . . . 139
C.1.1 Relative errors at a given library size reduced with the threshold-truncation method and the given weighting method for the total activity metric. . . . . 140
C.1.2 Relative errors at a given library size reduced with the threshold-truncation method and the given weighting method for the decay heat metric. . . . . 141
C.1.3 Relative errors at a given library size reduced with the threshold-truncation method and the given weighting method for the gamma source metric. . . . 141
C.2.1 Relative errors at a given library size reduced with the greedy algorithm method and the given weighting method for the total activity metric. . . . . 142

C.2.2Relative errors at a given library size reduced with the greedy algorithm method and the given weighting method for the decay heat metric . . . . . 143
C.2.3Relative errors at a given library size reduced with the greedy algorithm method and the given weighting method for the gamma source metric . . . . 143
C.3.1Relative errors at a given library size reduced with the cutting planes algorithm method and the given weighting method for the total activity metric.144
C.3.2Relative errors at a given library size reduced with the cutting planes algorithm method and the given weighting method for the decay heat metric 145
C.3.3Relative errors at a given library size reduced with the cutting planes algorithm method and the given weighting method for the gamma source metric . . . . . 145
C.4.1Relative errors at a given library size reduced with the subsystem determination method and the weighting method given for the total activity metric. . . 146
C.4.2Relative errors at a given library size reduced with the subsystem determination method and the weighting method given for the decay heat metric . . . 147
C.4.3Relative errors at a given library size reduced with the subsystem determination method and the weighting method given for the gamma source metric . 147
C.5.1Relative errors at a given library size reduced with the subsystem determination method and the weighting method given for the total activity metric. . . 148
C.5.2Relative errors at a given library size reduced with the subsystem determination method and the weighting method given for the decay heat metric . . . 149
C.5.3Relative errors at a given library size reduced with the subsystem determination method and the weighting method given for the gamma source metric . 149

# Chapter 1

# Introduction

Nuclear systems are inherently time-dependent due to the nature of the materials involved. These materials decay, and in nuclear reactors, deplete. However, considering the time evolution of the system has historically been of secondary importance in most fields of nuclear engineering, as the steady-state systems proved to be challenging and the main focus of method development. Material decay and depletion has been considered and originally developed as a stand-alone application in the 1970's and 1980's. With the development of modern codes and methods for simulating systems with nuclear materials the treatment of material time dependence is necessary in conjunction with the other aspects of the simulation in order to reach the desired model accuracy. With this development of depletion codes, there has been parallel research into the limits of how few nuclides need to be depleted to affect the results. Originally this resulted in macro-depletion and limited micro-depletion approaches but more modern approaches have been taken as well. These approaches, which are described in greater detail in chapter 2, seek to create reduced order models (ROM) through creating limited subspace spanning basis vectors for the transition matrix, machine learning based restructuring of an approximate transition matrix, or even expert topological reorganizations to truncate the transition matrix. Unlike my depletion ROM approach, these approaches do not seek to preserve the underlying physical meaning of the transition system and only approximate its macroscopic behavior.

## 1.1 Historic Depletion Approaches

The level of detail in the treatment of materials and their properties in nuclear codes varies greatly by code age and application, with the oldest reactor simulation codes, for example, supporting very few materials and treating their properties as constants. However, as time has gone on new, more accurate treatments have been developed. One of the earliest methods that is still around today is the so-called Macroscopic depletion. This is essentially a large pre-calculated table where the material properties are treated as bulk parameters (macroscopic cross sections chiefly) and current values are interpolated based on the material history (burnup, void coefficient, temperature, boron concentration). The tables are typically calculated via a separate program to generally cover the range of values for the interpolatable parameters for the particular reactor. While this method is very fast and requires little in the way of computational cost and storage, it is very inaccurate in cases where the actual material history is not close to one of the pre-calculated scenarios in the table.

As computational power increased microscopic-depletion became feasible. Microscopic depletion is what is commonly thought of as depletion, where isotopic concentrations are tracked and the nuclide inventory evolves based on microscopic cross section, number density, and scalar flux to calculate the answers to the Bateman equations

$$\underline{\dot{N}} = \underline{\underline{A}}\,\underline{N} \tag{1.1}$$

where,

- $\underline{N}$ is a vector of the concentrations of the nuclides in the problem
- $\underline{\dot{N}}$ is a vector of the rates of change of the nuclides in $\underline{N}$
- $\underline{\underline{A}}$ is the transition matrix containing the both the decay rates and reaction rates for the given flux

However, the memory and processing requirements to fully deplete all known nuclides are still too great for many code systems. To overcome this, both limited depletion and hybrid depletion methods have been devised. In the case of limited depletion, very few

nuclides are actually depleted. Typically, only a few fissile actinides and fission products with large cross sections such as Xe-135 and Gd-157 are actually depleted, with the rest being treated as a lumped material. It is more common for in the realm of 80-120 nuclides to be tracked. Hybrid methods are a combination of macroscopic depletion and microscopic depletion where a small number of nuclides are tracked and depleted explicitly (usually the nuclides that have the largest short-term impact on the problem like Xe-135 in PARCS[39]) and the rest of the system is treated as a homogenous mass macroscopically depleted.

Many modern code systems have the need though for full microscopic-depletion to be able to model a diverse set of scenarios and problems. Without spending many hours investigating key nuclides and the decay/reaction transitions that contribute to the problems being solved, a very large number of nuclides and transitions must be tracked to achieve the accuracies that are desired. This desired accuracy is the reason for the inclusion of this more complicated, but physical, depletion scheme. Presently, there are a number of codes that have been developed for microscopic depletion in different simulation types such as Mure for Monte Carlo Applications[13], FISPACT for the dynamic fuel cycle code CLASS[25, 31], and Origen for nodal deterministic transport codes. However, as more modeling applications have begun to require greater fidelity the inclusion of microscopic depletion has been more prolific. In many of these cases, Origen has been chosen to meet the depletion needs of the code systems due to its high accuracy, validation, and easy to implement API (application programming interface). Due to these qualities and the active support and development at Oak Ridge National Laboratory, Origen has recently been included in many codes such as the deterministic transport code MPACT [23], the diffusion and nodal expansion code Nestle [5], the fuel cycle codes Cyclus [30, 16] and ORION [26], the inverse depletion code INDEPTH [9], and the used nuclear fuel database tool UNF-ST&DARDS [29].

The downside to these depletion methods is that they can be very computationally expensive, requiring greater memory and much longer run times relative to other simpler depletion methods. In some applications this increase in runtimes can become prohibitively expensive. This is often the case for monte carlo simulations but can also be the case for any application calls for a high spatial and/or time resolution, resulting in the runtime costs being thousands (or more) times higher. To reduce this burden in the deterministic transport

code MPACT, a so-called "artisanal reduced" library was created for Origen. MPACT uses a 2D-1D (two dimensional method of characteristics on a plane and diffusion along the vertical axis) neutron transport method that requires a very large number of depletion calculations. The reduced library was based on a smaller (244 nuclide) library and expanded to 255 final nuclides[33].

## 1.2 Alternative Depletion Methods

For certain applications it is being investigated if it is possible to avoid explicit depletion calculations in order to decrease the computational costs. The most prevalent of these methods being researched currently is the application of neural networks and machine learning algorithms to circumvent calculating any of the costly physics and instead find a relationship between the problem type and input variables and the output quantities of interest, which is a common factor in many library reduction techniques that are also being developed. One of the most successful of these applications is in fuel cycles simulation tools, such as CLASS[25, 24] . Neural networks are more applicable in this field where intermediate depletion step information is not commonly needed and fewer physical quantities are of interest. These quantities are usually limited to plutonium content, decay heat, radiation hazard, and radio-toxicity. The methodology is based on neural networks for building a fuel loading model and a cross section predictor for a PWR reactor loaded with MOX fuel. By narrowing the scope of the dynamic fuel cycle code this method is capable doing fuel cycle level depletion calculations in less than one minute with an accuracy comparable to the conventional microscopic depletion methods. This methodology is still maturing and not currently capable of meeting the needs of general depletion codes, however, it is an innovative, if not physical, approach to the problem.

## 1.3 Origen

Origen (**O**ak **R**idge **I**sotope **Gen**eration Code) calculates time-dependent concentrations, activities and radiation source terms for a large number of isotopes simultaneously generated

or depleted by neutron transmutation, fission, and radioactive decay. Origen is used and maintained within SCALE by Oak Ridge National Laboratory. Origen is used internally by SCALE in the TRITON and Polaris sequences to perform depletion and decay calculations. As a stand-alone SCALE sequence, Origen provides additional unique capabilities to (1) simulate continuous nuclide feed and chemical removal, which can be used to model reprocessing or liquid fuel systems, and (2) generate alpha, beta, neutron, and gamma decay emission spectra. A standard decay library is provided to perform decay calculations and recoverable Q-values from fission and gamma-line energies, while other libraries provide information on spontaneous fission and matrix dependent ($\alpha$, n) source terms. For neutron activation and fuel depletion problems, neutron spectrum-dependent Origen libraries are required and may be created from (1) user-defined spectrum and self-shielding cross sections using the COUPLE module or (2) interpolation of existing Origen reactor libraries (pre-calculated by TRITON) using the Automated Rapid Processing (ARP) module in SCALE.

Origen solves the system of ordinary differential equations (Bateman equations in the form Eq.1.3) that describe nuclide generation, depletion, and decay [32].

$$\frac{dN_i}{dt} = \sum_{j \neq i} (l_{ij}\lambda_j + f_{ij}\sigma_j\phi)N_j(t) - (\lambda_i + \sigma_i\phi)N_i(t) + S_i(t) \tag{1.2}$$

where,

$N_i$ = amount of nuclide $i$ (atoms)

$\lambda_i$ = decay constant of nuclide $i$ ($s^{-1}$)

$l_{ij}$ = fractional yield of nuclide $i$ from the decay of nuclide $j$

$\sigma_i$ = spectrum-averaged one-group removal cross section for nuclide $i$ (barn)

$f_{ij}$ = fractional yield of nuclide $i$ from neutron-induced removal of nuclide $j$

$\phi$ angle and energy integrated time-dependent neutron flux (neutrons/$cm^2$-s)

$S_i$ = time-dependent source/feed term (atoms/s)

Origen solves these equations with the energy-dependent microscopic cross sections that have undergone a flux-weighted spectral collapse to 1-group cross sections, which preserves reaction rates and results in 1-group cross-sections and reaction yields. Origen also does not account for spatial dependencies, which can be interpreted as either the solution at a point or the average solution over some volume. The latter interpretation tends to be preferred so that the collapsing flux is the spatially averaged neutron flux magnitude over the material of interest. Origen transition matrices can either be loaded from a pre-existing library or can be created through COUPLE and the Origen API resources. In either case the matrix uses sparse-matrix storage to store one or more transition matrices and in fact most libraries contain many transition matrices to group matrices that where generated from the same initial problem configuration (enrichment, lattice configuration, average moderator density, etc.) but at different burnup points.

The API allows for easy integration of Origen's depletion capabilities into larger code frameworks. Modern Origen has been developed to be a modular code capable of being used independently of the larger SCALE framework. Its native capabilities can be used directly without the need for slow coupling methods like file passing, or other methods that rely on the user interface[34]. The API operates on a simple two-step process. The solver kernel requires a transition matrix, time step-average material flux and power level, as well as beginning-of-step isotopics. Outside of this management of a single material and time-step is the management of the rest of the materials for the problem and time-steps, which is cycled through invoking the solver in the inner process, thus calculating material compositions for all materials everywhere in the system.

## 1.4 Depletion Solver Methodology

$\underline{\underline{A}}$ in equation 1.1 is commonly referred to as the transition matrix. The representation of the transition-matrix (equation 1.3) as system of ordinary differential equations (ODE) holds the reaction, flux, and feed terms constant over each time step to find the numerical solution.

$$\underline{\underline{A}} = \underline{\underline{A_\sigma}} \cdot \phi + \underline{\underline{A_\lambda}} \tag{1.3}$$

As the sum of separate sub-matrices $A_\sigma$ (representing the reaction terms of the transition matrix) and $A_\lambda$ (containing decay terms) as observed in equation 1.3 is convenient. Adding a continuous removal process described with rate $\lambda$ simply modifies the decay constant, whereas a continuous feed process defines a nonzero component of the source vector $\underline{S}$. During each timestep, the burnup-dependent material properties (i.e., collapsed one-group cross-sections) are considered constant. To reduce the error from this assumption, the choice of length of the time step and the use of substeps to correct the collapsing flux for material changes are common considerations that must be taken at the time of the calculation.

In the transition matrix $\underline{\underline{A}}$ each matrix element is the first order rate constant for the formation of nuclide $i$ from nuclide $j$ given below, and the diagonal elements are the sum loss coefficients of nuclide $i$,

$$a_{ij} = \begin{cases} I_{ij}\lambda_j + f_{ij}\sigma_j\phi & \text{, if i}\neq\text{j} \\ -\lambda_i - \sigma_i\phi & \text{, otherwise} \end{cases} \tag{1.4}$$

where $I_{ij}$ and $f_{ij}$ are the branching ratios for that decay or reaction respectively that produce nuclide $i$. The transition matrix coefficients for decay and reaction transitions are stored separately and reaction transitions are always stored with $\phi = 1$. Later, during solution phase of the system, the transition matrix for step n, $A_n$, is scaled by the average flux corresponding to step n, $\phi_n$, thereby preserving the correct reaction rate. Decay coefficients are directly taken from decay data but the reaction coefficients are generated by calculating all removal cross sections and yields by folding provided flux spectrum $\phi_g$ with multigroup cross sections and yields from nuclear data under the infinite dilution assumption for the multigroup data [32]. The corrections for resonance self-shielding are handled separately (via CENTRM/BONAMI); these correction terms have been folded into the collapsed one-group cross-sections Origen uses.

Calculation of this matrix for reaction transitions is an expensive process, requiring the calculation of self-shielding factors to collapse the cross-sections in addition to neutron transport calculations. However, calculating the material average (indicated in the equation below with a subscript m) scalar flux magnitude as a function of time is a relatively simple

process, requiring only that an equation be applied relating the relative material flux and the power level of the system. Thus, depletion calculations are broken down into steps (j), and substeps (s), for which each step has a fixed transition matrix but the flux is recalculated at each substep. Thus, the equation[34] being solved

$$\frac{d\underline{N_m}}{dt} = (\underline{\underline{A_{\sigma m}}} * \phi_m + \underline{\underline{A_\lambda}})\,\underline{N_m}(t) \tag{1.5}$$

is calculated over each substep as

$$\frac{d\underline{N_m}}{dt} = (\underline{\underline{A_{\sigma m,j}}} * \overline{\phi}_{m,j}\Phi_{j+s/S} + \underline{\underline{A_\lambda}})\,\underline{N_m}(t)\ , \tag{1.6}$$

with substep $t_{j+\frac{s-1}{S}} < t < t_{j+\frac{s}{S}}$, and initial condition $\underline{N_m}(t_{j+(s-1)/S}) = \underline{N}_{m,j+(s-1)/S}$ , for $s = 1, 2, ..., S$

where $\overline{\phi}_{m,j}$ is the step average material total flux, $\underline{\underline{A_{\sigma m,j}}}$ is the step-average reaction transition matrix, and $\Phi_{j+s/S}$ is the substep system power normalization given by

$$\Phi_{j+s/S} = \frac{\overline{P}_j}{\Sigma_m \Sigma_i \kappa_{L,i} \overline{\sigma_{Lm,j,i}} \overline{\phi}_{m,j} V_m N_{m,j+fracs-1S}} \tag{1.7}$$

where

- $\overline{P}_j$ is the given total system power ($Watts$)
- $m$ is the material index
- $i$ is the nuclide index
- $\kappa_{L,i}$ is the energy release per *loss* (capture+fission) of nuclide $i$ *(Joules)*
- $\overline{\sigma_{Lm,j,i}}$ is the *loss* cross section over step $j$ of nuclide $i$ *(barn)*
- $\overline{\phi}_{m,j}$ is the relative (total) flux in material $m$ over step $j$ $(n/cm^2 s)$
- $V_m$ is the volume of material $m$ $(cm^3)$

- $N_{m,j+fracs-1S}$ is the beginning-of-substep $s$ number density of nuclide $i$ *(atom/barn-cm)*

Though Origen has traditionally used the MATREX (MATrix EXponential transform) [32] method to the solve the matrix exponential problem, the latest release has made available the Chebyshev Rational Approximation Method (CRAM) solver. CRAM, besides being able to solve the problem in a similar amount of time and at a higher accuracy in a more robust fashion for a wider range of problems, also enables the implementation of an adjoint solver. The full importance of this ability will be explained in more detail later, however being able to calculate adjoint solutions allows for Perturbation Theory-based calculations of nuclide problem sensitivities. The CRAM solver, detailed in depth in [20], formally considers the homogeneous problem, (i.e., with no sources) of the form

$$\frac{d\underline{N}}{dt} = \underline{\underline{A}}\,\underline{N} \tag{1.8}$$

for which the solution is

$$\underline{N}(t) = \underline{N}_0\, e^{\underline{\underline{A}}\, t} \tag{1.9}$$

The method is based on the observation that eigenvalues of the transition matrix are clustered around the negative real axis, which is taken advantage of by taking a Chebyshev rational approximation of the exponential along the entirety of the negative real axis. This approximation results in the pole-residual form

$$N(t) \approx a_0 N(0) + 2 * Re\left[\sum_{i=1}^{\frac{k}{2}} a_i (At + \theta_i I)^{-1}\right] \tag{1.10}$$

where $a_0$ is the limiting value of the approximation at infinity, $a_i$ and $\theta$ are the residues and poles of the decomposition of the rational function resulting from the expansion, and k is the order of the approximation. Due to the sparseness of the transition matrix A, this can efficiently be solved from systems of the form $(At + \theta_i I)y = N(0)$ for y by LU decomposition, where the transition matrix is factorized into a lower triangular (L) matrix

and and upper triangular (U) matrix allowing for matrix inversion through forward and backward substitutions rather than requiring Gaussian Elimination. The coefficients, poles and residues, do not depend on A, t, or N and can thus be pre-calculated and tabulated.

As mentioned previously, one of the motivations for implementing the CRAM solver, and the main reason it is the preferred solver in this case, is to solve the adjoint depletion equations [36]

$$-\frac{dN}{dt} = A^T N^* + S^*(t), N^*(t_f) = N_f^* \tag{1.11}$$

Adjoint calculation with CRAM are straightforward. The adjoint depletion equation is of the same form with only the transition matrix being transposed and the time steps being evaluated starting from the final time and moving towards the initial time. As transposing does not affect the eigenvalues of the matrix, the exponential of the matrix is solved just as accurately and efficiently as the original matrix is. The importance of the adjoint equation is in the contribution weighting of the nuclides based on perturbation theory.

## 1.5 Origen Validation

Through the employment of the API, highly accurate source terms for accident analysis[19, 38], used fuel characterization [28, 17, 12, 11, 38] (including activity, decay heat, radio-toxicity, radiation emission rates, and fissile content), material activation [38], and fuel cycle studies [18, 38]. Over the course of its development Origen has been subject to hundreds of validation cases using measured data. This validation data is based upon destructive isotopic assays of spent fuel, decay heat of spent fuel, gamma spectra resulting from burst fission, and neutron spectra resulting from spontaneous fission and alpha-n reactions. The data available for these validation studies comes from both domestic and international sources as well as publicly available data, measurement campaigns done through collaborations between government agency and the private sector, and experiments for the purpose of measuring the quantities and establishing benchmarks for codes such as Origen and a future development projects. The Origen data resources, and the SCALE data in general, have been rigorously

tested against these benchmark studies, as well as other codes that have been verified and are trusted tools in the field, by the SCALE data team at Oak Ridge National Laboratory lead by I. C. Gauld, D. Wiarda, M. Pigni, and W. Wieselquist.

Origen data resources include nuclear decay data, multigroup neutron reaction cross sections, neutron induced fission product yields, and decay emission data for photons, neutrons, alpha particles and beta particles including auger electrons. The nuclear decay data are based primarily on ENDF/B-VII.1 evaluations, including half-lives, decay modes, branching fractions, and recoverable energy per decay. Decay modes include beta ($\beta^{\pm}$), electron capture (EC), isometric transition (IT), alpha ($\alpha$), spontaneous fission (SF), delayed neutron emission ($\beta^{-}$n), neutron emission (n), double beta decay ($\beta^{-}\beta^{-}$), and decay by beta and alpha emission ($\beta^{-}\alpha$). The decay resources also include radiotoxicity factors for water and air based upon prescribed annual limits on intake values as defined in Part 10, Title 20 of the Code of Federal Regulation (10CFR20).

The multigroup nuclear reaction cross section libraries include evaluations from the JEFF-3.0/A neutron activation file containing data for 774 target nuclei, including ground and metastable excited states, and more than 12,000 neutron-induced reactions below 20 MeV of more than 20 different types of reactions, provided in various energy group structures. These cross section data are evaluated directly from the European Activation File (EAD-2003). The evaluation include many reactions that may be important for modeling fast fission and other high-energy systems. Neutron reactions are available for 23 reaction types, including (n,n'), (n,2n), (n,3n), (n,f), (n,n'), (n, 2n), (n,3n), (n,n'p), (n,n'd), (n,n't), (n,n'$^3He$), (n, 4n), (n,2np), (n,), (n,p), (n,d), (n,t), (n,$^3He$), (n, $\alpha$), (n, 2$\alpha$), (n, 2p), (n,p$\alpha$). The JEFF-3.0/A evaluations also include extensive compilations of energy-dependent branching fractions that define neutron reaction transitions to ground and metastable energy states, which is fully implemented in Origen.

Energy-dependent ENDF/B-VII.0-based fission product yields are available for 30 fissionable actinides. Gamma-ray and x-ray emission data libraries are based on ENDF/B-VII.1 [32]. The photon libraries contain discrete photon line energy and intensity data for decay gamma and x-rays emission for 1,132 radionuclides, prompt and delayed continuum spectra for spontaneous fission, ($\alpha$,n) reactions in oxide fuel or arbitrary matrices, and

bremsstrahlung from decay beta (electron and positron) particles slowing down in either a $UO_2$ fuel or water matrix. Methods and data libraries used to calculate the neutron yields and energy spectra for ($\alpha$,n) reactions are adopted from the SOURCES4C code[32], with updates made with ENDF/B-VII.1 to correct for errors. Capabilities to calculate the beta and alpha particle emission source and spectra have also been added in the latest release of Origen.

## 1.6 Applications of Origen

Origen was originally released in 1973. Its successor, and predesssor to modern Origen, ORIGEN-2 was released in 1980 and is still used today as a standalone depletion. Modern Origen, also called ORIGEN-S (S, for being the version in SCALE), has many advantages over Origen-2 and even 30 years ago was a substantial improvement over ORIGEN-2, with a more generally applicable capabilities and improved data accuracy. Origen was originally developed as a nuclear reactor licensing code, but was quickly applied to a range of other problems. These problems include

- Modeling of isotopic inventory for core modeling and neutronics in high fidelity modeling of existing reactors designs and advanced reactor designs [23].
- Isotopic modeling in spent fuel pools to take advantage the decreased reactivity worth of used fuel to increase storage capacity.
- Modeling in safeguards and non-proliferation to accurately model special nuclear material (SNM) inventory in spent fuel for both more accurate detection and more accurate tracking of a nations total inventory of SNM.

Also, more accurate isotopics are also seeing an increased emphasis[35] in fuel cycle modeling to predict the potential benefits or risks of advanced fuel cycles and better plan an intermediate and final repository design needing an accurate measure of decay heat production, reactive worth, and radiotoxicity over thousands to millions of years. To ensure the accuracy of the results of Origen for these problems, a concerted development effort was undertaken by Oak Ridge National Laboratory to update and expand the code's capabilities.

Though these efforts have greatly improved Origen as a generic problem depletion and UNF characteristic predictor, they have caused Origen to greatly increase in size for both on disk requirements and active memory requirements by increasing the detail of the libraries. These requirements are a byproduct of the increased demand for accuracy from all depletion codes, and is not unique to Origen.

To compensate for these memory and runtime costs, without the loss of the accuracy brought from these improved capabilities, many different methodologies have been researched to effectively reduce the amount of information being calculated. As discussed in the next chapter, these methods most typically target the transition matrix representation of the Bateman equations, though others attempt to reduce the problem through eliminating physical pathways, or transitions, between nuclides. In either case claims of order of magnitude reductions in run time are common with agreement to the original problem solution to within 1%. These claims would appear to resolve the issues with modern full microscopic depletion, however as with any problem that is simplified, certain assumptions need to be made. Thus, an ideal general problem reduction method should seek to limit the assumptions inherent in the reduction to those that have a basis in the physical meaning of the equations they are manipulating so as not to create unforeseen or limiting effects.

# Chapter 2

# Present Approaches to Reduced-Order Depletion Modeling

Determination of important nuclides and methods to reduce the dimensionality of depletion problems have been proposed and developed for many years. Most commonly, the selection of nuclides deemed to be of significance to a problem is done through expert judgment. Someone with extensive experience in depletion systems makes a judgment call on whether a nuclide needs to be present in the reduced chain. Usually this analysis is accompanied with some direct perturbation analysis to determine the induced error in the system and whether that error is acceptable. While this is a tried and true method, it is relying on that expert to manually repeat this for all systems and problems. To reduce the time required to form these simplified burnup chains, different mathematical approaches have been taken to reduce the dimensionality of the problem.

## 2.1 Transition Matrix Reduction Methods

The simplest and most numerous are approaches to reduce the computational cost of depletion through a reduction of the transition matrix via application linear algebra. Many of these approaches, like the ones described in [40] and [3], reduce the dimensionality of the transition matrix through selecting nuclides based on a physical aspect of the nuclide, such as absorption cross section. With sufficient error checking, these are a simple, fast, and direct

methods for reducing the number of tracked nuclides down to those primarily expected to significantly contribute to the problem. This type of method has even been shown to be viable for an on-the-fly reduction in DYN3D [25], where a reduced tracked nuclide inventory is found at every burnup step. However, using the derived reduced transition matrix does not take into account the interactions between nuclides and thus should not be used for any general form of reduction of the transition matrix.

Kajihara et al. take this method a step further [21] by applying singular value decomposition (SVD) to this matrix to obtain an information matrix. This information matrix defines the relationship between the chosen nuclides and those remaining. By applying SVD to the information matrix, additional candidate nuclides can be found. This process is repeated until the relationship matrix between the chosen nuclides and those remaining has a norm sufficiently close to zero, indicating that there is little to no relationship between the chosen nuclides and those being removed. This method quickly accounts for the interactions of a nuclide with its parents and daughters (overcoming the major short coming of the original method) but does not consider the physical properties of any nuclides after the initial nuclides are chosen.

## 2.2 Approximate Subspace Basis Methods

Outside of direct reduction of the size of the transition matrix there has been research into constraining the transition matrix to a smaller subspace. One of the most promising of these methods is Reduced Order Modeling (ROM). The basic idea of ROM is to build a compact model preserving essential features in the model behavior. Those features can be mathematically described as a basis of an influential subspace and the primary goal of ROM is to seek the minimum size basis (i.e., one the can be truncated to the highest order) with an error within tolerance. A randomized algorithm is used to search subspaces that most represent the dominant interactions between the inputs and the outputs, which in this case are the initial nuclide inventory and operating history, and the final nuclide inventory [1, 2].

This method does construct a simplified system, however, it has only been applied to reduce run time of coupled-physics problems, and not to generate reduced libraries as is the

goal of this research. In this case, coupled-physics problems mean those with multiple physics (also commonly known as "multiphysics") being considered, such as a neutron transport code with depletion and thermodynamics. In general, these complex problems either have a higher error tolerance (larger error in the physics models) or a much longer run time; meanwhile, even in these cases the generated reduced basis can only be used to accelerate that particular problem.

Another similar approach is that of using the Range Finding Algorithm (RFA) by Zhang et al [40]. This method also seeks to reduce the transition matrix through the construction of a orthonormal basis that spans the subspace of the transition matrix, however this method seeks to do this by restricting the range of the matrix rather than its rank. This is possible due to the effective dimensionality of the depletion problem being much lower than the nominal dimensionality for typical irradiation problems. The RFA has been proven as a viable tool for reducing big data sets that are encountered in other scientific fields. Also, similar to the ROM method, the RFA does a random sampling of the input parameters to constructs an orthonormal basis that spans the subspace of the transition matrix. However, this method strips out the physical meaning of the reduced libraries, as the method uses an orthogonalization algorithm after calculating a basis vector from the randomized input sampling. This completely obscures the meaning of the data and prevents any future changes. It also makes the library specialized to the problem type at any error constraint level, so the library is no longer even applicable as a general library. From a physical standpoint these methods not only reduce the number of nuclides tracked, they also replace all transitions with meta-transitions that which produce similar calculation results that have no physical analogue process.

## 2.3 Topological Transition Matrix Approximation

A novel approach that departs from these methods of transforming or reducing the transition matrix is the application of Graph Theory to reorganize the transition matrix to optimize the solve times [22]. This approach is taken in the new depletion code TNT (Topological Nuclide Transmutation). The theory behind this method of acceleration is that by ordering

the transition matrix by nuclide dependency rather than according to their Z-A number, the solver will require fewer iterations and thus reduce solve time. The dependencies are found based on the initial materials and then by using the C++ Boost library's available graph theory solvers, all possible transitions from these materials are found. The transition matrix is then reorganized by these chains, starting with the initial materials, and proceeding based on the number of dependent precursors each daughter has. Rather than a mathematical approach to reducing computational cost, this method takes a physical approach.

By organizing the transition matrix by physical order of dependence, the solver that they use (a Jacobi solver with the matrix exponential approximated with a Taylor series expansion) does not need to calculate the full matrix at every time step. However, their use of the topological structure of the transition system only considers nuclide production at any given time step. In problem with changing time scales, and non-constant conditions (i.e., irradiation to cooling in a single problem), the tracked inventory will change between time steps.

This method of nuclide selection also prevents the user from declaring what is the important physical aspect of the transition matrix other than a particular nuclide. Although they, they attempt to reconcile the problem of shifting time scales from early nuclide burnup to long decay periods for fuel cooling through user declared short, medium, and long time-scales. The current implementation considers one aspect of the physics of the transition system while ignoring the aspect that some of the matrix reduction methods account for and the way this is accounted for, by doing the reduction between time step so as to limit the error from not considering all aspects of the system, greatly slows the depletion calculation. This increase in computational cost defeats the purpose of the library reduction for my purposes.

## 2.4 Library Reduction by Expert Judgment

To reduce the computational resources used by Origen in high accuracy neutron transport codes, namely the Consortium for Advanced Simulation of Light Water Reactors (CASL) 2D-1D method of characteristics transport code MPACT, a reduction in the number of nuclides

and transitions tracked in an Origen library was performed by hand. This reduction [33, 32] was based on building important depletion chains into a small base library to capture just transitions with significant importance to depletion accuracy. With a starting inventory of 244 nuclides, after 8 iterations of changes the final library still only tracked a total 255 nuclides and 5035 transitions, nearly a factor of 10 reduction from standard Origen library composition, resulting in a reduction in run time from 237 ms per depletion step to 10 ms while only introducing less than 0.01% error in $k_{eff}$ at a burnup of 80 MWd/MTU (Megawatt-day per metric ton/tonne initial uranium). These final nuclides that were tracked were split across Origen's 3 sub-libraries that track the origin of the materials as 63 light-element nuclides (i.e. activated non-fuel structural materials), 42 actinides, and 150 fission products.

A simple depletion test case was developed to assess the quality of a simplified library. The test case was two 500-day cycles with 30 days of decay between cycles with a flux spectrum based on a middle-of-life pressurized water reactor (PWR) with an initial mass of 0.4MTU of $UO_2$ at 10.2 $\frac{g}{cm^3}$. To determine the accuracy of each iteration several factors were compared.

1. Power ($W$)
2. Mass ($g$)
3. Activity ($Bq$)
4. Macroscopic "disappearance" cross section ($cm^{-1}$)

This test case was run in the Triton sequence in SCALE with all of the 388 nuclides that full spectral cross section data and setting the number density to zero for those nuclides that are not present in the simplified library. The process of choosing nuclides to add to the starting library of 244 nuclides (referred to as origen_s1_244) was done based on a visual comparison of the system of nuclides in the test library to the original library (fig. 2.1) to identify important transition pathways. The difference in the macroscopic disappearance cross section between the results from the full library and simplified library was used to direct

the visual comparison, since it is believed to be a good indicator of a nuclides importance in depletion/transport systems.

When the nuclides to be added to the next simplified library iteration were identified, they were added through resources for library building in the Origen API. These resources allow for the translation of a transition matrix to a form referred to in the code (and this document) as a "transition system". The transition system form (described in the following section) allows for the nuclides and their gain and loss transitions to be directly added to this system in a straightforward and intuitive manner. Once updated, this form is then translated back into a transition matrix that now has the increased number of nuclides.

Though this work was done by hand for a specific application, it produced very promising results and developed the necessary tools to implement a more general method for this type of simplified library reduction in Origen. Over 7 iterations, a library was made that could replicate the answers of the full, library generated to within an accuracy of 3 to 10%, at $\frac{1}{20}$ the runtime and $\frac{1}{11}$ the memory. From this, it was postulated that similar results could be achieved through a general method to generate simplified libraries if problem-specific metrics were used to weight a nuclides contribution to the problem type, similar to how the disappearance cross section is used for depletion calculations.

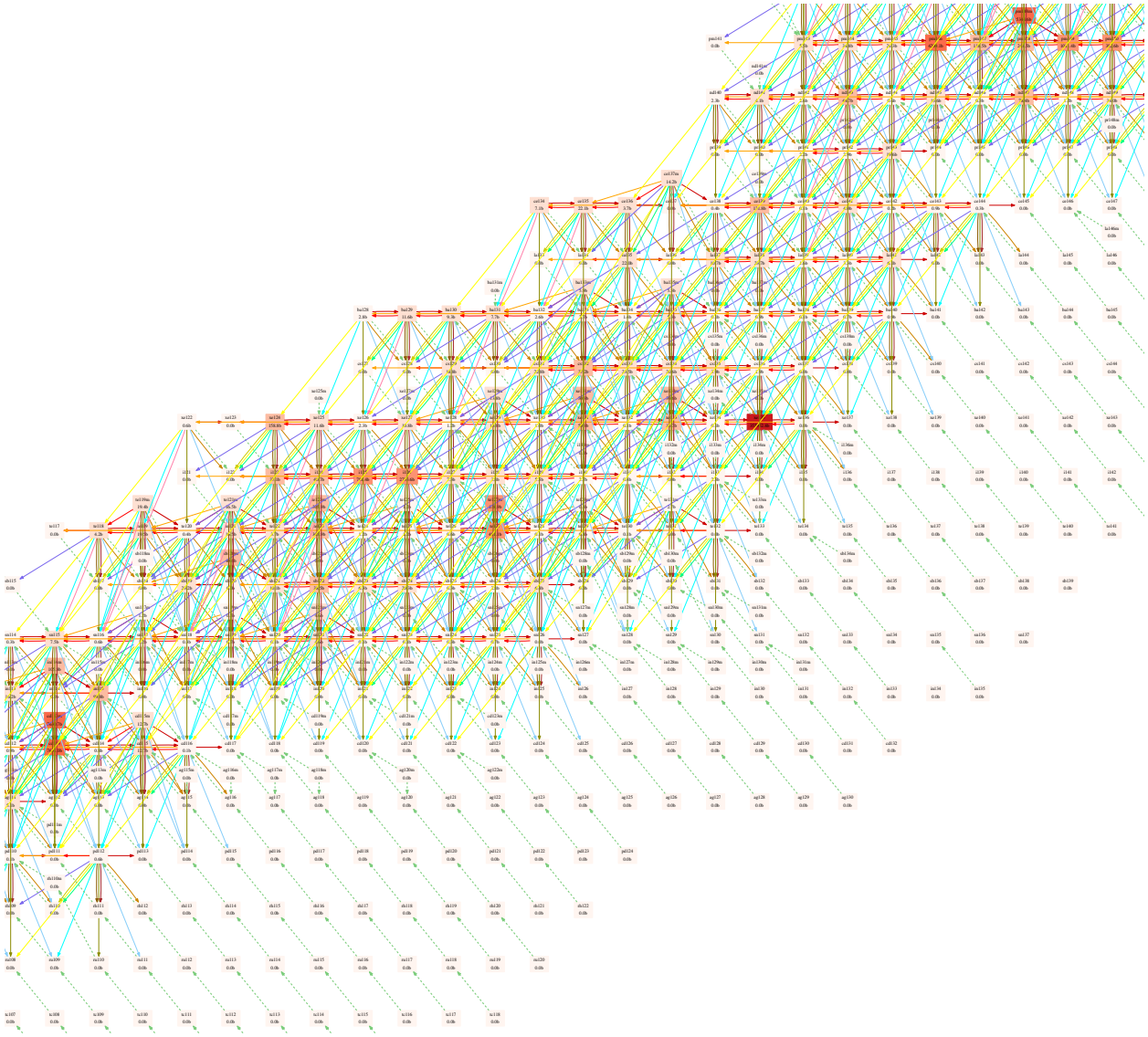


**Figure 2.1:** Subsection of the Transition System (Z=[43,62], A=[107,150]) with all nuclides and transitions except fission and byproduct production displayed. Absorption cross section is indicated by shade of red and transition type is indicated by arrow color, and whether the transition is a decay or a reaction is indicated by a dashed or solid line respectively

# Chapter 3

# Library Reduction Methods

I have developed a general automated method based on the approach and tools of the library reduction for MPACT. This work was done independently from the preliminary work described previously and merely built off of the existing code structures. This method uses the graph representation of the transition matrix and a combination of methods that have been created and tested to determine a nuclides importance to a problem and how the reduced graph system should be structured to best preserve the interactions between these important nuclides.

## 3.1 Graph Translation of Transition Matrix

One of the most important tools developed from the simplified burnup chain development by W. Wieselquist and K.S. Kim is a reliable method to translate a transition matrix from an Origen library to a graph representation as a transition system. A graph in a mathematical sense is a diagram of a set of points (vertices) together with lines (edges) joining certain pairs of these points. In the case of the transition system the vertices are nuclides and the edges are transitions. By representing the matrix in this manner, regardless of the organization of the graph (i.e., how the graph is drawn), two large advantages present themselves. Firstly, this representation makes much more intuitive sense as to how nuclides relate to one another and how the system as a whole is structured. This is very important when trying to discern patterns in such a complex and highly interconnected system such as

burnup chains. Secondly, this representation affords the use of Graph Theory to determine an optimum system with the fewest nuclides with the most importance to the problem being considered.

The transition system represents a cyclic directed graph; that is, a graph that is made of vertices and edges that are only traversed in one direction and contains potential cycles. Following a certain path of directed edges will therefore lead to a vertex that has already been traversed. The direction in the transition system leads from parent nuclide to daughters and the cyclic behavior is a property of the system containing both reactions and decays. This cyclic behavior is also inherited by the system from the potential chain of reactions of certain nuclides.

The transition system also is not universally reachable even though it connected. This means that there is not necessarily a path that connects any two nuclides. This is due to there being no reaction cross sections for the elements Astatine (Z=85), Radon (Z=86), and Francium (Z=87) that would result in an increase in mass. These three elements all lack a long-lived isotope (the longest lived of each being At-110 at 8.1 hours, Rn-222 at 3.823 days, and Fr-223 at 22 minutes half-lives) and primarily are born out of the decay of heavier nuclides, offering no pathways to transition from the light-element nuclides and fission products to the actinides. These properties make the transition system a very difficult graph to traverse by any of the typical algorithms employed in graph theory for finding the shortest path between two vertices, or to find a subgraph of the transition system that results in the smallest subgraph that contains a sufficient number of problem-contributing nuclides.

Within Origen this graph system is defined through a set of classes in object-oriented C++. Explicitly there are seven object types that make up the graph system representation of a transition matrix.

- TransitionSystem (overarching manager of the system)
- Field (type of radiation field and intensity i.e., alpha, electron, neutron)
- Nuclide (ID, gains, losses, the system that it belongs to)
- Species (charge number, mass number, mass)

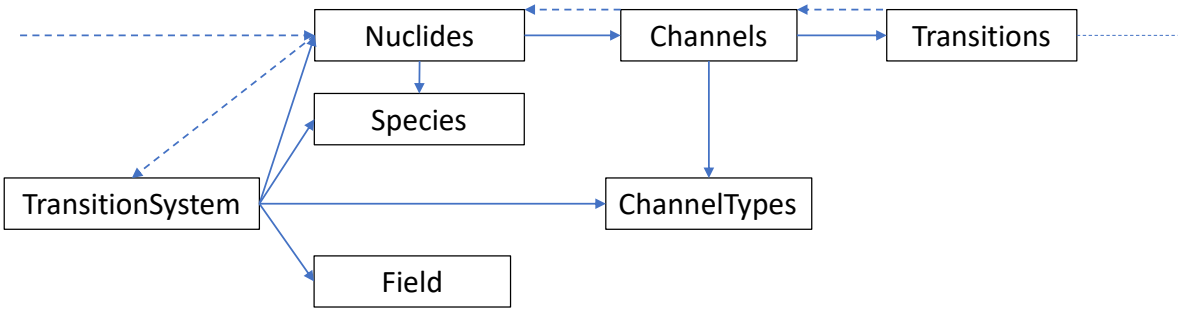



**Figure 3.1:** Diagram of transition system graph classes. A solid arrow indicates ownership (strong pointer) and a dotted arrow indicates a reference (weak pointer). The reference arrow from transitions to nuclides points to a different nuclide that has ownership over the transition

- Channel (channel type, transitions, parent, transition rate)
- ChannelType (decay/reaction MT number, byproducts)
- Transition (yield)

However, three of these are for information storage and platforms for convenience functions, only the four object types of TransitionSystem, Nuclide, Channel, and Transition represent important aspects of the graph diagram of the system. A transition system is an object, defined within Origen, that contains nuclide objects. These nuclide objects then contain loss channels which represent all of the loss transitions within that type of loss, whether it be fission, any decay mode, or any reaction type. The transitions lead from the parent nuclide to the daughter product, but the daughter only has knowledge of this gain without having ownership over it. In the terms of the code structure this means a hierarchy of ownership defined by strong-pointers (pointers that allow the right to change the content of that which they point to) that only follows the nuclide chain through their respective losses. The transition system has ownership of all objects, even if only indirectly (visualized in fig.3.1).

The method to translate the transition matrix, as described in section 1.3, that is used by the CRAM solver to the transition system is fairly straightforward, though requires a thorough understanding of the form that matrix is stored in an Origen library. While the

information is stored on the library and not being used by the solver, the sparse matrix is stored as a group of C++ vectors (a standardized doubly-linked list with bounds checking and dynamic size allocation) grouped by burnup-dependent information such as cross-sections, yields, and collapsing flux spectrum shape along with burnup-independent information like decay and reaction pathways of the system (i.e. the structure of the transition system). These vectors store the information of:

- the identification numbers of the nuclides on the library in the form of SIZZZAAA numbers (S-sublibrary, I-metastable state, ZZZ-atomic charge number, AAA-atomic mass number)
- the number of parents each nuclide has
- how many of those parents are from decay rather than reactions
- the parents' respective position indices within the set of nuclides
- Transition coefficients from parent to daughter nuclide
- Loss cross-sections for the nuclide
- Fission cross-section for the nuclide
- Direct yield from fission of the nuclide
- Kappa values (recoverable energy from absorption and fission reactions)
- the neutron spectrum that was used to collapse the cross sections to a single group

Due to the fact that each nuclide's gain terms are independent in the transition matrix whereas the loss terms are combined into a single term in the diagonal, the creation of the transition system must be bottom up, populating parents and losses relative to a daughter (shown in figure 3.2), rather than the top down architecture inherent to the transition system, where parents and the losses would lead to the daughter products. In this way a nuclide is added to the transition system, then a loss channel (of the type defined by the transition type) is added to the parent nuclide (via lookup of the parent ID from the parent position),

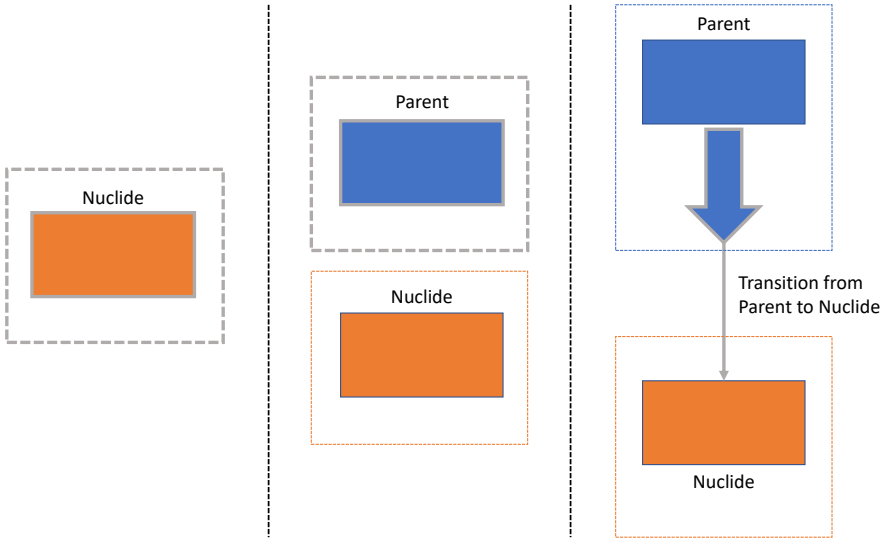
and finally a transition leading back to the originally added nuclide is created, with the yield defined on the transition.

While the details of this translation itself are unimportant for the method of creating the reduced problem-specific libraries, this procedure details the conceptual differences between the structures of the two representations of the same system, both in hierarchy of ownership and the intuitive structure of the system. The translation method also sets the basic framework for editing these transition systems to be designed around problem-specific needs. This then governs how the nuclide reduction method would need to ultimately manage the pointers between the various objects to ensure the reduced system maintains the proper order of ownership and preserves system mass (i.e., not allowing transitions that do not begin and terminate at nuclides like what is possible in the transition matrix).

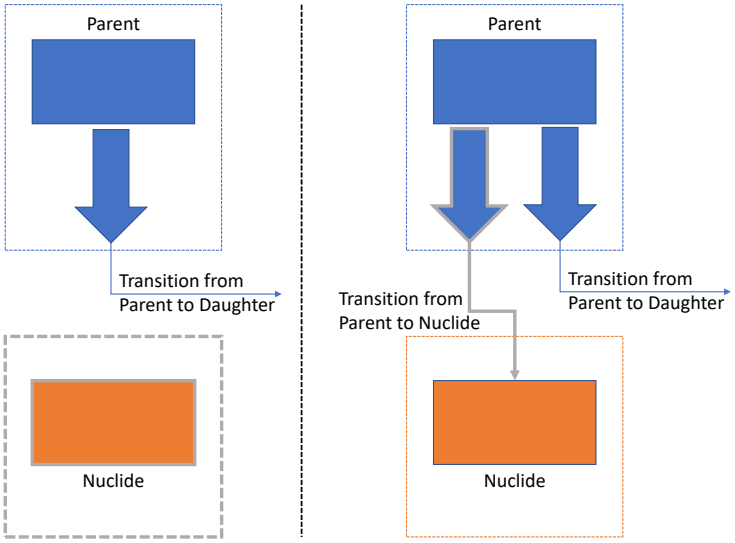
## 3.2 Removals

The method for removing nuclides from the transition system must be done in such a way as to maintain important physical quantities of the system as a whole. Most importantly, mass must be conserved in the system. As mentioned previously, this is done by ensuring that no transitions are made that fail to include both an origin and termination at a nuclide, but more importantly such that no existing transition is edited in such a way that it remains within the transition system after the removal of either its parent or daughter nuclide. The removal of a nuclide from a transition system involves tracing all pointers leading away from the nuclide, through both its loss and gain transitions. This transition relationship to a nuclide can be seen in figure 3.3. The process then proceeds in 4 steps:

1. Deletion of all references not owned by the nuclide targeted for removal
2. Deletion of transitions owned by other nuclides that represent gains into the nuclide to be removed, and if that was the only transition within that loss channel, deletion of the channel as well
3. Deletion of all references to other nuclides, owned by the nuclide to be removed




**(a)** Addition of a nuclide to the transition system in the case where the parent does not exist.




**(b)** Addition of a nuclide in the case where the parent is already in the system.

**Figure 3.2:** Addition of a nuclide to the transition system for cases where a parent nuclide doesn't exist and one where it does. Bold grey outline indicates the added components at that step and boxes indicate ownership of all components originating in the box by the respective nuclide

4. Deletion of all objects that are owned by the nuclide, all pointers and references between them, and finally the nuclide itself

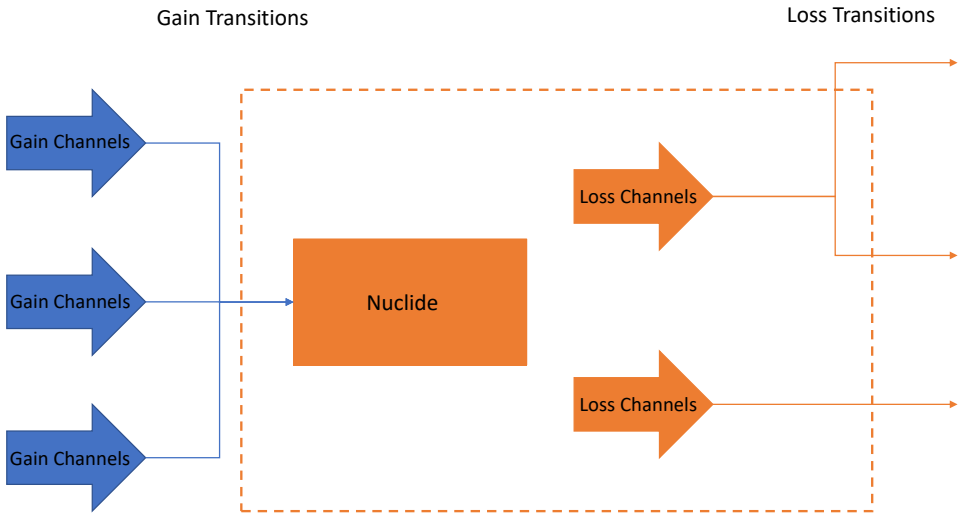



**Figure 3.3:** Diagram of nuclide and transition structure in TransitionSystem, Nuclide has ownership of everything that originates inside the orange dashed lines. Removal from the system entails removing everything leading into and out of the box.

Following the final removal of the nuclide from the transition system, all related information on the transition system is updated to be accurate to its new state; that is, the list of nuclides in the system, the number of gains and losses for all previously connected nuclides, and if the nuclide removed was a fission product its fission yield is distributed evenly to its former daughter products as a cumulative fission yield. The way that the fission yield is distributed is based on fractional transition rates. The transition rate is equivalent to the decay constant ($\lambda_i$) multiplied by the number density set on the nuclide, and for reactions it is the microscopic reaction cross section ($\sigma_{x,i}$) multiplied by the number density set on the nuclide and the intensity or magnitude of the particle field relevant for that type of reaction (i.e. neutron flux, alpha flux, proton flux, or gamma flux). While the redistribution of fission yields potentially results in a larger system (via creation of new fission transition), the goal

is to preserve another important physical quantity, total fission yield, while also preserving as many mass chains as possible.

The total yield of any fission reaction must sum to two, meaning that there must be two daughter products from the reaction (ignoring delayed neutron byproducts). While this could be accomplished through renormalization of the remaining daughter products, that method would result in a loss of a major mass flow vector through a portion of the transition system, resulting in large errors resulting from the removal of fission products at the head of a high-contribution nuclide's mass-chain. These fission products can have a low importance/contribution due to the fact that most fission products are very short lived (on the order of milliseconds) with no reaction cross section information, resulting in very low final masses and contribution factor for almost any metric.

## 3.3 Determination of Removals

Determining which nuclides need to be removed is a combinatorial problem, where the the smallest inventory combination of nuclides that behave within the desired error to the full inventory. Solving for this perfect combination represents an NP-hard problem, that is one where a proposed solution can be validated in polynomial time, but the solution cannot be found in polynomial time. Finding the analytic solution would require $2^N$ evaluations, where N is the number of nuclides in the full inventory. In Origen, this number of evaluations is approximately $10^{662}$, which even on the world's fastest computers would still be evaluating after the heat death of the universe. Even if the assumption is made that each removal is independent of all other removals, the number of evaluations would be $N(N+1)/2$, or approximately 2.5 million for Origen libraries. Even with this assumption (which will later be shown to be incorrect) the evaluation time is prohibitive. Therefore, to solve this intractable problem an alternative method must be used.

The methods developed and presented in this dissertation determine which nuclides need to be removed iteratively, using an interpolating variant of Newton's Method for one dimensional optimization of the reduced-libraries' induced error based on number of nuclides tracked in the library. The initial assumption is made that the relative error is 0% when all

nuclides are being tracked and 100% when no nuclides are being tracked. This assumption is used to set a minimum and maximum bound on the number nuclides removed with the bounds being updated after each iteration. The form used is

$$x_{n+1} = x_U + (Y - y_U) \cdot \frac{x_L - x_U}{y_L - y_U} \tag{3.1}$$

where,

- $x_U$, is the upper bound on library size i.e., lowest number of nuclides in the library with the error not exceeding the threshold $Y$
- $y_U$, is the error in the total metric from the reduced library based on $x_U$ nuclides in the library
- $x_L$, is the lower bound on library size i.e., highest number of nuclides in the library with the error exceeding the threshold $Y$
- $y_L$, is the error in the total metric from the reduced library based on $x_L$ nuclides in the library
- $Y$, is the maximum allowable error in the total metric from the reduced library

Following this method, the contributions of each nuclide based on the original library are stored and act as the function relating nuclide position in the system and $x_n$. The current upper and lower bound on library size and total metric errors at those bounds are also stored to calculate the slope at the current point in the nonlinear function relating minimum nuclide inventory and error in the total system metric. This function, besides being nonlinear and exhibiting stepwise behavior, is also different for each problem type, problem specification, weighting method, subsystem determination method, and library. This therefore precludes the use of quicker methods, including conventional Newton's method, due to the need for its robustness and stability.

This method must also be modified such that it only approaches the optimal system from one direction, meaning the system is only ever reduced from one iteration to the next, never

expanded. This modification prevents the need to reintroduce nuclides and their respective transition to the system. The reintroduction of nuclides is much more inefficient than their removal as it requires a comparison of the entire original system to the current system at every step, rather than just a direct examination of the current reduced library. To accommodate this modification, the transition system for both the upper bound $x_U$ and current iteration's $x_n$ system must be stored, so that in the case that the current iteration exceeds the allowable error, the $x_U$ iteration's system is reduced in the next iteration. The next reduction then attempts to reduce this system again with the altered minimum contribution threshold. There are currently four subsystem estimation methodologies, three native and one external method, and three weighting schemes to determine the nuclide subset that will be used in the $n^{th}$ iteration of the reduced library.

### 3.3.1 Threshold-Truncation Method

The first methodology implemented is the simplest, and that is a direct comparison of the weights of the nuclides. The $x_n$ highest weighted nuclides are included and all other nuclides in question are removed. This is analogous to representing the system as a collection of its highest peaks as seen in figure 3.4.

This methodology is very fast in that it must only make a single sort of all nuclides in the upper bound iteration's library once, thereby requiring the fewest operations per iteration and no operation in successive iterations if the upper bound does not change. However, it requires a greater number of iterations and typically will not result in the optimally reduced system. This shortcoming is due to its inability to consider collections of nuclides beyond what the weighting scheme may do. This often results in eliminating important linking nuclides, significantly altering the mass flow of the system. To compensate for these induced discontinuities in important mass-chains, more iterations are needed to either fix the discontinuities or introduce sufficient unrelated nuclides to compensate for their loss. This issue is less pronounced in library reductions with low error tolerances. This is due to fewer nuclides being removed and a lower threshold (smaller contribution), and thus a lower likeliness of any important linking nuclides falling below this threshold.

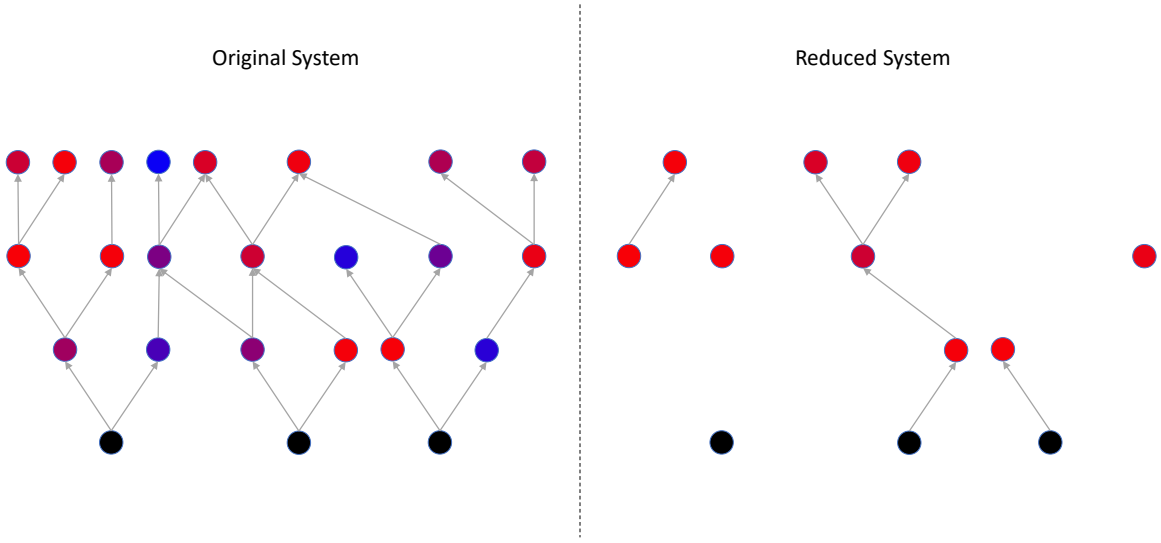



**Figure 3.4:** Diagram of a polytree representation of a simple transition system and the resultant system from using the threshold-truncation method to reduce the system size by 50%. The black nodes indicate source materials and the other nodes are depicted using a blue-red scale with red indicating a higher weight.

Another shortcoming of the direct comparison removal method is the possibility that it may result in the creation of *orphaned* nuclides (i.e., nuclides that have no gains or loss, isolated from the system acting as a disconnected graph with only a single vertex). Due to these orphaned nuclides having no path for either creation or destruction, if the nuclide is not defined as an initial material in the problem, it is also removed from the system regardless of the contribution weight it may have to the problem metric. In figure 3.4, these orphaned nuclides account for nearly half of the nuclides that the method wishes to preserve in the final system. These further removals need to be done after the main reduction step for the iteration. Though they do not result in any additional loss of accuracy in the reduced library, the required checking and removal does increase the computation time. The effects of truncating the transition system in this manor and these shortcomings will be further discussed in chapter 5.

### 3.3.2 Graph Theory Methods

The second methodology is based on taking advantage of the graph representation and behavior of the system to reduce the system to maximum degree by maintaining mass flow chains, rather than considering individual nuclide contribution to the metric. This methodology has resulted in three different schemes to be implemented. The first two are native capabilities in the library reduction code. Applying methods from a field of mathematics pertaining to the optimization of systems, Graph Theory solvers can be applied to find a route connecting the nuclides in the transition system with the highest sum weight (i.e., the lowest error in the reduced library). The optimization finds the minimum number of nuclides that compose a connected subgraph of the transition system that has a sum weight that is within the allowable tolerance from the metric measure of the un-reduced library.

The native algorithms construct multi-rooted polytree graphs (a directed acyclic graph with a underlying tree structure with multiple root nodes), with the roots being the initial materials. As the graphs are constructed they are allowed to (and almost certainly will) intersect into a single polytree (a singly-connected acyclic structure that has a unique path from a "root" nuclide to any other reachable nuclide). The initial materials are used as roots to form these polytree graphs. This is to ensure that all materials defined in the problem remain as part of the system regardless of their contribution so that comparison of the simplified library to the full resolution one is evaluated using the same metric. Secondly, using the initial materials as roots mimics the physical mass flow as the material transmutes following the heaviest weighted paths to their final products. An advantageous feature of this scheme is that it allows any materials to be preserved in the problem (such as any nuclides that may be of interest but not directly related to the problem) by including them amongst the root nuclides.

The non-native algorithm uses the PYOMO optimization module to formulate a integer programming model of the system and uses PYOMO's built-in standard solver libraries to solve it. This method is not coupled to the library reduction code, and is therefore not used in the iterative construct that the native methods use, but it is a far more developed and optimized package for solving similar problems. The PYOMO results are most useful as a

point of comparison to the native method solutions, or as an initial estimate of the final system to inform the native methods.

### Greedy Algorithms

The first graph-based method is a greedy algorithm. Greedy algorithms are an optimization paradigm in which the locally optimal choice is made at each stage in order to find the global optimum. In the context of finding the optimal reduced transition system this algorithm constructs the rooted polytree graphs by determining which nuclides to include in the next iteration rather than which to eliminate. This decision is made by beginning with the root nuclides, including the product with either the highest weight or the lowest mean transition time depending on which mode the algorithm is run in. Determination of this nuclide is done efficiently with the use of a sorted multi-map of potential nuclides keyed on either their weights or the mean transition time. That is a map that allows multiple entries to have the same key value that relates to the entries position in the map, sorted on the key value of the nuclide in descending order of weights or ascending order of transition time. Therefore, the first entry in the map will always be the potential nuclide with the largest potential impact on the system. Once the nuclide has been taken from the map of candidate nuclides and included in the system its loss channels are checked and any unique nuclides, those not already in the system or in the map of potential nuclides, are added to the sorted map of potential nuclides. This process is repeated until the sum of the weights of the included nuclides is within the allowable error to the total metric for the original library. On subsequent iterations, it is the deviation from this total allowable (system-wide) error that is changed based Newton's method, rather than minimum threshold contribution for each nuclide.

The two modes of this method are due to trying to optimize different characteristics in the final reduced library. In the greedy algorithm where the optimal local solution is decided on weight, the goal is to preserve the metric with the fewest nuclides as determined by the weighting method chosen (section 3.4), this mode is illustrated in figure 3.5. One of the drawbacks can be seen in this figure, in that if a low-weighted nuclide precedes a highly weighted branch of the tree this method will avoid the branch entirely. This behavior

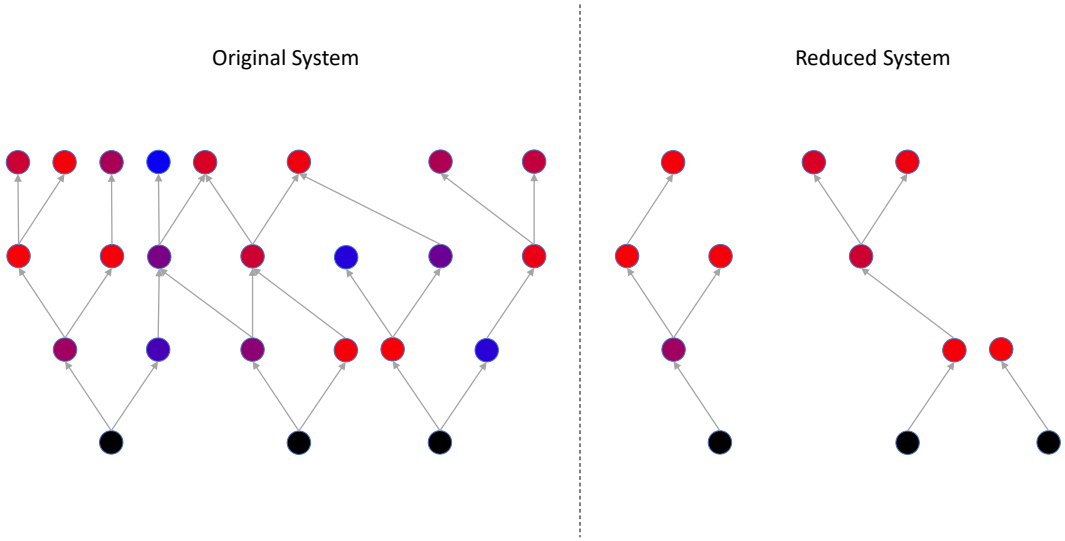



**Figure 3.5:** Diagram of a polytree representation of a simple transition system and the resultant system from using the greedy algorithm with nuclide weight to reduce the system size by 50%. The black nodes indicate source materials and the other nodes are depicted using a blue-red scale with red indicating a higher weight.

can result in larger systems to meet the threshold requirement. In the second mode where the optimal local solution is decided through shortest mean transition time (i.e., the path with the highest transition rate averaged over time for the reducing problem), the goal is to preserve global transition rates through choosing the paths that have the most transitions over the problem and the convergence to a minimal subsystem is determined by the weighting method. While this mode does not have the same shortcoming as the previous one, it does have the shortcoming that the traversal order of the nodes is independent of problem type. The advantages and drawbacks of each of these methods will be demonstrated and discussed further in chapter 5.

### Cutting Planes Algorithms

The second native graph-based scheme for nuclide determination involves the use of the cutting-plane optimization method. This method iteratively sets limiting conditions on the subsystem to restrict the solution space that the optimal system should lie within. It does

this by means of linear inequalities or *cuts*. These inequalities are formulated in the distance-weight vector space of the transition system. This subspace is a linear transformation of A-Z space - in which the systems nodes are located at the indices of their mass number (A) and their charge number (Z) - by mapping a nodes shortest distance to a source node as measured by either number of transitions of total mean transition time ($t_n$) as defined in equation 3.2

$$\begin{aligned} T_{i-1,i} &= \int_0^T \sigma_{i-1,i}(t) \cdot \phi(t) \cdot N_{i-1}(t)dt + \int_0^T \lambda_{i-1,i} \cdot N_{i-1}(t)dt \\ t_n &= \sum_{i=1}^{n} \frac{t_f}{T_{i-1,i}} \end{aligned} \tag{3.2}$$

where,

- $T_{i-1,i}$, is the total number of transitions from nuclide $i-1$ to $i$
- $\sigma_{i-1,i}(t)$, is the burnup dependent (and therefore time) cross section for the reaction for nuclide $i-1$ to $i$, or zero for decay transitions.
- $\lambda_{i-1,i}$, is the decay constant for the decay of nuclide $i-1$ to $i$
- $\phi(t)$, is the time dependent scalar flux
- $N_{i-1}(t)$, is the number density of nuclide $i-1$ at time $t$
- $t_f$, is the total combined irradiation and cooling time of the user defined sample problem for the library reduction
- $t_n$, is the total mean time to transition from a source material to nuclide $n$ along the quickest path

These shortest distances are found in a similar way to the pathing method described in the previous section, in that they both start wiht the source material and traverse the system through the present transition pathways. Starting at the source nodes, which are assigned a distance of zero by definition for both measures of distance, the product nodes are assigned distances either one greater than the parent node or a distance calculated by

equation 3.2. To account for the cyclic nature of the transition system, if product node is found that already has a distance assigned it is either ignored if the new distance would be larger, or re-assigned if it is shorter. In the case where the distance is re-assigned, the nodes products also are then reevaluated to check if their distances also need to be updated to reflect the change to one of their parent nodes.

In this subspace, relationships between nuclide weight and distance (and by extension library size) can be formulated. These relationships are the inequalities that create the cutting-planes that are used to prune the aforementioned polytree structures by setting limb dependent distance and weight minima. These inequalities take the form of relationship stating that beyond a certain distance along the limb of the polytree there are no longer any nodes with a sufficiently high weight for which the node may be required as a path to the source. From these inequalities, terminal nodes are determined and the pathing from the terminal node to the source nodes follows the tree structure. It should also be mentioned that it is also commonly the case that nuclides with high weights are not terminal nodes; that is, they will lie along the path of inclusion for another nuclide. Thus this is not an exceptional case, and this method will not necessarily result in a substantially larger set of nuclides to track in the reduced library. An illustration of the results of this algorithm, when distance is determined by number of transitions, can be seen in figure 3.6. From this figure it can also be seen how equal-length parallel paths are handled. During the pathing of polytree if multiple paths are found of equal length, then all paths are tracked until the optimization is evaluated. During the optimization, the path lengths are evaluated by the distance to the nearest preserved node, not by distance to a source node. This results in nodes pathing through other nodes that are to be preserved, minimizing the size of the optimal system. A consequence of this can be seen in figure 3.6, where several nodes have two paths of equal length to a source node, with only one path being preserved due to there being another preserved node along that path.

This algorithm type compensates for many of the shortcomings of the other native methods, but the purpose of the two methods for determining distance between nodes is similar to that of the reasoning behind the two modes of the greedy algorithm methods. However, in this case using mean transition time in this algorithm type results in many of

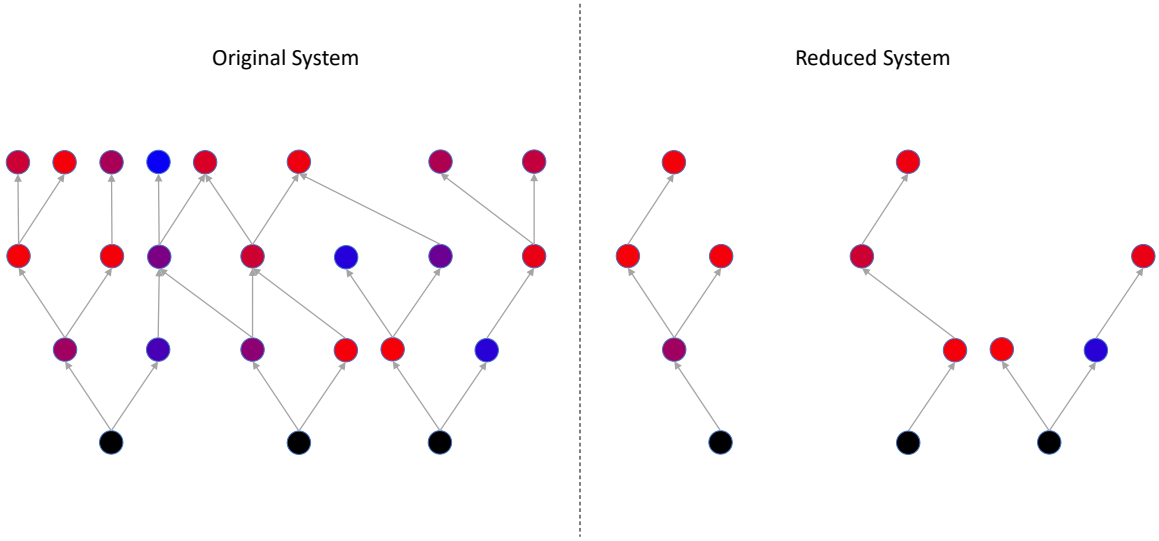



**Figure 3.6:** Diagram of a polytree representation of a simple transition system and the resultant system from using the cutting-planes algorithm to reduce the system size by 50%. The black nodes indicate source materials and the other nodes are depicted using a blue-red scale with red indicating a higher weight.

the same difficulties that arise from the stiffness of the matrix in matrix inversion. The matrix eigenvalues can vary by many orders of magnitude and so can to the mean transition times. While using this method for determining distance does result in the path of most frequent creation, it makes a preliminary estimation of final system size difficult. This difficulty makes the subspace transformation more computationally expensive and the formulation of the *cut* inequalities slower. However, there is also no inherent relationship between the number of transitions to reach a nuclide and its importance to the system. Despite this, these methods offer a significant improvement over the previously mentioned methods, as will be demonstrated in chapter 5.

### PYOMO Integer Programming Formulation

The last of the system determination methods, and the one method that is not natively implemented, is an integer programming formulation of the system optimization problem in the Python Optimization Modeling Objects (PYOMO) module. PYOMO supports the

formulation and analysis of mathematical models for complex optimization applications within the framework of a high level programming language (Python) that allows for more complicated formulations and embedded analysis over other algebraic modeling languages like AMPL, AIMMS, and GAMS [14]. Optimization models are mathematical models that include functions that represent goals or objectives for the system being modeled and constraining functions that limit that objective. In the case for library reduction, the optimization is set to maximize the preserved transition rate and can be formulated as a multivariable binary programming problem. This problem is formulated as:

$$
\begin{aligned}
MAX: & \sum_{\{i,j\}\in Y} y_{i,j}T_{i,j}\,, \quad \text{given} \\
& \sum_{i\in X} x_i W_i > 1-\tau \\
& \sum_{i\in X} x_i \leq N \\
& y_{i,j} \leq x_i \quad \forall\{i,j\}\in Y \\
& y_{i,j} \leq x_j \quad \forall\{i,j\}\in Y \\
& x_i = 1 \quad \forall\{i\}\in S \\
& x_i \leq \sum_{\{i,j\}\in Y} x_j y_{i,j}
\end{aligned}
\tag{3.3}
$$

where,

- X is the set of nuclide indexes
- Y is the 2-dimensional set of indexes of nuclides connected by a transition
- S is the subset of X consisting of the indexes of the source nuclides
- $x_i$ is the binary variable of whether nuclide $i$ should (1) or should not (0) be included in the optimal solution found
- $y_{i,j}$ is the binary variable of whether the transition connecting nuclide $i$ and nuclide $j$ should be included in the optimal solution

- $W_i$ is the weight of nuclide $i$
- $T_{i,j}$ is the mean transition rate of the transition connecting nuclide $i$ to nuclide $j$
- $\tau$ is the user defined error allowed in the system
- $N$ is the maximum number of nodes - nuclide - allowed in the solution

The inequalities in this formulation impose the following constraints to the optimization condition of maximizing the transition rate int eh subsystem.

1. The weight of the nuclides included must be expected to exceed the threshold for the user required accuracy of the reduced library
2. System must have fewer than, or equal to, the maximum number of nodes $N$
3. A transition can only be included in the system if the product of that transition is also included
4. Similarly a transition can only be included in the system if the parent of that transition is also included
5. All source nuclides must be included in the final system
6. A node can only be included in the system if its parent nodes is also included

These constraints serve the purpose that reduced library must be sufficiently accurate and reduced (constraint 1 and 2 respectively), that all nuclides and transitions in the system must be physically representative of the process they represent (constraints 4 and 5), and all nuclides must have a creation pathway that is connected to a source node (constraints 6 and 7). While most of these constraints are self-evident, the last one requires elaboration. This constraint is to ensure that the subsystem is fully connected and the cyclic nature of the system does not result in isolated subsystems. It is implemented as a recursive function that checks the parents of the nuclide in question for inclusion in the system, as well as the parents of those nuclides so on and so forth until a source nuclide is reached, which are included in the system regardless of any other constraints. This recursive function circumvents cycles

by building a list of all nuclides that are in each level of recursion and passing that list to the next level. At each level of recursion, the list of nuclides checked at the previous level is passed in in order to exclude them from subsequent checks, thereby breaking the cycle.

This method exploits the extensive optimization and testing of the solvers in this package, the extent of which is infeasible to replicate in any of the removal determination schemes that are implemented natively in the library reduction method. The use of the PYOMO software enables constraints on transition rate, library size, and library accuracy that is not constrained in any of the other system determination methods, either because it is not coherent with the method or because its inclusion would complicate and slow the method when it is intended to be a simpler and faster option. These constraints are important as the manner in where the connectedness of the system, and its physical meaning, is preserved is one of the biggest advantages to this method for library reduction. This method likewise leaves the decision of whether the priority in the library reduction is limiting library size or preserving transition rate and thus physical fidelity. However, the fact that this method is not being implemented natively within the Origen code system reduces its future potential utility, until a time in which it can be incorporated. Until such a time though, the method acts as a point of comparison for the efficiency of the native methods, shown in chapter 5, and as a proof of the compatibility of general use software to this approach.

## 3.4 Weighting and Problem Type Metrics

Determination of the nuclides to be eliminated is always based on time-integrated quantities. Each nuclide's final weight is determined as the sum of its contribution at the middle of each depletion step, scaled by a user-defined time weighting factor. As such, the selection of nuclides for elimination is always based upon a time-integrated quantity. These weights that are calculated are always based on physical, macroscopic quantities; metrics of interest that need to be preserved in a system are inherently macroscopic, as those are the quantities that affect a system. These weights are defined as a physical property proportional to the system property to be preserved; in essence, representing a response function. These responses can be a scalar multiple of the nuclide number density (representing factors such as activity

or decay heat) or integral quantities (such as contributions to emitted radiation spectra); examples of this are enumerated in Table 3.1. Weights are always compared as dimensionless quantities as a fraction of the total metric. Also, the different weighting schemes apply different methodologies in how to capture how the physical quantity affects the system.

The applications that were considered for the development of metrics of interest were those that most often require a large number of burnup calculations. These applications are

- Nonproliferation, security, and verification
- Health physics and shielding
- Fuel Cycles
- Neutronics

Each of these applications have certain quantities that are of common importance and it is for these quantities that metrics are considered, with these metrics and their respective weighting factors being listed in table 3.1. The weighting factors are chosen in most case to be the physical constants that determine the rate at which an important transition type happens; an example would be the decay constant determining the decays per second or activity and gamma source then being rate at which these decays result in a gamma ray with an energy within the energy bounds. Out of the weighting factors listed though, criticality stands out in that it consists of two physical quantities of importance: the fission cross section and the absorption cross section. Despite a majority of nuclides not having a fission cross section, many contribute to the total absorption cross section and therefore must have a weight. In order to rectify this, the criticality weighting factor is treated differently than those of the other problem types. Each nuclides weight is calculated by equation 3.4, which is derived from the equation for propagation of uncertainty under the assumption the error tolerance is equivalent to an uncertainty from the truncation of the system.

**Table 3.1:** Examples of candidate problem-based weighting factors for the given error to be minimized

| Weighting Factor | Problem |
|---|---|
| $\lambda_i$ | Total Activity |
| $\frac{\nu\sigma_f}{\sigma_a}$ | Criticality |
| $\int_{E_1}^{E_2} \gamma_i(E) \cdot \lambda_i dE$ | Gamma Energies |
| $\sigma_{a,i}$ | Depletion[15] |
| $A_i$ | Total Mass (fissile mass is a subset of this metric) |

$$w_i = \sqrt{\left(\frac{\nu_i\sigma_{f,i}}{\sum_{j=0}^{N}\nu_j\sigma_{f,j}}\right)^2 + \left(\frac{\sigma_{a,i}}{\sum_{j=0}^{N}\sigma_{a,j}}\right)^2} \tag{3.4}$$

To apply these weighting factors to the determination of nuclides vital to the problem three weighting schemes were implemented. I have chosen these schemes, based on the current transition matrix reduction methods mentioned previously, due to their applicability to the graph system, the ease of application and use, and most importantly the capacity of the scheme to take into account the physical meaning and properties of the transition system.

### 3.4.1 Contribution Weighting Method

The first method for determining a nuclides problem importance is contribution weighting. This method is analogous to the simplest transition matrix reduction method, the physical attribute-based extraction matrix. This method calculates the weights of the nuclides at each mid-step point of the forward depletion calculation and then translates this to a dimensionless quantity as the fraction of contribution to the total metric. This method uses the time-integrated contribution of a nuclide to the problem metric as the only contributing factor of the importance of a nuclide to a problem. In mathematical terms this is expressed in equation 3.5.

$$W_i = \int_0^T w_i(t) \cdot n_i(t) dt$$
$$q = \sum_i W_i \tag{3.5}$$

Where,

1. $w_i$ is the nuclide weighting factor
2. $n_i$ is the nuclide number density
3. $W_i$ is the weight of the nuclide
4. $q$ is the total problem metric

While this method is fast and the most straightforward and obvious approach, its shortcomings rule out its use in all but the most basic cases. These shortcomings stem from this approach not taking advantage, or even consideration of, the interconnectedness of the system and the inter-nuclide relationships inherent in the system. This is the result of the assumption that a nuclide only contributes to a problems properties via its direct physical interactions with the system rather than also being a potential pathway for the creation of other nuclides. The most direct result of this is that contributing transitions can be eliminated due to the daughter of the transition not being weighted.

Another application of this weighting method is to weight the problem-relevant transition. Due to the library reduction method being nuclide-centric, this translates to applying the weight calculated in equation 3.5 to both the parent and product of the problem-relevant transition as both are necessary in the final library to maintain the contributing attribute of the parent nuclide. This application of the method does yield better results than the standard method in that contributing transitions are preserved independent of the subsystem determination method and does not rely on it to implicitly maintain the transition, however this also treats a nuclide's only importance to the system as its contribution to a transition of direct importance to the metric.

### 3.4.2 Distributed Contribution Weighting Method

To no longer make the assumption that results in the contribution weighting method (i.e., that a nuclide only contributes to a problem directly through its direct physical interactions with the system), the equation to calculate the weight of nuclide must be expanded to include higher order terms. This generalization of the contribution weighting method is the distributed contribution weighting method. This method is designed to preserve the transition rates of the contributing transitions, rather than just the transitions themselves, by weighting the nuclides that contribute to the parents number density and the competing loss transitions proportionally to their branching ratios. The mathematical representation for the first degree distributed contribution weighting is given in equation 3.6.

$$\begin{aligned} W_i &= \int_0^T w_i(t) \cdot n_i(t) + \sum_d w_d(t) \cdot n_d(t) \cdot \frac{a_{i\to d}}{\sum_j a_{j\to d}} + \sum_p w_p(t) \cdot n_p(t) \cdot \frac{a_{p\to i}}{\sum_j a_{p\to j}} dt \\ q &= \sum_i W_i \end{aligned} \tag{3.6}$$

In equations 3.6 $W_i$ , $n_i$ , $w_i$ , and $q$ have the same meaning as in the contribution weighting method (eq. 3.5). The subscript $d$ indicates that the variable is with respect to the daughter product of nuclide $i$, as all daughter products' weights contribute to the importance of the nuclide. The subscript $p$ indicates that the variable is with respect to the parent predecessor of the nuclide. The daughter's weight is fractionally split amongst all of its parent nuclides by the fractional transition rate, where $a_{i,d}$ is the transition rate from nuclide $i$ to nuclide $d$ and $\sum_j a_{j\to d}$ is the total rate at which nuclide $d$ is created from all parent nuclides. Similarly, the parent's weight is fractionally distributed to the nuclide by the portion of the parents losses that result in the nuclide.

To expand this expression to any higher degree, a term is added for each transition step further that needs to be accounted for (i.e. a second-degree weighting would have a term for the daughter products daughter products). The implementation of this weighting scheme is to make each nuclide's weight a vector quantity, iteratively calculating weights at

higher degrees of removal. Each iteration in the implementation represents a higher degree weighting and requires iterating over every daughter for each nuclide.

Although this method is formulated as nuclides weight being increased by its proximity to others (i.e., an out-to-in approach) it is implemented as a nuclide distributing its weights to parents and daughters (i.e., an in-to-out approach). This is to accelerate the method through taking advantage of the early stages not having a significant number of nuclides with a non-zero weight; rather than the entirety of the system needing to be iterated over only the few with weights need to be considered. Taking this approach is also useful in the cases where a non-symmetric weighting is desired, that is having a different distribution degree for daughters than parents. In the context of the distributed contribution method preserving the transition *rates* of interest, the theoretical limit for distributing to daughters is the first degree as they are the only products that influence the nuclides number density and thus the transition rate of interest. However, there is no theoretical limit to the degree that parents should be weighted at, as even a parent that is many degrees removed from the nuclide of interest will have some measure of importance to the creation rate.

In theory, the optimal number of iterations varies for each burnup chain, and as such the higher degree weights should cease to be calculated for a chain when the previous iteration's weight falls below a set threshold relative to the desired accuracy of the system. The final nuclide weight for determination of nuclides to remove from the transition system would then be the sum of the direct and higher order weights integrated over time. This method is similar to the SVD method in that it increases a nuclides weight (importance) based on its proximity to nuclides that have a high direct contribution similar to how the information matrix is used to include the nuclides that are important due to the burnup chain that they are a part of. Also, similar to how the SVD approach with the transition matrix requires multiple iteration to fully capture the inter-nuclide interactions, the forward higher degree weighting must also be done iteratively to capture anything other than the relationships of the nearest transitional neighbors.

Capturing the higher-degree interactions is prohibitively time-consuming and in practice detrimental to the accuracy of the weight as a measure of the nuclide importance to the system. This is especially true in the most important case where the relations of an entire

burnup chain are captured. In the case of fission (and some minor actinide decay chains), this may require 20 or more iterations over the entire system for minimal gains over the contribution weighting method. As will be demonstrated in chapter 4, this is due to the diffusion of the weight through the system, obscuring the origin of the weight at the directly contributing nuclides and transitions. Due to this, the true optimal weighting order for this method is around first – or second – order. This degree of weighting balances between preserving the transition rates of the contributing transition and this weighting diffusion effect. At these orders the improvement over contribution weighting and the complete system distributed contribution weighting method are the most significant.

### 3.4.3 Adjoint Weighting Method

Taking this idea of accounting for the connectivity of the transition system and propagating nuclide importance through this system to preserve the rate of creation of the nuclide of interest to its terminus, you arrive at the adjoint weighting method. Using the solution of the adjoint depletion equation for the purpose of determining nuclides important to a system was first suggested by Gandini [10] in 1975. Gandini suggested, and demonstrated, that the sensitivity of an isotope density's first derivative (with respect to the accuracy of experimental values for nuclide cross-sections and decay rates) at a given final time could be found by exploiting the time-dependent generalized perturbation techniques. Required for this technique are the real and adjoint solution to the depletion equations, which are then derived in [36, 37] and stated here in equation 1.11.

The application of this to determining important nuclides is described in [4], where in the case of coupled neutronics and depletion calculations contribution functions are defined based on the change in average neutron flux in a fuel region. These contribution functions are the product of the adjoint nuclide number density, with the response being a defined change to the nuclide number densities, and the real nuclide number densities normalized to a standard relative change. In this work it is found that for a perturbation, or more generally a response function, the adjoint nuclide number density function as an importance function for a nuclide and the number density function as a contribution function. This finding is widely accepted in the field, though not commonly used due to the difficulty of implementing

an adjoint depletion solver in most depletion codes. This difficulty has already been overcome with previously mentioned adjoint capability with the CRAM solver in Origen.

In this application of using adjoint weighting, the contribution weighting function is used as the response function. With the adjoint depletion equation being the time inverse of the depletion equation, and the initial adjoint nuclide density distribution being the forward direct weights, the adjoint solution at any time will be a distribution of the origins of those final weights. Where the forward higher-degree weighting method distributes weights is based on number of transitions removed. The adjoint weighting distributes weights by decay and reactions that happen in the problem, making it a time-based distribution of importance rather than a just an overall system-based distribution. This method requires an iterative approach to weighting, solving the adjoint depletion problem for each nuclide, making it slower than the other weighting methods but more mathematically rigorous, and more physically meaningful. The mathematical representation for this weighting scheme is derived from the adjoint calculation of a system response in perturbation theory.

$$\begin{aligned} R_n &= \left\langle \overline{N}(n,t), \overline{N}^*(n,t) \right\rangle \; = \int dt\, \overline{N}(n,t) \cdot \overline{N}^*(n,t) \\ q &= \sum_n R_n \end{aligned} \tag{3.7}$$

with,

$$\overline{N}^*(T_f) = \begin{cases} w_0(n) & \text{for n} \\ 0\ , & \text{otherwise} \end{cases} \tag{3.8}$$

Where in these equations $R_n$ is the response to the adjoint initial nuclide number density being equal to the contribution weight 3.8 for the nuclide $n$. The weights for the nuclides are the time integrated nuclide dependent contributions to the response vector. These weights are derived from equation 3.7 in equations 3.9 and 3.10 by expanding dot products and rearranging the order of summation.

$$
\begin{aligned}
R_n = \int dt\, \overline{N}(n,t) \cdot \overline{N}^*(n,t) = \int_{t=0}^{T} dt \sum_{m=0}^{M} N_m(n,t) \cdot N_m^*(n,t) \ , \quad \text{by definition} \\
\overline{N} \cdot \overline{N}^* = \sum_{m=1}^{M} N_m \cdot N_m^*
\end{aligned}
\tag{3.9}
$$

$$
\begin{aligned}
q = \sum_{n=1}^{M} R_n = \sum_{n=1}^{M} \int_{t=0}^{T} dt \sum_{m=0}^{M} N_m(n,t) \cdot N_m^*(n,t) \ , \quad \text{rearranging} \\
= \sum_{m=0}^{M} \int_{t=0}^{T} dt \sum_{n=1}^{M} N_m(n,t) \cdot N_m^*(n,t) \ , \quad \text{therefore} \\
W_m = \int_{t=0}^{T} dt \sum_{n=1}^{M} N_m(n,t) \cdot N_m^*(n,t)
\end{aligned}
\tag{3.10}
$$

The resultant weight of each nuclide then is the sum of its contributions the creation of all other nuclides, and can be seen to match the form of the time integrated weights in the other weighting methods.

This weighting method results in a more distributed weight, while maintaining a majority of the weight in the directly contributing nuclides. However, the diffusion of weight still impacts the convergence of some system determination methods. This is because the weights are static as those methods evaluate the system. The weight attributed to a nuclide due to its contribution to the transition of interest should not be counted if the path that leads from that nuclide to the transition is broken. This is most significant in the case were the daughter product of the transition is removed from the system. This can happen due to the adjoint weighting method not attributing any weight to the daughter product. This can be accounted for by either changing the initial adjoint number density from the contribution weight to the alternated transition contribution weight discussed in section 3.4.1 or by applying the first degree contribution weighting for the daughters of the directly contributing nuclides (referred to as Corrected Adjoint Weighting in chapter 4). However, both of these are ad

hoc methods, thus it is better to account for this behavior through the use of a subsystem correction method.

## 3.5 Subsystem Estimation and Correction Methods

To either accelerate the determination of an adequate subsystem or to correct the subsystem found from the previously mentioned methods, further methods have been applied to this library reduction problem. The accelerating methods take advantage of either the behavior of the nuclides in concert with the user defined problem, or the data structure properties of the libraries in Origen and similar depletion codes. These methods accelerate the reduction by preconditioning the system by removing nuclides that are predicted to have the least effect on the system. The correction methods, however, are to account for behavior of the transition system that is non-congruent with the subsystem determination methods and the weighting methods outlined previously. This behavior arises from the many parallel pathways that can be taken to connect two nodes and the change of behavior of the system in restricting these potential pathways. The effects that the use of these methods have on the results of the library reduction method will be demonstrated in chapter 6.

### 3.5.1 Competition Correction

As was alluded in the adjoint weighting section, it is beneficial to correct the reduced transition system to account for the potential losses along a mass chain that do not contribute to any transition that has a metric contribution. These so-called *parasitic transitions* account for all mass flow in the system that does not contribute to the metric. These are inherently not captured in any of the weighting algorithms or the subsystem determination algorithms because they were developed for the goal of capturing a nuclide's direct and indirect importance to a metric and determining the most efficient way to connect these nuclides within the bounds of the transition system, respectively. These parasitic transitions then, being a detraction from rather than a contribution to the problem and being divergent from important mass flow chains, should not be attempted to be captured by those methods.

They can also represent a significant source of error in highly-reduced systems, due to them being tightly constrained to the paths of highest contribution.

This method corrects for the loss of these pathways that compete with the metric-contributing pathways through a comparison of the optimally-reduced system and the fully-populated transition system. For every nuclide in the reduced system, the loss pathways that are preserved are compared to those that are being removed and if it is found that a significant difference to the total loss rate of the nuclide of interest would be present in the reduced system, the most significant of the parasitic transitions are no longer removed. How to determine what constitutes a *significant* difference in the loss rate is further explored in chapter 6, but nuclides that are preserved through this function often represent a greater reduction to the error than would be seen by nuclides preserved by the weighting and subsystem determination methods in the reduction regimes that this method is deployed in.

### 3.5.2 Path Correction

Similar to the competition correction algorithm the path correction algorithm seeks to preserve select pathways that have been removed from the system. However, rather than preserving transitions that do not contribute to the problem metric directly, significant gains to the accuracy of the reduced system can be made by analyzing the reduced systems contributing pathways that have been removed. The path correction algorithm compares the found optimally-reduced system to fully-populated transition system for potential broken mass flow paths for problem contributing nuclides that have a single nuclide removed from the complete path. These are parallel paths that are less significant than the preserved path, but contribute to the accuracy of the rate of an important transition. These paths are preserved in the case that a parent to an important nuclide is not to be preserved, and a nuclide that is to be preserved in the reduced system has a potential transition to that parent. In this case the parent nuclide is decided to be preserved as well.

The circumstances for this correction to have a significant impact on the accuracy of the final system are more limiting than the previously discussed, competition correction. This correction should only be used in the case that the weighting method used is either

direct, first, or second-degree contribution weighting. This is because both the higher-degree distributed contribution weighting methods and the adjoint weighting method account for these parallel pathways in the weight. The correction should also only be used when the system is sufficiently reduced (i.e., after the final reduction iteration), or else the number of preserved nuclides from this method will be too large. Both of these restrictions arise because the behavior that this correction is for is resultant from the native graph methods treatment of the transition system as a polytree to avoid cycles. Even then however, if the graph theory methods are used in the alternative modes where transition rates are considered (as opposed to library size) this correction becomes redundant. Despite these restrictions, this correction can be used to increase the accuracy of the subsystem if the user has opted for the quicker reduction methods.

### 3.5.3 Removal of Duplicate Nuclides

Despite describing the transition system as a single graph in the previous descriptions, the transition system in practice is two non-connected graphs. This division is based on the determination of sub-libraries in the Origen libraries. Origen tracks the origin of certain materials independently through these sub-libraries. These elements are ones that can be an activation product of one of the original inventory, or a fission product nuclide being created through the fission of heavy nuclides and their corresponding decay products. These two sub-libraries describe everything that is not considered in the actinide sub-library (i.e., a nuclide with $Z \leq 87$). Origen's sub-libraries are described in the table 3.2. The graph representations of the transition matrix separates the light-element nuclides into an independent graph to maintain this sub-library information. The two graphs are treated independently, but these duplicate nuclides in the two graphs are physically identical, and the separation is not necessary in most cases. As such, the two graphs can be unioned, reducing these duplicates to a single instance in the fission product sub-library, with the exception of a few very light nuclides that must be preserved in the activation sub-library due to restrictions in the Origen material irradiation framework. This represents a free reduction in the problem with no loss of accuracy in the system.

**Table 3.2:** Sub-libary specification

| Integer ID | Short ID | Description | Purpose |
|---|---|---|---|
| 1 | LT | Activation/Light Nuclides | Non-actinides that are present in materials before irradiation or created through activation |
| 2 | AC | Actinide/Heavy Nuclides | Heavy nuclides ($Z \geq 88$) |
| 3 | FP | Fission Product Nuclides | Nuclides produced by fission directly or indirectly |

This removal can be done *a priori* if no material in the user defined reduction problem is declared to be an activation nuclide. The standard Origen library has approximately 500 such duplicate nuclides that can be eliminated prior to any weighting or evaluation of the transition system. This prior estimation of the optimal subsystem can accelerate the subsystem determination methods and effectively *precondition* the transition system and improve the initial estimate of the optimal reduced system size. This accelerating method does not change the outcome of the library reduction, as the duplicate activation/light nuclides would have a zero weight. However, if a material is declared within the activation/light nuclide sub-library this accelerating method cannot be used as it would have to change user defined parameters to maintain the zero-weight assumption.

### 3.5.4 Removal of Short-Lived Fission Products

To further accelerate the optimization of the reduced transition system another group of nuclides can be targeted for a priori removal, short-lived fission products. These nuclides are direct fission products that have no cross section data and typically only have one or two decay paths, but most importantly, these nuclides have half-lives that are on time scales many orders of magnitude less than the time scale of the problem. In the full transition system visualization in Appendix A figure A.1, these nuclides can be distinctly seen as a large section below the band of more stable nuclides. These nuclides therefore decay to insignificant quantities too quickly to meaningfully contribute to the problem. The threshold for the length of half-lives for nuclides to be removed by this method is dependent on the time discretization of the user defined problem. Depending on the problem defined and its time

resolution, upwards of 200 nuclides can be removed from the system with this acceleration method.

This method determines which nuclides can be removed by iterating through all nuclides with a fission yield and calculating the nuclides half-life as a fraction of the shortest time-step in the user defined problem. If the half-life is less than $\frac{1}{20}$ of the length of the time step, the nuclide is removed and its fission yield redistributed to its daughter products. This fraction was determined to be sufficient in most cases (see chapter 6) as less than 0.1% of the original material will remain at the midpoint depletion calculation. However, this threshold results in very few removals in problems that occur over short time spans - like a pulse irradiation - and when the problem is discretized into small time steps. Conversely, this threshold over conditions the transition system in cases where the user has opted for a more coarse time description of the reducing problem (the consequences for choosing either a fine or coarse description of the problem are also presented in chapter 6). As was the case with the duplicate removal acceleration method, the use of this method is optional and should not affect the resultant optimized transition system.

# Chapter 4

# Problem-Dependent Transition System Behavior

As described in the previous chapter, the transition system is the graph representation of the transitional relationships of nuclides. Through analysis of behavior the transition system, in the transition rates and relative contributions of specific reaction pathways it is possible to determine the underlying relationships and important factors in how the problem-specific metric is generated. This is done by altering the user-defined problems and weighting methods, and then analyzing the resultant changes in the relative importances of the constituent nuclides. A nuclide's calculated importance or weight only has meaning relative to other nuclide's weights, as only that has a significant impact on what nuclides will be included in a final reduced library of a given size. Also, a variance in the ordering of nuclides by weight resulting from changing the problem definition is indicative of how specific to that problem the reduced library will be and when particular care needs to be taken in the selection of weighting methods and subsystem determination methods.

A representative sample problem (figure 4.1) is used for all of the analysis in this chapter, unless stated otherwise. This problems material composition is of uranium oxide fuel that is 5% U-235 by weight, a gadolinium burnable absorber, and aluminum cladding and/or structural material. The irradiation history is given to be three power cycles, each one year in length, at 40 MW/MTU power density with 20 days inter-cycle cooling and a final decay time of 600 days after the final power cycle. This problem was constructed to loosely match

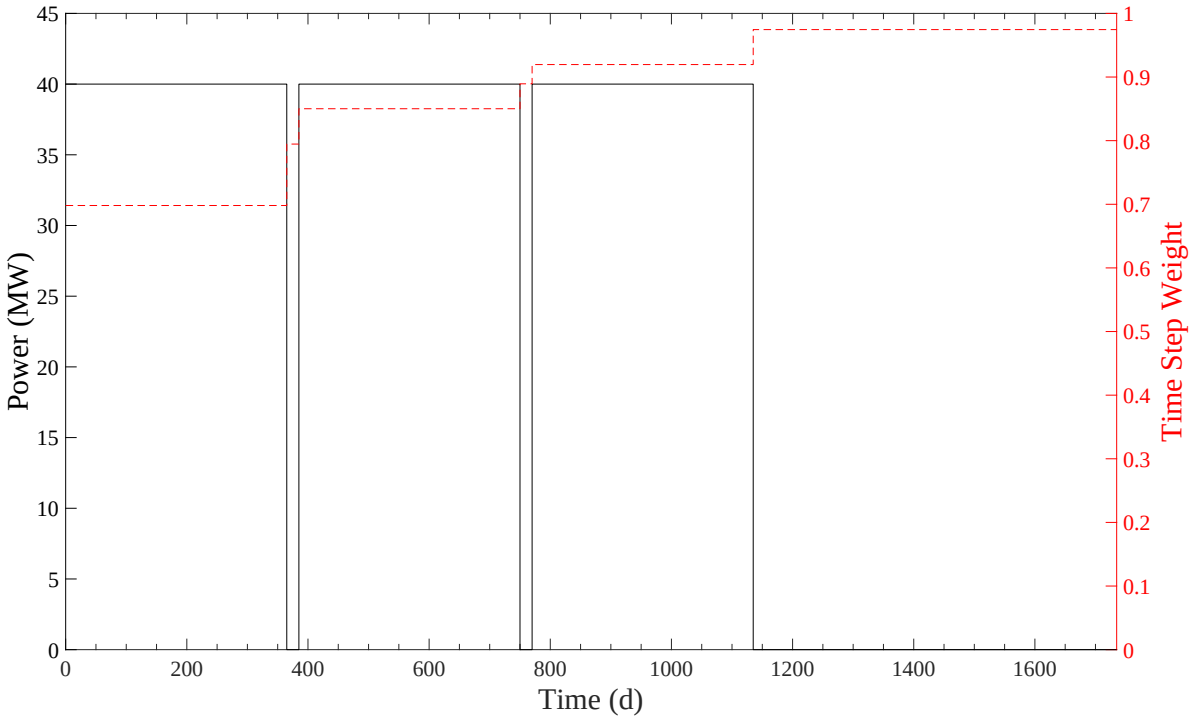



**Figure 4.1:** The irradiation history and time-step weighting for the default problem.

reactor conditions in a variety of reactors and to have starting materials in each of the three sub-libraries in Table 3.2. This sample problem also uses the default logarithmic time-step weighting (Eq. 4.1).

$$W_{t_i} = \frac{ln(\frac{t_i + t_{i+1}}{2})}{ln(t_f)} \tag{4.1}$$

Where,

1. $t_i$ is the problem time at the beginning of time-step $i$
2. $t_f$ is the total problem time
3. $W_{t_i}$ is the weighting given to time step spanning from $t_i$ to $t_{i+1}$

This weighting is chosen as the default behavior because it considers all time steps but with emphasis on the accuracy of the metric at times closer to the final time of the problem. While not all metrics should be considered at all times due to them having little importance

during the conditions of the steps — such as gamma source during irradiation — for testing purposes it is important to be able to observe their behavior under all conditions. Placing a higher importance on the later time steps is also due to most of these problems attempting to recreate present observables in used nuclear fuel.

## 4.1 Problem Type

Though there are more problem types, and therefore problem metrics, that can be used to create reduced source terms than are covered in this dissertation, they are degenerate cases of those metrics in table 3.1. In fact even those described in table 3.1 are redundant as there are only two completely distinct types of metric: metrics that are flux-dependent and those that are not. In the first case this would be metrics that seek to preserve a particular reaction rate or cross section, like the depletion and criticality metrics. This type of metric also encompasses any specific reaction rate or creation rate for a particular reaction byproduct. The second type of metric is one that is a scalar multiple of number density, usually based on a physical property (such as half-life, energy from decay, etc.). These metrics are for problems such as total activity, decay heat, and gamma source. This metric type is also used in cases of preserving the quantity of a specific isotope or element, in which case the metric would not have a weighting factor and the quantity to be preserved would be each of the nuclides number densities. The distinction between these problem types can be seen by the comparison in table 4.1, where the two most degenerate case — total activity and decay heat — share a large percentage of contributors at all levels but the comparison between the other problem types do not. In the table independence of the metrics is shown by the shared number of nuclides that any two metrics have in the ordering of the nuclides by contribution (e.g., the depletion metric and the total activity metric only have 7 of their 100 highest contributing nuclides and 72 of their 400 highest contributing nuclides in common).

The two independent types of metrics dominate any problem type's behavior, however, specification of the metric within the type does significantly change the results. To exhaustively examine the problem-dependent behavior of the transition system and to identify both helpful and troublesome characteristics four metrics are tested in detail. The

**Table 4.1:** Shared contributing nuclides for depletion (DEPL), total activity (TACT), decay heat (DECH), and gamma source (GAMM) problem types for the given number of highest contributing nuclides

| Top *N* Contributing Nuclides | DEPL vs TACT | DEPL vs DECH | DEPL vs GAMM | TACT vs DECH | TACT vs GAMM | DECH vs GAMM |
|---|---|---|---|---|---|---|
| 100 | 7 | 2 | 17 | 77 | 8 | 5 |
| 200 | 37 | 24 | 43 | 175 | 24 | 19 |
| 300 | 56 | 43 | 77 | 276 | 91 | 89 |
| 400 | 72 | 61 | 121 | 362 | 147 | 153 |

four metrics (from table 3.1) are depletion, total activity, decay heat, and gamma source. These problem types have the distinct qualities:

- Depletion — A flux-dependent metric that has a non-zero weight during zero flux time-steps (as opposed to the absorption rate problem type that would have the same weighting factor but a zero weight during zero flux time-steps).
- Total Activity — A flux-independent metric that has a non-zero weight at all time-steps, but only has physical importance (can be measured) during zero-flux time-steps.
- Decay Heat — A flux-independent metric that has physical importance during all time-steps, as light water reactors generate approximately 7% of their energy from decay heat [6].
- Gamma Source — A flux-independent metric that is further specified by a bounded energy range — i.e., an energy integral metric.

However, there is another class of flux independent problems that aren't included in these problem types, and that is problem types that don't seek to preserve any particular transition type but instead seek to preserve physical quantities solely dependent on the nuclide concentrations and not any transitions, like mass, radiotoxicity, and relative concentrations of particular isotopes. These problem types present their own contributing nuclides, but they do not exhibit any characteristics that differ from either the flux-dependent and flux-independent problem types. The behavior is a combination of the two problem types, as

the contributions are dependent only on preserving number densities as opposed to the other metrics which must also preserve the transition of importance for the problem type to preserve the physical quantity that they are weighted by (cross section, decay constant, etc.). This combined behavior results in the metric having more contributing nuclides, and trends more difficult to identify thus these problem types are excluded from this analysis.

### 4.1.1 Depletion Problems

The depletion problem type is the total absorption cross section case of the cross section weighted problems. These problems differ from the similar reaction rate problem types in that they have a non-zero importance on cooling steps, which has significance in used fuel problems such as criticality safety (burnup credit) and active interrogation methods. The depletion weighting factor is also the one that is most commonly used for determining a nuclide's importance for reactor applications like the library reductions performed in the preliminary reduction effort.

The depletion problem type weighting factors for the sample problem result in the contributions visualized in figure 4.2. In total there are 791 non-zero contributors to the depletion metric, of which only 298 have a contribution large enough to be depicted on the chart. As can be seen in the figure, the contributions are dominated by the actinides as well as several fission products. In the case of the heavily contributing actinides (U-238, U-235, Pu-239, Pu-240, and Pu-241) these nuclides possess moderate to high absorption cross sections, but more importantly these nuclides encompass more than 75% of the total mass in the problem at any given time step. This is similarly true for the highly contributing fission products (Xe-135, Nd-143, Rh-103, and Sm-149) which have very high cross sections but also either have a large fission yield or a parent nuclide with a large fission yield.

Despite the apparent dispersion of contribution among the nearly 300 nuclides with weights high enough to be shaded in the figure, just 39 nuclides contribute greater than 99% of the total metric. Furthermore, of those 39, two contribute greater than 50% of the total metric. Those two, unsurprisingly are U-238 and U-235. This distribution of contributions would seem to support the first of my assumptions in the development of my reduction method: that there is a limited number of nuclides that are important to any given problem

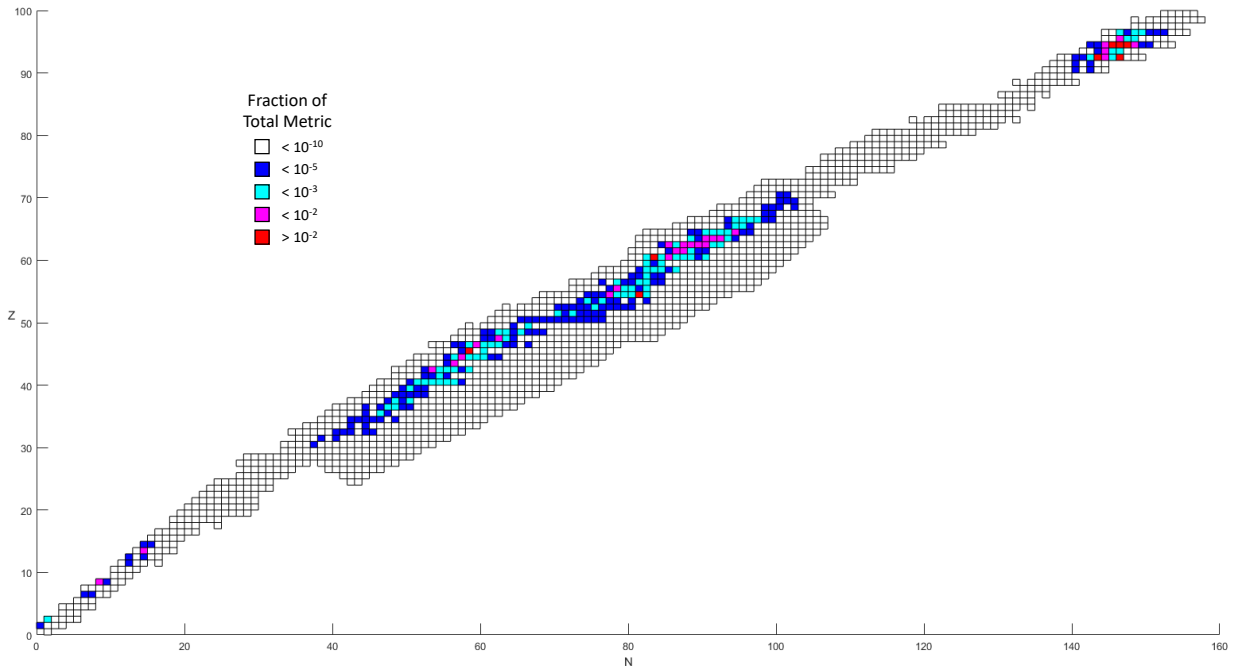



**Figure 4.2:** Nuclide chart of relative contributions in the sample problem for the depletion metric.

type. This is further emphasized in this case by the largest weighting factors clustering near, and along, the band of stable nuclides, though is also the result of the lack of cross section data for the shorter lived nuclei that are further away from this band. This clustering implies that many of the fission product precursors to these nuclides are not needed in this problem type and the acceleration method from section 3.5.4 would be appropriate for this problem type.

### 4.1.2 Total Activity Problems

The total activity problem type is the decay-weighted case of the flux-independent problem types. The metric used for this problem type is the most general case of the decay transition-preserving metrics, with all others — including those that include energy-bound constraints (such as radiation emissions) — being subtypes of this problem type. The total activity is a rate preserving metric, as are its other derivative problem types. Therefore, for this problem type the nuclides with the largest contributions are going to be those that decay the fastest; the metric will be far more sensitive to changes in the reduction problem. For

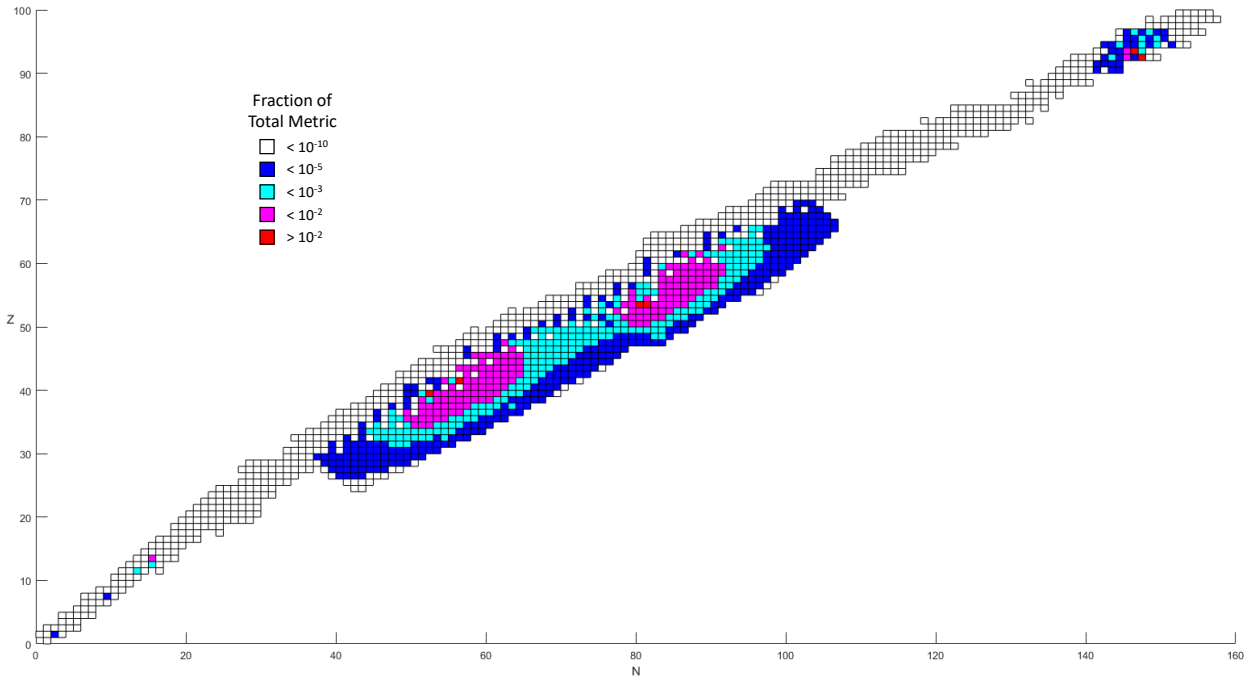



**Figure 4.3:** Nuclide chart of relative weights in the sample problem for the total activity metric.

example, the transition from an irradiation time step to a cooling time step will remove the gain pathways for the fastest-decaying nuclides (the short-lived fission products) as they will decay to negligible quantities very fast. In fact, most of the top 100 contributors have half-lives on the order of 1 hour.

The total activity weighting factors for the sample problem result in the contributions visualized in figure 4.3. As can be seen, this is a far more distributed metric than that of the depletion problem type. There are 1436 non-zero contribution nuclides, 846 of which have high enough contributions to be depicted in the figure, which is more nuclides than contributors to the depletion metric. Even more significant is that 44 nuclides contribute to 50% of the metric and 270 contribute greater than 99% of the total activity metric. Though the highest contributors are in the actinides (slightly greater than 16% of the metric is U-239 and Np-239 resulting the high abundance of U-238 and its associated absorption and progeny decays), the next 101 highest contributors are all fission products and notably ones with the highest fission yields. This dependence on the fission yield can be clearly seen in

the figure, with a skew in the weighting factors resulting from increasing instability for the nuclides further away from the region of more stable nuclides.

### 4.1.3 Decay Heat Problems

The decay heat metric is very similar to the one for the total activity problem type, with the addition of using the recoverable energy values for each decay to alter their relative importances. Despite this scaling of the weighting factor, there is very little change in the relative ordering of nuclides by their importance. The major difference between the behavior of the contributions to this metric, as compared to that for the total activity, is that within each "tier of contribution" (depicted as a different color in figure 4.4) the contributions are relatively even, as opposed to the contribution being significantly concentrated in the top nuclides of each tier. The actinides U-239 and Np-239 contribute approximately 3.8% of the total metric for the sample problem (despite still being two out of the three highest contributing nuclides) is perhaps the most significant example of this, especially considering that 46 nuclides contribute 50% of the total decay heat metric which is only 2 more than the total activity metric.

As noted, the decay heat problem type has all of the same contributors as the more general total activity problem type. In fact the similarity between the two goes beyond that, in that at any level of comparison the two metrics share a significant number of the same contributing nuclides. Out of the top 100 contributing nuclides for the total activity, 77 are also in the top 100 for the decay heat metric and as you increase the number of nuclides that are compared the greater the proportion that are shared (see table 4.1). At the typical sizes of the reduced libraries, this shared proportion increases to 479 out of 500 and 583 out of 600 of the highest-contributing nuclides. This is due to the degeneracy of this problem type to the total activity problem type making it redundant in most cases as the more general metric would result in a more robust reduced library. This degeneracy also results in libraries being reduced for one of the problem types being functional, and in general sufficiently accurate, for the other problems. This is also true for the other possible metrics, that are not covered in this dissertation, that are degenerate cases. The specialized

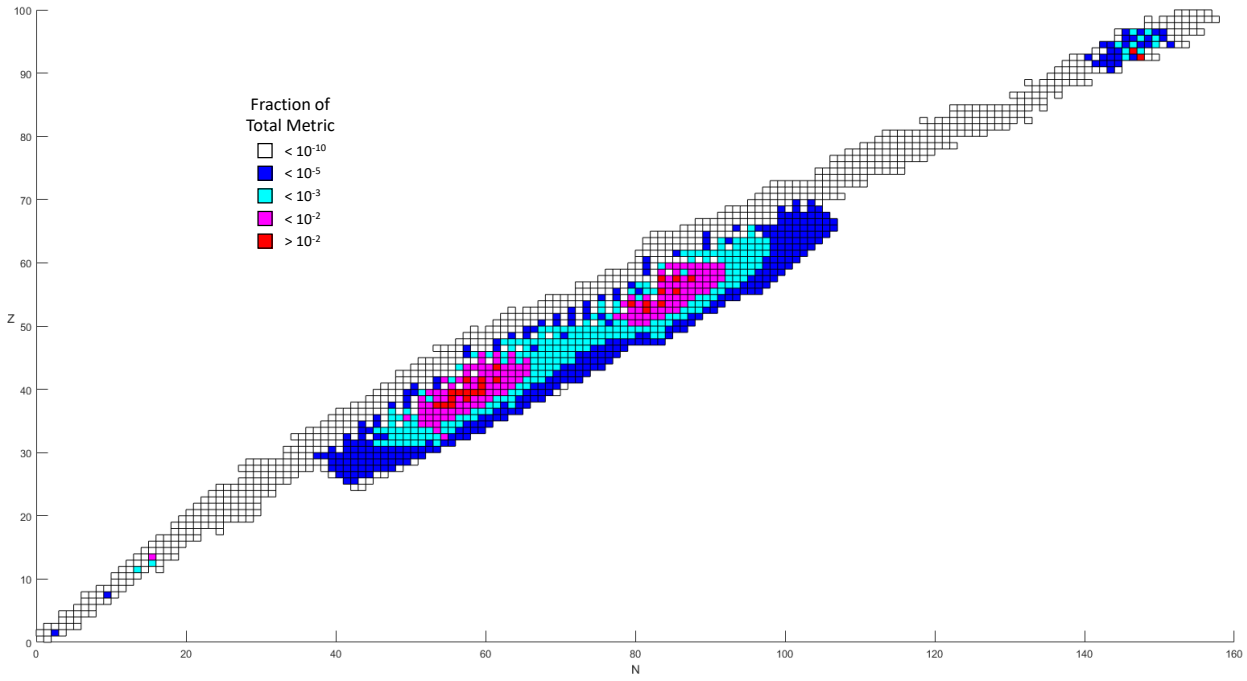



**Figure 4.4:** Nuclide chart of relative weights in the sample problem for the decay heat metric.

metrics are only needed when they are being used for highly-optimized cases with very low tolerances for error in the reduced library.

### 4.1.4 Energy-Dependent Radiation Source Problems

The energy-dependent radiation source problem type is a further specification of the total activity problem type. This metric further specifies a proportionality factor, the yield per decay of the particular radiation type, as well as energy bounds over which to consider the radiation type. These energy bounds allow the user to specify the limits in which the radiation is significant, such as for dose or radiation damage calculations where a threshold model is being used, or perhaps more commonly to preserve the accuracy of energy emissions for a window of energies for detection applications. In any end us case, the specification of energy bounds results in a significant departure from the degenerate behavior of the decay heat problem type since there is such a significant down-selection of important nuclides by both limiting the decay byproduct and the byproduct energy. This down-selection is evident in figure 4.5 by the few nuclides that fall into each of the "tiers of contribution". In the case depicted, specifying gamma rays emitted between 0.5 MeV and 1.5 MeV for the sample problem, approximately only one third of the nuclides that contribute to the total activity metric have non-zero contributions, with only 462 nuclides contributing. These energy bounds were chosen for this sample case as to include all of the common gamma spectroscopy peaks in a passive gamma measurement that are used in the nondestructive assay of used nuclear fuel. These peaks are those produced by Cs-137 (661.7 KeV), Cs-134 (604.72 and 795.86 KeV), Eu-154 (723.3, 873.2, 996.3, 1004.8 and 1274.4 KeV), and Rh-106 (621.9 KeV).

This is by far the most limited set of contributing nuclides of the problem types considered and many of the gamma emissions contributing are the indirect product of a decay or reaction; of the 462 contributing nuclides only 51 nuclides have a sufficiently large contribution to be depicted in the figure. Furthermore the largest contributer (Cs-134 resulting from the neutron capture of Cs-133) represents 58% of the total metric. This is a result of three factors: Cs-133 being the most common fission product, with a cumulative yield of 6.89% and having a moderately large neutron capture cross section of approximately 29 barns, and Cs-134 producing on average 2.2 gammas per decay within the range of energies that the metric is considering for this problem (though Cs-134 has 8 independent gamma

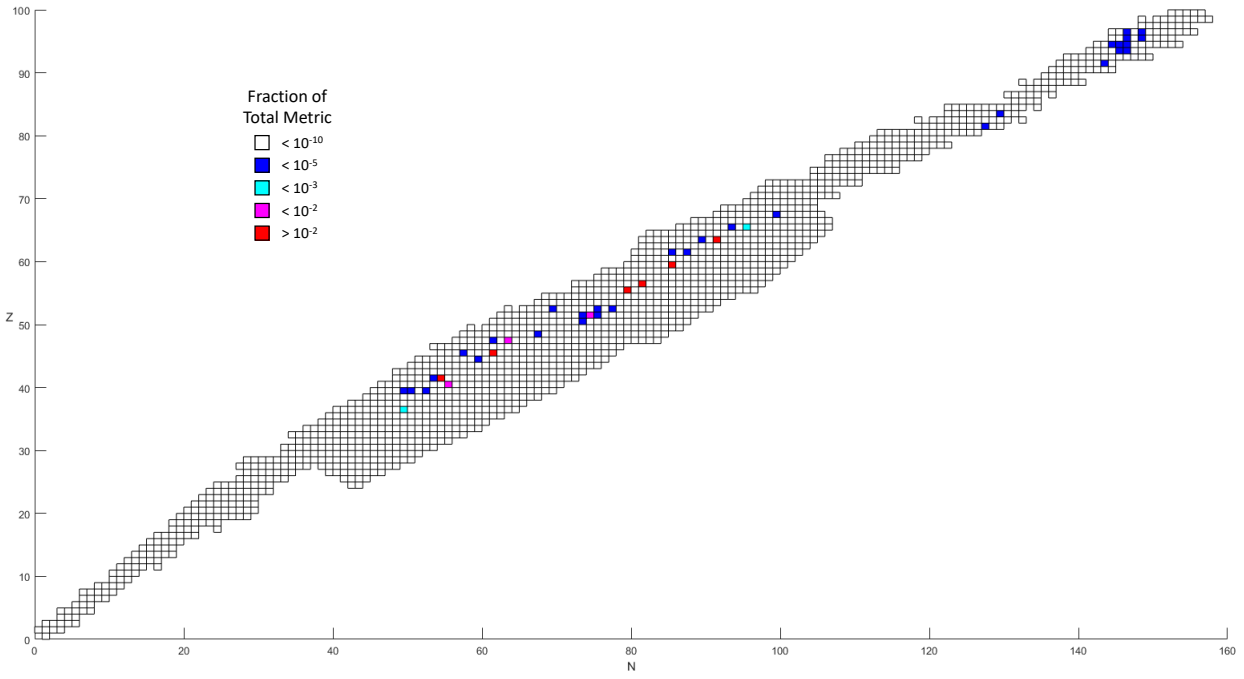



**Figure 4.5:** Nuclide chart of relative weights in the sample problem for the gamma source metric, with energy bound of 0.5MeV and 1.5MeV.

lines with greater than a 1% intensity that falls within this energy bounds). The next largest contributor is Ba-137m, at approximately 21% of the total metric. This nuclide is the direct product of Cs-137 beta decay, and the gamma that it produces is commonly attributed to this parent nuclide as Cs-137 has a have life of 30.17 years and Ba-137m has a half life of 2.55 minutes they are in secular equilibrium at most time scales of interest. Including these two most significantly contributing nuclides, 99% of the total metric is contributed by only 7 nuclides. This concentration of the metric into a small number of nuclides is further support for my first assumption but also illustrates the need for a method to determine importance beyond a direct comparison of contributions or weighting factors.

## 4.2 Weighting Methods

The methods described in chapter 3 for determining the importance of a nuclide to the problem specific metric have similar behavior for all problem types. However, due to the different behavior of the metrics in relation to both the sample problem as well as subsection

of the transition system that the metric is concentrated in, the appropriate weighting type will vary for each problem type. This decision generally comes down to choosing between three of the methods: contribution weighting of degree one or two, and the corrected adjoint weighting. The cases observed where these are not the optimal weighting methods will be discussed in chapter 6.

### 4.2.1 Contribution Weighting

As described in section 3.4.1, the contribution weighting method is theoretically valid at all degrees but not practically applicable above a degree of 2. As such the analysis of this method, both in this chapter, and subsequent chapters will be limited to contribution weighting of degree 0, 0.5, 1, and 2. Contribution weighting of degree 3 is illustrated for the depletion metric in figure 4.6. As can be seen when compared to figure 4.7, degrees higher than 2 result in weight diffusion causing a washed-out or blurred importance resolution. In this particular case it causes the dominant contributions from U-235 and U-238 to spread significantly to their direct fission product chains and actinide neutron capture products, over emphasizing their contributions. This loss of resolution creates difficulty in estimating convergence within the system determination methods, as well determination of the source of the direct contribution to the metric. This behavior is the reason that the analysis of this weighting method has been restricted to these integer degrees and the weighting degree of 0.5.

In this case the integer degrees of weighting are exactly as described by the equations in section 3.4.1, whereas the weighting of degree 0.5 is a nomenclature chosen to fit the trend of the integer values. The degree 0.5 weighting indicates weighting following the scheme of the integer values but only to the transition product. More explicitly, contribution of degree 0.5 is one where a nuclide's weight is calculated as its direct contribution to the metric plus the sum of the parent nuclides' weights scaled by the fractional yield to the nuclide of interest.

In theory, this method seeks to preserve an additional physical quantity at each higher degree of weighting. At a weighting degree of zero, the nuclide concentration is the main quantity that is preserved; at degree of 0.5, the contributing transition is preserved by giving importance to just its parent and product nuclides; at a degree of 1, the transition

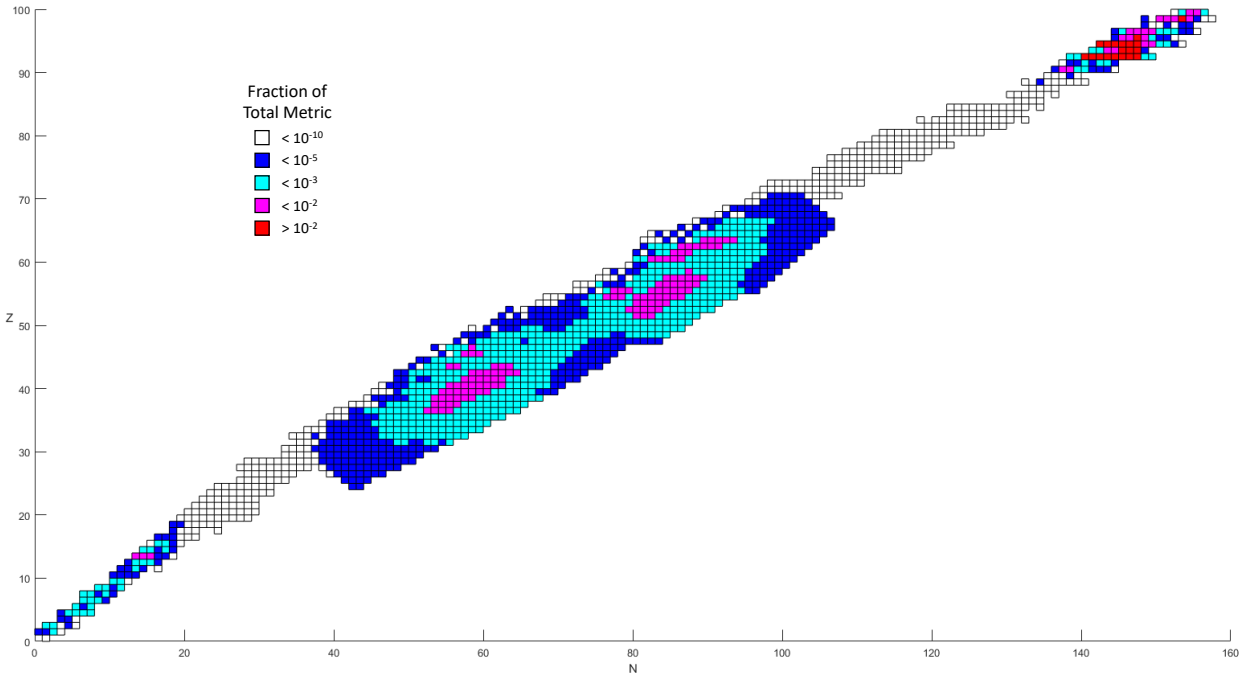
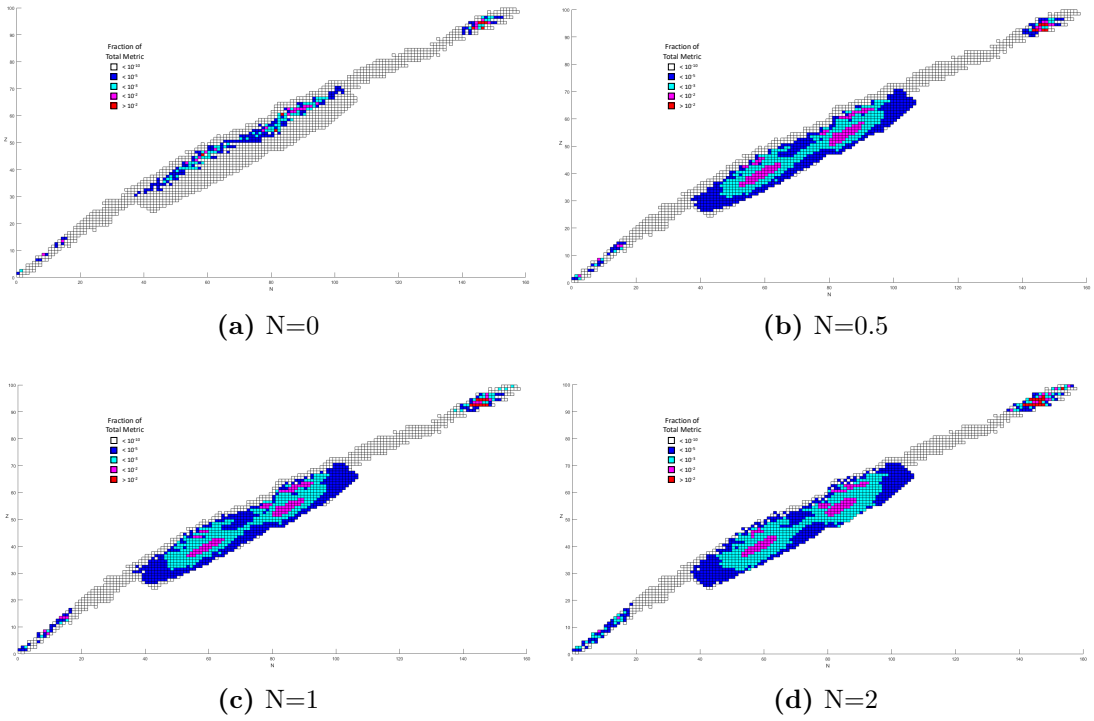



**Figure 4.6:** Nuclide chart of depletion metric contribution weights of degree 3 for the sample problem.

rate is preserved by giving importance to the all of the contributing nuclides creation and loss channels, effectively increasing the accuracy of the nuclide concentration. At higher degrees, it remains the nuclide concentration term of the weighting factor that is increasing in accuracy by preserving parents concentrations further up the burnup chain with each increase in the order of the weighting degree. This diffusion of the nuclide weight can be clearly seen in figures 4.7, 4.8, 4.9, and 4.10 as the degree of weighting is increased, though it is most evident with the depletion and gamma source metrics as the total activity and decay heat metrics have heavily dispersed direct contributions.

The most significant change in the weighting distribution, using the contribution weighting method, is from increasing the weighting from direct contribution to a weighting degree of 0.5; this increase in weighting degree also has the most significant improvement in relative accuracy as a measure of importance for the system determination methods but a majority of that is due to the physicality of system representation needing both the parent and product of a transition. Though the change is most significant in the depletion problem, as can be seen with the weighting of the direct fission products that have no direct



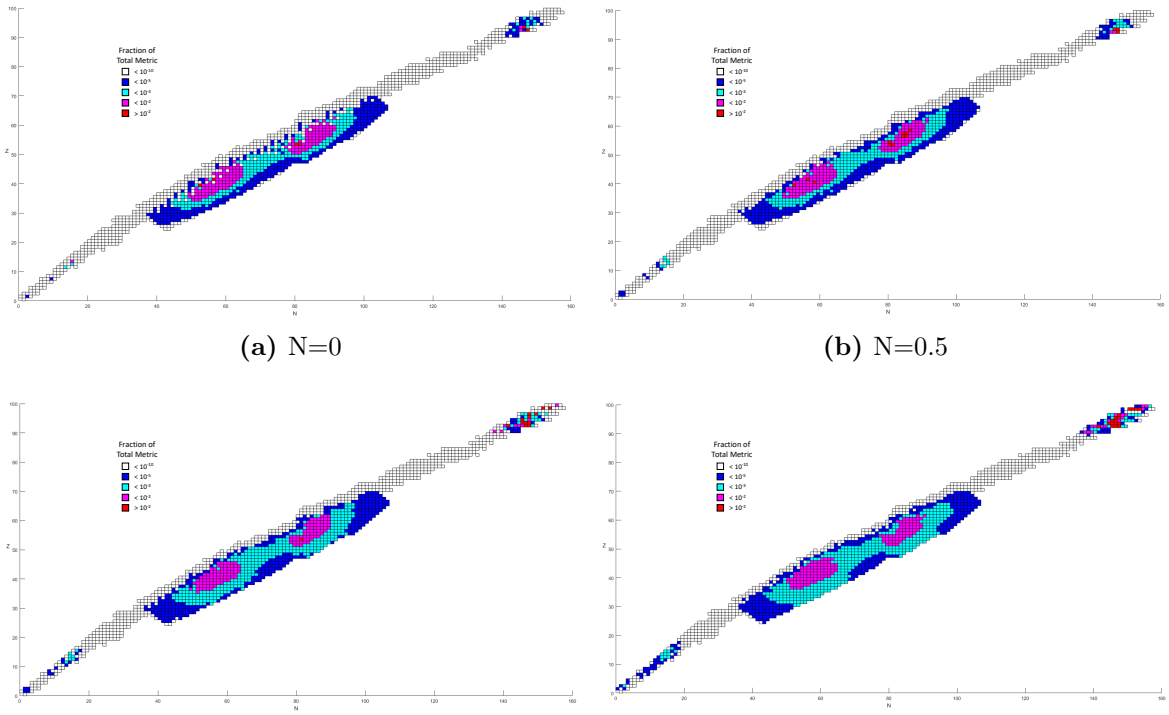

**(a)** N=0 **(b)** N=0.5

**(c)** N=1 **(d)** N=2

**Figure 4.7:** Contribution weights for the sample problem depletion metric for different weighting degrees, including the nuclide only (N=0), the nuclide and products (N=0.5), the nuclide and all directly contributing species (N=1), and further up the burnup chain (N=2)

contribution (due to having no cross section data), the gamma source metric also has a visible redistribution of the relative weights. Though this redistribution is not entirely visible as most of the highest-contributing nuclides are metastable states that transition to their ground state and thus are depicted in the same location on the charts. The redistribution in the activity-dependent metrics differ from those that are flux dependent in that the shift in weights is towards the stable nuclides rather than to the highly unstable fission products and the nuclides that are the products of neutron capture. This is to be expected given the nature of underlying transition types that contribute to each metric, but this does mean that despite using the same algorithm conflicting results are achieved. This difference will be further evident in the optimization process and results in chapter 6.

In most code frameworks that would benefit from improved source term accuracy, the nuclide-specific concentrations are not of interest. Rather, what is of interest is either the reaction rate or the generation rate of the source, thus the aim in the reduction process of the library should also be to preserve these rates. It also important to preserve reaction




**(a)** N=0 **(b)** N=0.5

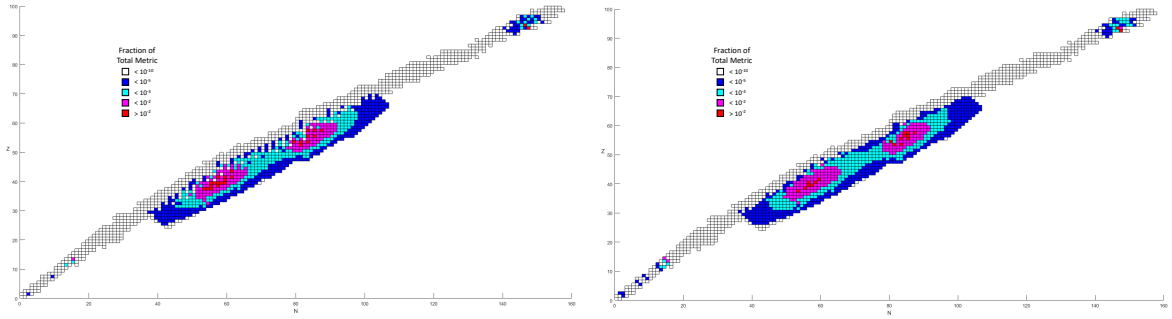
**(c)** N=1 **(d)** N=2

**Figure 4.8:** Contribution weights for the sample problem total activity metric for different weighting degrees, including the nuclide only (N=0), the nuclide and products (N=0.5), the nuclide and all directly contributing species (N=1), and further up the burnup chain (N=2)

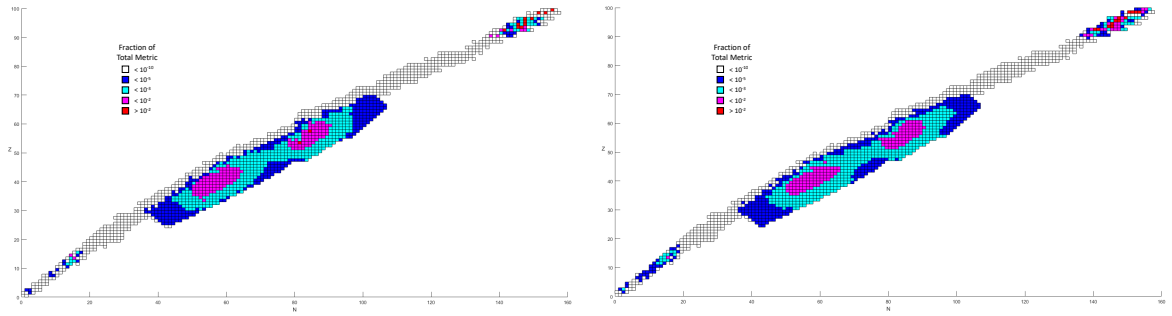
rates regardless of problem types since the data used in the depletion code are one-group cross sections that are flux weighted to perform the group collapse; this weighting is used to preserve reaction rates. The degree one weighting, which should preserve the transition rates of interest directly, also preserves significant transitions that are precursors to those of interest. This behavior results in the weighting of both reactions and decays that are important to the system regardless of whether the metric is flux dependent. For all metrics, this does result in a shift of the relative contributions to the actinides. This shift is the result of the actinides being fissionable and thus considered to be a parent to the contributing fission product nuclide, regardless of how low the fission yield or parent fission cross-section is, or in the case of higher actinides, the parent fissioning species' abundance. The most striking example of this is in the gamma source metric (figure 4.10c) where due to Cs-134 having a significantly larger yield from Cf-249 than any other fissionable nuclide, Cs-134 representing a majority of the direct contribution, and Cf-249 contributing to the other

fission product contributor nuclides, Cf-249 is the fourth-highest weighted nuclide for degree one weighting. This behavior can be both considered an advantage and a shortcoming of degree one weighting, depending on the problem type, since it will emphasize the rates of the highest contributing transition which can be of benefit in diffuse metrics like total activity and decay heat but detrimental in metrics that are already dominated by a few nuclides as it will quickly obscure the lesser nuclides' contributions. This behavior, though, also has the benefit that decay and reaction byproducts of interest are not entirely attributed to the short-lived direct parent. In the case of the gamma source metric, this results in the commonly known gamma producers (such as Cs-137) being weighted nearly equally to their metastable parent contributors. Thus, despite Cs-137 decaying by beta decay for example, it is given the importance associated with being the primary producer of the 661.7 KeV. This is also important for other metrics that have compound contributors, such as the delayed neutron emitters being preceded by beta decays from the direct fission products or any short-lived metric contributor that is in secular equilibrium with a parent. This effect, though not visible in the figures, plays a significant role in accurately attributing importance for both the total activity and decay heat problems.

As weight is distributed by higher degrees, it results in certain nuclides concentrating weight. These weight-aggregating nuclides are given this higher importance, commonly having a weight greater than the direct contributors that they are the predecessors of, due them being common parents or intersecting members of multiple burnup chains of highly contributing nuclides. The most common nuclides to experience this aggregating effect is the fissionable nuclides, which is evident in the figures — even at contribution weighting of degree two. While this behavior exhibits benefits at degree one, at higher degrees it results in the actinides — even the rarer actinides so long as they are fissionable — having higher weights than directly contributing nuclides despite them resulting in few fission reactions that could create those daughter products. This issue is further compounded by the weights being distributed to all fission products in order to preserve the transition rates of these nuclides that are deemed "important". This issue worsens as the weighting degree increases as further fissionable nuclides begin accumulating weight, ultimately resulting in a very similar distribution of weights independent of the problem and metric. In the sample problem,



**(a)** N=0 **(b)** N=0.5

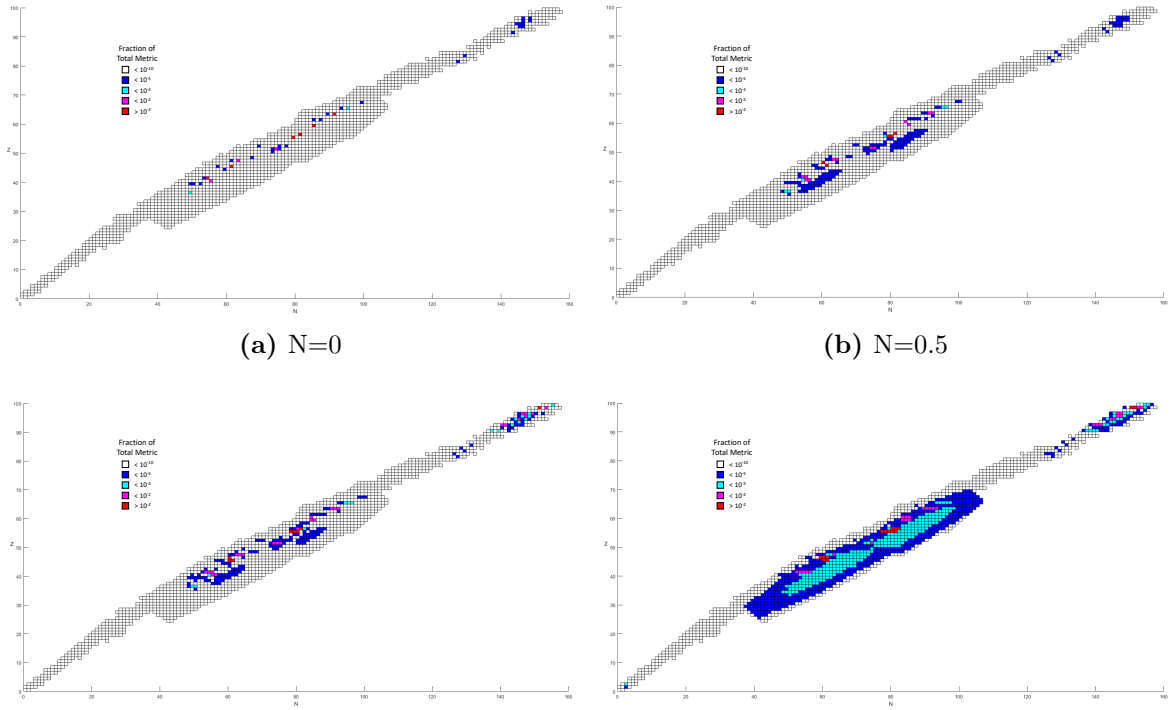


**(c)** N=1 **(d)** N=2

**Figure 4.9:** Nuclide contribution weights for the decay heat metric for the sample problem for different weighting degrees, including the nuclide only (N=0), the nuclide and products (N=0.5), the nuclide and all directly contributing species (N=1), and further up the burnup chain (N=2)

which is fission-dominated, this issue becomes visible in figure at degrees of weighting as low as two for even the gamma source metric 4.10d — despite it being the most resistant to weight diffusion of the four metrics being analyzed — where the weighting of nuclides follows the fission yield curve for a majority of the nuclides and a significant number of the highest weighted nuclides are minor actinides despite neither region directly contributing to the metric.

Higher-degree weightings can be useful though, in problems that inherently concentrate weights along a longer burnup chain. One example of this would be any case concerned with the concentration of a particular nuclides and its products after a long period of decay, though in principle this would apply to capturing the importance of the creation path of any nuclide that has a small yield from whatever the starting material may be. However, in these cases it would be preferable to use alternate methods. These methods can be either relying




**(a)** N=0 **(b)** N=0.5

**(c)** N=1 **(d)** N=2

**Figure 4.10:** Nuclide contribution weights for the gamma metric for the sample problem with energy bounds of 0.5 and 1.5 MeV for different weighting degrees, including the nuclide only (N=0), the nuclide and products (N=0.5), the nuclide and all directly contributing species (N=1), and further up the burnup chain (N=2)

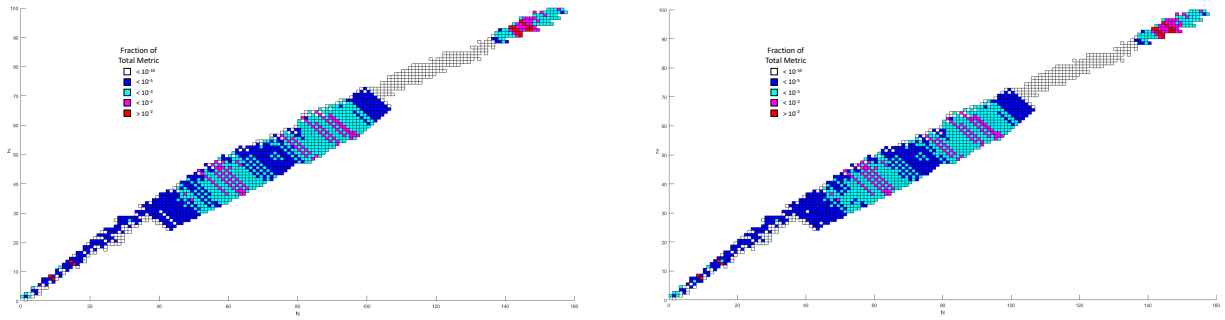
on an appropriate system determination method to properly traverse the system without the continuous weighting information, or to use an alternative weighting method such as an adjoint weighting method.

### 4.2.2 Adjoint Weighting

The adjoint weighting method, as detailed in chapter 3, is a method of distributing the metric contribution through the calculation of each nuclide's importance in the creation of the nuclides that directly contribute to the metric via the adjoint depletion-weighted contributon concentrations. Determining importance in this way concentrates weights along the burnup chains that are most commonly followed. These burnups chains are dominated by certain transition types in each region of the transition system in any fission dominated system, including the sample problem considered. These transition types are neutron capture

reactions in the activation/light nuclides and actinides, and beta decay in the fission product region. The neutron capture dominance is not surprising in the activation/light nuclide and actinide regions due to that being the only path to the heavier isotopes in both regions given their limited source material inventory and the irradiation only considering neutron flux, and not fluxes derived from non-neutron decay and reaction byproducts. However, the strong dominance of the beta decay pathways for the entirety of the fission product region was not initially expected due to there being many common transition types. For the short-lived direct fission products, beta decay is the only loss transition that there is data for — on the nuclide charts these nuclides make up the neutron (and proton) drip lines — but the other fission products have many potential paths over which the importance should be split. This dominance of the beta pathways is caused not because it is the most dominant transition for the contributing nuclides, but rather it is a commonly shared path that other reactions and decay branch off of. This characteristic of being a common path among all nuclides results in significant weight aggregation from all contributing nuclides. This weight aggregation does not exhibit the shortcomings that it does in the contribution weighting method due to both the adjoint weighting and corrected adjoint weighting methods being unidirectional algorithms. This means, that though the weight is aggregating to common important pathways it is not being redistributed amongst the alternative, non-contributing paths that branch from the path. The adjoint weighting method also does not account for the importance of the product of a contributing transition to preserving that transition. However, this shortcoming is accounted for in the corrected adjoint weighting method described in section 3.4.3.

The adjoint weighting method acts to preferentially weight pathways that contribute to the creation of the highest-weighted nuclides, whereas the contribution weighting method at the optimal degrees results in the weight being distributed over a contributing nuclides neighborhood to create local regions of importance for all contributing nuclides. As seen in figures 4.11 and 4.14, this results in a dispersion of weight across the transition system for the depletion and gamma source metrics, much like increasing the weighting degree for contribution weighting — though with different resulting distributions. However, for the total activity and decay heat metrics the weighting becomes more concentrated about the

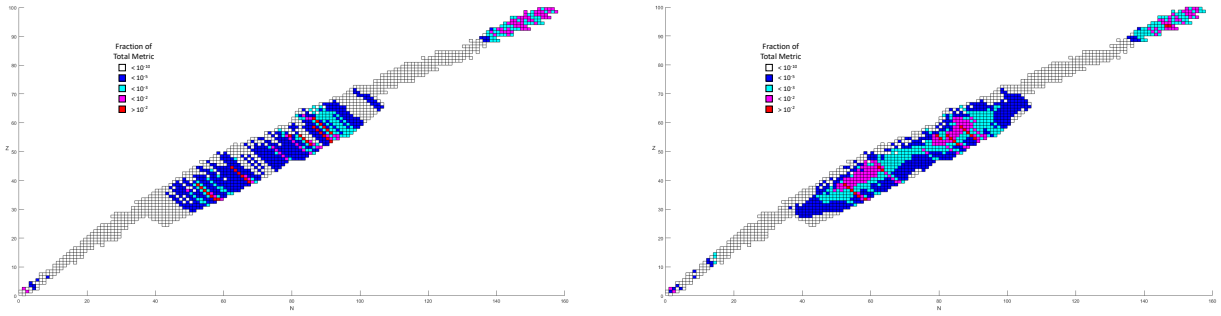



**(a)** Adjoint Weighting **(b)** Product-Corrected Adjoint Weighting

**Figure 4.11:** Nuclide adjoint weights for the depletion metric for the sample problem for the product-corrected and uncorrected adjoint weighting.

highest contributors. This is due to these metrics, and other metrics with many contributors, lacking the highly defined "tiering" of weights that is inherent to the other metrics with a few dominate contributors. Thus instead of spreading the weight of a highly contributing nuclide to its preceding nuclides, the weight is concentrated from the many sources to the common burnup chains, and perhaps even more noticeably, the common parents — like the fissionable nuclides. This behavior was also seen in the contribution weighting method, and further emphasizes the need to preserve fissionable nuclides for any depletion calculations that involve irradiation steps, regardless of the metric, due to their significance of being highly connected nodes and the accuracy of their concentrations directly affecting the accuracy of the entire fission product region. The clear pathways from these fissionable materials, and the source materials, to the contributing nuclides greatly increase the effectiveness of the simpler system determination methods, making convergence to a reduced system much faster, but not enough to offset the long computational time to calculate the adjoint weights compared to even high degree contribution weights.

The adjoint weighting's clear pathways has a shortcoming in common with the direct contribution weights, and that is that it does not consider the importance of preserving the transition product in the transition system physical representation. The adjoint method is formulated to weight the most important creation pathways as to best preserve the nuclide concentration of the contributor. While this is useful for standard perturbation analysis that seeks to determine the pathways that contribute to uncertainty and error in a nuclide or the

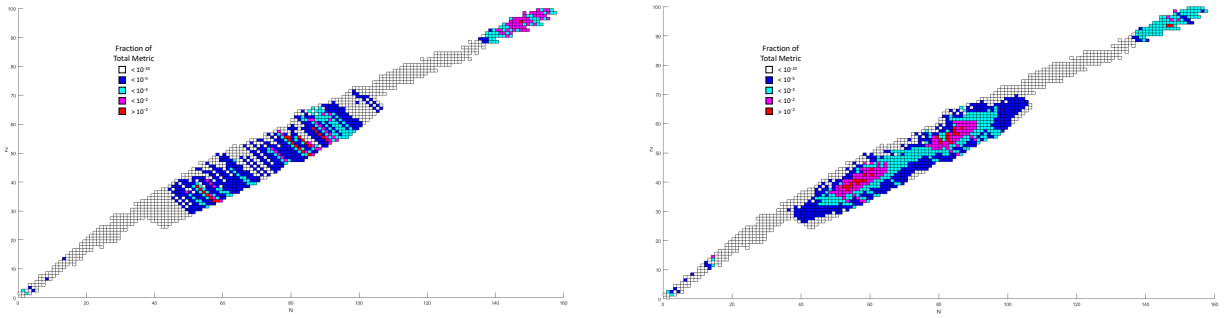



**(a)** Adjoint Weighting **(b)** Product-Corrected Adjoint Weighting

**Figure 4.12:** Nuclide adjoint weights for the total activity metric for the sample problem for the product-corrected and uncorrected adjoint weighting.

impact of uncertainty in a cross section, it is insufficient for this physical system approach to library reduction. This insufficiency could be corrected with a number of methods in either the system determination method or a post processing method, like the competition correction method, if it was implemented after every reduction iteration. However, the insufficiency is a result of the weighting method not capturing the importance of the product nuclides, and thus in theory that is the most appropriate place to address the issue. In practice, though, any correction performed to the adjoint method to correct this would not be coherent with the underlying adjoint weighting method theory. The most practical of these ad-hoc weighting correction is the implementation of a hybrid method where the adjoint weighting distribution is corrected by the contribution weights of degree 0.5 that are calculated for the product nuclides. This weighting will be referred to as the corrected adjoint weighting for the remainder of this dissertation.

The corrected adjoint weighting does not result in a significant departure from the adjoint weighting for the sample problem in the depletion and gamma source metrics. Similar to the weight dispersion effect already discussed, this is due to the dominance of the contribution of relatively few nuclides. The most significant changes in these weighting are the increase in weight of Pu-242, Pu-243, and Sm-150 for the depletion metric and in the increase in weight of Pd-106 and Cs-135 for the gamma source metric. The nuclides in the case of the depletion metric are all products of neutron capture and are required to maintain 3 of the top 10 most frequent, and thus most important, neutron capture pathways. Pd-106 plays a similar role in

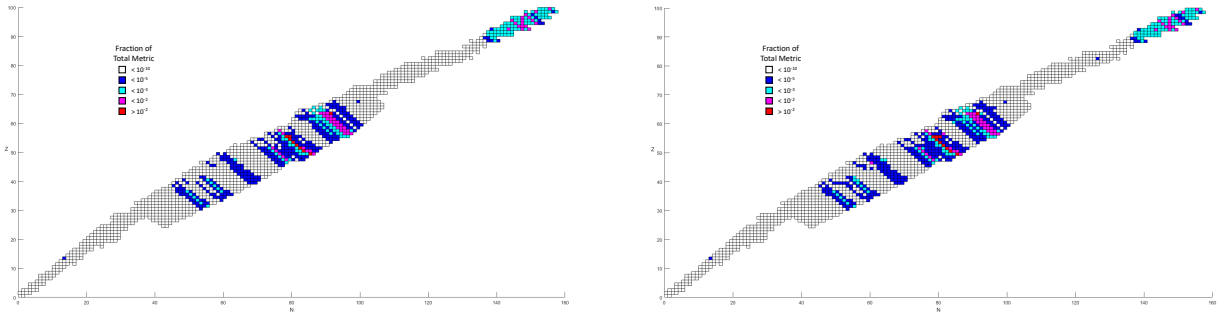



**(a)** Adjoint Weighting **(b)** Product-Corrected Adjoint Weighting

**Figure 4.13:** Nuclide adjoint weights for the decay heat metric for the sample problem for the product-corrected and uncorrected adjoint weighting.

the gamma source metric being a product from the decay of Rh-106, however, Cs-135 plays a more significant but less direct role in maintaining the accuracy of the library. The Cs-135 weight is derived from it being the neutron capture product of Cs-134. This reaction has a cross section of 129 barns, making it the largest competing pathway to the decay pathway that results in the contributing gammas, thus resulting in Cs-135 being one of the highest weighted nuclides by the adjoint weighting method.

For the total activity and decay heat metrics, visualized in figures 4.12 and 4.13, the consolidation of the weights is countered by the introduction of the correction. The correction factors applied for these two metrics are larger than the weights of most of the nuclides from the adjoint weighting and the result is that corrected adjoint weighting is very similar to the contribution weighting of degree 0.5. This ability to capture contradictory aspects of the problem types based on the physics rather than any inputs or decisions by the users is the biggest advantage of this weighting method, and is not shared by the alternate methods for accounting for the deficiency in the adjoint method. Also, as a result of this capability the corrected adjoint weighting has distinguished itself as one of the two optimal choices for library reduction problems alongside contribution weighting of degree 1, as mentioned previously.




**(a)** Adjoint Weighting **(b)** Product-Corrected Adjoint Weighting

**Figure 4.14:** Nuclide adjoint weights for the gamma source metric for the sample problem for the product-corrected and uncorrected adjoint weighting.

## 4.3 Problem Definition

The previous section explored the interactions between the problem metrics and the different weighting methods that have been developed. These interactions are dependent of the nuclear data from the library and transition structure, however, there is another variable, that can possibly alter the behavior of the system, that needs to be explored. This variable is the definition of the sample problem that is used to determine weights and importances. Alterations to these variables do not change the transition structure but they can affect the nuclear data used due to it being one group data and the change in any of these variables changing the flux used in the multigroup cross section collapse. Ideally, requiring a high accuracy sample problem would not be needed and the reduced libraries will cover a similar application space for the problem type to the original libraries, and therefore a library reduction would not be necessary for every specific case and a single reduction can be as an upfront computational cost for all following simulations, rather than a systematic one, as that would greatly reduce the applicability of these methods to their stated goal. It is already known, and key to the initial assumption that the limited selection of nuclides is **problem type** specific, that the sets of nuclides that contribute to each metric type are greatly different — as emphasized in table 4.1. However, any changes to the problem definition that are within the original conditions of the library generation, would need to result in minimal changes in the weighting and metric distribution for the assumption to hold true.

### 4.3.1 Burnup Dependence

Burnup is a measure of power extracted from nuclear fuel and is commonly measured in megawatt-days per metric tonne initial uranium (MWd/MTU) or initial heavy metal (MWd/MTHM)to be inclusive of other nuclear fuel types. Burnup is also used as reference to the amount of fission events that have taken place and thus both the quantity of fission products generated, and also the intensity of radiation damage that the fuel assembly components have undergone. High burnup fuels are currently an area of study in nuclear materials to develop fuels, claddings, and structural components that can better withstand the intense or prolonged irradiation necessary. Accurately simulating the effect of nuclear fuel burnup is also the original purpose for the development of Origen and other depletion codes. As the burnup increases, the change in the fission events from being entirely from uranium, to also being from other actinides as they are bred in — most notably plutonium isotopes, and most significantly Pu-239 — results in a shift in the neutron spectrum as neutrons born from these fission event have on average higher energies. The other effect of the increased burnup — the fission products increasing in concentration — increases the number of parasitic neutron captures, particularly at low energies where their cross sections are the highest, though some fission products do reach an equilibrium concentration — like Xe-135. These two effects both serve to result in significant changes to the neutron spectrum and thus the one group cross sections have to be calculated at multiple burnups, using the appropriate flux weighting.

In the Origen standard libraries being used to test my library reduction methods, this weighting spectrum is found by the TRITON lattice physics code as detailed in section 1.3. However, a transition matrix for every possible burnup point cannot be stored on a library. To account for this, burnup points spanning the operating ranges for each reactor design are stored in the standard libraries and the one group values at those points are interpolated between to generate approximate transition matrices at the intermediate burnups. The number of burnup points on an Origen library varies but they are generally between one and two dozen. The pressurized water reactor library used in the sample problem has 17 burnup points ranging from 0 to 70,500 MWd/MTU. Light water reactor fuel in the US reaches a

final burnup between 40,000 and 50,000 MWd/MTU[8], so many of these burnup points are above what is reached for the bulk of commercial reactor fuel but these burnups are possible in localized areas in the reactors that experience a higher than the assembly-average power. To meet the standards necessary for this method this range of burnups need to have the same order of nuclides contributors, or at least each tier of contributors needs to be have a very similar inventory.

As a comparison over the range of typical operating burnups reached, as well as those transition matrices that are used in the depletion calculations of the unaltered sample problem, contributions to the four metrics that have been analyzed previously were calculated for final burnups ranging from 500 MWd/MTU to 40 GWd/MTU. The final burnups for these calculations are 0.5, 1, 5, 10, 20, 30, and 40 GWd/MTU with these burnups being reached by changing the power during the irradiation steps of the sample problem but not the length of time for the irradiation. Unsurprisingly, the conditions under which the nuclide orders of contribution are most different is between the problems with final burnups of 500 MWd/MTU and 40,000 MWd/MTU. The differences across the four problem types, plotted in figure 4.15 for weighting tiers of size 100–600, are most significant for the depletion metric for the most highly contributing nuclides. All four metrics have an average difference in inventory of 6% to 10% across all list sizes of the highest contributing nuclides. These differences in inventory are mostly constrained to nuclides that are near the cut-off at a given list size and are caused by minor re-orderings, but a non-negligible quantity are contributing significantly less. This effect becomes vanishingly small though as the final burnups being compared approach one another, as expected. This reduction is significant enough that at 20,000 MWd/MTU, the inventories are on average only 2% different from those at 40,000 MWd/MTU while still following similar trends to those of the comparison in figure 4.15 but with a peak of only 9% difference for the depletion metric at a list size of 100 nuclides. These inventories are sufficiently similar enough to where a reduction with a higher burnup sample problem can be expected to accurately model the transition system at lower burnups, provided that the burnup at each time step is within 50% of that of the actual problems. This restriction is due to the weights being time integrated quantities, thus placing importance on the preservation of a nuclide relative to the length of time that it is a significant contributor for.

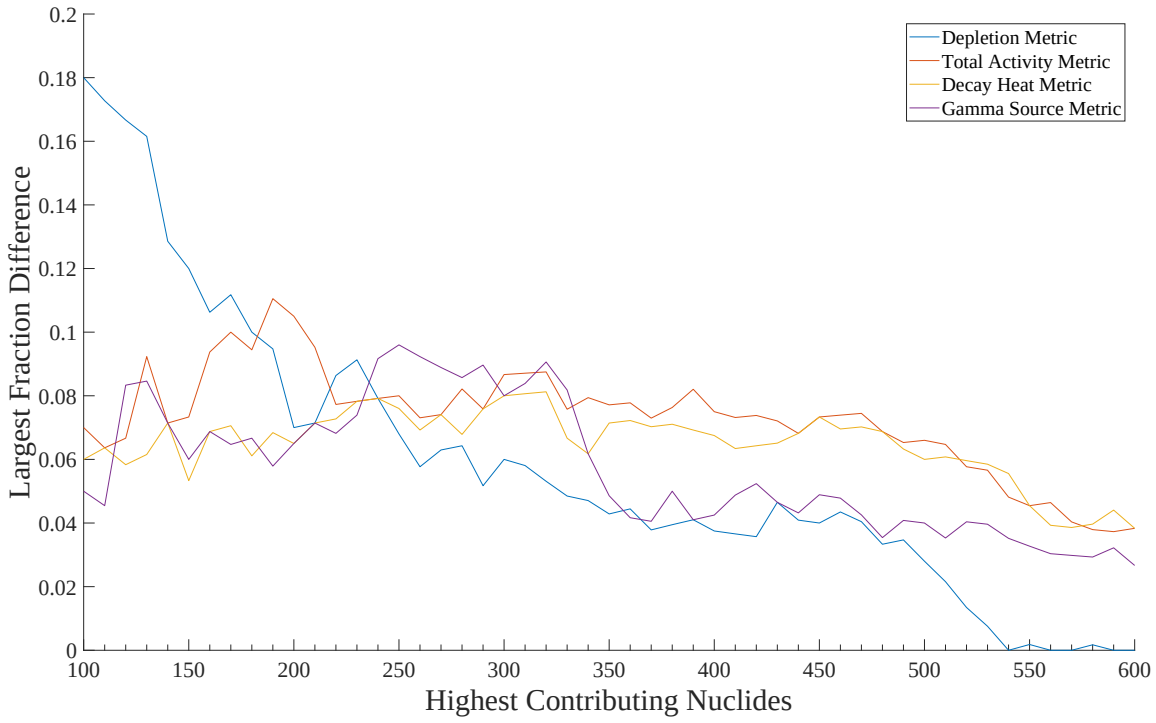
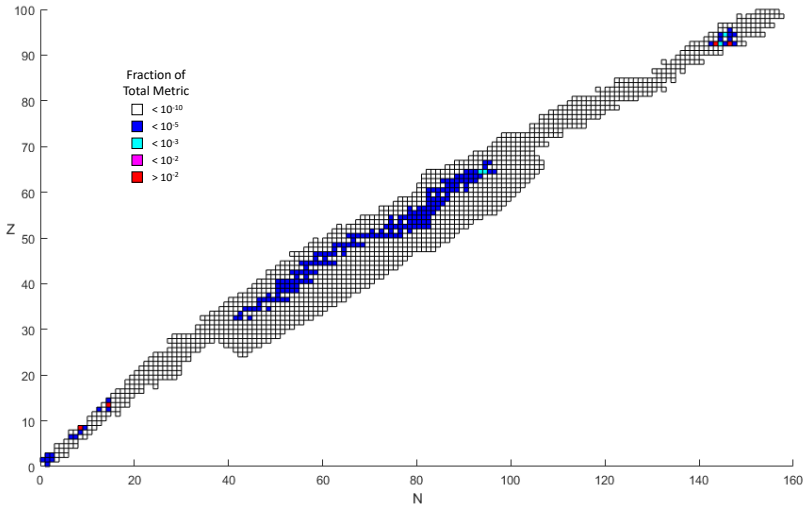
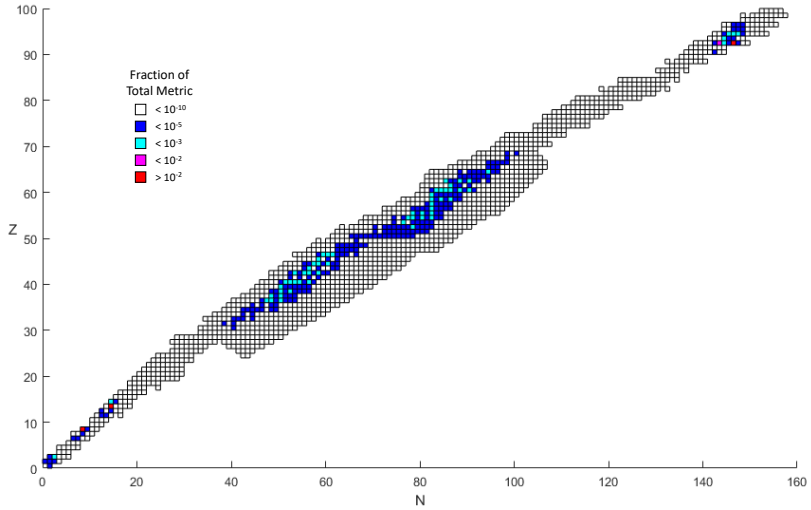



**Figure 4.15:** Graph of the fewest nuclides shared between final burnups for a given metric type relative to the number of most highly contributing nuclides compared.

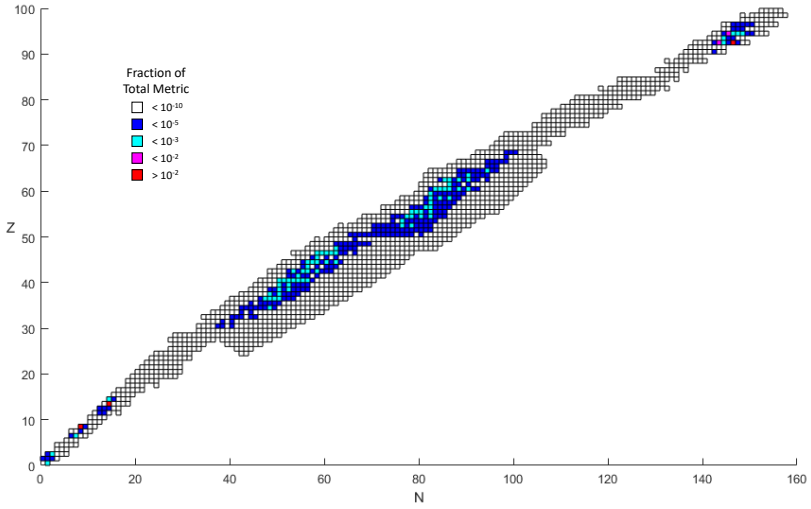
The differences in inventory as burnup increases is due to the significantly lower reaction rates at lower burnups. This results in decay transitions becoming more frequently used as the loss competition is less from the reaction pathways, and nuclides that are important at higher burnups cannot be bred into the system at sufficient quantities because their primary creation pathways are neutron captures. In the case of the actinides this is amplified as they rely on multiple subsequent captures for their formation. This effect can be seen in the regions where the mass in concentrated in figure 4.16. At lower burnups, and particularly in figure 4.16a, the mass is still mostly concentrated in the source materials, but it is also cumulated in the stable region of the fission products. As the final burnup increases, the mass extends into heavier minor actinides and into a greater portion of the fission product region that is not just the decay products of high yield fission products. There is also some minor increase in the concentrations of other light nuclides as neutron activation becomes more prevalent at higher burnups and so too do the activation products. These differences resulting from fewer reactions are magnified due to this being a comparison of final burnups, and not a comparison of the distributions during intermediate burnups. In the latter case the reaction rates would not be affected, just the time over which the reaction products have had to accumulate would be lessened. In this case there would not be an increase in the relative frequency of decays and the reaction parent nuclides contributions to the metric are more similar to those at higher burnups. The difference in contribution ordering is thus less pronounced for changes of burnup during a problem than it is between problems. The restriction to the applicability range of the reduced libraries is a factor that needs to be considered when formulating the sample problem, but it does not represent a sufficient reduction in robustness to the final libraries to invalidate the initial assumption. This restriction can also be partially mitigated by overestimating the final burnups so that problems that the reduced library are applied to fall within the range of burnups that were calculated as of importance in the reducing problem.



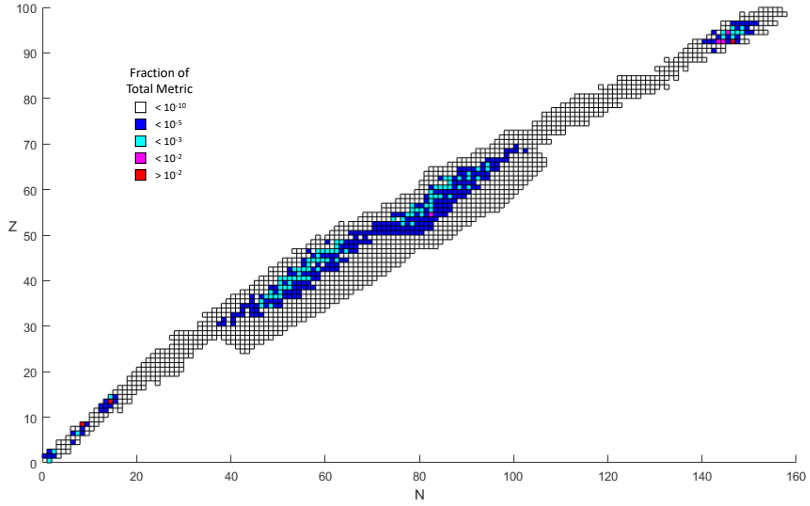

**(a)** 500 MWd/MTU Final Burnup




**(b)** 10,000 MWd/MTU Final Burnup




**(c)** 20,000 MWd/MTU Final Burnup




**(d)** 40,000 MWd/MTU Final Burnup

**Figure 4.16:** Nuclide charts of the mass distributions for the sample problem with varying final burnups.

### 4.3.2 Enrichment Dependence

Thermal reactors are typically fueled by low enriched uranium (LEU), that is a uranium-based fuel with a weight percentage of uranium-235 that is below 20%, and most light water reactors operate with fuel that has an initial enrichment of 3-4.5%, though higher enrichments can be used to facilitate reaching higher burnups and heavy water moderated reactors, like CANDU, operate with natural uranium. This LEU is enriched from its natural abundance of 0.711% U-235 by weight, generating depleted uranium — tails material that has below the natural abundance of U-235 — as a byproduct of this enrichment that can be used as filler material or to down-blend re-purposed high enriched uranium (HEU) and weapons grade material. While not many reactors exceed 6% enrichment due to criticality safety concerns, there are reactor designs that operate with lower enrichments and even natural uranium, like the CANDU reactors. The fuels initial enrichment can have an impact in the final composition of the UNF and alter the neutron spectrum. Lowering the enrichment hardens the neutron spectrum — shifts the average neutron energy to higher energies — as more of the fission reactions are the result of fast fission and fission of heavier actinides, and U-238, being a larger component of the fuel, absorbs more neutrons in the epithermal energy range due to its large radiative capture cross section resonance at 6.67 eV. Origen standard libraries capture this initial enrichment dependence by having created different libraries for a set of expected enrichments for each reactor design and using high order interpolation methods to generate intermediate enrichment libraries. These initial enrichments considered are between depleted uranium at a minimum of 0.5% enriched and a maximum of 6% enriched for most reactor types, with the only notable exceptions being CANDU reactors that only have libraries for 0.711% and a few libraries for experiments and research reactors involving HEU [32].

As a comparison over the range of available initial enrichments in the Origen standard libraries, including the one used in the depletion calculations of the unaltered sample problem, contributions to the four metrics, that have been analyzed previously. These initial enrichments are 0.5%, 1.5%, 2.0%, 3.0%, 4.0%, 5.0%, and 6.0% for the PWR libraries used in the sample problem. The most significant differences are plotted in figure 4.18, where

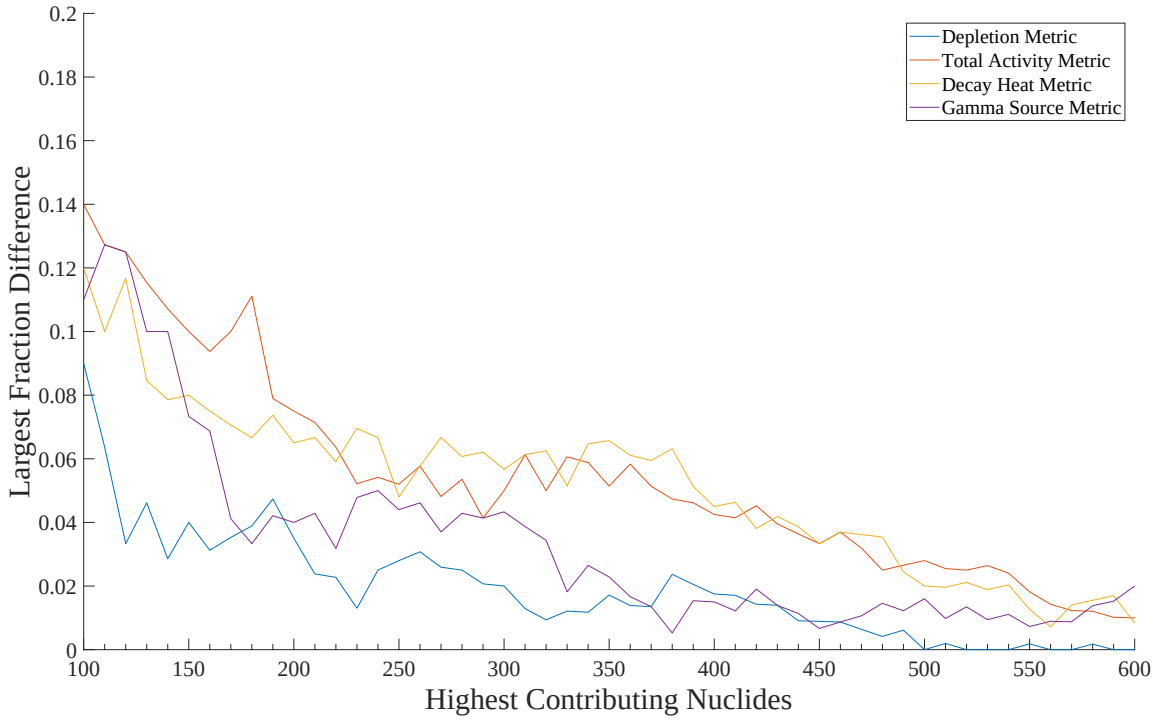
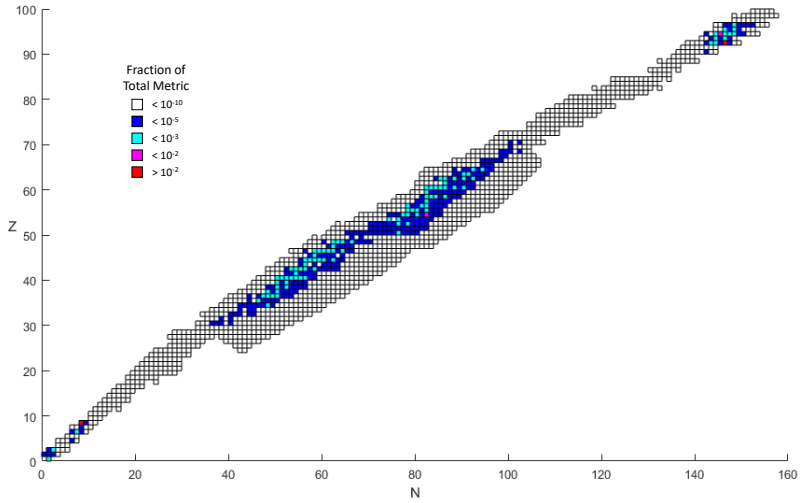



**Figure 4.17:** Graph of the fewest nuclides shared between initial enrichments of U-235 for a given metric type relative to the number of most highly contributing nuclides compared.

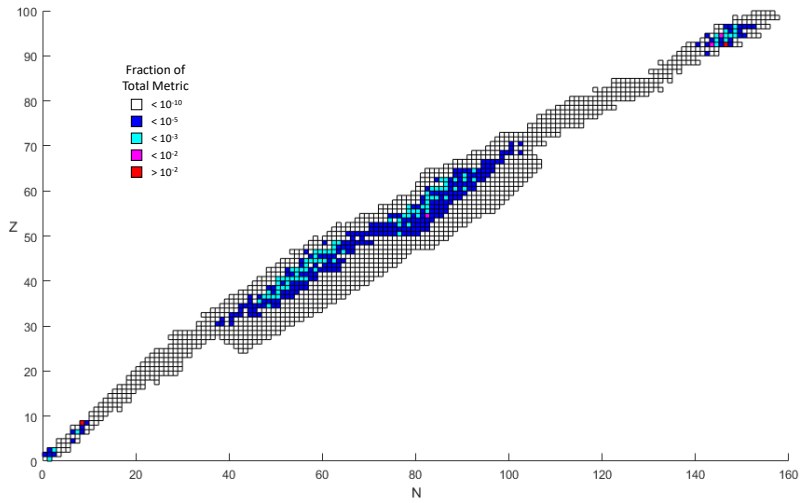
similar to the burnup inventory comparison, the two most extreme cases had the greatest differences in inventory at all ordered contribution list lengths for all problem types. However, the fractional difference in inventory decrease with increasing inventory list length for all metric whereas for burnup the fractional difference was nearly constant. This indicates that any dependence the reduced libraries inventory have on initial enrichment in the reducing problem can be offset by not over-optimizing the reduction and allowing for a slightly larger final library. This increase in library size is not a concern, as will be chapters 5 and 6, as the differences are minimal at the library sizes the reduction reaches at practical error bounds.

The minimal difference in the contributors at different initial enrichment levels can be also be seen in the mass distribution for the sample problem at those enrichments. The changes in the distributions, visualized in figure 4.18, are slight between sub figures despite having relatively large changes in initial enrichment. The most obvious difference is a shift from heavier actinides, in the range of plutonium to curium, to the mass being concentrated in the uranium to plutonium range for the actinides. There is also a shift in the mass

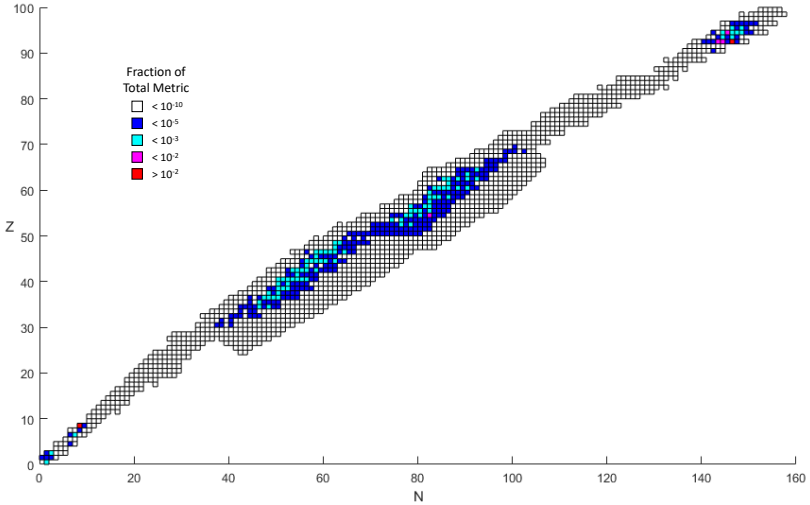
distribution in the fission products to lighter nuclides accounted for by the difference in fission yields from fission of Pu-239 and U-235 (figure B.1 in Appendix B), where the majority fissioning material at lower enrichments is Pu-239 and U-235 at higher enrichments. These differences though, as emphasized in figure 4.17, are slight to the point of being difficult to identify through visual inspection and do not result in any restrictions that are not present in the standard Origen libraries. These inventory differences are also entirely attributed to minor re-orderings in which the missing nuclides are within 10 positions in the list order, but past the cutoff length. At more similar enrichments the differences are also minimal, as any 2 enrichments within 2% U-235 content have no more than 4% difference in list order for any metric giving a more than sufficient range that different initial enrichment libraries can use the same nuclide set without a significant increase in error. The overall similarity in the nuclide sets between enrichments is favorable for having optimized library reductions across all enrichments and the robustness of this set would allow for tracking of nuclide concentrations across all enrichments without having to create a larger, unioned set of nuclides. The robustness of the nuclide set for each problem type further proves the initial assumptions correct and therefore validates the methods in another variable space.



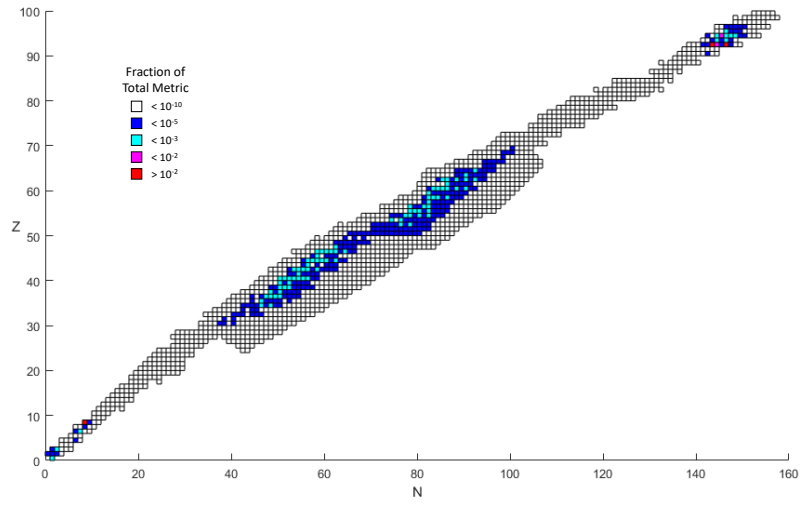

**(a)** 0.5% U-235




**(b)** 2.0% U-235




**(c)** 4.0% U-235




**(d)** 6.0% U-235

**Figure 4.18:** Nuclide charts of the mass distributions for the sample problem with varying initial enrichments of U-235.

### 4.3.3 Irradiation Condition Dependence

The structure of the irradiation in a depletion problem can significantly alter the relative concentrations of nuclides, while maintaining the total irradiation characteristics. That is, by altering the order and length of irradiation and cooling, the same burnup, total cooling time, and total irradiation time in a simulation can yield significantly different results for nuclides with unstable predecessors. A similar effect was observed in the comparison of final burnups where a lower flux resulted in the decay pathways being more important during irradiation cycles than they were at higher fluxes. This relationship is a result of reaction transition rates being flux dependent, but decay rates are not, thus the fraction of loss to decay increases at lower fluxes. However, in this case it isn't intra-cycle decay that presents the difficulty but the structure of the inter-cycle cooling. The cycle structure plays an important role in nuclides that main path of creation is from neutron capture by an unstable parent nuclide. The inter-cycle cooling allows the unstable parents to decay, while nuclides that are created from the decays see an increase in their concentration.

In the case of passive NDA techniques and inverse depletion methods this can alter the ratios of significant marker nuclides if one marker has an unstable parent (like Cm-244 and Rh-106) and the other does not (like Cs-134 and Eu-154). Thus changing inter-cycling cooling times is of concern as a method of masking the irradiation conditions of the fuel if they were altered to facilitate weapons material production. In reactor physics, the presence of inter-cycle cooling is also important as Xe-135 has period of time to reach its steady state levels and a period of time after shut down where its concentration peaks, precluding any immediate startup activity. This behavior is due to it almost exclusively being produced as a decay product of I-135 as thus Xe-135 has a delay associated with its 6.57 hour half-life (known as the "Iodine Pit" phenomenon). The steady state levels of Xe-135 are reached approximately 50 hours after startup and peaks about 11 hours after shutdown. Many other nuclides also have this delayed behavior and so to fully simulate them, shorter time steps at the beginning and end of power cycles are commonly used, and occasionally a "power ramping" step as well.

It is difficult to know the exact cycle structure that the final use problems will have and so the reduced libraries need to be expected to be able to accurately model a large range of situations from the same reducing problem. To assess the viability of the using a simple reducing problem (i.e, one that is accurate for the total irradiation conditions but with no temporal discretization) a comparison was done of the ordering of contributions from nuclides under varying irradiation and cooling conditions that had the same final burnup, irradiation time, and total cooling. The four cases were:

1. No cycle information - an irradiation reaching the final burnup, followed by a zero-power steps reaching the total cooling time, i.e., the sample problem with no temporal resolution.

2. Standard cycle structure - Three full power cycles, with two short inter-cycle cooling periods and a final long cooling period, i.e., the sample problem.

3. Highly detailed structure - The same cycle structure as the standard cycle case but with the time broken into much smaller time steps and power ramping.

4. Alternate cycle structure - Five full power cycles, with 4 short inter-cycle cooling periods and a final long cooling period.

Cases 1 through 3 represent the same problem at different resolution levels. Case 4 is a completely different cycle structure intended test the opposite of case 1 when compared to the sample problem, or a more extreme case of irradiation history simplification when compared to case 1. Figure 4.19 demonstrates though, that none of these difference in cycle structure alter significantly alter the contribution ordering. The largest difference were between case 1 and case 4, but even then the average differences for the four metrics was only a 1% fractional difference in inventory and a max difference of 6%. These differences are very similar to those comparing case 1 to either case 2 or case 3, as those two case have no differences with the case 4 inventory at almost all contribution list sizes. Even for the largest discrepancies, these amount to differences of fewer than 5 places in the ordering of the contribution sorted nuclide set. This leads to the conclusion that the factors affected by the change in cycle structure are insignificant compared to the macro irradiation history

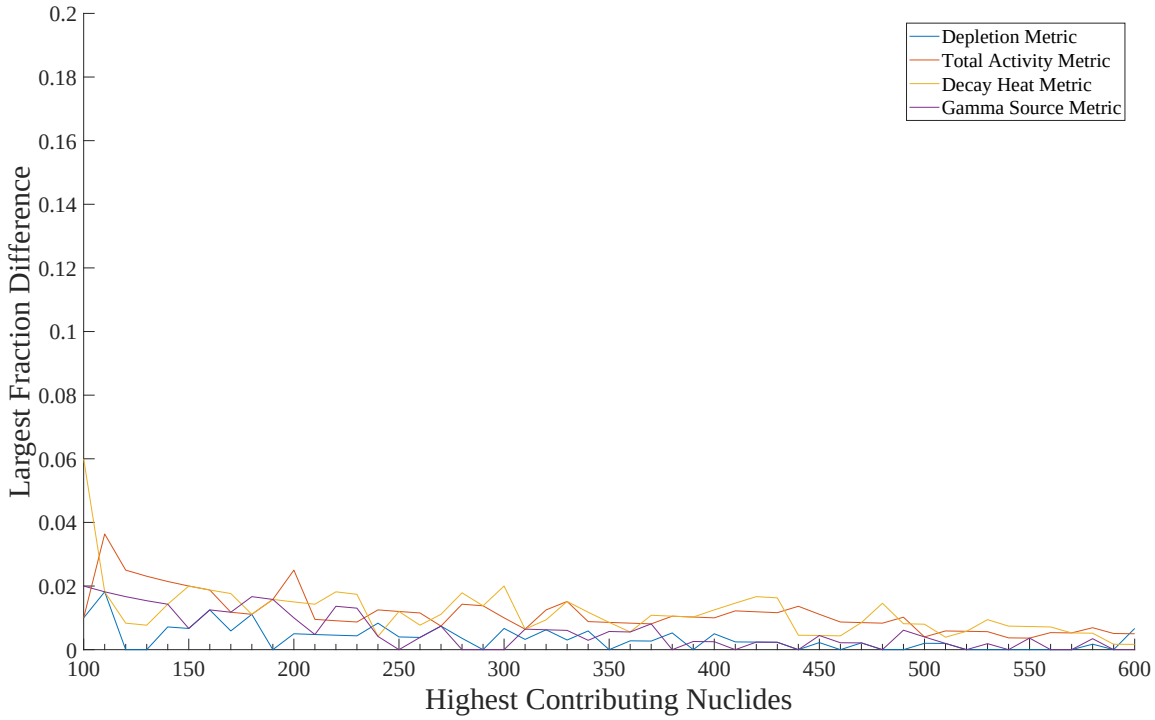
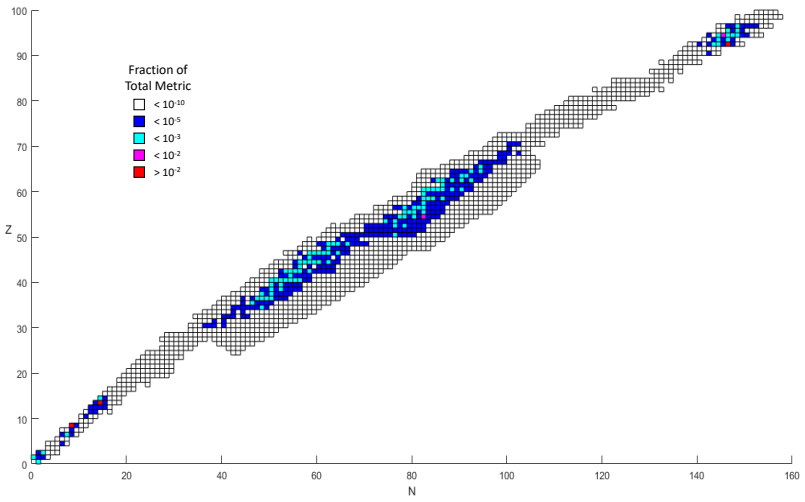
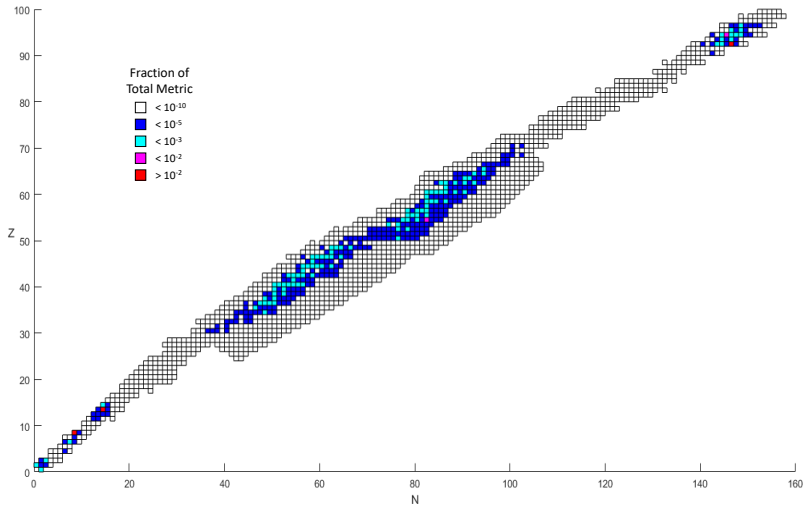



**Figure 4.19:** Graph of the fewest nuclides shared between different levels of detail and different cycle structures for a given metric type relative to the number of most highly contributing nuclides compared.

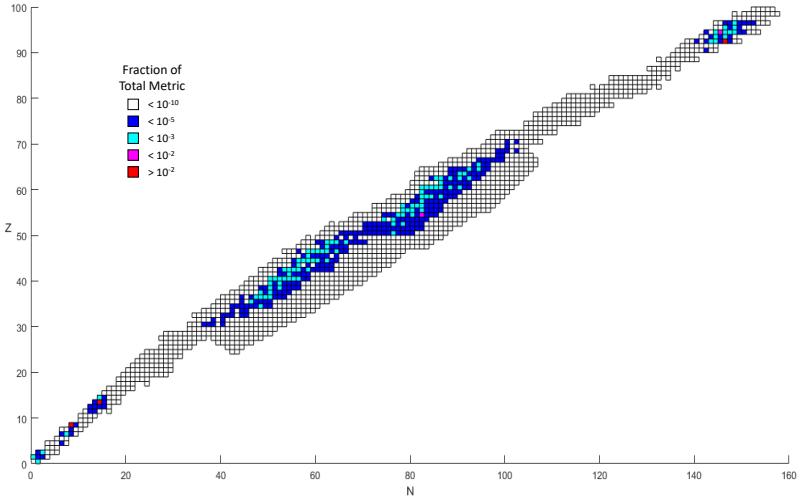
variables. This conclusion is further emphasized by figure 4.20, in that only the lowest tier of nuclides with sufficiently high contributions to be depicted in the chart change between the cycle structures, show any changes. These changes are also restricted to at most 9 nuclides not being depicted between case 1 and case 4, but only changes to 2 nuclides masses are sufficient to change their tier between the sample problem case and the case with two more power cycles. Besides further proving the key assumption to my method, this also informs that reducing problem can be very simple for all but the most strictly optimized problems and the user has no need to consider how to best detail the irradiation history for optimal resolution as the effects are smaller enough to not impact the reduction.



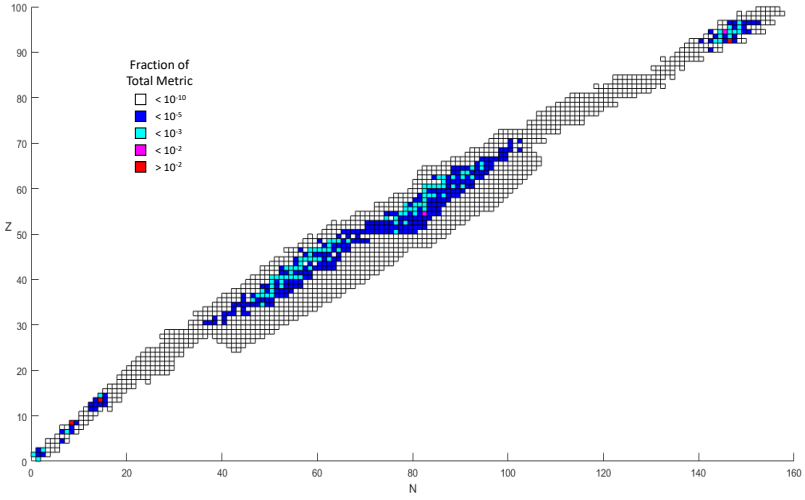

**(a)** 1 irradiation cycle and 1 cooling cycle




**(b)** 3 irradiation cycles, 2 down cycles, and 1 cooling cycle




**(c)** 3 irradiation cycles, 2 down cycles, and 1 cooling cycle wiht a detailed power history and fine time mesh




**(d)** 5 irradiation cycles, 4 down cycles, and 1 cooling cycle

**Figure 4.20:** Nuclide charts of the mass distributions for the sample problem with varying detail in cycle structure.

### 4.3.4 Reactor Type Dependence

The last of the major component of the reducing problem definition that has a significant impact on the nuclear data in the library is the reactor type. Though it is mostly commonly used for boiling water reactors (BWR) and pressurized water reactors (PWR), Origen supports many reactor types. Four of the commercial reactor types that Origen has standard libraries for are PWR, BWR, advanced gas reactor (AGR), and CANDU reactor types. These four reactor types have libraries covering similar conditions, with the exception of the CANDU libraries only supporting natural uranium enrichment, and are thus appropriate for a comparison to determine the effect of reactor type on the contributing nuclide set. Differing from the previous comparison, this one does not have an impact on the viability of my reduction method or the robustness of the reduced library, however, its outcome does have an impact on the potential range of applications of the method. If the contributing nuclide sets do not strongly vary between reactor types then it would be sufficient to determine the reduced libraries nuclide set and structure from a single optimization and apply it to all similar libraries rather than having to optimize each library independently for the same reducing problem.

The reactor type impacts the nuclear data values through the materials and the geometry. The material variables are structural components, cladding, moderator type and density, and coolant though burnable absorbers in the fuel pins, breeding blankets, neutron reflectors, criticality control devices, and measuring devices can all be present in the models that generate the depletion libraries. The geometry can vary in number of pins per assembly, the size of the pins, the fuel shape (pins v.s. pebbles), the shape and size of the assembly, the location of control devices and holes for measuring devices, the distance between assemblies and pins (commonly referred to as the "pitch"), the thickness of structural materials, and the height and width of the reactor. All of these variables impact the flux spectrum and specific power both globally and locally for the reactor. It is beyond the scope of this work to investigate the effect of the variables on the order of the contributing nuclide set, but it is possible to approximate their bulk impact through their pre-generated standard libraries. To this end a comparison was done using the four aforementioned reactor type

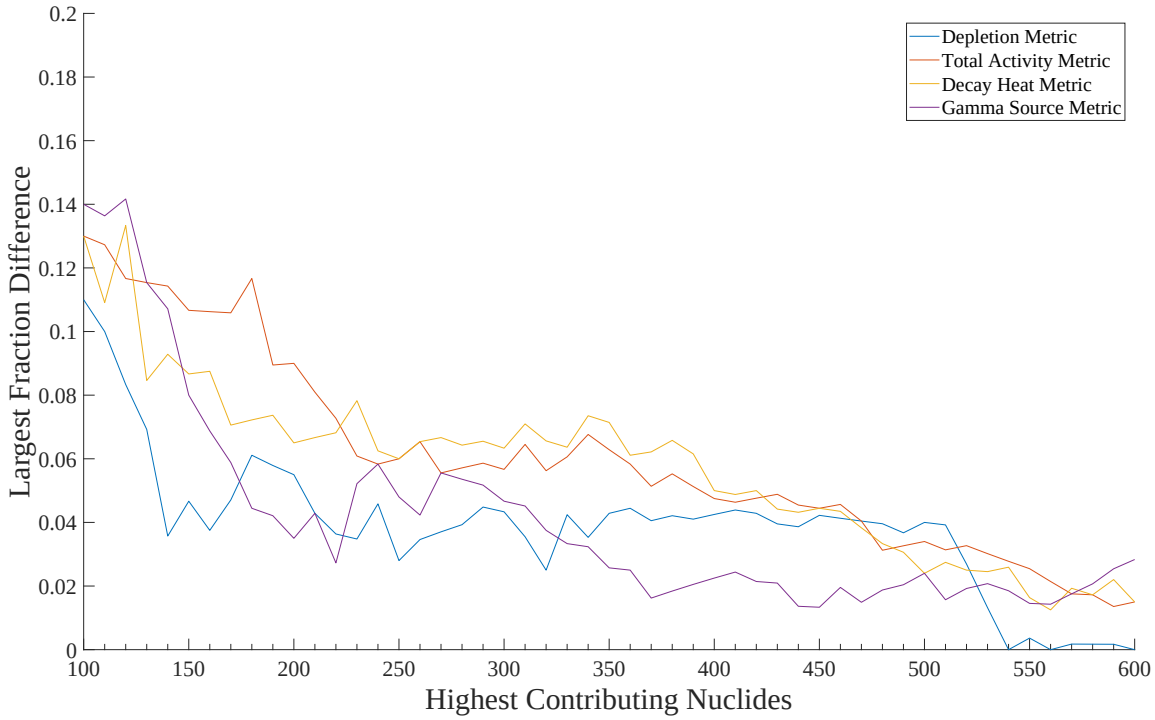
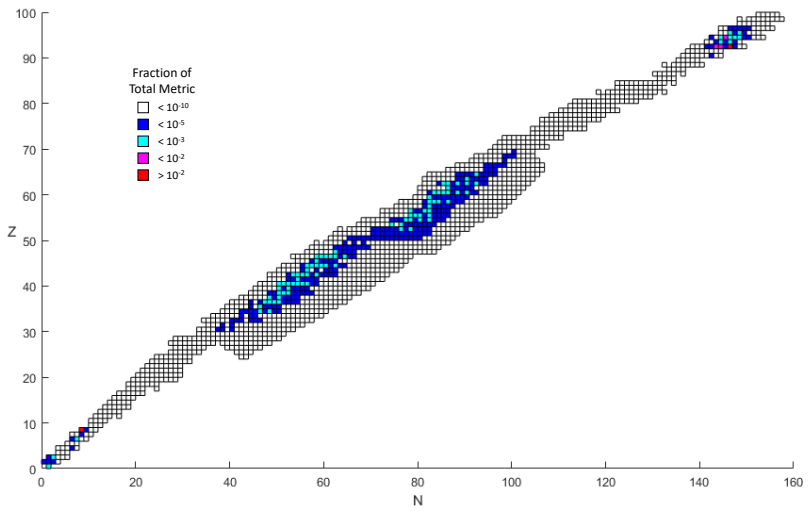



**Figure 4.21:** Graph of the fewest nuclides shared between reactor types (candu28_e07 and w15_e50) for a given metric type relative to the number of most highly contributing nuclides compared.

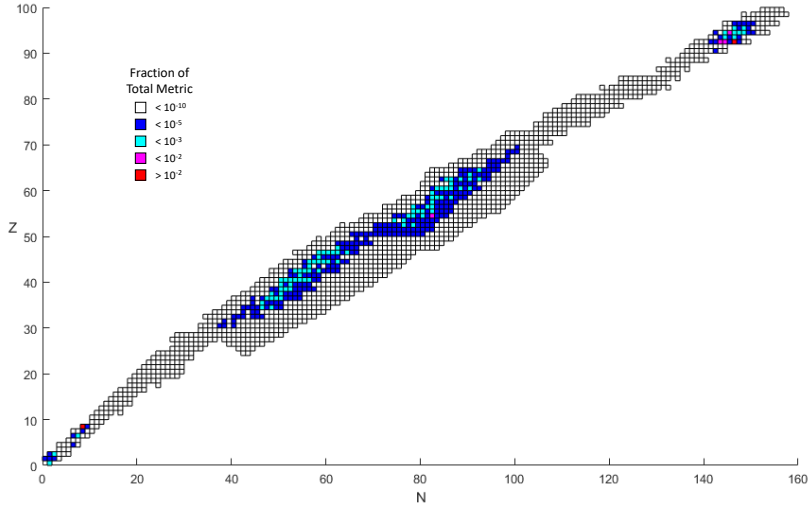
under the conditions of the sample problem with the exception of using natural uranium for the CANDU problem but the sample problems irradiation history. The lattice libraries used for this comparison are Westinghouse 15x15 (w15_e50) PWR, GE 10x10 (g10_e50w05) BWR, Advanced Gas Reactor (agr_e50), and CANDU 28 element bundle (candu28_e07) heavy water moderated reactor.

The comparison of the ordered list of contributors is plotted in figure 4.21 for the four metrics being analyzed, with the greatest differences being between the CANDU reactor library and the PWR library. However, this can be misleading as the CANDU library presented the same difference in inventory when compared to any of the other three reactor types, and if compared to figure 4.15 the differences strongly resemble those resulting from a large difference in initial enrichment to the point where the differences in all four metrics are nearly identical across the entire range. That they are so similar indicates that the reactor type did not have an effect equal to the initial enrichment, and this is supported by the

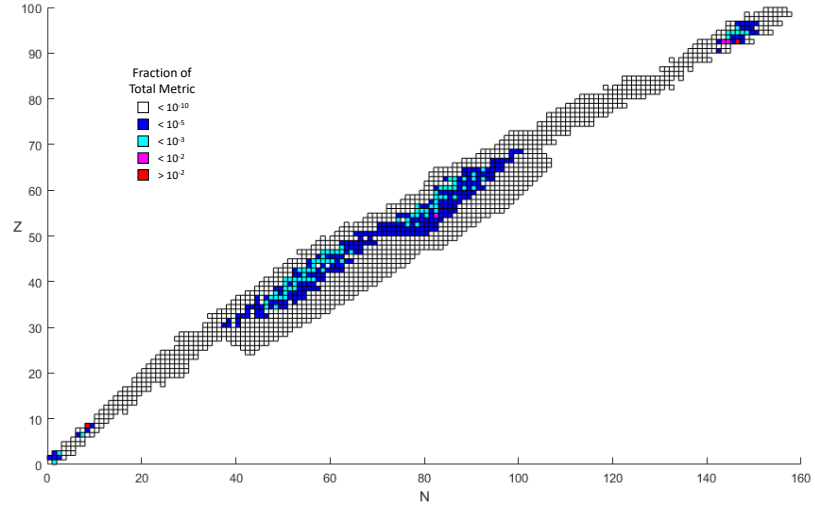
comparison for the other reactor types where there were 4 nuclides or fewer differences in inventory at any ordered list length. In figure 4.22 this lack of variance amongst the tree reactors types with the same initial enrichment is visible, with two major exceptions: the tier of concentration for Xe-156 in the PWR nuclide chart, and the tier of concentration for Pu-239 in the AGR nuclide chart. In the case of Xe-156, the concentration is 94% of the value found in the BWR and AGR, but this is just sufficient to be below the threshold for that tier despite being in the same place in the ordered lists. The difference in the Pu-239 concentration is more significant, being half of the concentration in the PWR and BWR, however this decrease in concentration only results in it dropping two places as compared to those reactors — from fourth to sixth. Though these difference appear drastic for major nuclides, they do not significantly impact which nuclides are contributing and have almost no impact on the order of contributors. The conclusion that is drawn from this is that it is indeed possible to use the same optimized nuclide set for multiple reactor types being used for the same reducing problem as the effect of changing the reactor type is insignificant to the effects from burnup, initial enrichment, and problem materials.



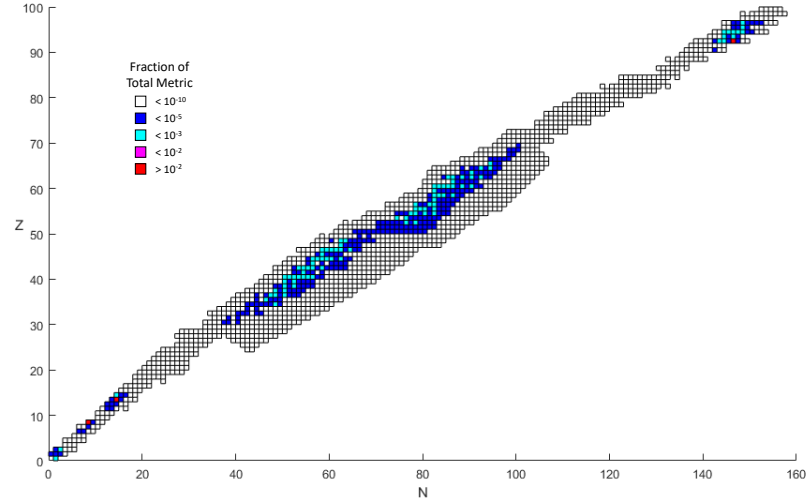

**(a)** Pressurized Water Reactor




**(b)** Boiling Water Reactor with 0.7 moderator density




**(c)** Advanced Gas Reactor




**(d)** CANDU Reactor with natural enrichment uranium

**Figure 4.22:** Nuclide charts of the mass distributions for the sample problem with varying reactor types and following restrictions associated with their designs.

# Chapter 5

# Library Reduction Effects

The weighting methods act as approximations (or estimates) of the importance of a nuclide and under ideal weighting conditions – the expected increase in error if the nuclide is excluded. In practice, due to the complex nuclide-nuclide and nuclide-problem interactions, this is not the case. The weighting methods in general have more difficulty in capturing the complexity of the nuclide-nuclide interactions, the system determination methods are focused on accounting for this behavior. How well the system determination methods compensate for the weighting method are dependent on several factors and no one combination of weighting method and system determination method is best for all problems and metrics. With this said, there are certain combinations that perform well under most scenarios, even if they do not result in the most optimized library reduction. Similar to the previous chapter, this chapter will be focused on analyzing the reduction behavior for the sample problem under the four metric types, with special focus given to the depletion metric as it is most similar to the benchmark of the preliminary work.

This chapter concentrates on the subsystem determination methods and how they work using the different weighting methods and metric types. Analysis of the error resulting from the removal order of the nuclides and error expected from the weighting scheme is key to determination of the most suited and robust reduction methods. Comparison of the errors from the method combinations as well as analysis of the cause of these errors in relation to the mass-flow chains is the focus of this chapter. For analysis of these methods in relation to

meeting the accuracy and computational efficiency goals, as well as the convergence of these methods to the optimized reduced library, see chapter 6.

## 5.1 Threshold-Truncation Method

The threshold-truncation method is the simplest method to determine the order of nuclide removals, and at its core it is based on the ideal weighting assumption. As stated in the previous section and chapter 3, this assumption is not practically achievable which leads to many of the issues described. However, because of this assumption, this system determination method is ideal for gauging the adequacy of the weighting methods in capturing a nuclides importance to the system and metric. This can be judged based on figure 5.1 for the depletion metric (the matching graphs for the other three metrics can be found in Appendix C), which depicts the error in the metric at all library sizes based on each of the weighting methods.

Over most of the range of errors, the optimum weighting is one of the contribution weighting methods, with second-degree weighting being the best over most the range of higher errors and either 0.5 or degree 1 weighting being the best at lower error thresholds. However, the weighting method with the most unique convergence, being significantly worse than the rest of the methods for all metrics, is the direct contribution method. It is simply the case that this metric does not provided sufficient information to determine system importance, and further belies the fact that nuclides can cannot be chosen solely due to a single characteristic. For the sample case, the direct contribution weighting removed Mg-27 and Al-28 as the sixth and seventh removal respectively and U-239 as the $689^{th}$ removal. All three of these are very important nuclides, two being the neutron capture products of two of the source materials, but are given no importance. The reason that the other contribution weighting methods due well by this measure, but the two adjoint methods do poorly, is because they concentrate weight around those nuclides that most directly contribute to the metric and thus protect their important pathways, which is exactly what the methods were designed to do: preserve contributing transition rates. The adjoint methods distribute the weight across all potential pathways, detracting from the importance of the end of

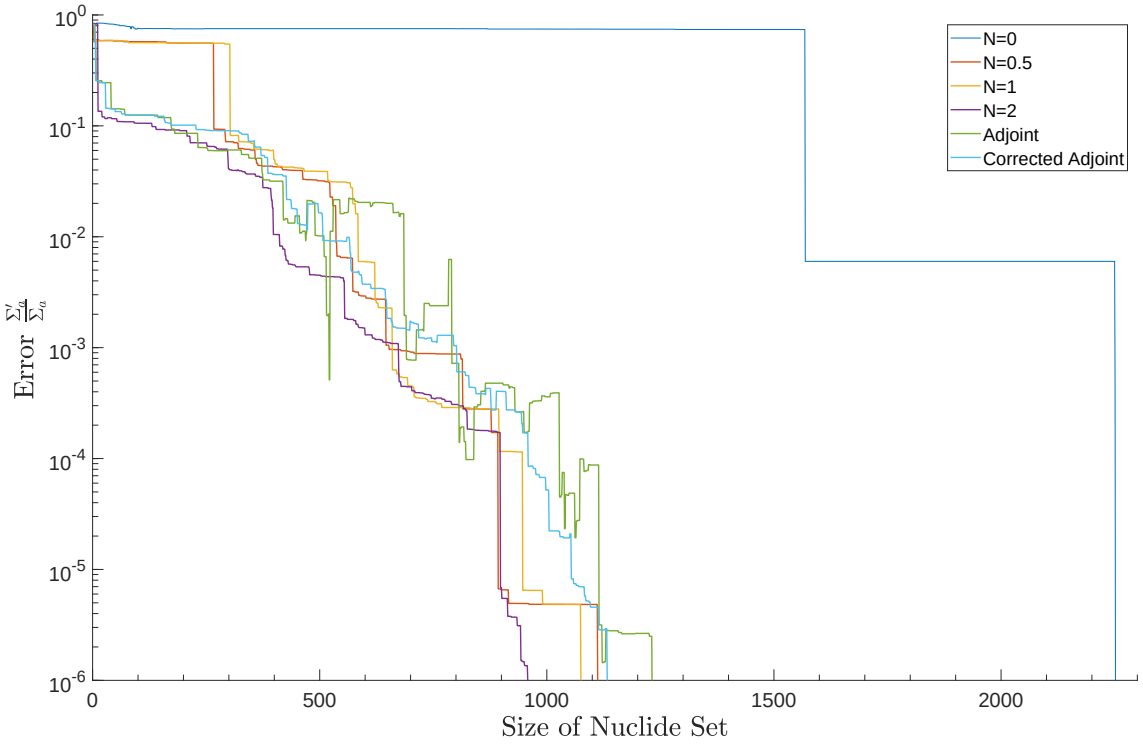



**Figure 5.1:** Relative errors at a given library size reduced with the threshold-truncation method and the given weighting method.

the path that is the actual contributor to the metric, instead favoring the preservation of nuclides that are at the intersection of many mass-flow chains. The corrected adjoint method partially mitigates this by re-emphasizing contribution nuclides daughter products. This path weighting behavior is advantageous in other system determination methods but not as a direct measure of importance.

One of the key features of many of the weighting methods in figure 5.1 is the step like behavior including extended "plateaus" where the error does not increase following several dozen removals, and sections where the error will decrease with further removals. In both cases, it is the result of the weighting methods incorrectly estimating a nuclides importance. If the importance is underestimated, it results in the nuclide being removed prematurely and subsequent nuclides being isolated from the system, creating whole orphaned subsystems like the scenario described in section 3.3.1, but on a larger scale. These premature removals can result the elimination of important mass-flow pathways or the emphasis of parasitic

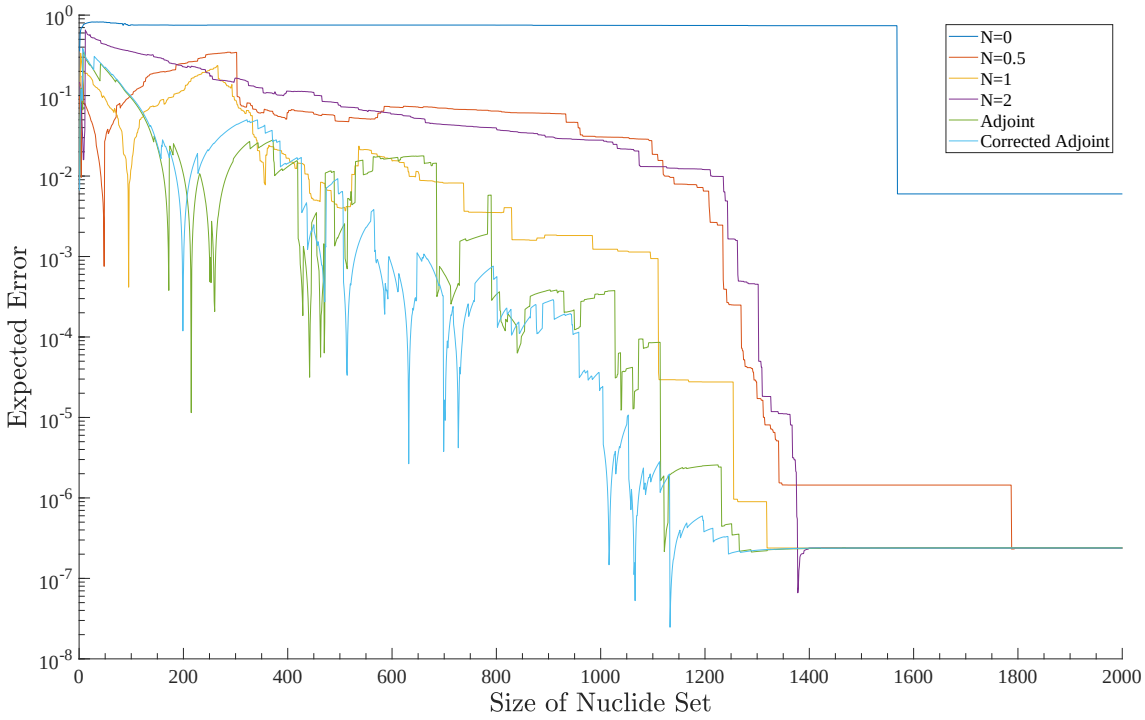



**Figure 5.2:** Absolute difference between the actual error at a nuclide set size and the error estimated by the given weighting method.

pathways. When these paths are removed, the mass-flow is redirected to different paths that may not contribute as significantly; when this flow is corrected by a future removal, the error is subsequently reduced. This correction can either be from the dominant path reasserting itself or more commonly, the result of offsetting errors reaching the same contributor through an alternate but less likely path. These steps are indicators of gaps in the importance information, and thus weighting methods that more completely capture the importances will be smoother and more importantly monotonically increasing. The higher-degree weighting methods and the corrected adjoint method have the most complete importance information by this measure, whereas the adjoint weighting and the direct contribution weighting are the most incomplete exhibiting many corrections and large error plateaus respectively.

By the nature of the library reduction process, the loss of information in the library will induce errors if reduced beyond a certain minimum nuclide set. This minimum error-less nuclide set is dependent on the metric and the reducing problem conditions, but is much larger than most meaningful reduction levels – it being a reduction of less than one half. To

account for this unavoidable error in the reduction, the errors from truncating the system by the weightings should also be assessed by the expected error from the reduction: that is, the error as a measure of the difference between the actual error in the metric and the fraction of the original weight being excluded via the reduction. This measure is commensurate with the error beyond what is expected from the weighting assuming that the weights are an exact measure of the importance, i.e., if the weight is is a perfect measure of the error that will be caused by its removal the expected error will be zero for all nuclide set sizes. This comparison to error and weighting is plotted in figure 5.2. Many of the features that were present in the error plot are still present in this comparison, indicating that the weighting also did not predict the errors to increase as much as they do and are therefore under valuing the importance of the nuclide being removed. However, there are three significant features to this plot that are indicative of characteristics that are not represented in the error. The first is that nearly all methods plateau in expected error between 500 and 1000 nuclides in the nuclide set. This would indicate that while the early removals (lowest-weighted nuclides) are undervalued, the intermediate-weighted nuclides — those that would fall into the two lowest tiers in the nuclides charts in chapter 4 — importance is accurately measured. The second feature is similar in that the large dip in expected error by all weighting method at very small nuclide sets would indicate the highest weighted nuclides are being overvalued. The last feature is primarily observed in the adjoint weighting methods, which is the apparent sharp decreases in error where the error changes sign. These two methods give the most accurate measure of true importance of a nuclide, or at least appear to be the most accurate predictors of the error that can be expected from removing a nuclide. This actually due to another case of offsetting errors and both exhibit oscillatory behavior as a result of this. The negative inversion points for these oscillations are all associated with the removal of an intersection nuclide (e.g., the removal at a set size of 632 being Sn-111 that has pathways to cadmium isotopes with moderate to large absorption cross section) and the positive inversions being removals that are significant to direct contributors. The intersection nuclides are so overly valued that the estimated error from their removal is in excess of the true error but the opposite is true for the contributing nuclides due to the weight being diffused across the system. These dips caused by the error inverting contrast to the dips in error of the

degree 1 (N=1) contribution weighting. These dips are caused by the importance of a region being concentrated in a few nuclides and those nuclides being removed offsetting the underestimation from the surroundings being removed. This correcting, rather than offsetting, behavior is preferable and contributes to the better overall accuracy of these weighting methods at capturing the importance of a nuclide.

## 5.2 Greedy Algorithms

As a direct response to the inadequacies of truncating the system, the greedy algorithms are designed to ensure full connectedness to the source material. As described in 3, this is achieved by expanding subsystems of the Transition System, centered on the source material, to include what is judged to be the most important nuclide on any of the subsystems peripheries. This method takes the onus of determining the nuclide shape from the weighting methods, but still relies on them to make the determination of path importance. The trade-off of the algorithms method of forcing connectivity is that it abandons the global scope that the truncation method has, restricting the knowledge of the method to make determinations based only on the peripheral nuclides. In figure 5.3 this can be seen to result in an advantage for the adjoint weighting methods in the significant improvement in the optimization convergence and convergence behavior of the corrected adjoint and adjoint weighting methods respectively. The corrected adjoint method on average converges to a nuclides set that is 100 nuclide smaller than the set with the equivalent error found by the truncation method. The greedy algorithm though has the negative consequence of the convergence worsening for higher-degree contribution weighting methods. This is the result of the algorithm selecting predecessors to multiple contributing nuclides rather than selecting the contributing nuclides and their products. This is because the predecessor nuclides are closer to the source nuclides, and are seen on the subsystem peripheries sooner. This isn't the case for the N=1 contribution weighting because if a predecessor nuclide is chosen then the contributing nuclide is then directly on the periphery. Furthermore, because the weights are still fairly concentrated for N=1, the contributing nuclides will have significantly higher weights than the surrounding nuclides. These effects are lessened for the more disperse

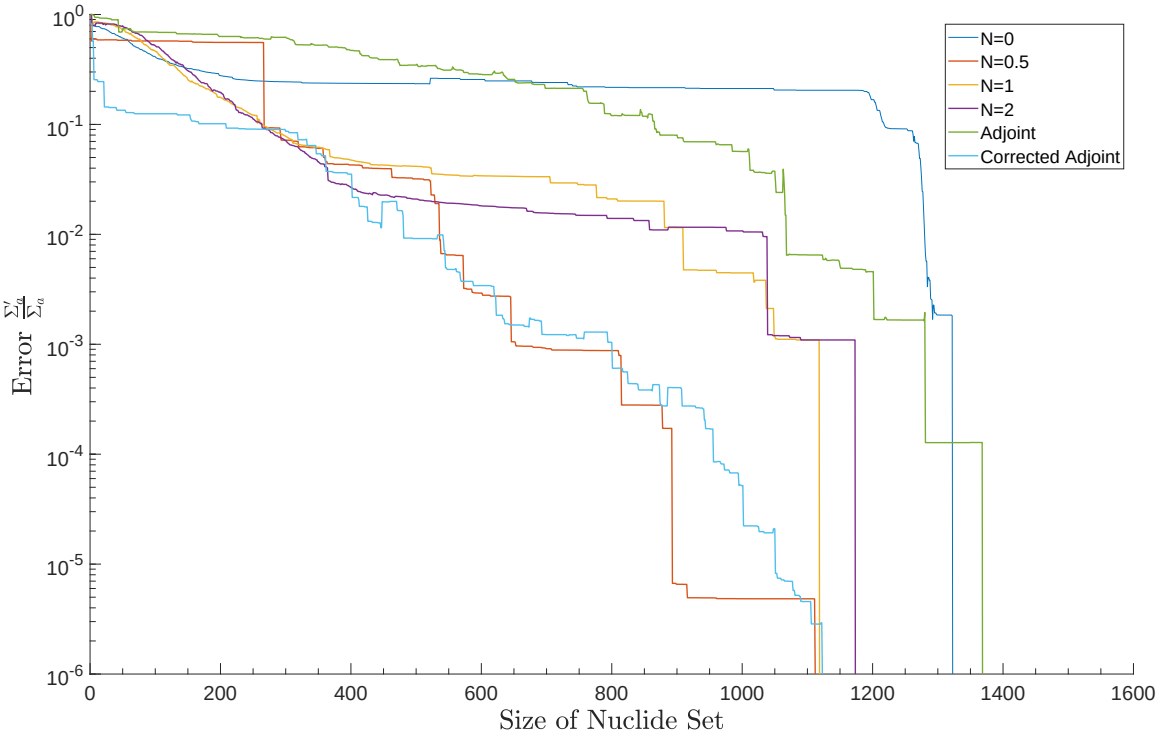



**Figure 5.3:** Relative errors at a given library size reduced with the weight-based greedy algorithm method and the given weighting method.

metrics, like total activity and decay heat, so the convergence for the different weighting methods are similar to those seen for the truncation method.

The difference in the convergence behavior and optimal weighting for the greedy algorithm between the metric types is a direct result of the algorithm's expansion search being effectively random for the weightings and metric types with limited contributors. In the diffuse metrics, the nearest contributors or their predecessors are commonly fewer than two transitions from a source material; therefore, the weights quickly intersect with the periphery of the search. However, for metrics like depletion and gamma source, where the contributors are several transitions from the nearest source, there may be no nuclides with any given weight for multiple subsystem expansions. In the case that no nuclide on the periphery has a weight the next nuclide to be chosen is first that was to be "seen". In this way the system should expand in the areas of the highest-weighted nuclides first, but after multiple expansions with no weighting information this can no longer be guaranteed to be the case.

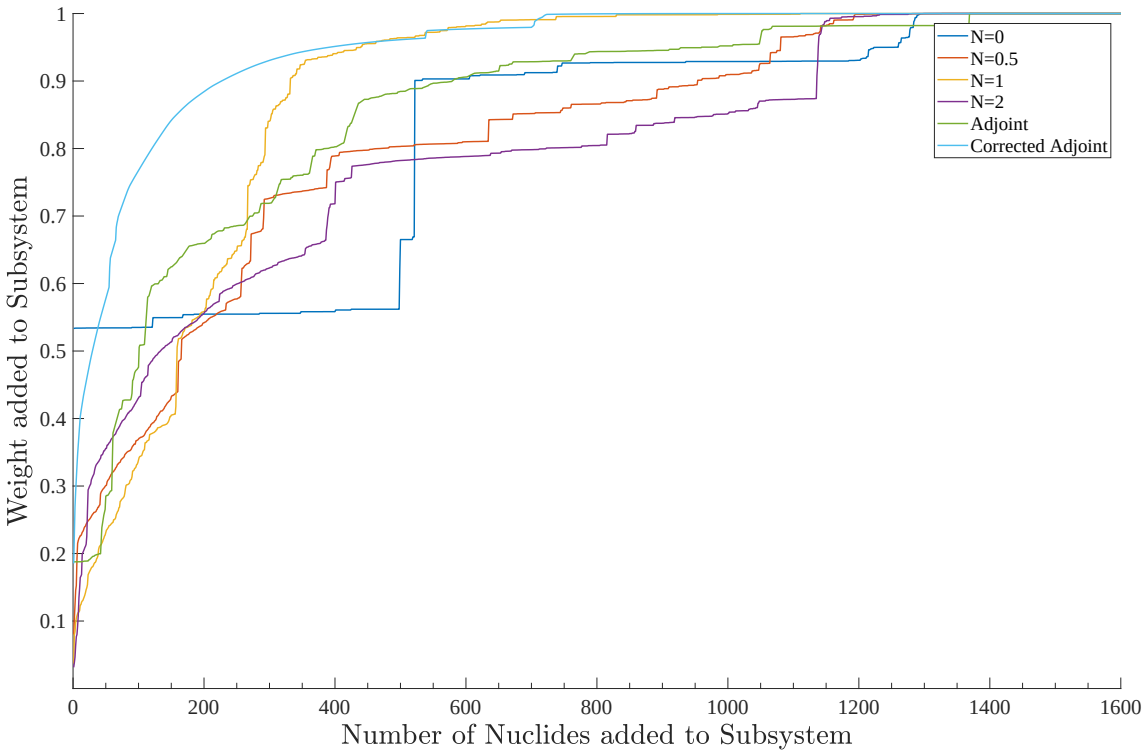



**Figure 5.4:** Relative weight in the subsystems at a total given size, excluding the weight of source nuclides that are required to be in the subsystems.

Figure 5.4 demonstrates this uninformed expansion as plateaus in the weight that has been included in the reduced system resulting from a nuclide being included that has no weighting. This blind search is most common in the weightings that least diffuse the total weight, like N=0 and N=0.5 contribution weighting. That behavior contrasts to the smooth increase seen in the corrected adjoint weighting. The weight being more dispersed for the N=1 and N=2 weightings does result in more informed traversal behavior, but the increase in degree from N=1 to N=2 crosses a transition point in the behavior where more nuclides need to be included for N=2 due to the weight being too dispersed. These observations for the case of concentrated metrics do not hold true for dispersed ones, as all weighting methods exhibit a smooth, rather than step-like, increase in subsystem relative weight.

As opposed to putting the onus on the weighting method to determine the correct pathways for the system expansion search to follow, the alternative greedy algorithm makes that decision based on average transition rates. The transition-based greedy algorithm

expands along the path of the greatest average transition rate either into or out of the subsystem. Theoretically this would eliminate the case of the algorithm having to make uninformed expansion decisions and follow important mass-flow chains as well as the user having to make the decision of the most adequate weighting method for the their reducing problem. The consequence of making these decisions based on transition rates though, is that the transition rate is a property of the reducing problem and not related to either the metric or the weighting type – making them irrelevant to the expansion order. This means this algorithm can be misled to include more common competing pathways rather than the contributing pathways, where as the standard greedy algorithm could be misled or blocked by a lack of correct weighting information. In the later case it can be compensated for by the choice of weighting method, but there is no currently developed method for accounting for the former. For the depletion metric of the sample problem, figure 5.5 shows how poorly performing this method can be, being worse than any of the weightings using the standard greedy algorithm. The difference in convergence is not nearly as pronounced for the more dispersed metrics, but in all cases the standard algorithm out performs it for several weightings.

The standard greedy algorithms determination of the order of nuclides to remove is not congruent with most of the weighting methods, as can be seen in figure 5.6. This is concluded due to the removal order based solely on the weights having a lower expected error across the range of nuclide set sizes for the N=0.5, N=2 contribution weightings and the adjoint weighting. However, for the N=1 contribution weighting there is almost no effect on the expected error and even a minute improvement in the direct contribution weighting. The most noticeable beneficial feature of this method is that the number of inversion points for all weightings have decreased but corrected adjoint weighting especially. The elimination of the oscillatory behavior from greatly over- and under-predicting error from a removal is due to subsequent removals starting from the end of a mass-flow chain and progressively approaching the source material. This results in far fewer instances of flow being redirected through parallel paths and the "snap" corrections when those competing paths are also broken. This also leads to the degradation of the performance for the adjoint weighting method as the contributing transition products will be removed earlier, being that they are at

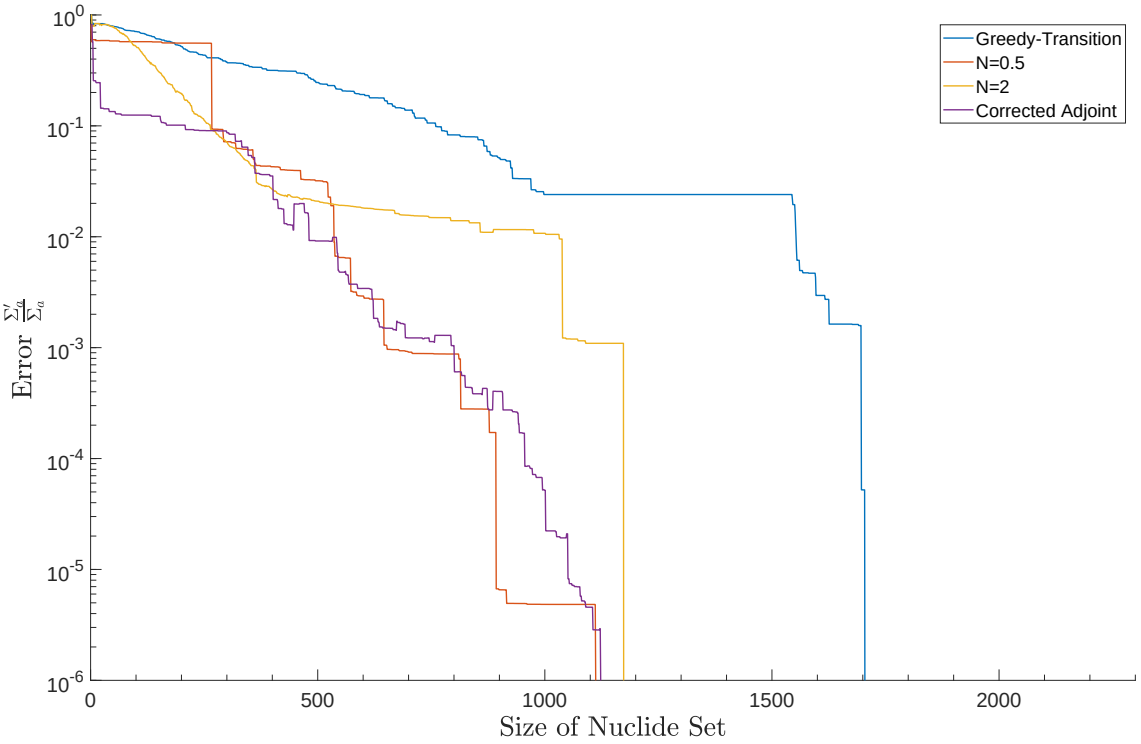



**Figure 5.5:** Relative errors at a given library size reduced with the weight-based greedy algorithm or the transition-based greedy algorithm.

the end of the contributing path and the method does not denote their importance. Despite these weaknesses of the greedy algorithm, the method results in more robust reductions, being fully connected systems, but larger reduced sets at any given error threshold.

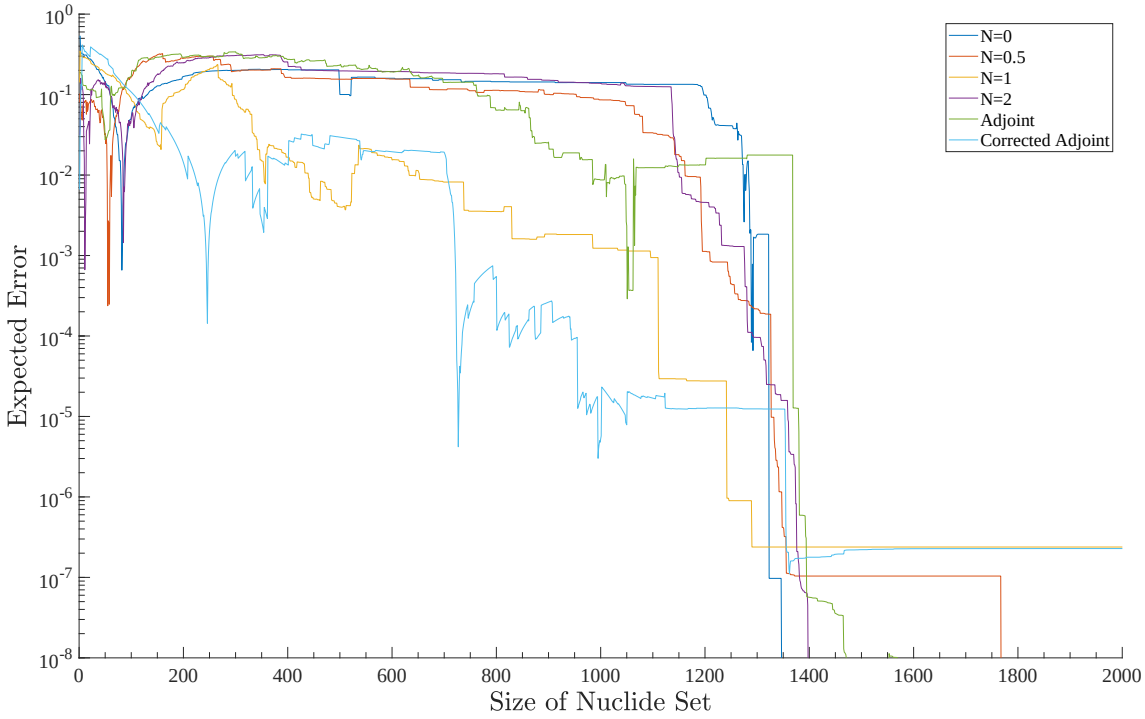



**Figure 5.6:** Absolute difference between the relative error at a nuclide set size and the error estimated by the given weighting method.

## 5.3 Cutting Planes Algorithms

The last of the natively implemented system determination methods, and the most complex, is the cutting planes algorithm. As compared to the previous two methods that make decisions by considering nuclides as individuals, this method considers the global impact of a nuclide and its place in mass-flow chains without the consideration of the weighting method. The dominant global structure of the transition system — which by its nature is difficult to traverse being directional and cyclic — is determined through the space transform outlined in chapter 3. The resultant tree structured graph in the distance-weight subspace is depicted in figure 5.7 using number density rather than metric contribution or weighting for the sake of generality and clarity. In this figure the nuclides generated from successive neutron captures of the activation/light source nuclides as well as the heavier actinides are clearly defined branches of the tree with nuclides created through decays branching off of the main neutron capture path. While they are less apparent, the space transform also creates

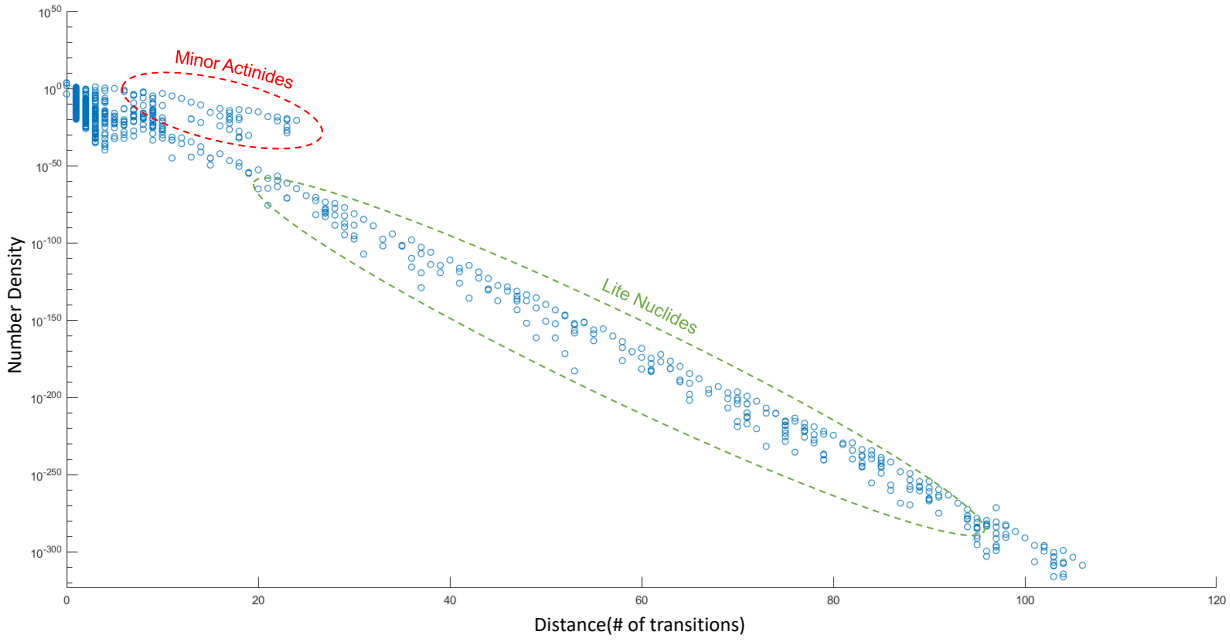



**Figure 5.7:** Nuclide chart of relative contributions in the standard problem for the depletion metric.

many much shorter branches for the fission products. However, because no nuclide in the fission product section of the transition system is more than 9 transitions from a source material, this region becomes very visually congested. The more dominant the shortest path to creation is, the more defined the branch will be in the subspace. Alternatively, the more competing loss pathways by a parent or gain pathways by a product, the less accurate the resulting reduction will be. In this space transform there is a clear relationship between the number of transitions in the dominant creation pathway of a nuclide and the number density of that nuclide, implying that the shortest paths are highly dominant. The linearity of this relationship — on the semi-log scale in the figure — is the result of many of the nuclides in the two most most visible branches having similar neutron capture cross sections and there being no competition via alternative creation paths.

The subspace transform with distance measured by the fewest number of transitions for creation should result in the most reduced for a given error threshold. This method should also be a valid estimate of the most dominate pathway and is supported by the proportionality relationships observed. In figure 5.8 this relationship clearly creates a

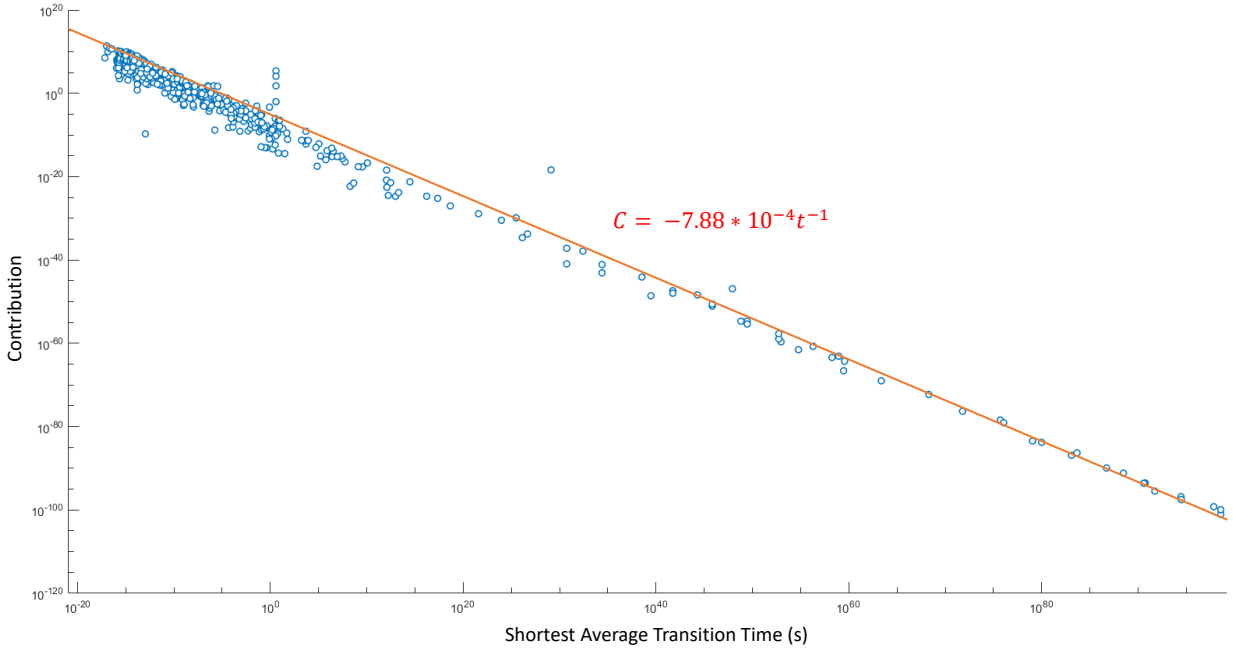



**Figure 5.8:** Nuclide chart of relative contributions in the standard problem for the depletion metric.

limiting equation for the contribution that is proportional to the shortest average time to traverse the path. In the case of figure 5.8 the shortest path is determined by the traversal time as calculated in equation 3.2, rather than by number of transitions. While this space transform is significantly more computationally expensive to perform, it has stronger theoretical foundation as it acts to estimate the results of the depletion calculations as a single combined term inversely proportional to time. This equation is the limit of the contribution that a nuclide could be expected to have if the shortest path was the only path, with no competing gains or losses. The fit of this limiting equation also demonstrates that the variance resulting from the magnitude of the weighting factor. This is because the number density will follow the trend defined by the Bateman equations and is proportional to path distance, whereas the weighting factor has no relationship to a nuclide's location in the transition system and will thus be the key factor in determining the spread of the points in figure 5.8. The lack of significant spread indicates the weighting faction is not as significant to estimating a nuclide's contribution as the number density, and this method that best utilizes that trend should result in a more optimal system reduction.

Given the manner in which the cutting planes method determines removals, with the method determining connectedness and the pathing to nuclides, the most compatible weighting type would be expected to be one that gives clear end points to a branch of the subspace tree to make the system reducing cut. The weightings that result in the best optimization are the first and second degree contribution weighting methods, for which the second degree weighting is the most reduced for most error thresholds. Though, the N=0.5 contribution weighting performs similarly to the higher degree weightings for the depletion metric in figure 5.9 the superiority of the higher degree weightings is more pronounced with the other metrics, and the more disperse metric in particular. The superiority of these methods is the result of them incentivizing the cutting planes method to include the contributing nuclides parents and daughters in the shortest paths which leads to more accurate calculations of their number density. This expansion of the transition system in the regions of contributing nuclides also occurs from the N=0.5 contribution weighting and the corrected adjoint weighting, however this expansion is limited to the loss products of the contributing nuclides and not the parents. Furthermore, the adjoint-weighted methods conflict with the cutting planes method resulting in worse optimizations than if the system is truncated on weights. This conflict arises from the the adjoint methods distributing weight along the burnup chains rather than in a concentrated area making it more difficult for the algorithm to determine the optimal cut along a branch. However, this behavior is difficult to show directly as the cutting planes method does not allow for the expected error comparison as the other methods did due to the cuts not being determined in a single nuclide sequence. The method instead determines cuts based on distance-weight correlations that may remove multiple nuclides concurrently, resulting in no definite relationship between what nuclides are in a system and their weights. However, what can be observed from the relative error far smoother and monotonic increase in error, indicating that the nuclide removals are not resulting in significant redirections or re corrections of mass-flow for the optimal weighting methods. Similar to the greedy algorithm, this characteristic of the reduction results in more robust reduced system. However, in this case the robustness does not come at the cost of the size of the reduced nuclide set.

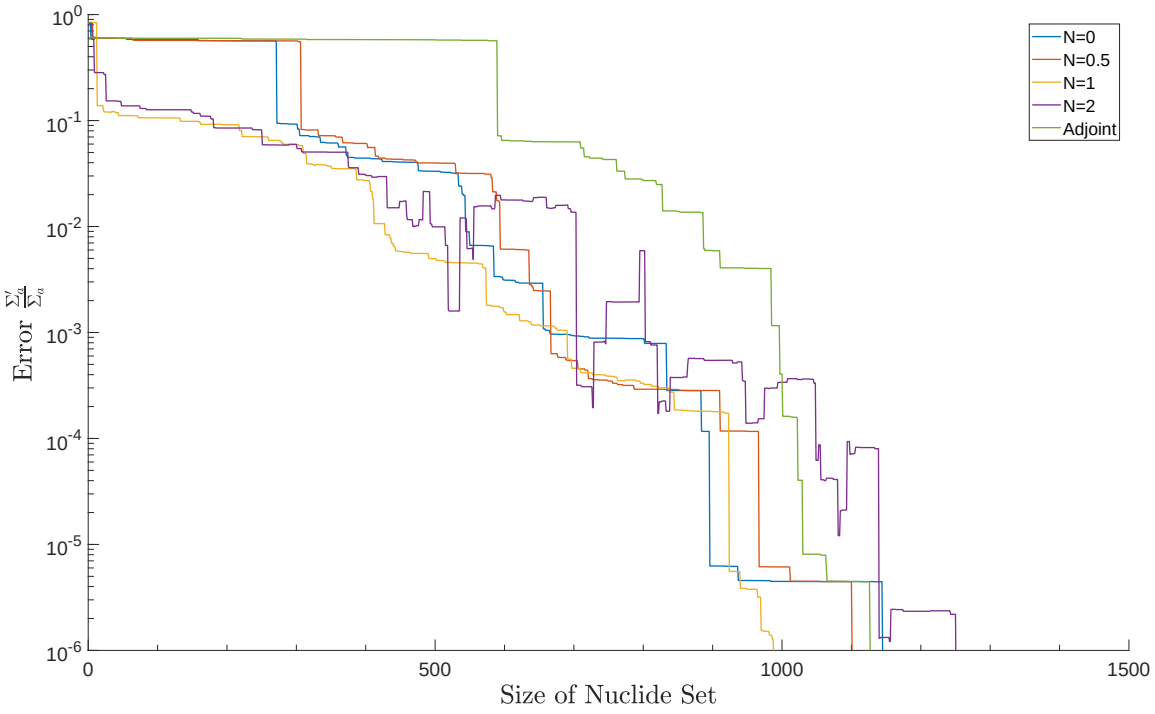



**Figure 5.9:** Relative errors at a given library size reduced with the cutting planes based algorithm and the given weighting method.

## 5.4 PYOMO Model

The integer programming model implemented in PYOMO, described in section 3.3.2, is considered in this dissertation to be the benchmark by which the native system determination methods are compared. The solver used from the module, GNU Linear Programming Kits (GLPK) revised simplex method solver, is far more theoretically studied and developed than the native methods are. The PYOMO module as a whole is also more optimized and stable than the code base that was developed to implement these library reduction methods. However, PYOMO does not currently have the back-end software capability to be coupled with the code base and therefore is limited in its application. It is also the case that the native methods were explicitly developed to solve this problem with full knowledge of physics of the system and expected trends. While the integer programming model was also developed with this information, it cannot fully take advantage of it. Despite this, PYOMO yielded similar results as the native methods.

Figure 5.10 illustrates the error convergence of the PYOMO reductions for the standard weighting methods with the limitation that optimal reduced system was only found at max nuclide set size increments of 50. The trends present in the reduction convergence using PYOMO are very similar to those using the cutting planes native method, considering the lack of size resolution in the figure, though a unique characteristic is that all metrics exhibit the same behavior regardless of the metric dispersion. All of the weighting methods yield similar reductions for threshold errors below 0.1%, with the exception of the adjoint weighting. The adjoint weighting weighting is the only method to exhibit an unstable convergence, but at small nuclide set sizes follows the trend of the other weighting methods. Though the weighting methods are more comparable using PYOMO for system determination than the native methods, it is again the case that the N=1 and N=2 contribution weighting are the optimal weightings. This indicates the integer programming model prioritizes regions of highest weight given that those methods are the best at concentrating the weight with nuclides that are the most important to accurately preserving the contributing transition rates and nuclide number densities. This behavior is similar to the cutting planes method, however, in this case N=1 is slightly better than N=2. This indicates that the integer programming model is more sensitive to the weighting method diffusing the importance and would explain the poor results of the adjoint methods.

With the N=1 contribution weighting being the most optimal weighting for the reduction with PYOMO for all metrics, it is the best to use for the aforementioned benchmarking for the native reduction methods. The weighting and reduction methods chosen for comparison in figure 5.11 are those that were the best combination for each of the subsystem determination methods. Despite the PYOMO reduction being more advanced than the native reduction methods, it is not optimal reduction method for low error thresholds. Surprisingly, the N=2 weighting truncation results in the minimal error for any moderate reduction, being a slight improvement over the N=2 weighting used with the cutting planes method. These two methods result in nearly identical errors for a given nuclide set size, both in magnitude of error and convergence. This is the result of a significant number of the contributing nuclides in the depletion, total activity, and decay heat metric being of distance 2 or less from a source nuclide. The proximity of the contributors results in the cutting planes method

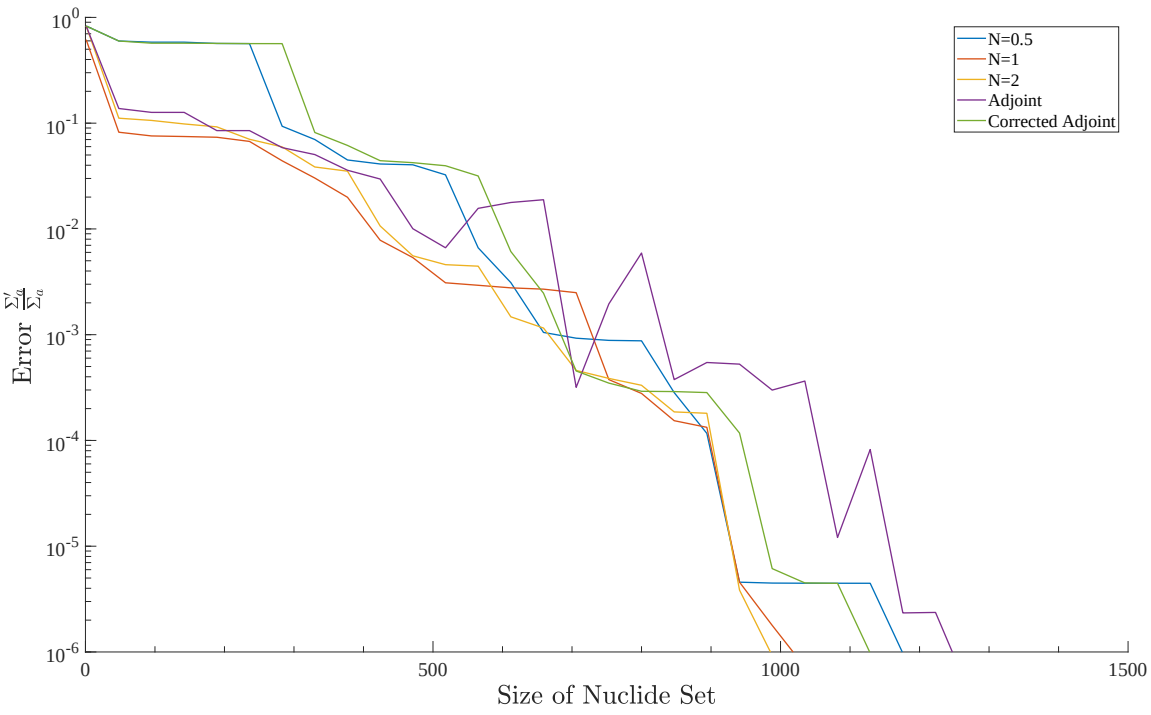



**Figure 5.10:** Relative errors at a given library size reduced based on the PYOMO integer programming model and the given weighting method.

creating many short branches that encompass the same regions of the system preserved in the truncation. However, this should not be taken as an indication that the N=2 weighting truncation is the optimal reduction method, as is explained in chapter 6, but rather that it is possible to account for behavior using either weighting methods or subsystem determination methods and the most appropriate combination is both problem- and metric-dependent.

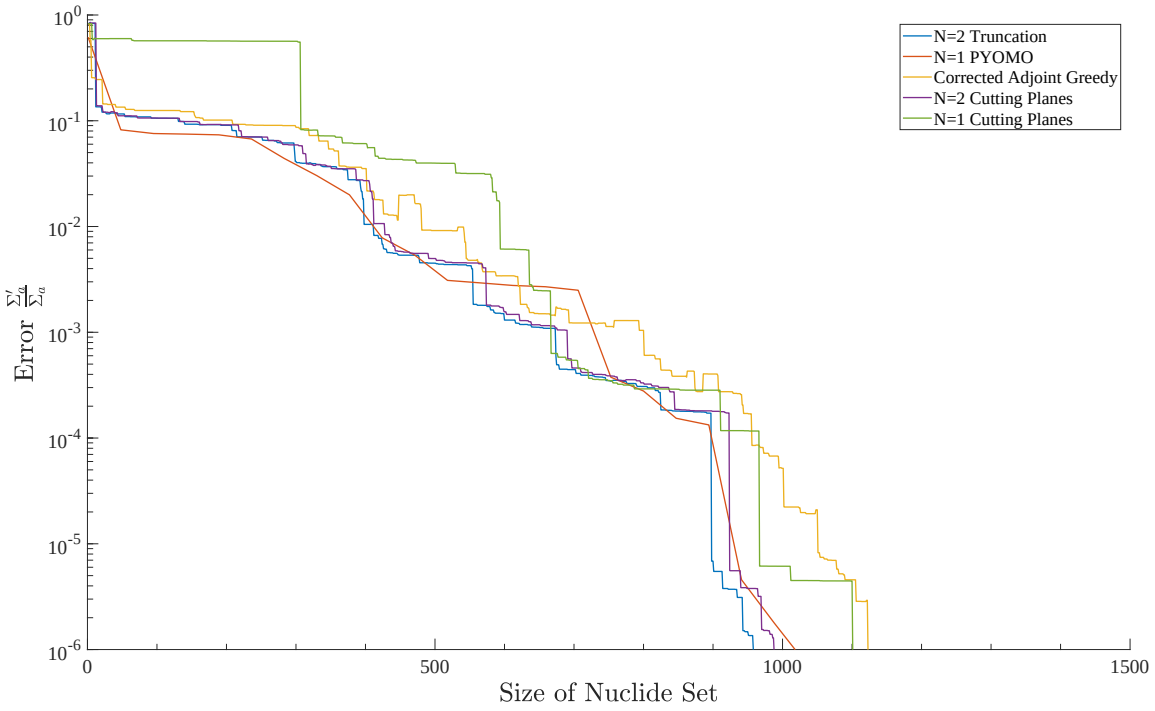



**Figure 5.11:** Relative errors at a given library size reduced with the most suited weighting for each subsystem determination method.

# Chapter 6

# Reduced Library Results

While the previous chapters analyzed the behavior of the components of my method for library reduction, this analysis was concerned with identification of benefits and weaknesses of each of the methods in relation to the complete transition system. The ultimate goal of this research though, is to produce robust, problem-specific reduced libraries. The reduction process and behavior of the reduced libraries will not exactly match the analysis from the previous chapters. This is due to slight change of behavior that will result from the drastic change of the system structure and the inefficiency in the reduction process that is used to generate the data needed in the analysis of the subsystem determination methods. The library reduction is an iterative process that is intended to compensate for the weaknesses that otherwise could not be addressed in either the weighting or subsystem determination methods. However, this process is ultimately reliant on how good the best of the reduction methods are at preserving the behavior observed in chapter 4 and this will determine the successfulness of the development of this method.

## 6.1 Reduction Convergence

The reduction algorithm seeks to find the minimum subgraph of the transition system that meets the accuracy needs of the user, in the fewest iterations possible. Finding this minimum through the process used to create the relative error plots in the previous chapter is impractical, as it requires evaluation after every reduction in the maximum nuclide set

size. As such the size of system that will most closely approach, but not exceed, the error threshold is predicted by the reduction algorithm by either the bisection or Newton's methods as detailed in chapter 4. However, as seen in the relative error plots in chapter 5 the correlation between system size and error has many features that make convergence difficult such as regions of zero slope and local minima. These features are more common in certain combinations of weighting method and subsystem determination method. This should be considered, as well as the peak optimization, when evaluating the combinations of methods as these features can result in slow reduction convergence and convergence to a local minimum rather than the optimal solution.

The three best combinations for reduction optimization using each of the subsystem determination methods — truncation with N=2 contribution weighting, the greedy algorithm with corrected adjoint weighting, and the cutting planes algorithm with N=2 contribution weighting — are also fairly well-behaved for convergence purposes. All three have zero slope regions, but those are unavoidable given that the reduction is discrete and the bisection method can handle those regions, but only the greedy algorithm has local minima that can cause issues for both root finding methods. These local minima though are fairly narrow and shallow and therefore do not pose as much of a problem for this application as they might as others, as evidenced in figure 6.1. As can be seen in the figure, and noting the system sizes at each iteration, all three methods have a stable convergence after the initial iterations and require no more than 15 iterations to converge on a final reduced system. Also, most of the reduction in the nuclide set for all combinations occurs in the first 5 iterations, being within 100 nuclides of the final set. The only issue of note is that though the methods converge very rapidly to the neighborhood of the solution they can stagnate there, only changing the size of the system by 5 or fewer nuclides per iteration. This is another effect of the step-like nature of the relative error from removal. If the the optimal solution is near one of these steps, with the step being near the boundary of the known viable region for that iteration, the linear approximation from Newton's method will result in these very minor reductions. Unfortunately, unlike local minima or zero slope regions that can be detected and can be bypassed using bisection, this effect is very difficult to detect during the reduction. Considering the strongly nonlinear behavior of error-system size relationship, the rate and

stability of convergence has exceeded initial expectations and is more than sufficient for the purposes of the reductions, so long as they remain a up-front computational cost and not adapted to an on-the-fly reduction. Though all three methods perform comparatively, the cutting planes algorithm does exhibit the fastest convergence. By the 10th iteration at all error thresholds, the method was either converged or had restricted the viable size to within 10.

It is possible to improve the convergence of these methods further with the inclusion of subsystem correction steps to the iterative process or a subsystem estimation preprocessing step. These methods, described in section 3.5, can either improve the behavior of the convergence or reduce the maximum size of the subsystem and leading to the most important reduction estimate – the initial one – being more accurate. Of these methods, one is already applied to the adjoint weighting method to calculate the correct adjoint weights, and another is handled implicitly for the sample problem by none of the duplicate nuclides in the activation/light nuclide sub-library being a source material. Of the final two methods, both were analyzed in the context of the depletion metric for the sample problem. The elimination of short-lived fission products requires a relatively long half-life threshold to be of value to this problem type, though that is not likely the case for total activity and decay heat problem types in which the short-lived fission products can overwhelm the nuclides that may be of interest to the user. The standard Origen library has only 86 nuclides with a half-life of less than 0.1 seconds with the shortest being 0.001 seconds. However, if the minimum half-life is increased to approximately 1 minute, 543 nuclides can be eliminated. This half-life is long enough though that it effects the delayed neutron precursors that are important for reactor kinetics and change the rate of production for the cumulative fission products — most notably Xe-135 — as the calculation for the cumulative yield considers the branching ratios of the direct products loss channels, but not the average transition time. The last of the methods is the competition correction method. This method reduces the relative error in the system by reintroducing nuclides that represent paths that are parallel to contributing paths, but do not contribute themselves. These losses that "compete" with the weighted paths can represent significant portions of the parent nuclides loss rate in the standard library and by reintroducing them offer large corrections to the contributing

transition rate. This correction is done after reduction steps that exceed the threshold error, but not significantly. This would be reduction steps when the viable range has already been narrowed to 100 or fewer nuclides. Five methods to determine which competing loss channels are significant were explored:

1. Any nuclides largest loss channel that is not already included in the system
2. Any loss channels that constitute greater than 10% of a nuclides total loss rate
3. Any loss channels that are have a higher transition rate than the highest loss channel already to be preserved
4. All loss channels for all nuclides to be preserved
5. All of the largest loss channels that are required to total to a majority (>50%) of a nuclides total loss rate.

To compare the effectiveness of these ways to classify significant loss channels two reduction cases were considered, one thought to have the most to gain from this method and the other, the least to gain. The first case is a reduction done by truncation using the adjoint weights for the depletion metric, and the second is a reduction done with the greedy algorithm using N=1 contribution weights for the total activity. The first case is a metric with concentrated contributions and using methods that should establish creation paths, but no concern for losses and therefore a method that corrects this should yield a large increase in accuracy. The second case is a very diffuse metric with a weighting scheme that gives importance to losses and a method that ensures connectivity and so all significant loss paths should already be included. To ensure that an equal amount of paths and nuclides can be potentially reintroduced, both cases were reduced to 450 nuclides, resulting in errors of 3.03% and 4.84% respectively, with table 6.1 summarizing the results.

The second method for determining important loss channels proved to be the best method in both the "best" and "worst" case for having improvements from this method. This method not only causes little expansion to the system, considering the goal of the main method is to make reduce the system, but also results in a significant reduction in the error, reducing

**Table 6.1:** Comparison of the effectiveness of the Competition-Correction methods

| | **Nuclides Added (Case-1)** | **Improvement per Nuclide (Case-1)** | **Nuclides Added (Case-2)** | **Improvement per Nuclide (Case-2)** |
|---|---|---|---|---|
| 1 | 116 | $5.76e^{-5}$ | 61 | $5.66e^{-4}$ |
| 2 | 61 | $2.92e^{-4}$ | 27 | $5.47e^{-3}$ |
| 3 | 78 | $1.64e^{-4}$ | 33 | $5.18e^{-3}$ |
| 4 | 736 | $4.00e^{-5}$ | 715 | $5.50e^{-5}$ |
| 5 | 44 | $2.46e^{-4}$ | 30 | $5.33e^{-3}$ |

it by a factor of 3 and 5 respectively to 1.2% and 0.86%. These improvements are far above the error reduction from removing the same number of nuclides fewer, which as shown in figures 5.1 and C.2.1 would be expected to be about half as effective. Though, this method was not included in the reductions of the libraries to preserve the behavior and characteristic inherent to the core methods, it is a valuable tool for optimizing the reduction.

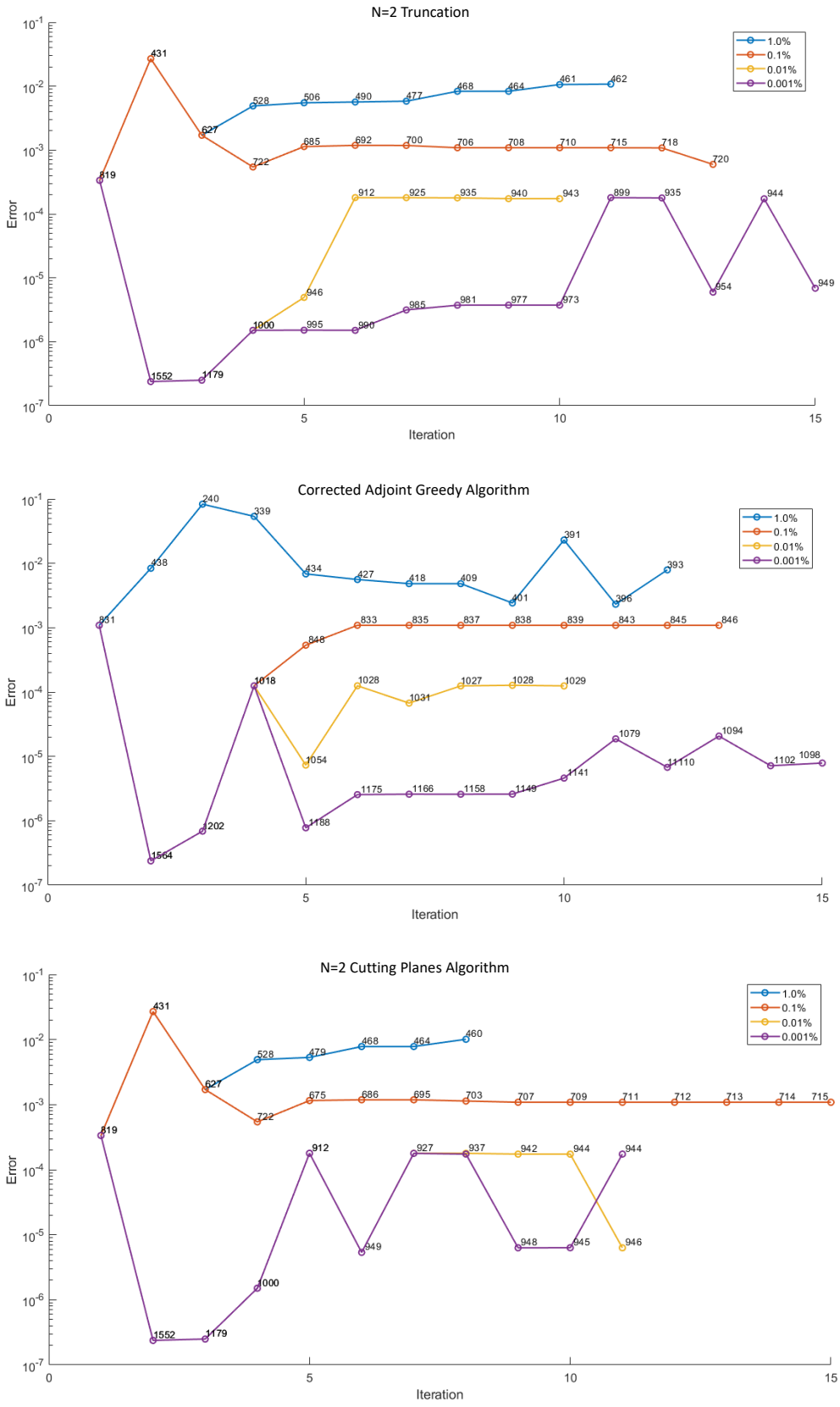



**Figure 6.1:** Convergence of relative error for the three best combinations of weightings and subsystem determination methods.

## 6.2 Reduced Library Robustness

The robustness of the reduced library is paramount to the practicality of my reduction method. The method would not be considered useful if the range of problems that the reduced library can be applied to, and retain the accuracy prescribed in the original reduction for, is strictly limited by the reducing problem. While most of the conditions that are variable to the standard Origen libraries were shown in section 4.3 to not impact the distribution of the metric contributions, several variables did have a significant effect. The variables that did not change the distribution would result in the same weighting and reduction order independent of changes to those variables. However, the final burnup and initial enrichment were shown to have significant impact on the weight distribution and are therefore areas of concern for maintaining accuracy. Also, despite not resulting in a significant change in weight distribution, differences in the level of accuracy for Origen depletion cases are so ubiquitous that it has been deemed a special case for further study. The results that the reduced libraries give for problems that have conditions of these variables that depart from the reducing problem is used as the measure of robustness. The reduced library used to determine the robustness of the method are the reduced libraries for the depletion metric and the sample reducing problem, created by the three best combinations for reduction optimization described at the start of this chapter. The threshold error used in these reductions was 0.01% to maximize the differences in the different methods nuclide set.

**Table 6.2:** Accuracy of reduced libraries for varying final burnups, relative to the Origen standard library.

| **Burnup** ($\frac{\mathbf{GWd}}{\mathbf{MTU}}$) | **N=2, Truncation** | **Corrected Adjoint Greedy Alg.** | **N=2, Cutting Planes Alg.** |
|---|---|---|---|
| 20 | 0.999632 | 0.999711 | 0.999704 |
| 30 | 0.999591 | 0.999686 | 0.999690 |
| 40 | 0.999507 | 0.999603 | 0.999605 |
| 50 | 0.999419 | 0.999493 | 0.999493 |
| 60 | 0.999384 | 0.999402 | 0.999416 |

The reduced libraries performed within the threshold error in the metric for all conditions tested for all three reducing methods. The results of these tests are summarized in tables 6.2 and 6.3 for the standard time and power resolution. The tests were repeated using the highly detailed power history from section 4.3.3 but there was no systematic increase in errors for any of the reducing methods and the accuracies in all cases were within 0.00001 of those in which the standard power history structure was used. This confirms the pre-reduction analysis of the importance of the variable. Surprisingly, though, the reduced libraries were more accurate for lower burnups than they were for the burnup in the reducing problem. This is counter to the analysis of the weight distribution, as it was expected that the accuracy would decrease as the burnup difference between the problem and the reducing problems increased. Upon analysis of the reduced libraries structures (Appendix A) and the distribution of the low burnup metric, this is likely due to the higher concentration of the weights in the starting materials and their nearest daughter products. These nuclides are also the most strongly preserved nuclides regardless of the the reducing problem, metric, or method as they are required as predecessors to all other nuclides in the system. An unexpected trend is also seen in the accuracy of the initial enrichments, and that is that the reduced libraries are more accurate for the higher initial enrichment than for the reduced enrichment. After similar analysis that was done to identify the cause of the trend in burnup, I have determined that the likely cause of this is that the higher the initial enrichment the more fission dominated the problem becomes – as opposed to being dominated by transmutations in the actinides. Also, similar to the burnup trend, this trend is the result of the reducing methods placing an emphasis on the fission products and fission transitions from U-235.

Out of the three methods, the greedy algorithm appears to produce the most robust libraries. However, given that the optimized inventory for this library is 80 nuclides larger than the other 2 this is not entirely unexpected. Though the reductions do end with different library sizes, the accuracy of the reduced libraries does not seem to be heavily dependent of the definition of the reducing problem, as the metric is within the threshold errors for a large range of conditions departing from the reducing problem. However, the reduction results in a negative bias in the results, as the reduced libraries calculation of the metric was below that

**Table 6.3:** Accuracy of reduced libraries for varying initial enrichments, relative to the Origen standard library.

| Enrichment (% mass U-235) | N=2, Truncation | Corrected Adjoint Greedy Alg. | N=2, Cutting Planes Alg. |
|---|---|---|---|
| 2% | 0.999348 | 0.999662 | 0.999355 |
| 3% | 0.999402 | 0.999684 | 0.999406 |
| 4% | 0.999456 | 0.999700 | 0.999453 |
| 5% | 0.999506 | 0.999712 | 0.999503 |
| 6% | 0.999550 | 0.999719 | 0.999548 |

of the original library for every test problem. This bias is the result of the unavoidable loss of minor contributing nuclides. There is also the bias that the contributing nuclides that remain in the reduced system will increase in contribution as alternative paths are removed, their number densities will increase. The magnitude of this effect on a given nuclide is dependent of the reduced transition systems structure and does result in significantly different ordering of shared nuclides contributions (fig. 6.2). This effect is most minimized through the cutting planes reduction method, as it is more efficient in structuring the reduced system. This can be seen in a comparison of the reduced transition systems between figures A.2 and A.4 in appendix A. These two reductions use the same weighting method and result in the same sized nuclide set (fig. 6.2), with the same metric redistribution, however, the cutting planes reduced system has 200 more transitions maintained. These transitions result in more accurate transition rates between the actinide region and the fission product region, normalizing the mass flows in both.

## 6.3 Computational Efficiency

The reduction methods have proven to successful at preserving the underlying physics of the full libraries for the problem types, and not just the reducing problem. This is important as the benefits of the reduced libraries is dependent on the libraries being able to be used many times, as the reduction process is expensive compared to the source term generation. If multiple reduced libraries were required to cover the same solution space as the original

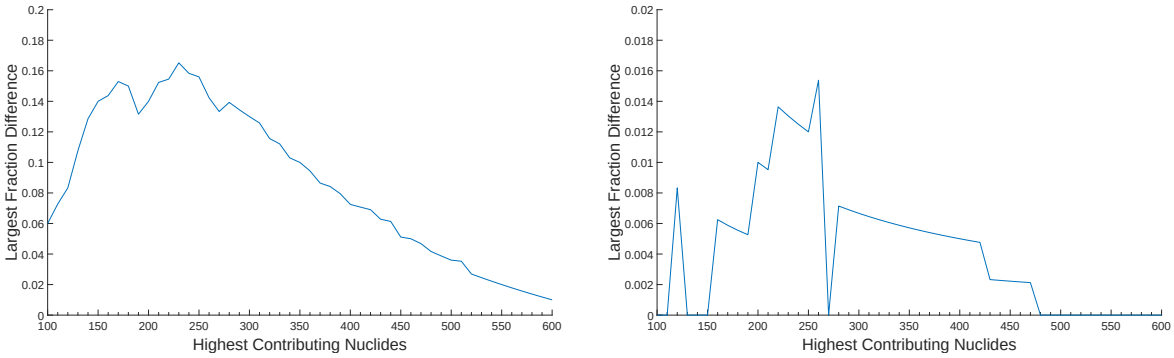



**Figure 6.2:** Graph of the difference in highest contributing nuclides after reduction by truncation vs. the greedy algorithm (left) and the truncation vs. the cutting planes algorithm (right).

**Table 6.4:** Computational efficiency gains from N=2 cutting planes reduction for the depletion metric.

| Threshold Error | Actual Error | Memory Size (kB) | Nuclide Set Size | Number of Transitions | Run Time (ms) |
|---|---|---|---|---|---|
| 0 | 0 | 4869 | 2304 | 60969 | 78 |
| $10^{-5}$ | $6.30 \cdot 10^{-6}$ | 2382 | 945 | 30388 | 30 |
| $10^{-4}$ | $6.30 \cdot 10^{-6}$ | 2382 | 945 | 30388 | 30 |
| $10^{-3}$ | $5.43 \cdot 10^{-4}$ | 1702 | 722 | 21553 | 22 |
| $10^{-2}$ | $7.81 \cdot 10^{-3}$ | 959 | 464 | 11914 | 13 |

library, besides being cumbersome for the code framework implementing it, would also diminish any reductions in on-disk memory requirements. Thanks to the proven robustness of the methods, though, this is not the case and the increased computational efficiency of the reduced libraries are summarized in table 6.4.

The reduction methods used will have an impact on the optimization, as made evident in the prior sections. However, the reduction in run time and memory requirement is directly proportional to the reduction in the number of transitions in the system and thus the relationship established in table 6.4 is valid for all reductions. This is because the number of transitions is equivalent to the number of non-zero elements in the transition matrix. Due to the sparsity of the transition matrix, matrix storage and operations do not use standard matrix representations but rather a collapsed vector form of the matrix. This representation means the number of operations and amount of information store is relative

to the number non-zero elements rather than the the square of the number of nuclides as it would be in the standard representation. Though there is a relationship between the nuclide set size and the number of transition, being roughly 1:30, this is not a strict relationship with the ratio being much higher for fissionable nuclides and much lower for activation/light nuclides. Furthermore, the relationship between the error threshold and the reduction in computational costs is also not proportional. The most significant reduction in cost comes from even a very strict error tolerance of 1 pcm, which results in a factor of two speed up and reduction in memory size. Increasing the allowed error up to three orders larger only results in another factor of two improvement over the more strict reduction. This is the result of the initial, now proven, assumption that most nuclides do not contribute to a given problem; because these nuclides don't contribute at all they can be removed with no increase in error. The number of non-contributing nuclides will vary by metric and problem definition and therefore, so will too the difference in gains from a strict tolerance as compared to less restrictive error limits.

# Chapter 7

# Conclusions

The primary goal of this research was to develop an automated library reduction method that would be able to generate robust problem-specific libraries. The need for this method being derived from current code development efforts to increase source term accuracy via the coupling of micro-depletion codes like Origen, as well the current state of depletion library reduction techniques being very time consuming or very specific to the problem that they are being applied to. The method developed achieves this through a novel approach based on the graph representation of the transition matrix to preserve the physically important characteristics of the source generation. This combines the available weighting methods and methods to determine optimal system structure. The procedure for nuclide removal and the basic methods for creating a reduced library that preserves problem specific quantities was already proven possible[27]. However, to create the more advanced methods and determine the best applications for the various weighting and system methods theorized, extensive analysis of the problem that was to be solved was needed. This problem consisted of how the importance of nuclide should be measured, as well as how best to determine the important aspect of the system structure given the difference in weighting methods that these methods would be reliant on for optimization. This analysis revealed that not only are the problem-contributing nuclides dependent on the problem type, but so too are the optimal combinations of methods as the behavior of the different problem types was significantly different depending on two factors. These two factors are whether the weighting factor for the metric flux-dependent, and whether the contributions are dispersed or concentrated.

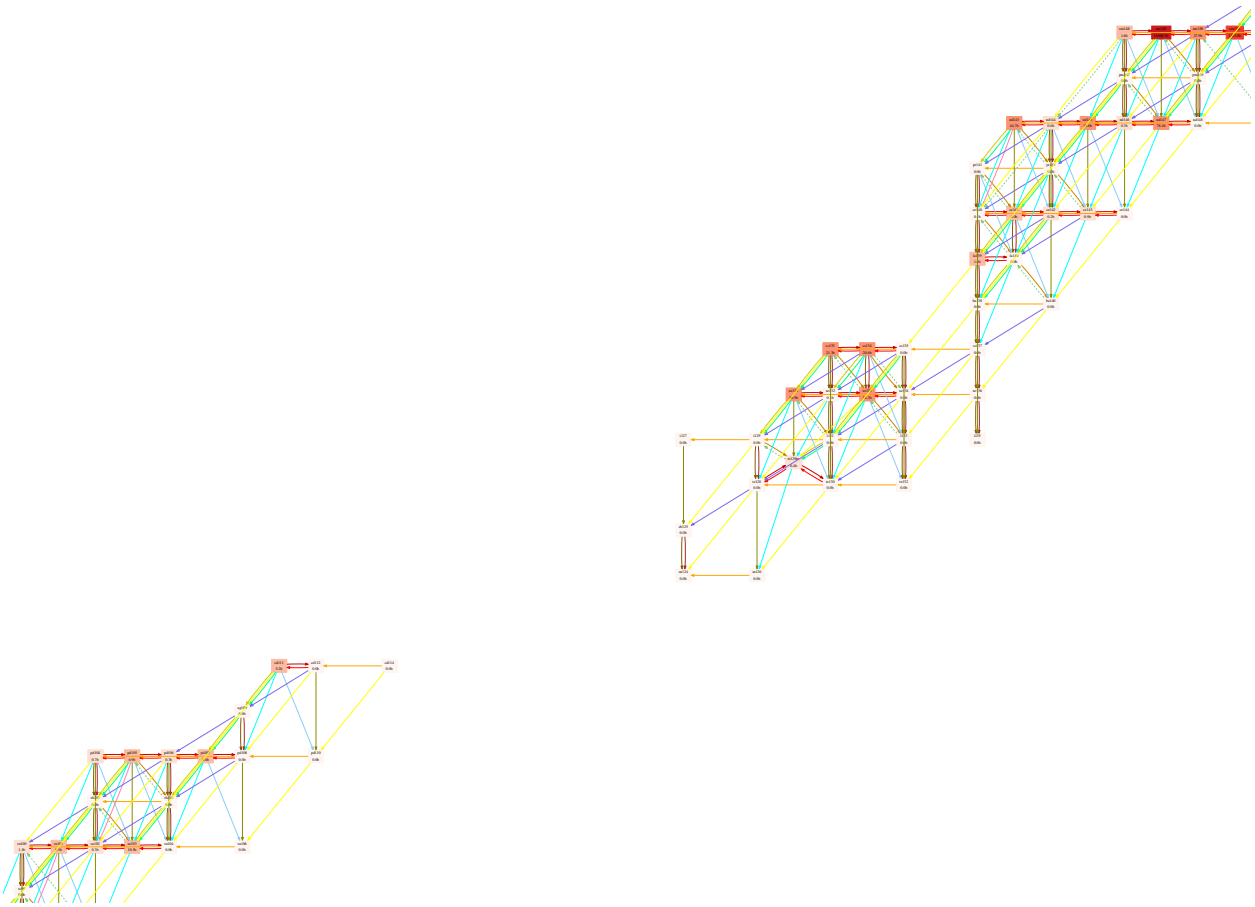


**Figure 7.1:** Partial reduced-transition system using direct contribution weighting and system truncation to an error of 1%.

These factors result in the contribution weighting of degree two being effective at low error tolerances for all problem-types and removal methods, but at greater level of inventory reduction the corrected adjoint weighting with the greedy reduction method, or contribution weighting of degree one and the cutting planes reduction method provide better optimization for the total activity problem and the gamma source problem respectively. The simple methods for library reduction, besides being inefficient and not generally finding the optimal transition system, do not consider these factors. The direct contribution weighting scheme and removal by truncation, can find a significantly reduced system that has the same results (within the allowable error) as the full library for a given problem, with one example of a system reduced in this fashion being figure 7.1. However, using this unsophisticated approach has the same shortcomings as many of the other approaches currently being used - shortcomings that the more advanced reduction methods address and overcome.

The simple reduction scheme, as well as the library reduction methods currently being used, cannot make generic libraries. In other words, the simple reduction scheme and the matrix based reduction methods create libraries which may be accurate for the exact problem – and also similar problems – for which they are being reduced. However, by noting the invariance of the contributing nuclides for these problem variables like burnup, initial enrichment, irradiation history, and even reactor type, the methods were developed to target and preserve these contributions; they consider all physics but do not give any special (or hard-coded) weight to a single factor in the problem. The resulting reduced libraries have proven to be within error tolerances for problems that significantly depart from the reducing problem. By preserving meaning and physics, rather than the approximating the observed behavior, the end sub-transition system is much more likely to be accurate. This is important since the full behavior is an attribute of the system not just the individual nuclides or their immediate transitions.

The reduction methods, implemented in an iterative framework, reliably converge to an optimal reduced system. The convergence is very rapid, converging in fewer than 15 iterations in all cases tested. The iterative method used, being a hybrid bisection and Newton's method, is stable even with the troublesome behavior exhibited by the error increase relative to the library size. These two factors are achieved through identifying the similarities of the relative error change and using the bisection method only when necessary to circumvent the convergence issues that the behavior would cause for Newton's method. The rate of convergence of this hybrid method also appears to be independent of the threshold error, but it is dependent on the reducing methods as they determine the error behavior with some being more well behaved than others. For both the optimization and convergence the contribution weighting of degree two and the cutting planes algorithm proved to be the most well suited for the depletion metric type.

The reduced libraries accomplish the main goal of this research, and that is to reduce the computational cost of micro-depletion. The reduced libraries that were restricted to 1% error relative to the standard library that it was reduced for showed a factor of 5 reduction in on disk size, and a 6-fold reduction in runtime. When compared to the results of the preliminary work, that resulted in a final runtime of 10ms and inventory of 255 nuclides and

5035 transitions, it can be expected that the automated methods presented here can achieve similar results if given the 3%-10% error threshold that those efforts allowed for. This is backed by the trend in reduction size relative to error and transition system size relative to run time. However, the automated methods can achieve this reduction in a matter of minutes and result in a robust final library making them better suited then the manual method.

# Chapter 8

# Future Work

Potential future developments for this method include both basic additions to the method, as well exploring major structural changes to the method. Basic additions which would be most useful are the addition of other problem and source types. These source types would include a radiation energy metric as well as an absorption rate metric. A more significant effort could be applied to exploring the impact of determining importance based on transitions, rather than nuclides as is done in the method currently. This could potentially result in a better optimization of the reduction of the system. Also, a study should be done on the applicability of these methods to reactor types that are not included in the Origen standard libraries, most notably fast reactors.

# Bibliography

[1] Bang, Y. and Abdel-Khalik, H. S. (2013). Hybrid reduced order modeling algorithms for reactor physics calculations. *Transaction of the American Nuclear Society*, 109:1385–1388. 15

[2] Bang, Y., Abdel-Khalik, H. S., Jesse, M. A., and Mertyurek, U. (2015). Hybrid reduced order modeling for assembly calculations. *Nuclear Engineering and Design*, 295:661–666. 15

[3] Benkharfia, H., Zidi, T., and Belgaid, M. (2016). Lumped psuedo fission products during burnup step in mcnp5-origen coupling system. *Progress in Nuclear Engergy*, 88:277–284. 14

[4] Chiba, G., Tsuji, M., Narabayashi, T., Ohoka, Y., and Ushio, T. (2015). Important fission product nuclides identification method for simplified burnup chain construction. *Nuclear Science and Technology*, 52:953–960. 46

[5] Collins, P. E., Luciano, N., and Maldonado, G. I. (2014). Modernization and expansion of isotopic depletion capabilities within the nestle 3d nodal simulator. *Transactions of the American Nuclear Society*, 111:1230–1233. 3

[6] Duderstadt, J. J. and Hamilton, L. J. (1976). *Nuclear Reactor Analysis*. John Wiley and Sons. 57

[7] Duluc, M., Caplin, G., and Haeck, W. (2013). Evidence of a criticality accident occurring with spent fuels: Basic considerations and lessons learned from the fukushima accident. xvii, 139

[8] EIA (2015). *U.S. Energy Information Administration Independent Statistics and Analysis*. https://www.eia.gov/nuclear/spent_fuel/ussnftab3.php. 78

[9] Favalli, A., Vo, D., Grogan, B., Jansson, P., Liljenfeldt, H., Mozin, V., Schwalbach, P., Sjland, A., Tobin, S. K., Trellue, H., and Vaccaro, S. (2016). Determining initial enrichment, burnup, and cooling time of pressurized-water-reactor spent fuel assemblies by analyzing passive gamma spectra measured at the clab interim-fuel storage facility in sweden. *Nuclear Instruments and Methods in Physics Research*, 820:102–111. 3

[10] Gandini, A. (1975). Time-dependent generalize perturbation methods for burn-up analysis. *Comitato Nazionale Energia N*, RT/FI(75)4. 46

[11] Gauld, I. C., Ilas, G., Murphy, B. D., , and Weber, C. F. (2010a). Validation of scale 5 decay heat prediction for lwr spent nuclear fuel. NUREG CR-6972, U.S.NRC. 10

[12] Gauld, I. C., Ilas, G., and Murphy, B. D. (2010b). Analysis of experimental data for high burnup pwr spent fuel isotopic validation–ariane and rebus programs (uo2 fuel). NUREG CR-6969, U.S.NRC. 10

[13] Griesheimer, D. P. and Carpenter, D. C. (2016). Techniques for practical monte carlo reactor depletion calculations. In *PHYSOR*, pages 477–493. 3

[14] Hart, W. E., Laird, C., Watson, J.-P., and Woodruff, D. L. (2012). *Pyomo - Optimization Modeling in Python*. Springer, Alburque, NM. 38

[15] Huang, K., Wu, H., Li, Y., and Cao, L. (2016). Generalized depletion chain simplification based on significance analysis. In *PHYSOR*, pages 108–117. 42

[16] Huff, K., Gidden, M., Carlsen, R., Flanagan, R., McGarry, M., Opotowsky, A., Schneider, E., Scopatz, A., and Wilson, P. (2016). Fundamental concepts in the cyclus nuclear fuel cycle simulation framework. *Advances in Engineering Software*, 94:46–59. 3

[17] Ilas, G., Gauld, I. C., Difilippo, F. C., and Emmett, M. B. (2010). Analysis of experimental data for high burnup pwr spent fuel isotopic validation–calvert cliffs, takahama, and three mile island reactors. NUREG CR-6968, U.S.NRC. 10

[18] Ilas, G., Gauld, I. C., and Liljenfeldt, H. (2014). Validation of origen for lwr used for lwr fuel decay heat analysis with scale. *Nuclear Engineering and Design*, 273:58–67. 10

[19] Ilas, G., Gauld, I. C., and Radulescu, G. (2012). Validation of new depletion capabilities and endf/b-vii data libraries in scale. *Annals of Nuclear Engergy*, 46:43–55. 10

[20] Isotalo, A. E. and Wieselquist, W. A. (2015). A method for including external feed in depletion calculations with cram and implemenation into origen. *Annals of Nuclear Engergy*, 85:68–77. 9

[21] Kajihara, T., Tsuji, M., Chiba, G., Kawamoto, Y., Ohoka, Y., and Ushio, T. (2016). Automatic construction of a simplified burn-up chain model by the singular value decomposition. *Annals of Nuclear Engergy*, 94:742–749. 15

[22] Kasselmann, S., Schitthelm, O., Tantillo, F., Scholthaus, S., Rossel, C., and Allelein, H.-J. (2016). Development of a new nuclide generation and depletion code using a topological solver based on graph theory. *Nuclear Engineering and Design*, 306:154–159. 16

[23] Kochunas, B., Jabaay, D., Stimpson, S., Graham, A., , and Downar, T. (2015). Vera core simulator methodology for pwr cycle depletion. CASL-U 0155-000, U.S. Department of Energy. viii, 3, 12

[24] Leniau, B., Mouginot, B., and Thiolliere, N. (2016). Generation of sfr physics models for the nuclear fuel cycle code class. In *PHYSOR*, pages 1120–1131. 4

[25] Leniau, B., Mouginot, B., Thiolliere, N., Doligez, X., Bidaud, A., Courtin, F., Ernoult, M., and David, S. (2015). A neural network approach for burn-up calculation and its application to the dynamic fuel cycle code class. *Annals of Nuclear Energy*, 81:125–133. 3, 4, 15

[26] Peterson, J., Sunny, E., Wieselquist, W., and Worrall, A. (2016). Generating cross sections for orion fuel cycle models. In *PHYSOR*, pages 2207–2217. 3

[27] Richards, S. M. and Skutnik, S. E. (2017). Problem-dependent origen library compression to increase computational efficiency. In *International Conference on Mathematics & Computational Methods Applied to Nuclear Science & Engineering*. 123

[28] Sanders, C. E. and Gauld, I. C. (2001). Isotopic analysis of high-burnup pwr spent fuel samples from the takahama-3 reactor. NUREG CR-6798, U.S.NRC. 10

[29] Scaglione, J. M., Peterson, J. L., Banerjee, K., Robb, K. R., and LeFebvre, R. A. (2014). Integrated data and analysis system for commercial used nuclear fuel safety assessments. In *Waste Management Conference*. 3

[30] Skutnik, S., Littel, J., and Sly, N. (2016). Cyborg: An origen-based reactor analysis capability for cyclus. *Transactions of the American Nuclear Society*, 115:299–301. 3

[31] Sublet, J.-C., Fleming, M., and Gilbert, M. R. (2016). Fispact-ii: An advanced simulation platform for inventory and nuclear observables. In *PHYSOR*, pages 130–139. 3

[32] Wieselquist, W., Hart, S., Isotalo1, A., Havlj, F., Skutnik, S., Lefebvre, R., Gauld, I., Wiarda, D., Lefebvre, J., Hu, G., Sly, N., and Lago, D. (2016). Origen: Neutron activation, actinide transmutation, fission product generation, and radiation source term calculation. Training Manual 39, Oak Ridge National Laboratory. 5, 7, 9, 11, 12, 18, 82

[33] Wieselquist, W. and Kim, K. S. (2014). Simplified burnup chain development. CASL PHI Ticket 3432, Oak Ridge National Laboratory. 4, 18

[34] Wieselquist, W. A. (2015). The scale 6.2 origen api for high performance depletion. CASL-U 0165-000, U.S. Department of Energy. 6, 8

[35] Wigeland, R., Taiwo, T., Ludewig, H., Todosow, M., Halsey, W., Gehin, J., Jubin, R., Buelt, J., Stockinger, S., Jenni, K., and Oakley, B. (2014). Nuclear fuel cycle evaluation and screening - final report. D.O.E. Report FCRD-FCO-2014-000106, Fuel Cycle Research and Development. 12

[36] Williams, M. L. (1979). Development of depletion perturbation theory for coupled neutron/nuclide fields. *Nuclear Science and Engineering*, 70:20. 10, 46

[37] Williams, M. L. (1991). Generalized contributon response theory. *Nuclear Science and Engineering*, 108:355–383. 46

[38] Williams, M. L. and Ilas, G. (2009). Endf/b-vii nuclear data libraries for scale 6. In *Advances in Fuel Management Conference*. 10

[39] Xu, Y., Kim, T. K., and Downar, T. J. (2010). Depletor - a depletion code for parcs. PU NE-00-28, Purdue University. 3

[40] Zhang, S. and Abdel-Khalik, H. (2016). The reducibility of reactor physics depletion and decay calculation. *Reactor Analysis Methods*, 4:1293–1296. 14, 16

# Appendices

# A Transition Systems

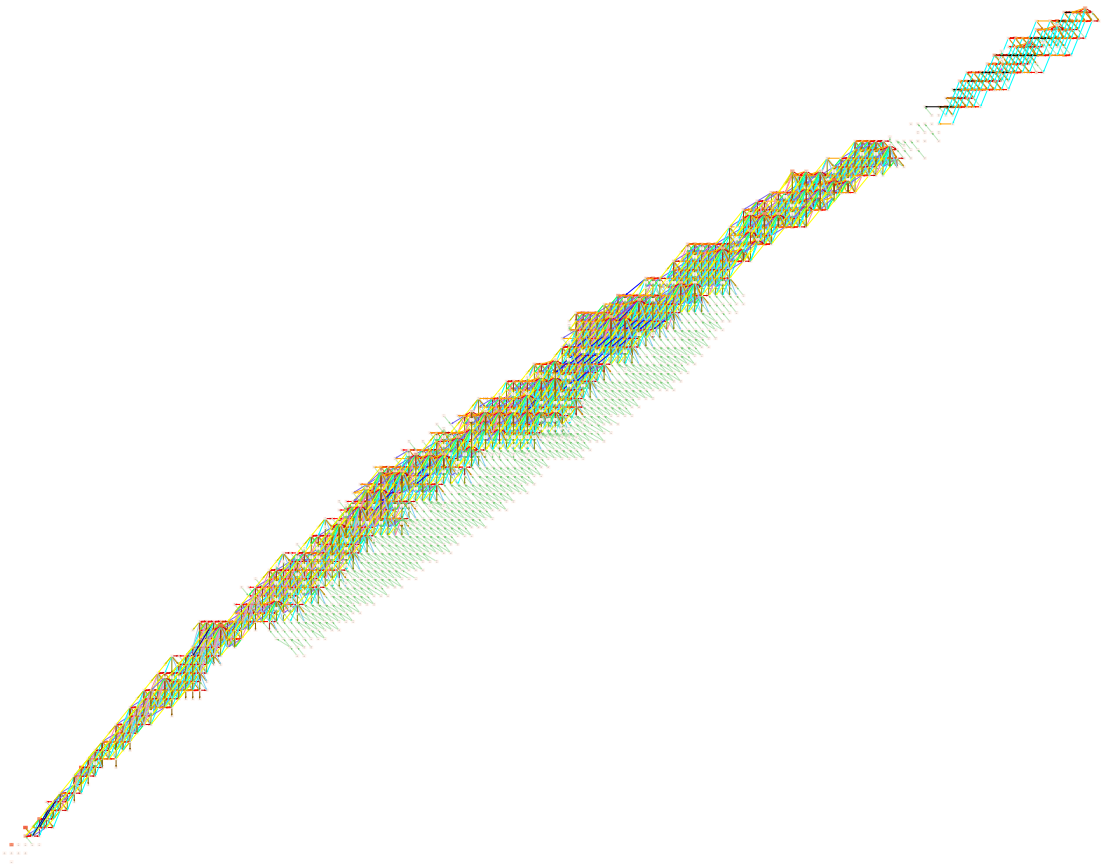


**Figure A.1:** Transition system for Westinghouse 15x15 PWR library with 5% initial enrichment (fission and byproduct transition are not illustrated).



**Figure A.2:** Transition system for Westinghouse 15x15 PWR library with 5% initial enrichment reduced by N=2 contribution weighting truncation with a 0.01% threshold error (fission and byproduct transition are not illustrated).

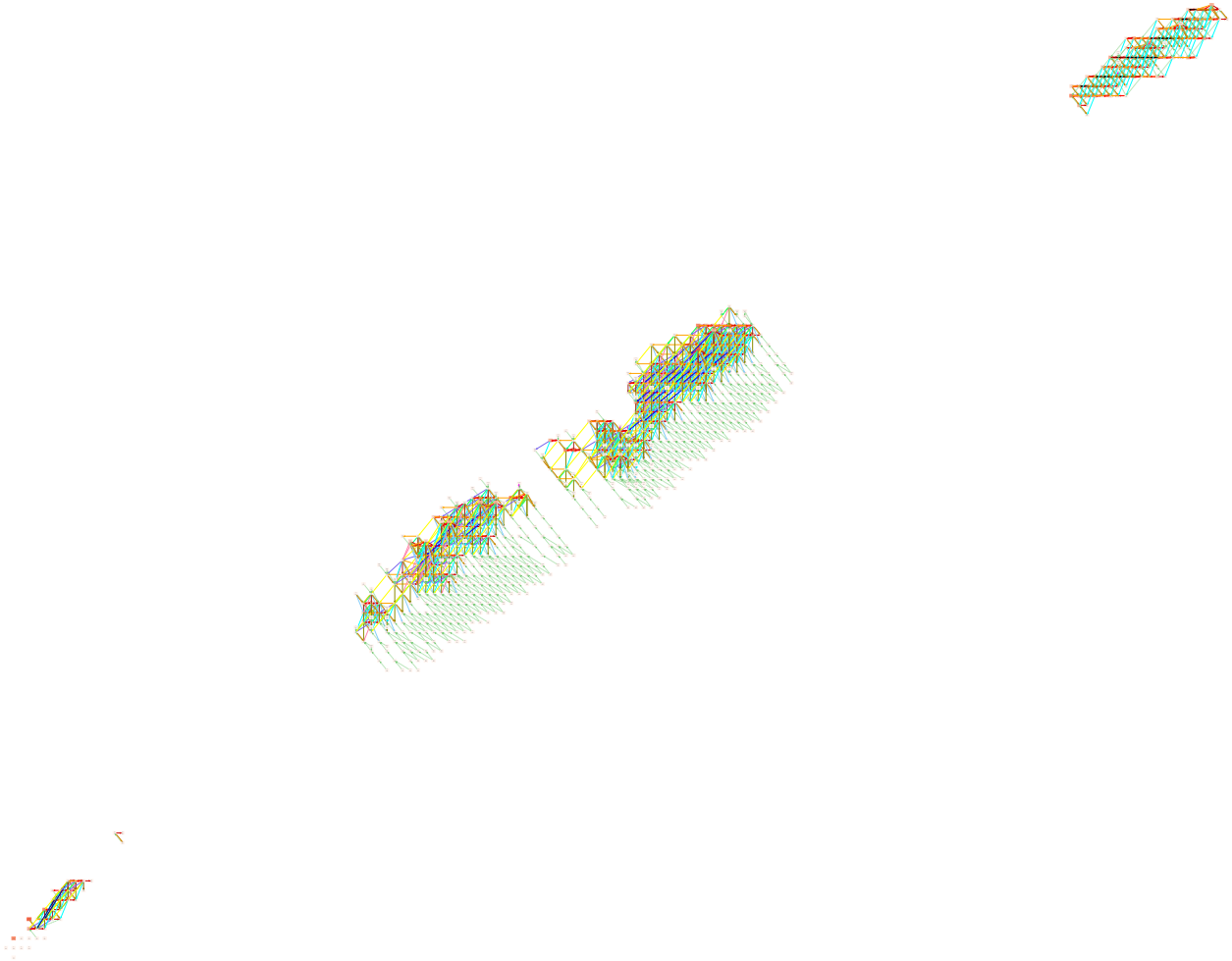


**Figure A.3:** Transition system for Westinghouse 15x15 PWR library with 5% initial enrichment reduced by corrected adjoint weighting and the greedy algorithm with a 0.01% threshold error (fission and byproduct transition are not illustrated).



**Figure A.4:** Transition system for Westinghouse 15x15 PWR library with 5% initial enrichment reduced by N=2 contribution weighting and the cutting planes algorithm with a 0.01% threshold error (fission and byproduct transition are not illustrated).

# B Fission Yields

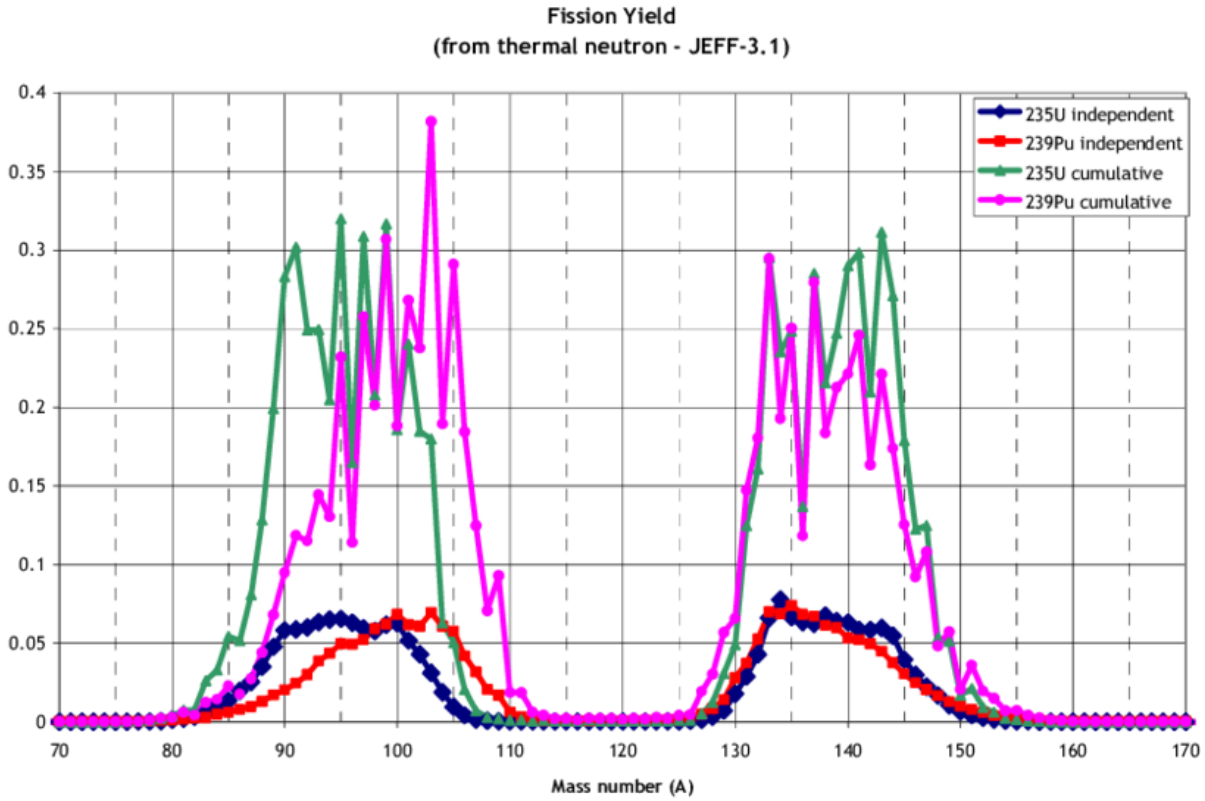



**Figure B.1:** Fission yield curves for fission of U-235 and Pu-239. [7]

# C System Determination Method Convergence

## C.1 Threshold Truncation Method

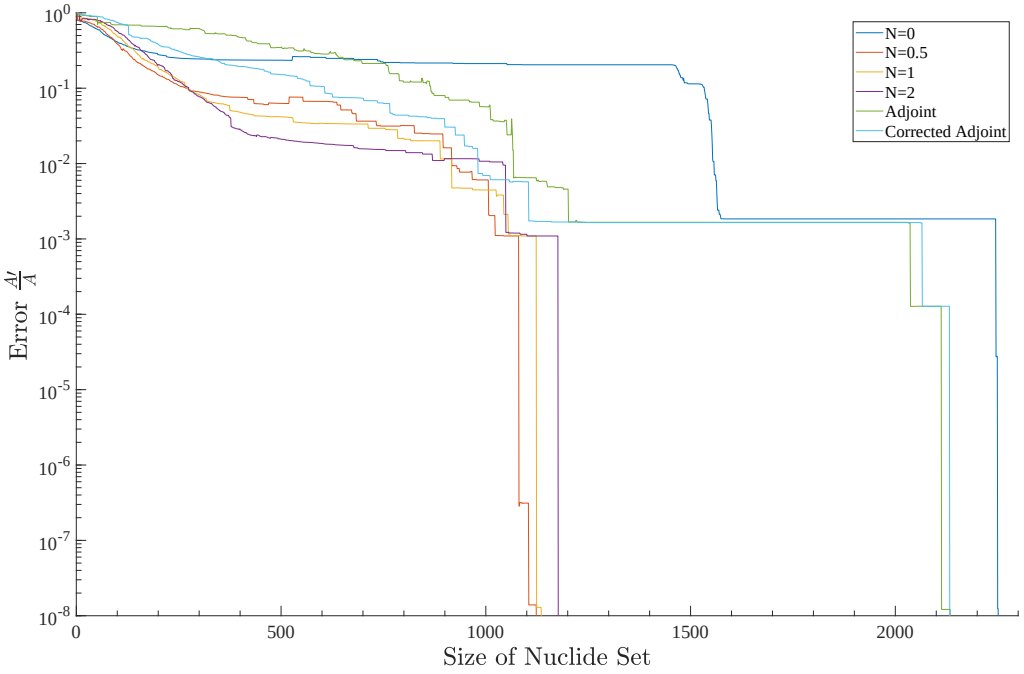



**Figure C.1.1:** Relative errors at a given library size reduced with the threshold-truncation method and the given weighting method for the total activity metric.

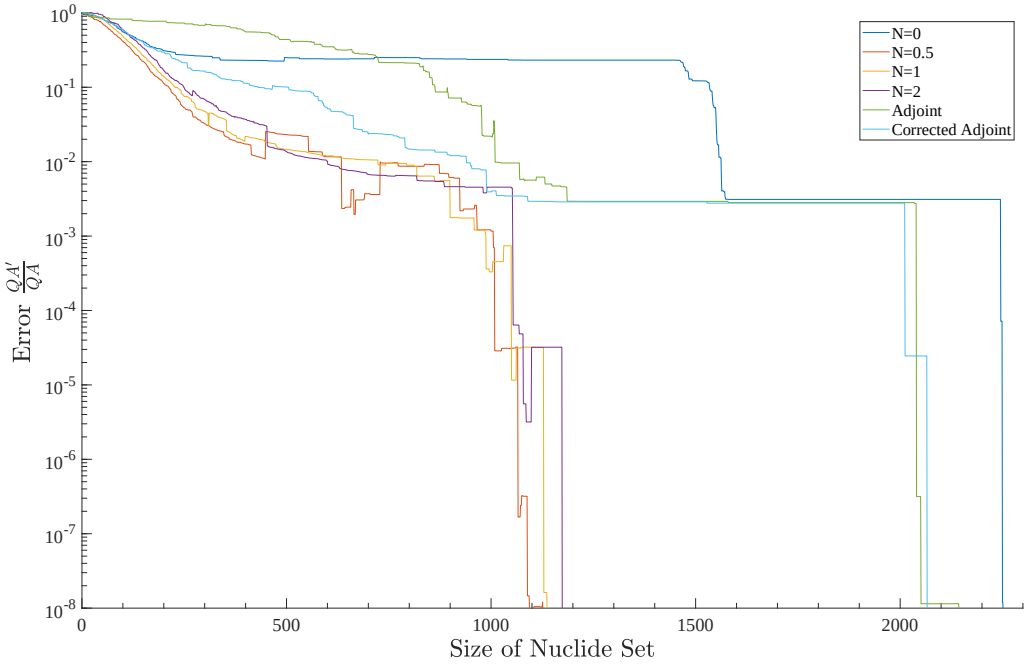



**Figure C.1.2:** Relative errors at a given library size reduced with the threshold-truncation method and the given weighting method for the decay heat metric.

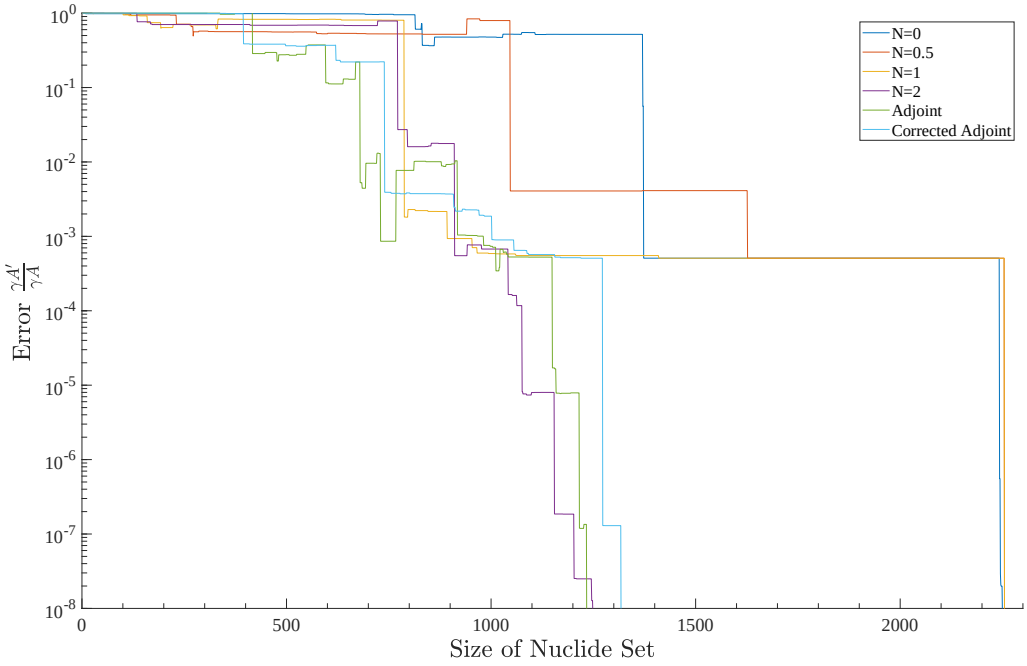



**Figure C.1.3:** Relative errors at a given library size reduced with the threshold-truncation method and the given weighting method for the gamma source metric.

## C.2 Greedy Algorithms

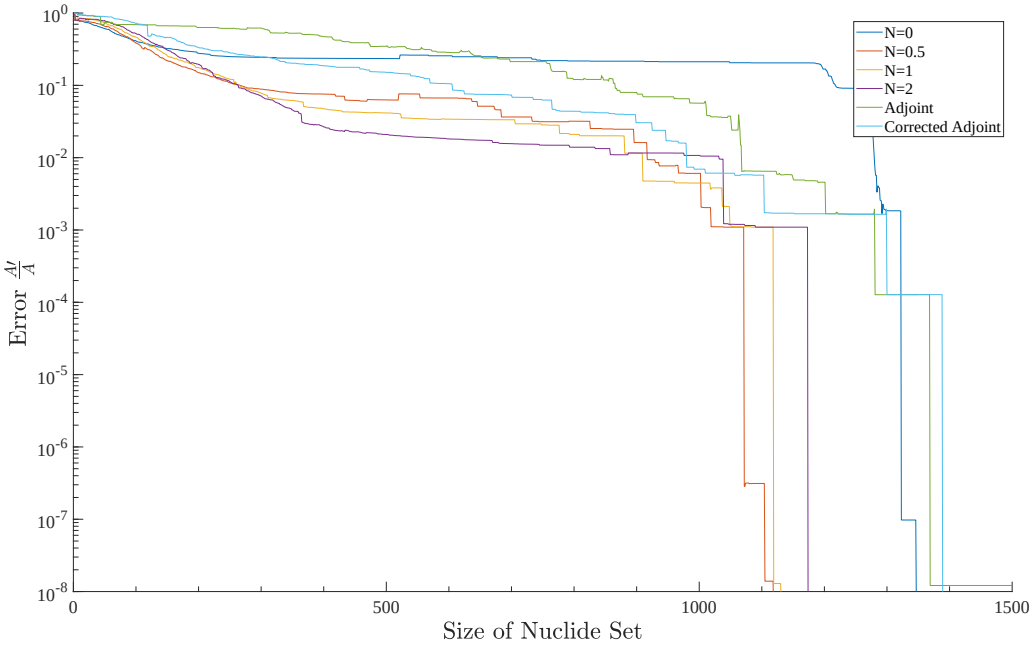



**Figure C.2.1:** Relative errors at a given library size reduced with the greedy algorithm method and the given weighting method for the total activity metric.

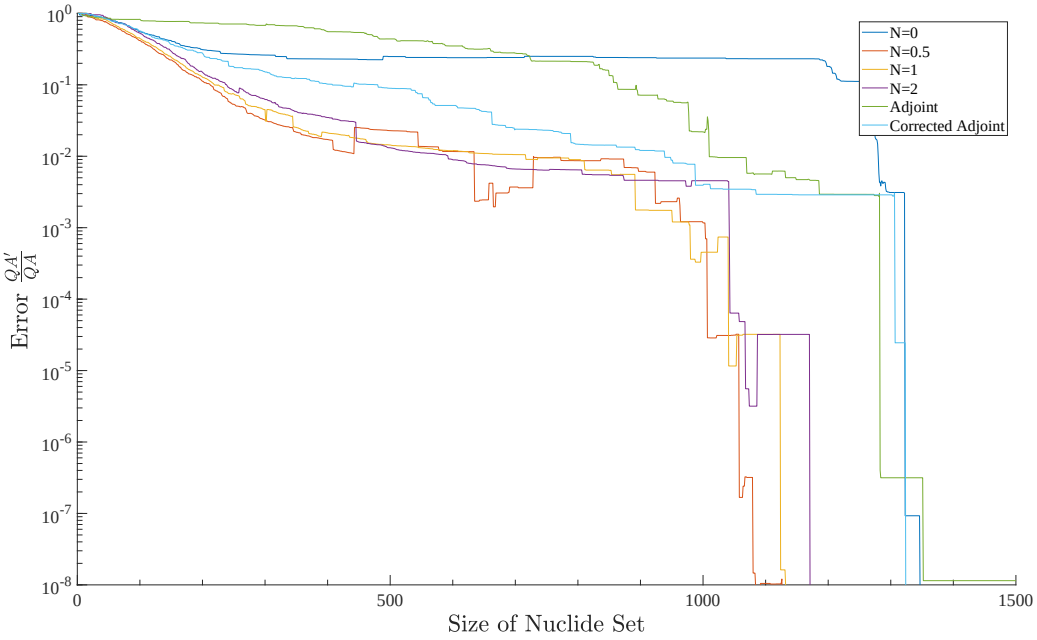



**Figure C.2.2:** Relative errors at a given library size reduced with the greedy algorithm method and the given weighting method for the decay heat metric

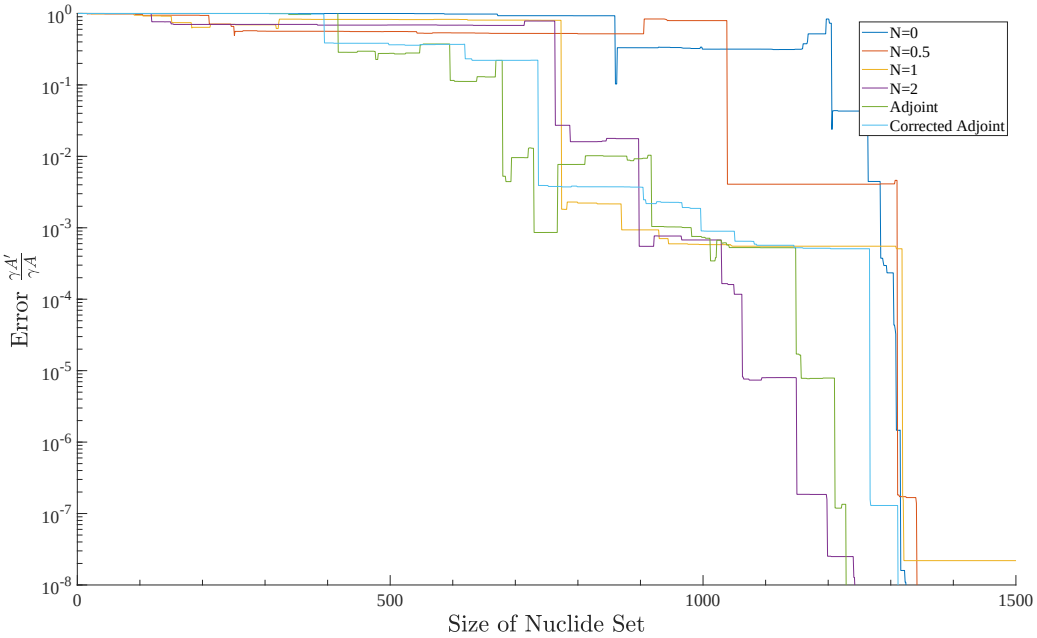



**Figure C.2.3:** Relative errors at a given library size reduced with the greedy algorithm method and the given weighting method for the gamma source metric

## C.3 Cutting Planes

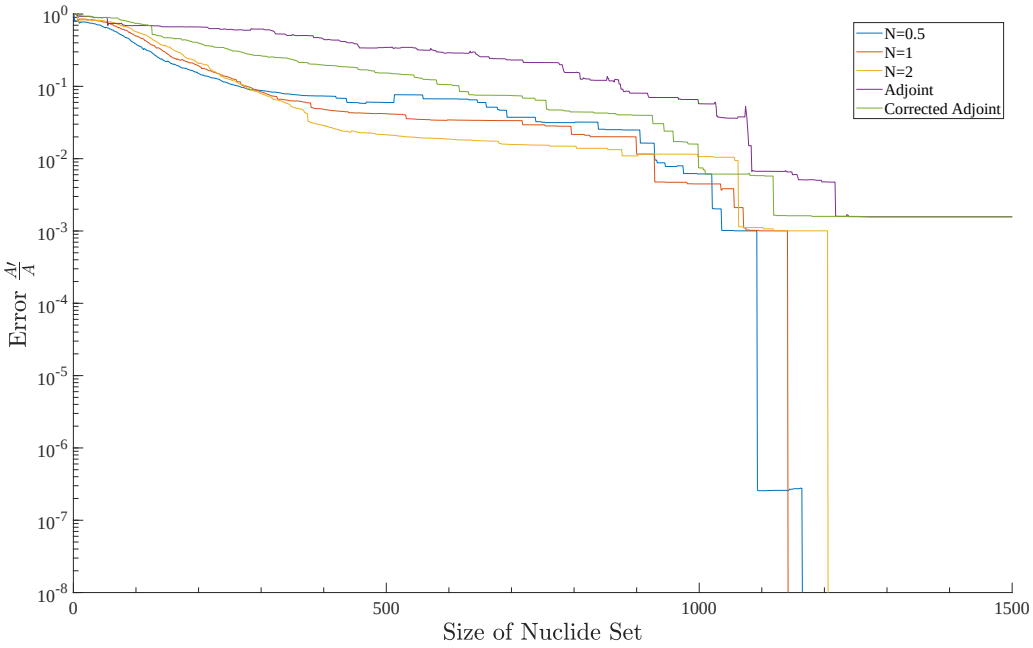



**Figure C.3.1:** Relative errors at a given library size reduced with the cutting planes algorithm method and the given weighting method for the total activity metric.

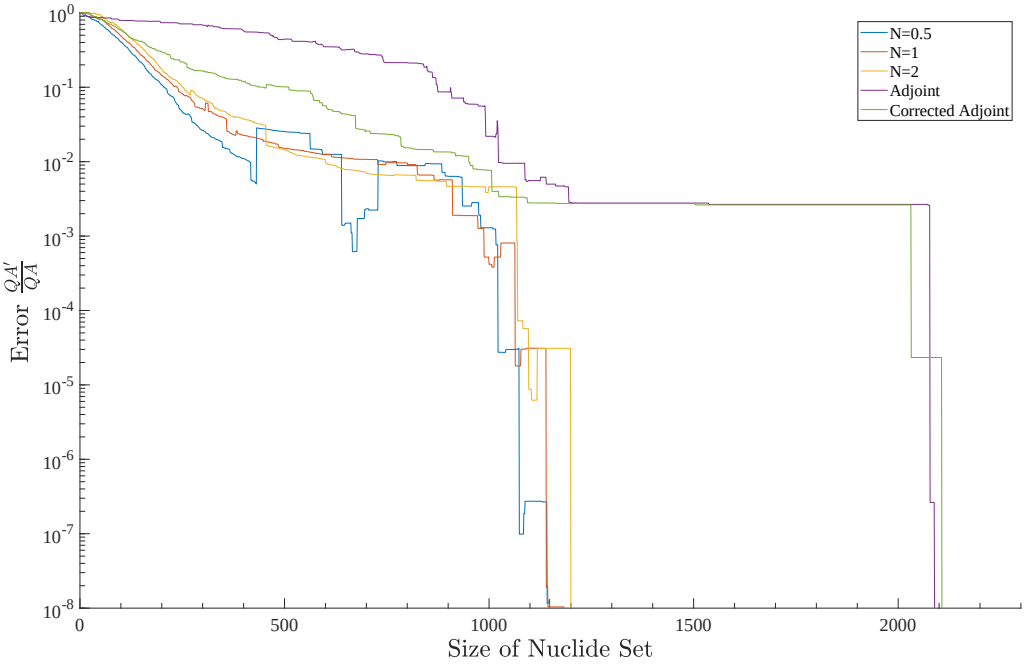



**Figure C.3.2:** Relative errors at a given library size reduced with the cutting planes algorithm method and the given weighting method for the decay heat metric

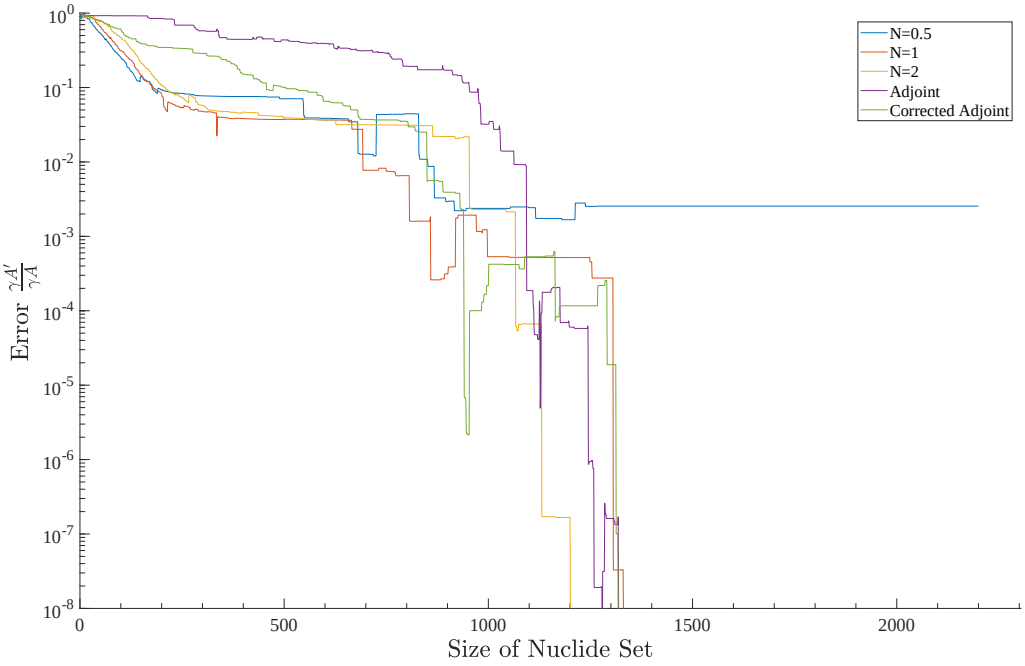



**Figure C.3.3:** Relative errors at a given library size reduced with the cutting planes algorithm method and the given weighting method for the gamma source metric

## C.4 Best Method Combinations

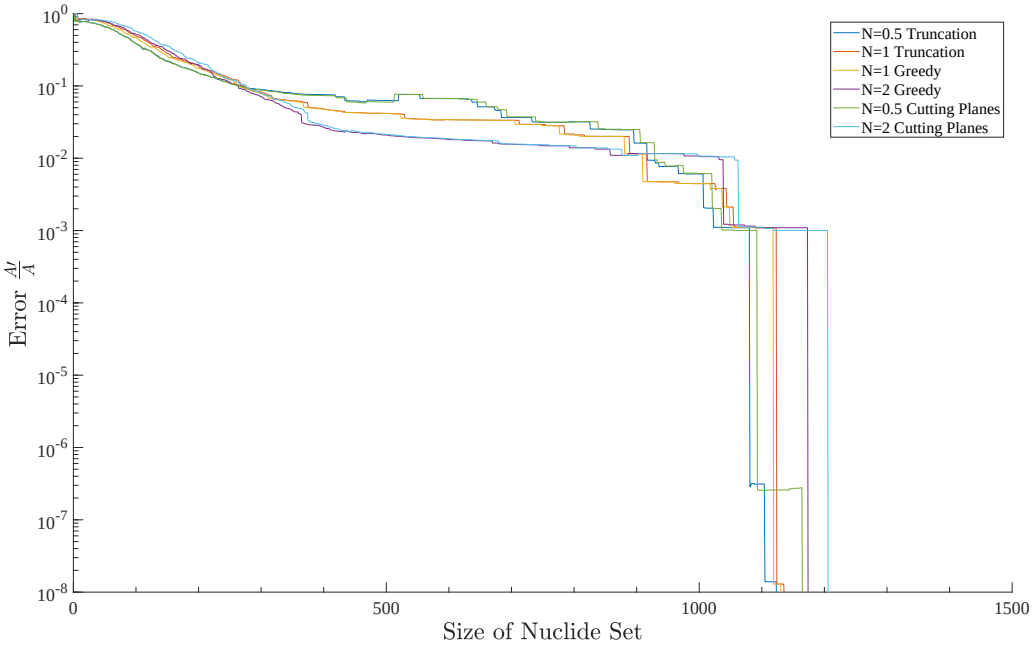



**Figure C.4.1:** Relative errors at a given library size reduced with the subsystem determination method and the weighting method given for the total activity metric.

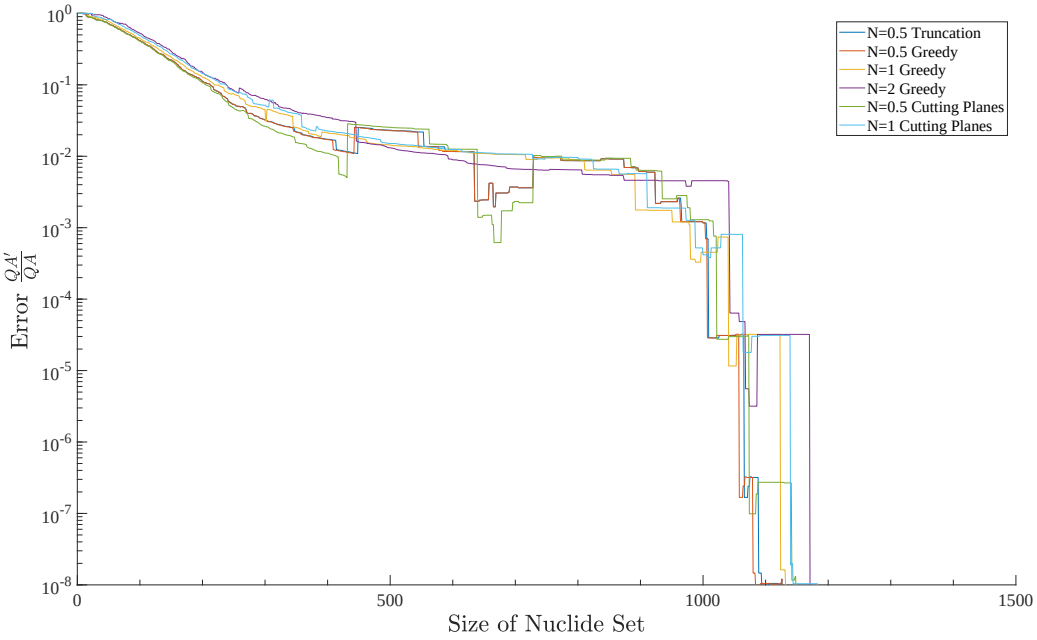



**Figure C.4.2:** Relative errors at a given library size reduced with the subsystem determination method and the weighting method given for the decay heat metric

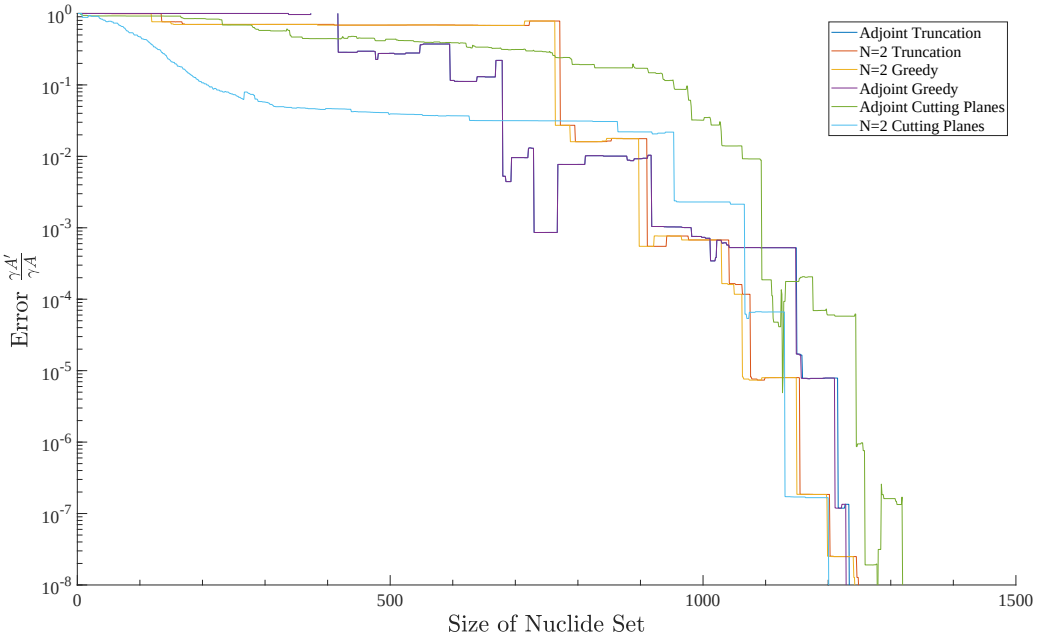



**Figure C.4.3:** Relative errors at a given library size reduced with the subsystem determination method and the weighting method given for the gamma source metric

## C.5 Expected Reduction Errors

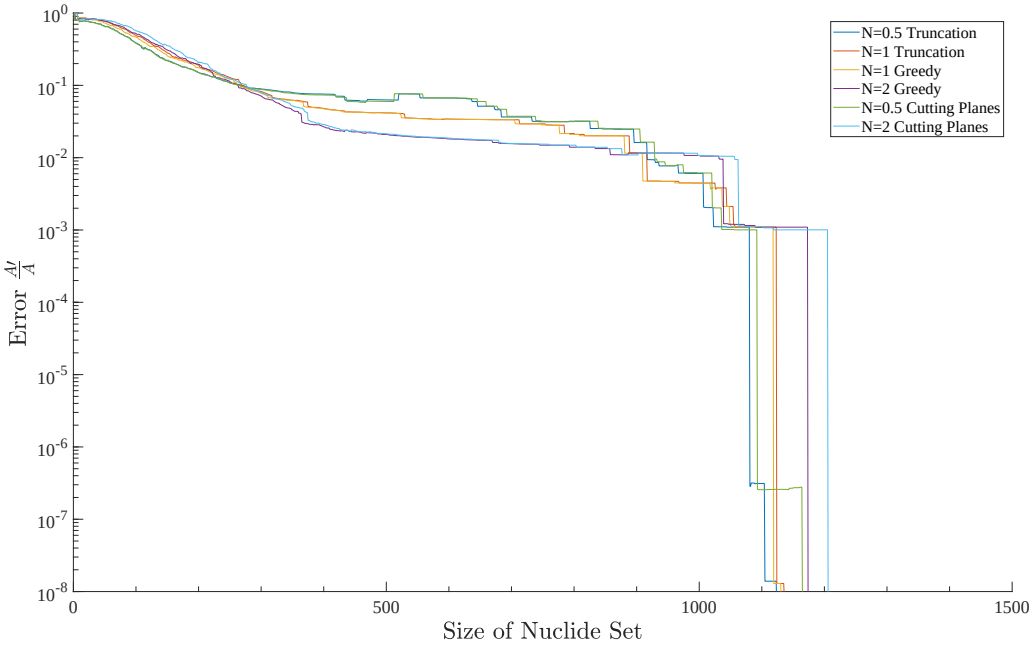



**Figure C.5.1:** Relative errors at a given library size reduced with the subsystem determination method and the weighting method given for the total activity metric.

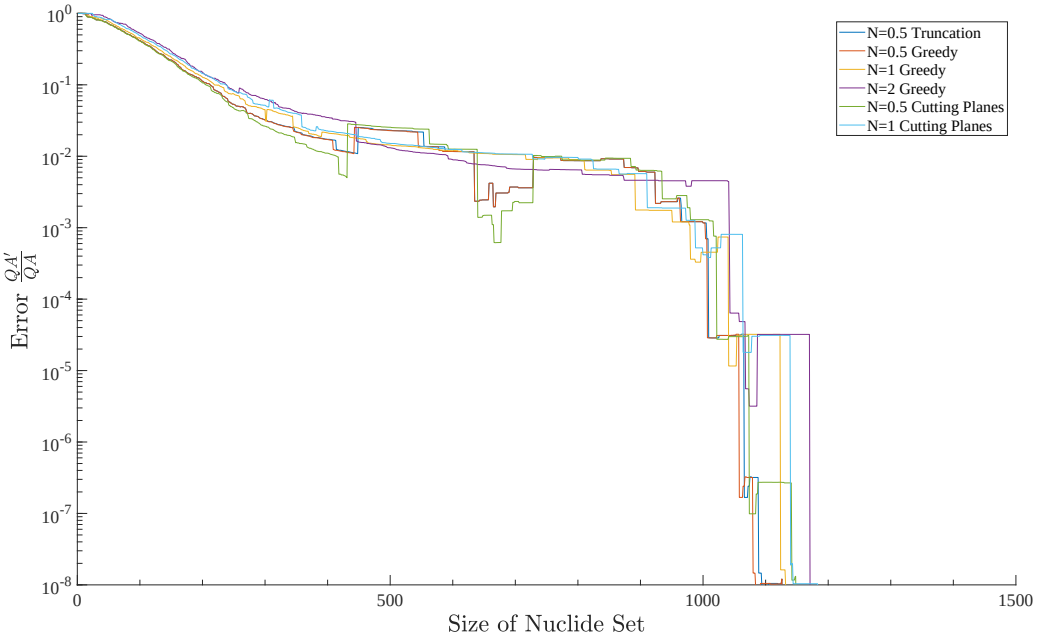



**Figure C.5.2:** Relative errors at a given library size reduced with the subsystem determination method and the weighting method given for the decay heat metric

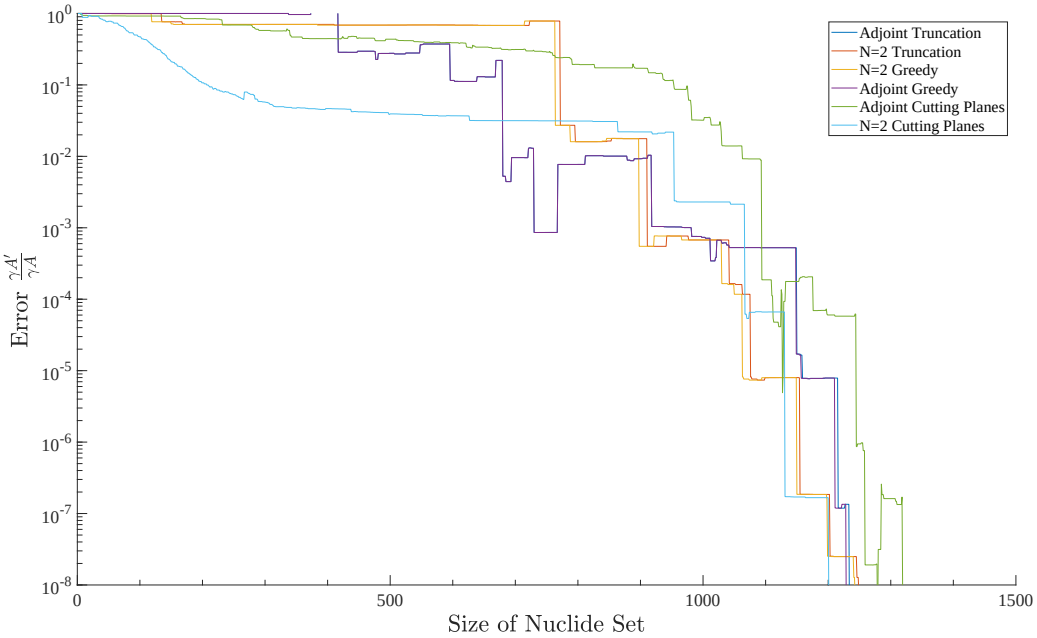



**Figure C.5.3:** Relative errors at a given library size reduced with the subsystem determination method and the weighting method given for the gamma source metric

# Appendix

# User's Manual

This section details the code that makes up the library reduction methods as well as its uses. The main code body is contained in the *LibraryReducer* class, which is at *packages/Origen/Manager/libld/* in the scale_dev directory for user sricha31 on the nuclear engineering cluster (necluster.engr.utk.edu). There is also example uses in the the connected */test/* directory in the file *tstLibraryReducer.cpp*.

The Origen libraries that are shipped with SCALE 6.2 do not use the most update conventions for the listing of decay MT numbers, nor do they explicitly treat reaction and decay products and byproducts in some scenarios. While this has been rectified in an updated version of the COUPLE module in SCALE, the other libraries need to be corrected for the library reduction techniques to functions correctly and preserve the original accuracy of the libraries as detailed in this dissertation. To do this the *changeDecayMTs* and *libraryUpdate* functions are available in the test file. Both functions take as input an Origen class smart pointer to a library (*Origen::SP_Library*) as the only input and make the changes to the library in-place.

## A Functions

The *LibraryReducer* class has 3 constructor functions, 6 reduction functions, and 6 visualization functions. Used with these functions are 3 enumeration types that are within

the scope of the *LibraryReducer* class. These 3 enumerations are to denote and inform the user of the currently supported options. These types are:

1. *ProblemType*

   Is used to denote the problem type and metric of interest for the user. There are 6 currently supported options: *DEPL* the depletion metric, *TACT* the total activity metric, *DECH* the decay heat metric, *GAMM* the gamma source metric, *CRIT* the criticality metric, *MASS* the total mass metric. This option is used in the *setProblem* function.

2. *RemovalType*

   Is used to denote the subsystem determination method the user wishes to use. There are 3 currently supported options: *DIR* the system truncation method, *MAP* the greedy algorithm method, *CP* the cutting planes method. This option is used in the *reduceLibrary* function that does not use a reduction template, the *graphError* function, and the *competition_comparison* function.

3. *WeightingType*

   Is used to denote the weighting method that the user wishes to use. There are 3 currently supported options: *CON* for the contribution weighting method of any degree, *ADJ* for the adjoint weighting function, and *ANA* for an analytic determination of weights. For the analytic option, nuclides are removed individually and independently with their weights being assigned as the change in metric from their removal. This weighting method is **very** slow, and not significantly more informative than the lower-degree contribution weighting methods. This option is used in the *setProblem* function.

These options shall be referred to by their declared type name for the remainder of this manual. Though they are primarily used in the the reduction functions, the users option choice for any function input is used in all subsequent function calls (if needed), unless that function has an option as an input.

## A.1 Constructors

***LibraryReducer()***

Base constructor that requires the user to independently set the library and resources later.

***LibraryReducer(SP_Library)***

Constructor sets the library that is input as a standard shared pointer to an Origen library class object as the original library. This library is assumed to be the intended target to both calculate weights and create the reduced library. The library that is pointed to is not altered by the *LibraryReducer* class. This constructor is sufficient for the creation of the sample problem, but to reduce the library the other resources must be set independently.

***LibraryReducer(SP_Library, SP_NuclideResource, SP_DecayResource)***

Constructor sets the library that is input via the shared pointer as the original library and the *NuclideResource* and *DecayResource* as the resources to be used to rebuild the library from the *TransitionSystem*. All three are to be input as standard shared pointers to the objects. All three will only be referenced and not altered. A default *NuclideResource* can be made from the object itself, while a decay resource can either be made via the instructions in the Origen manual or be found in the Scale Origen data directory.

## A.2 Reduction Functions

***setLibrary(SP_Library)***

Function set the input library as the original library for the reducing problem and the library reduction. This functions input will override a library input via a constructor and can be used to change the library to be reduced at any point. While, this can result in strange behavior if not used properly it is intended so that different libraries can be used for weighting and reducing, and so that a single *LibraryReducer* instance can be reused for multiple reductions.

***setResources(SP_NuclideResource, SP_DecayResource, SP_GammaResource)***

This function set the resources to be used to rebuild the library from the *Transition-System*, overriding those set in a constructor or previous function calls. This function is also the only way to set the gamma line source information that is needed to use the *GAMM ProblemType*. The *GammaResource* can be loaded via the associated loading function and a gamma line data file. An example of a gamma line data file can be found in the SCALE 6.2 Origen data directory.

***setProblem(Vec_Int, Vec_Dbl, Vec_Dbl, Vec_Dbl, Vec_Dbl, Vec_Dbl, ProblemType, std::vector⟨std::pair⟨double, double⟩⟩, WeightingType)***

This function acts to set the user defined reducing problem as well as set all of the original libraries metric contributors and their contributions. The 9 inputs to this function, in order, are: an ordered vector of integer IDs for the initial materials in the IZZZAAA format (metastable state I, atomic number ZZZ, mass number AAA), and ordered vector of doubles correlating the the mass of the corresponding initial material in kilograms, a vector of doubles for the length of each time-step in the problem in seconds (these time-steps are independent and not cumulative), a vector of doubles indicating the weight to be used for the associated time-step (a negative value weight at any position will result in the use of the default logarithmic weighting), a vector of doubles indicating the flux level at each time-step, a vector of doubles indicating the power level in watts at each time-step, a vector of paired upper and lower bounds for the energy ranges to be considered for any problem type that uses an energy integral metric, and the weighting type to be used to measure the importance of a nuclide. Of these inputs, the number of elements in the vectors for flux, power, and time weights must be equal to the number of time steps. Also, each ID given as an initial material must be given an inventory, even if it is zero. Due to the initial materials being preserved, it is possible to preserve nuclides that would otherwise be removed by including them as initial materials with zero inventory. Finally, both the flux values and the power values do not need to be entered. If a flux is given, it will be used over

the power value. If neither is given then the problem is set to be decay only. In any of these cases the information type that will not be used must be input as a vector of zeros with length equal to the number of time-steps.

***reduceLibrary(SP_Library&, int, RemovalType, double, bool)***

This function creates a new reduced library based on the previously given information and the inputs. The input library reference is used to create the new reduced library, resetting the smart pointer. A library does not need to be initialized outside of the function call as one is initialized and passed by reference using the pointer. The integer value is the degree of weighting to be used for the reduction. If adjoint weighting was selected in the *setProblem* function call, this value should be set to a negative number; otherwise, any zero or positive integer value will work. The *RemovalType* is the subsytem determination method to be used in this reduction. The double value is the error threshold that the user wishes to allow in the metric. The boolean input whether the user intends to use the visualization functions for this reductions. The default value is false, but if set to true then the library at each reduction iteration will be saved appended with the iteration number in its file name. If the *setProblem* function is not called prior to this function a default problem will be used. This default problem is composed of uranium metal, 5% enriched, and irradiated at $3.5x10^{19}\frac{n}{cm^2 s}$ for 3 cycles of 550 days with 30 days cooling between the power cycles and after the final power cycle. The depletion metric is used, as is contribution weighting, and as such the integer input value must not be negative if the default problem is used.

***reduceLibrary(SP_Library&, SP_Library)***

This function performs a library reduction of the set library to conform to the second input, template library. The reference to the library smart pointer is assigned to created, reduced library. This reduction copies the structure and nuclide set of the template library. If there is a nuclide or transition that is in the template library that is not in the set original library it is ignored in the reduction. The reduced library then is a result of the intersection of the two libraries.

***reduceLibrary(SP_Library&, Vec_Int)***

This function reduces the set original library to match the nuclide set input. The vector of integers denotes the nuclides that the user wishes to be the reduced library's nuclide set, with nuclide IDs given in SIZZZAAA format (sub-library S, metastable state I, atomic number ZZZ, mass number AAA). If a nuclide in the input vector is not in the original library, then it is ignored. As in the other functions the reference to the library smart pointer is assigned to the created, reduced library. The removals in this function are done by truncation, however direct fission products are removed first. This removal is done iteratively to ensure that the fission yields remain representative of their preceding depletion chains.

## A.3 Visualization Functions

The six visualization functions are split into 2 types, denoted by the function name syntax. Those functions that are written in "snake case" create an independent *TransitionSystem* from the reducing system to create the visualization information that is their output. The functions that are written in "camel case" create their visualization information from the *TransitionSystem* that is used in the reduction. These functions do need to, and do not alter the information. However, the first type of function replicates the removal in some fashion and thus needs an independent copy.

The user should be aware that these functions were written in order to aid in the development process, and thus at this time be considered in a functional but experimental state. There is some behavior that is currently hard-coded into these functions, though it is minimal, in order to create and test reduction effects that are not possible to analyze in the reduction functions. However, because these methods are very useful, and even necessary to fully replicate the work presented in this dissertation, their descriptions and uses are included in this manual.

***system_map(int, int)***

This function returns the nuclide weighting information at a given iteration steps that is used to create the nuclide chart type figures. The inputs to this function

are the iteration number that the user would like the weighting information for, and the weighting degree. To use this function the reduction must already be performed, with the visualization flag set to true. To use adjoint weighting any negative integer will work so long as adjoint weighting was selected when the reducing problem was set. The output of this function is a std::vector⟨std::vector⟨std::pair⟨int,double⟩⟩⟩. That is for every time step it will return a vector of nuclide IDs in SIZZZAAA format and their associated weight at that time.

***graph_error(int, RemovalType)***

This function determines the incremental error of the library reduction if performed on the original set library determined by the inputs. The integer value input is the weighting order, and like the *system_map* function a negative value will result in adjoint weighting if that option was used in the reducing problem. The RemovalType input will likewise determine the subsystem determination method. This visualization function can be used without reducing the library prior, so long as the reducing problem has been set. This function will determine the order of removals from the library, then assess the resulting cumulative error in the library after each reduction. This removal can take multiple hours of runtime. The output of this function is a std::vector⟨std::pair⟨int,double⟩⟩ that contains the ID of the nuclide that was removed and the resulting cumulative error following its removal in order from first removal to last possible removal. In the case of the option RemovalType::CP the integer value will not be the ID of the nuclide removed, but rather the size of the reduced library's nuclide set. This is because, the cutting planes method does not determine individual removals so multiple or no removals may happen in each step if the nuclide is preserved by being in the chain of a preserved nuclide.

***weighting_order(int)***

This function is similar to the*system_map* function, and is a simpler quicker version that does not require a reduction to be done. With this though the function is restricted to generating the visualization information to create the nuclide chart figures for just contribution weighting, input as the integer value equal to the desired degree.

The output from this function is formatted the same as the previous two, being a std::vector⟨std::pair⟨int,double⟩⟩. That is a vector that contains the nuclide IDs and weights as pairs.

***competition_comparison(int, RemovalType)***

This function performs a reduction on a library copy to a final nuclide set size of 400. This reduction uses the input weighting degree (input integer following the convention of the other methods) and the input RemovalType. After the reduction the competition correction subsystem correction method is used on the reduced library. This function is to test the efficacy of the competition correction and any changes in the method that it uses. The output of this function is a std::vector⟨std::pair⟨int,double⟩⟩ using the same format as the other visualization functions outputs. The information in this output is the IDs and weights of all of the nuclides in the corrected reduced library.

***systemEdges()***

This function generates a map of all the transitions in the library. This map is returned as a std::multimap⟨std::pair⟨int,int⟩⟩. This map is keyed on the transitions parent ID in SIZZZAAA format, and the values that are connected to that key are the IDs (also in SIZZZAAA format) of all products of transitions from that parent.

***transitionSpaceMap()***

This function generates the information that would result from the subspace transform used in the cutting planes method. That is it fully maps the transition system tracking a nuclides distance from a source material as both the number of transitions and the shortest mean transition time. This information is returned as a map keyed on the nuclides ID in SIZZZAAA format and the values connected to that key are the nuclides transition distance, the nuclides parent on the shortest transitional path, the shortest mean transitional time, and the parent on the shortest mean transition time path. The structure of this output is a std::map⟨int, std::pair⟨std::pair⟨int, int⟩, std::pair⟨double, int⟩⟩⟩ with the information referenced stored in the order given. This function is unique

in the visualization functions as it is also used as an integral part of the cutting planes method.

# B Sample Problems

This section details the inputs of 3 cases of library reduction, including the code input and a description of the purpose of each section. These samples are intended to detail the use of the *LibraryReducer* class's basic functions and options, starting with the simplest use possible, to an example that uses all of the problem specification options. It is assumed that the user has a level of proficiency with the C++ programming language.

## B.1 Default Problem

The simplest input for the *LibraryReducer* class is the use of the default problem.

```
#include "Origen/Core/Definitions.h"
#include "ScaleUtils/IO/DB.h"
#include "Origen/Core/re/DecayResource.h"
#include "Origen/Core/io/DecayResourceIO.h"
#include "Origen/Core/re/NuclideResource.h"
#include "Origen/Core/io/LibraryIO.h"
#include "Origen/Manager/libld/LibraryReducer.h"

//Set Resources
  ScaleUtils::IO::DB opts;

  SP_DecayResource dr = std::make_shared<Origen::DecayResource>();
  dr = std::make_shared<Origen::DecayResource>();
  Origen::DecayResourceIO dr_io;
  dr_io.load(*dr, decay_filepath, opts);

  SP_NuclideResource nr =std::make_shared<Origen::NuclideResource>();

  std::str libraryName = "w15_e50.f33";
  SP_Library lib = SP_Library(loadLibrary(libraryName, opts));
```

```
21   if (opts.numError() != 0) std::cerr << opts.dumpErrorStack() << std::endl;
22
23
24 //Set up Library Reduction
25   LibraryReducer libR(lib, nr, dr);
26   SP_Library libNew = SP_Library(new Library);
27   libR.reduceLibrary(libNew, 1, LibraryReducer::RemovalType::DIR, 0.01, false)
       ;
28
29 //Save Reduced Library
30   Origen::saveLibrary(*libNew, "reducedLibrary", opts);
31   if (opts.numError() != 0) std::cerr << opts.dumpErrorStack() << std::endl;
```

In lines 1–7 the file dependencies are listed in order for which they are used in the input, with file paths given relative to the Origen home directory. These dependencies include the input and output classes that are necessary to load the resources from their on-disk files and the resource classes. The two files that are not self explanatory are the *Definitions.h* and the *DB.h* files. The *Definitions.h* file contains the Origen version of the standard vector types that are used as inputs in some functions. The *DB.h* file contains a data container class that allows for the loading and changing of options for the resource types, as well as error handling in the loading and saving process. This data container has other uses and functionality as well, but this is functions that it serves in the context of the library reduction input.

Lines 9–21 are the loading of the resource files and the library that is to be used for the reducing problem and creation of the reduced library with the default loading options. The resource and library files used in the loading operations should contain the full path to file relative to the location of the reducing code location. The *NuclideResource* does not need an input file (though one can be used) as default information is loaded as part of the resource's constructor. After all of the on-disk information has been loaded any error that resulted from the operations are handled through the data container.

Lines 24–27 detail the declaration of the *LibraryReducer* with the library and resources set via the constructor. With no problem set prior to the library reduction call on line 27, the default problem is used. The weighting type for this reduction will be contribution

weighting of the first degree and the subsystem determination method is truncation. The final allowable error in the metric is given to be 0.001 fractional error (1%).

After the reduced library has been created it is saved with the name *reducedLibrary* to the current directory using the *saveLibrary* function call on line 30.

## B.2 Setting a Simple Problem

This example is an expansion of the previous example, with the problem being set rather than using the default problem. Only the code that is not present, or changed in the previous example is described.

```
1  #include "Origen/Core/Definitions.h"
2  #include "ScaleUtils/IO/DB.h"
3  #include "Origen/Core/re/DecayResource.h"
4  #include "Origen/Core/io/DecayResourceIO.h"
5  #include "Origen/Core/re/NuclideResource.h"
6  #include "Origen/Core/io/LibraryIO.h"
7  #include "Origen/Manager/libld/LibraryReducer.h"
8
9  //Set Resources
10   ScaleUtils::IO::DB opts;
11
12   SP_DecayResource dr = std::make_shared<Origen::DecayResource>();
13   dr = std::make_shared<Origen::DecayResource>();
14   Origen::DecayResourceIO dr_io;
15   dr_io.load(*dr, decay_filepath, opts);
16
17   SP_NuclideResource nr =std::make_shared<Origen::NuclideResource>();
18
19   std::str libraryName = "w15_e50.f33";
20   SP_Library lib = SP_Library(loadLibrary(libraryName, opts));
21   if (opts.numError() != 0) std::cerr << opts.dumpErrorStack() << std::endl;
22
23 //Set up Reducing Problem
24   //Initial materials are: 1MT uranium in UO2 form (5% Enriched), 100kg Al, 1
       kg Gd
```

```
25  Origen::Vec_Int initial_ids = {92235, 92238, 8016, 13027, 64157};
26  Origen::Vec_Dbl inventory;
27  inventory.push_back(50.0);
28  inventory.push_back(950.0);
29  inventory.push_back(134.5396);
30  inventory.push_back(100.0);
31  inventory.push_back(1.0);
32
33  //One irradiation cycle (1.5 years at 40 MW), one cooling cycle for 20 days
34  Origen::Vec_Dbl timeSteps, powers, fluxes, t_weights;
35  double powerCycleLength = 1.5*365.25*24*60*60;
36  double coolingCycleLength = 20*24*60*60;
37  timeSteps = {powerCycleLength, coolingCycleLength};
38  powers    = {40e6, 0.0};
39  fluxes    = {0.0, 0.0};
40  t_weights = {-1.0, -1.0};
41
42  //energy bounds that will not be used but must be passed to the function
43  std::vector<std::pair<double, double>> energies = {std::make_pair(1.0, 0.0)
      };
44
45 //Set up Library Reduction
46  LibraryReducer libR(lib, nr, dr);
47
48  libR.setProblem(initial_ids, inventory,
49                  timeSteps, t_weights, fluxes, powers,
50                  LibraryReducer::ProblemType::DEPL, energies,
51                  LibraryReducer::WeightingType::CON)
52
53  SP_Library libNew = SP_Library(new Library);
54  libR.reduceLibrary(libNew, 2, LibraryReducer::RemovalType::DIR, 0.01, false)
      ;
55
56 //Save Reduced Library
57  Origen::saveLibrary(*libNew, "reducedLibrary", opts);
58  if (opts.numError() != 0) std::cerr << opts.dumpErrorStack() << std::endl;
```

In lines 25–31 the initial materials are initialized. These materials are U-235, U-238, O-16, Al-27, and Gd-157 with 50 kg, 950 kg, 134.5396 kg, 100 kg, 1 kg amounts respectively. The amount of oxygen is relative the equivalent amount that would be found in $UO_2$ that contains 1 metric tonne of uranium. The order of the IDs must be the same as the order of the quantities.

Lines 34–40 detail the irradiation history. The history in this case is a single one and half year irradiation at 40 MW followed by 20 days cooling. The weighting for the time-steps is set to use the default logarithmic weighting. All of the vectors in this section need to be of the same length, which is why the fluxes are declared as zeros. This indicates that power should be used.

On line 43 an arbitrary energy bounds is declared. This is a required input for the *setProblem* function, even if the metric to be used will not make use of it. The bounds given are a single range that includes all energies up to 1 MeV.

On line 48–51 the information for the reducing problem that has been initialized is set for the *LibraryReducer*. The further options to use the depletion metric and contribution weighting are also declared in this function call. Once this function has been called the initial solve of the problem is done, setting the subsequent results that the reduced libraries will be compared to.

The remainder of the reduction input is the same except that instead of first-degree contribution weighting, the reduction is to use second-degree contribution weighting. This change is on line 54.

## B.3 Setting a Detailed Problem and Reduction

This last example problem is similar to the previous two with the exceptions that it uses a more detailed irradiation history, custom time weighting, a metric that makes use of the energy bounds, and uses a different weighting and subsystem determination method.

```
1 #include "Origen/Core/Definitions.h"
2 #include "ScaleUtils/IO/DB.h"
3 #include "Origen/Core/re/DecayResource.h"
4 #include "Origen/Core/io/DecayResourceIO.h"
```

```
#include "Origen/Core/re/NuclideResource.h"
#include "Origen/Core/io/GammaResourceIO.h"
#include "Origen/Core/re/GammaResource.h"
#include "Origen/Core/io/LibraryIO.h"
#include "Origen/Manager/libld/LibraryReducer.h"

//Set Resources
  ScaleUtils::IO::DB opts;

  SP_DecayResource dr = std::make_shared<Origen::DecayResource>();
  dr = std::make_shared<Origen::DecayResource>();
  Origen::DecayResourceIO dr_io;
  dr_io.load(*dr, decay_filepath, opts);

  SP_NuclideResource nr =std::make_shared<Origen::NuclideResource>();

  SP_GammaResource gr = std::make_shared<Origen::GammaResource>();
  Origen::GammaResourceIO gr_io;
  gr_io.load(*gr, "/opt/scale6.2_data/origen_data/origen.rev04.mpdkxgam.data",
    opts);

  std::str libraryName = "w15_e50.f33";
  SP_Library lib = SP_Library(loadLibrary(libraryName, opts));
  if (opts.numError() != 0) std::cerr << opts.dumpErrorStack() << std::endl;

//Set up Reducing Problem
  //Initial materials are: 1MT uranium in UO2 form (5% Enriched), 100kg Al, 1
   kg Gd
  Origen::Vec_Int initial_ids = {92235, 92238, 8016, 13027, 64157};
  Origen::Vec_Dbl inventory;
  inventory.push_back(50.0);
  inventory.push_back(950.0);
  inventory.push_back(134.5396);
  inventory.push_back(100.0);
  inventory.push_back(1.0);

```

```
39   //One irradiation cycle (1.5 years at 40 MW), one cooling cycle for 20 days
40   Origen::Vec_Dbl timeSteps, powers, fluxes, t_weights;
41   double powerCycleLength = 500*24*60*60;
42   double downCycleLength = 30*24*60*60;
43   double coolingCylceLength = 5*365.25*24*60*60;
44   timeSteps = {powerCycleLength, downCycleLength,
45                powerCycleLength, downCycleLength,
46                powerCycleLength, coolingCycleLength};
47   powers    = {40e6, 0.0, 40e6, 0.0, 40e6, 0.0};
48   fluxes    = {0.0, 0.0, 0.0, 0.0, 0.0, 0.0};
49   t_weights = {0.0, 1.0, 0.0, 1.0, 0.0, 1.0};
50
51   //energy bounds for the gamma energy lines to be considered for the metric
52   std::vector<std::pair<double, double>> energies = {std::make_pair(1.0, 0.0)
       };
53
54 //Set up Library Reduction
55   LibraryReducer libR(lib);
56   libR.setResources(nr, dr, gr);
57
58   libR.setProblem(initial_ids, inventory,
59                   timeSteps, t_weights, fluxes, powers,
60                   LibraryReducer::ProblemType::GAMM, energies,
61                   LibraryReducer::WeightingType::ADJ)
62
63   SP_Library libNew = SP_Library(new Library);
64   libR.reduceLibrary(libNew, -1, LibraryReducer::RemovalType::CP, 0.01, false)
       ;
65
66 //Save Reduced Library
67   Origen::saveLibrary(*libNew, "reducedLibrary", opts);
68   if (opts.numError() != 0) std::cerr << opts.dumpErrorStack() << std::endl;
```

The differences in this problem begin on line 6 and line 21. A gamma resource is initialized and loaded from the on-disk resource. The file path given for this gamma resource is an example of the file paths to SCALE 6.2 Origen data. In this case the directory given is

the one on the nuclear engineering computing cluster. This resource, with associated file dependencies, are used in the same manner as the previously mentioned resource types.

The problem described starting on line 31 has the same initial materials as the last example but the irradiation history has 3 power cycles of 500 days at 40 MW with 30 days down between each irradiation followed by 5 years of cooling. Furthermore, only non-power steps are considered of importance to this problem. This is declared on line 49 and indicated by time-steps correlated to zero-power steps being given weights of 1.0. Once again this problem uses power instead of flux but all vectors expand in length to match the number of time-steps.

One line 55 the library only constructor is used. This is because to set the gamma resource the *setResources* function must be called (line 56), so declaring resources in the constructor would be redundant.

On line 58 the *setProblem* function is called using the initialized problem information. However, in this case the problem is set to use the gamma source metric and calculate the adjoint weights. The declaration of the adjoint weighting option in this function call does not require that the adjoint weights be used in the library reduction, it merely allows them to be used by doing the requisite calculations. If however, the analytic weighting type is declared in the problem set it will be used in the reduction in the same manner and with the same options the contribution weighting would be.

The final difference is on line 64 in the *reduceLibrary* function call. The weighting degree is given as -1 to indicate that adjoint weighting should be used for the reduction and the removal type is set to CP to use the cutting planes method.

**NOTE**: The library reduction assumes a 3 sub-library structure to the library. While this is not always the case (such as in decay libraries), it is generally true for both the standard libraries and those made through COUPLE. This structure is required for the use of the library reduction class, as well as the solver in the *Material* class that it uses to manage the concentrations and the solver.

# C All Code Descriptions

While the previous sections detail the user interface with the library reduction code, they compose only a fraction of the code that makes up the code base of the *LibraryReducer* class. In order to aid in future development of the code, private functions shall also be listed and described in brief detail.

***setDefaultProblem***

Function sets the problem to be the previously mentioned default problem.

***setMaterial***

Function creates a *Material* object that manages the nuclide concentrations and the depletion solver.

***solveProblem***

Function goes through the set problem, or the adjoint problem depending on the options used, and manages the burnup dependent information for setMaterial. This function also calls for the weights and metric to be calculated.

***setWeights***

This function the direct contributions to the metric type based on the concentrations on the material object for all time steps.

***setSensitivities***

This function calculates the contributon contribution values for a single nuclides contributon concentrations. This function is called for each nuclides contribution concentration in the *setProblem* function if the adjoint option is used.

***setAnalyticWeights***

This function iteratively removes a single nuclide and then replaces it in the library, calculating the analytic importance of each nuclide.

***makeTransitionSystems***

This function manages the library's translation to a series of *TransitionSystem*s, one for each burnup on the library.

***makeTransitionSystem***

This function performs the actual translation of a library to two *TransitionSystem*s, one for light nuclides and one for fission products and actinides. This function is called by *makeTransitionSystems*.

***removeNuclide***

This function handles the removal of a nuclide from the *TransitionSystems*. This function manages all memory de-allocation through the use of smart pointers, and calculates the cumulative fission yields after a product is removed.

***eliminateDuplicates***

This functions iterates through all *Nuclide* objects in the *TransitionSystem* for light nuclides and compares it to those in the *TransitionSystem* for fission products and actinides. Any matching *Nuclides* have their transitions unioned into the *Nuclide* on the *TransitionSystem* for fission products and actinides removing the instance in the *TransitionSystem* for light nuclides.

***eliminateOrphans***

This function iterates through all *Nuclide* objects on the *TransitionSystems* removing any that have no gain or loss transitions, with the exception of initial materials.

***eliminateSLFP***

This function will remove an direct fission products that have a decay constant above the set threshold level. This removal is iterative and continues until no direct or cumulative fission products remain that have a decay constants above the threshold.

***compareLibraries***

This functions determines the error in the metric from the reduction by having the reduced library built and then the problem resolved using the new library, measuring error as the absolute relative difference in the metric.

***rebuildLibrary***

This function translates the *TransitionSystems* into a new library.

***matchDecayData***

This function ensures the a new libraries *DecayData* matches that of the original library.

***weightDepl***

This function calculates the metric contributions for the depletion metric.

***weightCrit***

This function calculates the metric contributions for the criticality metric.

***weightTact***

This function calculates the metric contributions for the total activity metric.

***weightMass***

This function calculates the metric contributions for the mass metric.

***weightGamm***

This function calculates the metric contributions for the gamma source metric.

***weightDech***

This function calculates the metric contributions for the decay heat metric.

***populateKappas***

This function sets the kappa values for captures and fission if those values are not already present on the library. Most libraries, and rebuilt libraries have these values, but those libraries built directly from resources (this functionality is not presented in this manual) do not have those values.

***weightNuclides_nth***

This function calculates the relative weights for the contribution weightings, of the degree input, from the direct metric contributions.

***weightNuclides_adjoint***

This function calculates the relative weights for the adjoint weighting from the absolute adjoint weights.

***findRemovals_direct***

This function compiles a list, via the truncation method, of nuclides that should be removed.

***findRemovals_dijkstra***

This function compiles a list, via the greedy algorithm method, of nuclides that should be removed.

***findRemovals_SYShybrid***

This function compiles a list, via a variant of the cutting planes method, of nuclides that should be removed. Rather than determining the end of a branch the standard way, this function clusters nuclides that are both close to each other and have sufficiently high weights. This method is not currently supported as an option.

***findRemovals_NUChybrid***

This function compiles a list, via the cutting planes method, of nuclides that should be removed.

***competitionCorrection***

This function makes corrections to a list of nuclides to be removed. The corrections remove a nuclide from the list if it is determined that the nuclide represents an important competing pathway.

# Vita

Scott Richards was born in Detroit, MI, to parents David and Judy Richards. He is the younger brother to his sister Nicole. He attended St. Peter and Paul, and St. Sabina Elementary Schools prior to moving to Chelsea, MI, where he attended Chelsea Middle School and High School. After graduation, he attended the University of Michigan in Ann Arbor, MI. There he received both his Bachelor's and Master's degrees in Nuclear Engineering and Radiological Sciences in 2013 and 2014 respectively. He accepted a graduate researcher position at The University of Tennessee, Knoxville in the Nuclear Engineering department. In 2016, Scott was made a fellow of the Consortium for Nonproliferation Enabling Capabilities. He is continuing his research career as a postdoctoral appointee at Argonne National Laboratory with the Nuclear Science and Engineering Division.