The "Virtual Density" Theory of Neutronics

Mark Wilbert Reed

S.M. Nuclear Science and Engineering, Massachusetts Institute of Technology, 2011

S.B. Nuclear Science and Engineering, Massachusetts Institute of Technology, 2011

S.B. Physics, Massachusetts Institute of Technology, 2010

Submitted in Partial Fulfillment of the Requirements
for the Degree of
DOCTOR OF SCIENCE IN NUCLEAR SCIENCE AND ENGINEERING
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
February 2014

Signature of Author: ....
Department of Nuclear Science and Engineering
January 22, 2014

Certified by: ....
Kord S. Smith, Thesis Supervisor
KEPCO Professor of the Practice of Nuclear Science and Engineering

Certified by: ....
Benoit Forget, Thesis Reader
Associate Professor of Nuclear Science and Engineering

Accepted by: ....
Mujid S. Kazimi, Chairman of Committee on Graduate Students
TEPCO Professor of Nuclear Engineering

This page is intentionally blank.

The “Virtual Density” Theory of Neutronics

-

Mark Wilbert Reed

-

Submitted to the Department of Nuclear Science and Engineering on January 22, 2014 in Partial Fulfillment of the Requirements for the Degree of Doctor of Science in Nuclear Science and Engineering

Abstract

Sustainable nuclear energy will likely require fast reactors to complement the current light water reactor paradigm. In particular, breed-and-burn sodium fast reactors (SFRs) offer a unique combination of fuel cycle and power density features. Unfortunately, large breed-and-burn SFRs are plagued by positive sodium void worth. In order to mitigate this drawback, one must quantify various sources of negative reactivty feedback, among which geometry distortions (bowing and flowering of fuel assemblies) are often dominant. These distortions arise mainly from three distinct physical phenomena: irradiation swelling, thermal swelling, and seismic events.

Distortions are notoriously difficult to model, because they break symmetry and periodicity. Currently, no efficient and fully generic method exists for computing neutronic effects of distortions. Computing them directly via diffusion would require construction of exotic hyperfine meshes with continuous re-meshing. Many deterministic transport methods are geometrically flexible but would require tedious, intricate re-meshing or re-tracking to capture distortion effects. Monte Carlo offers the only high-fidelity approach to arbitrary geometry, but resolving minute reactivities and flux shift tallies within large heterogeneous cores requires CPU years per case and is thus prohibitively expensive. Currently, the most widely-used methods consist of various approximations involving weighting the uniform radial swelling reactivity coefficient by the power distribution. These approximations agree fairly well with experimental data for flowering in some cores, but they are not fully generic and cannot be trusted for arbitrary distortions. Boundary perturbation theory, developed in the 1980s, is fully general and mathematically rigorous, but it is inaccurate for coarse mesh diffusion and has apparently never been applied in industry.

Our solution is the “virtual density” theory of neutronics, which alters material density (isotropically or anisotropically) instead of explicitly changing geometry. While geometry is discretized, material densities occupy a continuous domain; this allows density changes to obviate the greatest computational challenges of geometry changes. Although primitive forms of this theory exist in Soviet literature, they are only applicable to cases in which entire cores swell uniformly. Thus, we conceive a much more general and pragmatic form of “virtual density” theory to model non-uniform and localized geometry distortions via perturbation theory.

In order to efficiently validate “virtual density” perturbation theory, we conceive the “virtual mesh” method for diffusion theory. This new method involves constructing a slightly perturbed "fake" mesh that produces correct first-order reactivity and flux shifts due to anisotropic swelling or expansion of individual mesh cells. First order reactivities computed on a "virtual mesh" agree with continuous energy Monte Carlo to within 1σ uncertainty.

We validate "virtual density" theory via the "virtual mesh" method in 3-D coarse mesh models of the Fast Flux Test Facility (FFTF) and Jōyō benchmarks using the MATLAB-PETSc-SLEPc (MaPS) multigroup finite difference diffusion code, which we developed for this purpose. We model a panoply of non-uniform anisotropic swelling scenarios, including axial swelling of individual assemblies, axial swelling of each mesh cell in proportion to its fission power, and radial core flowering with arbitrary axial dependence. In 3-D coarse mesh Cartesian cores with explicit coolant gaps, we model individual assembly motion, assembly row motion with arbitrary axial dependence, and assembly row "s-shape" bowing. In all cases, we find that "virtual density" perturbation theory predicts reactivity coefficients that agree with "virtual mesh" reference cases to within 0.01%. These reactivity coefficients are two to four orders of magnitude more accurate than those computed via boundary perturbation theory. We also develop the Pseudo-Seismic (PseuSei) Animator within MaPS to explore point-kinetic effects of arbitrary assembly motion for 3-D coarse mesh Cartesian cases. In general, this "virtual density" perturbation method can precisely predict reactivity coefficients due to anisotropic swelling or expansion of any core region in *any* direction.

Furthermore, we compute flux and power shift distributions due to geometry distortions. We find that our "virtual density" formalism couples seamlessly with existing modal expansion perturbation theory (MEPT) formalism, and we use the resulting new hybrid method to compute flux and power shifts due to arbitrary anisotropic swelling of arbitrary core regions. We test this new method for a large, highly-heterogeneous Cartesian core, and we find that predicted (global and local) flux and power shift distributions typically agree with "virtual mesh" reference cases to within a few percent.

Development of the "Virtual Density" Theory (VirDenT) industry code constitutes the culmination of this work. This parallelized Python code computes "virtual density" reactivity coefficients given a DIF3D flux solution as input. VirDenT contains a flux reconstruction module that computes individual pin powers from a homogenized nodal diffusion solution. It also contains PyPinPlot, a high-resolution visualization tool for pin-level powers, fluxes, and current vector fields. Most importantly, VirDenT computes reactivity coefficients due to local anisotropic swelling of assembly zones (which direct diffusion theory cannot compute) in CPU seconds, while Monte Carlo (currently the only high-fidelity approach) requires CPU years to do the same.

Thesis Supervisor: Kord S. Smith
Title: KEPCO Professor of the Practice of Nuclear Science and Engineering
Thesis Reader: Benoit Forget
Title: Associate Professor of Nuclear Science and Engineering

Acknowledgements

I acknowledge much, because I haven't done this alone.

First and foremost, I must thank my thesis advisor Kord Smith, who graciously agreed to advise me in this grand endeavor. He has raised (and continues to raise) technical standards in NSE, and he ultimately raised my own standards for myself - even when I didn't appreciate it in the moment. Ben Forget helped me out a great deal early on in graduate school, and he served as a reader for both my S.M. and Sc.D. theses on wildly disparate topics. He has read over 700 pages of my writing, more than any other person. Professor Ron Parker was an excellent advisor for my S.M. thesis, which allowed me to "smoothly" transition from fusion to fission. Eugene Shwageraus served as my *de facto* advisor during the qualifying exam process, shared my enthusiasm for fission yield, and helped me procure some obscure Russian literature. Professor Lester has been extremely positive and affirmative, and his tenure as Department Head has revitalized NSE. Professor Hutchinson has always been willing to discuss my various concerns, both professional and personal. I must also thank Professor Buongiorno, who gave me first pick of a teaching assistantship when I desperately needed funding. Furthermore, he has always been willing to discuss welcomeness and inclusion issues within NSE, and he actually cares. Helping teach 22.101 with Professor Yildiz in Fall 2011 was a very rewarding experience, and she graciously agreed to serve on my defense committee. Serving as the gooey glue that held together 22.101 when taught by three lecturers - Professors Cappellaro, Li, and Yip - in Fall 2012 was a rewarding challenge. I also deeply appreciate Professors Driscoll and Todreas for allowing me to interview them for my *Fortnight Journal* essay project. They are wealths of knowledge.

MIT Professors beyond NSE have also deserve much thanks. Serving as a 2.005 teaching assistant with Professors Buie and Lermusiaux during the 2010-2011 academic year was an exhilarating experience. Alan Guth, my undergraduate advisor, is extremely caring, humble, and eager to help or serve anyone. Professor Bertschinger, Course 8 Department Head during my undergraduate years and now Institute Community and Equity Officer, truly cares about affirming disadvantaged people and cultivating talent where some don't expect it.

Various MIT staff members, especially Peter Brenton and Rachel Morton, have facilitated completion of my degrees. I needed a lot of facilitation. Also, David Randall is awesome.

Jeff Favorite of Los Alamos has been a priceless source of experience with respect to boundary perturbation theory. He probably knows more than anyone else on that topic. Mike Smith of Argonne seems to know *everything* about DIF3D and VARIANT and frequently goes the extra mile to help. Travis Trahan's and Professor Larsen's papers on anisotropic diffusion were instrumental for deciphering old Soviet literature. Professor Ganapol always responded to my questions about analytical diffusion solutions.

Various "nuclear hipsters" at TerraPower have been instrumental in terms of completing this thesis and, of course, agreeing to fund the entire project in the first place. Nick Touran

has been excessively helpful and patient with respect to helping me learn and use ARMI and MEPT. Pavel Hejzlar is a constant source of positive, affirming energy. Robert Petroski, Jesse Cheatham, and Josh Walter have also been quite helpful and positive. Also, Nick Horelik originally constructed the FFTF and Jōyō diffusion benchmarks as a TerraPower intern. I would have perished in the process of copy-pasting number densities from PDFs.

I must also thank my current and former fellow students in the wider nuclear community who have helped my research or made me feel more welcome. These include Matt Gidden, Mark Norsworthy, Lenka Kollar, Rachel Slaybaugh, Erin Dughie, Katy Huff, Tomi Akindele, Mahima Gupta, Nick Thompson, Wes Deason, and Will Sames. Furthermore, Nate Gibson, Katia Paramonova, and Sam Brinton ran a tremendous ANS student conference (loosely) based upon my original proposal. The estimable polymath (and polyglot) Mike Short labored over filming and editing for the "I'm a Nuke" Project, even while embarking on his assistant professorship. Tim Lucas created the capstone "I'm a Nuke" video while adrift somewhere near southeast Asia. Vlad Sobes and Nazar Lubchenko translated some Russian. Bryan Herman and Jeremy Roberts helped me learn PETSc/SLEPc and the art of preconditioning. Will Boyd, Lulu Li, and Sam Shaner graciously ran a few OpenMOC cases. Justin Ball and Will Boyd assisted me as graders (doing all the tedious work) for 22.101 in Fall 2012.

I would not be where I am today without good friends in Seattle, Boston, and around the world. In particular, Rong Chen, Chris Lumry, Alex Mieloszyk, Brian Myhre, and Karlen Ruleman have been wonderfully supportive during my years in graduate school. My fellow MIT NSE students Youho Lee, Sam Brinton, Tim Lucas, Rosie Sugrue, Mareena Robinson, Matt Everson, Sam Shaner, Giancarlo Lenci, Sasha Tan-Torres, Ethan Bates, and Jude Safo have always been kind to me.

My family in Yakima, Washington and Ewa Beach, Hawaii has always been extremely supportive regardless of what choices or mistakes I made. Although there have been many talented engineers in my family, I am the first to earn an engineering degree. My maternal grandfather quit school at sixteen to work construction on the Bay Bridge. My father grew up in poor, dysfunctional circumstances and spent forty years at sea. My mother, who is brilliant, devoted herself to raising my brother and me. I am indebted to their sacrifices, and my accomplishments rest upon their shoulders.

Biographical Note

Mark Wilbert Reed was born on April 1, 1987 in Bellevue, Washington. He was raised in the semirural outskirts of Woodinville, which holds an annual parade to commemorate his birth. His family later moved across the mountains to Yakima, his current official residence. His father was a chief engineer in the merchant marine, and his mother was a teacher and homemaker. He has an older brother, ten years his senior, who studied biology and works as an intelligence officer in the U.S. Navy. His childhood was spent tramping around the Cascade Mountains and Pacific Northwest hinterlands with his father and brother - hiking, camping, canoeing, mining, shooting, and pyrotechnics.

In 2005, he graduated as a valedictorian of Bellevue Christian High School, where he taught himself just enough math, physics, biology, and history to be named an Advanced Placement National Scholar largely via independent study. He also participated in various state competitions for math, music (bass clarinet), cross-country, and track.

He enrolled at MIT in the fall of 2005 and chose to pursue dual degrees in Physics and Nuclear Science and Engineering (NSE). After briefly dropping out and moving to California in 2008, he returned to complete an S.B. degree in Physics (focused option), complete S.B and S.M. degrees in NSE, and pass the NSE doctoral qualifying exam after a total of ten semesters at MIT.

Originally interested in physics and nuclear fusion, he transitioned into nuclear fission during his first and second years of graduate school. He returned to Washington in 2011 for an internship at TerraPower, a mile from his birthplace. He continued to work as a consultant for TerraPower, which funded this doctoral thesis work. During graduate school, he served as a teaching assistant four times, authored the winning proposal for the 2013 American Nuclear Society (ANS) Student Conference, represented MIT on the Nuclear Engineering Student Delegation three times (including once as a vice-chair), published a series of six essays on the history of nuclear technology, served one term on MIT's Graduate Student Council, and helped run the MIT NSE Nuclear Information Hub in the wake of the Fukushima earthquake and tsunami disaster.

He suffers from Tourette's syndrome and has endured incidents of public humiliation, reputation-ruining slander, and misunderstandings with near-cataclysmic consequences. He has persevered through years of negativity and stigma from his peers and occasionally even "mentors". Shame and false guilt are terribly destructive mindsets, but a person can overcome them through grit, character strength, and a sense of purpose.

For Dad

Contents

List of Tables

List of Figures

List of Common Terms

An **anisotropic** enlargement of a core zone in direction i changes its shape such that only the i component of each linear dimension $\vec{r}$ is scaled up by the same factor: $\vec{r'}_i = f\vec{r}_i$.

Boundary perturbation theory (BPT) is a surface integral expression derived in the early 1980s to predict the first order reactivity due to movement of a material interface.

Classic perturbation theory is a first order material density perturbation computed via unperturbed real and adjoint fluxes and operators. This is the most widely used form of perturbation theory in neutronics.

An **expansion** is an enlargement of a core zone such that density is conserved and mass increases.

First order perturbation theory is any perturbation technique that correctly predicts the first derivative of eigenvalue (a reactivity coefficient) with respect to a given perturbation.

An **isotropic** enlargement of a core zone preserves the precise shape of that zone such that all its linear dimensions $\vec{r}$ are scaled up by the same factor: $\vec{r'} = f\vec{r}$.

A **non-uniform** swelling or expansion scales up linear dimensions by different factors throughout the same core.

Second order perturbation theory is any perturbation technique that correctly predicts the first and second derivatives of eigenvalue with respect to a given perturbation. This is a subset of generalized perturbation theory and variational methods.

A **swelling** is an enlargement of a core zone such that mass is conserved and density decreases.

A **uniform** swelling or expansion scales up all linear dimensions by the same factor throughout a whole core.

"Virtual density" theory (VirDenT) encompasses both a new perturbation technique and a new set of generic neutronic principles. A VirDenT perturbation can be a swelling or an expansion, uniform or non-uniform, and isotropic or anisotropic.

List of Common Variables

This list is not inclusive, but it serves as a reference for certain variables that are defined once and used extensively throughout this thesis. We omit variables which are ubiquitous throughout the literature, such as cross-sections and other nuclear data. We use $\hat{x}$ to represent an operator (a matrix), $\vec{x}$ to represent a vector, and $x^\dagger$ to represent any adjoint quantity. Non-adjoint scalar quantities have no adornment.

Γ a spatially-dependent scalar value representing reactivity due to surface leakage.

Γ_i a spatially-dependent scalar value representing reactivity due to surface leakage in direction i.

$\Delta\rho$ the reactivity due to a small perturbation, computed via a reference eigenvalue calculation or via first order perturbation theory.

$\Delta\rho''$ the second order reactivity, computed via second order perturbation theory.

ϵ the "magnitude" of a swelling or expansion perturbation, usually equal to $f - 1$.

ϕ, $\phi^\dagger$ the scalar real and adjoint fluxes.

$\hat{A}$ the absorption operator in multigroup diffusion.

$\hat{F}$ the fission operator (or "neutron source" operator) in multigroup diffusion.

f the fraction by which a core region swells or expands in one or more directions. This is either slightly greater than or less than unity for small perturbations.

$\vec{J}$,$\vec{J}^\dagger$ the vector real and adjoint currents.

k equivalent to k_{eff}, the fundamental eigenvalue. This usually represents an unperturbed eigenvalue, while k' represents a perturbed eigenvalue.

$\hat{L}$ the leakage operator in multigroup diffusion.

L a spatially-dependent scalar value representing reactivity due to volume leakage.

L_i a spatially-dependent scalar value representing reactivity due to volume leakage in direction i.

$\hat{M}$ the "neutron loss" operator in multigroup diffusion. This includes absorption, leakage, and scattering.

N a material or nuclide density.

S a spatially-dependent scalar value representing reactivity due to "spectral" physics: fission, absorption, and scattering.

$\hat{T}$ the scattering (or "transfer") operator in multigroup diffusion.

1 Motivation and Overview

1.1 Chapter Abstract

Nuclear energy occupies an essential niche in our society, and it is necessary for future sustainability as the only source of energy that is base load, scalable, and carbon-free. However, truly sustainable deployment of nuclear reactors will likely require fast reactors to complement the current light water paradigm. In particular, breed-and-burn sodium fast reactors (SFRs) offer a unique combination of sustainability, fuel efficiency, and power density. Paramount to the wide deployment of these reactors is the assurance of their neutronic safety, which is encapsulated by reactivity coefficients. Unfortunately, large breed-and-burn SFRs suffer from positive sodium void worth. In order to mitigate this drawback, we must accurately quantify the negative reactivity feedback due to geometry distortions - assembly bowing and flowering. These distortions arise mainly from three distinct physical phenomena: irradiation swelling, thermal swelling, and seismic events.

1.2 Nuclear Energy

Seven decades ago, beneath some old bleachers in Chicago, we brought nuclear technology into the world. A few of our finest physicists piled up some blocks of uranium and graphite, and they changed society forever [167].

Few technologies progress from scientific discovery to practical application as rapidly as nuclear fission. The bomb dropped less than seven years after Meitner and Frisch discovered fission [166] and about thirteen years after Chadwick discovered the neutron [165].

Fortunately, nuclear technology did not end in violence. After Truman signed the Atomic Energy Act [168], a twenty-year global nuclear frenzy ensued. Scientists and engineers from around the world proposed all sorts of exotic designs for every conceivable application. However, none of this caught on in the private sector [170].

In the 1950s, Admiral Hyman G. Rickover chose to design and develop a reactor cooled by light water and fueled by uranium oxide (a solid ceramic) clad with a zirconium shell [169]. After five years of work, the world's first nuclear-powered submarine, the USS *Nautilus*, was commissioned and launched. This was an astounding feat, as the design methods and physics data that today's nuclear engineers require simply did not exist yet. Furthermore, the reactor was nestled within a submarine hull twenty-eight feet in diameter, while other reactors built in the 1950s were as large as city blocks. The USS *Nautilus* demonstrated her ability to remain submerged for exceptionally long periods of time by becoming the world's first vessel to sail beneath the North Pole, traversing the polar ice cap from the Pacific to the Atlantic [171].

Rickover's light water reactor was not only superb for naval vessels - it was also amphibious. Four years after the USS *Nautilus* set sail, Rickover oversaw completion of a land-based light water reactor: the Shippingport Atomic Power Station in Pennsylvania. It was connected to the electric grid, making it "the world's first full-scale atomic electric power plant devoted exclusively to peacetime uses" [172].

Terrestrial light water reactors (LWRs) quickly sprung up around the world and remain dominant across the globe. Unfortunately, they are pressurized and thus susceptible to loss of coolant accidents (LOCAs). After a few accidents, most infamously the Three Mile Island (TMI) incident in 1979, much of the world (save France and a few other nations) largely abandoned efforts to pursue nuclear energy programs.

Recently, however, the paradigm has shifted. The ascendancy of climate change as a salient political issue has been a major boon for nuclear energy. The 2005 Energy Policy Act authorized loan guarantees for potential new nuclear plants to help offset their large capital cost [173]. As of 2011, two new reactors are under construction in Georgia, the first built in the U.S. in over 30 years. Two more will soon be underway in South Carolina. Around the world, 109 commercial reactors are either under construction or on order as of December 2010. These new reactors, mostly in Asia, will increase worldwide nuclear energy production by 28% [174].

Ultimately, the world has realized that nuclear energy is the only source of electricity that is base load, scalable, and carbon-free, the three main criteria necessary for a sustainable energy future. Base load energy supplies are constant in time. Solar energy, which oscillates daily with the sun, and wind energy, which vacillates intermittently with the breeze, cannot provide constant power supply. Tidal energy, like solar energy, is diurnal. Even biomass energy (produced from living or recently living biological material) is often seasonal. Fossil fuels, such as coal, petroleum, and natural gas, are base load but not carbon-free - if the current consensus is correct, they exacerbate climate change. Many experts argue that biomass also shares this drawback. Scalable energy sources can be "scaled up" or multiplied without restriction to meet any level of electricity demand. Hydroelectric and geothermal energy, although base load and nearly carbon-free, are not scalable. There are only so many suitable dam sites, and there are only so many suitable geologic "hot spots". Thus, nuclear energy stands alone as a triune of sustainability - base load, scalable, and carbon-free.

Furthermore, nuclear energy is unique in that it is almost entirely geographically independent. The performance and viability of a wind or solar plant depends strongly on its location - wind doesn't blow steadily everywhere, and the sun doesn't shine brightly everywhere. Geography severely restricts hydroelectric, geothermal, and tidal energy - suitable sites are few and far between. Although we can build a fossil fuel plant anywhere, building it far from the fossil fuel resources is disadvantageous due to the costs and logistics of transporting large volumes of those fossil fuels over long distances. Consequently, we build most fossil fuel plants near natural deposits of fossil fuels. In contrast, nuclear fuel is so small in volume that its transportation costs are negligible. There is no advantage to building nuclear plants near uranium mines. The only geographic constraint on nuclear plant sites is that they must be proximate to a body of water (such as a river, lake, or ocean) for cooling purposes. However, even the absence of a natural body of water does not preclude a safe nuclear plant. For example, the Palo Verde Nuclear Generating Station operates in the Arizona desert by extracting water from a sewage treatment facility. Thus, nuclear energy is geographically independent in that its location does not affect its performance. It's the same everywhere. It's an excellent option for regions poor in natural resources.

Despite the perils of the 2011 Fukushima crisis and the even greater perils of political opportunism in reaction to it, the "nuclear renaissance" is still well underway. Unlike thirty years ago, the political salience of climate change precludes an *en masse* return to fossil fuels.

If the "nuclear renaissance" fails to be realized in the U.S., the reason likely won't be Fukushima, the green movement, or the waste conundrum. Instead, it will be economics. Currently, natural gas prices are low and have deterred utilities from constructing new nuclear plants. However, while the price and availability of uranium is stable, natural gas prices are volatile. In this vein, one recent study concludes that nuclear energy has lower long-term investment risk than natural gas [185]. Although natural gas is cheaper today, its volatility creates large uncertainties later in plant lifetime.

For more discussion on these points, see the author's six-part exposition on the history, virtues, and vices of nuclear technology [176,177,178,179,180,181]. This exposition grew out of an article written in MIT's *The Tech* in wake of the Fukushima crisis in the spring of 2011 [175]. Later, this six-part series was the basis for a talk given by Eric Loewen (2011-2012 President of the American Nuclear Society) at the National Academy of Sciences [182]. It was also the subject of a presentation at the 2012 American Nuclear Society Winter Meeting [183] and subsequently featured in the January 2013 edition of *Nuclear News* [184].

1.3 The Art of Reactor Design

The world's first man-made nuclear reactor was the Chicago Pile - Fermi's crude heap of uranium and graphite blocks. Subsequently, the Manhattan Project covertly constructed the world's first large-scale reactor at the Hanford Site in the deserts of eastern Washington State. Its purpose was to breed plutonium for the bombs. The next natural step was to design and develop a large-scale reactor optimized to produce a long-term steady flow of electricity.

The two most important aspects of nuclear reactor design are the choices of fuel and coolant. The fuel must contain a fissionable material - some combination of uranium, plutonium, or thorium - but it can take on just about any chemical form. It can be solid or liquid. If solid, it can be metal or ceramic. If liquid, it can be a molten salt or an aqueous solution. The coolant's primary purpose is to transport energy from the fuel to a turbine to generate electricity, but it also (sometimes incidentally and sometimes purposefully) serves to modify the behavior of the neutrons so that they can more or less readily spur fission in the fuel. Coolants can be liquids or gases. The wide range of coolants proposed or actually incorporated into real reactors includes light water, heavy water, sodium, sodium-potassium, lead, lead-bismuth, various molten salt mixtures, various organic compounds, helium, carbon dioxide, and even mercury. Each of these coolants has advantages and disadvantages in terms of its thermal and chemical properties, and each constrains the neutron behavior and thus the fission reaction and fuel composition. The fuel and coolant are interwoven such that nuclear design is an intricate process of weighing and optimizing the multifaceted interactions between the two.

Thus, reactor design is an art. There's never a "right" answer. Although there are certainly objective metrics by which some designs can be deemed infeasible or unsafe, choosing among a plethora of feasible and safe designs is a subjective process. One's choice depends on one's weighting of multifaceted design goals.

1.4 Fast Reactors

Nuclear reactors can be subdivided into two overarching classes: fast reactors and thermal reactors. Thermal reactors are based on an incongruity: while fission produces high-energy neutrons, low-energy neutrons can much more readily spur fission. Thermal reactors resolve this incongruity by slowing or moderating the high-energy neutrons so that they become low-energy neutrons.

Fast reactors don't bother resolving this incongruity - they simply obviate it by enriching the content of heavy metal that is highly susceptible to fission. Figure 1.1 shows cumulative distribution functions (CDFs) for the fission reaction rates in typical fast and thermal spectra. Figure 1.1 encapsulates the fundamental distinction between thermal and fast reactors. Since fast reactors contain no moderating material to slow their neutrons, they contain exotic coolants consisting of heavier elements such as sodium, lead, or lead-bismuth. They also offer a distinct set of advantages.

First, sodium has unparalleled thermal properties, which are advantageous for core power density. In LWRs, the coolant serves two separate purposes - the thermodynamic purpose of cooling the reactor, and the neutronic purpose of moderating the neutron spectrum. In contrast, fast reactor coolants serve only a thermodynamic purpose. Although these coolants certainly have neutronic *effects*, those effects are not their *purpose*. Therefore, fast reactor pin lattice design depends primarily upon thermal properties, not neutronics. Thus, sodium fast reactor (SFR) fuel pins are closely packed in hexagonal/triangular arrays to achieve high power density.

Second, fast reactors can produce waste that is less hazardous and less long-lived. Fast neutrons can more readily fission many of the most hazardous long-lived actinides, transmuting them into fission products.

Third, fast reactors can breed substantial amounts of fissile material. They can convert fissionable ^{238}U into fissile ^{239}Pu and fissionable ^{232}Th into fissile ^{233}U. Sometimes, fast reactors can breed even more fuel than they burn, and this scenario allows for far more efficient utilization of the world's uranium and thorium resources.

The main disadvantage of fast reactors is that they simply don't have as much operational and design experience as LWRs. Their exotic coolants often possess inconvenient chemical properties. For example, sodium reacts exothermically with air and violently with water. It is also opaque, complicating fuel shuffling and maintenance procedures. One cannot visually inspect the core when it is immersed in a sodium pool. Futhermore, many fast reactor designs cope with significant thermal hydraulic and materials challenges [141].

Nearly all operating commercial reactors worldwide today are thermal reactors. Most of those are LWRs - either pressurized water reactors (PWRs) or boiling water reactors (BWRs). This status quo originated from Rickover's original choice of pressurized water coolant, uranium oxide fuel, and zirconium alloy fuel cladding [169]. Rickover built one SFR submarine (the USS *Seawolf*) but later rejected the concept due to superheater troubles and concerns over sodium reacting with seawater. Nearly the whole commercial nuclear industry followed his lead in adopting LWRs, because they wished to take advantage of previous construction and operational experience from both submarines and the Shippingport reactor. Rather than risk investment in an untried technology, industry leaders pursued what had already been done. Consequently, the nuclear industry chose LWRs for *terrestrial* deployment based on Rickover's original choice for *marine* deployment.

Thus, since widespread adoption of LWRs was a choice of convenience, it is plausible that other reactor types (such as fast reactors) could be far more advantageous in terms of both safety and long-term sustainability.

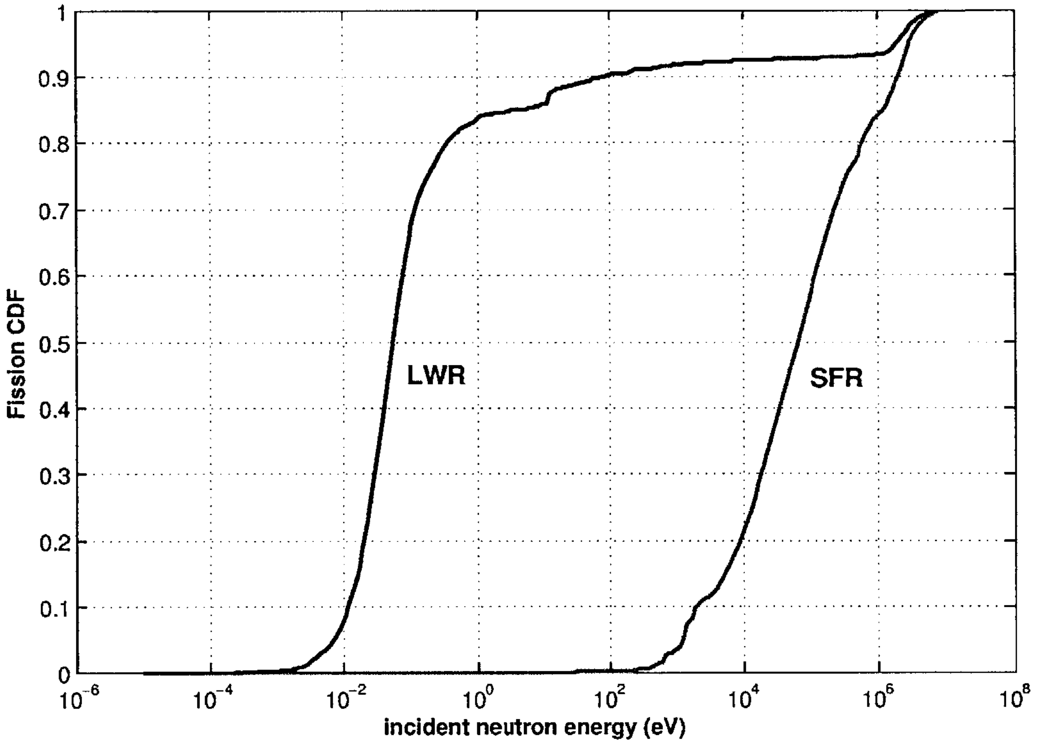

Figure 1.1: Cumulative distribution functions (CDFs) for the fission reaction rates in typical fast and thermal spectra.

1.5 Breed-and-Burn Fast Reactors

Let us consider which reactor type might be most advantageous in terms of long-term sustainability and efficient use of uranium and thorium resources. LWRs fission only a very small fraction of actinides and designate the remainder as waste. Although LWRs have served us well, they are clearly neither the most efficient nor sustainable choice.

An ideal nuclear fuel cycle would utilize 100% of uranium and thorium resources and fission 100% of actinides to leave a waste form that is purely fission products. Unfortunately, this objective is not possible to meet with pure fission reactors, because some actinides must always remain in order for the reactor to maintain criticality. Much more limiting than criticality, however, is the practical reality that solid fuel forms cannot withstand burnups beyond about 40%. In order to burn more actinides, one must reprocess spent fuel outside the reactor - a task that is both expensive and inherently vulnerable to proliferation.

Molten salt reactors (MSRs) come closest to this ideal fuel cycle. As MSR fuel is dissolved in a liquid salt, MSRs are not limited by the structural or chemical performance of any solid fuel form. Thus, they can achieve exceptionally high burnups. However, they still cannot fission *all* the actinides, because they must retain enough to maintain criticality. Furthermore, much skepticism surrounds MSRs due to their lack of conventional solid fuel forms as well as chemistry and materials challenges related to tritium containment [130]. Wide deployment of MSRs would require extensive research and testing in order to resolve these issues.

Fission-fusion hybrids have been touted as another option. Since these rely on a high-energy neutron source produced by fusion, they can operate subcritically and without any initial fissile enrichment [134]. Also, since they need not achieve criticality, they can (at least in theory) fission every last actinide. See Lidsky's comprehensive and prescient review of fission-fusion hybrid concepts [119]. Unfortunately, there is some "trouble with fusion" [122].

Given these challenges and doubts regarding MSRs and fission-fusion hybrids, much fuel cycle innovation research has centered around conventional SFRs operating as "breeders". These SFRs take ^{235}U and ^{238}U as fuel and breed ^{239}Pu. When spent fuel is removed from the core, the fissile ^{239}Pu is reprocessed as fuel for a host of other reactor types - including LWRs. In this way, the energy potential of the world's uranium resources can be multiplied by nearly 100 over what is possible with a fleet of pure LWRs. Unfortunately, reprocessing is stigmatized in the U.S. and considered vulnerable to proliferation throughout much of the world. Funding for the Integral Fast Reactor (IFR) design [126] and its prototype the Experimental Breeder Reactor II [123] was cut in the 1994 anti-nuclear deluge.

Fortunately, "breed-and-burn" fast reactors obviate expensive, precarious, and politically-charged reprocessing. First proposed by Feinberg [118], these were studied in-depth by Driscoll in the 1970s [121]. One can begin reactor operation with a small amount of fissile material (^{235}U or ^{239}Pu) and a large amount of fertile (fissionable) material (^{238}U or ^{232}Th), and then one can sustain reactor operation by continually breeding more fissile material from the fertile material. Eventually, one can extract energy from a very large fraction of the fissionable material. Ultimately, burnup is limited by mechanical fuel performance (typically between 30% and 40% of fissionable atoms), but a large core can still hold enough fertile material to potentially run for decades without refueling.

The Traveling Wave Reactor (TWR) is a type of fast reactor that breeds and burns plutonium in a "traveling" or "standing" wave within depleted uranium [133,135]. Only an minimal initial ^{235}U loading is necessary to achieve and maintain criticality. In order to maximize power density and consequently minimize capital cost, most TWR designs choose sodium coolant over lead or lead-bismuth eutectic. TWRs promise the most sustainable fuel cycle and fuel utilization options. A global fleet of TWRs could "generate electricity for 10 billion people at United States per capita levels for million-year time-scales" [135].

1.6 Fast Reactor Safety

As we have said before, fast reactor coolants have no neutronic purpose - only some unfortunate neutronic "side effects". Thus, they do not share the inherent *neutronic* safety of LWRs in loss-of-coolant accidents and various temperature transients (LWR safety revolves around cooling after depressurization, not neutronics). Thus, fast reactor designs must ensure that the core is "stable" during operation and plausible accident scenarios.

1.6.1 Reactivity Coefficients

Reactivity coefficients serve as metrics for a core's neutronic safety. We prefer the net temperature reactivity coefficient, the derivative of reactivity with respect to temperature, to be negative. Thus, when a small temperature excursion occurs, the core power will not multiply due to positive feedback. This temperature reactivity coefficient has several constituent parts, and we present the main ones here.

- The **fuel density reactivity coefficient** is the derivative of reactivity with respect to fuel density. Fuel swelling as a function of temperature is required input.
- The **structure density reactivity coefficient** is the derivative of reactivity with respect to the density of various structural materials - mostly pin cladding and assembly ducts. Structure swelling as a function of temperature is required input.
- The **coolant density reactivity coefficient** is the derivative of reactivity with respect to coolant density. Coolant density change as a function of temperature is required input.

- The **Doppler reactivity coefficient** is the derivative of reactivity with respect to changes in the *micro*scopic cross-sections due to the Doppler effect, manifested mainly via resonance broadening (and thermal scattering in thermal reactors). Note that this is the only reactivity coefficient that involves microscopic cross-sections. All the others can be defined in terms of only the macroscopic cross-sections.

- The **radial swelling reactivity coefficient** is the derivative of reactivity with respect to the core radius. When temperature increases, core structural materials swell in a way that causes the whole core to swell radially by a small factor - typically less than 1%. Nevertheless, experimental data from the Fast Flux Test Facility (FFTF) has shown that this small radial swelling is often the dominant reactivity feedback in fast reactors [108].

- The **axial swelling reactivity coefficient** is the derivative of reactivity with respect to the core axial height. When temperature increases, fuel swells axially, increasing the core height in the high-power region. Realistically, different assemblies (especially different assembly *types*) swell axially by different factors.

- The **geometry distortion reactivity** is the reactivity "inserted" by arbitrary geometry distortions within the core. Unlike radial and axial swelling, these distortions are non-uniform and often localized. Furthermore, this reactivity cannot easily be quantified as a "coefficient", because there is no quantity with respect to which we can differentiate reactivity. Thus, we typically express it as a reactivity *magnitude*, not a reactivity *coefficient*.

One goal of reactivity safety analysis is to evaluate and amalgamate all these constituent reactivity coefficients to obtain the net temperature reactivity coefficient. Ideally, this should be negative. However, a slight positive temperature coefficient is tolerable if other safety mechanisms exist.

Much effort has been poured into developing fast reactors that are passively safe based on their reactivity coefficients alone. For example, Experimental Breeder Reactor II (EBR-II) was demonstrated to be passively safe in the 1980s [123]. The core is immersed in a pool of sodium, and it swells enough to counteract positive reactivity insertion from various accident scenarios.

However, the EBR-II was really only a prototype for the planned Integral Fast Reactor (IFR), a passively-safe breeder reactor with excellent waste and non-proliferation features [126]. Unfortunately, funding for the IFR was terminated in 1994.

1.6.2 Sodium Void Worth

Large breed-and-burn sodium fast reactors (SFRs) have been touted as a means to achieve high fuel utilization and long-term sustainability. An initial loading of depleted uranium breeds plutonium, eventually achieving an "equilibrium" burnup level. Although such breed-and-burn SFR designs face design challenges in terms of materials, the most salient *neutronic* challenge is positive sodium void worth. As sodium coolant voids, the spectrum hardens slightly, and reactivity increases. Although *small* test reactors such as the EBR-II and the Fast Flux Test Facility (FFTF) [146] had negative sodium void worth, the issue remains a great challenge for *large* reactors with less leakage.

Physically, there are two main mechanisms for sodium void worth reduction: increased leakage and spectrum softening. Unfortunately, neither mechanism is viable in the case of large, breed-and-burn sodium fast reactors. Here we will discuss why, and we will also discuss two other potential strategies for mitigating void worth - thorium fuel and lead coolant.

Increasing leakage certainly reduces sodium void worth. Cores with "pancake" or "parfait" (axially multilayered) geometries have been shown to obviate the void worth problem [124,132]. Unfortunately, when neutrons are more likely to leak, they are less likely to breed. Thus, high-leakage geometries are ruinous for breeding and thus not suitable for large breed-and-burn cores.

Spectrum softening also reduces sodium void worth. Strategically inserting small quantities of moderating material into a large SFR core can soften the neutron spectrum enough such that sodium voiding does not significantly harden the spectrum or increase reactivity [125,128]. Unfortunately, this spectrum softening also results in power peaking and intolerably high equilibrium burnup.

Substituting a uranium-plutonium breed-and-burn cycle for a thorium-uranium cycle would also mitigate the void worth problem. Let $\eta = \nu\Sigma_f/\Sigma_a$ be the average number of neutrons produced (via fission) per neutron absorbed in the fuel. ^{233}U has a lower value of $d\eta/dE$ than ^{239}Pu in the fast spectrum energy range. Figure 1.2 illustrates this. Thus, when sodium voids and the spectrum shifts to higher energy, a reactor fueled primarily by ^{233}U will produce less reactivity than a reactor fueled primarily by ^{239}Pu. Furthermore, ^{233}U also has a higher delayed neutron fraction than ^{239}Pu, which further enhances safety. See Table 1.1. Unfortunately, metallic thorium fuel has a much lower density than metallic uranium fuel, and it also causes an intolerably high equilibrium burnup [139].

Table 1.1: Delayed Neutron Fraction β in a Fast Spectrum [163]

nuclide	β
^{232}Th	0.0203
^{233}U	0.0026
^{235}U	0.0064
^{238}U	0.0164
^{239}Pu	0.0020
^{240}Pu	0.0029
^{241}Pu	0.0054

Substituting sodium coolant for lead or lead-bismuth eutectic (LBE) coolant would certainly mitigate coolant void worth for two main reasons: (1) lead and LBE have high boiling points such that voiding is extremely unlikely and (2) lead and bismuth have relatively low neutron cross-sections and are heavy nuclides such that neutrons lose little energy per elastic scatter. Unfortunately, lead and LBE do not allow for power densities as high as sodium due to thermal properties and material limitations [143,140]. This presents a significant economic disadvantage in terms of capital cost. Recent work at MIT has shown that a new corrosion-resistant steel alloy could allow for dramatic increases in LBE flow rate (and thus core power density), but development is still underway [144].

Thus, sodium void worth remains the greatest *neutronic* challenge of large breed-and-burn SFR cores. There exist few known mechanism to *passively* mitigate this void worth. Various techniques involving absorber/poison insertion via gas expansion modules have been studied, but time scales are often too slow. Only *active* safety mechanisms, such as control rod insertion, can truly solve the problem. Fortunately, many negative reactivity coefficients (such as metallic fuel swelling and the Doppler effect) exist to counter sodium voiding such that the overall temperature coefficient can remain safely negative. However, any large fast reactor design will likely be wanting of negative reactivity feedback wherever it can be found.

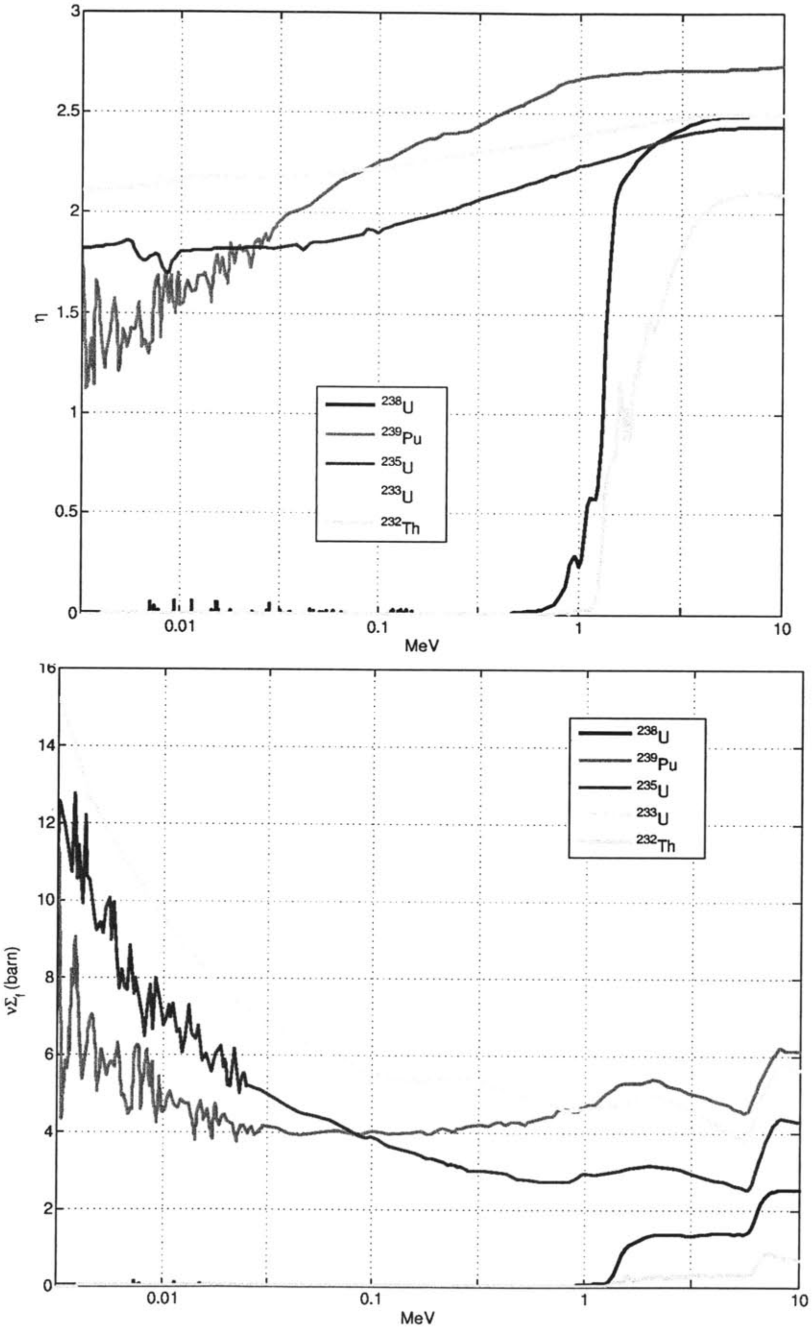

Figure 1.2: η and $\nu\Sigma_f$ as a function of energy for ^{233}U, ^{235}U, ^{238}U, ^{239}Pu, and ^{232}Th. This illustrates the fundamental reason why fast reactors often have positive coolant void reactivity coefficients. When coolant voids, the neutron spectrum hardens slightly, which increases the fission reacton rate. The slope $d\eta/dE$ is larger for ^{239}Pu than for ^{235}U or ^{233}U.

1.7 Geometry Distortions

As we discussed in the previous section, there is currently no *passive* solution to the sodium void worth problem in large breed-and-burn sodium fast reactors. Thus, anyone designing a large breed-and-burn SFR must meticulously count cents of negative worth wherever they can be found.

One place they can be found is in geometry distortions. This is the "geometry distortion reactivity" that we mentioned in Section 1.6.1. When fast reactors overheat, as in a reactivity or temperature transient, the hexagonal assemblies tend to bow or flower radially outward from the core center. This increases leakage and thus induces a negative reactivity feedback. Unlike the radial and axial swelling coefficients, which are uniform, geometry distortions are non-uniform and often localized. In order to count as many negative cents (or potentially dollars) as possible, we must ascertain how much these distortions are worth.

Unfortunately, distortions are notoriously difficult to calculate. As most neutronics calculations map a reactor geometry to a *fixed* computational mesh, capturing the neutronic effects of sinuous assembly shapes and flexuous flowering can be tortuous and toilsome. Of course, we could appraise this worth with Monte Carlo, which allows for arbitrary geometry. Unfortunately, that would be far too slow given the large number of distortions we wish to evaluate, especially considering the tight convergence needed to resolve small reactivities.

Assembly distortion can have significant reactivity effects in any reactor, but these effects are likely more pronounced in metal-fueled fast reactors [127]. Metal fuel tends to thermally swell much more than ceramic fuel, and the relatively low delayed neutron fraction in plutonium-fueled fast reactors means that every nudge in reactivity is "worth" more.

In one sense, geometry distortions are a "double-edged sword" in reactor design. On one hand, distortions greatly complicate reactor modeling and can cause mechanical failure that limits operational lifetime. On the other hand, distortions also provide crucial reactivity feedback to ensure safety.

Geometric distortion arises from three main physical phenomena: irradiation swelling, thermal swelling, and seismic events.

1.7.1 Irradiation Swelling

Irradiation swelling occurs gradually over very long time scales (years). This is an irreversible process. It is slow enough to be considered steady-state at each instant in the reactor lifetime. This swelling can cause significant bowing.

Although metal fuel swells significantly more than ceramic fuel, the neutronic desirability of metal fuel (to achieve an extremely hard spectrum for breeding) is often enough to outweigh any adverse effects due to swelling. Metal-fueled fast reactors must include extra space between fuel and clad to avoid the detrimental consequences of pellet-clad mechanical interaction (PCMI) and pellet-clad chemical interaction (PCCI) [127].

The code NUBOW-3D has been developed to quantify assembly displacements due to irradiation swelling (also known as irradiation "creep") [192]. However, NUBOW-3D does not account for reactivity or power distribution changes due to these irradiation-induced distortions.

1.7.2 Thermal Swelling

Thermal swelling occurs over very short time scales (seconds) during temperature transients. This is often a reversible process, as long as no melting occurs. Thermal swelling also occurs in steady-state due to axial temperature profiles that induce bowing. Figure 1.3 illustrates flowering in a small 19-assembly fast reactor model.

Although core flowering due to thermal swelling is an important negative reactivity feedback mechanism in transients, it can also be undesirable in terms of controlling the reactor. Thus, many fast reactors, including the FFTF, have "core restraints". These are mechanical binding mechanisms that clamp assemblies in place at a certain axial position, often at the core top. Unrestrained cores have a "free flowering" effect, while restrained cores have a "limited free bowing" effect [104].

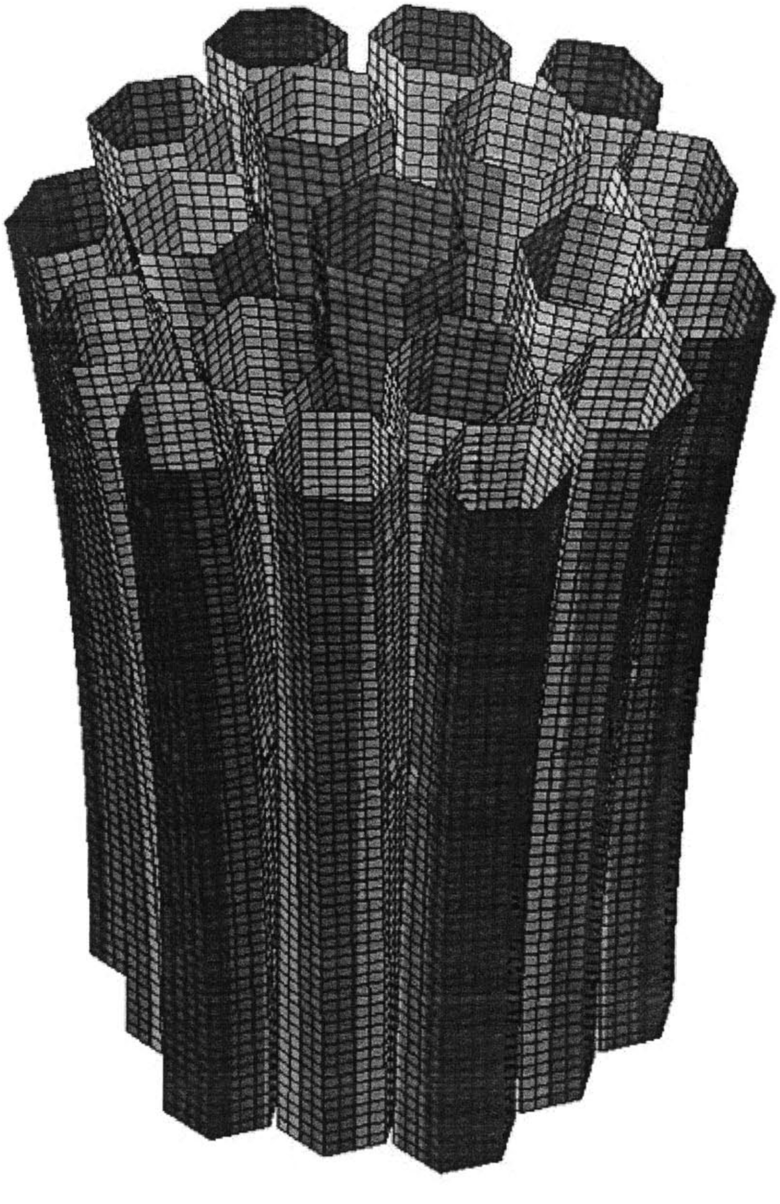

Figure 1.3: Illustration of a fast reactor core flowering scenario.

1.7.3 Seismic Events

Seismic events can also distort pin or assembly geometry. These distortions are distinct from irradiation swelling and thermal swelling in that they oscillate in time - they are rattling vibrations.

In terms of seismic effects, much work has been performed on determining the time-dependent *mechanical* response of core components to seismic oscillations [149,150]. However, very little (if any) work has been done to determine the *neutronic* response to seismic oscillations.

Recent probabilistic risk analysis (PRA) work by Johnson and Apostolakis concludes that seismic risk actually "dominates" SFR designs [151,152]. This risk stems primarily from mechanical failure of plant components, not neutronics. However, because neutronic responses to seismic events have never been quantified, doing so could better inform future PRA work.

Figure 1.4 compares the global distributions of earthquakes and nuclear plants. Most nuclear plants are clustered in three main regions: eastern North America, Europe, and eastern Asia. Two of those three regions tend not to have large earthquakes. However, the third region (eastern Asia) poses significant seismic risk, which manifested itself during the 2011 Fukushima crisis. The resulting salience of seismic analysis within the nuclear community means that *someone* should quantify the neutronic consequences of seismic events. Even though reactors are intended to immediately shut down during an earthquake, we should still know what might occur if that doesn't happen.

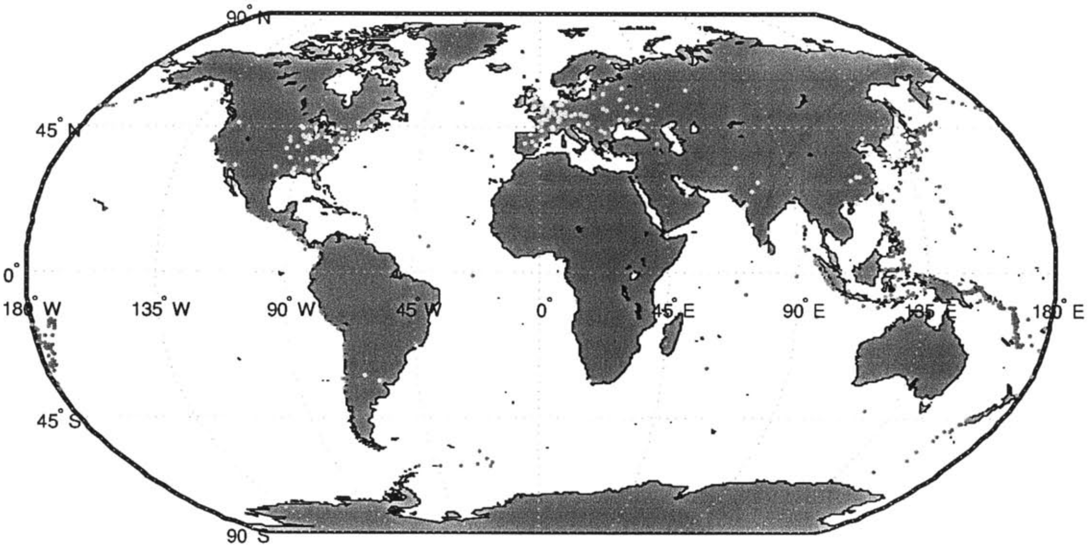

Figure 1.4: The global distributions of nuclear plants and large earthquakes. [illegible] dots represent nuclear plant sites in 2011 [174]. Red dots represent earthquakes of magnitude 7.0 or greater [153].

1.8 Summary and Chapter Outline

Nuclear energy has a rich, painful, and beautiful history. It has an important place in our society, and it is necessary for future sustainability as the only source of energy that is base load, scalable, and carbon-free. However, truly sustainable deployment of nuclear reactors will require fast reactors to supplement or complement LWRs. In particular, breed-and-burn SFRs offer a unique combination of sustainability, fuel efficiency, and power density. Unfortunately, large breed-and-burn SFRs suffer from positive sodium void worth. In order to mitigate this drawback, we must quantify negative reactivity feedback due to geometry distortions.

In this body of work, we aim to develop methods to rapidly quantify reactivity effects due to practical geometry distortions in realistic full-core 3-D fast reactor models. Although we could certainly quantify these effects via Monte Carlo, doing so would be very inefficient (and even impractical) given the vast hordes of histories required.

Instead, we turn to perturbation theory, around which this thesis revolves. Here we present a brief outline for the content of each chapter.

2: A Review of Perturbation Theory in Neutronics

We review perturbation theory throughout Western literature: material density perturbations and boundary perturbations in first and second order. We place particular emphasis on (1) classic first order material density perturbations, (2) second order material density perturbation functionals derived by Stacey, (3) traditional boundary perturbation theory as derived by Pomraning, Larsen, and Rahnema in the early 1980s, and (4) the higher order boundary perturbation techniques in transport theory studied by Favorite during the past decade. Finally, we review the models of Knutson, Wigeland, and Kamal for computing reactivities in fast reactor bowing and flowering scenarios.

3: A Review of Uniform Isotropic "Virtual Density" Theory

We explain the basic concept of "virtual density" in its most basic form: reactivities due to uniform isotropic swellings and expansions of whole cores. Although this concept is well-known within a small segment of the neutronics community, it appears only scarcely throughout the literature. Thus, we take this opportunity to define it formally and provide a few numeric examples. See Section 3.3.2 for explicit definitions of "swelling", "expansion", "isotropic", "anisotropic", "uniform", and "non-uniform".

4: A Review and Independent Derivation of Uniform *An*isotropic "Virtual Density" Theory

We review the Soviet and Russian literature pertaining to the same "virtual density" concept that appears in Western literature. The Russian literature develops the concept much further, to the point of explicitly laying out formalism for uniform *an*isotropic swellings and expansions of whole cores. Since this formalism is obscure in modern, readily-accessible literature, we take this opportunity to define it formally and provide several numeric examples. We perform our own independent derivation via anisotropic diffusion.

5: The *Non*-Uniform Anisotropic "Virtual Density" Theory: An Original Solution to the Problem of Non-Uniformity and a new Theory of Localized Non-Uniform Distortions

We derive an original theory for *non*-uniform anisotropic swellings and expansions. This constitutes the most general form of "virtual density" theory - the prior Russian work applies only to the special case of uniformity. This new theory allows one to compute reactivity coefficients due to anisotropic swellings or expansions of any arbitrary interior zone within a core.

6: The "Virtual Mesh" Method: Generating a "Virtual Reference" for Validating Reactivity Coefficients due to Non-Uniform Geometry Changes in Diffusion Theory

We develop an original technique for validating reactivity coefficients due to geometry distortions in finite difference diffusion theory. This involves constructing a "virtual mesh" and solving the generalized eigenvalue problem (a "virtual reference" case) on that mesh. We validate this technique via Monte Carlo.

7: Numeric Validation of Non-Uniform "Virtual Density" Theory via the "Virtual Mesh" Method

We employ the "virtual mesh" method to numerically validate the non-uniform anisotropic "virtual density" theory for a variety of cases, including simple 2-D and 3-D Cartesian geometries as well as 3-D hexagonal-z full-core benchmarks.

8: Performance of "Virtual Density" Theory Relative to Traditional Boundary Perturbation Theory

We directly compare our original "virtual density" theory to traditional boundary perturbation theory in both analytic and numeric examples.

9: The "Virtual Density" Theory (VirDenT) Code: Practical Distortion Scenarios in Fast Reactors

We develop the "virtual density" theory (VirDenT) code for computing reactivity coefficients due to practical geometry distortions in fast reactors with hexagonal assemblies. The VirDenT code is currently being used in industry, and we validate it for a few full-core benchmarks.

10: "Virtual Density" and Point Kinetics: Reactivity Safety via the Pseudo-Seismic (PseuSei) Animator

We demonstrate how reactivities obtained via "virtual density" theory can be inserted directly into point kinetic theory to show how core power evolves in time due to arbitrary, seismic-like assembly motions in 3-D Cartesian geometry.

11: The VirDenT-MEPT Method: Flux and Power Distributions due to Distortions

We couple "virtual density" theory to modal expansion perturbation theory (MEPT) to obtain perturbed flux and power distributions due to geometry distortions.

12: When Adding Material Increases Leakage: Reconstruction of Current and Adjoint-Weighted Quantities in Fast Reactors

In the process of developing the VirDenT code, we create a pin-level flux reconstruction module. In addition to the real flux, this module can reconstruct adjoint flux as well as real and adjoint current vector fields. Along the way, we notice an intriguing localized effect in which adding material actually increases leakage.

13: Executive Summary and Future Work

We succinctly summarize the entire thesis work with emphasis on crucial results and significance. We also speculate on future work that could stem from "virtual density" theory.

2 A Review of Perturbation Theory in Neutronics

2.1 Chapter Abstract

We review past work and literature on the subject of perturbation theory in neutronics within the Western literature. We focus mainly on (1) classic first order material density perturbation theory, (2) second order variational functionals for material density perturbations derived by Stacey, (3) first order boundary perturbation theory as derived by Pomraning, Larsen, and Rahnema in the early 1980s and (4) variational boundary perturbations in transport theory studied by Favorite during the past decade. Originally developed to derive analytic approximations for equations that have no precise analytic solution, perturbation theory has become less frequently used with the advent of nodal methods and high-performance computing. Howcver, we contend that perturbation theory remains relevant to neutronics. Not only can it expedite the computation of reactivity coefficients by orders of magnitude, but it can save countless manhours in the analysis of geometry distortions in nuclear reactors (no re-meshing is required). Thus, the forward thrust of perturbation theory research should be primarily in the area of geometry perturbations, not material density perturbations. We review past attempts to quantify geometry distortions in fast reactors using perturbation theory, most notably those of Knutson, Wigeland, and Kamal. Finally, we lay out the objective of this thesis work as it relates to perturbation theory.

2.2 A Brief History of Perturbation Theory

Here we briefly describe the origin and history of perturbation theory, which originated in celestial mechanics, experienced its heyday in quantum mechanics, and was later adopted by nuclear engineers.

2.2.1 The Origin of Perturbation Theory: Quantum Mechanics

Perturbation theory arose in the 19^{th} century. In those days, solving differential equations was hard, so mathematicians used perturbation theory to approximate solutions to celestial mechanics - how the earth, moon, and sun move about. If precise equations describing the orbit of a certain celestial body were too cumbersome to solve directly and analytically, a well-versed mathematician could solve a much simpler set of equations that differed only slightly - and that slight difference defined a perturbation.

Perturbation theory attained its zenith with the advent of quantum mechanics. The Schrödinger equation, which governs quantum mechanics, is notoriously difficult to solve analytically for most cases.

$$i\hbar\frac{d\psi(\vec{r},t)}{dt} = -\frac{\hbar^2}{2m}\nabla^2\psi(\vec{r},t) + V(\vec{r},t)\psi(\vec{r},t) = \hat{H}\psi(\vec{r},t) \tag{2.1}$$

Here $\psi(x,t)$ is the wavefunction, and $V(x,t)$ is the quantum potential [162]. The operator $\hat{H}$ is known as the Hamiltonian. This is a second order partial differential equation, and precise analytic solutions exist only for very few simple potentials. Thus, the great physicists of the early 20th century applied perturbation theory to solve very simple systems exactly, and then they could introduce complexities into those simple systems as perturbations.

For example, consider the well-known Zeeman effect, which describes energy level shifting in the presence of a magnetic field. One can solve the Schrödinger equation for energy levels in a spherical Coulomb potential - electron orbitals surrounding an atomic nucleus - because a spherical potential $V(r) \propto 1/r$ happens to be one of the few Hamiltonians for which the Schrödinger equation has an analytic solution. However, placing that atomic nucleus into a magnetic field causes those energy levels to shift, often breaking degeneracy. When a Hamiltonian includes the extra magnetic field term $-\vec{\mu}\cdot\vec{B}$, it is not possible to solve analytically, and so physicists introduced the magnetic field potential as a perturbation of the Hamiltonian. As long as the magnetic field is not too large, perturbation theory allows one to derive equations that accurately describe the Zeeman effect. This practice is ubiquitous throughout physics today [162].

2.2.2 Perturbation Theory and Variational Methods Applied to Material Densities in Neutronics

Quantum mechanics begat nuclear physics, which in turn begat nuclear engineering. The earliest nuclear engineers were nuclear physicists, and they were well-versed in quantum mechanics. Thus, they knew perturbation theory, and so it was natural for them to apply it to the physics of nuclear reactors. They did not have computers, and they needed methods for computing changes in reactivity due to additions of various materials or neutron sources [154].

The main limitation of basic perturbation theory is that it is linear - it only predicts the *first order* reactivity response due to small changes in system properties. This works well for very small changes or for larger changes that produce very linear effects, but it fails for larger changes that produce non-linear effects. In order to obtain higher order perturbation techniques, variational methods are needed.

Variational methods, often referred to as second order perturbation theory or "generalized perturbation theory", require one to choose an appropriate *functional* - a scalar function of the real and adjoint fluxes integrated over all space and energy. One must "guess" a functional that (1) yields the *exact* desired value when the *exact perturbed* fluxes are input and (2) yields an approximate value that differs from the exact value by $O(\epsilon^2)$ when the *un*perturbed fluxes are input [163]. Here $\epsilon << 1$ represents the small dimensionless magnitude of a perturbation. Guessing an effective functional often requires experience and is somewhat of an "art".

Roussopoulos is credited with pioneering variational methods in neutronics [1], although they were first proposed by Wigner as early as 1945 [26]. These methods can estimate perturbed quantities (such as reactivity) with second order error. While first order perturbation theory correctly predicts the first derivative of reactivity with respect to a perturbation, second order perturbation theory (variational methods) correctly predicts the second derivative of reactivity with respect to a perturbation. Thus, variational methods capture the "curvature" of an reactivity. For an excellent primer on this topic, see Chapter 6 of Bell and Glasstone's textbook [156]. Two of the most well-known variational functionals applied to material density perturbations are the Roussopoulos functional and the Schwinger functional, which are discussed in standard neutronics texts [155,156,163].

In the late 1950s, people derived explicit variational functionals for neutron transport [2]. Throughout the 1960s, 1970s, 1980s, and even the 1990s, much effort was poured into deriving improved variational techniques for diffusion and transport [3,6,14,18,21,24,25,30].

Particularly noteworthy is the work by Pomraning, who laid out much of the general theory [7,12,8,9,11]. Also noteworthy is the work by Cacuci, who aptly characterized the subject as "sensitivity theory" [22,23]. Gandini wrote a wonderful paper in 1987 summarizing the development of variational methods to date, and in that same paper he proposed a new "'heuristic approach" [26]. In the 1990s, Yang and Downar extended the earlier work of Gandini [20] to apply perturbation theory to depletion, employing the transmutation equations [27,28].

Stacey was also quite prolific in this area, writing a tome on the subject [155]. His academic papers focused on applying variational methods to realistic reactor models [15,16,17]. He also wrote the code VARI1D, which implemented these techniques in arbitrary 1-D problems [190]. Much later, other developers created VARI3D, which became widely used [205]. In the 1990s, Stacey revisited variational methods with his student Favorite, and they successfully applied second order perturbation techniques to simple light water reactor (LWR) problems involving flux tilts and point-kinetics [29,31]. Later, Favorite continued in this vein with improved variational functionals to preferentially treat negative or positive reactivities [33].

Since 2000, much less work has been done on variational methods applied to material density perturbations, although isolated papers occasionally appear for particular applications [32].

Note that all the perturbation work we have discussed this far has been limited almost entirely to material density perturbations, which are small changes in material compositions within individual reactor zones. None of these perturbation techniques were designed to capture changes in geometry or boundaries.

2.2.3 Modal Expansion Perturbation Theory (MEPT)

In the late 1970s, Gandini proposed a fundamentally new type of perturbation theory based on explicit computation of the modal flux harmonics [35]. This method was loosely based on a quantum mechanical problem introduced by Mitani [34]. It is often referred to as "explicit high order" and sometimes curiously dubbed "the standard method".

When one solves the neutron diffusion or transport eigenvalue equations deterministically, many eigenvalues exist. If one performs a standard power iteration, one will obtain only the largest eigenvalue and its corresponding eigenfunction - the others exist but rapidly dwindle away. However, if one performs an Arnoldi iteration (an eigenvalue iteration technique based on Krylov subspaces [186]), one can obtain a multitude of eigenvalues and corresponding eigenfunctions (commonly called "modes" or "harmonics"). These sets of eigenfunctions have two important properties: they are *orthogonal* and *complete.* Their orthogonality implies that the inner product (integration over all space and energy) of any non-identical pair of eigenfunctions will be zero. Their completeness implies that *any* function in the neutron flux domain (space and energy) can be expressed as a weighted sum of them. In order to express (or expand) any function in terms of such a weighted sum, one must know the weight (the expansion coefficient) of each. Of course, if *any* function can be expressed, then certainly perturbed fluxes can be expressed! Thus, it is possible to explicitly and precisely determine perturbed flux and power distributions in terms of the complete unperturbed flux harmonics.

Palmiotti first applied this method in diffusion theory to small and large fast reactor models [36]. He successfully computed perturbed flux distributions due to enrichment changes and control rod insertions. Soon after, Moreira and Lee applied the method to a high-temperature gas-cooled reactor (HTGR) model [37,38]. They achieved limited success with very few eigenfunctions. They noted that perturbed fluxes tend to have both "local" and "global" components. A given perturbation will introduce a "tilt" in the core-wide global flux distribution, and it will also produce a highly localized flux change in the immediate vicinity of the perturbation. Moreira and Lee found that while global flux changes were well predicted, local flux changes were not.

As the order of a flux harmonic increases, it tends to vary more rapidly within small spatial regions. The more localized a perturbation, the more harmonics are necessary to capture its effect on the flux distribution. Thus, it is not surprising that this modal expansion method would be much more accurate (and more feasible) for fast reactors than for thermal reactors, as longer mean free paths lead to smoother (and less localized) flux distribution effects.

Nearly a quarter century later, Lee and his student Touran revisited the topic. Touran derived a more general formula for the expansion coefficients and dubbed the process "modal expansion perturbation theory " (MEPT) [39,40]. Since computers sped up quite a bit from 1988 to 2012, Touran was able to compute hundreds of (real and adjoint) flux harmonics for the unperturbed flux distribution. He successfully used these harmonics to accurately compute both global and local perturbed fluxes in realistic, full-core sodium fast reactor (SFR) cases. Although global perturbed flux shapes can be captured with only a few harmonics, hundreds of harmonics are often necessary to achieve high accuracy for more localized effects.

However, Moreira and Lee surmised that a "global-local" or "modal-local" method might work best [38]. This involves using a modal expansion to obtain an accurate global flux shift and then using a direct calculation in the vicinity of a perturbation to obtain the local flux shift. These two can simply be summed to obtain the full "global-local" flux shift.

2.3 A Technical Overview of Material Density Perturbations

Here we show the basic mathematics necessary to apply perturbation theory to material density changes in first and second order. We will refer to these equations continually throughout this thesis document. When we refer to perturbation theory as "classic", we mean any perturbation theory that is applied to material density changes - not geometry changes.

2.3.1 Neutron Balance in Transport Theory

We solve the neutron diffusion and transport equations by conserving (or balancing) neutrons, which must be produced and lost at equal rates. Of course, if the reactor model is not precisely critical, then neutrons will *not* actually balance. However, we account for this by dividing the fission source by the eigenvalue k (the neutron multiplication constant). The neutron transport equation is [157]

$$\begin{aligned}
\hat{\Omega}\cdot\nabla\psi(\vec{r},\hat{\Omega},E) &= -\Sigma_t(\vec{r},E)\psi(\vec{r},\hat{\Omega},E) && (2.2)\\
&+ \int_0^\infty dE'\int_0^{4\pi} d\hat{\Omega}'\Sigma_s(\vec{r},\hat{\Omega}\cdot\hat{\Omega}',E'\to E)\psi(\vec{r},\hat{\Omega}',E') && (2.3)\\
&+ \frac{\chi(\vec{r},E)}{4\pi k}\int_0^\infty dE'\nu(\vec{r},E')\Sigma_f(\vec{r},E')\int_0^{4\pi} d\hat{\Omega}'\psi(\vec{r},\hat{\Omega}',E') && (2.4)
\end{aligned}$$

Here $\psi(\vec{r},\hat{\Omega},E)$ is the angular neutron flux at a certain point $\vec{r}$ with a certain energy E traveling in a certain direction $\hat{\Omega}$. Here $\hat{\Omega}$ is a unit vector representing a single direction out of a total 4π steradians in a sphere. Now let us define a *characteristic* as a direction of travel along $\hat{\Omega}$ at energy E.

The left-hand term in Equation 12.1 is the flux gradient along this characteristic. This represents how many net neutrons are gained or lost along this characteristic. Since we conserve neutrons, this term must be balanced by gains or losses from scattering, absorption, and fission.

The right-hand term in Equation 12.1 represents the total loss of neutrons along this characteristic. The total cross-section $\Sigma_t(\vec{r}, E)$ contains capture, fission, and scattering. In principle, *any* scattering reaction will remove a neutron from this characteristic due to changes in both energy and direction.

The term in Equation 12.2 represents the gain of neutrons from scattering. Neutrons scatter onto this characteristic from some other characteristic with a different energy E' and a different direction $\hat{\Omega}'$. The "transfer function" from one characteristic to another is represented by $\Sigma_s(\vec{r}, \hat{\Omega} \cdot \hat{\Omega}', E' \to E)$, the macroscopic scattering cross-section. Note that the angular dependence of scattering depends only upon $\hat{\Omega} \cdot \hat{\Omega}'$, which is equal to the cosine of the angle between the characteristics.

The term in Equation 12.3 represents the gain of neutrons from fission. $\Sigma_f(\vec{r}, E')$ is the fission cross-section. The quantity $\nu(\vec{r}, E')$ is the number of neutrons produced per fission induced by a neutron of energy E', while $\chi(\vec{r}, E)$ is the fraction of those fission-produced neutrons born at energy E. Thus, the combination of Σ_f, ν, and χ represents a transfer function from E' to E. The factor $1/4\pi$ represents the fraction of fission neutrons (per unit steradian) born traveling in direction $\hat{\Omega}$, because we assume that fission produces neutrons isotropically.

One can solve the transport equation stochastically via Monte Carlo methods, or one can solve it deterministically via a manifold of methods including discrete ordinates (S_N), spherical harmonics (P_N), method of characteristics (MOC), and the collision probability method (CPM) [163,164].

2.3.2 Neutron Balance in Diffusion Theory

The neutron transport equation depends on space, energy, and angle. Consequently, it is challenging to solve deterministically, especially when high resolution of all three variables is desired. One way to simplify the transport equation is to approximate the flux as having linear angular dependence. One can accomplish this by applying the P_N method to the transport equation for $N = 1$. This derivation, which is ubiquitous throughout neutronics curriculum, leads to diffusion theory, in which the flux is a scalar function $\phi(\vec{r}, E)$ of only space and energy [163].

Typically, we solve the neutron diffusion equation by discretizing cross-sections and other nuclear data in space and energy. Let there be a finite number of energy groups G. The index g represents the energy group number, which begins at 1 for the highest-energy group and increases to G, the lowest-energy group. The full multigroup diffusion equation for group g, neglecting up-scattering, is

$$-\nabla \cdot D_g \nabla \phi_g + \Sigma_{a,g}\phi_g + \sum_{g'=g+1}^{G} \Sigma_{s,g\rightarrow g'}\phi_g = \sum_{g'=1}^{g-1} \Sigma_{s,g'\rightarrow g}\phi_{g'} + \frac{1}{k}\chi_g \sum_{g'=1}^{G} \nu_{g'}\Sigma_{f,g'}\phi_{g'} \tag{2.5}$$

Here the left-hand side represents neutron loss, while the right-hand side represents neutron gain. D_g is the diffusion coefficient, which is a function of the transport-corrected total cross-section $\Sigma_{tr,g}$. $\Sigma_{a,g}$ is the macroscopic absorption cross-section, which includes neutron capture and fission. $\Sigma_{s,g\rightarrow g'}$ represents scattering from group g to group g'. $\Sigma_{f,g}$ is the fission cross-section. The quantity $\nu_{g'}$ is the number of neutrons produced per fission induced by a neutron in group g', while χ_g is the fraction of those fission-produced neutrons born in group g. Note that we could also write this same equation in terms of the removal cross-section $\Sigma_{r,g}$, which includes capture, fission, *and* out-of-group scattering.

$$-\nabla \cdot D_g \nabla \phi_g + \Sigma_{r,g}\phi_g = \sum_{g'=1}^{g-1} \Sigma_{s,g'\rightarrow g}\phi_{g'} + \frac{1}{k}\chi_g \sum_{g'=1}^{G} \nu_{g'}\Sigma_{f,g'}\phi_{g'} \tag{2.6}$$

All these group cross-section data and fluxes are functions of space. We typically discretize space in addition to energy. Let there be N discrete spatial regions. Then there are $NG + 1$ unknowns in this multigroup diffusion equation - the global eigenvalue k plus NG multigroup fluxes $\phi_{g,n}$. The natural mathematical approach to solve this is linear algebra. The left-hand and right-hand sides of Equation 2.5 can each be represented by a $NG \times NG$ matrix.

Consider what these matrices will look like for a simple 3-group problem. For convenience, we will collapse the matrices to be $G \times G$ so that each element actually represents an embedded $N \times N$ spatial matrix. The absorption matrix $\hat{A}$ is entirely diagonal, because it involves no transfer of neutrons between different energy groups.

$$\hat{A} = \begin{bmatrix} \Sigma_{a,1} & 0 & 0 \\ 0 & \Sigma_{a,2} & 0 \\ 0 & 0 & \Sigma_{a,3} \end{bmatrix} \tag{2.7}$$

The leakage matrix $\hat{L}$ is also entirely diagonal, because it also involves no transfer of neutrons between different energy groups. If we were considering $NG \times NG$ matrices, then these leakage terms would tridiagonal (for 1-D problems), pentadiagonal (for 2-D problems), or heptadiagonal (for 3-D problems).

$$\hat{L} = \begin{bmatrix} -\nabla \cdot D_1 \nabla & 0 & 0 \\ 0 & -\nabla \cdot D_2 \nabla & 0 \\ 0 & 0 & -\nabla \cdot D_3 \nabla \end{bmatrix} \tag{2.8}$$

The scattering matrix $\hat{T}$ contains out-of-group scattering as diagonal terms and into-group scattering as off-diagonal terms. Note that we still neglect up-scatting.

$$\hat{T} = \begin{bmatrix} \Sigma_{s,1\to 2} + \Sigma_{s,1\to 3} & 0 & 0 \\ -\Sigma_{s,1\to 2} & \Sigma_{s,2\to 3} & 0 \\ -\Sigma_{s,1\to 3} & -\Sigma_{s,2\to 3} & 0 \end{bmatrix} \tag{2.9}$$

Throughout this body of work, we will denote the total "neutron loss" matrix as $\hat{M} = \hat{A} + \hat{T} + \hat{L}$.

$$\hat{M} = \begin{bmatrix} -\nabla \cdot D_1 \nabla + \Sigma_{a,1} + \Sigma_{s,1\to 2} + \Sigma_{s,1\to 3} & 0 & 0 \\ -\Sigma_{s,1\to 2} & -\nabla \cdot D_2 \nabla + \Sigma_{a,2} + \Sigma_{s,2\to 3} & 0 \\ -\Sigma_{s,1\to 3} & -\Sigma_{s,2\to 3} & -\nabla \cdot D_3 \nabla + \Sigma_{a,3} \end{bmatrix} \tag{2.10}$$

Now the fission matrix $\hat{F}$ is typically the least sparse

$$\hat{F} = \begin{bmatrix} \nu_1 \chi_1 \Sigma_{f,1} & \nu_2 \chi_1 \Sigma_{f,2} & \nu_3 \chi_1 \Sigma_{f,3} \\ \nu_1 \chi_2 \Sigma_{f,1} & \nu_2 \chi_2 \Sigma_{f,2} & \nu_3 \chi_2 \Sigma_{f,3} \\ \nu_1 \chi_3 \Sigma_{f,1} & \nu_2 \chi_3 \Sigma_{f,2} & \nu_3 \chi_3 \Sigma_{f,3} \end{bmatrix} \tag{2.11}$$

Assuming access to veracious multigroup cross-section data, all quantities in these matrices are straightforward to compute from material properties. Now that we have defined the matrix operators $\hat{M}$ and $\hat{F}$, we can construct an eigenvalue problem. If $\hat{M}$ and $\hat{F}$ are both $NG \times NG$, then the scalar diffusion flux ϕ is an eigenvector of length NG.

$$\hat{M}\phi = \frac{1}{k}\hat{F}\phi \tag{2.12}$$

This is basic neutron balance in the diffusion approximation. It is a standard generalized eigenvalue problem, and we can solve for it via a standard power iteration, an Arnoldi iteration [186], a Krylov-Schur iteration [201], or many other iteration techniques.

2.3.3 The Importance Function

However, there is a second way to balance neutrons - a "backward" way. We call this the mathematical *adjoint* flux $\phi^\dagger$, and we can solve for it with a modified set of diffusion operators. Determining these modified adjoint operators from the unmodified operators is simple - an adjoint of a matrix is the complex conjugate of the transpose of that matrix, although the complex conjugate operation is unnecessary in neutronics given that nuclear data is never complex. Given different operators, the adjoint flux requires a separate iterative eigenvalue calculation. Both the real and adjoint neutron balances have the same eigenvalue $1/k$. The real and adjoint fluxes are identical for one-group problems but different for multigroup problems. The one-group diffusion equation is self-adjoint (meaning that the real and adjoint operators are identical), while the multigroup diffusion equation is not.

$$\hat{M}^\dagger\phi^\dagger = \frac{1}{k}\hat{F}^\dagger\phi^\dagger \tag{2.13}$$

See Bell and Glasstone's textbook for a thorough conceptual and mathematical explanation of the adjoint flux [156]. Essentially, it reflects the "importance" of a hypothetical neutron with respect to some arbitrary "adjoint source". For example, if we were analyzing a fixed-source detector problem, we could choose the adjoint source to be the detection rate, and then the adjoint flux would reflect neutron importance with respect to detection. However, throughout this body of work, we will always choose the adjoint source to be fission, and so the adjoint flux will reflect neutron importance with respect to the fission rate. This "importance" concept was first articulated by Lewins in the 1960s [4]. If we place a hypothetical neutron with a certain energy at a certain point in space, the adjoint flux reflects how likely that neutron is to contribute to the overall neutron population of the system. Knowledge of the adjoint distribution is essential for proper application of perturbation theory. Once we have defined a perturbation, we must know how "important" it will be.

See Figure 2.1 for a comparison of energy distributions of the real and adjoint fluxes in a typical sodium fast reactor (SFR) spectrum. The real flux is concentrated at high energies, because this is a fast neutron spectrum. In contrast, the adjoint flux is more evenly distributed. If one were to insert a neutron at very low energy, it would have a high likelihood of (sooner or later) spurring a fission reaction. Thus, the adjoint flux is high even at very low energies, where few real neutrons exist.

For completeness, we can write down the adjoint transport equation with a fission source term [157]:

$$\begin{aligned}
-\hat{\Omega}\cdot\nabla\psi^{\dagger}(\vec{r},\hat{\Omega},E) &= -\Sigma_t(\vec{r},E)\psi^{\dagger}(\vec{r},\hat{\Omega},E) && (2.14)\\
&+ \int_0^{\infty} dE' \int_0^{4\pi} d\hat{\Omega}'\Sigma_s(\vec{r},\hat{\Omega}\cdot\hat{\Omega}',E\rightarrow E')\psi^{\dagger}(\vec{r},\hat{\Omega}',E') && (2.15)\\
&+ \frac{\nu(\vec{r},E)\Sigma_f(\vec{r},E)}{4\pi k}\int_0^{\infty} dE'\chi(\vec{r},E')\int_0^{4\pi} d\hat{\Omega}'\psi^{\dagger}(\vec{r},\hat{\Omega}',E') && (2.16)
\end{aligned}$$

Comparing this with the real transport equation (Equations 12.1 - 12.3) shows that one can convert the real transport equation to the adjoint transport equation by performing three simple operations: (1) replacing $\hat{\Omega}$ with -$\hat{\Omega}$, (2) replacing $E \rightarrow E'$ with $E' \rightarrow E$, and (3) switching $\nu(\vec{r},E')\Sigma_f(\vec{r},E')$ and $\chi(\vec{r},E)$. These three operations elucidate why the adjoint flux is often referred to as the "backward" flux. In the adjoint solution, neutrons travel backward in the -$\hat{\Omega}$ direction rather than forward in the $\hat{\Omega}$ direction. Furthermore, neutrons undergo scattering and fission operations backwards.

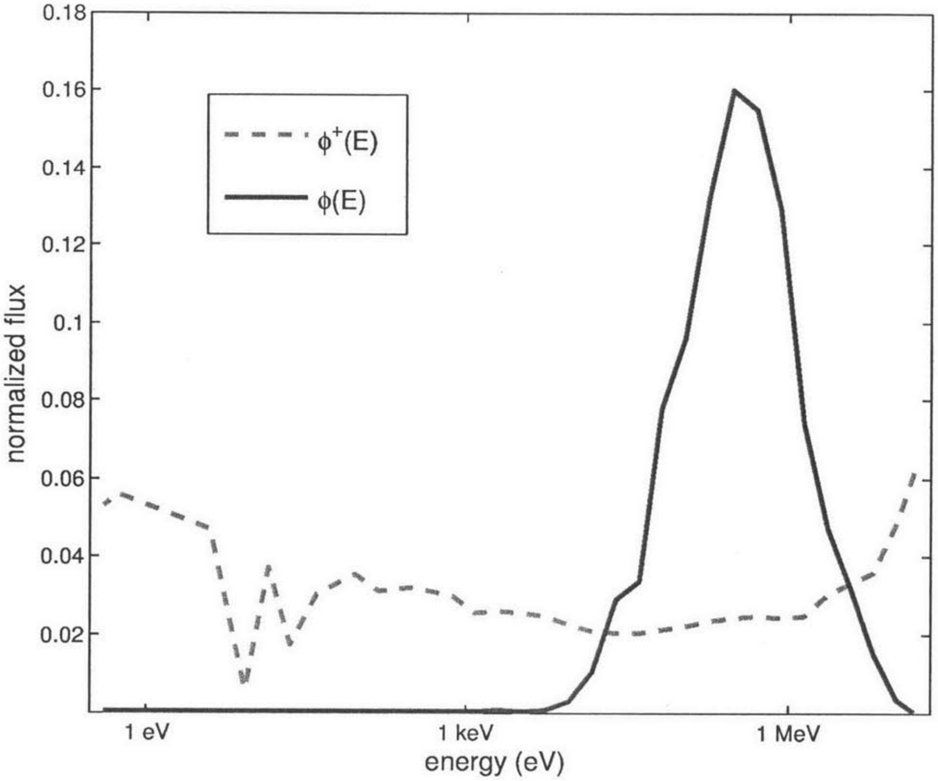

Figure 2.1: Energy distributions of the real and adjoint fluxes in a typical sodium fast reactor spectrum. The real flux is concentrated at high energies, while the adjoint flux is more evenly distributed.

2.3.4 Classic First Order Perturbation Theory

We can perturb both $\hat{M}$ and $\hat{F}$ operators in some arbitrary (small) way. These operators are composed entirely of macroscopic cross-sections and other nuclear data - they are functions of material composition. They are the *input* to perturbation theory. Any material density perturbation (fuel density, coolant density, Doppler, temperature, etc.) can be quantified in terms of them.

$$\hat{M}' = \hat{M} + d\hat{M} \tag{2.17}$$

$$\hat{F}' = \hat{F} + d\hat{F} \tag{2.18}$$

The perturbed neutron balance now includes a perturbed flux ϕ' and a perturbed eigenvalue $1/k'$.

$$\hat{M}'\phi' = \frac{1}{k'}\hat{F}'\phi' \tag{2.19}$$

The goal of perturbation theory is to estimate this perturbed eigenvalue $1/k'$ in order to determine the reactivity $\Delta\rho$ associated with the perturbation.

$$\Delta\rho = \frac{1}{k} - \frac{1}{k'} = \frac{k' - k}{k'k} \tag{2.20}$$

We can express the perturbed flux and eigenvalue in the same fashion. In contrast to $\hat{M}'$ and $\hat{F}'$, these constitute the *output* of perturbation theory. Although $\Delta\rho$ is the primary output that we are most interested in, determining $\Delta\rho$ to greater than first order accuracy will also require knowledge of $d\phi$.

$$\phi' = \phi + d\phi \tag{2.21}$$

$$\frac{1}{k'} = \frac{1}{k} + d\left(\frac{1}{k}\right) = \frac{1}{k} - \Delta\rho \tag{2.22}$$

A simple derivation leads to the first order (linear) change in whole-core reactivity $\Delta\rho$. This first order result is especially convenient in that it avoids computation of the perturbed flux $d\phi$. Only the perturbed operators along with the unperturbed real and adjoint fluxes are necessary.

$$\Delta\rho = \frac{\left\langle \phi^{\dagger} \middle| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi \right\rangle}{\left\langle \phi^{\dagger} \middle| \hat{F}\phi \right\rangle} \tag{2.23}$$

Equation 2.23 is ubiquitous throughout neutronics and derived in most standard textbooks [156,157,163]. It exactly predicts the *first derivative* (or sensitivity) of reactivity with respect to a given perturbation. Thus, it is ideally suited for the computation of reactivity coefficients. The bra-ket notation $\langle a|\, b\rangle$ represents the inner product of vectors a and b integrated over all space and summed over all energy groups.

2.3.5 Second Order Perturbation Theory (Variational Methods)

While first order perturbation theory precisely predicts the *first derivative* of eigenvalue with respect to a perturbation, second order perturbation theory precisely predicts the *second derivative* of eigenvalue with respect to a perturbation. While first order perturbation theory relies on the unperturbed flux, second order perturbation theory accounts for first order changes in the flux distribution.

Employing the same derivation (as in first order) while simply neglecting to truncate the second order terms leads to the second order reactivity $\Delta\rho''$[163] :

$$\Delta\rho'' = \frac{\left\langle \phi^{\dagger}\right| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi\Big\rangle}{\left\langle \phi^{\dagger}\right| \hat{F}\phi\Big\rangle} + \frac{\left\langle \phi^{\dagger}\right| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)d\phi\Big\rangle}{\left\langle \phi^{\dagger}\right| \hat{F}\phi\Big\rangle} - \frac{\left\langle \phi^{\dagger}\right| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi\Big\rangle \left\langle \phi^{\dagger}\right| \hat{F}d\phi\Big\rangle}{\left\langle \phi^{\dagger}\right| \hat{F}\phi\Big\rangle^{2}} \tag{2.24}$$

We can rearrange factors here to express the second order result as the first order result multiplied by a "correction factor". Note that the second and third terms in this "correction factor" are of order $d\phi/\phi$, which will be much smaller than 1 for any small perturbation.

$$\Delta\rho'' = \frac{\left\langle \phi^{\dagger}\right| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi\Big\rangle}{\left\langle \phi^{\dagger}\right| \hat{F}\phi\Big\rangle}\left[1 + \frac{\left\langle \phi^{\dagger}\right| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)d\phi\Big\rangle}{\left\langle \phi^{\dagger}\right| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi\Big\rangle} - \frac{\left\langle \phi^{\dagger}\right| \hat{F}d\phi\Big\rangle}{\left\langle \phi^{\dagger}\right| \hat{F}\phi\Big\rangle}\right] \tag{2.25}$$

The ratio of $\Delta\rho''$ to $\Delta\rho$ is simply

$$\frac{\Delta\rho''}{\Delta\rho} = 1 + \frac{\left\langle \phi^{\dagger}\right| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)d\phi\Big\rangle}{\left\langle \phi^{\dagger}\right| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi\Big\rangle} - \frac{\left\langle \phi^{\dagger}\right| \hat{F}d\phi\Big\rangle}{\left\langle \phi^{\dagger}\right| \hat{F}\phi\Big\rangle} \tag{2.26}$$

This result complicates the perturbation theory evaluation, as it requires knowledge of the perturbed flux $d\phi$. Since obtaining $d\phi$ from a separate deterministic calculation would severely detract from our goals of speed and efficiency, we will need to employ variational methods to estimate $d\phi$. By choosing an appropriate variational functional, we can arrive at a set of relations that specify $d\phi$ [155,163]. Sparing the lengthy derivation, the resulting functional is

$$\frac{\Delta\rho''}{\Delta\rho} = 1 + \left\langle \psi^{\dagger}\right| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi\Big\rangle \tag{2.27}$$

Here $\psi^\dagger$ is a generalized adjoint function, mathematically similar to an adjoint flux. In Equation 2.27, it serves as a modified "importance" weighting to reflect how the adjoint flux has shifted due to the perturbation.

We can determine $\psi^\dagger$ by expressing it in terms of known quantities:

$$\left(\tfrac{1}{k}\hat{F}^\dagger - \hat{M}^\dagger\right)\psi^\dagger = -\frac{\left(\tfrac{1}{k}d\hat{F}^\dagger - d\hat{M}^\dagger\right)\phi^\dagger}{\left\langle \phi^\dagger \middle| \left(\tfrac{1}{k}d\hat{F} - d\hat{M}\right)\phi \right\rangle} + \frac{\hat{F}^\dagger\phi^\dagger}{\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle} \tag{2.28}$$

Note that solving for $\psi^\dagger$ requires solving a matrix equation - inverting the operator on the left hand side of Equation 2.28. The matrix $\frac{1}{k}\hat{F}^\dagger - \hat{M}^\dagger$ is $NG \times NG$, where G is the number of energy groups and N the number of spatial grid cells. Essentially, Equation 2.28 is a full-core *fixed-source* problem. Thus, solving this second order perturbation formula has the same complexity as solving one outer iteration of a generalized eigenvalue problem. Although it is still faster than directly re-solving the full eigenvalue problem, it is only faster by a factor equal to the number of outer iterations required for that eigenvalue problem.

Note that Equations 2.27 and 2.28 define $\psi^\dagger$ as unitless. This allows us flexibility in terms of how we define the variational functional and its generalized adjoint. In fact, we can easily modify Equations 2.27 and 2.28 in terms of a modified generalized adjoint $\psi_2^\dagger$:

$$\psi_2^\dagger = \psi^\dagger \left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle \tag{2.29}$$

Now the modified variational functional in terms of $\psi_2^\dagger$ is

$$\frac{\Delta\rho''}{\Delta\rho} = 1 + \frac{\left\langle \psi_2^\dagger \middle| \left(\tfrac{1}{k}d\hat{F} - d\hat{M}\right)\phi \right\rangle}{\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle} \tag{2.30}$$

$$\left(\tfrac{1}{k}\hat{F}^\dagger - \hat{M}^\dagger\right)\psi_2^\dagger = \tfrac{\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle}{\left\langle \phi^\dagger \middle| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi \right\rangle}\left(\tfrac{1}{k}d\hat{F}^\dagger - d\hat{M}^\dagger\right)\phi^\dagger + \hat{F}^\dagger\phi^\dagger \tag{2.31}$$

$$= \tfrac{1}{\Delta\rho}\left(\tfrac{1}{k}d\hat{F}^\dagger - d\hat{M}^\dagger\right)\phi^\dagger + \hat{F}^\dagger\phi^\dagger \tag{2.32}$$

Now the modified functional in Equation 2.30 conveniently has the same form as a standard first order perturbation.

Finally, here is a third way to formulate the second order equations in terms of still another generalized adjoint $\psi_3^\dagger$:

$$\psi_3^\dagger = \psi_2^\dagger \frac{\left\langle \phi^\dagger \middle| \left(\tfrac{1}{k}d\hat{F} - d\hat{M}\right)\phi \right\rangle}{\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle} = \psi^\dagger \left\langle \phi^\dagger \middle| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi \right\rangle \tag{2.33}$$

The modified variational functional in terms of $\psi_3^\dagger$ is

$$\Delta\rho'' = \Delta\rho + \frac{\left\langle \psi_3^\dagger \middle| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi \right\rangle}{\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle} = \frac{\left\langle \phi^\dagger + \psi_3^\dagger \middle| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi \right\rangle}{\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle} \tag{2.34}$$

$$\left(\tfrac{1}{k}\hat{F}^\dagger - \hat{M}^\dagger\right)\psi_3^\dagger = \left(\frac{1}{k}d\hat{F}^\dagger - d\hat{M}^\dagger\right)\phi^\dagger + (\Delta\rho)\,\hat{F}^\dagger\phi^\dagger \tag{2.35}$$

Now the (further modified) functional in Equation 2.34 expresses the second order reactivity as the sum of the first order reactivity *and* a second order term of the same form. Note that in Equation 2.34, we can express the second order functional as a simple first order perturbation weighted by $\phi^\dagger + \psi_3^\dagger$ instead of only $\phi^\dagger$. Thus, we surmise that $\phi^\dagger + \psi_3^\dagger$ represents the perturbed adjoint flux $\phi^{\dagger\prime}$ and that $\psi_3^\dagger$ represents $d\phi^\dagger$.

As it turns out, we have surmised astutely. According to Stacey [155], the perturbed adjoint flux is proportional to the generalized adjoint function:

$$d\phi^\dagger = \psi^\dagger \left\langle \phi^\dagger \middle| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi \right\rangle \tag{2.36}$$

Comparing Equation 2.36 with Equation 2.33 reveals that $\psi_3^\dagger$ *does* in fact represent $d\phi^\dagger$ to first order. Thus, the fixed-source problem defined in Equation 2.35 will produce the change in adjoint flux without any proportionality factor. So we can now dispense with the "generalized adjoint" terminology and express the second order functional in terms of only $\phi^{\dagger\prime} = \phi^\dagger + d\phi^\dagger$.

$$\Delta\rho'' = \frac{\left\langle \phi^{\dagger\prime} \middle| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi \right\rangle}{\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle} \tag{2.37}$$

$$\left(\tfrac{1}{k}\hat{F}^\dagger - \hat{M}^\dagger\right)d\phi^\dagger = \left(\frac{1}{k}d\hat{F}^\dagger - d\hat{M}^\dagger\right)\phi^\dagger + (\Delta\rho)\,\hat{F}^\dagger\phi^\dagger \tag{2.38}$$

Although Equations 2.37 and 2.38 define how to obtain a second order eigenvalue estimate and a first order $d\phi^\dagger$ estimate, one also might wish to estimate the perturbed real flux $d\phi$. Stacey specifies how to obtain this perturbed flux distribution [155]. Just as we defined a generalized adjoint function $\psi^\dagger$ in Equation 2.28, we can also define a generalized real function ψ:

$$\left(\tfrac{1}{k}\hat{F} - \grave{}\hat{M}\right)\psi = -\frac{\left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi}{\left\langle \phi^\dagger \middle| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi \right\rangle} + \frac{\hat{F}\phi}{\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle} \tag{2.39}$$

Conveniently, the explicitly perturbed real flux is actually directly proportional to ψ, just as the perturbed adjoint flux is proportional to $\psi^\dagger$ in Equation 2.36:

$$d\phi = \psi \left\langle \phi^\dagger \middle| \left(\frac{1}{k} d\hat{F} - d\hat{M} \right) \phi \right\rangle \tag{2.40}$$

However, we can again eliminate the "generalized real function" terminology to express Equation 2.39 in terms of $d\phi$.

$$\left(\tfrac{1}{k}\hat{F} - \hat{M} \right) d\phi = \left(\frac{1}{k} d\hat{F} - d\hat{M} \right) \phi + (\Delta\rho)\, \hat{F}\phi \tag{2.41}$$

Note that Equation 2.41 is the analog of Equation 2.38.

In conclusion, we have now succeeded in expressing second order estimates of eigenvalue, flux, and adjoint flux without reference to any "generalized" flux functions. We can use Equations 2.37, 2.38, and 2.41 to determine all these perturbed quantities for arbitrary material density perturbations defined by $d\hat{F}$ and $d\hat{M}$.

2.4 Boundary Perturbation Theory

All the perturbation and variational techniques we have discussed so far have been centered around one class of perturbations: material density changes. However, there is a second major class of perturbations: geometry changes. Pomraning aptly characterized the fundamental distinction between these two perturbation classes [45]. Let $\epsilon << 1$ be a small perturbation parameter. Then material density perturbations are $O(\epsilon)$ cross-section changes over $O(1)$ volumes. In contrast, geometry perturbations are $O(1)$ cross-section changes over $O(\epsilon)$ volumes. Thus, geometry perturbations are inherently more difficult, because they require relatively large changes in cross-sections.

Komata published the first major paper on geometric perturbation theory in 1977 [41]. He showed that one can convert a boundary perturbation into a boundary *condition* perturbation [53]. This obviates part of the problem, but, of course, it only applies to external boundaries (where boundary conditions exist).

2.4.1 First Order Transport and Diffusion (1980s)

Pomraning, Larsen, and Rahnema carried out most of the seminal work on boundary perturbation theory in both diffusion and transport simultaneously. Their first paper "Boundary Perturbation Theory" succeeded in becoming the revered authority [42]. Larsen and Pomraning derived explicit first order perturbation expressions (in both diffusion and transport) to evaluate reactivities due to small perturbations in the *external* boundary of a reactor. Their result for one-group diffusion is

$$\Delta\rho = \frac{\int dS|\vec{r'} - \vec{r}| D\left(\hat{n}\cdot\nabla\phi^{\dagger}\right)\left(\hat{n}\cdot\nabla\phi\right)}{\int dV \phi^{\dagger}\nu\Sigma_f\phi} \tag{2.42}$$

Here we have negated the vacuum boundary extrapolation condition. The denominator contains the usual core-wide fission source integration, identical to that in classic perturbation theory. The numerator contains a surface integral over the unperturbed surface S. The quantity $|\vec{r'} - \vec{r}|$ is the magnitude of the distance between the unperturbed surface $\vec{r}$ and the perturbed surface $\vec{r'}$. The unit vector $\hat{n}$ is perpendicular to the unperturbed surface. Although we have only written this for one energy group, the extension to multigroup is simple - simply sum up the numerator and denominator over all energy groups.

Shortly after deriving this formula for *external* surfaces, the trio began to study *internal* interface perturbations - slightly moving an internal boundary between two materials so that one material "substitutes" the other. However, Rahnema and Pomraning discovered an "anomaly" in the application of boundary perturbation theory to these internal interface shifts. Specifically, the classic first order perturbation formula (Equation 2.23) does not correctly predict the first derivative of reactivity as it does for material density changes. This "anomaly" does *not* appear in transport theory - it is an artifact of the diffusion approximation that appears at internal material interfaces [43]. They corrected the "anomaly" and proposed a general expression for an internal interface perturbation in one-group diffusion:

$$\Delta\rho = \frac{\int dS|\vec{r'} - \vec{r}| \left[\left(D_R - D_L\right)\nabla\phi_R^{\dagger}\cdot\nabla\phi_L + \phi^{\dagger}\left(\left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_L - \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_R\right)\phi\right]}{\int dV \phi^{\dagger}\nu\Sigma_f\phi} \tag{2.43}$$

This applies to an internal interface shift *to the right.* The subscripts R and L denote quantities evaluated immediately to the right and left of the boundary. Note that although the flux gradient is discontinuous across the interface, the flux itself is always continuous. In terms of a material density perturbation, one could consider the R quantities to be the unperturbed case while the L quantities are the perturbed case. When a surface moves to the right, the L quantities replace the R quantities. Also note that Equations 2.42 and 2.43 have an additional minus sign that does not appear in the literature, but this is only because $\Delta(1/k) = -\Delta\rho$.

Later on, Pomraning revisited this "anomaly" and derived another corrected first order perturbation formula for an internal interface shift in one-group diffusion theory [45]:

$$\Delta\rho = \frac{\int dS|\vec{r'} - \vec{r}| \left[-\frac{D}{D'}dD\nabla\phi^{\dagger}\cdot\nabla\phi + \phi^{\dagger}\left[\frac{1}{k}d(\nu\Sigma_f) - d\Sigma_a\right]\phi\right]}{\int dV \phi^{\dagger}\nu\Sigma_f\phi} \tag{2.44}$$

Note that Equation 2.44 is precisely equivalent to Equation 2.43. Pomraning simply reformulates the perturbation in terms of a material density change. Here $d\Sigma_a = \Sigma_{a,L} - \Sigma_{a,R}$

and $d(\nu\Sigma_f) = \nu\Sigma_{f,L} - \nu\Sigma_{f,R}$. Similarly, $dD = D_L - D_R$, $D = D_R$, and $D' = D_L$. The two unperturbed flux gradients are evaluated to the right of the perturbation, because the rightward region is the unperturbed region. Conservation of (real and adjoint) current across the interface can allow one to convert the leakage term in Equation 2.44 to that in Equation 2.43.

Pomraning notes that Equation 2.44 is fully general to both material density and boundary perturbations. In the case of a small material density perturbation, one can estimate $D/D' \approx 1$. Then the numerator in Equation 2.44 is identical to a classic first order perturbation formula (for material densities).

These expressions in Equations 2.42, 2.43, and 2.44 correctly predict the first order (one-group diffusion) reactivity due to small perturbations in external and internal boundaries. However, the predictions will only be correct if one has obtained very accurate (real and adjoint) flux gradients on the unperturbed boundary.

Rahnema and Pomraning derived one-group first order perturbation formulas for changes in various quantities (anything that is a linear functional of the flux - not just eigenvalue) due to changes in *external* boundaries in fixed-source problems [46]. Rahnema and Pomraning later wrote a paper generalizing these methods to full multigroup transport and diffusion [48]. This leap was crucial, as previous papers had demonstrated the methods only for simple one-group examples.

Pomraning showed that one can obtain first order estimates for reactivities due to *non-analytic* perturbations in the *external* boundaries of a reactor [47]. The previous methods allowed for only an analytic continuous extension of the material adjacent to the external boundary, but Pomraning's work here allows for adding arbitrary material compositions to the external boundary such that the boundary perturbation is "non-analytic".

At last, Rahnema and Pomraning presented multigroup transport and diffusion perturbation formulas for *internal* interface perturbations coupled with material density perturbations [49]. This was the final step in demonstrating that *first order* boundary perturbation theory was possible for both internal and external boundaries in both transport and diffusion.

2.4.2 Higher Order Transport and Diffusion (1990s-2000s)

After the initial flurry of work by Pomraning, Larsen, and Rahnema, the literature is mostly silent on the topic for about a dozen years. Rahnema revisited the internal interface problem in the mid-1990s, and he re-derived the first order expressions via the "crossmultiplication" method [50].

Then, Gheorghiu and Rahnema developed the first higher order (variational) estimates of reactivity due to boundary perturbations [51,52]. This work covers both transport and diffusion, but it is only valid for *external* boundary perturbations. Internal boundaries are not covered.

Rahnema and Ravetto later built on the original work of Komata [41] to convert first order boundary perturbations into first order boundary *condition* perturbations [53]. Rahnema and McKinley did much further work on these boundary condition perturbation methods, and they applied higher order formulas for both diffusion and transport [54,55]. Note that because these methods require converting a boundary shift into a boundary condition change, they apply only to *external* reactor boundaries. Like the work by Rahnema and Gheorghiu, they are not applicable to internal boundaries.

During the past ten years, Favorite has performed most of the substantial work on boundary perturbation theory. His work is almost entirely in transport theory. He developed the first higher order (variational) estimates for *internal* interface perturbations by combining elements of the Roussopoulos and Schwinger functionals [59,60]. Favorite and Bledsoe discussed the sensitivity of eigenvalue with respect to uniform expansions or contractions of surfaces [61], and they showed equivalence with Rahnema's earlier work [49].

Favorite compared the Roussopoulos formula [1] to the first order linear functional equations derived by Rahnema and Pomraning for external boundary perturbations [46]. Essentially, he finds that the Roussopoulos formula fails in predicting volume-integrated quantities in external boundary perturbations but succeeds in predicting perturbed quantities on that boundary [64]. The methods developed by Rahnema and Pomraning succeed for the volume-integrated quantities but cannot be applied to quantities on the perturbed boundary.

Much of Favorite's work has been related to fixed-source shielding problems for applications in non-proliferation, and he has applied boundary perturbations in transport theory to such scenarios [65].

However, some aspects of Favorite's work are very pertinent to realistic distortions in critical reactors. He attempted to apply his higher order transport perturbation techniques to the Godiva and Zeus experiments [63,62], which we will discuss in detail shortly.

In summary, higher order (variational) boundary perturbation techniques have been developed for *external* boundaries in both transport and diffusion, but they have been developed for *internal* boundaries only in transport. We have not found higher order (variational) techniques for *internal* boundaries in *diffusion* in the literature.

2.4.3 Perturbed Flux and Power via Variational Methods

While computing the perturbed eigenvalue to first order requires only first order perturbation theory, computing the perturbed flux to first order requires *second* order perturbation theory (variational methods).

Although the literature is rife with examples for eigenvalue perturbations due to boundary shifts, flux perturbations due to boundary shifts are much less common. Pomraning and Rahnema derived variational functionals to evaluate changes in integrated reaction rates (such as fission or absorption) due to *external* boundary perturbations [46]. Much later, Rahnema and McKinley were able to estimate perturbed fluxes due to *external* boundary shifts [54,55].

Rahnema, Ilas, and Mosher employed variational methods to develop coarse mesh transport methods [56,58]. After solving a large heterogeneous system on a coarse mesh, they could apply variational techniques to each coarse mesh cell in order to determine the intra-cell flux distribution [57]. Given (1) a fission source within a cell and (2) a flux source on the cell boundary, one can construct trial functions to approximate the flux distribution within the cell. However, none of this work deals with boundary *changes.*

Although Favorite's work in transport theory succeeds in deriving variational methods for reactivities due to *internal* boundary perturbations, he has not yet extended his methods to predict perturbed flux distributions in critical reactors (although he has applied them to fixed-source detector problems). It appears that virtually no work has been done on perturbed flux distributions due to *internal* boundary perturbations in diffusion.

2.4.4 Application to Practical Reactor Models

Throughout the literature, analytic and numeric examples for boundary perturbation theory are given for only very simple geometries, often only 1-D slabs. Very little application to complex geometries exists in the literature, and no application to realistic full-core 3-D models can be found.

Rahnema, Pomraning, and others studied neutronic effects of practical "distorted core configurations" in a 2-D gas-cooled fast reactor (GFR) model [44]. Due to strong anisotropic neutronic effects in GFRs, they performed all the calculations in transport theory. However, even though this work came out of Rahnema's Ph.D. thesis on boundary perturbation theory, perturbation theory was *not* applied to the GFR model. All the GFR reactivity effects shown in this work were directly computed via transport theory.

Rahnema and McKinley applied their variational techniques for *external* boundaries to two-region and three-region fuel assembly slab problems [54,55].

Favorite applied variational boundary perturbation techniques in transport theory for a number of fairly realistic scenarios, but many of these were fixed-source shielding and detection problems [65]. However, Favorite's application of higher order boundary perturbation theory to the Godiva and Zeus experiments provides great insight into potential application to realistic reactors [63,62].

The Godiva device at Los Alamos consisted of a pure ^{235}U sphere centered within (and surrounded by) a larger spherical shell of pure light water. See Figure 2.2. Considered "naked" with neither reflector nor shield, it was named for the 11th century Anglo-Saxon noblewoman Lady Godiva. Favorite applies higher order perturbation methods (based on Rahnema's original work [49]) in transport to translate the sphere in one direction, breaking the problem's spherical symmetry. Thus, the perturbation methods require two surface integrals, each over a half sphere. Favorite finds that the perturbation methods work much better for each half-sphere perturbation individually than for the whole-sphere translation, which cannot even predict the reactivity to within 20% [62]. However, one must note that because this problem breaks the initial symmetry, the first order reactivity is precisely zero. Only the second order reactivity is non-zero. Thus, this 20% error is entirely second order, not first order. Figure 2 in Favorite's American Nuclear Society (ANS) summary [62] suggests that he *does* in fact correctly predict the zero first order reactivity.

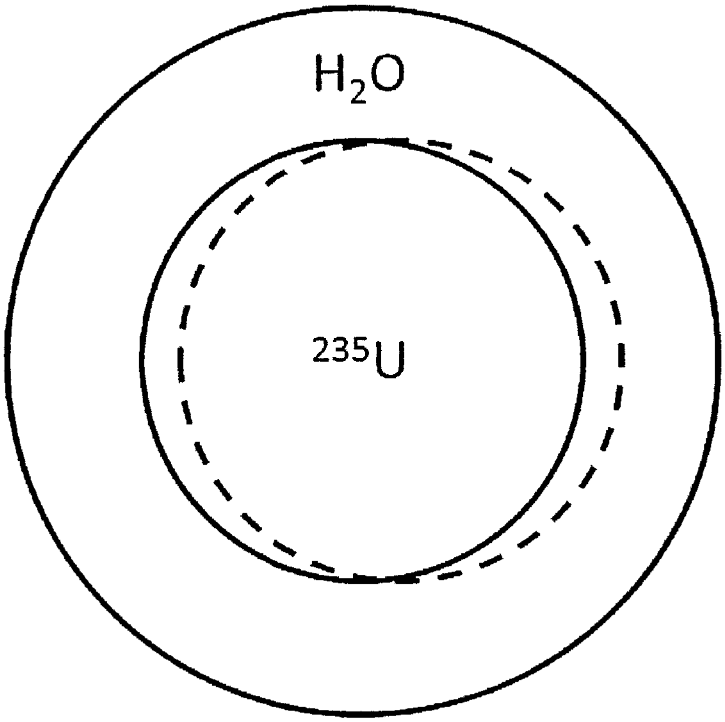

Figure 2.2: The Godiva sphere translation problem. A solid ^{235}U sphere is initially centered within a larger sphere of water. It is perturbed slightly in one direction (represented by the dashed circle), breaking the problem's initial spherical symmetry [62].

As illustrated in Figure 2.3, the Zeus problem consists of alternating 3-D finite slabs of highly-enriched uranium (HEU) and graphite [145,148]. Thus, the problem contains numerous internal interfaces between HEU and graphite. Favorite models the system in transport theory, and he demonstrates that first order boundary perturbation theory correctly predicts the first derivatives of the eigenvalue with respect to shifts of each *single* interface. However, he finds that the boundary perturbation techniques completely fail to predict the first derivative of the eigenvalue with respect to shifting the lower 40% of the core (many interfaces moving together). While the sum of first derivatives due to each *single* interface shift produces an error of only 1.3%, the first derivative due to multiple interface shifts produces an error of 32% [63]. So it appears that the reactivities due to internal interfaces do not add - not even in first order! Favorite even states that "the first order theory has no way to account for interacting effects of multiple perturbations" and that the derivative of eigenvalue "with respect to the location of the Zeus lower core is not the simple sum of the derivatives with respect to the location of the individual surfaces". Thus, one major barrier to applying boundary perturbation techniques to practical reactor models may be that the reactivity effects due to individual interface perturbations are not always additive.

In summary, application of boundary perturbation techniques to practical core distortion scenarios has been extremely limited - mostly to simple 1-D or 2-D models or to simple benchmark problems like Godiva. Zeus represents the most complex geometry to which boundary perturbation techniques have been applied in the literature, and the techniques fail for some practical scenarios in that problem.

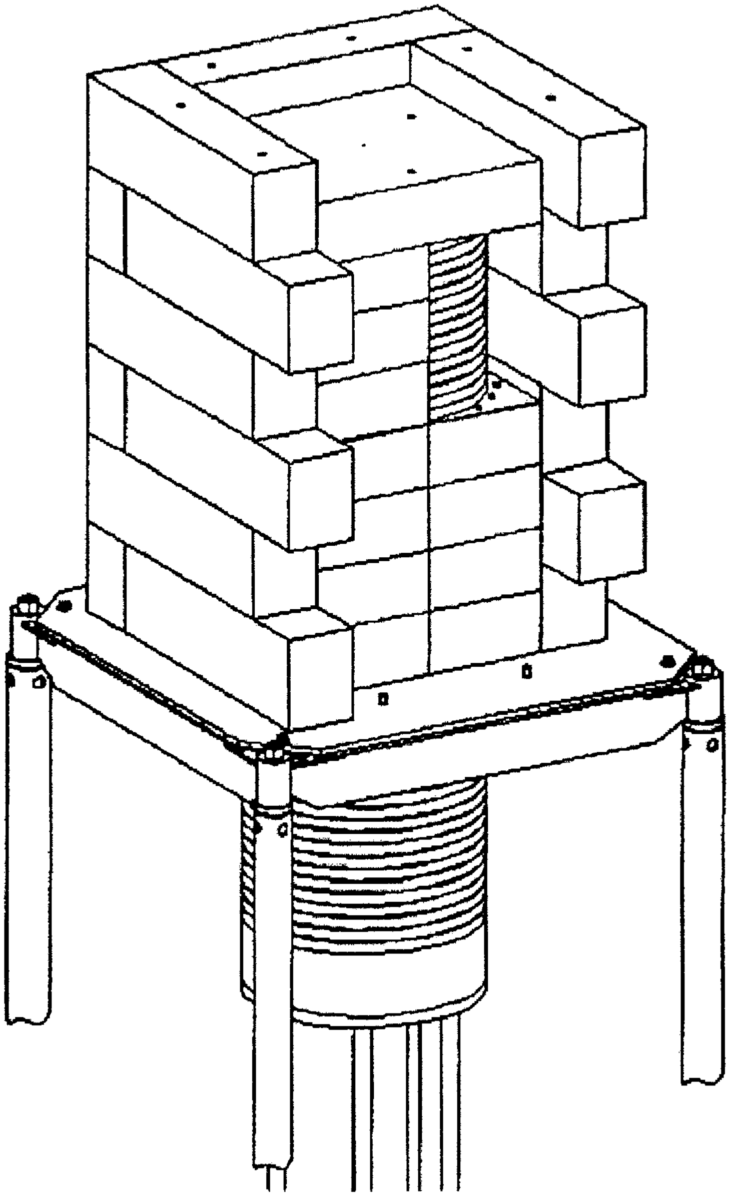

Figure 2.3: The Zeus critical assembly experiment at Los Alamos National Laboratory [145]. Copper reflector slabs surround an "assembly" of alternating cylindrical slabs of HEU and graphite. The bottom 40% of this assembly can be moved downward.

Table 2.1: What Has and Hasn't Been Done in Perturbation Theory

	mat. density	ext. boundary	int. boundary
1st order eigenvalue in transport			
2nd order eigenvalue in transport			
flux in transport			
modal expansion in transport		No	No
1st order eigenvalue in diffusion			
2nd order eigenvalue in diffusion			No
flux in diffusion			No
modal expansion in diffusion		No	No

2.5 A Summary of the Perturbation Theory Literature

Here we present a succinct summary of our literature review of perturbation theory applied to neutronics. Table 2.1 clearly shows which categories of perturbations have been implemented, and which haven't. Here we present a few general trends:

- Arbitrary material density perturbations have been solved in both transport and diffusion.

- *External* boundary perturbations have been much more widely studied than *internal* boundary perturbations, in part because external boundary perturbations can be converted into boundary *condition* perturbations.

- Internal boundary perturbations have not been applied beyond first order in diffusion, while they have been applied much more extensively in transport. Perhaps this is due to the diffusion "anomaly" that arises on internal surfaces [43].

- In the main body of literature, modal expansion techniques have only been applied to diffusion theory. However, since application to transport is straightforward, it would be surprising if no one has ever applied modal expansion to transport. Thus, we use the word "Maybe" for this place in Table 2.1.

- Modal expansion techniques have not been applied to boundary perturbations.

2.6 Attempts at Modeling Practical Assembly Distortions in Fast Reactors

Despite the fact that boundary perturbation theory has not been applied to practical reactor models, there have been valiant attempts to quantify practical distortion scenarios in fast reactors. Most notably, Knutson, Wigeland, and Kamal devised methods to model hexagonal assembly bowing and flowering.

2.6.1 Knutson's Model of Assembly Bowing

In 1981, Knutson, Lucoff, Harris, and Hecht conceived a model to estimate the reactivity due to individual assembly movements in the Fast Flux Test Facility (FFTF) [101,102]. He does not employ boundary perturbation theory, but only classic first order perturbation theory (for material densities). Consider a fuel assembly immersed in sodium. Now imagine that assembly shifting radially outward very slightly. There will be two reactivity effects - one due to the fuel motion, and another due to the sodium "backfill". For example, the fuel reactivity worth W_{fuel} per unit displacement is

$$W_{\text{fuel}} = \left(W_{\text{fuel,outer}} - W_{\text{fuel,inner}}\right) V_{\text{fuel}} \tag{2.45}$$

Here $W_{\text{fuel,outer}}$ is the fuel worth (per unit volume) on the outer edge of the assembly, while $W_{\text{fuel,inner}}$ is the fuel worth on the inner edge of the assembly. V_{fuel} represents the displaced sodium volume per unit radial displacement. Now in order to obtain the total reactivity effect, one must simply add W_{fuel} to W_{cool}, the sodium "backfill" worth gradient.

$$W = W_{\text{fuel}} + W_{\text{cool}} \tag{2.46}$$

Here W represents the reactivity worth gradient for a given assembly at a given axial position. Knutson calculates W in both the x and y directions. He obtains the mechanical displacements from NUBOW-3D [192].

Knutson benchmarks his model against a direct 2-D hexagonal diffusion calculation of uniform radial swelling in the FFTF core. He attempts to predict the reactivity due to a 100 mil = 1/10 inch uniform radial swelling. This very small shift corresponds to a radial increase of about 0.2%. Knutson's model predicts the exact reactivity to no less than 12%. He subsequently performs analysis to conclude that nearly all of this error arises from spatial effects rather than spectral effects. His model, which is based solely on material density perturbations, cannot capture the effect of decreased leakage due to increased core size.

2.6.2 Wigeland's Model of Assembly Bowing

In the 1980s, Wigeland developed a model [105,107] that was incorporated into the SASSYS and SAS4A codes [196,195]. The bulk of Wigeland's work focuses on computing assembly displacements as functions of temperature using simple models that agreed with the mechanical distortion code NUBOW-3D [192]. However, he also extends the work to reactivity analysis. In his model, the reactivity $\Delta\rho$ is

$$\Delta\rho = W\Delta R \tag{2.47}$$

Here ΔR is the (axially-dependent) radial displacement of an "average" outer assembly - the average distance by which the assemblies on the core edge are pushed radially outward. W is the "uniform dilation" radial swelling reactivity worth for the whole core, which is determined by a full 3-D neutronics eigenvalue calculation. The pitch of each assembly is increased by 1%, and the reactivity is recorded. In order to capture the axial dependence of W, Wigeland assumes that W is proportional to the axial power shape. He finds that the reactivities predicted by Equation 2.47 agree with detailed 3-D calculations to within 20%. In his detailed report, Wigeland estimates the reactivity uncertainty to be approximately 30% [107].

Wigeland compares his model to empirical correlations and to experimental results from the Fast Flux Test Facility (FFTF) [146] in two American Nuclear Society (ANS) conference proceedings [106,108]. Agreement is fairly strong, although the model differs from experimental data by as much as 20% in some cases.

It is important to note that Wigeland's model is *not* based on perturbation theory. The "uniform dilation" radial swelling worth input is a direct calculation, not a perturbation. This allows Wigeland's model to avoid the pitfall of Knutson's work, which does not account for changes in core size.

Nevertheless, Wigeland's model has a few limitations of its own. First, the "uniform dilation" worth does *not* account for non-uniformities in radial swelling [105]. The entire model is based on the amount by which the outer ring of assemblies is pushed outward. In reality, one or two hexagonal assembly rings (located in the high power gradient region) account for most of the whole-core radial swelling. Although these non-uniformities appear not to be terribly important for the FFTF, we surmise that they would be much more important for larger cores with annular power profiles (such that the peak power is *not* near the core center). Second, the assumption that the axial reactivity worth profile is equal to the axial power profile likely introduces considerable error. This error could be reduced by introducing multigroup adjoint-weighting into the axial worth distribution.

2.6.3 Kamal's Model of Assembly Bowing

In the mid-1980s, Kamal and Orechwa devised a third model for the reactivity worths of assembly bowing and core flowering [103,104]. Their reactivity model is conceptually similar to Knutson's, except that they allow each assembly to move in an arbitrary direction (instead of only radially outward). They compute "displacement reactivity worths" W_x and W_y for assembly movement in the x and y directions, and then they assume a simple cosine axial worth distribution shape. These displacement worths are computed one at a time via direct DIF3D eigenvalue calculations, although the precise methodology is unclear.

$$\Delta\rho = W_x \Delta x + W_y \Delta y \tag{2.48}$$

Here W_x and W_y are the worth gradients in the x and y directions, while Δx and Δy represent the small assembly displacements.

Kamal and Orechwa apply this method to both the "limited free bowing" and "free flowering" core restraint systems with distortion input obtained from NUBOW-3D [192]. They conclude that the "limited free bowing" case actually provides greater reactivity safety, because the upper core restraint forces the assemblies to bow outward in the central core region where reactivity worths are greatest.

While Wigeland's model requires direct calculation of the uniform radial swelling coefficient, Kamal's model seems to require a lengthy direct eigenvalue calculation for *each* assembly. Thus, the required pre-computation time is extremely large. Furthermore, Kamal's simple cosine axial worth shapes are even less accurate than Wigeland's power axial worth shapes.

Note that these three models proposed by Knutson, Wigeland, and Kamal all assume that reactivity is a *linear* function of assembly position. Thus, first order perturbation theory should be sufficiently accurate to model these distortions. This linearity is not surprising, as each assembly movement amounts to a sodium displacement, and the linearity of sodium reactivity in fast reactors is widely known. Stacey even states that "the sodium void coefficient varies directly with the ratio of the number of sodium atoms removed to the number of fuel atoms present" [163].

2.7 Why We Still Need Perturbation Theory

Since the inception of perturbation theory, computers have appeared and become quite fast. Thus, we can solve partial differential equations numerically. The fact that few analytic solutions exist for the Schrödinger equation is no longer such a big deal - if no analytic solution exists, we can just solve it numerically. This is also true for the neutron diffusion and transport equations.

So why do we still need perturbation theory? Why would we employ clever mathematical tricks to obtain an *approximate* solution when we can directly obtain the *exact* solution via numeric methods? Is there really still a need for this, or is it merely an outmoded relic of the past?

We believe that perturbation theory *is* still necessary for **two main reasons**:

First, perturbation theory is fast. Solving the neutron transport equation deterministically for large, heterogeneous, 3-D problems on a fine mesh is still an active area of research. Determining the neutron distribution as a function of energy, angle, and 3-D space is very computationally intensive. In contrast, perturbing an already-solved neutron distribution is fast. There are no eigenvalue problems to solve, no unknowns to determine. The process is simply adding and multiplying known values, and it can be thousands of times faster than re-solving the full problem. Furthermore, since classic first order perturbations are linearly independent, perturbation theory is "embarrassingly parallel". One can compute the reactivity effects due to multiple perturbations in parallel (on multiple processors) and then simply add up the results at the end. Although perturbation theory is fast, parallelization may still be desirable if working with high-level languages such as Python [207] or MATLAB [210].

One might argue that these gains in speed and computational efficiency are not worth the loss of accuracy and increase in uncertainty that perturbation theory introduces. However, this depends on the chosen application. For example, perturbation theory is widely used throughout the fast reactor community to compute reactivity coefficients - the instantaneous derivatives of reactivity with respect to material densities and temperatures. While perturbation theory is certainly not precise for large perturbations, it is *exactly* precise for infinitesimally small perturbations. First order perturbation theory can predict the *first derivative* of reactivity with respect to cross-sections changes *exactly*. Thus, in the case of computing reactivity coefficients, perturbation theory is no less accurate than directly re-solving the system - and it's orders of magnitude faster.

Second, perturbation theory acts within the same spatial domain - the same geometry - as the original unperturbed solution. In the case of neutronics, re-solving the neutron transport or diffusion equations for arbitrary material densities or compositions is trivial. One simply inputs different numbers and re-runs the code. In stark contrast, re-solving these equations for arbitrary geometries is most certainly *not* trivial. Constructing meshes to accurately model arbitrarily-perturbed geometries is a quagmire. However, the elegance of perturbation theory is that no change in domain (no re-meshing) is required. Perturbation theory can estimate changes in reactivity due to small geometry changes based on the solution in an unperturbed geometry. This obviation of re-meshing is tremendously expeditious - not just in computation expense, but in manhours.

Thus, we still need perturbation theory for (1) instantaneous eigenvalue derivatives in reactivity coefficients and (2) geometry distortions. All other material density perturbations beyond (1) may be unnecessary with today's computational tools. As we saw previously, second order material density perturbations require as much computational power as one outer iteration in a generalized eigenvalue problem. That's not a huge gain in computational efficiency considering the loss of accuracy and increase in uncertainty. First order perturbation theory to compute reactivity coefficients is rapid and accurate, but it is already fully known. What isn't fully known, however, is (2). Boundary perturbation theory has still not been developed to the extent that we can accurately apply it to arbitrary core distortion scenarios. In our humble opinion, *that* should be the forward thrust of perturbation theory research.

2.8 Summary and the Path Forward

The objective of this thesis work is to apply geometric perturbation theory to models of *practical* core distortions in *real* reactor models. So far, boundary perturbation theory hasn't quite made that leap. Knutson's model employed only material density perturbations(not boundary perturbations), and the Wigeland and Kamal models did not employ perturbation theory at all. None of these three models could predict distortion reactivities more precisely than the range 10-20%. Knutson's model could not account for decreased leakage due to increased core size. The Wigeland and Kamal models required lengthy direct eigenvalue calculations as pre-computation, and they employ simple power shape weighting instead of multigroup adjoint weighting.

In order to accomplish this objective, we will develop a new type of perturbation theory that treats material density and boundary perturbations as a single entity. We will validate this new theory numerically for numerous simple geometries in finite difference diffusion. Furthermore, we will show that this new theory is analytically equivalent (but not numerically equivalent) to traditional boundary perturbation theory. The culmination of this work will be application of perturbation theory to distortions in full-core, hexagonal-z fast reactor diffusion models with unprecedented accuracy.

3 A Review of Uniform Isotropic "Virtual Density" Theory

3.1 Chapter Abstract

We introduce the uniform isotropic "virtual density" theory of neutronics, which equates uniform isotropic changes in core size to uniform density changes. A change in core size is isotropic if the "shape" of each core region remains fixed. We present the basic concept, which has been mentioned sparsely throughout the literature but rarely stated formally. We opt to define new formalism and demonstrate its validity, both analytically and numerically. First, we show that it is equally valid in diffusion, deterministic transport, and Monte Carlo. We then apply it to perturbation theory to predict reactivity coefficients for reactor swellings and expansions in isotropic cases. This limits us to two classes of perturbations: (1) uniform swellings and expansions of 2-D heterogeneous full-core models and (2) arbitrary geometry perturbations in 1-D problems. We demonstrate (1) with a simple 2-D Cartesian sodium fast reactor (SFR) model and with a 2-D hexagonal model of the Fast Flux Test Facility (FFTF). Furthermore, we show the precise equivalence of "virtual density" and traditional boundary perturbation theory using analytic solutions to simple homogeneous geometries. We leave (2) for discussion in Chapter 8. We conclude that the uniform isotropic "virtual density" theory allows for highly accurate prediction of reactivities in coarse mesh finite difference solutions, even for highly-heterogeneous cores.

3.2 Introduction

Core reactivities are sensitive to geometry distortions arising from three distinct phenomena: (1) irradiation swelling of fuel throughout core lifetime, (2) thermal swelling of fuel during transients, and (3) mechanical oscillations during seismic events. Performing comprehensive reactivity analysis of these distortions requires methods for rapidly computing a multitude of small shifts. Traditionally, these reactivity effects have been studied via boundary perturbation theory developed by Pomraning, Larsen, and Rahnema [42,46,48,49]. However, those methods were never applied to full-core 3-D reactor models, and there is evidence to suggest that multiple boundary perturbations interfere with one another [63].

Thus, we introduce the "virtual density" theory of neutronics as a new perturbation method based on fundamentally different principles. Essentially, this "virtual density" theory converts geometric perturbations into equivalent material density perturbations, which are more accurate and much simpler to evaluate. In this chapter, we introduce and validate this technique for uniform isotropic cases, which include swellings and expansions of 2-D heterogeneous full-core models.

3.3 The "Virtual Density" Concept

For now, let us consider an idealized uniform core swelling in which every material in the entire reactor swells by the same factor. Of course, in real scenarios the liquid coolant would not swell at the same rate as other materials, but we will neglect that effect for now. Let the reactor volume be V, and let any given material atom density be N. If mass is conserved, then N will vary inversely proportional to V.

$$N \propto \frac{1}{V} \tag{3.1}$$

If we let λ be any given neutron mean free path in the core, then λ will vary proportional to V. Let Σ_t and σ_t be the macroscopic and microscopic total cross-sections, respectively.

$$\lambda \propto \frac{1}{\Sigma_t} = \frac{1}{N\sigma_t} \propto V \tag{3.2}$$

During this core swelling, the neutron mean free path increases at a rate greater than the rate at which all linear core dimensions increase. Thus, the net reactivity will be negative - the negative reactivity effect due to reduced material densities overcomes the positive reactivity effect due to increased core size, and so the net effect is increased leakage. This is true for any arbitrary reactor (homogenous or heterogeneous) except an array of infinite 1-D slabs, for which the net reactivity is precisely zero.

If we wished to keep reactivity constant during a uniform swelling of an arbitrary reactor, we would need to scale up the neutron mean free path proportional to the core linear dimensions ($V^{1/3}$).

$$\lambda \propto V^{1/3} \rightarrow N \propto V^{-1/3} \tag{3.3}$$

Thus, if we uniformly swell the core volume V and simultaneously reduce all core material densities proportional to $V^{1/3}$, the neutron leakage rate does not change. Of course, it follows that the relative magnitudes of neutron fluxes between internal core regions do not change such that the spatial neutron distribution does not change. Furthermore, because the relative proportions of all materials in the core are fixed, the neutron energy spectrum is also fixed. Thus, the reactor is essentially "scaled up" with no change to reactivity, spatial flux distributions, or local flux spectra.

3.3.1 Three Axioms

Intuitively, we can summarize this generic principle with three closely-related axioms:

Axiom 1: Swelling all linear dimensions of any reactor by a certain factor while simultaneously reducing all material densities by that same factor will result in exactly zero change to reactivity and relative flux distributions.

Axiom 2: The reactivity and flux distribution effects of a uniform core swelling (or contraction) can be exactly replicated by manipulating material densities with no change to core geometry.

Axiom 2 is a direct logical consequence of Axiom 1. While Axiom 1 states that we can *counteract* a dimension change with a density change, Axiom 2 states that we can *replicate* a dimension change with a density change. Naturally, this begs for a third axiom:

Axiom 3: If any arbitrary dimensional change can be *counteracted* by a material change, then that same dimension change can be *replicated* by a different material density change.

These three axioms encapsulate the basic "virtual density" theory for isotropic expansions and swellings.

3.3.2 Definitions: "Swelling", "Expansion", "Isotropic", "Anisotropic", "Uniform", "Non-Uniform"

Throughout this thesis, we define a *swelling* as a dimension change with a corresponding density change that conserves mass. Thermal enlargement of a solid is one example of a swelling. In contrast, we define an *expansion* as a dimension change without a density change. Coolant flowing into an expanding inner core region is one example of this - the size of the region increases, but the coolant density occupying that region remains constant. Note that while expansions and swellings are both geometry changes, they differ from each other by only a density change. We define *enlargement* as a neutral term to describe a change in core size that could be either a swelling or an expansion.

A swelling or expansion of a reactor core zone is *isotropic* if it retains its "shape" when it enlarges. Thus, a cube expanding isotropically in all directions is still a cube - it has retained its shape. However, a cube expanding *an*isotropically in only one or two directions becomes a rectangular parallelepiped - its shape has changed. Similarly, a sphere expanding isotropically is still a sphere, but a sphere expanding *an*isotropically is an ellipsoid. Thus, isotropic swellings and expansions have only very limited application in neutronics. We can isotropically (radially) swell 2-D heterogeneous full-core models. We could also do the same for 3-D heterogeneous full-core models, but that would not be terribly realistic, as cores do not swell axially and radially by the same factor. Finally, we can always apply isotropic expansion to infinite 1-D slabs, because 1-D slabs always retain their "shape" regardless of their thickness.

An isotropic or anisotropic swelling or expansion is *uniform* if it occurs throughout a whole core by the same factor, such as uniform radial or axial swelling of a whole fast reactor core. An isotropic or anisotropic swelling or expansion is *non-uniform* if it occurs in only one subregion of a core (such as a single assembly swelling axially) or by different factors in different subregions (such as fuel and control rod assemblies swelling axially by different factors).

See Figure 3.1 for illustrations of these terms. In this chapter, we analyze uniform isotropic swellings and expansions, which are mentioned sparsely in Western literature. In Chapter 4, we will analyze uniform *an*isotropic swellings and expansions, which are studied in old Soviet and Russian literature. Most of the remainder of this thesis is concerned with *non*-uniform anisotropic swellings and expansions, which apparently do not exist in prior literature.

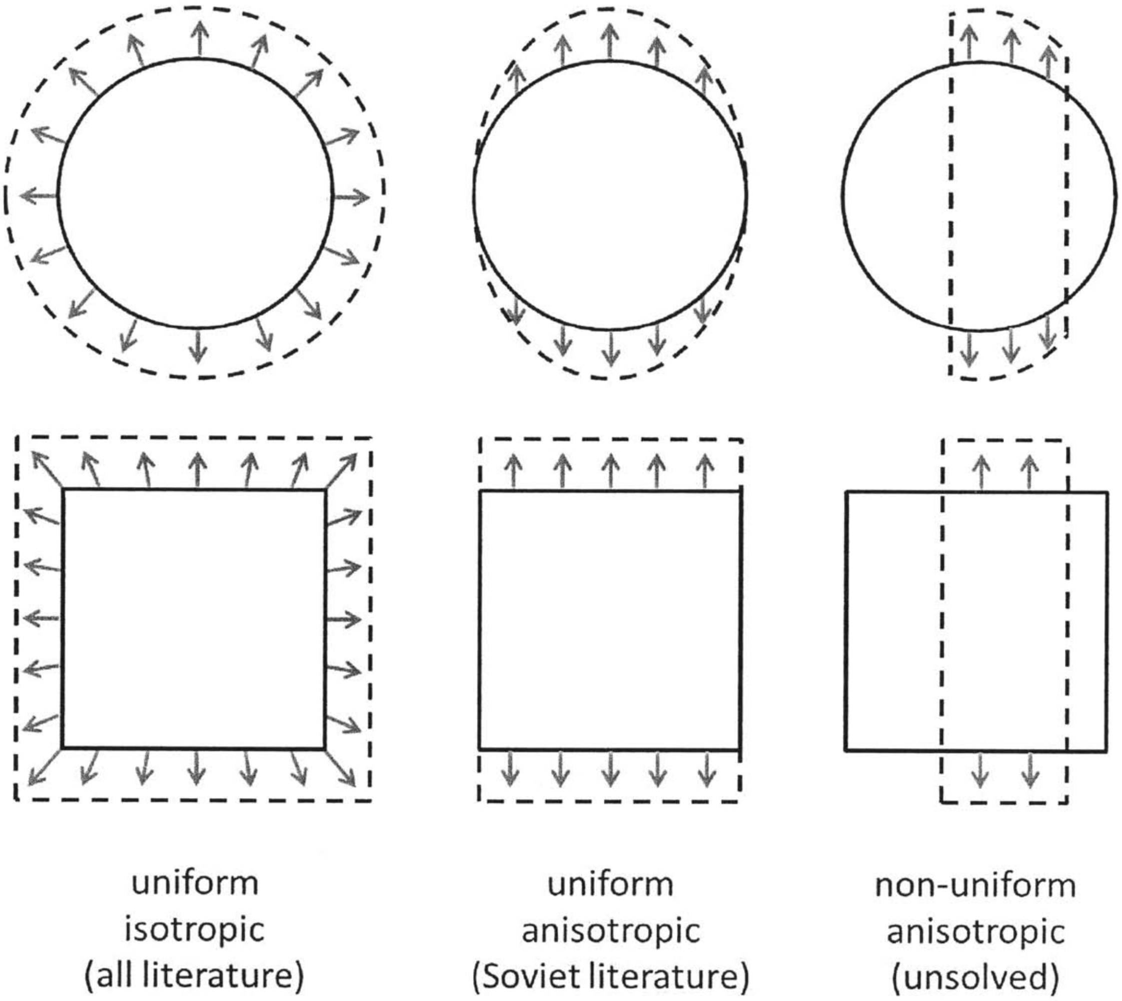

Figure 3.1: Pictorial definitions of "isotropic", "anisotropic", "uniform", and "non-uniform" geometry changes. Uniform isotropic cases are considered well-known and appear sparsely throughout all neutronics literature. Uniform anisotropic cases appear only in old Soviet and Russian literature. Non-uniform anisotropic cases constitute an unsolved problem, and a major contribution of this thesis is to solve it.

3.3.3 Literature Review

Shikhov proposed what he called "similarity theory" in 1959 [87,90]. Since then, the Russian reactor physics community has used it to determine uniform core swelling reactivity coefficients. The design of the BN-800, a sodium fast reactor currently under construction in Russia, employed some of these methods [95]. However, these methods appear not to have been extended to any non-uniform swellings or expansions. Thus, they are only useful for whole-core uniform swellings and expansions, and they are not applicable to localized geometry distortions.

Two Russian authors wrote a paper in 1985 that references both work by Shikhov and the boundary perturbation theory methods that Pomraning, Larsen, and Rahnema had recently developed [93]. However, this paper focuses on boundary perturbation theory and does not draw any comparisons to Shikhov's "similarity theory".

Physicists working on the Manhattan Project also used the generic relationship between material density and system dimensions [86]. Back then, performing an accurate eigenvalue calculation (even for extremely simple geometries) was laborious, and so this "density law" provided a simple way for them to know what would happen if they were to scale up (or down) the dimensions of a critical system. As described in a later report by Stratton, they knew that the critical radius of a system is inversely proportional to density and that the critical mass is inversely proportional to the square of density [89]. Stratton also states that the law applies precisely to the extrapolation distance in diffusion theory, which one would expect. This early literature does not have any specific name for the principle, but it is described as "the only law in criticality physics which is simultaneously exact, simple, and useful" [89]. Stratton elegantly phrases the law like this:

> "In a critical system, if the densities are increased everywhere to x times their initial value and all the linear dimensions are reduced to $1/x$ times their initial value, the system will remain critical."

However, it appears that application of this uniform isotropic "virtual density" theory was limited to the early criticality experiments. The western literature does not describe its application to full-scale reactor design, perturbation theory, or reactivity coefficients. It appears that Shikhov conceived the concept independently, and then the Soviets developed it further than their Cold War foes.

Furthermore, Cullen has published two reports on this subject. He recognizes the relationship between density and dimensions for isotropic uniform core swelling, and he gives numeric examples for many simple geometries [96,97].

It is intriguing that Soviet authors could read western literature, while western authors could not read Soviet literature until the early 1990s. Thus, all the Russian reactor physicists had access to the boundary perturbation work developed by Pomraning, Larsen, and Rahnema, but those three had no knowledge of the Russian work. If not for the Iron Curtain, many of these connections and comparisons would have likely been made much earlier.

The central purpose of this thesis work is to generalize Shikhov's original theory so that it applies to *non-uniform* core swellings and expansions, which can describe geometry distortions. We will refer to this new generalized theory as **the "virtual density" theory of neutronics**. We will closely follow and build upon our previously-published work on this topic [98,99].

3.3.4 Defining a "Virtual" Uniform Radial Core Swelling in 2-D

Now let us examine a realistic uniform isotropic "virtual density" swelling scenario. Consider a 2-D sodium fast reactor (SFR). We will define a uniform radial core swelling as a uniform increase in pin pitch. Thus, the core swells uniformly in the radial direction but not in the axial direction. If the core swells from radius R_1 to a larger radius R_2, the homogenized fuel and structure densities in each assembly will decrease by a factor of $(R_2/R_1)^2$. As the pin pitch increases, sodium flows into the core. Thus, the homogenized sodium density will increase by the factor $(R_1/R_2)^2[(R_2/R_1)^2$ - $F]/(1$-$F)$, where F is the initial volume fraction of fuel and structure.

Given our three axioms, we can exactly replicate the reactivity of such a uniform core swelling without actually swelling the core. Instead of enlarging the core, we simply contract the neutron mean free paths by increasing all material densities by a factor of R_2/R_1 above what they would be during an "actual" core swelling. For example, we increase the fuel and structure densities by a factor of R_2/R_1 after they have already been decreased by a factor of $(R_2/R_1)^2$ during the "actual" swelling. Thus, the net effect of the "virtual" swelling is that the fuel and structure densities are decreased by a factor of R_2/R_1, which is less than the factor they would have been decreased by during an "actual" swelling. See Figure 3.2 for a summary.

3.3.5 Numeric Equivalence in Diffusion, Deterministic Transport, and Monte Carlo

We can demonstrate the universality of this "virtual density" theory by showing that it is equally valid in diffusion, deterministic transport, and Monte Carlo simulations.

In all three simulations, we use the same very simple geometry: a 5x5 array of square enriched uranium assemblies surrounded by a square sodium blanket. See Figure 3.3 for an illustration. Without swelling, this reactor has $k_{\text{eff}} = 1.04$. The vacuum boundary condition provides plenty of leakage. The fuel and structure volume fraction is $F = 0.62$, although the square assemblies are homogenized with no pin-level detail. As each homogeneous assembly

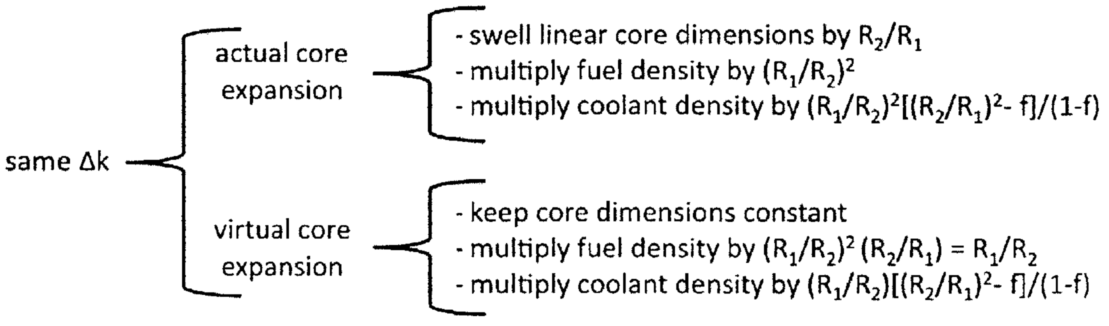
same Δk
actual core expansion
- swell linear core dimensions by R_2/R_1
- multiply fuel density by $(R_1/R_2)^2$
- multiply coolant density by $(R_1/R_2)^2[(R_2/R_1)^2- f]/(1-f)$
virtual core expansion
- keep core dimensions constant
- multiply fuel density by $(R_1/R_2)^2 (R_2/R_1) = R_1/R_2$
- multiply coolant density by $(R_1/R_2)[(R_2/R_1)^2- f]/(1-f)$

Figure 3.2: "Virtual" and "actual" uniform radial core swellings for a 2-D fast reactor model. The core radius swells from R_1 to R_2. The factor f is the unperturbed fuel and structure volume fraction.

swells, f decreases such that the homogenized fuel and coolant densities change by different factors. Refer back to Figure 3.2 for how to compute these factors.

We perform the diffusion simulation with a MATLAB [210] finite difference code integrated with the PETSc [200] and SLEPc [203] linear algebra packages. We dub this MATLAB-PETSc-SLEPc solver MaPS. See Appendix A for a detailed description with validation. MaPS uses the SLEPc generalized eigenvalue solver with Arnoldi or Krylov-Schur iterations. We utilize 33-group cross-sections produced by the code MC**2, which is tailored to process fast reactor cross-sections [187]. We subdivide each assembly in Figure 3.3 into 16 square mesh cells, each 4 cm x 4 cm.

We should note here that the "virtual density" theory involves altering material densities and macroscopic cross-sections but *not* microscopic cross-sections. In a uniform swelling core scenario, sodium flows into the core, causing the spectrum to soften very slightly. Thus, the microscopic group cross-sections also change very slightly. However, because spectral shifts are a second order effect, so are changes in group cross-sections. Thus, altering group cross-sections does *not* change the first derivative of eigenvalue. Consequently, we do not regenerate microscopic cross-sections when applying the "virtual density" theory. Doing so would be fruitless.

We perform the deterministic transport simulation with OpenMOC, an open-source method of characteristics (MOC) code developed at MIT [213]. We utilize the identical 33-group MC**2 cross-sections. We subdivide the fuel into square 1 cm x 1 cm flat source regions (FSRs). The track spacing is 0.5 cm with 16 azimuthal angles.

We perform the continuous energy Monte Carlo simulation with MCNP5 [202]. We utilize continuous energy cross-sections from ENDF/B-VII.0.

In each of these simulations, we compare the reactivity induced by an "actual" core

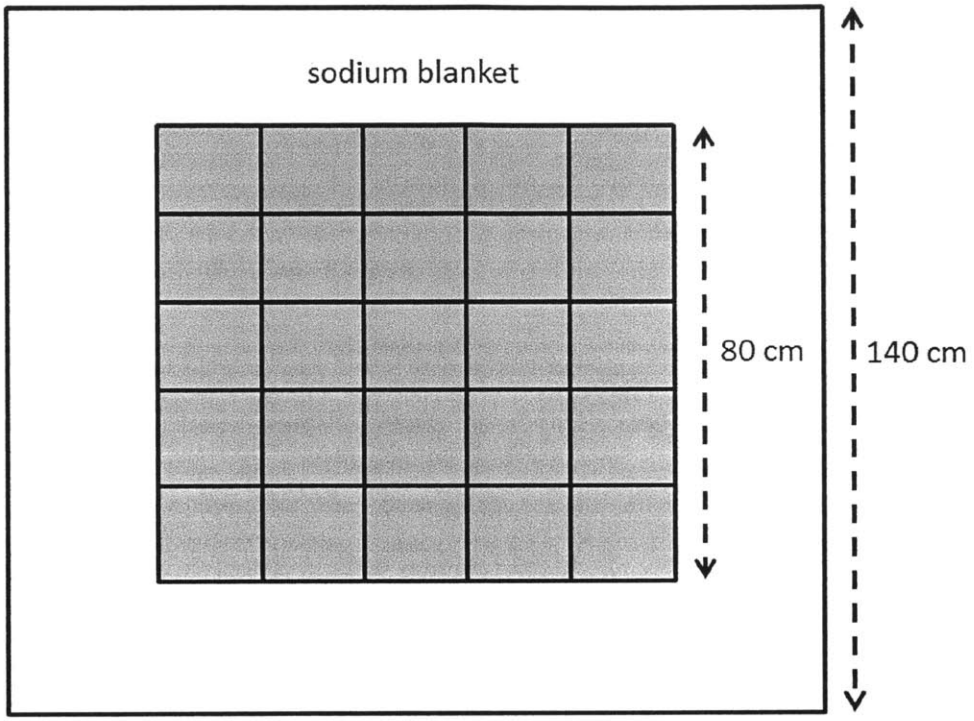

Figure 3.3: A 5x5 array of homogenized fuel assemblies immersed in a sodium blanket with zero-flux vacuum boundaries. This 2-D sodium fast reactor (SFR) fuel is enriched to approximately 10%, and the unperturbedfuel volume fraction within each assembly is $F = 0.62$. The dimensions of all assemblies (and the reflector) swell isotropically by the same factor, which causes F to decrease.

swelling to that induced by a "virtual" core swelling (as defined in Figure 3.2), and we see that they are precisely equivalent. Tables 3.1, 3.2, and 3.3 show the complete results. Indeed, the "virtual density" theory is universal, and we could reproduce similar results for any core geometry.

Note that the MCNP results agree surprisingly well. In all four cases, the k_{eff} uncertainty is 1 pcm, and all four sets of two k_{eff} values round to the same nearest pcm. Although this is not the most likely outcome, it is certainly not unlikely enough to warrant serious concern. We examine Shannon entropy to ensure that the fission source distribution has converged.

Figure 3.4 illustrates and classifies methods for solving neutron distributions in reactors. All methods can be classified as (1) either transport or diffusion and (2) either deterministic or stochastic. Thus, the Venn diagram in Figure 3.4 puts methods in one of three categories: deterministic diffusion, stochastic transport, or deterministic transport. The fourth category, stochastic diffusion, has not been implemented for obvious reasons. The point of this diagram

Table 3.1: Diffusion k_{eff} Values for "Actual" and "Virtual" Uniform Core Swellings (10^{-10} convergence)

R_2/R_1	actual k_{eff}	virtual k_{eff}	k_{eff} error (pcm)
1	1.0335072167	N/A	N/A
1.001	1.0332285619	1.0332285619	0.00000
1.01	1.0307346898	1.0307346898	0.00000
1.1	1.0071396247	1.0071396247	0.00000
2	0.8708997837	0.8708997837	0.00000

Table 3.2: MOC k_{eff} Values for "Actual" and "Virtual" Uniform Core Swellings (10^{-6} convergence)

R_2/R_1	actual k_{eff}	virtual k_{eff}	k_{eff} error (pcm)
1	1.041188	N/A	N/A
1.001	1.040907	1.040907	0.0
1.01	1.038388	1.038388	0.0
1.1	1.014440	1.014440	0.0
2	0.872294	0.872294	0.0

is to show that the "virtual density" theory is equally valid in all three categories, which encompass all methods people use to analyze neutrons in reactors.

Table 3.3: Monte Carlo k_{eff} Values for "Actual" and "Virtual" Uniform Core Swellings (4 billion histories)

R_2/R_1	actual k_{eff}	virtual k_{eff}	k_{eff} error (pcm)
1	1.03867 ± 0.00001	N/A	N/A
1.001	1.03841 ± 0.00001	1.03841 ± 0.00001	0 ± 1.4
1.01	1.03593 ± 0.00001	1.03593 ± 0.00001	0 ± 1.4
1.1	1.01257 ± 0.00001	1.01257 ± 0.00001	0 ± 1.4
2	0.87772 ± 0.00001	0.87772 ± 0.00001	0 ± 1.4

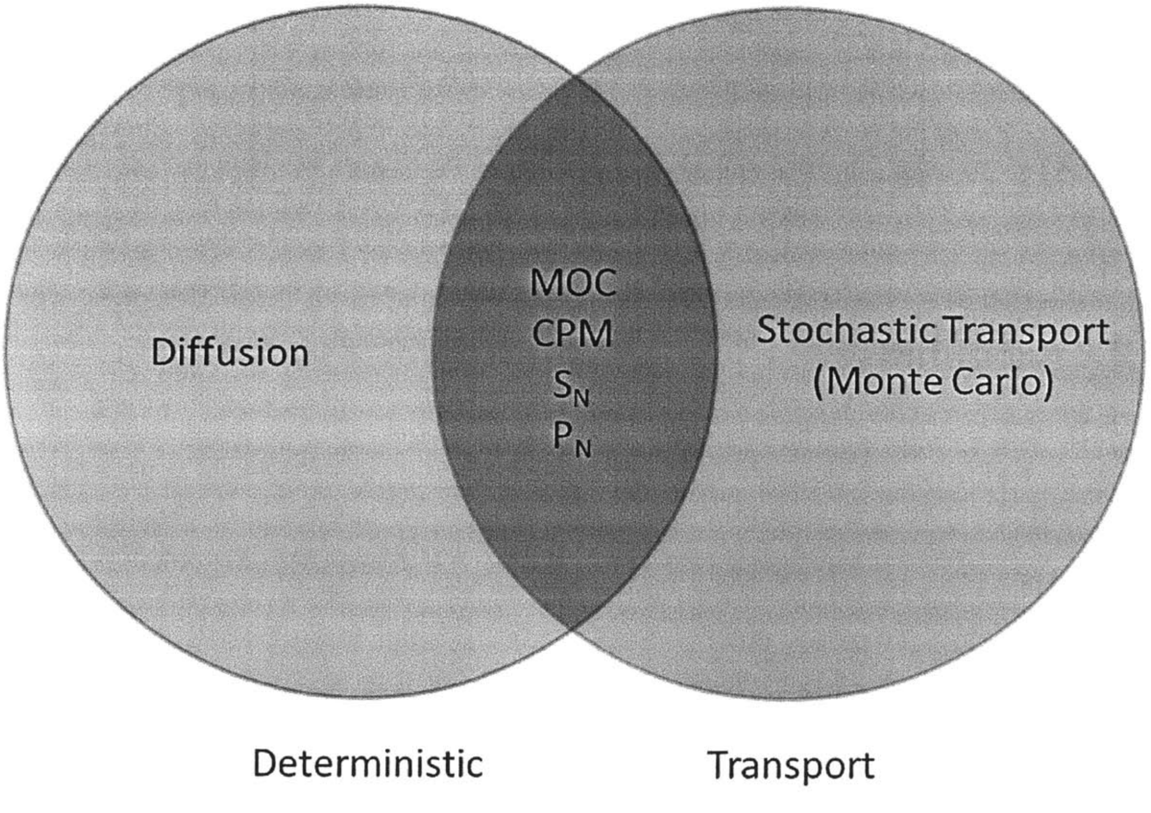

Figure 3.4: A Venn diagram of methods for solving neutron distributions in nuclear reactors.

3.4 Application to Perturbation Theory

Why does this "virtual density" theory matter? How is computing a "virtual" core swelling any more efficient than directly computing an "actual" core swelling? The advantage is twofold.

First, application of the "virtual density" theory does not require altering the computational mesh, which can be an arduous task for complex distortions. Spatial dimensions must always be discretized, but material densities occupy a continuous domain. Thus, altering material densities is tremendously simpler than altering spatial dimensions.

Second, the "virtual density" theory is expeditious if we wish to save time and effort by evaluating core swelling reactivity effects using perturbation theory. Material density perturbations are much more straightforward to evaluate than geometry perturbations, because they require nothing more than a classic (first or second order) perturbation theory calculation. This process is identical to computing the reactivity coefficient of a material density. No manipulation of the computational mesh is required, and no surface integrals are required.

3.4.1 Diffusion Theory Formalism

Although the "virtual density" theory is equally valid for deterministic transport and Monte Carlo, we will now proceed to narrow our scope to multigroup homogenized diffusion theory. This section builds on the diffusion theory and perturbation theory described in Sections 2.3.2 - 2.3.4. Although the uniform isotropic "virtual density" theory appears sporadically throughout prior literature, no perturbation formalism is ever laid out. Thus, we do so now.

Let us decompose the multigroup diffusion operators into four parts corresponding to fission $\hat{F}$, absorption $\hat{A}$, scattering or transfer $\hat{T}$, and leakage $\hat{L}$. Let the real and adjoint neutron fluxes be isotropic scalar functions ϕ and $\phi^\dagger$, respectively. In classic first order perturbation theory, we compute the adjoint-weighted average of each of these terms operating on the real flux to obtain the reactivity $\Delta\rho$. Here k is the unperturbed eigenvalue.

$$\Delta\rho = \frac{\left\langle \phi^\dagger \middle| \left(\frac{1}{k}d\hat{F} - d\hat{A} - d\hat{T} - d\hat{L}\right)\phi \right\rangle}{\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle} \tag{3.4}$$

Essentially, we have split the familiar neutron "loss" operator $\hat{M}$ into its three constituent parts:

$$d\hat{M} = d\hat{A} + d\hat{T} + d\hat{L} \tag{3.5}$$

In addition to performing adjoint weighting for small perturbations of these operators (such as $d\hat{F}$), we can also perform adjoint weighting for the whole operators (such as $\hat{F}$).

This is useful, as "virtual density" usually allows the perturbation magnitudes (the fractional density changes) to fall out of the spatial integral in Equation 3.4. Thus, one can pre-compute all spatial integrals based on unperturbed quantities. Evaluating these quantities for the $\hat{F}$, $\hat{A}$, and $\hat{T}$ operators is straightforward [188]:

$$\left\langle \phi^\dagger \middle| d\hat{F}\phi \right\rangle = \int dV \left[\sum_{g'=1}^{G} \chi_{g'}\phi^\dagger_{g'} \sum_{g=1}^{G} d\left(\nu_g \Sigma_{f,g}\right)\phi_g \right] \tag{3.6}$$

$$\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle = \int dV \left[\sum_{g'=1}^{G} \chi_{g'}\phi^\dagger_{g'} \sum_{g=1}^{G} \nu_g \Sigma_{f,g}\phi_g \right] \tag{3.7}$$

$$\left\langle \phi^\dagger \middle| d\hat{A}\phi \right\rangle = \int dV \left[\sum_{g=1}^{G} \phi^\dagger_g d\Sigma_{a,g}\phi_g \right] \tag{3.8}$$

$$\left\langle \phi^\dagger \middle| \hat{A}\phi \right\rangle = \int dV \left[\sum_{g=1}^{G} \phi^\dagger_g \Sigma_{a,g}\phi_g \right] \tag{3.9}$$

$$\left\langle \phi^\dagger \middle| d\hat{T}\phi \right\rangle = \int dV \left[\sum_{g=1}^{G} \sum_{g'=g+1}^{G} \left(\phi^\dagger_g - \phi^\dagger_{g'}\right) d\Sigma_{s,g\to g'}\phi_g \right] \tag{3.10}$$

$$\left\langle \phi^\dagger \middle| \hat{T}\phi \right\rangle = \int dV \left[\sum_{g=1}^{G} \sum_{g'=g+1}^{G} \left(\phi^\dagger_g - \phi^\dagger_{g'}\right) \Sigma_{s,g\to g'}\phi_g \right] \tag{3.11}$$

Here Σ_f, Σ_a, and Σ_s are the fission, absorption, and scattering macroscopic cross-sections. There are G energy groups. ν_g is the number of neutrons produced per fission induced by a neutron in group g, while $\chi_{g'}$ is the fraction of those fission-produced neutrons born in group g'.

Note that if we uniformly change a material density N by the small fraction $\epsilon = dN/N$, then this factor simply falls out of the integrals in Equations 3.6 - 3.11. Now we can write

$$\left\langle \phi^\dagger \middle| d\hat{F}\phi \right\rangle = \epsilon \left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle \tag{3.12}$$

$$\left\langle \phi^\dagger \middle| d\hat{A}\phi \right\rangle = \epsilon \left\langle \phi^\dagger \middle| \hat{A}\phi \right\rangle \tag{3.13}$$

$$\left\langle \phi^\dagger \middle| d\hat{T}\phi \right\rangle = \epsilon \left\langle \phi^\dagger \middle| \hat{T}\phi \right\rangle \tag{3.14}$$

Evaluating $d\hat{L}$ is a bit more complicated and requires application of the divergence theorem [188]. We begin with direct integration of the leakage term:

$$\left\langle \phi^\dagger \middle| d\hat{L}\phi \right\rangle = -\int dV \left[\sum_{g=1}^{G} \left(\phi^\dagger_g \nabla \cdot dD_g \nabla \phi_g\right) \right] \tag{3.15}$$

Let the diffusion coefficient be D. We can apply the divergence theorem once to obtain

$$\left\langle \phi^\dagger \middle| d\hat{L}\phi \right\rangle = -\int d\vec{S} \cdot \left[\sum_{g=1}^{G} \left(\phi_g^\dagger \nabla \phi_g \right) dD_g \right] + \int dV \left[\sum_{g=1}^{G} \left(\nabla \phi_g^\dagger \cdot \nabla \phi_g \right) dD_g \right] \tag{3.16}$$

Now we have a surface term and a volume term. The surface term is zero for one of two reasons: (1) the adjoint flux vanishes on the outer reactor surface and (2) dD_g vanishes beyond the extent of the perturbation. When either of these cases applies, we need only evaluate the volume term, which contains the dot product of the real and adjoint flux gradients. Thus, the entire first term is zero.

Before proceeding, we can linearize the perturbed diffusion coefficient so that it is proportional to perturbed material density. Here let Σ_{tr} be the macroscopic transport cross-section.

$$dD_g = -\frac{d\Sigma_{tr,g}}{3\Sigma_{tr,g}^2} \tag{3.17}$$

When perturbing the leakage term, the only tangible perturbation resides in the transport cross-section $d\Sigma_{tr}$. Thus, the first order reactivity due to a material density perturbation will always be linear as a function of material density. Note that Equation 3.17 introduces an (important) extra minus sign to the leakage term. Intuitively, this minus sign shows that an *increase* in Σ_{tr} corresponds to a *decrease* in D.

Now the entire leakage perturbation term is

$$\left\langle \phi^\dagger \middle| d\hat{L}\phi \right\rangle = -\int dV \left[\sum_{g=1}^{G} \left(\nabla \phi_g^\dagger \cdot \nabla \phi_g \right) \frac{d\Sigma_{tr,g}}{3\Sigma_{tr,g}^2} \right] \tag{3.18}$$

$$\left\langle \phi^\dagger \middle| \hat{L}\phi \right\rangle = \int dV \left[\sum_{g=1}^{G} \left(\nabla \phi_g^\dagger \cdot \nabla \phi_g \right) D_g \right] \tag{3.19}$$

Application of Equation 3.17 introduced a minus sign in Equation 3.18 that does not exist in Equation 3.19. Now if we again define $\epsilon = dN/N = d\Sigma_{tr}/\Sigma_{tr}$, then we can write

$$\left\langle \phi^\dagger \middle| d\hat{L}\phi \right\rangle = -\epsilon \left\langle \phi^\dagger \middle| \hat{L}\phi \right\rangle \tag{3.20}$$

The minus sign here in Equation 3.20 is not present in Equations 3.12 - 3.14. This is crucial, as we shall see in the next section.

In Russian literature, Equations 3.18 and 3.19 are often expressed in terms of neutron currents [91,95]:

$$\left\langle \phi^{\dagger} \middle| d\hat{L}\phi \right\rangle = \int dV \left[\sum_{g=1}^{G} \left(\vec{J}_g^{\dagger} \cdot \vec{J}_g \right) 3d\Sigma_{tr,g} \right] \tag{3.21}$$

$$\left\langle \phi^{\dagger} \middle| \hat{L}\phi \right\rangle = -\int dV \left[\sum_{g=1}^{G} \left(\vec{J}_g^{\dagger} \cdot \vec{J}_g \right) 3\Sigma_{tr,g} \right] \tag{3.22}$$

Here the real and adjoint currents are proportional to the real and adjoint flux gradients.

$$\vec{J}_g = -D\nabla\phi_g \tag{3.23}$$

$$\vec{J}_g^{\dagger} = D\nabla\phi_g^{\dagger} \tag{3.24}$$

The adjoint current has no minus sign, which is consistent with the "backward" flow of adjoint current in transport theory.

3.4.2 The Mathematical Definition of an Isotropic "Virtual Density" Perturbation

We can always express an isotropic whole-core "virtual density" perturbation in one of three ways: as purely a spatial perturbation, as purely a spectral perturbation, or as a combination of both. We consider the $\hat{L}$ operator to determine the *spatial* components of perturbations, while the other operators determine the *spectral* components of perturbations. Here ϵ again represents the perturbation magnitude - the small fractional change in a given material density. The first order "virtual density" perturbation reactivity is

$$\Delta\rho = \epsilon \frac{\left\langle \phi^{\dagger} \middle| \left(\frac{1}{k}\hat{F} - \hat{A} - \hat{T} + \hat{L} \right) \phi \right\rangle}{\left\langle \phi^{\dagger} \middle| \hat{F}\phi \right\rangle} = 2\epsilon \frac{\left\langle \phi^{\dagger} \middle| \hat{L}\phi \right\rangle}{\left\langle \phi^{\dagger} \middle| \hat{F}\phi \right\rangle} = 2\epsilon \frac{\left\langle \phi^{\dagger} \middle| \left(\frac{1}{k}\hat{F} - \hat{A} - \hat{T} \right) \phi \right\rangle}{\left\langle \phi^{\dagger} \middle| \hat{F}\phi \right\rangle} \tag{3.25}$$

These three expressions are precisely equivalent, but *only* for uniform whole-core perturbations. Note the "+" sign in front of the $\hat{L}$ operator in the first expression, which follows from Equations 3.4 and 3.20. These equalities (especially the conspicuous factor of 2) may seem puzzling, but we can elucidate them with some simple mathematics based on the formalism we developed in the previous section. The basic diffusion neutron balance is

$$\left(\hat{A} + \hat{T} + \hat{L} \right) \phi = \frac{1}{k}\hat{F}\phi \tag{3.26}$$

Now simply rearrange terms to express leakage in terms of the other three operators.

$$\hat{L}\phi = \left(\frac{1}{k}\hat{F} - \hat{A} - \hat{T}\right)\phi \tag{3.27}$$

Now consider a uniform, whole-core density *increase* of magnitude $\epsilon = dN/N$ encapsulated within $d\hat{F}$, $d\hat{A}$, $d\hat{T}$, and $d\hat{L}$.

$$-d\hat{L}\phi = \left(\frac{1}{k}d\hat{F} - d\hat{A} - d\hat{T}\right)\phi \tag{3.28}$$

Note that a minus sign has appeared in front of $d\hat{L}$, which has the form of Equation 3.15. Incorporating a density *increase* to the operators $\hat{F}$, $\hat{A}$, and $\hat{T}$ will result in positive signs in front of $d\hat{F}$, $d\hat{A}$, and $d\hat{T}$. However, because $\hat{L}$ is inversely proportional to density, $d\hat{L}$ will have a minus sign in front of it.

Now weight each perturbation by the adjoint and integrate over all space (the whole reactor) and energy.

$$-\left\langle \phi^\dagger \middle| d\hat{L}\phi \right\rangle = \left\langle \phi^\dagger \middle| \left(\frac{1}{k}d\hat{F} - d\hat{A} - d\hat{T}\right)\phi \right\rangle \tag{3.29}$$

Now apply Equations 3.12, 3.13, 3.14, and 3.20 to each respective operator perturbation.

$$\epsilon\left\langle \phi^\dagger \middle| \hat{L}\phi \right\rangle = \epsilon\left\langle \phi^\dagger \middle| \left(\frac{1}{k}\hat{F} - \hat{A} - \hat{T}\right)\phi \right\rangle \tag{3.30}$$

Equations 3.29 and 3.30 show that we have introduced a minus sign in the leakage term but not in any other terms. We can apply Equation 3.30 directly to the left-most expression in Equation 3.25 in order to obtain the other two expressions in Equation 3.25. This shows that for uniform, whole-core perturbations, the *spatial* and *spectral* components of reactivity worth are precisely equal. Every uniform geometry change directly induces a spatial reactivity, but that spatial reactivity is balanced (throughout the whole core) by an equal spectral reactivity.

Intuitively, we can consider a uniform core swelling in which leakage increases. When neutrons are more likely to leak, they are less likely to be absorbed or scatter down to lower energies. Thus, the *spatial* reactivity due to increased leakage induces a *spectral* reactivity due to decreased absorption and spectral hardening. The remarkable truth here is that these two reactivities are precisely equal.

For perturbations within core subregions, the surface term in Equation 3.16 is non-zero and invalidates the equalities in Equation 3.25. In these cases, only the first expression in Equation 3.25 (containing both spectral and leakage operators) is valid.

The perturbation magnitude ϵ in Equation 3.25 depends on the particular expansion or swelling type. Throughout the remainder of this paper, let f be the fraction by which the reactor linear length scales increase. For example, a 1% increase in core dimensions corresponds to $f = 1.01$.

$$\epsilon = \frac{dN}{N} = \begin{cases} f - 1 & \text{virtual expansion of any reactor} \\ 0 & \text{virtual swelling of a 1-D reactor} \\ \frac{1}{f} - 1 & \text{virtual swelling of a 2-D reactor} \\ \frac{1}{f^2} - 1 & \text{virtual swelling of a 3-D reactor} \\ \frac{1}{f^{Q-1}} - 1 & \text{virtual swelling of a Q-D reactor} \end{cases} \tag{3.31}$$

Equation 3.31 fully specifies how to define material density perturbations for isotropic expansions or swellings of a reactor. Again, if the isotropic expansion or swelling is uniform throughout the whole core, all three expressions in Equation 3.25 are valid. However, if the isotropic expansion or swelling occurs only within a core subregion (such as one slab in an array of infinite 1-D slabs or one half of a spherical reactor), only the first expression in Equation 3.25 is valid.

Finally, Equation 3.25 need not be applied evenhandedly to every nuclide in the core. As shown in Figure 3.2, we can apply it separately to fuel and coolant nuclides. Each nuclide would simply have a different value of ϵ. However, even though we can treat nuclides separately, the perturbation on *each* nuclide must be uniform and whole-core for the equalities in Equation 3.25 to apply.

3.4.3 Uniform Isotropic "Virtual Density" Shorthand Notation

For convenience, let us define the scalar values S and L to be

$$S = \frac{\left\langle \phi^\dagger \middle| \left(\frac{1}{k}\hat{F} - \hat{A} - \hat{T}\right)\phi \right\rangle}{\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle} \tag{3.32}$$

$$L = \frac{\left\langle \phi^\dagger \middle| \hat{L}\phi \right\rangle}{\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle} \tag{3.33}$$

Here S represents the total reactivity weight of the "spectral" operators: fission, absorption, and scattering. L represents the total reactivity weight of leakage. Note that the fission perturbation denominators in both S and L are always integrated throughout the whole core, while the numerators can be integrated over any core subregion. Thus, S and L are scalar values with spatial dependence, since they can be evaluated for each mesh cell in a finite difference solution. When they are each integrated throughout the whole core, Equation 3.25 implies

$$S = L \tag{3.34}$$

This simple equality is a cornerstone of "virtual density" theory, and it will become even more paramount when we examine anisotropic cases in the ensuing chapters. Essentially, it elegantly expresses our earlier assertion that spectral and leakage reactivities balance each other throughout a whole core volume. When a core swells uniformly, the spectral and leakage reactivities will be equal in magnitude - each will constitute precisely half the total reactivity.

3.5 Analytic Examples

Now that we have outlined the uniform isotropic "virtual density" formalism in diffusion theory, we can begin with a few very simple analytic examples - homogenous slabs and spheres. Although all these examples are one-group diffusion, extension to multigroup diffusion amounts to a straightforward sum of the perturbation theory numerator and denominator over all groups. Although these simple examples do not constitute full validation of the "virtual density" theory, they serve to elucidate the physical phenomena at work.

3.5.1 1-D Slab Expansion

First let us consider a homogenous, infinite 1-D slab that is critical. We know from intuition that swelling this slab will result in precisely zero reactivity - the neutron mean free path will increase proportional to the slab thickness, and leakage rates will remain constant. So applying the "virtual density" theory to slab swelling is fruitless.

However, let us consider *expanding* this 1-D slab, which *will* alter the eigenvalue. Let the slab width be a, and let $x = 0$ correspond to the slab center. The simple neutron balance is

$$-D\nabla^2\phi(x) + \Sigma_a\phi(x) = \frac{1}{k}\nu\Sigma_f\phi(x) \tag{3.35}$$

The eigenvalue is

$$k = \frac{\nu\Sigma_f}{DB^2 + \Sigma_a} \tag{3.36}$$

The flux distribution is

$$\phi(x) = \cos\left(\frac{\pi x}{a}\right) \tag{3.37}$$

We define the geometric buckling as

$$-\nabla^2\phi(x) = B^2\phi(x) = \left(\frac{\pi}{a}\right)^2\phi(x) \tag{3.38}$$

Now consider a uniform expansion of this slab by a factor f so that its new width is a'.

$$f = \frac{a'}{a} \tag{3.39}$$

Now the exact reactivity increase is

$$\Delta\rho = \tfrac{1}{k} - \tfrac{1}{k'} \tag{3.40}$$

$$= \tfrac{DB^2+\Sigma_a}{\nu\Sigma_f} - \tfrac{DB'^2+\Sigma_a}{\nu\Sigma_f} \tag{3.41}$$

$$= \tfrac{D}{\nu\Sigma_f}\left(B^2 - B'^2\right) \tag{3.42}$$

$$= \tfrac{D}{\nu\Sigma_f}\left(\tfrac{\pi}{a}\right)^2\left(1 - \tfrac{1}{f^2}\right) \tag{3.43}$$

The first derivative of this reactivity increase with respect to f is

$$\left[\frac{d(\Delta\rho)}{df}\right]_{f=1} = \frac{2D}{\nu\Sigma_f}\left(\frac{\pi}{a}\right)^2 \tag{3.44}$$

This is the exact first derivative. If we perform first order perturbation theory correctly, we should obtain a result that is precisely equal to this.

First we define a "virtual density" perturbation that is equivalent to the specified expansion. Instead of increasing the slab width by the factor f, we simply increase all the macroscopic cross-sections Σ by the factor f.

$$d\Sigma = \Sigma' - \Sigma = \Sigma(f-1) \tag{3.45}$$

The flux gradient is

$$\nabla\phi = -\frac{\pi}{a}\sin\left(\frac{\pi x}{a}\right)\hat{x} \tag{3.46}$$

Here $\hat{x}$ denotes a unit vector in the x direction. Now the numerator in first order perturbation theory is simple to compute. Remember that the real and adjoint fluxes are equal in one-group diffusion.

$$\left\langle \phi^\dagger \middle| \left(\tfrac{1}{k}d\hat{F} - d\hat{M}\right)\phi \right\rangle = \tag{3.47}$$

$$= \int_{-a/2}^{a/2} dx \left[(\nabla\phi(x))^2 \tfrac{f-1}{3\Sigma_{tr}} + \left(\tfrac{1}{k}\nu\Sigma_f - \Sigma_a\right)(f-1)(\phi(x))^2\right] \tag{3.48}$$

$$= (f-1)\left[D\left(\tfrac{\pi}{a}\right)^2\tfrac{a}{2} + \left(\Sigma_a - \tfrac{1}{k}\nu\Sigma_f\right)\tfrac{a}{2}\right] \tag{3.49}$$

The perturbation theory denominator is even simpler to compute.

$$\left\langle \phi^{\dagger} \middle| \hat{F}\phi \right\rangle = \int_{-a/2}^{a/2} dx\, \nu\Sigma_f \left(\phi(x)\right)^2 = \nu\Sigma_f \frac{a}{2} \tag{3.50}$$

So now the first order reactivity is simply the ratio of the numerator to the denominator.f

$$\Delta\rho = \frac{(f-1)\left[D\left(\frac{\pi}{a}\right)^2 + \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)\right]}{\nu\Sigma_f} \tag{3.51}$$

This is linear in f, and its slope is

$$\left[\frac{d(\Delta\rho)}{df}\right]_{f=1} = \frac{D\left(\frac{\pi}{a}\right)^2 + \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)}{\nu\Sigma_f} \tag{3.52}$$

Now there are two terms here - a spatial term and a spectral term. However, neutron balance in diffusion theory requires that these two terms be precisely equal.

$$\frac{1}{k}\nu\Sigma_f - \Sigma_a = DB^2 \tag{3.53}$$

So now we can express the reactivity one of two ways, either as twice the leakage term or twice the spectral term.

$$\left[\frac{d(\Delta\rho)}{df}\right]_{f=1} = \frac{2D}{\nu\Sigma_f}\left(\frac{\pi}{a}\right)^2 = \frac{2\left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)}{\nu\Sigma_f} \tag{3.54}$$

Note that this is exactly equivalent to the actual reactivity coefficient that we obtained in Equation 3.44. So this proves that "virtual density" perturbation theory works for a 1-D homogenous slab case. Also note that Equations 3.52 and 3.54 can be written in shorthand notation as

$$\left[\frac{d(\Delta\rho)}{df}\right]_{f=1} = L + S = 2L = 2S \tag{3.55}$$

3.5.2 Sphere Expansion

Now we can perform the same "virtual density" verification for a bare sphere reactor of radius R. The one-group flux distribution is

$$\phi(r) = \frac{1}{r}\sin\left(\frac{\pi r}{R}\right) \tag{3.56}$$

The geometric buckling is

$$B^2 = \left(\frac{\pi}{R}\right)^2 \tag{3.57}$$

Consider a uniform expansion in the sphere radius by a factor f.

$$f = \frac{R'}{R} \tag{3.58}$$

Now the exact reactivity is

$$\Delta\rho = \frac{1}{k} - \frac{1}{k'} = \frac{D}{\nu\Sigma_f}\left(B^2 - B'^2\right) = \frac{D}{\nu\Sigma_f}\left[\left(\frac{\pi}{R}\right)^2 - \left(\frac{\pi}{Rf}\right)^2\right] \tag{3.59}$$

The first derivative of this shift with respect to f is

$$\left[\frac{d(\Delta\rho)}{df}\right]_{f=1} = \frac{2D}{\nu\Sigma_f}\left(\frac{\pi}{R}\right)^2 \tag{3.60}$$

Now we can derive the same expression using a "virtual density" perturbation.

$$d\Sigma = \Sigma' - \Sigma = \Sigma(f-1) \tag{3.61}$$

The flux gradient is

$$\nabla\phi = \left[\frac{\pi}{rR}\cos\left(\frac{\pi r}{R}\right) - \frac{1}{r^2}\sin\left(\frac{\pi r}{R}\right)\right]\hat{r} \tag{3.62}$$

The first order perturbation numerator is

$$\left\langle \phi^\dagger \middle| \left(\tfrac{1}{k}d\hat{F} - d\hat{M}\right)\phi \right\rangle \tag{3.63}$$

$$= \textstyle\int_0^R 4\pi r^2 dr \, \left[\left(\nabla\phi(r)\right)^2 \frac{f-1}{3\Sigma_{tr}} + \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)(f-1)\left(\phi(r)\right)^2\right] \tag{3.64}$$

$$= 4\pi(f-1)\left[D\left(\tfrac{\pi^2}{2R}\right)\tfrac{R}{2} + \left(\Sigma_a - \tfrac{1}{k}\nu\Sigma_f\right)\tfrac{R}{2}\right] \tag{3.65}$$

The first order perturbation denominator is

$$\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle = \int_0^R 4\pi r^2 dr \, \nu\Sigma_f\left(\phi(r)\right)^2 = 4\pi\nu\Sigma_f\frac{R}{2} \tag{3.66}$$

The first order reactivity is

$$\Delta\rho = \frac{(f-1)\left[D\left(\frac{\pi}{R}\right)^2 + \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)\right]}{\nu\Sigma_f} \tag{3.67}$$

Just as in the 1-D slab case, the first derivative of Equation 3.67 can be represented as purely spatial, purely spectral, or half spatial and half spectral. The diffusion equation requires that the spatial and spectral parts be equal.

$$\left[\frac{d(\Delta\rho)}{df}\right]_{f=1} = \frac{D\left(\frac{\pi}{R}\right)^2 + \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)}{\nu\Sigma_f} = \frac{2D}{\nu\Sigma_f}\left(\frac{\pi}{R}\right)^2 = \frac{2\left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)}{\nu\Sigma_f} \tag{3.68}$$

This is equivalent to the exact value in Equation 3.60. The fascinating fact here is that one can use the "virtual density" theory to evaluate a geometry perturbation without even utilizing the leakage term. The right-most expression in Equation 3.68 contains no diffusion coefficient and no flux gradients. Thus, we can capture the reactivity effect due to a geometry change by perturbing only fission, absorption, and scattering. Also note that Equation 3.68 can be written in shorthand notation as

$$\left[\frac{d(\Delta\rho)}{df}\right]_{f=1} = L + S = 2L = 2S \tag{3.69}$$

This is the identical result we obtained for the slab expansion case in Equation 3.55.

3.5.3 Sphere Swelling

Now let us consider the case of sphere *swelling*, in which mass is conserved as the sphere radius increases. Unlike a swelling 1-D slab, a swelling sphere *will* experience a decrease in reactivity.

Here the unperturbed flux solution is the same. Due to conservation of mass, we must alter more than only R to evaluate the actual reactivity. We must also alter the material densities, which manifest themselves in the macroscopic cross-sections and the diffusion coefficient.

$$f = \frac{R'}{R} \tag{3.70}$$

$$B'^2 = B^2/f^2 \tag{3.71}$$

$$\Sigma' = \Sigma/f^3 \tag{3.72}$$

$$D' = Df^3 \tag{3.73}$$

The exact reactivity is

$$\Delta\rho \qquad = \tfrac{1}{k} - \tfrac{1}{k'} = \tfrac{DB^2+\Sigma_a}{\nu\Sigma_f} - \tfrac{D'B'^2+\Sigma'_a}{\nu\Sigma'_f} \tag{3.74}$$

$$= \tfrac{DB^2+\Sigma_a}{\nu\Sigma_f} - \tfrac{DB^2f+\Sigma_a/f^3}{\nu\Sigma_f/f^3} = \tfrac{DB^2+\Sigma_a}{\nu\Sigma_f} - \tfrac{DB^2f^4+\Sigma_a}{\nu\Sigma_f} \tag{3.75}$$

$$= \tfrac{D}{\nu\Sigma_f}\left(1 - f^4\right) \tag{3.76}$$

The first derivative of this reactivity with respect to f is

$$\left[\frac{d(\Delta\rho)}{df}\right]_{f=1} = -\frac{4D}{\nu\Sigma_f}\left(\frac{\pi}{R}\right)^2 \tag{3.77}$$

Now let us define the equivalent "virtual density" perturbation. After dividing all material densities by f^3 to reflect the density decrease, we multiply them by f to shorten the neutron mean free path by a factor of f.

$$d\Sigma = \Sigma' - \Sigma = \Sigma\left(\frac{1}{f^2} - 1\right) \tag{3.78}$$

Performing the same integrals as in the sphere expansion case leaves us with a first order reactivity:

$$\Delta\rho = \left(\frac{1}{f^2} - 1\right)\frac{\left[D\left(\frac{\pi}{R}\right)^2 + \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)\right]}{\nu\Sigma_f} \tag{3.79}$$

The first derivative of this reactivity is

$$\left[\frac{d(\Delta\rho)}{df}\right]_{f=1} = -2\frac{D\left(\frac{\pi}{R}\right)^2 + \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)}{\nu\Sigma_f} = -\frac{4D}{\nu\Sigma_f}\left(\frac{\pi}{R}\right)^2 = -\frac{4\left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)}{\nu\Sigma_f} \tag{3.80}$$

This is equal to the exact solution in Equation 3.77. Also note that we can again express this reactivity effect without utilizing the leakage term and without computing the flux gradient. Also, we can again conveniently express this solution (Equation 3.80) in shorthand notation:

$$\left[\frac{d(\Delta\rho)}{df}\right]_{f=1} = -2(L+S) = -4L = -4S \tag{3.81}$$

This is a *swelling* scenario, and so the result here is different than in the expansion cases.

3.6 Numeric Demonstration: Uniform Isotropic Swelling of a 2-D Cartesian SFR Core

Now that we have developed the formalism in diffusion theory and worked out a few simple analytic examples, we will proceed to validate uniform isotropic "virtual density" perturbation theory using numeric multigroup diffusion. We use the same 2-D sodium fast reactor (SFR) described in Section 3.3.5 and shown in Figure 3.3, which we used to show equivalence between diffusion, deterministic transport, and Monte Carlo. We also use the same MaPS diffusion code with 33-group MC**2 cross-sections [187]. We define the "virtual density" perturbation as presented in Figure 3.2.

Figure 3.5 shows results from a first order perturbation theory calculation of a "virtual" uniform radial core swelling. First order estimates can predict the first derivative of eigenvalue with respect to core swelling (213 pcm per percent) to within 0.1%.

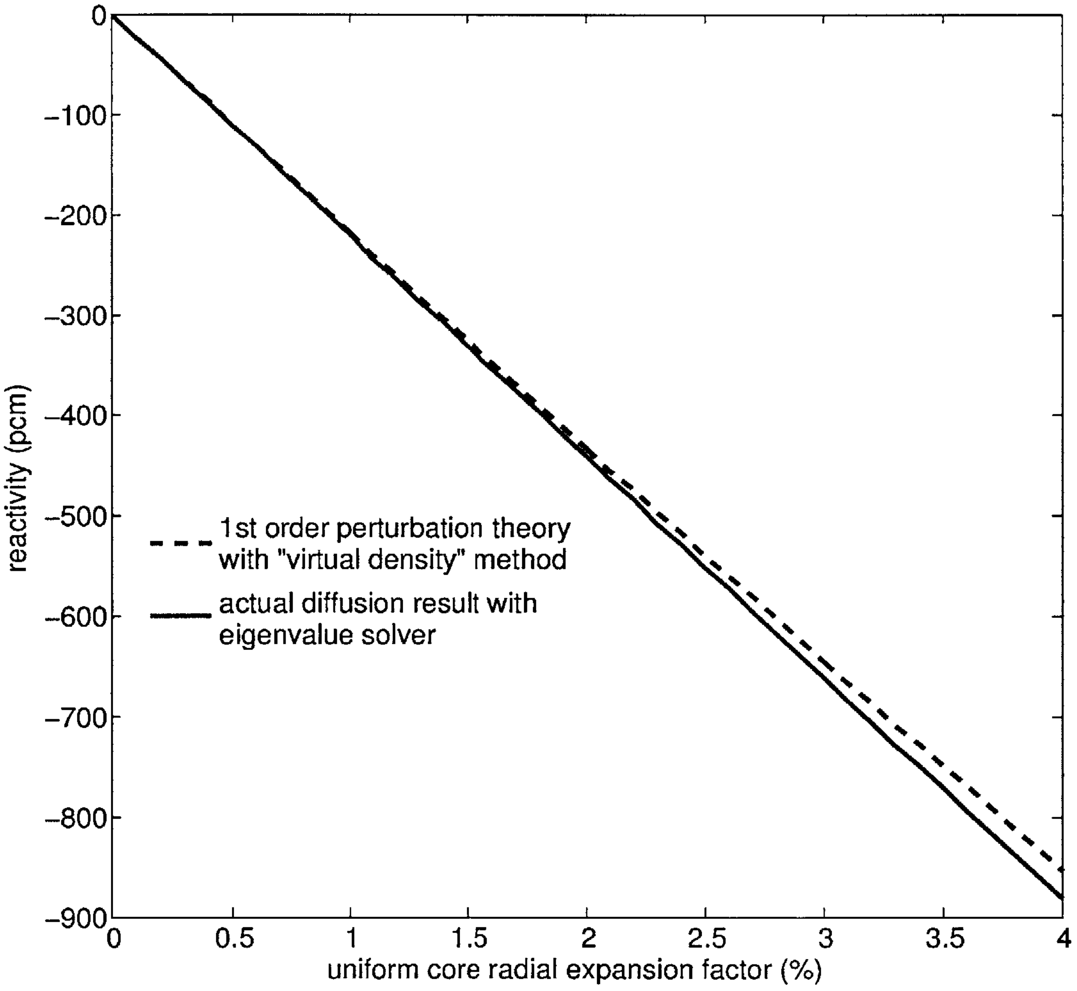

Figure 3.5: "Virtual density" perturbation theory prediction of the core radial swelling reactivity coefficient. First order perturbation theory can predict this coefficient (213 pcm per percent) to within 0.1%.

3.7 Numeric Demonstration: Uniform Isotropic Swelling of a 2-D Triangular FFTF Benchmark

Now let us test the "virtual density" perturbation theory for the Fast Flux Test Facility (FFTF) benchmark [146] with homogenized assemblies in 2-D hexagonal geometry using MaPS. In order to construct this 2-D model, we obtain material densities from the FFTF axial midplane. See the FFTF benchmark description and core map in Appendix G. Also see Figure 11.3, which shows the FFTF pin power distribution with three "lobes".

We compare the reactivity effects due to directly computing an "actual" swelling with those predicted by applying first order perturbation theory to a "virtual" swelling. Figure 3.6 shows the first order perturbation theory prediction of this 2-D FFTF uniform radial swelling. Figure 3.7 shows the error between first order perturbation theory and the actual diffusion eigenvalue result. Essentially, one can predict the reactivity due to an $x\%$ radial core swelling to less than $x\%$. All the diffusion meshes here are the same with six triangular mesh cells per assembly.

We should compare these results to those obtained by Knutson and his collaborators in the early 1980s [101,102]. They attempted to predict the reactivity due to a 100 mil = 1/10 inch uniform radial swelling of FFTF using their model described in Section 2.6.1. This very small shift corresponds to a radial increase of about 0.2%. They could use first order perturbation theory to predict their diffusion eigenvalue solution to no less than 12% error. They subsequently perform analysis to conclude that nearly all of this error arises from spatial effects rather than spectral effects. First order perturbation theory uses the unperturbed spatial and spectral flux distributions. Material density perturbations are simple, because they include only spectral error. Geometry perturbations, however, include both spectral error and (usually much larger) spatial error. However, because the "virtual density" method essentially converts geometric perturbations into density perturbations, it completely eliminates the spatial error, leaving only the (usually much smaller) spectral error. In this core swelling case, the spectral error arises from the flow of sodium into the core, which softens the spectrum very slightly. Since the "virtual density" theory eliminates the spatial error, we are left with only this tiny spectral shift as the source of error. Thus, for the same 100 mil radial swelling of FFTF, we can achieve 0.13% error with first order "virtual density" perturbation theory.

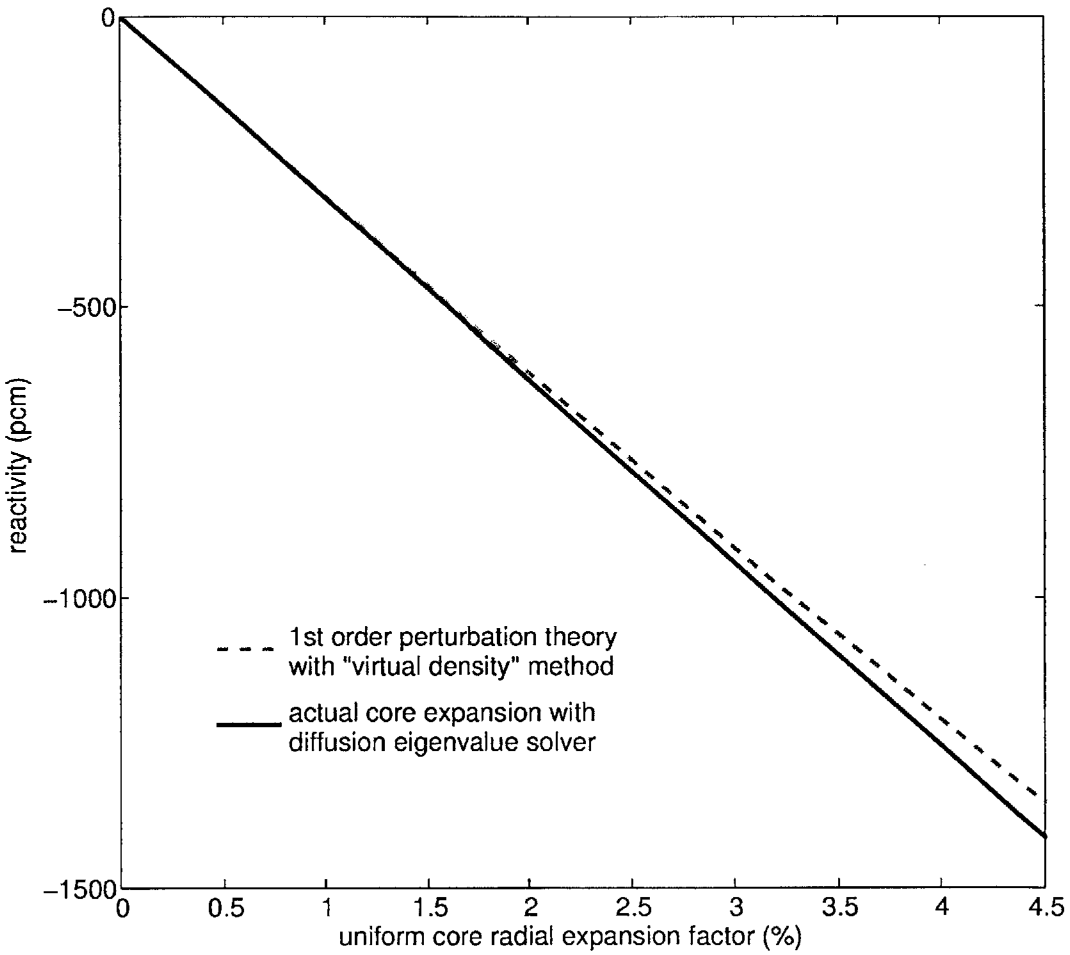

Figure 3.6: Reactivity as a function of uniform core radial swelling for a 2-D FFTF benchmark. First order perturbation theory (using "virtual density" theory) can predict the reactivity to 0.13%.

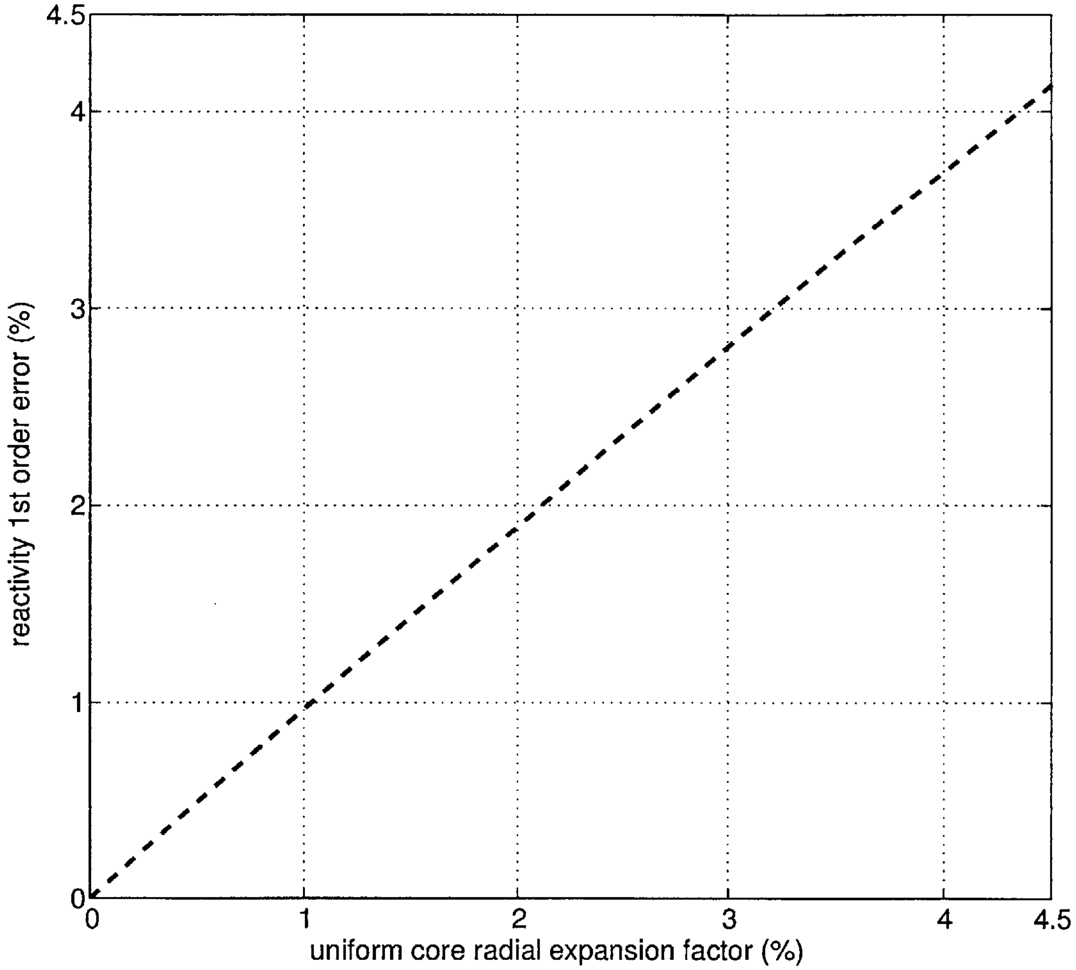

Figure 3.7: First order error in reactivity due to uniform radial swelling of the 2-D FFTF benchmark. This corresponds to the data in Figure 3.6.

3.8 Summary

We have conceived diffusion theory formalism for the "virtual density" theory of neutronics, which converts geometric perturbations into precisely equivalent material density perturbations. Here we have treated only uniform isotropic cases, which apply to (1) uniform isotropic swellings and expansions of whole cores and (2) arbitrary geometry perturbations in 1-D problems. We numerically verified the theory for a simple 2-D Cartesian sodium fast reactor (SFR) model and a 2-D model of the Fast Flux Test Facility (FFTF).

In the next chapter, we will introduce the uniform *an*isotropic "virtual density" theory, which is applicable to a host of localized swellings, expansions, and distortions in 2-D and 3-D cores.

4 A Review and Independent Derivation of Uniform *An*isotropic "Virtual Density" Theory

4.1 Chapter Abstract

We unearth and review the uniform anisotropic "virtual density" theory described in old Soviet and Russian literature, and we independently derive its formalism via anisotropic diffusion. We demonstrate this uniform anisotropic formalism for simple analytic solutions. Finally, we explain the unsolved problem of non-uniformity. Refer back to Figure 3.1 for a conceptual illustration of uniform isotropic and uniform anisotropic "virtual density" cases.

4.2 Introduction

Previously, we described the basic concept of "virtual density" and laid out the most simple version of its formalism for uniform isotropic swellings and expansions. However, the uniform isotropic "virtual density" theory is extremely limited, and it is not practically applicable to full-core 3-D reactor design. Real, heterogeneous reactors do not swell isotropically! Instead, reactors swell radially and axially with different coefficients. Thus, in order to apply the "virtual density" theory to any practical distortion scenarios in real reactors, we must allow neutrons to diffuse at different rates in different directions.

4.3 Anisotropic Diffusion

Physically, neutron diffusion is isotropic. A neutron traveling through a *homogenous* medium will have the same mean free path (and collision probabilities) regardless of its direction. It follows that a neutron in any homogenous region within any reactor will always have the same collision probability per unit length in *any* direction.

However, we don't always model reactors by breaking them down into tiny regions that are truly homogenous. Instead, we simplify models by amalgamating numerous heterogeneous regions into a single homogenous region. This process is called *homogenization*. In fast reactors, long mean free paths allow for homogenization of whole assembly blocks.

Unfortunately, neglecting pin detail via assembly homogenization also neglects the directional preference of neutron streaming. Neutrons have much longer mean free paths in sodium than in fuel or structure, so they can "stream" axially between fuel pins. When a model contains pin-level geometry, it can capture this effect. However, when a model contains homogenized assemblies, this effect is lost.

However, nuclear engineers often correct their homogenized models to regain much of this lost effect. One correction is anisotropic diffusion, which employs directional diffusion coefficients. Even though neutron diffusion is isotropic, we can pretend that it is anisotropic in order to capture heterogeneous streaming effects within a homogenized region. One might aptly characterize this as a "transport correction" to diffusion.

4.3.1 A Brief Literature Review

Behrens first studied neutronic effects due to "holes" within reactors as early as the late 1940s [67]. He developed equations to correct homogenous transport theory to account for neutron streaming through these holes - regions in which the neutron mean free path is much longer than it is throughout most of the core [68]. In the early 1960s, Carter improved upon Behrens's work, which he showed to be sufficient for streaming parallel along cylindrical holes but insufficient for streaming transversely across those holes [70].

Benoist developed the first general formalism for anisotropic diffusion in the late 1950s [69]. His equations are equivalent to those of Behrens to lowest order [70]. Nearly a decade later, Benoist improved upon his model using transport collision probabilities to compute diffusion coefficients to account for streaming effects [71].

During the 1970s and 1980s, much work in anisotropic diffusion was geared toward evaluating coolant voiding effects in sodium fast reactors (SFRs). As fast reactors have long neutron mean free paths, coarse mesh diffusion is much more suitable for fast reactors than for thermal reactors. However, mean free paths tend to be significantly longer in sodium coolant than in fuel and structure. Thus, streaming occurs parallel to pins in the axial direction. It follows that this streaming contributes to the sodium void reactivity coefficient, which is crucial to SFR safety. As sodium density decreases, streaming effects become more important. Consequently, incorporation of anisotropic diffusion coefficients can lead to more accurate prediction of sodium void worth.

Nakagawa applied Benoist's anisotropic diffusion model to predict sodium void reactivity in a single homogenized assembly model of the Jōyō reactor [147]. Nakagawa shows that isotropic diffusion calculations underestimate the sodium void coefficient by about 30-40% in 2-D and by about 10% in 3-D [72]. Wade and Gelbard demonstrated that Benoist's model works well for sodium voiding in SFRs and for streaming in gas-cooled fast reactors (GCFRs) [73]. Yoshida also applied Benoist's work to sodium voiding [74], and he also applied it to streaming through withdrawn control assemblies [75]. Rowlands summarizes past work on anisotropic diffusion, proposes his own model, and compares many models in the case of streaming through axial cylindrical channels [76]. Finally, Lee compares isotropic and anisotropic diffusion models of 3-D sodium voiding to transport theory solutions, and he concludes that anisotropic diffusion agrees well with transport while isotropic diffusion does not [79].

Benoist revisited anisotropic diffusion in the 1980s with a detailed numeric study of sodium void reactivity in hexagonal pin lattices [77,78]. He develops a modified anisotropic diffusion formalism tailored to the sodium voiding scenario, which had become the most prominent application of his original theory.

In 2002, Petrovic and Benoist summarized the current state of "B_N theory", which includes anisotropic (or buckling-dependent) leakage models [81]. This work has been widely cited in various homogenization and diffusion studies.

Beyond fast reactors, anisotropic diffusion has been applied more recently to various types of thermal reactors. Mathews studied anisotropic diffusion in deterministic pebble bed lattices in the 1990s [80]. These reactors are gas-cooled and graphite-moderated, so the diffusion coefficient varies dramatically between fuel and coolant. Thus, streaming effects are significant enough that isotropic diffusion models cannot accurately predict neutronic properties.

Poveschenko and Laletin studied anisotropic diffusion applied to the Voda Voda Energo Reactor (VVER) [82]. The VVER is a variation of a PWR with a triangular pin lattice (within hexagonal assemblies) and other modifications. They computed separate diffusion coefficients in the axial direction to capture neutron streaming effects due to heterogeneity. They found that this process changes the axial diffusion coefficient by approximately 2% in the fast group and approximately 15% in the thermal group. This amounts to an eigenvalue difference in the range of 200-300 pcm.

More recently, Larsen and Trahan studied anisotropic diffusion in gas-cooled reactors with "optically thin" channels, such as the Very High Temperature Reactor (VHTR) [83,84]. Coolant channels in the VHTR are "optically thin", because one or two of their dimensions are thin *relative* to the neutron mean free path in helium. This fact introduces strong anisotropic effects within these channels. Thus, just as in the case of gas-cooled pebble bed configuration, modeling VHTR neutronics requires either transport theory or anisotropic diffusion.

Van Rooijen and Chiba employed Monte Carlo and method of characteristics (MOC) codes to pre-calculate homogenized diffusion coefficients for SFRs [85]. They point out the well-known drawback of Benoist's anisotropic diffusion model - that the diffusion coefficients diverge in planar voids - as well as the simple fact that Benoist's early models are derived from approximate analytic solutions.

In summary, past research in anisotropic diffusion theory has been applied almost exclusively to capturing neutron streaming effects due to heterogeneity or coolant voiding. As far as we can see, anisotropic diffusion formalism has never been applied to distortions or geometry changes.

4.3.2 Formalism

Now let us define and lay out the anisotropic diffusion formalism that we will employ throughout the remainder of this thesis. The mathematics here was independently developed but is similar to that laid out by Trahan and Larsen [84]. We use the same notation and variable names as in Chapter 3.

First let us briefly review traditional isotropic diffusion, which is based on Fick's Law:

$$\vec{J} = -D\nabla\phi \tag{4.1}$$

Here the isotropic diffusion coefficient D is a scalar quantity equal to $1/3\Sigma_{\text{tr}}$. We multiply the vector $\nabla\phi$ by the scalar $-D$ to obtain the vector $\vec{J}$. The full one-group diffusion equation is

$$-\nabla\cdot D\nabla\phi + \Sigma_a = \frac{1}{k}\nu\Sigma_f\phi \tag{4.2}$$

We can express the leakage term explicitly in 3-D Cartesian geometry:

$$-\nabla\cdot D\nabla\phi = -\left[\frac{d}{dx}D\frac{d\phi}{dx} + \frac{d}{dy}D\frac{d\phi}{dy} + \frac{d}{dz}D\frac{d\phi}{dz}\right] \tag{4.3}$$

In anisotropic diffusion, the diffusion coefficient is no longer a scalar function of position and energy. Instead, it is a diagonal tensor with three components corresponding to the three spatial directions. We can represent this tensor as a diagonal 3x3 matrix:

$$\hat{D} = \begin{bmatrix} D_x & 0 & 0 \\ 0 & D_y & 0 \\ 0 & 0 & D_z \end{bmatrix} = \begin{bmatrix} \frac{1}{3\Sigma_{tr,x}} & 0 & 0 \\ 0 & \frac{1}{3\Sigma_{tr,y}} & 0 \\ 0 & 0 & \frac{1}{3\Sigma_{tr,z}} \end{bmatrix} \tag{4.4}$$

Now Fick's Law is a bit different. The matrix operator $\hat{D}$ operates on the vector $\nabla\phi$ to obtain the vector $\vec{J}$.

$$\vec{J} = -\hat{D}\nabla\phi \tag{4.5}$$

The full one-group diffusion equation is

$$-\nabla\cdot\left(\hat{D}\nabla\phi\right) + \Sigma_a = \frac{1}{k}\nu\Sigma_f\phi \tag{4.6}$$

Evaluating the leakage term yields

$$-\nabla\cdot\left(\hat{D}\nabla\phi\right) = -\left[\frac{d}{dx}D_x\frac{d\phi}{dx} + \frac{d}{dy}D_y\frac{d\phi}{dy} + \frac{d}{dz}D_z\frac{d\phi}{dz}\right] \tag{4.7}$$

This result is crucial. In diffusion, the leakage term determines the transfer of neutrons through space. In the above equation, we can see that the contribution to this distribution from each spatial direction i is multiplied by a factor D_i. Thus, by manipulating the values of D_i, we can alter the neutron diffusion rate by different factors in different directions. This is anisotropic diffusion!

4.4 The Uniform Anisotropic "Virtual Density" Theory

So far, we have developed and validated the "virtual density" theory only for expansions and swellings that occur uniformly in all directions. These are "isotropic" applications of the "virtual density" theory. However, few practical expansions or swellings occur uniformly in all three dimensions. Most fast reactors have different reactivity effects for axial and radial swellings, because cores swell by different factors in the axial and radial directions. In order to apply the "virtual density" to more practical geometric perturbations, we must somehow disentangle the three spatial dimensions from one another. We must change the neutron mean free path in one direction, while leaving it unchanged in the other two directions. We must devise a way to introduce a kind of "directional density" so that the material density a neutron "sees" depends upon which direction that neutron is traveling.

We could explore this concept in a number of ways, but for now let us formulate it in terms of anisotropic diffusion. This concept does *not* introduce a directional dependence into material densities, but it *does* introduce directional dependence into the diffusion coefficient. Neutrons diffuse more or less readily in different directions, but they still "see" the same absorption, scattering, and fission cross-sections in each direction.

4.4.1 Anisotropic Perturbation Theory

Now we can define perturbation theory for anisotropic diffusion. We use the same notation defined in Section 3.4.1, which defines the isotropic perturbation theory formalism. We encourage the reader to look over that section once again before proceeding.

The perturbed diffusion operator is proportional to the perturbed transport cross-section operator.

$$d\hat{D} = \begin{bmatrix} dD_x & 0 & 0 \\ 0 & dD_y & 0 \\ 0 & 0 & dD_z \end{bmatrix} = -\frac{1}{3\Sigma_{tr}^2} \begin{bmatrix} d\Sigma_{tr,x} & 0 & 0 \\ 0 & d\Sigma_{tr,y} & 0 \\ 0 & 0 & d\Sigma_{tr,z} \end{bmatrix} = -\frac{d\hat{\Sigma}_{tr}}{3\Sigma_{tr}^2} \tag{4.8}$$

Now the anisotropic leakage contribution to reactivity via perturbation theory is

$$\left\langle \phi^\dagger \middle| d\hat{L}\phi \right\rangle = -\int dV \left[\sum_{g=1}^{G} \left(\nabla\phi_g^\dagger \cdot \left(d\hat{\Sigma}_{tr,g} \nabla\phi_g \right) \right) \frac{1}{3\Sigma_{tr,g}^2} \right] \tag{4.9}$$

Compare this with the isotropic case in Equation 3.18. The perturbed transport cross-section is an operator acting on the real flux gradient but *not* on the adjoint flux gradient. The *un*perturbed transport cross-section is a scalar. We can evaluate the quantity within this integral for the Cartesian case as

$$\nabla\phi_g^\dagger \cdot \left(d\hat{\Sigma}_{tr,g}\nabla\phi_g\right) = \frac{d\phi_g^\dagger}{dx}\frac{d\phi_g}{dx}d\Sigma_{tr,g,x} + \frac{d\phi_g^\dagger}{dy}\frac{d\phi_g}{dy}d\Sigma_{tr,g,y} + \frac{d\phi_g^\dagger}{dz}\frac{d\phi_g}{dz}d\Sigma_{tr,g,z} \tag{4.10}$$

Note that each of these three terms corresponds to one spatial dimension. Thus, it is natural to decompose the leakage operator $\hat{L}$ into spatial components $\hat{L}_i$, where i denotes a particular spatial direction (x, y, or z). The reactivity contribution of $\hat{L}_i$ is simply

$$\left\langle\phi^\dagger\middle|\,\hat{L}_i\phi\right\rangle = \int dV\left[\sum_{g=1}^{G}\left(\nabla\phi_g^\dagger\right)_i\left(\nabla\phi_g\right)_i D_g\right] \tag{4.11}$$

The notation $\left(\nabla\phi_g\right)_i$ represents the real flux gradient in direction i. This could alternatively (and equivalently) be written as $\hat{\imath}\cdot\nabla\phi_g$, where $\hat{\imath}$ is the unit vector in direction i. Note that Equation 4.11 lacks the minus sign that appears in Equation 4.9; this is consistent with Equations 3.18 and 3.19 in the isotropic formalism.

The total leakage reactivity contribution is always the sum of all three directional leakage reactivity contributions. This is *always* true for *any* perturbation, because it is a simple mathematical definition from Equations 4.9 and 4.10.

$$\left\langle\phi^\dagger\middle|\,\hat{L}\phi\right\rangle = \sum_i\left\langle\phi^\dagger\middle|\,\hat{L}_i\phi\right\rangle \tag{4.12}$$

Furthermore, *if and only if* our (isotropic or anisotropic) perturbation is whole-core and uniform, then the surface integral in the divergence theorem (Equation 3.16) vanishes. Thus, we can write the equality

$$\frac{\left\langle\phi^\dagger\middle|\,\left(\frac{1}{k}\hat{F}-\hat{A}-\hat{T}\right)\phi\right\rangle}{\left\langle\phi^\dagger\middle|\,\hat{F}\phi\right\rangle} = \sum_i\frac{\left\langle\phi^\dagger\middle|\,\hat{L}_i\phi\right\rangle}{\left\langle\phi^\dagger\middle|\,\hat{F}\phi\right\rangle} \tag{4.13}$$

This equality is essentially the same as its isotropic counterpart in Equation 3.25, except that it separates the leakage term into its three anisotropic components. The justification for this equality is the same as that for the isotropic case derived in Equations 3.26 - 3.30. We can also re-write Equation 4.13 to be in the same form as Equation 3.25, and now the factor of 2 is apparent.

$$\frac{\left\langle\phi^\dagger\middle|\,\left(\frac{1}{k}\hat{F}-\hat{A}-\hat{T}+\hat{L}\right)\phi\right\rangle}{\left\langle\phi^\dagger\middle|\,\hat{F}\phi\right\rangle} = 2\sum_i\frac{\left\langle\phi^\dagger\middle|\,\hat{L}_i\phi\right\rangle}{\left\langle\phi^\dagger\middle|\,\hat{F}\phi\right\rangle} = 2\frac{\left\langle\phi^\dagger\middle|\,\left(\frac{1}{k}\hat{F}-\hat{A}-\hat{T}\right)\phi\right\rangle}{\left\langle\phi^\dagger\middle|\,\hat{F}\phi\right\rangle} \tag{4.14}$$

Essentially, this equality states that each whole-core *spatial* perturbation must be balanced by a whole-core *spectral* perturbation. When we considered only isotropic "virtual density" perturbations, we saw that a uniform whole-core density perturbation includes equal spatial and spectral contributions. When we swell or expand a reactor, we alter the leakage rate. If neutrons are more or less likely to leak, they will be less or more likely to be absorbed or scatter. Thus, any spatial perturbation induces a corresponding spectral perturbation. Since those two parts are precisely equal, we only needed to compute *one* and then multiply by 2.

4.4.2 Uniform Anisotropic Shorthand Notation

Before we proceed, let us revise the shorthand notation we defined in Section 3.4.3 to describe anisotropic cases. Again, let the scalar value S represent the *spectral* reactivity contribution.

$$S = \frac{\left\langle \phi^{\dagger} \middle| \left(\frac{1}{k}\hat{F} - \hat{A} - \hat{T} \right) \phi \right\rangle}{\left\langle \phi^{\dagger} \middle| \hat{F}\phi \right\rangle} \tag{4.15}$$

Now let the scalar value L_i represent the *leakage* reactivity contribution in direction i.

$$L_i = \frac{\left\langle \phi^{\dagger} \middle| \hat{L}_i\phi \right\rangle}{\left\langle \phi^{\dagger} \middle| \hat{F}\phi \right\rangle} \tag{4.16}$$

Furthermore, the scalar value L represents the total leakage reactivity contribution.

$$L = \frac{\left\langle \phi^{\dagger} \middle| \hat{L}\phi \right\rangle}{\left\langle \phi^{\dagger} \middle| \hat{F}\phi \right\rangle} \tag{4.17}$$

Note that the lack of a "hat" on S, L, and L_i distinguishes them as scalars rather than operators.

Now we can express the spectral-leakage equality in Equation 4.13 simply and elegantly:

$$S = L = \sum_i L_i \tag{4.18}$$

The shorthand variables S, L, and L_i allow us to conveniently avoid writing the full adjoint-weighted expressions (and fission denominator) a multitude of times. However, whenever these variables are used, one must assume that the spatial integral in the fission denominators covers the entire reactor, while the spatial integral in the numerators covers only the extent of the swelling or expansion. We will use this shorthand notation for the remainder of this thesis.

4.4.3 "Virtual Density" Formalism for Uniform Whole-Core Swellings and Expansions

Now we consider the anisotropic "virtual density" expressions for reactivity coefficients due to uniform swelling (in one or more directions) of whole cores. Refer back to Section 3.3.2 for precise definitions of "swelling" and "expansion". We can simply guess the result by intuition via adjusting the neutron mean free path in different directions. Suppose a core swells in the direction i by the small fraction ϵ_i. In Chapter 3, we discussed the obvious fact that swelling an infinite 1-D slab produces zero reactivity - the effects of reduced density and increased slab width precisely counteract each other. Similarly, we can see that any core swelling in the direction i will produce zero change to leakage in the i direction. However, leakage will always increase in all *other* directions due to decreased density. So we can express the reactivity by lengthening the diffusion coefficient (or decreasing the transport cross-section) in all those *other* directions. Thus, the first order reactivity is

$$\Delta\rho_{\text{swell}} = \sum_i \left(-2\epsilon_i \sum_{j \neq i} L_j \right) \tag{4.19}$$

This is the most general expression for a different uniform swelling magnitude ϵ_i in each direction i. The leakage reactivity contribution is non-zero in all directions *except* i. Throughout the remainder of this thesis, we will always define $\epsilon_i = f_i - 1$, where f_i is the fractional dimension change in direction i. So a 1% swelling in the x direction corresponds to $f_x = 1.01$ and $\epsilon_x = 0.01$. This is a simplification of the definition of ϵ_i given in Equation 3.31 and used throughout Chapter 3. We find this simplification to be much more convenient in the generalized anisotropic formalism.

Note the factor of 2 in Equation 4.19. Just as in the isotropic swelling case (see Equation 3.25), this factor accounts for the fact that any leakage reactivity induces an *equal* spectral reactivity. In an isotropic case, we know that S represents the spectral reactivity, so we can easily express the total reactivity using only S, only L, or both S and L. Thus, the reactivity due to any uniform whole-core isotropic perturbation can be expressed using only S, a truly remarkable fact. However, in an anisotropic case, we cannot break up S into directional components. Thus, we cannot express the reactivity using only S. Fortunately, we can still place a factor of 2 in front of the anisotropic leakage reactivity to account for the spectral reactivity, even though we cannot compute it directly.

Now let us turn our attention to uniform whole-core *expansions*. Converting a swelling into an expansion is intuitively easy - we can simply add a density perturbation to the swelling. Equation 4.19 expresses a swelling of magnitude ϵ_i in each direction i. In order to convert this swelling into an expansion, we can express the reactivity due to a uniform whole-core density increase. This means that the densities of all nuclides increase by the same factor $\sum_i \epsilon_i$.

$$\Delta\rho_{\text{den}} = \sum_i \epsilon_i \left(S + L_x + L_y + L_z\right) \tag{4.20}$$

$$= \sum_i \epsilon_i \left(S + \sum_j L_j\right) \tag{4.21}$$

Now we can easily write down the expression for a uniform whole-core expansion by adding Equation 4.21 to Equation 4.19.

$$\Delta\rho_{\text{expan}} = \sum_i \epsilon_i \left(S + L_i - \sum_{j \neq i} L_j\right) \tag{4.22}$$

This result is intuitive. An expansion does not conserve mass as a swelling does, so the positive S term represents added material. The positive L_i term represents decreased leakage in the same direction as the expansion, because the core size in that direction increases without a density change.

Note that for the special case of isotropic swelling, Equation 4.19 reduces to Equation 3.25. Let $(\epsilon_x,\epsilon_y,\epsilon_z) = (\epsilon,\epsilon,\epsilon)$ so that the swelling is isotropic. Let $\epsilon = f - 1$ for a reactor swelling uniformly by a factor f in all directions. If we write this as a reactivity coefficient $d\rho/df$, then

$$\left[\frac{d\rho}{df}\right]_{\text{swell,iso}} = -4L = -4S \tag{4.23}$$

Note that this is equivalent to Equation 3.25 if one differentiates ϵ with respect to f (using Equation 3.31) and evaluates the result at $f = 1$. We have also applied the equality $S = L$ in Equation 4.18 to the expression above to show that the isotropic swelling reactivity can be expressed via either only spectral or only leakage quantities.

We can do the same thing for isotropic expansion. Applying $(\epsilon_x,\epsilon_y,\epsilon_z) = (\epsilon,\epsilon,\epsilon)$ to Equation 4.22 yields $3S - L$, but we can again apply the equality $L = S$ to express the isotropic expansion reactivity as either only spectral or only leakage quantities.

$$\left[\frac{d\rho}{df}\right]_{\text{expan,iso}} = 3S - L = 3L - S = 2L = 2S \tag{4.24}$$

These equivalent expressions lead us to wonder how many ways one can express the anisotropic swelling and expansion reactivities.

4.4.4 An Infinitude of Equalities

The most intriguing, advantageous, and confusing aspect of “virtual density” theory is that every reactivity can always be expressed in an *infinite* number of ways. Showing this is easy.

First, we can take Equation 4.18 and rearrange terms to obtain a quantity that is equal to zero:

$$S - L_x - L_y - L_z = 0 \tag{4.25}$$

$$S - \sum_i L_i = 0 \tag{4.26}$$

What happens when we add zero to something? That "something" does not change. So we can add Equation 4.26 (multiplied by any factor) to Equation 4.19 to obtain an infinitude of equivalent expressions for the swelling reactivity. Suppose that we restrict the multiplication factor in each direction i to be $M_i \epsilon_i$, where M_i is any integer. This allows various S and L_i terms to potentially cancel one another out. Now the swelling reactivity is

$$\Delta\rho_{\text{swell}} = \sum_i \epsilon_i \left[-2\sum_{j\neq i} L_j + M_i \left(S - \sum_j L_j \right) \right] \tag{4.27}$$

This is the fully general expression for a reactivity due to anisotropic swelling. The M_i values can be any integer, and so the expression can take on an infinite number of forms. Only a few forms may be useful, however.

We can do precisely the same thing for anisotropic expansion:

$$\Delta\rho_{\text{expan}} = \sum_i \epsilon_i \left[\left(S + L_i - \sum_{j\neq i} L_j \right) + M_i \left(S - \sum_j L_j \right) \right] \tag{4.28}$$

In our Mathematics and Computation (M&C) 2013 conference paper, we gave three expressions for anisotropic swelling and three expressions for anisotropic expansion [99]. The three swelling expressions can be obtained from Equation 4.27 above. They correspond to (M_x,M_y,M_z) = (0,0,0), (-2,-2,-2), and (-1,-1,-1), respectively. Here they are in our shorthand notation:

$$\begin{aligned} \Delta\rho_{\text{swell}} &= \sum_i \left(-2\epsilon_i \sum_{j\neq i} L_j \right) && (4.29) \\ &= \sum_i 2\epsilon_i \left(L_i - S \right) && (4.30) \\ &= \sum_i \epsilon_i \left(-S + L_i - \sum_{j\neq i} L_j \right) && (4.31) \end{aligned}$$

Note that we used a different sign convention for the L quantities in the M&C paper, but the expressions are equivalent.

The three expansion expressions given in the M&C paper can be obtained from Equation 4.28 above. They correspond to (M_x,M_y,M_z) = (-1,-1,-1), (1,1,1), and (0,0,0), respectively. In our shorthand notation, they are

$$\Delta\rho_{\text{expan}} = \sum_i 2\epsilon_i L_i \tag{4.32}$$

$$= \sum_i 2\epsilon_i \left(S - \sum_{j\neq i} L_j \right) \tag{4.33}$$

$$= \sum_i \epsilon_i \left(S + L_i - \sum_{j\neq i} L_j \right) \tag{4.34}$$

Equation 4.32 is insightful, especially when compared with Equation 4.29. As we've discussed in depth, Equation 4.29 shows that reactivity due to swelling in direction i consists of *increasing* leakage in all directions *except* i. In tandem, Equation 4.32 shows that reactivity due to expansion in direction i consists of *decreasing* leakage in *only* direction i.

We can simplify Equations 4.27 and 4.28 for the isotropic swelling and expansion cases in which (ϵ_x,ϵ_y,ϵ_z) = (ϵ,ϵ,ϵ).

$$\left[\frac{d\rho}{df}\right]_{\text{swell,iso}} = -4L + M\,(S - L) \tag{4.35}$$

$$\left[\frac{d\rho}{df}\right]_{\text{expan,iso}} = (3S - L) + M\,(S - L) \tag{4.36}$$

Here M can be any integer, and it has no directional components in the isotropic case.

So now we have laid out the general anisotropic equations that define arbitrary uniform whole-core swellings and expansions.

4.4.5 Cartesian Application

Now let us show a few examples for common scenarios. The expressions in previous sections are fully generic and allow different swelling or expansion fractions in each dimension. Now we consider a swelling fraction $f_x = f$ in only the x direction. Applying Equation 4.19 for this case yields

$$\left[\frac{d\rho}{df}\right]_{\text{swell,x}} = -2\,(L_y + L_z) \tag{4.37}$$

We can use Equation 4.27 to obtain an infinite list of equivalent expressions. Here are several:

$$\begin{aligned}
\left[\frac{d\rho}{df}\right]_{\text{swell,x}} &= 4\left(-S+L_x\right)+2\left(L_y+L_z\right) && (4.38)\\
&= 3\left(-S+L_x\right)+L_y+L_z && (4.39)\\
&= 2\left(-S+L_x\right) && (4.40)\\
&= -S+L_x-L_y-L_z && (4.41)\\
&= -2\left(L_y+L_z\right) && (4.42)\\
&= S-L_x-3\left(L_y+L_z\right) && (4.43)\\
&= 2\left(S-L_x\right)-4\left(L_y+L_z\right) && (4.44)
\end{aligned}$$

Now let us consider an *expansion* fraction $f_x = f$ in only the x direction. Applying Equation 4.28 for this case yields another infinite series of expressions, several of which are:

$$\begin{aligned}
\left[\frac{d\rho}{df}\right]_{\text{expan,x}} &= -2S+4L_x+2\left(L_y+L_z\right) && (4.45)\\
&= -S+3L_x+L_y+L_z && (4.46)\\
&= 2L_x && (4.47)\\
&= S+L_x-L_y-L_z && (4.48)\\
&= 2S-2\left(L_y+L_z\right) && (4.49)\\
&= 3S-L_x-3\left(L_y+L_z\right) && (4.50)\\
&= 4S-2L_x-4\left(L_y+L_z\right) && (4.51)
\end{aligned}$$

These lists of equivalent expressions may seem superfluous, but they can be quite useful when one does not know L in every direction. Perhaps one can only easily compute L in one or two directions. Then one can choose an equivalent expression that only requires knowledge of L in those one or two directions. In Equations 4.38 - 4.51, we have chosen to list enough expressions to cover all the unique combinations of S and L_i, but one can easily see the pattern and write down the rest of the infinite sequences.

4.4.6 Hexagonal-Z or R-Z Application

Most fast reactors consist of triangular fuel pin arrays within hexagonal assemblies. Thus, a hexagonal-z coordinate system is most appropriate, and we can condense L_x and L_y into a single quantity L_r:

$$L_r = L_x + L_y \tag{4.52}$$

Fast reactors typically swell axially and radially by different factors. Analysts compute the "axial swelling reactivity coefficient" and the "radial swelling reactivity coefficient". We can apply Equation 4.27 to obtain numerous expressions for each of these coefficients:

$$\begin{aligned}
\left[\frac{d\rho}{df}\right]_{\text{swell,ax}} &= 4\left(-S + L_z\right) + 2L_r && (4.53)\\
&= 3\left(-S + L_z\right) + L_r && (4.54)\\
&= 2\left(-S + L_z\right) && (4.55)\\
&= -S + L_z + -L_r && (4.56)\\
&= -2L_r && (4.57)\\
&= S - L_z - 3L_r && (4.58)\\
&= 2\left(S - L_z\right) - 4L_r && (4.59)
\end{aligned}$$

Note that the list of equivalent expressions in Equations 4.53 - 4.59 can be easily obtained by modifying Equations 4.38 - 4.44. One must only replace L_x with L_z and also replace $L_y + L_z$ with L_r. When a fast reactor core swells axially, leakage remains constant in the axial direction but *increases* in the radial direction. This is apparent in Equation 4.57 above. Note that all these expressions are equally valid for an R-Z coordinate system in cylindrical geometry.

Now let us consider radial swelling. We can obtain these expressions via Equation 4.27 with $(\epsilon_x,\epsilon_y,\epsilon_z) = (\epsilon,\epsilon,0)$. The result is

$$\begin{aligned}
\left[\frac{d\rho}{df}\right]_{\text{swell,rad}} &= -6S + 4L_r + 2L_z && (4.60)\\
&= -5S + 3L_r + L_z && (4.61)\\
&= -4S + 2L_r && (4.62)\\
&= -3S + L_r - L_z && (4.63)\\
&= -2S - 2L_z && (4.64)\\
&= -S - L_r - 3L_z && (4.65)\\
&= -2L_r - 4L_z && (4.66)\\
&= S - 3L_r - 5L_z && (4.67)\\
&= 2S - 4L_r - 6L_z && (4.68)
\end{aligned}$$

In contrast to axial swelling, radial swelling increases leakage in *both* the axial and radial directions. This is most evident in Equation 4.66, which shows that the axial leakage increase is weighted by twice as much as the radial leakage increase.

Note that Equations 4.53 - 4.68 assume that all nuclides in the core swell uniformly. Although this is certainly true for fuel and structure nuclides, it is *not* true for coolant nuclides. In a practical fast reactor scenario, one would apply Equations 4.53 - 4.68 and then subsequently apply an appropriate coolant density perturbation (using traditional perturbation theory methods). The amount by which the coolant density must be adjusted depends upon the detailed pin and duct geometry in a specific design, so we will not delve into that here.

Since axial and radial *expansions* do not practically occur in fast reactors, we will skip that analysis.

Conceptually, it should be obvious that simultaneous radial and axial swellings (by the same factor f) are equivalent to an isotropic swelling. We can demonstrate this mathematically. Adding any expression in Equations 4.53 - 4.59 to any expression in Equations 4.60 - 4.68 and then subsequently applying the equality $S = L_r + L_z$ will yield the result $-4S$, which we already derived in Equation 4.23.

Furthermore, uniformly reducing all densities in a reactor by a fraction f (without changing core dimensions) will result in *precisely half* the reactivity decrease that an isotropic swelling causes. This should be evident from

$$\left[\frac{d\rho}{df}\right]_{\text{swell,rad}} + \left[\frac{d\rho}{df}\right]_{\text{swell,ax}} = \left[\frac{d\rho}{df}\right]_{\text{swell,iso}} = -2\left[\frac{d\rho}{df}\right]_{\text{den}} = -2\left[\frac{d\rho}{df}\right]_{\text{expan,iso}} = -4S \quad (4.69)$$

Here we see that uniformly increasing all nuclide densities throughout a reactor is worth $2S$, precisely equal to the worth of a uniform isotropic core expansion. Also note that adding three density perturbations to an isotropic swelling is equivalent to an isotropic expansion so that $-4S + 3(2S) = 2S$.

4.5 Comparison to the Russian Literature

Now that we have independently derived the uniform anisotropic "virtual density" formalism via anisotropic diffusion, we compare it to what already exists in the old Soviet literature. While the western literature contains only brief, scattered mentions of the uniform isotropic "virtual density" theory, Soviet literature on the topic is much more extensive and extends the theory to uniform anisotropic cases.

As we stated in Section 3.3.3, Shikhov is credited with first describing the isotropic "virtual density" theory for uniform whole-core isotropic cases in the 1950s and 1960s [87,90], and Cullen independently wrote two relatively recent reports on the subject [96,97].

The Soviet neutronics community also applied the anisotropic "virtual density" theory to uniform whole-core cases, although it is not clear how they derived it or what formalism they used. Design documentation for the BN-800, a Russian sodium fast reactor, contains Equation 4.57 for axial swelling and Equation 4.65 for radial swelling [95]. A description of

Table 4.1: A 1-D expansion of a 2-D reactor.

actual expansion	virtual expansion
$b' = bf$	$\Sigma'_{tr,y} = \Sigma_{tr,y} f$

Soviet regulatory practices (published in an East German (DDR) journal) from the 1960s also contains the same two equations for fast reactor axial and radial swelling coefficients [91]. These reports and documents give no explanation or derivation for these equations; they only reference Shikhov's early work on isotropic cases. They also do not show any of the numerous equivalent swelling expressions shown in Equations 4.53 - 4.59. Their choice of Equation 4.57 for axial swelling is sensible (it is the most direct expression that follows most directly from Equation 4.19), but we do not understand why they chose Equation 4.65 (instead of the more direct Equation 4.66) for radial swelling. We have not found the original formalism or derivation for these equations in the Soviet literature, so we derived our formalism in Sections 4.3.2 - 4.4.6 independently. We do not know whether the Soviet nuclear community obtained these equations via anisotropic diffusion or via some other route. We also do not know whether the Soviet literature contains *all* the expressions and equalities we have shown here; it may only contain some of them. In any case, no formal description of the anisotropic theory exists in the readily available literature, so we took it upon ourselves to lay it out in the previous sections.

4.6 Analytic Examples for Uniform Anisotropic Expansions and Swellings

Simple analytic examples are often helpful to build intuition for how methods work. Here we will walk through several examples of the uniform anisotropic "virtual density" theory.

4.6.1 1-D Expansion of a 2-D Reactor

Let us consider a homogenous 2-D "box" reactor of finite dimensions a and b. What is the reactivity effect when this box is expanded in *only* the y direction? We cannot simply uniformly alter the material densities by one factor, because that would inevitably change leakage in the x direction. Instead, we can define a virtual expansion by manipulating the macroscopic transport cross section in *only* the y direction.

Let us consider a one-group model. The unperturbed flux is

$$\phi(x,y) = \cos\left(\frac{\pi x}{a}\right)\cos\left(\frac{\pi y}{b}\right) \tag{4.70}$$

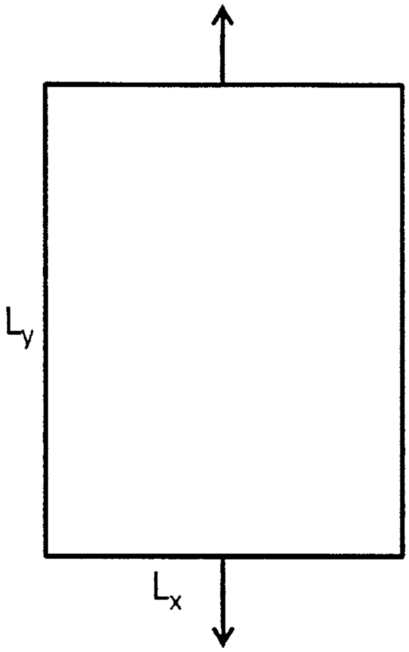

Figure 4.1: A 1-D expansion of a 2-D reactor.

The expansion fraction is

$$f = \frac{b'}{b} \tag{4.71}$$

Keep in mind that this is an expansion, not a swelling. Therefore, as the box reactor grows in the y direction, extra mass is added. The material density does not change. Thus, the exact reactivity as a function of f is

$$\begin{aligned}
\Delta\rho &= \tfrac{1}{k} - \tfrac{1}{k'} && (4.72)\\
&= \tfrac{DB^2+\Sigma_a}{\nu\Sigma_f} - \tfrac{DB'^2+\Sigma_a}{\nu\Sigma_f} && (4.73)\\
&= \tfrac{D}{\nu\Sigma_f}\left(B^2 - B'^2\right) && (4.74)\\
&= \tfrac{D}{\nu\Sigma_f}\left[\left(\tfrac{\pi}{b}\right)^2 - \left(\tfrac{\pi}{bf}\right)^2\right] && (4.75)
\end{aligned}$$

The slope of this shift for infinitesimally small perturbations, which first order perturbation theory should be able to predict, is

$$\left[\frac{d\rho}{df}\right]_{f=1} = \frac{2D}{\nu\Sigma_f}\left(\frac{\pi}{b}\right)^2 \tag{4.76}$$

Now let us test whether the anisotropic "virtual density" theory can match this result. Instead of expanding the reactor in the y direction, we shrink the neutron mean free path in the y direction. We can do this by increasing the macroscopic transport cross-section in the y direction.

$$d\Sigma_y = \Sigma_y' - \Sigma_y = \Sigma(f-1) \tag{4.77}$$

Now the unperturbed flux gradient in the y direction is

$$(\nabla\phi)_y = -\frac{\pi}{b}\cos\left(\frac{\pi x}{a}\right)\sin\left(\frac{\pi y}{b}\right) \tag{4.78}$$

Now we can evaluate the reactivity perturbation numerator with one double integral:

$$\epsilon_y\left\langle\phi^\dagger\middle|\,\hat{L}_y\phi\right\rangle = \textstyle\int_{-a/2}^{a/2}\int_{-b/2}^{b/2}dxdy\,(\nabla\phi)_y^2\,\frac{f-1}{3\Sigma_{tr}} \tag{4.79}$$

$$= (f-1)D\left(\tfrac{\pi}{b}\right)^2\tfrac{a}{2}\tfrac{b}{2} \tag{4.80}$$

The reactivity perturbation denominator is

$$\left\langle\phi^\dagger\middle|\,\hat{F}\phi\right\rangle = \int_{-a/2}^{a/2}\int_{-b/2}^{b/2}dxdy\,\nu\Sigma_f\phi^2 = \nu\Sigma_f\frac{a}{2}\frac{b}{2} \tag{4.81}$$

Now the whole reactivity is

$$\Delta\rho = 2\frac{(f-1)D\left(\frac{\pi}{b}\right)^2}{\nu\Sigma_f} \tag{4.82}$$

This is a linear function of f. Its derivative is

$$\left[\frac{d\rho}{df}\right]_{f=1} = \frac{2D}{\nu\Sigma_f}\left(\frac{\pi}{b}\right)^2 \tag{4.83}$$

This agrees precisely with the exact result we derived above. If we wish to use our shorthand notation (defined in Section 4.4.2), then we can rewrite this result as

$$\left[\frac{d\rho}{df}\right]_{f=1} = 2L_y \tag{4.84}$$

$$L_y = \frac{D}{\nu\Sigma_f}\left(\frac{\pi}{b}\right)^2 \tag{4.85}$$

Note that this is Equation 4.47 for a uniform expansion in one direction.

Table 4.2: A 1-D expansion of a 3-D reactor.

actual expansion	virtual expansion
$c' = cf$	$\Sigma'_{tr,z} = \Sigma_{tr,z} f$

4.6.2 1-D Expansion of a 3-D Reactor

Now let us consider a 1-D expansion of a 3-D "box" reactor. This "box" expands in only the z direction.

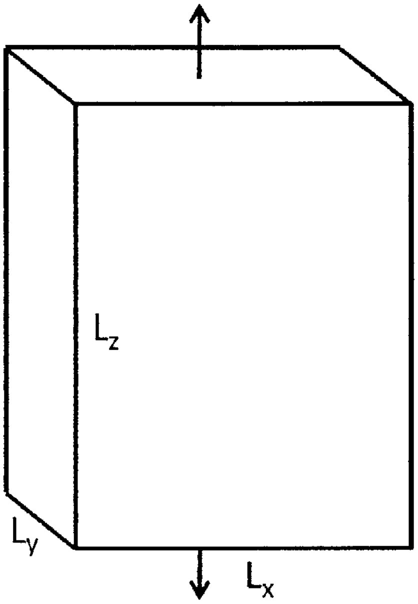

Figure 4.2: A 1-D expansion of a 3-D reactor.

The unperturbed one-group flux is

$$\phi(x, y, z) = \cos\left(\frac{\pi x}{a}\right) \cos\left(\frac{\pi y}{b}\right) \cos\left(\frac{\pi z}{c}\right) \tag{4.86}$$

The expansion factor in the z direction is

$$f = \frac{c'}{c} \tag{4.87}$$

The exact reactivity as a function of f is

$$\begin{aligned}\Delta\rho \quad & = \tfrac{1}{k} - \tfrac{1}{k'} && (4.88)\\ & = \tfrac{DB^2+\Sigma_a}{\nu\Sigma_f} - \tfrac{DB'^2+\Sigma_a}{\nu\Sigma_f} && (4.89)\\ & = \tfrac{D}{\nu\Sigma_f}\left(B^2 - B'^2\right) && (4.90)\\ & = \tfrac{D}{\nu\Sigma_f}\left[\left(\tfrac{\pi}{c}\right)^2 - \left(\tfrac{\pi}{cf}\right)^2\right] && (4.91)\end{aligned}$$

The slope of this curve for the smallest perturbation magnitudes is

$$\left[\frac{d\rho}{df}\right]_{f=1} = \frac{2D}{\nu\Sigma_f}\left(\frac{\pi}{c}\right)^2 \tag{4.92}$$

Now we can see how the "virtual density" theory yields the same result. Instead of expanding this box reactor in the z direction, we can simply decrease the neutron mean free path in the z direction. So let us increase the z component of the macroscopic transport cross section in the z direction:

$$d\Sigma_z = \Sigma_z' - \Sigma_z = \Sigma(f-1) \tag{4.93}$$

The unperturbed flux gradient in the z direction is

$$(\nabla\phi)_z = -\frac{\pi}{c}\cos\left(\frac{\pi x}{a}\right)\cos\left(\frac{\pi y}{b}\right)\sin\left(\frac{\pi z}{c}\right) \tag{4.94}$$

The first order reactivity numerator is

$$\begin{aligned}\epsilon_z\left\langle\phi^\dagger\middle|\,\hat{L}_z\phi\right\rangle & = \textstyle\int_{-a/2}^{a/2}\int_{-b/2}^{b/2}\int_{-c/2}^{c/2} dxdydz\ (\nabla\phi)_z^2\,\frac{f-1}{3\Sigma_{tr}} && (4.95)\\ & = (f-1)D\left(\tfrac{\pi}{c}\right)^2\tfrac{a}{2}\tfrac{b}{2}\tfrac{c}{2} && (4.96)\end{aligned}$$

The denominator is

$$\left\langle\phi^\dagger\middle|\,\hat{F}\phi\right\rangle = \int_{-a/2}^{a/2}\int_{-b/2}^{b/2}\int_{-c/2}^{c/2} dxdydz\,\nu\Sigma_f\phi^2 = \nu\Sigma_f\frac{a}{2}\frac{b}{2}\frac{c}{2} \tag{4.97}$$

So the whole first order reactivity is

$$\Delta\rho = 2\frac{(f-1)D\left(\frac{\pi}{c}\right)^2}{\nu\Sigma_f} \tag{4.98}$$

Note that we have again included an additional factor of 2. Now the slope of this linear function of f is

$$\left[\frac{d\rho}{df}\right]_{f=1} = \frac{2D}{\nu\Sigma_f}\left(\frac{\pi}{c}\right)^2 \tag{4.99}$$

This agrees precisely with the exact result we derived above. If we wish to use our shorthand notation (defined in Section 4.4.2), then we can rewrite this result as

$$\left[\frac{d\rho}{df}\right]_{f=1} = 2L_z \tag{4.100}$$

$$L_z = \frac{D}{\nu\Sigma_f}\left(\frac{\pi}{c}\right)^2 \tag{4.101}$$

Note that this is Equation 4.47 for a uniform expansion in one direction.

Table 4.3: A 2-D expansion of a 3-D reactor.

actual expansion	virtual expansion
$a' = af$	$\Sigma'_{tr,x} = \Sigma_{tr,x} f$
$b' = bf$	$\Sigma'_{tr,y} = \Sigma_{tr,y} f$

4.6.3 2-D Expansion of a 3-D Reactor

Now let us consider a uniform 2-D expansion of a homogenous 3-D box reactor.

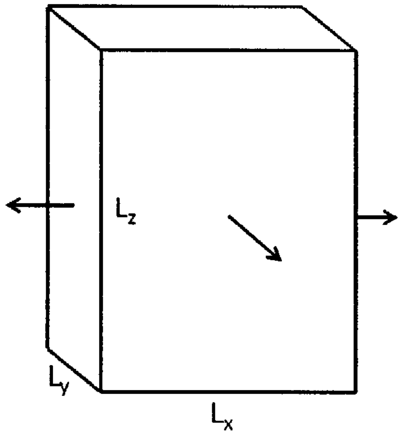

Figure 4.3: A 2-D expansion of a 3-D reactor.

The unperturbed flux is

$$\phi(x, y, z) = \cos\left(\frac{\pi x}{a}\right) \cos\left(\frac{\pi y}{b}\right) \cos\left(\frac{\pi z}{c}\right) \tag{4.102}$$

The expansion fraction f is

$$f = \frac{a'}{a} = \frac{b'}{b} \tag{4.103}$$

The exact reactivity is

$$\begin{aligned} \Delta\rho \quad &= \tfrac{1}{k} - \tfrac{1}{k'} && (4.104)\\ &= \tfrac{DB^2+\Sigma_a}{\nu\Sigma_f} - \tfrac{DB'^2+\Sigma_a}{\nu\Sigma_f} && (4.105)\\ &= \tfrac{D}{\nu\Sigma_f}\left(B^2 - B'^2\right) && (4.106)\\ &= \tfrac{D}{\nu\Sigma_f}\left[\left(\tfrac{\pi}{a}\right)^2 + \left(\tfrac{\pi}{b}\right)^2 - \left(\tfrac{\pi}{af}\right)^2 - \left(\tfrac{\pi}{bf}\right)^2\right] && (4.107) \end{aligned}$$

The slope of this reactivity with respect of f is

$$\left[\frac{d\rho}{df}\right]_{f=1} = \frac{2D}{\nu\Sigma_f}\left[\left(\frac{\pi}{a}\right)^2 + \left(\frac{\pi}{b}\right)^2\right] \tag{4.108}$$

Now let us consider an equivalent "virtual density" perturbation. Instead of expanding the core in the x and y directions, we can simply increase the macroscopic transport cross-sections in those directions.

$$d\Sigma_x = \Sigma'_x - \Sigma_x = \Sigma(f-1) \tag{4.109}$$

$$d\Sigma_y = \Sigma'_y - \Sigma_y = \Sigma(f-1) \tag{4.110}$$

The unperturbed flux gradients in the x and y directions are

$$(\nabla\phi)_x = -\frac{\pi}{a}\sin\left(\frac{\pi x}{a}\right)\cos\left(\frac{\pi y}{b}\right)\cos\left(\frac{\pi z}{c}\right) \tag{4.111}$$

$$(\nabla\phi)_y = -\frac{\pi}{b}\cos\left(\frac{\pi x}{a}\right)\sin\left(\frac{\pi y}{b}\right)\cos\left(\frac{\pi z}{c}\right) \tag{4.112}$$

The reactivity numerator due to this perturbation is

$$\begin{aligned} \epsilon_{x,y}\left\langle\phi^\dagger\right|\left(\hat{L}_x + \hat{L}_y\right)\phi\right\rangle &= \int_{-a/2}^{a/2}\int_{-b/2}^{b/2}\int_{-c/2}^{c/2} dxdydz\,\left[(\nabla\phi)_x^2 + (\nabla\phi)_y^2\right]\tfrac{f-1}{3\Sigma_{tr}} && (4.113)\\ &= (f-1)D\left[\left(\tfrac{\pi}{a}\right)^2 + \left(\tfrac{\pi}{b}\right)^2\right]\tfrac{a}{2}\tfrac{b}{2}\tfrac{c}{2} && (4.114) \end{aligned}$$

The denominator is

$$\left\langle\phi^\dagger\right|\hat{F}\phi\right\rangle = \int_{-a/2}^{a/2}\int_{-b/2}^{b/2}\int_{-c/2}^{c/2} dxdydz\,\nu\Sigma_f\phi^2 = \nu\Sigma_f\frac{a}{2}\frac{b}{2}\frac{c}{2} \tag{4.115}$$

The whole first order reactivity is

$$\Delta\rho = 2\frac{(f-1)D}{\nu\Sigma_f}\left[\left(\frac{\pi}{a}\right)^2 + \left(\frac{\pi}{b}\right)^2\right] \tag{4.116}$$

As usual, we have added the factor of 2. Now the slope of this function with respect to f is

$$\left[\frac{d\rho}{df}\right]_{f=1} = \frac{2D}{\nu\Sigma_f}\left[\left(\frac{\pi}{a}\right)^2 + \left(\frac{\pi}{b}\right)^2\right] \tag{4.117}$$

This agrees precisely with the exact result we derived above. If we wish to use our shorthand notation (defined in Section 4.4.2), then we can rewrite this result as

$$\left[\frac{d\rho}{df}\right]_{f=1} = 2\left(L_x + L_y\right) \tag{4.118}$$

$$L_x = \frac{D}{\nu\Sigma_f}\left(\frac{\pi}{a}\right)^2 \tag{4.119}$$

$$L_y = \frac{D}{\nu\Sigma_f}\left(\frac{\pi}{b}\right)^2 \tag{4.120}$$

Note that this is Equation 4.47 for a uniform expansion in one direction.

Table 4.4: A 1-D swelling of a 2-D reactor.

actual expansion	virtual expansion
$b' = bf$	$\Sigma'_{tr,x} = \Sigma_{tr,x}/f$
$\Sigma' = \Sigma/f$	

4.6.4 1-D Swelling of a 2-D Reactor

Now we consider swelling cases, in which mass is conserved. As the dimensions of a reactor expand, the material densities uniformly decrease by an appropriate factor.

First, let us consider the 1-D swelling of a homogenous 2-D reactor.

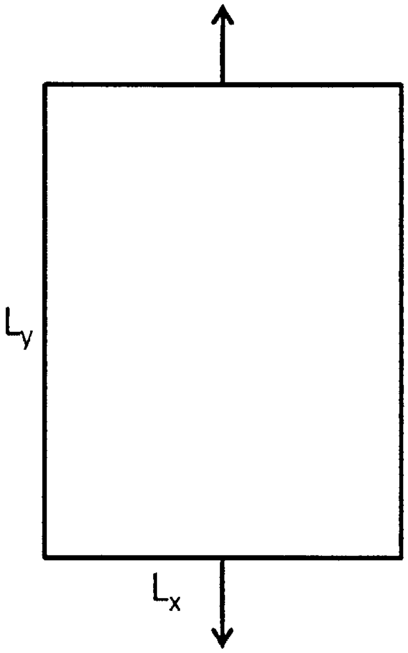

Figure 4.4: A 1-D swelling of a 2-D reactor.

The unperturbed one-group flux is

$$\phi(x, y) = \cos\left(\frac{\pi x}{a}\right)\cos\left(\frac{\pi y}{b}\right) \tag{4.121}$$

We define the swelling in terms of an increase in b and a corresponding decreases in Σ, which represents a generic macroscopic cross-section. We can easily see that mass is conserved.

$$f = \frac{b'}{b} \tag{4.122}$$

$$\Sigma' = \Sigma / f \tag{4.123}$$

We can compute the exact reactivity for this swelling, which only requires one or two more steps than computing the shift due to an expansion.

$$\Delta\rho = \tfrac{1}{k} - \tfrac{1}{k'} \tag{4.124}$$

$$= \tfrac{DB^2+\Sigma_a}{\nu\Sigma_f} - \tfrac{DB'^2+\Sigma_a}{\nu\Sigma_f} \tag{4.125}$$

$$= \tfrac{DB^2+\Sigma_a}{\nu\Sigma_f} - \tfrac{Df\left[\left(\frac{\pi}{a}\right)^2+\left(\frac{\pi}{Lyf}\right)^2\right]+\Sigma_a/f}{\nu\Sigma_f/f} \tag{4.126}$$

$$= \tfrac{DB^2+\Sigma_a}{\nu\Sigma_f} - \tfrac{D\left[\left(\frac{\pi}{a}\right)^2 f^2+\left(\frac{\pi}{b}\right)^2\right]+\Sigma_a}{\nu\Sigma_f} \tag{4.127}$$

$$= \tfrac{D}{\nu\Sigma_f}\left(\tfrac{\pi}{a}\right)^2(1-f^2) \tag{4.128}$$

The first derivative of this reactivity for the smallest perturbation magnitudes is

$$\left[\frac{d\rho}{df}\right]_{f=1} = -\frac{2D}{\nu\Sigma_f}\left(\frac{\pi}{a}\right)^2 \tag{4.129}$$

Now we perform the equivalent "virtual density" perturbation. Defining the perturbation is slightly trickier than for an expansion. When the core swells in only one direction, the leakage *in that one direction* does not change. We can shrink the neutron mean free path (in that one direction) by the factor f to account for the larger dimension, but the neutron mean free path has already been increased by the factor f due to the density reduction. Thus, there is a net zero change to leakage in that direction. However, *every other direction* does in fact experience a change in leakage. The material density has decreased, and so the neutron mean free paths in the other directions increases by the factor f. In order to define our perturbation, then, we simply decrease the macroscopic transport cross-section magnitudes in those other directions.

In this specific case, that means we decrease the macroscopic transport cross-section in the x direction. The macroscopic transport cross-section in the y direction does not change.

$$d\Sigma_{tr,x} = \Sigma'_{tr,x} - \Sigma_{tr,x} = \Sigma_{tr}\left(\frac{1}{f} - 1\right) \tag{4.130}$$

The unperturbed flux gradient in the x direction is

$$(\nabla\phi)_x = -\frac{\pi}{a}\sin\left(\frac{\pi x}{a}\right)\cos\left(\frac{\pi y}{b}\right) \tag{4.131}$$

The reactivity worth numerator is

$$\begin{aligned}\epsilon_y \left\langle \phi^\dagger \middle| \hat{L}_x\phi \right\rangle &= \textstyle\int_{-a/2}^{a/2}\int_{-b/2}^{b/2} dxdy\ (\nabla\phi)_x^2 \frac{1/f-1}{3\Sigma_{tr}} && (4.132)\\ &= \textstyle\left(\frac{1}{f}-1\right) D \left(\frac{\pi}{a}\right)^2 \frac{a}{2}\frac{b}{2} && (4.133)\end{aligned}$$

The denominator is

$$\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle = \int_{-a/2}^{a/2}\int_{-b/2}^{b/2} dxdy\, \nu\Sigma_f\phi^2 = \nu\Sigma_f \frac{a}{2}\frac{b}{2} \tag{4.134}$$

The entire reactivity worth is

$$\Delta\rho = 2\left(\frac{1}{f}-1\right)\frac{D}{\nu\Sigma_f}\left(\frac{\pi}{a}\right)^2 \tag{4.135}$$

As always, we have added the factor of 2. The slope of this function with respect to f is

$$\left[\frac{d\rho}{df}\right]_{f=1} = -\frac{2D}{\nu\Sigma_f}\left(\frac{\pi}{a}\right)^2 \tag{4.136}$$

This agrees precisely with the exact result we derived above. If we wish to use our shorthand notation (defined in Section 4.4.2), then we can rewrite this result as

$$\left[\frac{d\rho}{df}\right]_{f=1} = -2L_x \tag{4.137}$$

$$L_x = \frac{D}{\nu\Sigma_f}\left(\frac{\pi}{a}\right)^2 \tag{4.138}$$

Note that this is the 2-D version of Equation 4.42 for a uniform swelling in one direction.

Table 4.5: A 1-D swelling of a 3-D reactor.

actual expansion	virtual expansion
$c' = cf$	$\Sigma'_{tr,x} = \Sigma_{tr,x}/f$
$\Sigma' = \Sigma/f$	$\Sigma'_{tr,y} = \Sigma_{tr,y}/f$

4.6.5 1-D Swelling of a 3-D Reactor

Now let us consider the 1-D swelling of a homogenous 3-D reactor. This is equivalent to uniform axial swelling of a real reactor.

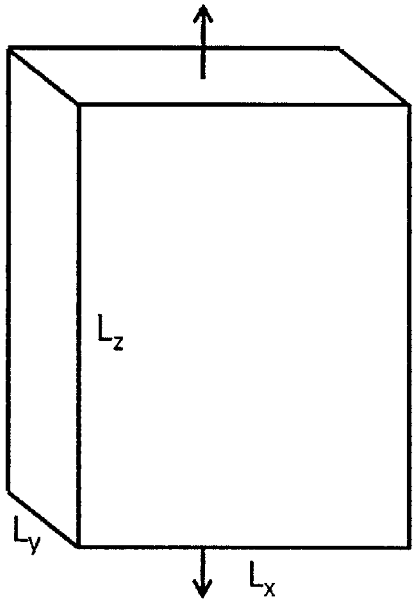

Figure 4.5: A 1-D swelling of a 3-D reactor.

The unperturbed flux is

$$\phi(x, y, z) = \cos\left(\frac{\pi x}{a}\right)\cos\left(\frac{\pi y}{b}\right)\cos\left(\frac{\pi z}{c}\right) \tag{4.139}$$

We define this swelling as an expansion of the z dimension and a corresponding uniform decrease of all the material densities.

$$f = \frac{c'}{c} \tag{4.140}$$

$$\Sigma' = \Sigma / f \tag{4.141}$$

The exact reactivity is

$$\begin{aligned}
\Delta\rho &= \tfrac{1}{k} - \tfrac{1}{k'} && (4.142)\\
&= \tfrac{DB^2+\Sigma_a}{\nu\Sigma_f} - \tfrac{DB'^2+\Sigma_a}{\nu\Sigma_f} && (4.143)\\
&= \tfrac{DB^2+\Sigma_a}{\nu\Sigma_f} - \tfrac{Df\left[\left(\frac{\pi}{a}\right)^2+\left(\frac{\pi}{b}\right)^2+\left(\frac{\pi}{L_z f}\right)^2\right]+\Sigma_a/f}{\nu\Sigma_f/f} && (4.144)\\
&= \tfrac{DB^2+\Sigma_a}{\nu\Sigma_f} - \tfrac{D\left[\left(\frac{\pi}{a}\right)^2 f^2+\left(\frac{\pi}{b}\right)^2 f^2+\left(\frac{\pi}{c}\right)^2\right]+\Sigma_a}{\nu\Sigma_f} && (4.145)\\
&= \tfrac{D}{\nu\Sigma_f}\left[\left(\tfrac{\pi}{a}\right)^2 + \left(\tfrac{\pi}{b}\right)^2\right](1-f^2) && (4.146)
\end{aligned}$$

The slope of this function with respect to f is

$$\left[\frac{d\rho}{df}\right]_{f=1} = -\frac{2D}{\nu\Sigma_f}\left[\left(\frac{\pi}{a}\right)^2 + \left(\frac{\pi}{b}\right)^2\right] \tag{4.147}$$

Now consider an equivalent "virtual density" perturbation. There is no change to leakage in the z direction - only leakage in the x and y directions decrease. So we decrease the macroscopic transport cross-sections in the x and y directions:

$$d\Sigma_{tr,x} = \Sigma'_{tr,x} - \Sigma_{tr,x} = \Sigma_{tr}\left(\frac{1}{f} - 1\right) \tag{4.148}$$

$$d\Sigma_{tr,y} = \Sigma'_{tr,y} - \Sigma_{tr,y} = \Sigma_{tr}\left(\frac{1}{f} - 1\right) \tag{4.149}$$

The unperturbed flux gradient in the x and y directions is

$$(\nabla\phi)_x = -\frac{\pi}{a}\sin\left(\frac{\pi x}{a}\right)\cos\left(\frac{\pi y}{b}\right)\cos\left(\frac{\pi z}{c}\right) \tag{4.150}$$

$$(\nabla\phi)_y = -\frac{\pi}{b}\cos\left(\frac{\pi x}{a}\right)\sin\left(\frac{\pi y}{b}\right)\cos\left(\frac{\pi z}{c}\right) \tag{4.151}$$

The reactivity worth numerator is

$$\begin{aligned}
\epsilon_z\left\langle\phi^\dagger\right|\left(\hat{L}_x+\hat{L}_y\right)\phi\Big\rangle &= \int_{-a/2}^{a/2}\int_{-b/2}^{b/2}\int_{-c/2}^{c/2} dxdydz\,\left[(\nabla\phi)_x^2 + (\nabla\phi)_y^2\right]\tfrac{1/f-1}{3\Sigma_{tr}} && (4.152)\\
&= \left(\tfrac{1}{f}-1\right) D\left[\left(\tfrac{\pi}{a}\right)^2 + \left(\tfrac{\pi}{b}\right)^2\right]\tfrac{a}{2}\tfrac{b}{2}\tfrac{c}{2} && (4.153)
\end{aligned}$$

The denominator is

$$\left\langle \phi^{\dagger} \middle| \hat{F}\phi \right\rangle = \int_{-a/2}^{a/2}\int_{-b/2}^{b/2}\int_{-c/2}^{c/2} dxdydz\, \nu\Sigma_f \phi^2 = \nu\Sigma_f \frac{a}{2}\frac{b}{2}\frac{c}{2} \tag{4.154}$$

So the total first order reactivity is

$$\Delta\rho = 2\left(\frac{1}{f} - 1\right)\frac{D}{\nu\Sigma_f}\left[\left(\frac{\pi}{a}\right)^2 + \left(\frac{\pi}{b}\right)^2\right] \tag{4.155}$$

The factor of 2 is here again. The slope of this curve for the smallest perturbations is

$$\left[\frac{d\rho}{df}\right]_{f=1} = -\frac{2D}{\nu\Sigma_f}\left[\left(\frac{\pi}{a}\right)^2 + \left(\frac{\pi}{b}\right)^2\right] \tag{4.156}$$

This agrees precisely with the exact result we derived above. If we wish to use our shorthand notation (defined in Section 4.4.2), then we can rewrite this result as

$$\left[\frac{d\rho}{df}\right]_{f=1} = -2\left(L_x + L_y\right) \tag{4.157}$$

$$L_x = \frac{D}{\nu\Sigma_f}\left(\frac{\pi}{a}\right)^2 \tag{4.158}$$

$$L_y = \frac{D}{\nu\Sigma_f}\left(\frac{\pi}{b}\right)^2 \tag{4.159}$$

Note that this is equivalent to Equation 4.42 for a uniform swelling in one direction, except that here the direction is z instead of x.

Table 4.6: A 2-D swelling of a 3-D reactor.

actual expansion	virtual expansion
$a' = af$	$\Sigma'_{tr,x} = \Sigma_{tr,x}/f$
$b' = bf$	$\Sigma'_{tr,y} = \Sigma_{tr,y}/f$
$\Sigma' = \Sigma/f^2$	$\Sigma'_{tr,z} = \Sigma_{tr,z}/f^2$

4.6.6 2-D Swelling of a 3-D Reactor

Now consider 2-D uniform swelling of a homogenous 3-D box reactor. This is equivalent to radial swelling of a real reactor.

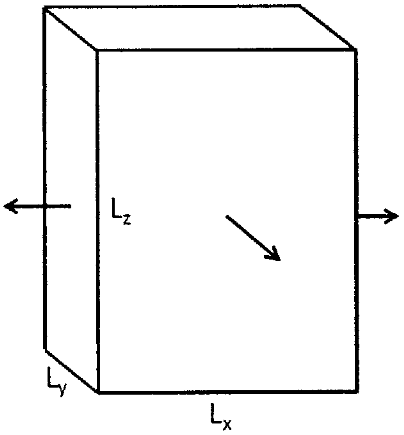

Figure 4.6: A 2-D swelling of a 3-D reactor.

The unperturbed flux is

$$\phi(x,y,z) = \cos\left(\frac{\pi x}{a}\right)\cos\left(\frac{\pi y}{b}\right)\cos\left(\frac{\pi z}{c}\right) \tag{4.160}$$

We can model this swelling by increasing both a and b by the same factor f. We must also uniformly decrease all material densities by the factor f^2 to conserve mass.

$$f = \frac{a'}{a} = \frac{b'}{b} \tag{4.161}$$

$$\Sigma' = \Sigma/f^2 \tag{4.162}$$

The exact reactivity is

$$\Delta\rho = \tfrac{1}{k} - \tfrac{1}{k'} \tag{4.163}$$

$$= \tfrac{DB^2+\Sigma_a}{\nu\Sigma_f} - \tfrac{DB'^2+\Sigma_a}{\nu\Sigma_f} \tag{4.164}$$

$$= \tfrac{DB^2+\Sigma_a}{\nu\Sigma_f} - \tfrac{Df^2\left[\left(\frac{\pi}{af}\right)^2+\left(\frac{\pi}{bf}\right)^2+\left(\frac{\pi}{c}\right)^2\right]+\Sigma_a/f^2}{\nu\Sigma_f/f^2} \tag{4.165}$$

$$= \tfrac{DB^2+\Sigma_a}{\nu\Sigma_f} - \tfrac{D\left[\left(\frac{\pi}{a}\right)^2f^2+\left(\frac{\pi}{b}\right)^2f^2+\left(\frac{\pi}{c}\right)^2f^4\right]+\Sigma_a}{\nu\Sigma_f} \tag{4.166}$$

$$= \tfrac{D}{\nu\Sigma_f}\left[\left(\left(\tfrac{\pi}{a}\right)^2+\left(\tfrac{\pi}{b}\right)^2\right)(1-f^2)+\left(\tfrac{\pi}{c}\right)^2(1-f^4)\right] \tag{4.167}$$

The slope of this function with respect to f is

$$\left[\frac{d\rho}{df}\right]_{f=1} = -\frac{2D}{\nu\Sigma_f}\left[\left(\frac{\pi}{a}\right)^2+\left(\frac{\pi}{b}\right)^2+2\left(\frac{\pi}{c}\right)^2\right] \tag{4.168}$$

Now we consider an equivalent "virtual density" perturbation. For each spatial direction, we multiply the macroscopic transport cross-section in that direction by the factor f if the reactor dimension has increased in that direction. We also divide the macroscopic transport cross-section in *all* directions by f^N, where N is the total number of reactor dimensions that have increased by the factor f. The resulting perturbation definition is

$$d\Sigma_{tr,x} = \Sigma'_{tr,x} - \Sigma_{tr,x} = \Sigma_{tr}\left(\frac{1}{f}-1\right) \tag{4.169}$$

$$d\Sigma_{tr,y} = \Sigma'_{tr,y} - \Sigma_{tr,y} = \Sigma_{tr}\left(\frac{1}{f}-1\right) \tag{4.170}$$

$$d\Sigma_{tr,z} = \Sigma'_{tr,z} - \Sigma_{tr,z} = \Sigma_{tr}\left(\frac{1}{f^2}-1\right) \tag{4.171}$$

The unperturbed flux gradients in all three directions are

$$(\nabla\phi)_x = -\frac{\pi}{a}\sin\left(\frac{\pi x}{a}\right)\cos\left(\frac{\pi y}{b}\right)\cos\left(\frac{\pi z}{c}\right) \tag{4.172}$$

$$(\nabla\phi)_y = -\frac{\pi}{b}\cos\left(\frac{\pi x}{a}\right)\sin\left(\frac{\pi y}{b}\right)\cos\left(\frac{\pi z}{c}\right) \tag{4.173}$$

$$(\nabla\phi)_z = -\frac{\pi}{c}\cos\left(\frac{\pi x}{a}\right)\cos\left(\frac{\pi y}{b}\right)\sin\left(\frac{\pi z}{c}\right) \tag{4.174}$$

The reactivity numerator is

$$\epsilon_{x,y}\left\langle\phi^\dagger\right|\left(\hat{L}_x+\hat{L}_y+\hat{L}_z\right)\phi\Big\rangle = \tag{4.175}$$

$$= \int_{-a/2}^{a/2}\int_{-b/2}^{b/2}\int_{-c/2}^{c/2} dxdydz \left[\left((\nabla\phi)_x^2 + (\nabla\phi)_y^2\right)\left(\frac{1}{f} - 1\right) + (\nabla\phi)_z^2\left(\frac{1}{f^2} - 1\right)\right]\frac{1}{3\Sigma_{tr}} \tag{4.176}$$

$$= D\left[\left(\left(\frac{\pi}{a}\right)^2 + \left(\frac{\pi}{b}\right)^2\right)\left(\frac{1}{f} - 1\right) + \left(\frac{\pi}{c}\right)^2\left(\frac{1}{f^2} - 1\right)\right]\frac{a}{2}\frac{b}{2}\frac{c}{2} \tag{4.177}$$

The reactivity denominator is

$$\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle = \int_{-a/2}^{a/2}\int_{-b/2}^{b/2}\int_{-c/2}^{c/2} dxdydz\, \nu\Sigma_f\phi^2 = \nu\Sigma_f\frac{a}{2}\frac{b}{2}\frac{c}{2} \tag{4.178}$$

The total first order reactivity is

$$\Delta\rho = 2\frac{D}{\nu\Sigma_f}\left[\left(\left(\frac{\pi}{a}\right)^2 + \left(\frac{\pi}{b}\right)^2\right)\left(\frac{1}{f} - 1\right) + \left(\frac{\pi}{c}\right)^2\left(\frac{1}{f^2} - 1\right)\right] \tag{4.179}$$

Again, we have added the factor of 2. The slope of this curve for the smallest perturbation magnitudes is

$$\left[\frac{d\rho}{df}\right]_{f=1} = -\frac{2D}{\nu\Sigma_f}\left[\left(\frac{\pi}{a}\right)^2 + \left(\frac{\pi}{b}\right)^2 + 2\left(\frac{\pi}{c}\right)^2\right] \tag{4.180}$$

This agrees precisely with the exact solution we derived above.

However, we must remember that

$$\frac{1}{k}\nu\Sigma_f - \Sigma_a = DB^2 = D\left[\left(\frac{\pi}{a}\right)^2 + \left(\frac{\pi}{b}\right)^2 + \left(\frac{\pi}{c}\right)^2\right] \tag{4.181}$$

So we can also write this first order result as

$$\left[\frac{d\rho}{df}\right]_{f=1} = -\frac{2D}{\nu\Sigma_f}\left(\frac{\pi}{c}\right)^2 - \frac{2\left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)}{\nu\Sigma_f} \tag{4.182}$$

This form of the reactivity might be simpler to compute, as it requires accurate evaluation of the leakage (and the flux gradient) in only one direction instead of in all three.

If we wish to use our shorthand notation (defined in Section 4.4.2), then we can rewrite this result as

$$\left[\frac{d\rho}{df}\right]_{f=1} = -2\left(L_x + L_y + 2L_z\right) = -2L_z - 2S \tag{4.183}$$

$$S = \frac{\left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)}{\nu\Sigma_f} \tag{4.184}$$

$$L_x = \frac{D}{\nu\Sigma_f}\left(\frac{\pi}{a}\right)^2 \tag{4.185}$$

$$L_y = \frac{D}{\nu\Sigma_f}\left(\frac{\pi}{b}\right)^2 \tag{4.186}$$

$$L_z = \frac{D}{\nu\Sigma_f}\left(\frac{\pi}{c}\right)^2 \tag{4.187}$$

Note that this is equivalent to Equation 4.42 for a uniform swelling in one direction, except that here the direction is z instead of x.

4.7 The Unsolved Problem of Non-Uniform "Virtual Density"

The whole history of "virtual density" consists of different people in different parts of the world conceiving the uniform isotropic case independently. This is not surprising, as the uniform isotropic case is quite obvious. As we mentioned in Section 3.3.3, early work at Los Alamos (as part of the Manhattan Project) employed relationships between density and dimensions for uniform isotropic cases [86]. Some of these simple relationships were used to design the first atomic bombs. A later report referencing this "law in criticality physics" stipulates "that the entire system be treated by the same factor throughout" [89]. So in western literature, it appears that application of the generic "virtual density" theory was limited to uniform isotropic cases.

Shikhov conceived the concept independently, and then the Soviets developed it further to apply to uniform anisotropic cases. No one, however, has successfully generalized the "virtual density" theory for *non-uniform* cases: swelling and expansion of interior zones. All previous work is uniform and whole-core. Thus, *this is an unsolved problem.*

4.7.1 A Conceptual Interpretation

One can glean additional conceptual insight directly from the k eigenvalue expression. Consider the general one-group anisotropic diffusion expression for the k eigenvalue of a generic reactor:

$$\frac{1}{k} = \frac{\int dxdydz\, M\phi}{\int dxdydz\, F\phi} = \frac{\int dxdydz\left(\Sigma_a - \nabla\cdot\hat{D}\nabla\right)\phi}{\int dxdydz\, \nu\Sigma_f\phi} \tag{4.188}$$

$$= \frac{\int dxdydz \left(\Sigma_a - \frac{d}{dx}D_x\frac{d}{dx} - \frac{d}{dy}D_y\frac{d}{dy} - \frac{d}{dz}D_z\frac{d}{dz}\right)\phi}{\int dxdydz\, \nu\Sigma_f\phi} \tag{4.189}$$

Now let us consider a uniform whole-core expansion in the x direction. If our theory is correct, we should be able to adjust the x-component of the diffusion coefficient to completely counteract the expansion so that k remains unchanged. Let $dx' = dxf$. Now let $D'_x = D_x f^2$. This will result in zero net change to the term $\frac{d}{dx}D_x\frac{d}{dx}\phi$. Also note that $\frac{d}{df}\left(f^2 - 1\right) = 2f$, so that this modification of D_x would produce the infamous factor of 2 seen repeatedly throughout Chapters 3-8. Of course, when integrating by dx' throughout the core, the factor f will cancel out in the numerator and denominator of the above equation. So we can modify D_x to fully and precisely counteract any uniform modification of the x coordinate.

Now, however, let us consider a *non*-uniform core expansion in the x direction. Consider that dx expands by different factors in different regions. In these cases, we can still modify D_x non-uniformly to keep the term $\frac{d}{dx}D_x\frac{d}{dx}\phi$ constant. However, now the dx' integrands will *not* cancel out in the numerator and denominator. Thus, we can *not* precisely counteract a modification of the x coordinate with a corresponding modification of D_x only.

This conundrum also holds true for swellings. Evidently, these whole-core uniform expressions fail for non-uniform expansions and swellings. Clearly, something is missing.

4.7.2 A Past Attempt at Non-Uniformity

In the 1990s, Abramov published a paper called "Calculation of Reactivity Effects in Deformed Zones of Nuclear Reactors" in which he attempts to do precisely what we do in this thesis - generalize the earlier uniform whole-core "virtual density" theory to apply to non-uniform localized cases [94]. Unfortunately, he is not successful. He shows an equality that is equivalent to Equation 4.13, and he also shows Equation 4.19 for anisotropic swelling. He claims that Equation 4.19 also applies to interior zones of a reactor, if only one integrates the numerator in each L_i value over only a single interior zone. Unfortunately, this assertion is false. He provides no numeric examples, and we have found that the expressions he derives do not work. If they produce suitable answers for some specific cases, that is merely accidental. We will see why in Chapter 5.

However, we are certainly glad that Abramov wrote this paper in the late 1990s, because it provides us reasonable assurance that the "virtual density" theory was never generalized to non-uniform cases in the Soviet literature. Abramov was a student of Shikhov, and the two co-authored several papers. Thus, it is extremely unlikely that Abramov would have been unaware of prior Russian work on this topic.

Solving the non-uniformity problem would dramatically enhance the usefulness of "virtual density" theory, which is currently only applicable to uniform radial and axial swelling (via the Soviets) along with uniform isotropic swelling or compression of weapon materials (via the Manhattan Project). In reality, fast reactor cores do *not* swell uniformly. Different fuel assemblies swell axially by different fractions, and radial swelling is *never* really uniform - it is a complex jumble of bowing and flowering with an axial profile determined by core restraint. Thus, substantial capabilities wait to be unlocked.

4.8 Summary

We have independently derived uniform anisotropic "virtual density" theory via anisotropic diffusion, and we have shown that it agrees with what we have unearthed of the old Soviet literature. We have demonstrated that it works precisely for simple 2-D and 3-D analytic solutions, and we will demonstrate it numerically for full-core heterogeneous benchmarks in Chapter 7. Furthermore, we have explained the problem of non-uniformity.

So far, this thesis has been largely a review and conceptual explanation of existing literature. Since much of this literature is obscure, not readily available, and not written in English, we took time to independently derive and cleanly lay out the formalism. In doing so, we have set the stage for our original non-uniform "virtual density" theory. Everything from this point onward is our original theory.

5 Non-Uniform Anisotropic "Virtual Density" Theory for Localized Distortions

5.1 Chapter Abstract

We conceive a more general form of "virtual density" theory - that applied to arbitrary non-uniform anisotropic swellings and expansions of interior zones within reactors. Previously, the "virtual density" theory has only been applied to whole-core uniform cases. While the western literature contains only brief mentions of the uniform *isotropic* theory, which is widely known, the Soviet literature extends the theory to uniform *anisotropic* cases. However, no previous literature contains the more general form of the theory, which is applicable to *non*-uniform anisotropic cases. Refer back to Figure 3.1 of an elucidating illustration.

In this work, we define new leakage terms via the divergence theorem that do not previously exist in the literature. We use these terms to derive a new "virtual density" formalism that successfully describes non-uniform anisotropic swellings and expansions. This more general formalism constitutes our original theory, and it allows for rapid and precise prediction of reactivity coefficients due to arbitrary non-uniform swelling scenarios (distortions) in reactors. For example, we can swiftly and precisely compute axial swelling reactivity coefficients for each individual assembly (and each of its axial segments).

5.2 Introduction

We now generalize the anisotropic "virtual density" theory for non-uniform localized swellings and expansions. We aim to devise a solution that is as elegant as the uniform anisotropic detailed in Section 4.4.

5.3 The Key to Non-Uniform "Virtual Density": The Divergence Theorem

In Section 3.4.1, we laid out the isotropic diffusion perturbation theory formalism, and we applied the divergence theorem to the adjoint-weighted leakage term in Equation 3.16. Here it is again:

$$\left\langle \phi^\dagger \middle| d\hat{L}\phi \right\rangle = -\int dV \left[\sum_{g=1}^{G} \left(\phi_g^\dagger \nabla \cdot dD_g \nabla \phi_g \right) \right] \tag{5.1}$$

$$= -\int d\vec{S} \cdot \left[\sum_{g=1}^{G} \left(\phi_g^\dagger \nabla \phi_g \right) dD_g \right] + \int dV \left[\sum_{g=1}^{G} \left(\nabla \phi_g^\dagger \cdot \nabla \phi_g \right) dD_g \right] \tag{5.2}$$

We chose to disregard the surface term, mostly because everyone else does. Using only the volume term for density changes is standard practice in traditional perturbation theory. The arguments for discarding the surface term are that (1) the adjoint flux vanishes on the outer reactor surface and (2) dD_g is zero beyond the extent of the density perturbation. With respect to (2), one can simply draw the "control volume" V to be slightly larger than the region in which dD is non-zero, and then the surface term becomes zero.

This reasoning is all quite valid for traditional density perturbations, but it is *not* valid for "virtual density" perturbations. Consider a localized anisotropic swelling within a certain interior cuboid volume enclosed by six planar surfaces. Since this anisotropic swelling does not cover the whole core, we must account for neutrons crossing the surfaces *perpendicular to all non-zero leakage changes.* When we say that a surface is perpendicular to a leakage change, we mean that the surface normal vector is *parallel* to the direction of the leakage that is changing. The conceptual basis for this goes back to neutron mean free paths - when we increase (or decrease) a mean free path in a certain direction within a localized region, we must account for the fact that some of those increased (or decreased) mean free paths will "straddle" the boundary of that region - a boundary that marks an abrupt "virtual" discontinuity in the mean free path and in the system dimensions scaling. Thus, we cannot simply choose to draw the divergence theorem "control volume" to extend beyond the swelling region, because this "virtual" discontinuity would still exist. In summary, the surface term in Equation 5.2 is zero for material density perturbations but *non*-zero for "virtual density" perturbations.

So in any "virtual density" perturbation, each L_i term should be accompanied by a directional surface term. Since no such surface terms are necessary in material density perturbations, this fact was likely missed by others.

However, there is still one problem. If swelling is anisotropic, how can we decompose the surface term into directional parts? This is not trivial, but we can make a guess based on intuition. In Equation 5.2, we see that the surface term contains the dot product of the surface vector integrand $d\vec{S}$ and the real flux gradient $\nabla\phi_g$. Here $d\vec{S}$ defines an arbitrary closed surface that forms the entire boundary of a swelling or expanding volume within a reactor. It is a unit vector normal to the surface and pointing outward. The dot product looks like this:

$$d\vec{S}\cdot\nabla\phi_g = \left(d\vec{S}\cdot\hat{x}\right)\left(\nabla\phi_g\right)_x + \left(d\vec{S}\cdot\hat{y}\right)\left(\nabla\phi_g\right)_y + \left(d\vec{S}\cdot\hat{z}\right)\left(\nabla\phi_g\right)_z \tag{5.3}$$

Here $\hat{x}$ is a unit vector in the x direction, and $\left(\nabla\phi_g\right)_x$ once again represents the magnitude of the x component of $\nabla\phi_g$. Just as we separated out directional terms for the volume quantity in Equation 4.10, we can also separate out directional terms for this surface quantity. We will use $\left\langle\phi^\dagger\middle|\,\hat{\Gamma}_i\phi\right\rangle$ to express this quantity in direction i.

$$\left\langle \phi^{\dagger} \middle| \hat{\Gamma}_i \phi \right\rangle = \int \left(d\vec{S} \cdot \hat{i}\right) \left[\sum_{g=1}^{G} \phi_g^{\dagger} \left(\nabla \phi_g\right)_i D_g \right] \tag{5.4}$$

We could also choose to write this quantity in terms of the real neutron current:

$$\left\langle \phi^{\dagger} \middle| \hat{\Gamma}_i \phi \right\rangle = -\int \left(d\vec{S} \cdot \hat{i}\right) \left[\sum_{g=1}^{G} \phi_g^{\dagger} \left(J_g\right)_i \right] \tag{5.5}$$

Here $(J_g)_i$ is the component of the real neutron current in direction i. Conceptually, this quantity is simply the adjoint-weighted rate at which neutrons traveling in direction i pass through the surface $\vec{S}$. We assume the convention that a positive $\nabla\phi$ points *outward* through the closed surface, while a positive $\vec{J}$ points *inward.*

As we shall see in the coming sections, these "directional surface current" quantities are the missing piece to the non-uniform "virtual density" theory. In uniform whole-core swellings, these surface quantities still exist in principle, but they are zero because the adjoint flux vanishes on the vacuum boundary. Also, the surface quantities between adjacent interior zones cancel each other out due to the swelling uniformity. All the previous work in this area, which is restricted to uniform whole-core applications, is merely a *special case* of the more general theory.

5.3.1 Non-Separability of the Diffusion Equation and the Divergence Theorem

The diffusion equation is *not* separable; every reactor physicist knows that. The divergence theorem is *not* separable; every mathematician knows that. When solving a system from scratch, one can *not* simply separate the diffusion equation (applied to neutrons, heat, or anything else) into three individual equations. The same is true of the divergence theorem, which is an integral part of not only neutron diffusion, but also heat transfer (via the heat equation), fluid mechanics (via the Navier-Stokes equations), and a myriad of other physical processes.

When we split the leakage volume term in Section 4.4.1 and the leakage surface term in Section 5.3, one might *think* that we assumed full separability of the divergence theorem and the diffusion equation. Fortunately, we did not. We never assumed separability for the purpose of solving systems from scratch. We only assumed separability for the purpose of applying perturbation theory to determine reactivity coefficients. Even though we can *not* disentangle the x, y, and z components of dot products when we solve an original unperturbed system, we *can* do so when we merely perturb that original system. This "limited separability" allows us to obtain accurate derivatives of eigenvalue with respect to anisotropic geometry perturbations.

5.3.2 Non-Uniform Spectral-Leakage Shorthand Notation

First let us add the scalar quantities Γ and Γ_i to our shorthand notation.

$$\Gamma_i = \frac{\left\langle \phi^\dagger \middle| \hat{\Gamma}_i \phi \right\rangle}{\left\langle \phi^\dagger \middle| \hat{F} \phi \right\rangle} \tag{5.6}$$

Note that the numerator here applies only to one interior zone, while the denominator is always integrated over the whole core. Of course, the isotropic Γ value is the sum of all three directional Γ_i values.

$$\Gamma = \sum_i \Gamma_i \tag{5.7}$$

The equality $S = L$ in the uniform theory has an analog in the non-uniform theory:

$$S = L - \Gamma = \sum_i (L_i - \Gamma_i) \tag{5.8}$$

This is the most general form of the equality, and it is applicable to any arbitrary zone within a reactor. Now we see that $S = L$ is really only a special case with very limited applicability.

5.3.3 Two Types of Leakage: Intra-Cell Volume Leakage and Inter-Cell Surface Leakage

Before moving on, it is important to note that L and Γ represent two distinct types of leakage. L is a volume integral, and it represents leakage within a core zone (a mesh cell in numeric calculations). Γ is a surface integral, and it represents leakage through a surface between two adjacent core zones (or two adjacent mesh cells). In the previous *uniform* "virtual density" formalism, opposing surface leakage terms canceled one another out in pairs on each surface. However, non-uniformity means that these pairs no longer cancel out, and so the Γ_i terms have net non-zero values. Thus, we can say that the failure of the uniform formalism is that it accounts for only one type of leakage (the intra-cell volume leakage) but not both types.

5.4 General Non-Uniform Formalism

Now we can proceed to lay out the general "virtual density" formalism for non-uniform swelling and expansion. This will roughly parallel the uniform whole-core formalism in Sections 4.4.3 - 4.4.6.

The reactivity due to arbitrary swelling of an *arbitrary interior zone* is

$$\Delta\rho_{\text{swell}} = \sum_i \epsilon_i \sum_{j \neq i} \left(-2L_j + \Gamma_j\right) \tag{5.9}$$

Compare this with Equation 4.19, which describes the uniform whole-core case. We have simply added a surface term Γ_j corresponding to each non-zero leakage change L_j. As we explained conceptually in Section 5.3, this accounts for the "virtual" discontinuity in the dimension scaling in direction j.

In order to construct multiple non-uniform swellings throughout a core, one can simply evaluate Equation 5.9 above for each individual swelling region separately - this theory is additive.

In order to convert this swelling expression into its corresponding expansion expression, we need to add a density expression. This is precisely unchanged from Equations 4.20 and 4.21. Traditional density perturbations, even when non-uniform and localized, require no surface terms.

$$\begin{aligned} \Delta\rho_{\text{den}} &= \sum_i \epsilon_i \left(S + L_x + L_y + L_z\right) && (5.10) \\ &= \sum_i \epsilon_i \left(S + \sum_j L_j\right) && (5.11) \end{aligned}$$

Now we can simply add the density perturbation (Equation 5.11) to the swelling perturbation (Equation 5.9) to obtain the expansion perturbation:

$$\Delta\rho_{\text{expan}} = \sum_i \epsilon_i \left[S + L_i + \sum_{j \neq i} \left(-L_j + \Gamma_j\right)\right] \tag{5.12}$$

At this point we can show expressions for non-uniform *isotropic* swelling and expansion just as we did in the uniform theory. However, we display these expressions with the significant warning that very few non-uniform isotropic swelling or expansion cases are geometrically feasible. For example, a single interior zone that does not touch the outer reactor boundary can *not* swell or expand isotropically, because it would (impossibly) overlap with adjacent zones. A rare example of a feasible isotropic scenario would be slicing a reactor in half and allowing one full half to swell isotropically. Thus, not every mathematical expression represents a feasible scenario. It is each reactor physicist's responsibility to ensure that these swelling or expansion equations are applied in ways that do not violate the laws of geometry. That said, the non-uniform isotropic swelling expression is

$$\left[\frac{d\rho}{df}\right]_{\text{swell,iso}} = -4L + 2\Gamma \tag{5.13}$$

So even isotropic swellings of interior zones require surface terms, although here the surface term in the divergence theorem is *not* split into directional components. Furthermore, we cannot express Equation 5.13 using only S as we did in the *uniform* isotropic case. The non-uniform isotropic *expansion* expression is

$$\left[\frac{d\rho}{df}\right]_{\text{expan,iso}} = 3S - L + 2\Gamma = 2L - \Gamma \tag{5.14}$$

Again, this is similar to what we found for the uniform case, except that it is *not* equivalent to the result of $2S$ in the uniform case. We cannot express it without surface terms, no matter how many different ways we apply Equation 5.8.

5.5 The Infinitude of Equalities for Non-Uniform Cases

Now we can formulate the infinite series of equivalent expressions for this new non-uniform theory just as we did for the old uniform theory. Using Equation 5.8, we can write an expression that is exactly zero:

$$S - L_x - L_y - L_z + \Gamma_x + \Gamma_y + \Gamma_z = 0 \tag{5.15}$$

$$S + \sum_i (-L_i + \Gamma_i) = 0 \tag{5.16}$$

Adding Equation 5.16 above to the swelling expression (Equation 5.9) in integer quantities M_i in the same way we did for the uniform theory yields

$$\Delta\rho_{\text{swell}} = \sum_i \epsilon_i \left[\sum_{j\neq i} (-2L_j + \Gamma_j) + M_i \left(S + \sum_j (-L_j + \Gamma_j)\right)\right] \tag{5.17}$$

Here the M_i values can each be any integer. Adding Equation 5.16 to the expansion expression (Equation 5.12) in the same way yields

$$\Delta\rho_{\text{expan}} = \sum_i \epsilon_i \left[\left(S + L_i + \sum_{j\neq i} (-L_j + \Gamma_j)\right) + M_i \left(S + \sum_j (-L_j + \Gamma_j)\right)\right] \tag{5.18}$$

Equations 5.17 and 5.18 constitute the most general expressions for non-uniform anisotropic swelling and expansion. The M_i values can be any integer.

For the sake of completeness, we can write down similar expressions for non-uniform *isotropic* swelling and expansion:

$$\left[\frac{d\rho}{df}\right]_{\text{swell,iso}} = (-4L + 2\Gamma) + M\,(S - L + \Gamma) \tag{5.19}$$

$$\left[\frac{d\rho}{df}\right]_{\text{expan,iso}} = (3S - L + 2\Gamma) + M\,(S - L + \Gamma) \tag{5.20}$$

Here M can be any integer.

5.6 Cartesian Application for Non-Uniform Cases

We can use Equation 5.17 to obtain an infinite series of equivalent expressions for swelling by a fraction f_x in the x direction in Cartesian coordinates. The most straightforward expression for $M_x = 0$ is

$$\left[\frac{d\rho}{df}\right]_{\text{swell,x}} = -2\,(L_y + L_z) + (\Gamma_y + \Gamma_z) \tag{5.21}$$

We use all possible M_x values in Equation 5.17 to obtain the infinite series:

$$\begin{align}
\left[\frac{d\rho}{df}\right]_{\text{swell,x}} &= -4S - 4(-L_x + \Gamma_x) + 2\,(L_y + L_z) - 3\,(\Gamma_y + \Gamma_z) \tag{5.22}\\
&= -3S - 3(-L_x + \Gamma_x) + (L_y + L_z) - 2\,(\Gamma_y + \Gamma_z) \tag{5.23}\\
&= -2S - 2(-L_x + \Gamma_x) - (\Gamma_y + \Gamma_z) \tag{5.24}\\
&= -S - (-L_x + \Gamma_x) - (L_y + L_z) \tag{5.25}\\
&= -2\,(L_y + L_z) + (\Gamma_y + \Gamma_z) \tag{5.26}\\
&= S + (-L_x + \Gamma_x) - 3\,(L_y + L_z) + 2\,(\Gamma_y + \Gamma_z) \tag{5.27}\\
&= 2S + 2\,(-L_x + \Gamma_x) - 4\,(L_y + L_z) + 3\,(\Gamma_y + \Gamma_z) \tag{5.28}
\end{align}$$

Now we can use Equation 5.18 to do the same thing for an *expansion* in the x direction:

$$\begin{align}
\left[\frac{d\rho}{df}\right]_{\text{expan,x}} &= -3S + 5L_x - 4\Gamma_x + 3\,(L_y + L_z) - 3\,(\Gamma_y + \Gamma_z) \tag{5.29}\\
&= -2S + 4L_x - 3\Gamma_x + 2\,(L_y + L_z) - 2\,(\Gamma_y + \Gamma_z) \tag{5.30}\\
&= -S + 3L_x - 2\Gamma_x + (L_y + L_z) - (\Gamma_y + \Gamma_z) \tag{5.31}\\
&= 2L_x - \Gamma_x \tag{5.32}\\
&= S + L_x - (L_y + L_z) + (\Gamma_y + \Gamma_z) \tag{5.33}\\
&= 2S + \Gamma_x - 2\,(L_y + L_z) + 2\,(\Gamma_y + \Gamma_z) \tag{5.34}\\
&= 3S - L_x + 2\Gamma_x - 3\,(L_y + L_z)\,3\,(\Gamma_y + \Gamma_z) \tag{5.35}\\
&= 4S - 2L_x + 3\Gamma_x - 4\,(L_y + L_z) + 4\,(\Gamma_y + \Gamma_z) \tag{5.36}
\end{align}$$

Although expansions of entire cores are not practical (because they do not conserve fuel mass), expansions of interior zones are actually quite common. For example, if assemblies are immersed in a coolant pool, then assembly motion can be expressed as a superposition of expanding (or contracting) interstitial coolant zones. Individual expansions do not conserve mass, but the superposition of multiple expansions *can* conserve mass. Also, sometimes coolant mass need not be conserved - core flowering scenarios involve widening of interstitial zones such that net coolant flows into the core.

Equation 5.32 is a particularly attractive option for expansions, because it requires only leakage quantities in the direction parallel to the expansion. This could potentially be very convenient, as one does not always know the neutron current vector in every direction at every point in a core.

In the same way that we related uniform whole-core expansions and swellings for isotropic and anisotropic cases, we can also relate the non-uniform quantities. First, we can always add one density perturbation to each swelling direction i in order to obtain an expansion perturbation.

$$\left[\frac{d\rho}{df}\right]_{\text{expan,i}} = \left[\frac{d\rho}{df}\right]_{\text{swell,i}} + \left[\frac{d\rho}{df}\right]_{\text{den}} \tag{5.37}$$

Thus, we can add three density perturbations to an isotropic swelling to obtain an isotropic expansion. Also, one anisotropic swelling in each direction is equivalent to a single isotropic swelling.

$$\sum_i \left[\frac{d\rho}{df}\right]_{\text{swell,i}} = \left[\frac{d\rho}{df}\right]_{\text{swell,iso}} = \left[\frac{d\rho}{df}\right]_{\text{expan,iso}} - 3\left[\frac{d\rho}{df}\right]_{\text{den}} \tag{5.38}$$

These relationships are intuitive and obvious, but they are still worth stating formally.

5.7 R-Z or Hexagonal-Z Application for Non-Uniform Cases

Now we turn back to fast reactors modeled in hexagonal-z or R-Z coordinate systems. We define L_r in the same way as in the uniform case, and we now also define Γ_r in a similar fashion.

$$L_r = L_x + L_y \tag{5.39}$$

$$\Gamma_r = \Gamma_x + \Gamma_y \tag{5.40}$$

We can easily convert Equations 5.22 - 5.28 to express radial swelling by substituting L_x with L_z, Γ_x with Γ_z, $L_x + L_y$ with L_r, and $\Gamma_x + \Gamma_y$ with Γ_r.

$$\begin{aligned}
\left[\frac{d\rho}{df}\right]_{\text{swell,ax}} &= -4S + 4\left(L_z - \Gamma_z\right) + 2L_r - 3\Gamma_r && (5.41)\\
&= -3S + 3\left(L_z - \Gamma_z\right) + L_r - 2\Gamma_r && (5.42)\\
&= -2S + 2\left(L_z - \Gamma_z\right) - \Gamma_r && (5.43)\\
&= -S + \left(L_z - \Gamma_z\right) - L_r && (5.44)\\
&= -2L_r + \Gamma_r && (5.45)\\
&= S - \left(L_z - \Gamma_z\right) - 3L_r + 2\Gamma_r && (5.46)\\
&= 2S - 2\left(L_z - \Gamma_z\right) - 4L_r + 3\Gamma_r && (5.47)
\end{aligned}$$

Equation 5.45 is attractive, because it requires knowledge of leakage quantities in only the radial direction. Also, note that if the swelling zone consists of an entire through-core axial slice, the surface term Γ_r will be zero. Thus, for some special cases, non-uniform swellings do not require computing surface terms.

Equation 5.44 is also potentially convenient. Consider a single assembly swelling axially by the same factor along its whole length. The surface term Γ_z will be zero, because the top and bottom of the assembly coincide with the top and bottom of the whole reactor. Thus, single assembly axial swelling (or multiple assemblies each swelling by a different factor) is another special case in which no surface terms are needed.

We can derive the non-uniform radial swelling expressions directly from Equation 5.17.

$$\begin{aligned}
\left[\frac{d\rho}{df}\right]_{\text{swell,rad}} &= -6S + 2L_z - 4\Gamma_z + 4L_r - 5\Gamma_r && (5.48)\\
&= -5S + L_z - 3\Gamma_z + 3L_r - 4\Gamma_r && (5.49)\\
&= -4S - 2\Gamma_z + 2L_r - 3\Gamma_r && (5.50)\\
&= -3S - L_z - \Gamma_z + L_r - 2\Gamma_r && (5.51)\\
&= -2S - 2L_z - \Gamma_r && (5.52)\\
&= -S - 3L_z + \Gamma_z - L_r && (5.53)\\
&= -4L_z + 2\Gamma_z - 2L_r + \Gamma_r && (5.54)\\
&= S - 5L_z + 3\Gamma_z - 3L_r + 2\Gamma_r && (5.55)\\
&= 2S - 6L_z + 4\Gamma_z - 4L_r + 3\Gamma_r && (5.56)
\end{aligned}$$

We can quickly spot two more special cases in which surface terms are not required - any expression that contains one of Γ_r and Γ_z but not both.

Equation 5.52 contains only Γ_r, which vanishes if the swelling region is a through-core axial slice. Thus, we can divide up a reactor into numerous axial slices and apply Equation

5.52 to model radial swelling with any axial profile. Of course, we could use *any* equation in the above list, but Equation 5.52 is most convenient due to its lack of surface terms.

The other special case is trickier. Equation 5.54 contains only Γ_z, which vanishes if the swelling region is a set of whole assemblies. However, single assemblies cannot simply swell without unphysically overlapping neighboring assemblies, and so this case is merely a specter. We will return to this conundrum in Chapter 11 when we attempt to model arbitrary single assembly motion in the radial plane.

Although we did not bother laying out equations for axial and radial *expansion* in the uniform whole-core theory, these scenarios do have potential practical application in the non-uniform theory. Refer back to the discussion of expanding interior coolant zones in Section 5.6. The axial expansion expressions are

$$\begin{aligned}
\left[\frac{d\rho}{df}\right]_{\text{expan,ax}} &= -3S + 5L_z - 4\Gamma_z + 3L_r - 3\Gamma_r && (5.57)\\
&= -2S + 4L_z - 3\Gamma_z + 2L_r - 2\Gamma_r && (5.58)\\
&= -S + 3L_z - 2\Gamma_z + L_r - \Gamma_r && (5.59)\\
&= 2L_z - \Gamma_z && (5.60)\\
&= S + L_z - L_r + \Gamma_r && (5.61)\\
&= 2S + \Gamma_z - 2L_r + 2\Gamma_r && (5.62)\\
&= 3S - L_z + 2\Gamma_z - 3L_r + 3\Gamma_r && (5.63)\\
&= 4S - 2L_z + 3\Gamma_z - 4L_r + 4\Gamma_r && (5.64)
\end{aligned}$$

The radial expansion expressions are

$$\begin{aligned}
\left[\frac{d\rho}{df}\right]_{\text{expan,rad}} &= -4S + 4L_z - 4\Gamma_z + 6L_r - 5\Gamma_r && (5.65)\\
&= -3S + 3L_z - 3\Gamma_z + 5L_r - 4\Gamma_r && (5.66)\\
&= -2S + 2L_z - 2\Gamma_z + 4L_r - 3\Gamma_r && (5.67)\\
&= -S + L_z - \Gamma_z + 3L_r - 2\Gamma_r && (5.68)\\
&= 2L_r - \Gamma_r && (5.69)\\
&= S - L_z + \Gamma_z + L_r && (5.70)\\
&= 2S - 2L_z + 2\Gamma_z + \Gamma_r && (5.71)\\
&= 3S - 3L_z + 3\Gamma_z - L_r + 2\Gamma_r && (5.72)\\
&= 4S - 4L_z + 4\Gamma_z - 2L_r + 3\Gamma_r && (5.73)
\end{aligned}$$

Note that we can obtain these expansion equations directly from the swelling equations by merely adding density perturbations of the form $S + L_r + L_z$. Here are a few useful and insightful equations for swelling and expansion in hexagonal-z geometry.

$$\left[\frac{d\rho}{df}\right]_{\text{swell,ax}} + \left[\frac{d\rho}{df}\right]_{\text{swell,rad}} = \left[\frac{d\rho}{df}\right]_{\text{swell,iso}} = \left[\frac{d\rho}{df}\right]_{\text{expan,iso}} - 3\left[\frac{d\rho}{df}\right]_{\text{den}} \tag{5.74}$$

$$\left[\frac{d\rho}{df}\right]_{\text{swell,ax}} + \left[\frac{d\rho}{df}\right]_{\text{den}} = \left[\frac{d\rho}{df}\right]_{\text{expan,ax}} \tag{5.75}$$

$$\left[\frac{d\rho}{df}\right]_{\text{swell,rad}} + 2\left[\frac{d\rho}{df}\right]_{\text{den}} = \left[\frac{d\rho}{df}\right]_{\text{expan,rad}} \tag{5.76}$$

This concludes our discussion of the non-uniform anisotropic "virtual density" formalism. The remainder of this chapter will focus on its numeric demonstration.

5.8 Summary and Originality

The theory we have developed throughout Chapter 5 is, as far as we know, an original contribution to diffusion theory. Although the most basic uniform isotropic "virtual density" concept is obvious, widely known, and mentioned sporadically throughout the western literature, it appears that no one made the leap to the less obvious anisotropic theory. Soviet reactor physicists developed the concept much further to include the uniform anisotropic theory, but no one leapt to the non-uniform theory. The sole attempt by Abramov in the late 1990s produced incorrect expressions that do not contain the surface leakage terms Γ_i in the divergence theorem [94]. His expressions are equivalent to the uniform anisotropic formalism we derive in Chapter 4. The modern Russian literature, which is accessible online, does not contain any other work on this topic that we have found. This is the basis upon which we conclude that our non-uniform anisotropic theory is most likely original.

This novel non-uniform theory dramatically enhances the applicability of the "virtual density" concept to a myriad of scenarios. Previously, the uniform isotropic principle was only applicable to the design of weapons and simple criticality experiments. Even the much more advanced uniform anisotropic theory developed by the Soviets is only useful for computing the uniform axial and radial swelling reactivity coefficients, which do not capture the true non-uniformity of core swelling. In contrast, our non-uniform anisotropic theory could potentially be widely applicable to a plethora of non-uniform swelling and distortion scenarios. It might even quantify neutronic effects due to seismic events, something which is not currently well known.

Figure 5.1 classifies different versions of the "virtual density" theory via a Venn diagram. Previously, all work on this topic fell within the and red circles. The enveloping blue circle represents our non-uniform anisotropic theory, which constitutes the more general version of the principle. All previous work by the Soviets is really a special case of this new theory. Since these theories are not collectively known by any standard name, we will dub our new general theory (along with its many special cases) the **"virtual density" theory of neutronics (VirDenT)**.

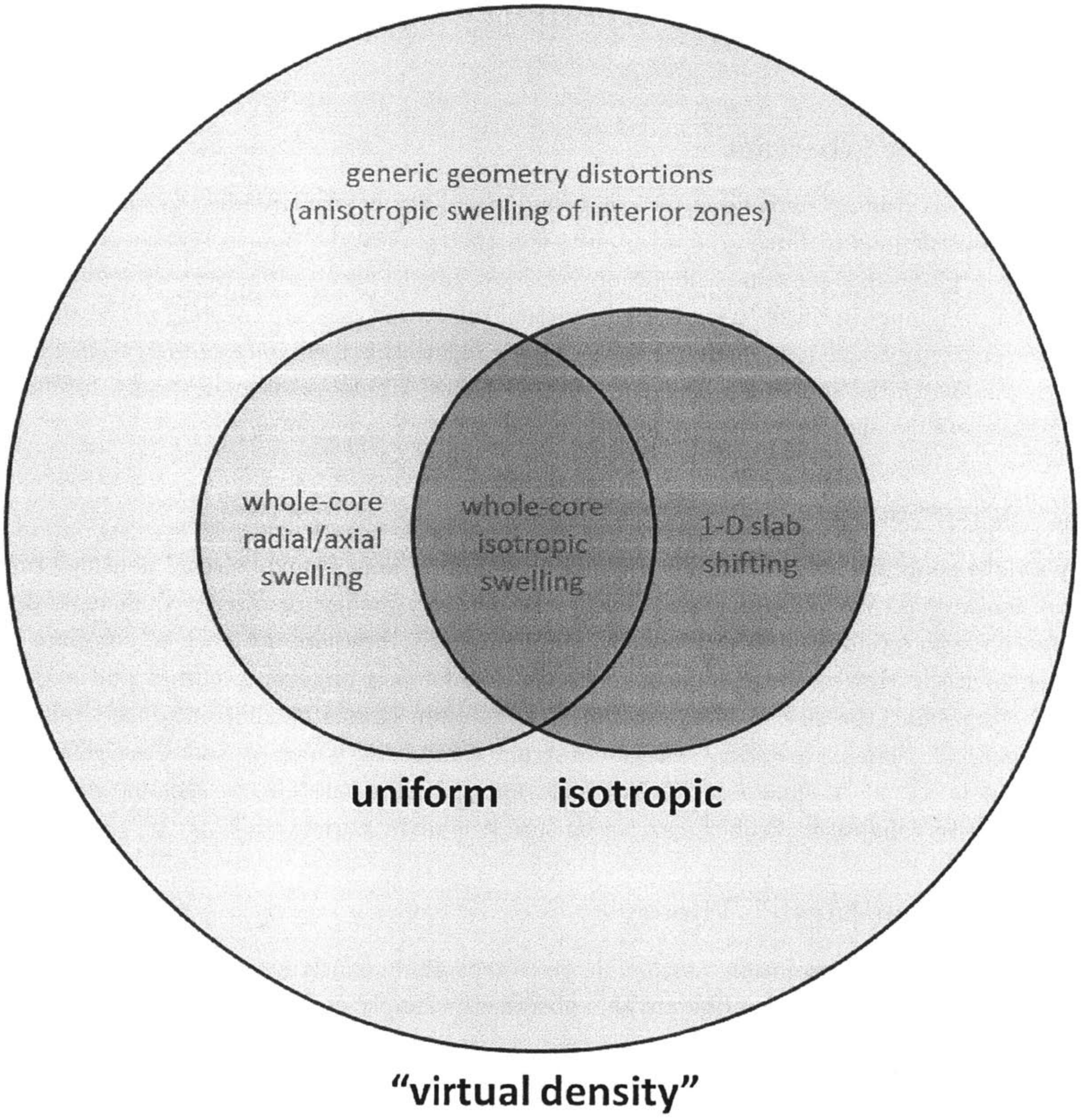
generic geometry distortions
(anisotropic swelling of interior zones)
whole-core radial/axial swelling
whole-core isotropic swelling
1-D slab shifting
uniform
isotropic
"virtual density"

Figure 5.1: Classification of "virtual density" theories. Everything within the green circle (and likely also the red circle) exists in prior literature, but the enveloping blue circle is our original contribution.

6 The "Virtual Mesh" Technique: Constructing a "Virtual Reference" for Geometry Change Reactivity Coefficients

6.1 Chapter Abstract

We develop an original technique for validating reactivity coefficients due to non-uniform geometry distortions in finite difference diffusion theory. Rather than construct an exotic mesh for a reference eigenvalue calculation, we show how to very simply perturb a standard Cartesian or triangle-z mesh to generate a "virtual reference" case from which we can obtain precise reactivity coefficient reference values. We validate this technique against continuous energy Monte Carlo and find agreement within the 1σ Monte Carlo uncertainty for non-uniform swellings and expansions of less than 1%.

6.2 Introduction

In previous chapters, we derived our original non-uniform anisotropic "virtual density" theory of neutronics. We claim that this theory can precisely predict reactivity coefficients due to non-uniform swellings and expansions. Unfortunately, these are not easy to compare to reference cases. How can one build a reference case for non-uniform swellings and expansions? Increasing the size of individual mesh cells often causes neighboring mesh cells to be misaligned. Thus, we cannot easily construct a mesh on which to solve the reference eigenvalue problem. Although our theory is elegant and self-consistent, we have no reference with which to validate it. So how can we be sure it is really correct?

6.3 "Virtual Mesh" Theory

Fortunately, there is a solution. Although we cannot easily construct a reference case with misaligned mesh cells, we can obviate this obstacle by simply constructing a "virtual reference" case with a "virtual mesh".

Validating uniform whole-core (radial and axial) swelling is simple. We construct a reference case by simply increasing the axial mesh spacing (for axial) or the assembly pitch (for radial) along with appropriate material density changes. Some non-uniform reference cases are also trivial to construct. For example, swelling of a through-core axial slice requires increasing the axial mesh spacing in only that slice - this does not cause misaligned mesh cells. However, what if we wish to construct a reference for axial swelling of a single assembly?

Consider Figures 6.1 and 6.2, which illustrate a system in an unperturbed and perturbed state, respectively. In Figure 6.1, the unperturbed "finite differences" dx and dy between neighboring Cartesian mesh cells are perpendicular. In Figure 6.2, a large central region of the core has swollen anisotropically in the x direction by a factor f. Thus, the dx and dy differences are no longer always perpendicular. The dx values in the swollen region have increased by a factor f, while the dy values have remained constant. The dy values *between* the swollen and unswollen regions are now oblique, and neighboring mesh cells are misaligned. This is the problem - how can we solve such a mesh?

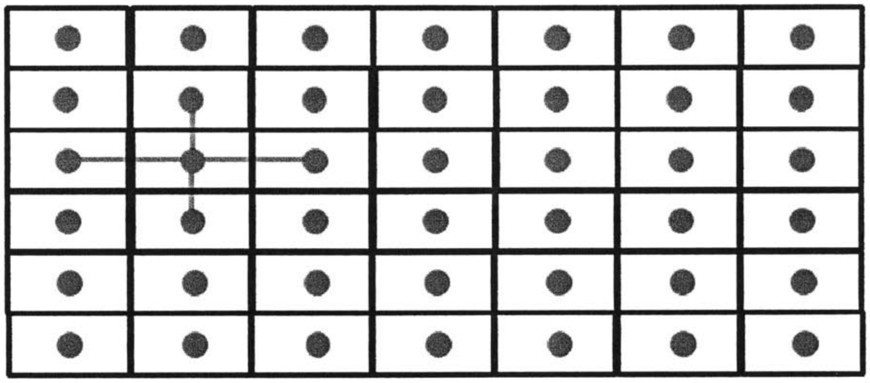

Figure 6.1: An unperturbed finite difference geometry. Finite differences are computed between each cell and its six neighbors (four neighbors in 2-D). The dx and dy differences, marked by red lines, are perpendicular.

Let us look more closely at these oblique dy differences. See Figure 6.3, in which we let s be the distance between neighboring (swollen and unswollen) mesh cell centroids. The unperturbed distance is a vertical s_0, while the perturbed difference is an oblique s. The lower mesh cell swells such that its centroid is displaced a distance α to the left. We know that s is

$$s = \sqrt{s_0^2 + \alpha^2} \tag{6.1}$$

The derivative of s with respect of α is

$$\frac{ds}{d\alpha} = \frac{\alpha}{\sqrt{s_0^2 + \alpha^2}} \tag{6.2}$$

The second derivative is

$$\frac{d^2s}{d\alpha^2} = \frac{s_0^2 - \alpha^2}{(s_0^2 + \alpha^2)^{3/2}} \tag{6.3}$$

These first and second derivatives evaluated at $\alpha = 0$ represent the *sensitivity* of s with respect to α, and the first derivative represents the *coefficient* of s with respect to α.

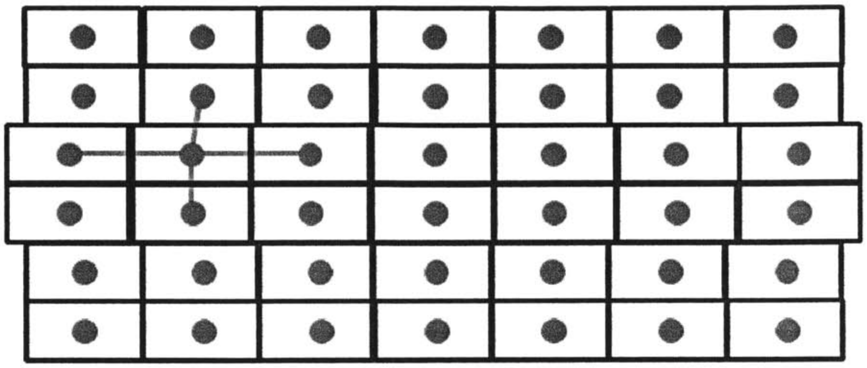

Figure 6.2: A perturbed finite difference geometry in which a central region swells anisotropically. Finite differences are computed between each cell and its six neighbors (four neighbors in 2-D). The dx and dy differences, marked by red lines, are no longer always perpendicular as they were in Figure 6.1. Specifically, the dy differences between the swollen and unswollen regions are oblique. Neighboring mesh cells are misaligned.

$$\left[\frac{ds}{d\alpha}\right]_{\alpha=0} = 0 \tag{6.4}$$

$$\left[\frac{d^2s}{d\alpha^2}\right]_{\alpha=0} = \frac{1}{s_0} \tag{6.5}$$

So the linear coefficient $ds/d\alpha$ is zero. This is all we need to know. It means that this swelling scenario will *not* affect the finite difference equations between swollen and unswollen regions *to first order*. Thus, because the finite difference equations entirely determine k_{eff}, changes in s between swollen and unswollen regions will not affect k_{eff} to first order. Changes in s will not affect the reactivity coefficient $d\rho/d\alpha$, and $d\rho/ds$ will be precisely zero.

As Equation 6.5 shows, the *second* derivative is non-zero, and so this mesh cell misalignment *will* affect k_{eff} in *second* order - the *second* derivative $d^2\rho/d\alpha^2$ will not be correct without considering changes in s.

So in order to accurately construct a reference mesh to compute $d\rho/df$ for the swelling case in Figure 6.2, we need only change each finite difference quantity q if and only if $dq/df \neq 0$ (when dq/df is evaluated at the unperturbed value $f = 1$). So in the case of Figure 6.2, we must only do two things: (1) decrease material densities in the swollen region by f and (2) increase dx in the swollen region by f. That's it. No alteration of dy (even between the swollen and unswollen regions) is necessary, and no fuss related to the misaligned mesh is necessary. Of course, this will create a mesh that is geometrically impossible, but that is of no concern. We can simply solve this "geometrically impossible" mesh to obtain eigenvalues

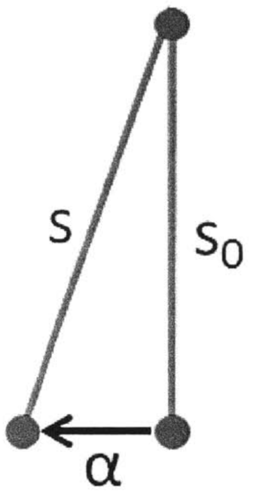

Figure 6.3: The centroids of two cells are initially separated by a vertical distance s_0, but they are separated by an oblique distance s in the perturbed case. Here the bottom mesh cell centroid shifts to the left by a distance α.

for $f = 1$ and $f = 1 + \delta$ (where δ is a very small number less than 0.01) and obtain the reactivity coefficient:

$$\left[\frac{d\rho}{df}\right]_{f=1} = \lim_{\delta \to 0} \frac{1/k_{f=1} - 1/k_{f=1+\delta}}{\delta} \tag{6.6}$$

In the limit of $\delta \to 0$, this will produce the reactivity coefficient *exactly*. No approximations are involved, except those already inherent within diffusion theory. We can succinctly summarize the general principle in an axiom:

Axiom: Let q be any arbitrary quantity in the finite difference equations. Let $[d\rho/dv]_{v=v_0}$ be any arbitrary linear reactivity coefficient with respect to some perturbed quantity v, which is v_0 in the unperturbed state. If and only if $[dq/dv]_{v=v_0} = 0$, then any change in q induced by a perturbation in v will have no effect on $[d\rho/dv]_{v=v_0}$.

This technique is really nothing more than eliminating all complexities in the system that do not affect the first derivative of reactivity with respect to the perturbation. We will dub this technique the "virtual mesh" method, because it allows us to construct a "virtual reference" case with which we can validate reactivity coefficients with respect to core geometry changes.

6.4 Monte Carlo Validation

6.4.1 Swelling of a Large Through-Core Slice

Even though the "virtual mesh" method makes sense intuitively, and even though we can prove it mathematically, prudence prompts us to validate it via continuous energy Monte Carlo.

We will start with the swelling case illustrated in Figure 6.2. Three 3-D slabs sit atop one another, each with the same composition: pure uranium metal at 7% enrichment. All three slabs have a length and width of 100 cm. The top and bottom slabs (which do not swell) have a thickness of 40 cm, while the middle slab (which swells) has a thickness of 20 cm. So the entire unperturbed core is a homogenous 100 cm x 100 cm x 100 cm cube. The perturbation consists of swelling the middle slab by a small factor in the x direction. The absence of reflectors means that strong transport leakage effects will strain any agreement between diffusion and Monte Carlo - this is a "stretch case".

We can build this precise geometry in MCNP to calculate the precise reactivity of this swelling in continuous energy Monte Carlo [202]. We can also build the unperturbed geometry in MaPS and test the "virtual mesh" method by building a "virtual mesh" to model the swelling. Essentially, a swelling factor of f requires reducing the uranium metal density in the middle slab by a factor f and increasing the mesh cell dx values by a factor f within that middle slab. Nothing else must be done.

Table 6.1 shows the Monte Carlo reactivities of various swelling magnitudes. All 1σ eigenvalue uncertainties are 1 pcm. Here swelling magnitudes are $f - 1$ expressed as a percentage. Thus, when the percentage is negative, the middle core slab actually contracts inward. Validation of reactivity coefficients via Monte Carlo is challenging, because (1) larger swelling magnitudes include higher order effects that obscure the linear coefficient and (2) smaller swelling magnitudes have large uncertainties that do not allow us to resolve the coefficient. Thus, we survey a wide range of swelling magnitudes.

Table 6.2 shows the finite difference diffusion reactivities for two swelling magnitudes (1% and 0.1%) generated using a "virtual mesh". We model this simple cubic core with a 10x10x10 mesh and a 20x20x20 mesh, and both spatial resolutions produce very similar results. Note that the unperturbed diffusion and Monte Carlo eigenvalues differ by over 900 pcm; this is probably due to strong transport leakage effects at the interface between metal fuel and vacuum.

In general, the coefficients computed via Monte Carlo (Table 6.1) and "virtual mesh" diffusion (Table 6.2) agree fairly well. All the coefficients computed via "virtual mesh" diffusion lie between -41.65 pcm/% and -41.95 pcm/%. This range lies within the Monte Carlo uncertainty for all swelling magnitudes less than 0.5%, although the uncertainties for smaller magnitudes are relatively large.

Figure 6.4 plots the Monte Carlo data points and fits them to a parabolic curve. The linear reactivity coefficient predicted via the "virtual reference" is also shown in red. Although the true reactivity curve represented by Monte Carlo is *not* quadratic, a parabola is the simplest polynomial that captures both (1) a distinct first derivative and (2) curvature to reflect higher order effects. The first derivative of this "best fit" parabola is -43.6 pcm/%.

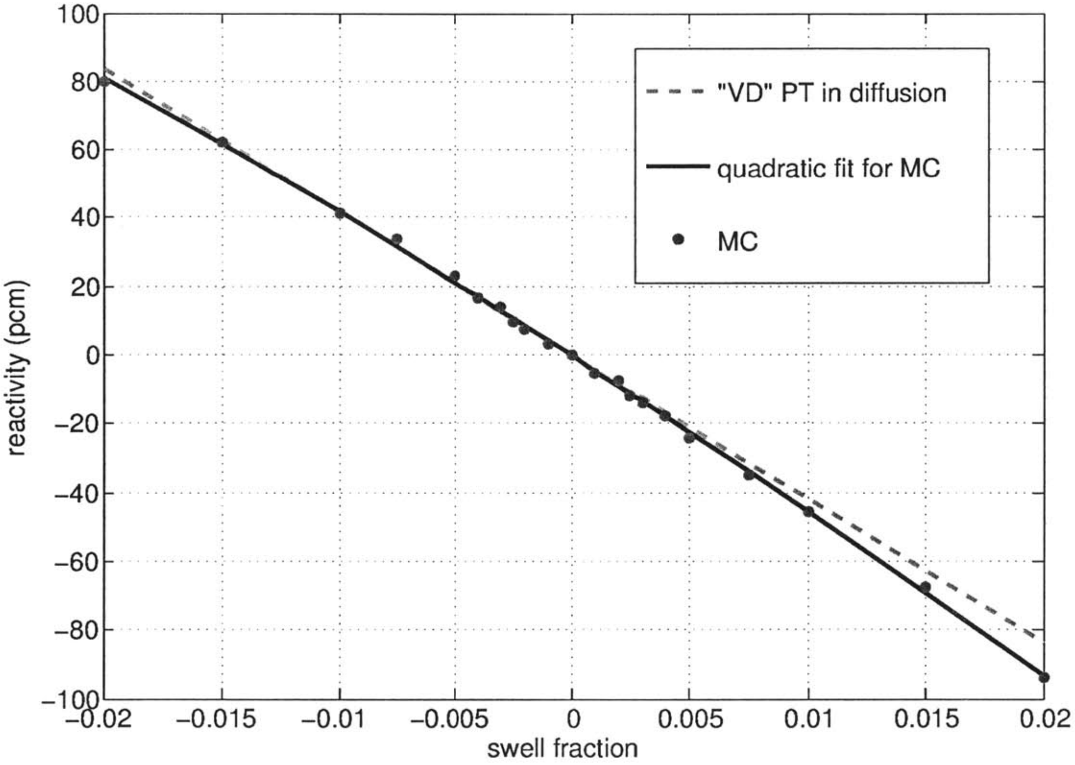

Figure 6.4: The reactivity due to swelling of a large through-core slice computed via Monte Carlo (the blue dots) fit to a parabolic curve (the black line) and compared with the "virtual mesh" coefficient (the red line). The "swell fraction" is equal to $f - 1$. Note that the Monte Carlo curve is slightly asymmetric.

Table 6.1: Monte Carlo (MCNP) Reactivity Due to Swelling of a Large Through-Core Slice

	k_{eff}	pcm	pcm/%
unperturbed	0.97430		
-2% swell	0.97506	80.0 ±1.4	-40.0 ±0.7
-1.5% swell	0.97489	62.1 ±1.4	-41.4 ±0.9
-1% swell	0.97469	41.1 ±1.4	-41.1 ±1.4
-0.75% swell	0.97462	33.7 ±1.4	-44.9 ±1.9
-0.5% swell	0.97452	23.2 ±1.4	-46.4 ±2.8
-0.4% swell	0.97446	16.9 ±1.4	-42.2 ±3.5
-0.3% swell	0.97443	13.7 ±1.4	-45.7 ±4.7
-0.25% swell	0.97439	9.5 ±1.4	-38.0 ±5.6
-0.2% swell	0.97437	7.4 ±1.4	-37.0 ±7.0
-0.1% swell	0.97433	3.2 ±1.4	-32.0 ±14.0
0.1% swell	0.97425	-5.3 ±1.4	-53.0 ±14.0
0.2% swell	0.97423	-7.4 ±1.4	-37.0 ±7.0
0.25% swell	0.97419	-11.6 ±1.4	-46.4 ±5.6
0.3% swell	0.97417	-13.7 ±1.4	-45.7 ±4.7
0.4% swell	0.97413	-17.9 ±1.4	-44.7 ±3.5
0.5% swell	0.97407	-24.2 ±1.4	-48.4 ±2.8
0.75% swell	0.97397	-34.8 ±1.4	-46.4 ±1.9
1% swell	0.97387	-45.3 ±1.4	-45.3 ±1.4
1.5% swell	0.97366	-67.5 ±1.4	-45.0 ±0.9
2% swell	0.97341	-93.8 ±1.4	-46.9 ±0.7

Table 6.2: "Virtual Mesh" Diffusion (MaPS) Reactivity Due to Swelling of a Large Through-Core Slice

	k_{eff}	pcm	pcm/%
10x10x10 unperturbed	0.983969097		
10x10x10 0.1% swell	0.983928486	-4.194110	-41.941101
10x10x10 1% swell	0.983565073	-41.746795	-41.746795
20x20x20 unperturbed	0.983245229		
20x20x20 0.1% swell	0.983204876	-4.174230	-41.742302
20x20x20 1% swell	0.982842528	-41.671354	-41.671354

6.4.2 Shifting of a Large Through-Core Slice

Now we consider an intriguing thought experiment. Figure 6.5 shows the same simple homogenous core, except now the central slab *shifts* instead of swells. There is no material density change - the whole middle of the core simply translates to the right or left. What will be the reactivity coefficient?

It will be precisely zero. Our "virtual reference" axiom shows that the oblique dy differences and the misaligned mesh cells do not affect the linear reactivity coefficient. Thus, applying "virtual density" perturbation theory *or* solving a "virtual reference" case will predict precisely zero change in eigenvalue. Figure 6.6 shows the Monte Carlo results fit to a parabolic curve. Note that the curve is symmetric and that its slope at the zero-shift point is precisely zero. So this simple, obvious case confirms that our intuition is consistent with the "virtual mesh" method.

Note that the reason for this zero reactivity coefficient is the same reason for the zero reactivity coefficient in the Godiva sphere translation problem studied by Favorite and discussed in Section 2.4.4. Favorite cannot predict the reactivity very well when translating the whole sphere (as opposed to each half sphere separately) [62]. As we discussed, Favorite's unexpectedly large error arises from the fact that the first derivative of eigenvalue is zero - only the second derivative is non-zero. So in the limit of the sphere displacement approaching zero, the computed reactivity coefficient will also approach zero.

If, instead, the slab shifted *and* changed density (or swelled) simultaneously, then the reactivity would have a non-zero first derivative and "virtual density" theory would predict that derivative exactly.

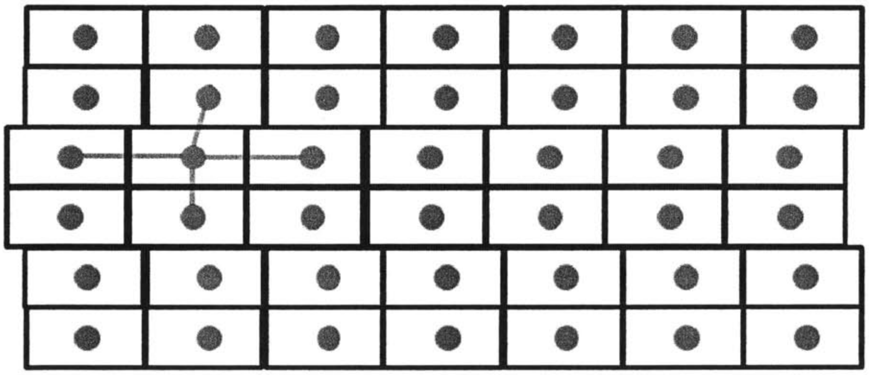

Figure 6.5: A perturbed finite difference geometry in which a central region shifts in one direction.

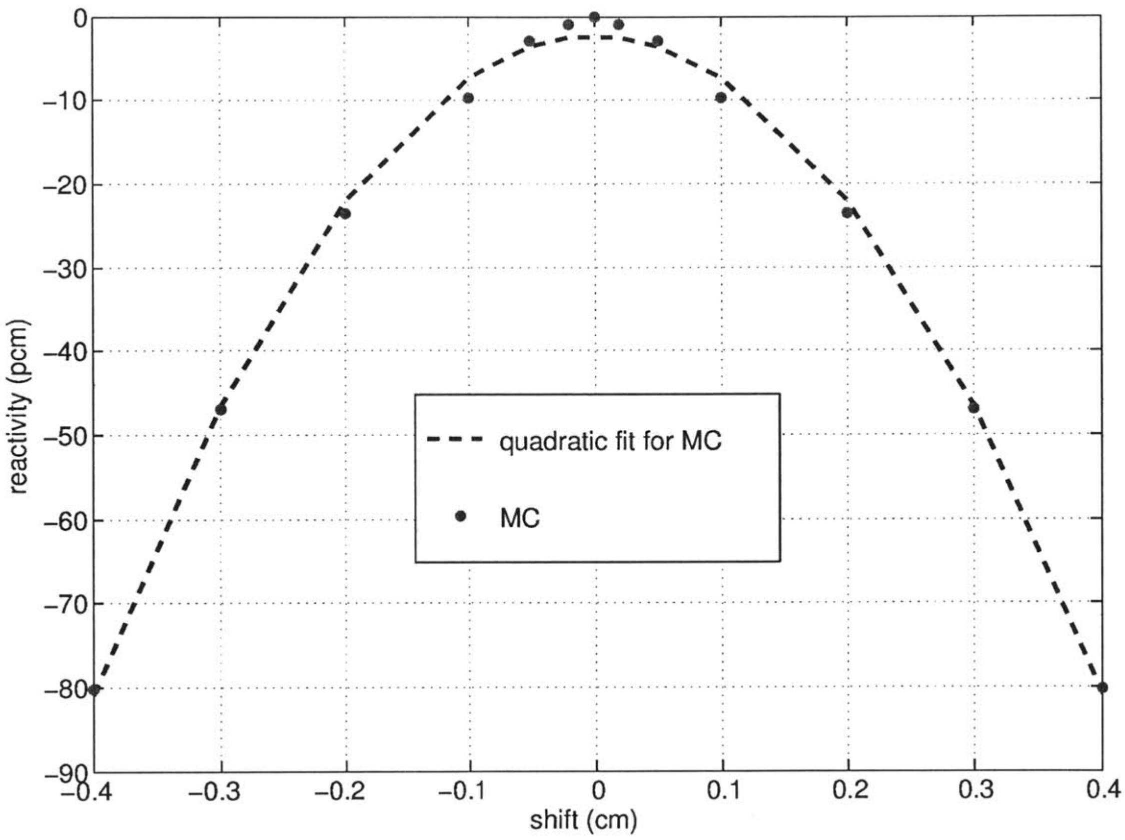

Figure 6.6: The reactivity due to shifting of a large through-core slice computed via Monte Carlo (the blue dots) fit to a parabolic curve (the black line). The parabolic fit does not capture the Monte Carlo data well near the zero-shift point, but that is only because the true transport theory curve is neither parabolic nor polynomic - we only choose a parabolic fit so that we can easily distinguish first and second eigenvalue derivatives.

6.4.3 Axial Swelling of a Single Assembly

Consider a 6x6 array of square assemblies as shown in Figure 6.7. Each assembly has side length b, and all assemblies are separated from one another by an interstitial width a. The geometry is constant through its axial height c. We will use this geometry often throughout the remainder of this thesis.

Suppose that a single assembly swells axially by a factor f such that its height increases from c to fc and its density decreases by f. Suppose that each of the 36 assemblies can be identified by the index pair (i, j), where i = 1-6 and j = 1-6. We will select assembly (4,4) to swell axially. See Figure 6.8 for a side view illustration. We build this scenario in both MCNP and MaPS, and Table 6.3 shows the results. The "virtual mesh" method predicts a reactivity coefficient of -14.7262 pcm per percent swelling. All the MCNP runs agree with this to within the Monte Carlo uncertainty.

Now suppose that the four adjacent assemblies (4,4), (4,5), (5,4), and (5,5) all swell axially by the same factor. Table 6.4 shows the results. The "virtual reference" method predicts a reactivity coefficient of -46.8396 pcm per percent swelling. All the MCNP runs for swellings of 1.5% or less agree with this to within the Monte Carlo uncertainty. Four assemblies swelling together produces a much larger reactivity than one assembly swelling alone, and thus the reactivity as a function of swelling magnitude has more curvature. Therefore, we expect the MCNP results to depart from the linear coefficient at relatively large swelling magnitudes.

Figure 6.9 shows both cases (one assembly and four assemblies) on the same axes. We fit a parabolic curve to the MCNP data points, and we compare it to the linear coefficient computed via a "virtual reference" in MaPS. Although the Monte Carlo data acquires curvature (due to its second derivative) for relatively large swelling magnitudes in the four assembly case, the "virtual reference" clearly predicts the first derivative correctly.

Note that the degree to which reactivity has "curvature" usually depends more on the magnitude of the reactivity, not the magnitude of the perturbation. We have observed this throughout a multitude of various perturbation types.

Table 6.3: "Virtual Reference" Validation via Monte Carlo for a Single Assembly Swelling Axially

	k_{eff}	pcm	pcm/%
"virtual reference"			-14.7264
MC unperturbed	0.99761		
MC 0.5% swell	0.99753	-8.0 ± 1.4	-16.0 ± 2.8
MC 0.7% swell	0.99751	-10.0 ± 1.4	-14.3 ± 2.0
MC 1% swell	0.99747	-14.1 ± 1.4	-14.1 ± 1.4
MC 1.5% swell	0.99739	-22.1 ± 1.4	-14.7 ± 0.9
MC 2% swell	0.99732	-29.1 ± 1.4	-14.6 ± 0.7
MC 2.5% swell	0.99723	-38.2 ± 1.4	-15.3 ± 0.6
MC 3% swell	0.99716	-45.2 ± 1.4	-15.1 ± 0.5

Table 6.4: "Virtual Reference" Validation via Monte Carlo for Four Assemblies Swelling Axially

	k_{eff}	pcm	pcm/%
"virtual reference"			-46.8396
MC unperturbed	0.99761		
MC 0.2% swell	0.99753	-8.0 ± 1.4	-40.0 ± 7.0
MC 0.5% swell	0.99738	-23.1 ± 1.4	-46.2 ± 2.8
MC 0.7% swell	0.99729	-32.2 ± 1.4	-46.0 ± 2.0
MC 1% swell	0.99715	-46.2 ± 1.4	-46.2 ± 1.4
MC 1.5% swell	0.99690	-71.4 ± 1.4	-47.6 ± 0.9
MC 2% swell	0.99665	-96.6 ± 1.4	-48.3 ± 0.7
MC 2.5% swell	0.99638	-123.7 ± 1.4	-49.5 ± 0.6
MC 3% swell	0.99610	-152.0 ± 1.4	-50.7 ± 0.5

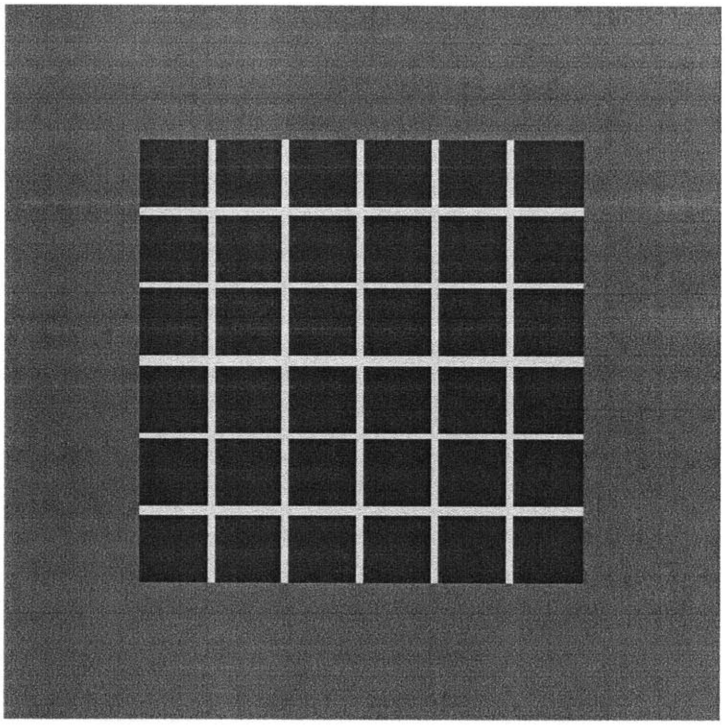

Figure 6.7: An unperturbed 6x6 array of square assemblies. Each assembly (blue) has a side length of $b = 10$ cm. The interstitial sodium coolant zones (green) are $a = 1$ cm thick. The outer natural uranium reflector (red) is $2b = 20$ cm thick. The geometry is constant through its axial height, which is $c = 110$ cm. The fuel assemblies have a 38% coolant volume fraction, and the metal fuel is enriched to 13%. We neglect structural materials. Typically, we set a to be 1 mesh cell thick, b to be 2 or 3 mesh cells thick, and c to be 20 mesh cells thick.

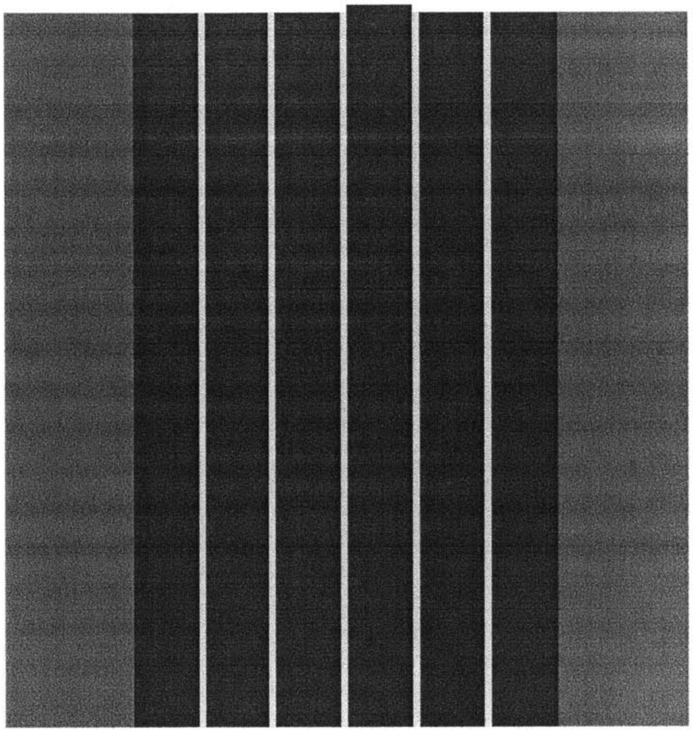

Figure 6.8: One assembly swelling axially. This is a side view of the same geometry shown in Figure 6.7.

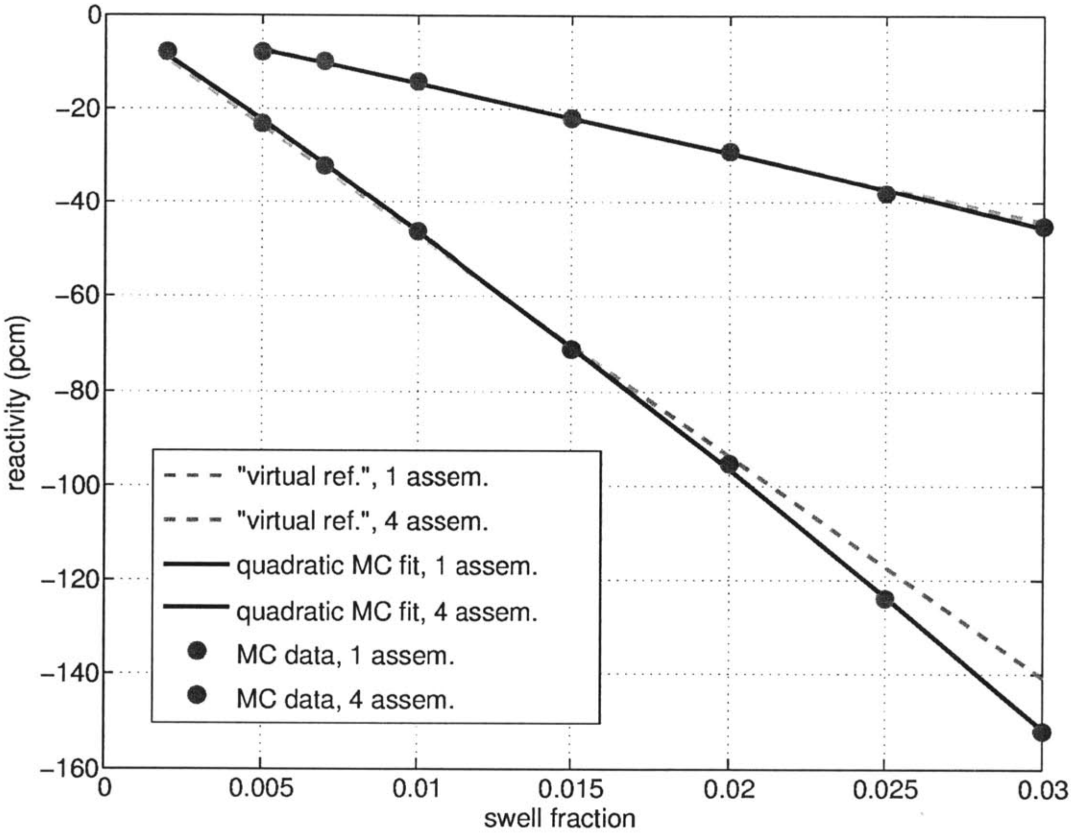

Figure 6.9: Comparing the "virtual mesh" method against Monte Carlo for one and four assemblies swelling axially. The Monte Carlo calculations are performed via MCNP, and the "virtual mesh" method is implemented via MaPS finite difference diffusion. The linear reactivity coefficient computed via the "virtual mesh" method accurately captures the first derivative of the true reactivity to within the Monte Carlo uncertainty, which is ≈ 1.4 pcm for all data points.

6.4.4 "Swinging" of an Assembly Row

Now let us try a more complex case. Consider the same geometry that is shown in Figure 6.7, which we just used for the assembly axial swelling. Now suppose that an entire assembly row (six assemblies) remains fixed at the core top but free at the core bottom. Thus, this row can "swing" back and forth as if it were hinged at the core top. See Figure 6.10 for an MCNP geometry illustration.

In order to model this scenario in "virtual density" theory, we recognize that the "swinging" motion is equivalent to anisotropically expanding and contracting the interstitial coolant zones on either side of the assembly row (with an axial dependence). This is *not* a swelling case, because the interstitial coolant density does not change - it only changes volume. We anisotropically *expand* the coolant to the *right* of the "swinging" assembly row, and we anisotropically *contract* the coolant to the *left* of the "swinging" row.

We can apply Equation 5.32, which becomes

$$\Delta\rho \approx \epsilon \left(\frac{z_{\text{top}} - z}{z_{\text{top}}} \right) \left([2L_x - \Gamma_x]_R - [2L_x - \Gamma_x]_L \right) \tag{6.7}$$

Here the subscripts R and L denote spatial integrals over the interstitial sodium regions to the right and left of the "swinging" row. The core top is at position $z = z_{\text{top}}$. The small parameter $\epsilon = f - 1$ represents the maximum interstitial zone expansion at the core bottom.

In order to construct a "virtual reference" case for this scenario, we must do nothing more than vary the dx finite difference within the interstitial coolant regions like this:

$$dx_R = dx_0 \left(1 + z_{\text{top}} - z\right) \tag{6.8}$$

$$dx_L = dx_0 \left(1 - (z_{\text{top}} - z)\right) \tag{6.9}$$

Here dx_0 is the initial interstitial coolant zone width $a = 1$ cm.

Table 6.5 shows the numeric comparison between the "virtual reference" linear coefficient and the Monte Carlo reactivities. Here a "full swing" indicates that the assembly row bottom just barely touches the neighboring assembly row, and a "half swing" is half as much movement. Figure 6.11 plots this same data, which is highly linear.

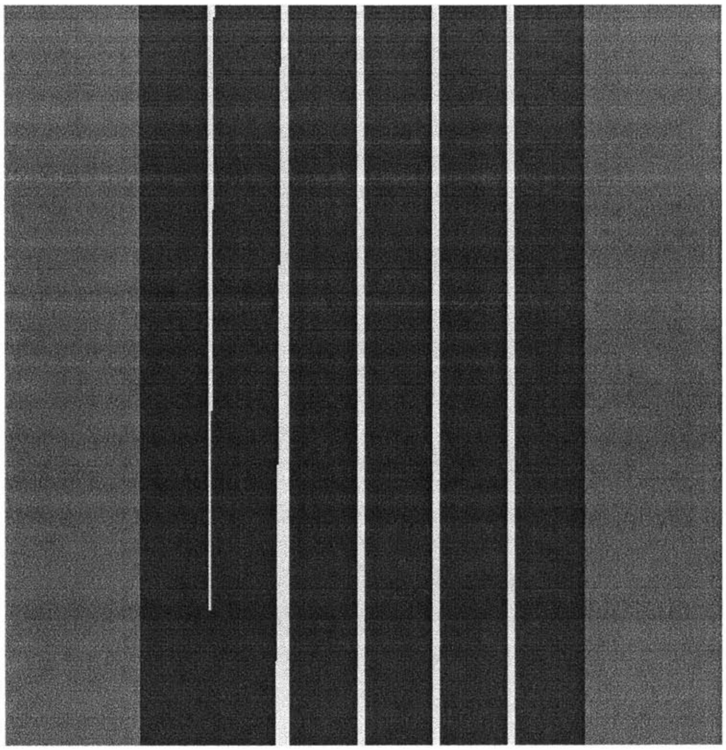

Figure 6.10: One row of assemblies "swinging" outward. This is a side view of the same geometry as in Figure 6.7, the same side view (with a different perturbation) as in Figure 6.8. In this case, the assembly row "swings" enough to just barely touch the neighboring assembly row at the core bottom. We increase the number of axial mesh cells from 20 to 40.

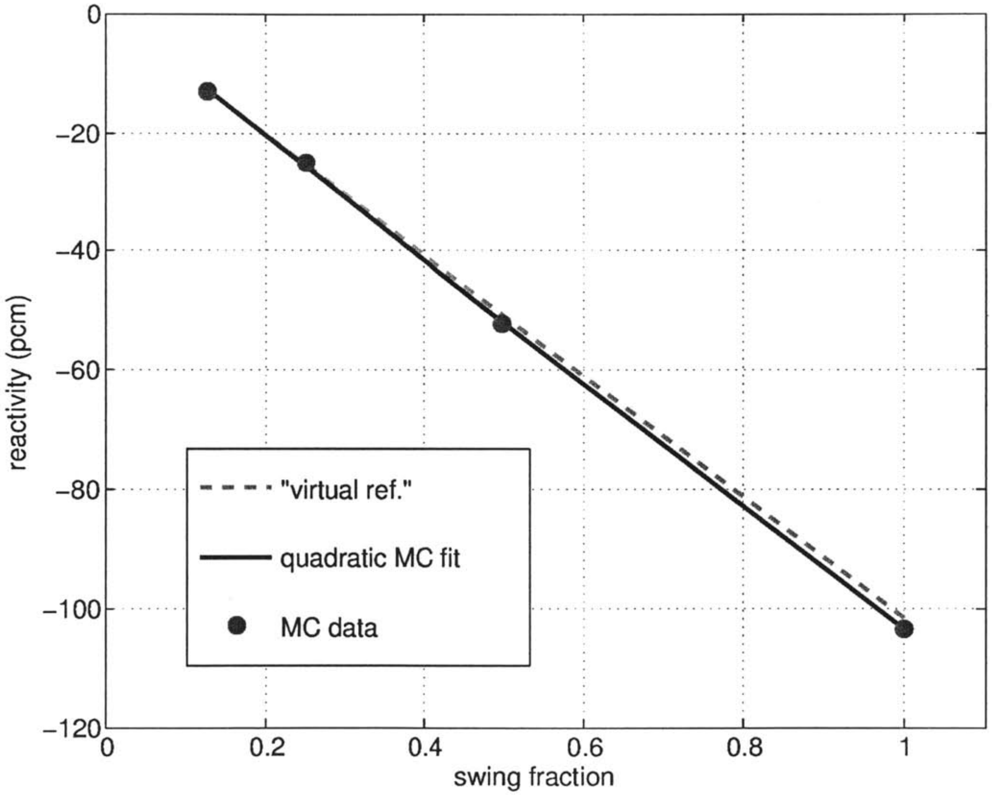

Figure 6.11: Comparing the "virtual mesh" method against Monte Carlo for one assembly row "swinging" outward as shown in Figure 6.10. The Monte Carlo calculations are performed via MCNP, and the "virtual mesh" method is implemented via MaPS finite difference diffusion. The "swing fraction" is the fraction by which the assembly row bottom "swings" over to touch the neighboring row.

Table 6.5: "Virtual Reference" Validation via Monte Carlo for One Assembly Row "Swinging"

	k_{eff}	pcm	pcm per full swing
"virtual reference"			-101.503
MC unperturbed	0.99761		
MC full swing	0.99865	-103.4 ± 1.4	-103.4 ± 1.4
MC half swing	0.99814	-52.2 ± 1.4	-104.4 ± 2.8
MC quarter swing	0.99787	-25.1 ± 1.4	-100.4 ± 5.6
MC eighth swing	0.99775	-13.1 ± 1.4	-104.8 ± 11.2

6.5 Summary and a Warning

We have now demonstrated that we can easily construct a "virtual mesh" in order to solve a "virtual reference" case for non-uniform anisotropic swellings and expansions within reactor cores. The linear coefficients agree quite well with continuous energy Monte Carlo calculations via MCNP, usually within 1σ for small perturbations.

However, we must stress to any potential users that a "virtual mesh" will only accurately predict the *linear* reactivity coefficient for very small perturbation magnitudes. Typically, we choose swelling magnitudes of 0.1% or even less to ensure accuracy. Constructing and solving a "virtual reference" case for much larger swellings will *not* produce a meaningful reference, because non-linear effects become significant.

7 Numeric Validation of Non-Uniform Anisotropic "Virtual Density" Theory

7.1 Chapter Abstract

We numerically validate the non-uniform anisotropic "virtual density" theory in finite difference diffusion using standard reference cases and, when necessary, "virtual mesh" reference cases. We find that the non-uniform anisotropic "virtual density" perturbation theory predicts reactivity coefficients that agree precisely with all diffusion reference cases. We verify this for simple 2-D and 3-D Cartesian cases as well as for various swelling scenarios in full-core 3-D hexagonal-z FFTF and Jōyō benchmarks. Reference case errors are typically below 0.1%, but we show that all errors approach zero as the reference case perturbation magnitude approaches zero. Thus, the non-uniform anisotropic "virtual density" formalism predicts diffusion reactivity coefficients *exactly.*

7.2 Introduction

Now that we have derived the original non-uniform anisotropic "virtual density" perturbation theory and also developed a new technique with which to validate it, the time has come to produce convincing numbers for realistic cases! First, however, we must show how to derive finite difference equations for the non-uniform anisotropic "virtual density" theory.

7.3 Finite Difference Equations for the Non-Uniform "Virtual Density" Theory

Although we have laid out the "virtual density" formalism and demonstrated it for very simple analytic cases, we have yet to show how one might discretize it for implementation in a finite difference solution. We will do that now for both Cartesian and triangular/hexagonal cases. Here we assume that both real and adjoint multigroup flux distributions have already been solved on a finite difference mesh for the unperturbed system. The finite difference equations could be set up in more than one way, so we must be cautious to construct the perturbation difference equations in a way that is *precisely* consistent with the difference equations used to solve the unperturbed system.

7.3.1 Cartesian Coordinates

Evaluation of the "virtual density" shorthand quantities S, L_i, and Γ_i in a finite difference solution is not difficult, but it *can* be ambiguous. See Figure 7.1 and consider evaluating these quantities for the central rectangular cell. Computing the perturbation denominator

$\left\langle \phi^{\dagger} \middle| \hat{F}\phi \right\rangle$ is straightforward - one only needs the cell fission cross-section and the volume-averaged fluxes ϕ and $\phi^{\dagger}$. Computing the numerator of S is similarly straightforward - one only needs the fission, absorption, and scattering cross-sections in the cell as well as k_{eff}. We will not review these here.

Computing L_i and Γ_i is a bit less straightforward. Consider computation of L_x and Γ_x in Figure 7.1. A 2-D rectangular cell is surrounded by surfaces 1-4 and neighboring cells 1-4. We can obtain the surface fluxes ϕ_{s1} and $\phi_{s3)}$:

$$\phi_{s1} = \frac{\frac{D_0\phi_0}{dx/2} + \frac{D_1\phi_1}{dx_1/2}}{\frac{D_0}{dx/2} + \frac{D_1}{dx_1/2}} \tag{7.1}$$

$$\phi_{s3} = \frac{\frac{D_0\phi_0}{dx/2} + \frac{D_3\phi_3}{dx_3/2}}{\frac{D_0}{dx/2} + \frac{D_3}{dx_3/2}} \tag{7.2}$$

This is nothing more than applying Fick's law while conserving current at surfaces 1 and 3. The expressions for the adjoint $\phi_s^{\dagger}$ values are identical in form. Now consider computing two flux gradients in the x direction within the central cell:

$$\nabla\phi_1 = \frac{\phi_{s1} - \phi_0}{dx/2} \tag{7.3}$$

$$\nabla\phi_3 = \frac{\phi_{s3} - \phi_0}{dx/2} \tag{7.4}$$

One could say that $\nabla\phi_1$ is the x-directed gradient in the right half of the cell, while $\nabla\phi_3$ is the x-directed gradient in the left half of the cell. Both gradients point *outward* from the cell, toward its neighbors. Now the numerator of L_x for one energy group is

$$\left\langle \phi^{\dagger} \middle| \hat{L}_x\phi \right\rangle = \frac{1}{2}dxdy\left(\nabla\phi_1^{\dagger}\nabla\phi_1 + \nabla\phi_3^{\dagger}\nabla\phi_3\right)D_0 \tag{7.5}$$

This is how we evaluate the volume integral in Equation 4.11 in finite difference. Computation of L_y is the same process, and extension to 3-D Cartesian is trivial.

Multiplying the real and adjoint gradients in Equation 7.5 nullifies the direction in which we chose to compute the gradients - we could just have easily computed them to point inward to the cell or in the positive or negative x directions. The factor of 1/2 represents averaging the two gradient products over the whole cell.

Now the numerator of Γ_x for one energy group is

$$\left\langle \phi^{\dagger} \middle| \hat{\Gamma}_x\phi \right\rangle = dy\left(\phi_{s1}^{\dagger}\nabla\phi_1 + \phi_{s3}^{\dagger}\nabla\phi_3\right)D_0 \tag{7.6}$$

This is how we evaluate the surface integral in Equation 5.4 in finite difference. Again, computation of L_y is the same process, and extension to 3-D Cartesian is trivial. Note that in Equation 7.6 above, the direction of the flux gradient matters. We have appropriately chosen it to point *outward* from the cell to be consistent with Equation 5.4.

If an arbitrary region within a core swells uniformly and contains multiple finite difference cells, these Γ_i surface terms will cancel one another out (in pairs) on all interior cell surfaces - only cell surfaces that form part of the boundary of the whole region will have non-zero surface terms. For example, in Figure 7.1, $\phi_{s1}^{\dagger}\nabla\phi_1$ (the surface term on the right surface of the central cell) will be equal and opposite to the surface term on the left surface of neighboring cell 1. Thus, if both the central cell and cell 1 swell by the same factor in the same direction, the portion of Γ_x evaluated on surface 1 will be zero.

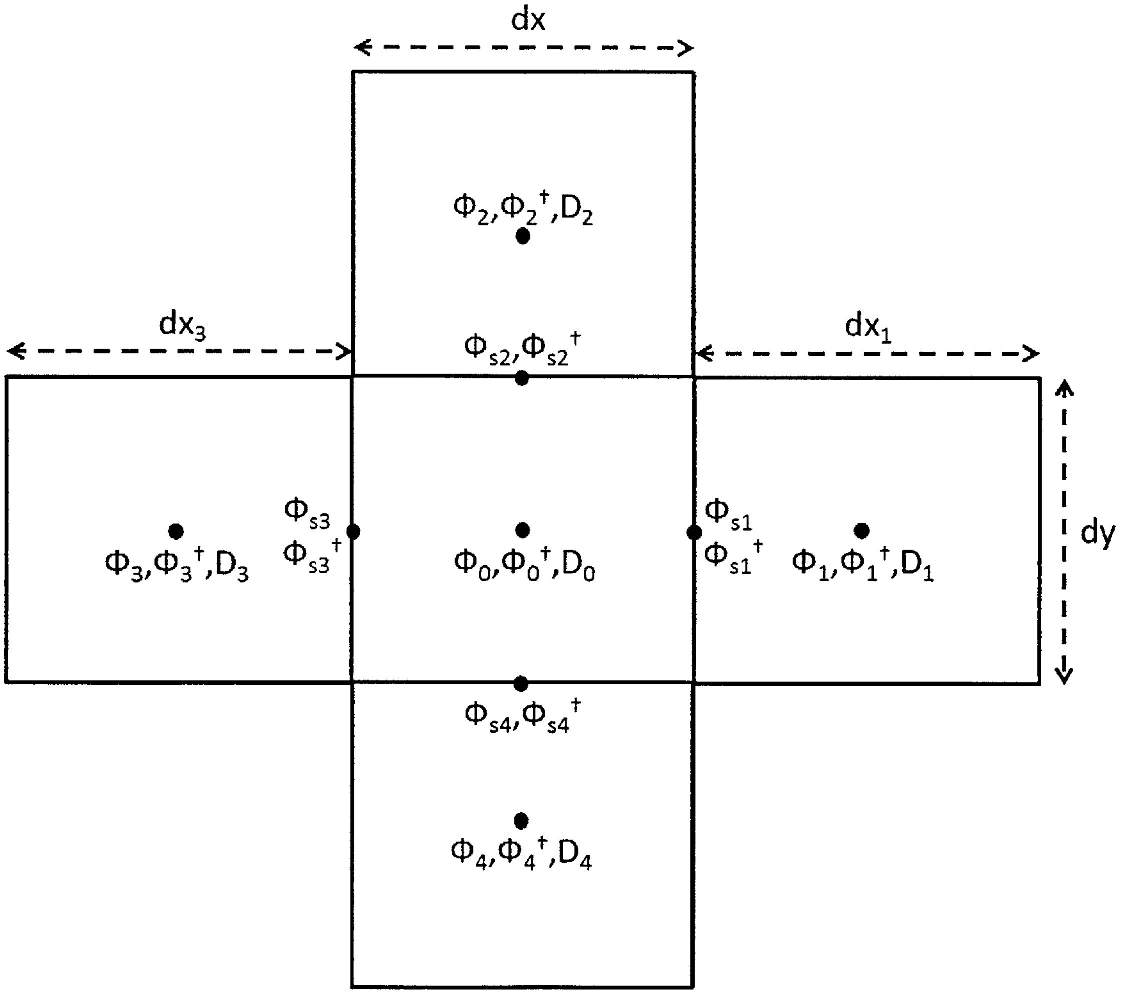
dx
$\Phi_2,\Phi_2^\dagger,D_2$
dx_3
$\Phi_{s2},\Phi_{s2}^\dagger$
dx_1
Φ_{s3}
$\Phi_{s3}^\dagger$
Φ_{s1}
$\Phi_{s1}^\dagger$
dy
$\Phi_3,\Phi_3^\dagger,D_3$
$\Phi_0,\Phi_0^\dagger,D_0$
$\Phi_1,\Phi_1^\dagger,D_1$
$\Phi_{s4},\Phi_{s4}^\dagger$
$\Phi_4,\Phi_4^\dagger,D_4$

Figure 7.1: "Virtual density" implementation in Cartesian geometry. Each cell has volume-averaged real and adjoint fluxes ϕ and $\phi^\dagger$ as well as a diffusion coefficient D. Each interface between cells has real and adjoint surface fluxes ϕ_s and $\phi_s^\dagger$.

7.3.2 Triangular-Z or Hexagonal-Z Coordinates

We can also lay out the difference equations for triangular-z geometry. This is actually equivalent to hexagonal-z geometry, because each hexagonal cell is usually subdivided into six triangular cells in finite difference meshes. Again, we will not bother with the fission denominator or with the numerator of S, which are trivial. The z component in triangular-z geometry is identical to the Cartesian case, so we will derive the equations for the r component in an array of 2-D equilateral triangular cells as shown in Figure 7.2. The surface fluxes are

$$\phi_{s1} = \frac{D_0\phi_0 + D_1\phi_1}{D_0 + D_1} \tag{7.7}$$

$$\phi_{s2} = \frac{D_0\phi_0 + D_2\phi_2}{D_0 + D_2} \tag{7.8}$$

$$\phi_{s3} = \frac{D_0\phi_0 + D_3\phi_3}{D_0 + D_3} \tag{7.9}$$

Note that ds cancels out, because it must be the same in every cell. If *half* the distance between neighboring triangle centroids is $\sqrt{3}ds/6$, then the three outward-directed flux gradients are

$$\nabla\phi_1 = \frac{\phi_{s1} - \phi_0}{\sqrt{3}ds/6} \tag{7.10}$$

$$\nabla\phi_2 = \frac{\phi_{s2} - \phi_0}{\sqrt{3}ds/6} \tag{7.11}$$

$$\nabla\phi_3 = \frac{\phi_{s3} - \phi_0}{\sqrt{3}ds/6} \tag{7.12}$$

If the triangle area is $\sqrt{3}(ds)^2/4$, then the numerator of L_r for one energy group is

$$\left\langle \phi^\dagger \middle| \hat{L}_r\phi \right\rangle = \frac{2}{3}\left(\frac{\sqrt{3}}{4}(ds)^2\right)\left(\nabla\phi_1^\dagger\nabla\phi_1 + \nabla\phi_2^\dagger\nabla\phi_2 + \nabla\phi_3^\dagger\nabla\phi_3\right)D_0 \tag{7.13}$$

The factor of 2/3 is really the product of two factors: (1) a factor of 1/3 for averaging the three flux gradient products throughout the cell and (2) a factor of 2 to account for the fact that $L_r = L_x + L_y$ is *two* dimensions.

The numerator of Γ_r for one energy group is

$$\left\langle \phi^\dagger \middle| \hat{\Gamma}_r\phi \right\rangle = ds\left(\phi_{s1}^\dagger\nabla\phi_1 + \phi_{s2}^\dagger\nabla\phi_2 + \phi_{s3}^\dagger\nabla\phi_3\right)D_0 \tag{7.14}$$

Again, the fact that each flux gradient points outward from the cell is crucial.

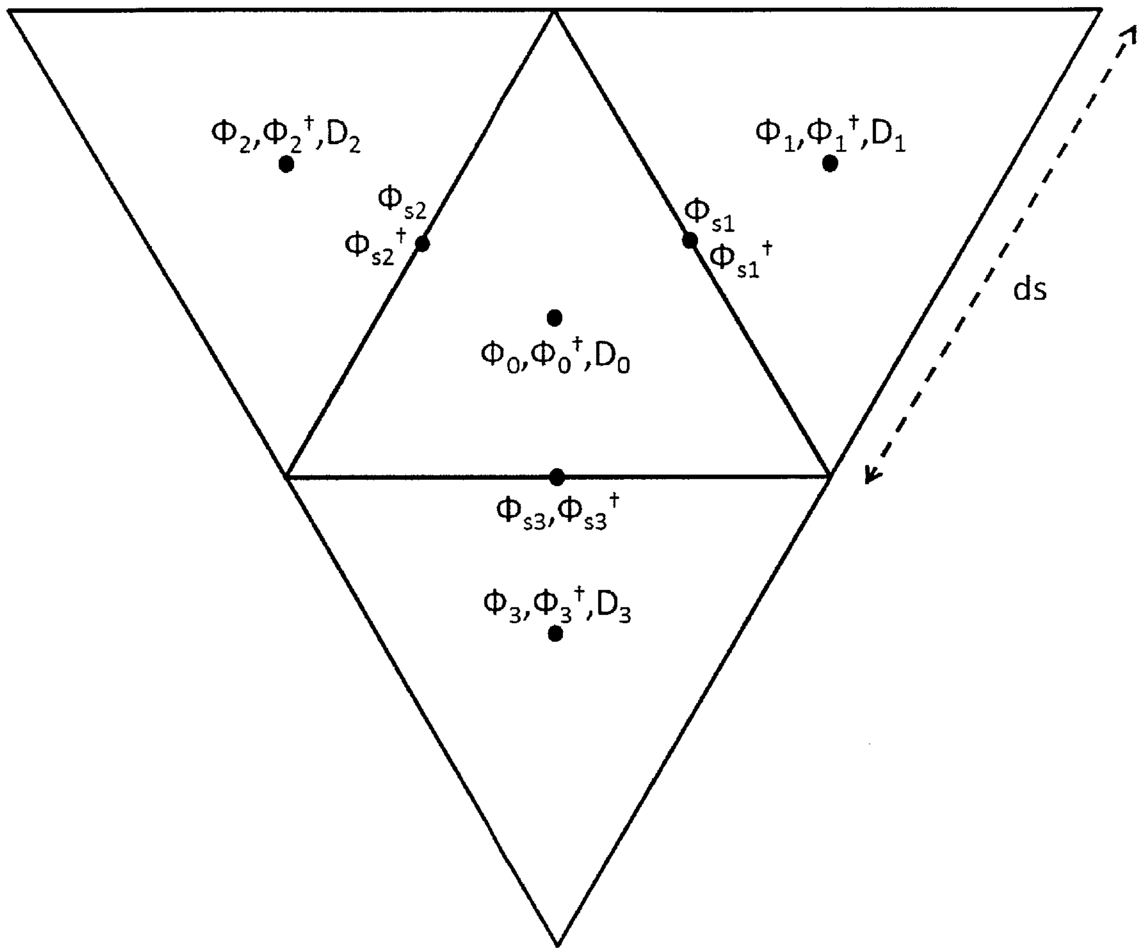

Figure 7.2: "Virtual density" implementation in triangular geometry. This is equivalent to hexagonal geometry, because finite difference solutions always subdivide each hexagonal cell into six triangular cells. Each cell has volume-averaged real and adjoint fluxes ϕ and $\phi^\dagger$ as well as a diffusion coefficient D. Each interface between cells has real and adjoint surface fluxes ϕ_s and $\phi_s^\dagger$.

Table 7.1: FFTF and Jōyō Finite Difference Diffusion Benchmarks

Benchmark	k_{eff}	deviation (pcm) from DIF3D
Jōyō fine axial DIF3D	0.974222662	
Jōyō fine axial MaPS	0.974266044	4.57
Jōyō coarse axial MaPS	0.973187327	-109.20
FFTF fine axial DIF3D	0.982859575	
FFTF coarse axial MaPS	0.980185174	-277.60

7.4 The FFTF and Jōyō Finite Difference Diffusion Benchmarks

In order to validate the anisotropic non-uniform "virtual density" formalism for realistic cores, we will employ two full-core 3-D benchmarks: FFTF and Jōyō. See Appendices J and K for detailed benchmark descriptions and core maps.

We build these two benchmarks in coarse mesh finite difference diffusion via our MATLAB-PETSC-SLEPc (MaPS) code, which we describe in Appendix A. In each benchmark, we fill in the outermost hexagonal assembly ring (with additional shield assemblies) for convenience. This alters k_{eff} by less than 20 pcm. We divide each hexagonal assembly into 6 triangles and generate 33-group cross-sections for each benchmark via MC**2 [187].

Furthermore, we have two different versions of each benchmark: a fine axial mesh and a coarse axial mesh. The fine axial mesh versions are obtained directly from DIF3D models within the Advanced Reactor Modeling Interface (ARMI) [211]. We also coarsen each axial mesh (by a factor of 4 for FFTF and a factor of 3 for Jōyō) in order to achieve greater efficiency for a multitude of eigenvalue calculations in MaPS. Coarsening the mesh involves simply homogenizing 3 or 4 axial mesh cells to conserve average cross-sections.

With outer assembly rings filled in, both FFTF and Jōyō have 11 full rings (331 assemblies). Both also have 6 triangles per assembly and 33 energy groups. In the FFTF model, the fine and coarse axial meshes have 111 and 28 cells, respectively. In the Jōyō model, the fine and coarse axial meshes have 66 and 22 cells, respectively.

Table 7.1 shows k_{eff} values for the fine and coarse versions of FFTF and Jōyō in DIF3D and MaPS. In the Jōyō fine mesh case, MaPS agrees with DIF3D to less than 5 pcm. Coarsening the axial meshes results in losses of approximately 115 pcm for Jōyō and 280 pcm for FFTF. We will use only the coarse mesh versions of FFTF and Jōyō from this point onward.

Table 7.2: Whole-Core Uniform Anisotropic "Virtual Density" Spectral and Leakage Quantities in the FFTF

FFTF	pcm/%
S	239.500929
L_r	148.90210
L_z	90.587839
$S - L_r - L_z$	0.011078

Table 7.3: Whole-Core Uniform Anisotropic "Virtual Density" Spectral and Leakage Quantities in the Jōyō

Jōyō	pcm/%
S	349.349521
L_r	223.980760
L_z	125.492411
$S - L_r - L_z$	-0.123650

7.5 Numeric Results for Whole-Core Uniform Swelling

First let us validate the anisotropic whole-core uniform "virtual density" formalism laid out in Section 4.4. We can calculate the quantities S, L_r, and L_z in coarse mesh finite difference as shown in Section 7.3 for the FFTF and Jōyō benchmarks. Tables 7.2 and 7.2 show the quantities in units of "pcm per percent", where the "percent" is a one percent change in a linear dimension or a one percent density change. Note that we carefully defined the quantities S, L_r, and L_z in Sections 4.4.1, 4.4.2, 5.3, and 5.3.2 so that they will always be positive when integrated through a whole core (although they could be negative when integrated over an interior zone).

Note that the equality $S = L_r + L_z$ holds true even in this coarse mesh problem. As shown in Tables 7.2 and 7.3, the quantity $S - L_r - L_z$ is negligible in size relative to the quantities S, L_r, and L_z separately. It should be zero in theory, and we find that it is 0.0046% of S. We deem this "zero enough".

$$\frac{S - L_r - L_z}{S} = 0.0046\% \tag{7.15}$$

In Jōyō, the same quantity is

$$\frac{S - L_r - L_z}{S} = -0.0354\% \tag{7.16}$$

This is also "zero enough" for our purposes.

Now we can easily compute the whole-core uniform reactivity coefficients for radial swelling, axial swelling, and isotropic swelling. We can also compute the reactivity coefficient for whole-core uniform density reduction. We simply use the whole-core uniform formalism for hexagonal-z geometry laid out in Section 4.4.6. Table 7.4 shows the coefficients for the FFTF, while Table 7.5 shows the same coefficients for Jōyō.

We compare these coefficients directly to a reference case, which is a direct eigenvalue solution in MaPS. For example, in order to compute a reference reactivity coefficient $d\rho/df$ for a swelling magnitude of 1% (or $f = 1.01$), we simply perform two full eigenvalue calculations and take their difference:

$$\left[\frac{d\rho}{df}\right]_{f=1} \approx \frac{1/k_{f=1} - 1/k_{f=1.01}}{0.01} \tag{7.17}$$

Unless stated otherwise, we will use swelling/expansion fractions of $f = 1.01$, 1.001, or 1.0001 for reference cases throughout this chapter. Constructing a reference case for uniform axial swelling requires increasing all axial mesh heights by f while reducing all material densities by f. Constructing a reference case for uniform radial swelling requires increasing all assembly hexagonal "flat-to-flat" pitches by f while reducing all material densities by f^2. Constructing a reference case for uniform isotropic swelling requires increasing both axial mesh heights and assembly pitches by f while reducing all material densities by f^3.

As shown in Tables 7.4 and 7.5, "virtual density" perturbation theory (VirDenT) produces reactivity coefficients that agree with reference values to within 0.1%. The PETSc-SLEPc Arnoldi eigenvalue convergence is 10^{-10}, so all the decimal places are warranted. Evidently, VirDenT is quite precise even in coarse mesh hexagonal-z geometry for a highly heterogeneous full-core benchmark. Furthermore, these numbers verify the simple fact that an isotropic swelling coefficient is (1) exactly twice a density reduction coefficient and (2) the exact sum of axial and radial swelling coefficients.

Note that any realistic axial or radial swelling scenario would require adjusting the coolant density by a different factor than the fuel and structure materials. This can be easily accomplished by performing a traditional coolant density perturbation *after* the uniform swelling computation. Since knowing the factor by which to adjust the coolant density depends upon the heterogeneous pin and duct detail of each assembly, we will neglect the coolant density effect in this chapter - it is trivial, and it adds complexity that is both unnecessary and distracting.

Table 7.4: Uniform Anisotropic "Virtual Density" Reactivity Coefficients for FFTF

FFTF	k_{eff} ref.	pcm/% ref.	pcm/% VirDenT	error (%)
unperturbed	0.980185174			
density reduce	0.979725102	-479.085954	-478.9978652	0.018%
isotropic swell	0.979264821	-958.8400146	-957.9957304	0.088%
axial swell	0.979899114	-297.8294504	-297.8080172	0.007%
radial swell	0.979550936	-660.5673529	-660.1907084	0.057%

Table 7.5: Uniform Anisotropic "Virtual Density" Reactivity Coefficients for Jōyō

Jōyō	k_{eff} ref.	pcm/% ref.	pcm/% VirDenT	error (%)
unperturbed	0.973187327			
density reduce	0.972526291	-698.4371254	-699.0095938	0.082%
isotropic swell	0.971865128	-1397.958826	-1398.019188	0.004%
axial swell	0.972763834	-447.3447816	-447.6509682	0.068%
radial swell	0.97228858	-949.8298723	-950.1353052	0.032%

Table 7.6: Non-Uniform Anisotropic "Virtual Density" Spectral and Leakage Quantities within an Internal Zone of Jōyō

Jōyō interior zone	pcm/%
S	81.1912694381
L_r	23.4637520600
L_z	8.0590066902
Γ_r	-24.4585146833
Γ_z	-25.2099959654
$S - L_r - L_z + \Gamma_r + \Gamma_z$	3.9×10^{-8}

7.6 Numeric Self-Consistency of the Non-Uniform Anisotropic "Virtual Density" Formalism in Triangular-Z Geometry

Now let us turn to the *non-uniform* anisotropic "virtual density" formalism, which is our original invention. We select an arbitrary internal zone within Jōyō as illustrated in Figure 7.3. This zone consists of an axial slice (4 mesh cells thick) of 24 contiguous assemblies.

We can now proceed to evaluate S, L_r, L_z, Γ_r, and Γ_z for this arbitrarily-selected internal zone. We compute the Γ values as described in Section 7.3. The unperturbed real and adjoint flux distributions for Jōyō are pre-computed with 10^{-16} eigenvalue convergence.

Table 7.6 shows the five spectral-leakage shorthand quantities. It also demonstrates that the non-uniform "virtual density" equality $S = L_r + L_z - \Gamma_r - \Gamma_z$ holds true such that

$$\frac{S - L_r - L_z + \Gamma_r + \Gamma_z}{S} = 4.8 \times 10^{-8}\% \tag{7.18}$$

This is tighter agreement than what we saw for the whole-core uniform cases in Equations 7.16 and 7.17. The reason for this may be due in part to tighter eigenvalue convergence for the unperturbed case, but it may also be due in part to a lack of vacuum boundary leakage in this internal zone.

Table 7.7 shows the various expressions for axial swelling shown in Equations 5.41 - 5.47 in Section 5.7. This verifies that the sets of equivalent expressions shown throughout Chapter 5 are in fact equivalent. Here the equivalent expressions differ by no more than 4×10^{-7} %. Even tighter eigenvalue convergence for the unperturbed case would make these equivalent expressions agree even *more* closely. As the unperturbed eigenvalue convergence approaches zero, the differences among these equivalent expressions also approach zero.

Note that computing an "axial swelling coefficient" for any arbitrary interior zone is perfectly reasonable. Figure 7.4 illustrates this. Axial segments of a single assembly can swell by different factors, causing the assembly top to "stick out" above the core top. This

Table 7.7: "Virtual Density" Equivalent Expressions for Axial Swelling of an Interior Zone

Jōyō interior zone	pcm/%
$-4\mathrm{S} + 4(\mathrm{L}_z - \Gamma_z) + 2L_r - 3\Gamma_r$	-71.3860189602
$-3\mathrm{S} + 3(\mathrm{L}_z - \Gamma_z) + L_r - 2\Gamma_r$	-71.3860189209
$-2\mathrm{S} + 2(\mathrm{L}_z - \Gamma_z) - \Gamma_r$	-71.3860188817
$-\mathrm{S} + (\mathrm{L}_z - \Gamma_z) - L_r$	-71.3860188425
$-2\mathrm{L}_r + \Gamma_r$	-71.3860188033
$\mathrm{S} - (\mathrm{L}_z - \Gamma_z) - 3L_r + 2\Gamma_r$	-71.3860187641
$2\mathrm{S} - 2(\mathrm{L}_z - \Gamma_z) - 4L_r + 3\Gamma_r$	-71.3860187249

is allowed geometrically, and it is consistent with what would actually occur physically - assemblies (and fuel pins) swell more near the core center, where power and temperature are highest.

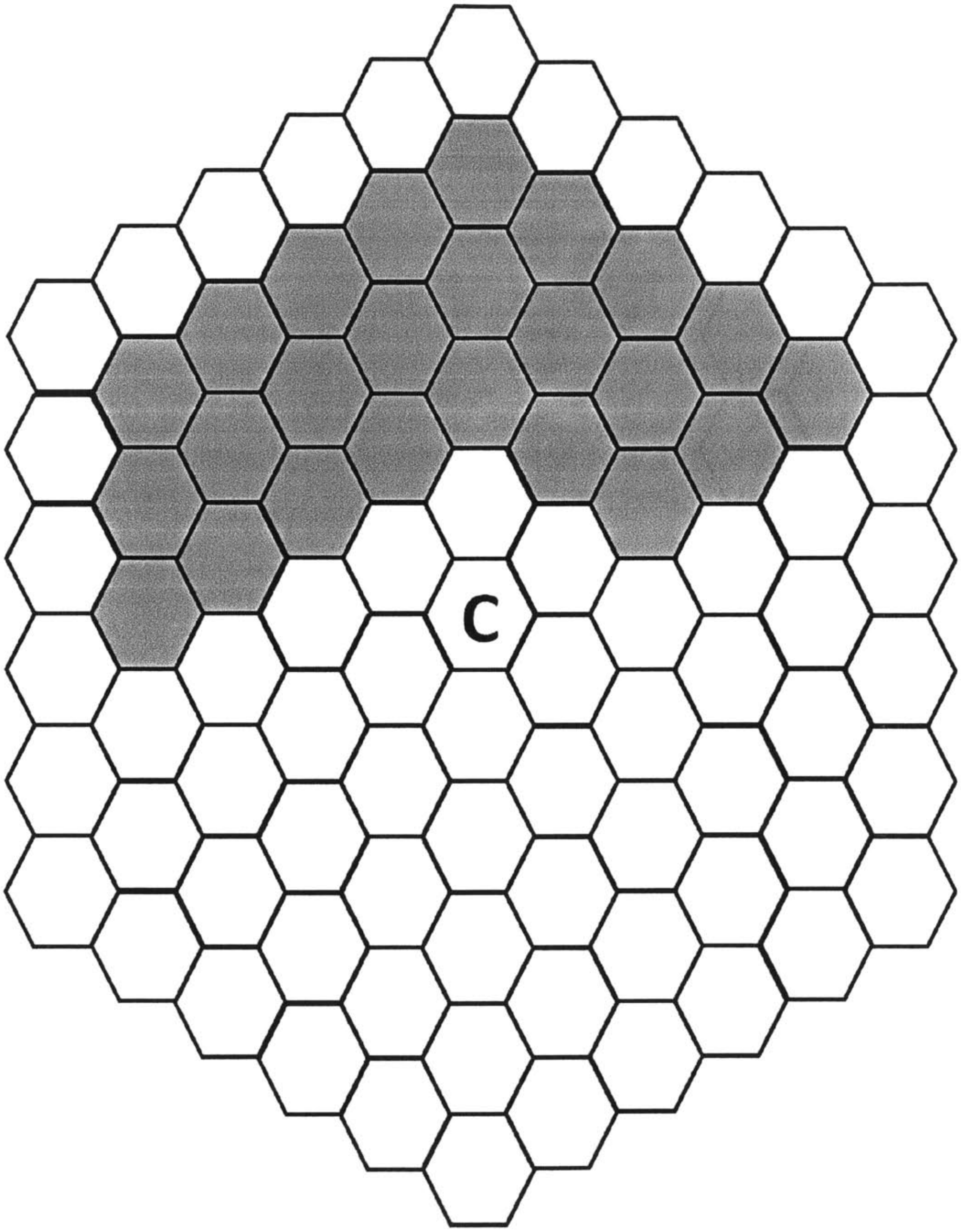

Figure 7.3: An arbitrary internal zone within Jōyō marked by the salmon/coral color. This is a contiguous cluster of 24 assemblies in rings 3, 4, and 5. The "C" marks the central core assembly. Within each of these 24 assemblies, we select axial zones 11-14.

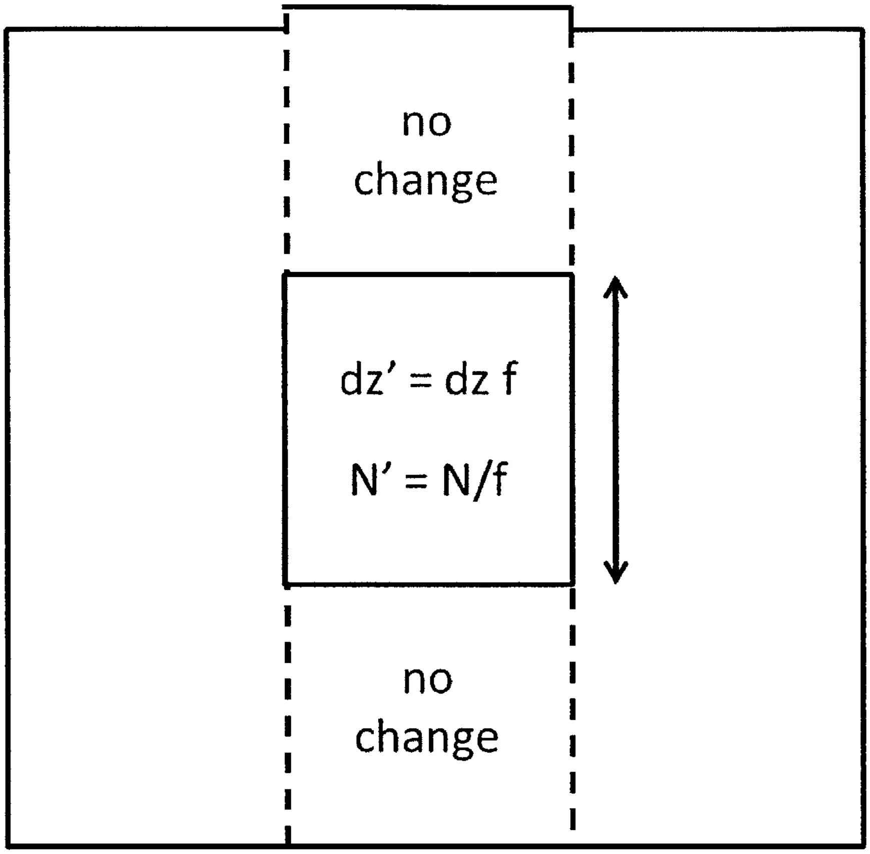

Figure 7.4: Axial swelling of an interior core zone. Suppose a certain assembly (bounded by the dashed line) extends through the entire core height. Now suppose that a certain axial segment of that assembly swells axially. Within that segment, the dimensions in the axial (z) direction swell by a factor f, while the material densities decrease by that same factor f in order to conserve mass. Other segments of this assembly (above and below this swelling segment) are left unchanged. However, all segments *above* the swelling segment shift upward such that the assembly now "sticks out" at the core top. Evaluating the reactivity coefficient $d\rho/df$ requires integrating the spectral-leakage quantities throughout *only* the swelling segment.

7.7 Numeric Results for 2-D Cartesian Assembly Distortions

In Chapter 6, we validated the "virtual reference" against Monte Carlo. Moving forward, we will validate "virtual density" perturbation theory against "virtual references" and, when feasible, "actual references". When a reference case is "virtual", we will label it in data tables as "virtual ref.". When a reference case is *not* virtual, we will label it as simply "ref.".

Now let us return to the 6x6 square assembly array first illustrated in Figure 6.7. In this section, we will reduce it to a 2-D problem. See Figure 7.5 for an illustration. Each assembly has side length b and area b^2. Each interstitial sodium region has width a. We can divide a and b into as many or as few mesh cells as we wish. The accuracy of the "virtual density" theory is not sensitive to mesh fineness, as we demonstrated in Section 8.6 for 1-D cases.

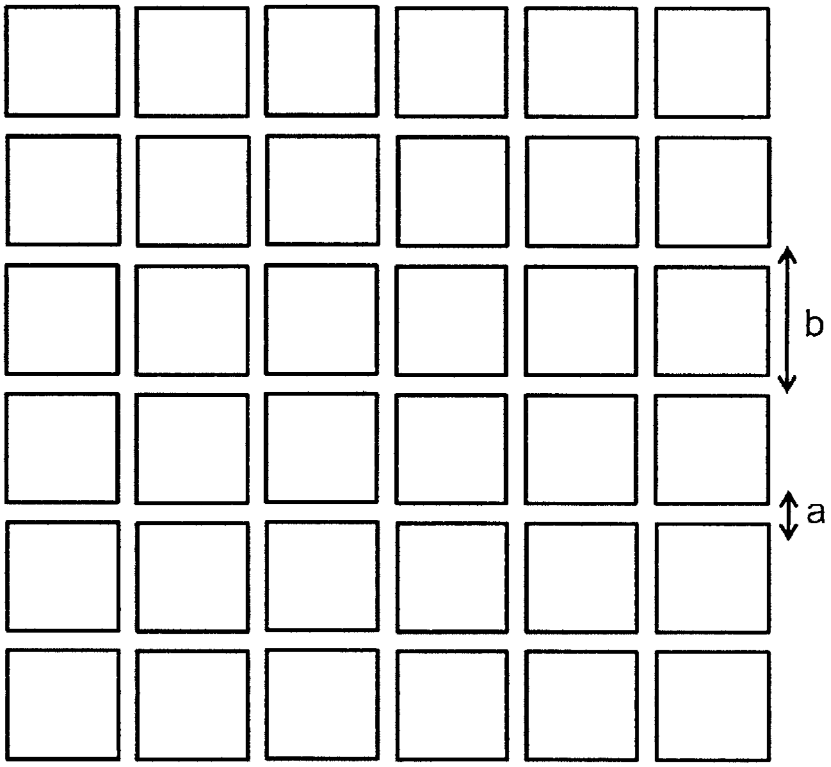

Figure 7.5: A 6x6 array of square enriched uranium assemblies immersed in a sodium pool. Each assembly has edge length b and is separated from its neighbors by an interstitial width a filled with sodium. A depleted uranium reflector surrounds the entire reactor. We can vary both a and b in terms of both their absolute size (cm) and the number of mesh cells into which we divide them.

7.7.1 Radial Assembly Flowering

First let us demonstrate "virtual density" for a case of uniform assembly "flowering" in the 2-D Cartesian geometry shown in Figure 7.5. Let a be 1 cm and 1 mesh cell thick, and let b be 11 cm and 2 mesh cells thick. Let us divide up these 36 assemblies into 3 square "rings". The inner ring contains the 4 central assemblies. The outer ring contains the 20 outer assemblies. The middle ring contains the remaining 12 assemblies.

First suppose that all assembly rings move radially outward such that a increases while b remains fixed. We can apply "virtual density" to this scenario by anisotropically expanding all the interstitial sodium regions such that $a' = fa$. We also consider the case in which all assemblies move outward from the core in the x direction (not radially). Applying "virtual density" to this second case requires anisotropically expanding only the vertically-directed interstitial zones. Since each of these interstitial sodium zones slices through the entire core, we need not construct a "virtual reference" case - a true reference case will suffice. See Figure 7.6, which shows that "virtual density" predicts the reference case reactivity coefficient to less than 0.1%.

Now consider moving each assembly ring separately. Figures 7.7, 7.8, and 7.9 compare "virtual density" to reference cases for the inner, middle, and outer rings, respectively. The non-uniform anisotropic "virtual density" equations always predict the reference case reactivity coefficients to within 0.1%.

Throughout all cases involving anisotropic expansion of Cartesian interior zones, we prefer to employ Equation 5.32.

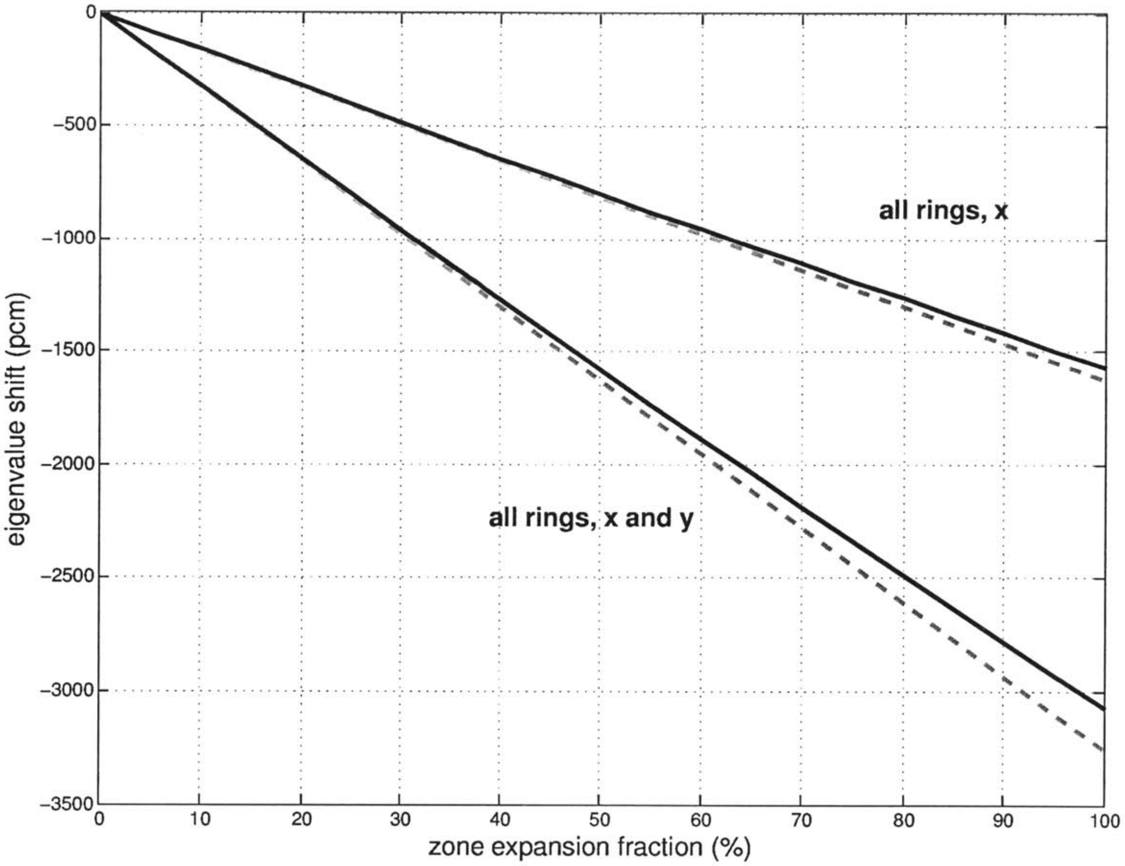

Figure 7.6: The reactivity due to all 36 assemblies moving outward from the core center. The "zone expansion fraction" is the factor by which all the interstitial sodium regions widen so that $a' = fa$, where a 10% "zone expansion fraction" corresponds to $f = 1.1$. Here we consider outward assembly movement in only the x direction *and* in both the x and y directions. The solid black line represents the reference case, while the dashed red line represents the linear reactivity coefficient computed via "virtual density". The reactivity coefficients agree to within 0.1%.

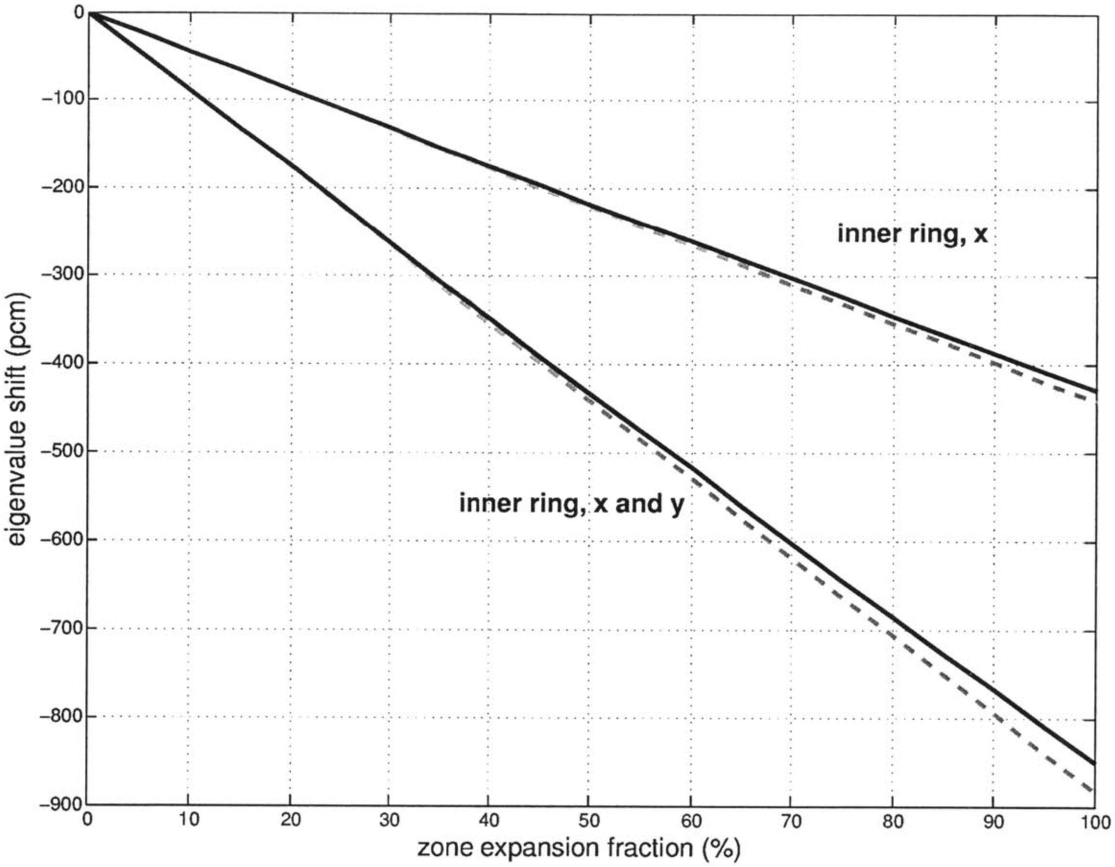

Figure 7.7: The reactivity due to the inner ring of 4 assemblies moving outward from the core center. Here we consider outward assembly movement in only the x direction *and* in both the x and y directions. The solid black line represents the reference case, while the dashed red line represents the linear reactivity coefficient computed via "virtual density". The reactivity coefficients agree to within 0.1%.

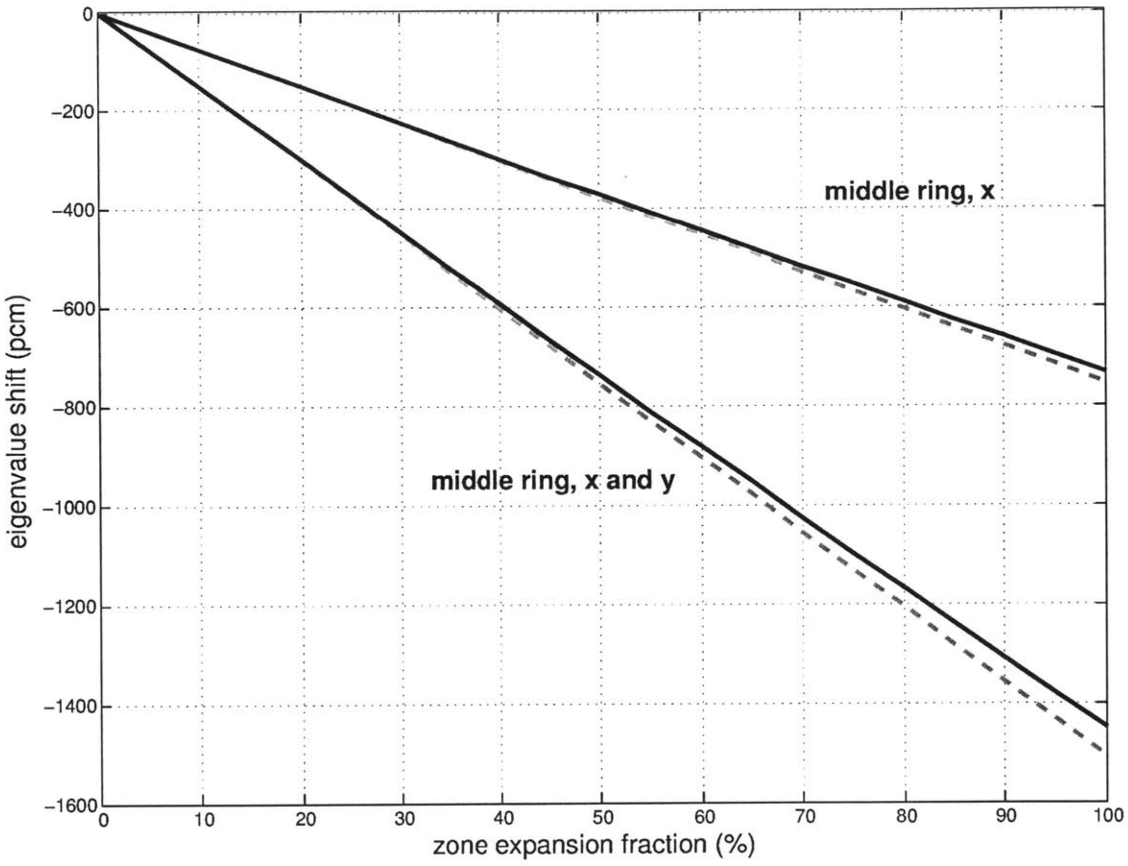

Figure 7.8: The reactivity due to the middle ring of 12 assemblies moving outward from the core center. Here we consider outward assembly movement in only the x direction *and* in both the x and y directions. The solid black line represents the reference case, while the dashed red line represents the linear reactivity coefficient computed via "virtual density". The reactivity coefficients agree to within 0.1%.

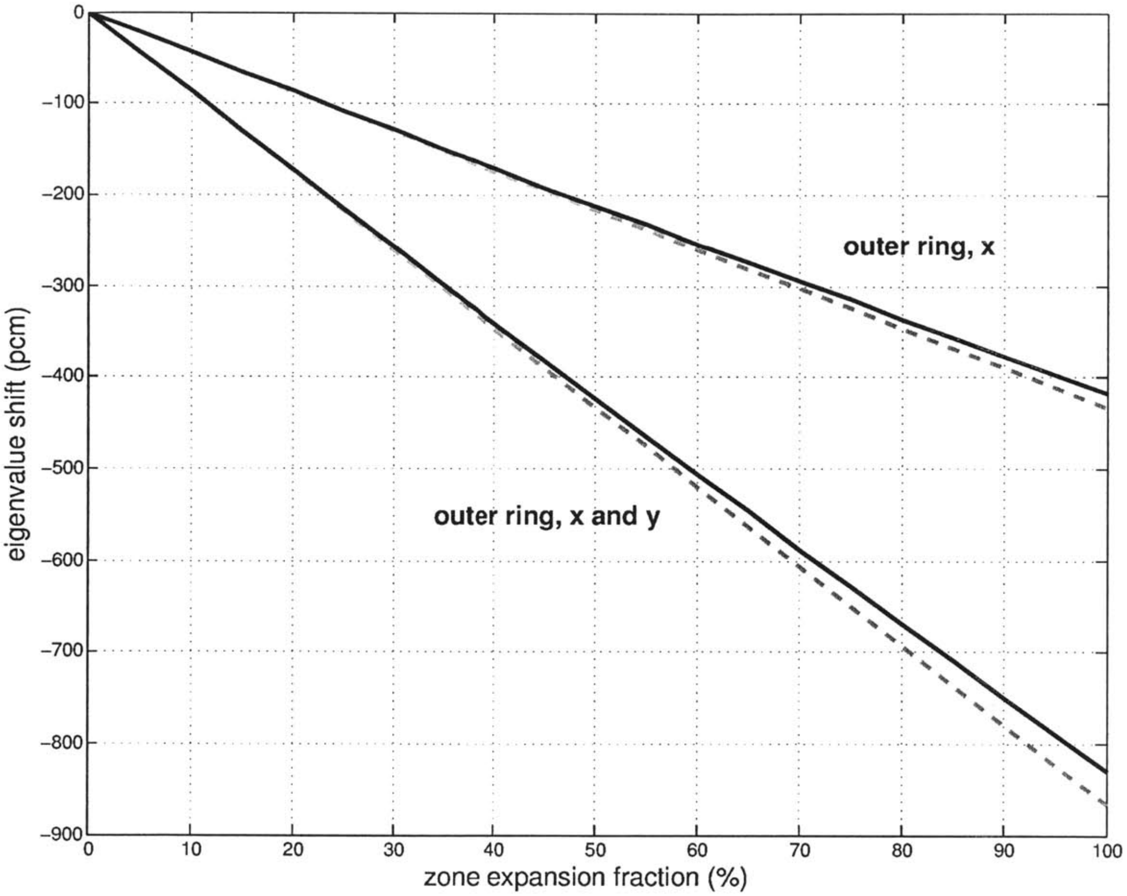

Figure 7.9: The reactivity due to the outer ring of 20 assemblies moving outward from the core center. Here we consider outward assembly movement in only the x direction *and* in both the x and y directions. The solid black line represents the reference case, while the dashed red line represents the linear reactivity coefficient computed via "virtual density". The reactivity coefficients agree to within 0.1%.

7.7.2 Assembly Row Motion

Now consider motion of a whole row of 6 assemblies in the same 2-D geometry shown in Figure 7.5. Figure 7.10 illustrates this motion. Consider that each vertical row is bounded by two vertically-directed interstitial sodium zones - one on each side. If the assembly row shifts to the left, the right-side sodium zone expands anisotropically, while the left-side sodium zone contracts anisotropically. So we can simply apply Equation 5.32 to each zone separately.

Figure 7.11 compares the "virtual density" reactivity coefficient prediction to reference cases *and* examines its sensitivity to geometry parameters. We vary the "aspect ratio", which is a/b, while keeping b fixed. When the interstitial sodium zones are relatively thin, the reference case reactivities are (1) smaller and (2) more linear. Remember that reactivity linearity scales more closely with reactivity magnitude than with perturbation magnitude. Due to (2), "virtual density" can more accurately predict relatively large assembly movements when the interstitial sodium zones are thinner relative to the assembly size and the neutron mean free path. Figure 7.11 shows three row shifting scenarios:

- A single row of 6 assemblies shifting in a 6x6 assembly array with aspect ratio $a/b = 0.1$.
- A single row of 6 assemblies shifting in a 6x6 assembly array with aspect ratio $a/b = 0.05$.
- A single row of 6 assemblies shifting in a 6x6 assembly array with aspect ratio $a/b = 0.0357$.
- A single row of 10 assemblies shifting in a 10x10 assembly array with aspect ratio $a/b = 0.0357$.

We chose $a/b = 0.0357$ to be approximately consistent with the FFTF benchmark, although the FFTF has hexagonal assemblies.

As expected, non-uniform "virtual density" predicts the reference case reactivity *coefficients* to within 0.1%. However, the reference case reactivities exhibit appreciable curvature in the case of $a/b = 0.1$ in the 6x6 assembly array. Thus, "virtual density" may underpredict the reactivity *magnitudes* by as much as 14% when an assembly row shifts enough to touch a neighboring row. Fortunately, very few reactors have an aspect ratio this large, and very few reactors have so few assemblies. As shown in Figures 7.11 and 7.12, we find that (1) decreasing a/b and (2) increasing the number of assemblies both result in more linear reactivities that "virtual density" can better predict. In the case of $a/b = 0.0357$ in the 10x10 assembly array, non-uniform "virtual density" predicts the reactivity to within 5% when assembly rows shift enough to touch one another. When the assembly array is 18x18, the "virtual density" reactivity magnitude is accurate to within 3%. Since most realistic cores have many hundreds of assemblies, we can wager that the "virtual density" reactivity magnitudes will be accurate to within a few percent for even the largest distortions.

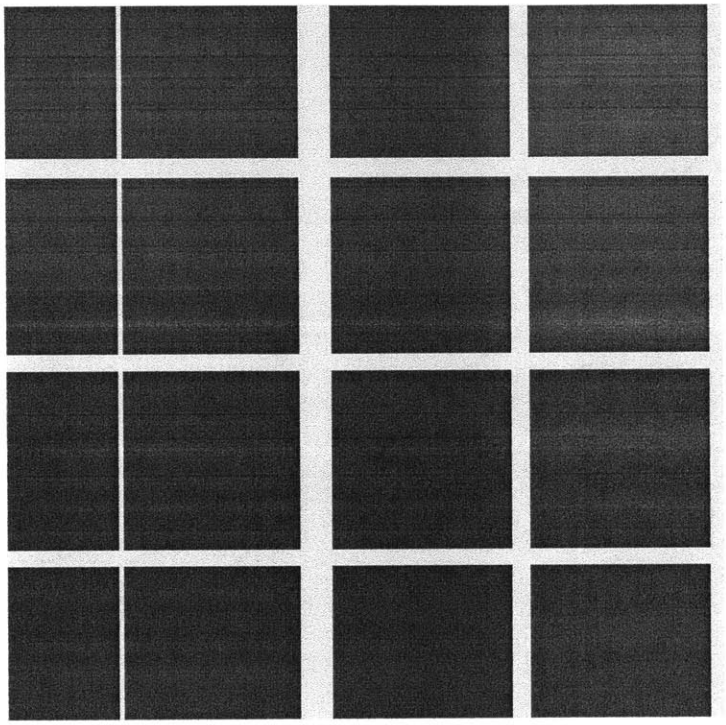

Figure 7.10: An illustration of a whole row of assemblies shifting to the left within a 6x6 assembly core. The vertically-directed interstitial zone to the right of this row widens, while the interstitial zone to the left of it contracts.

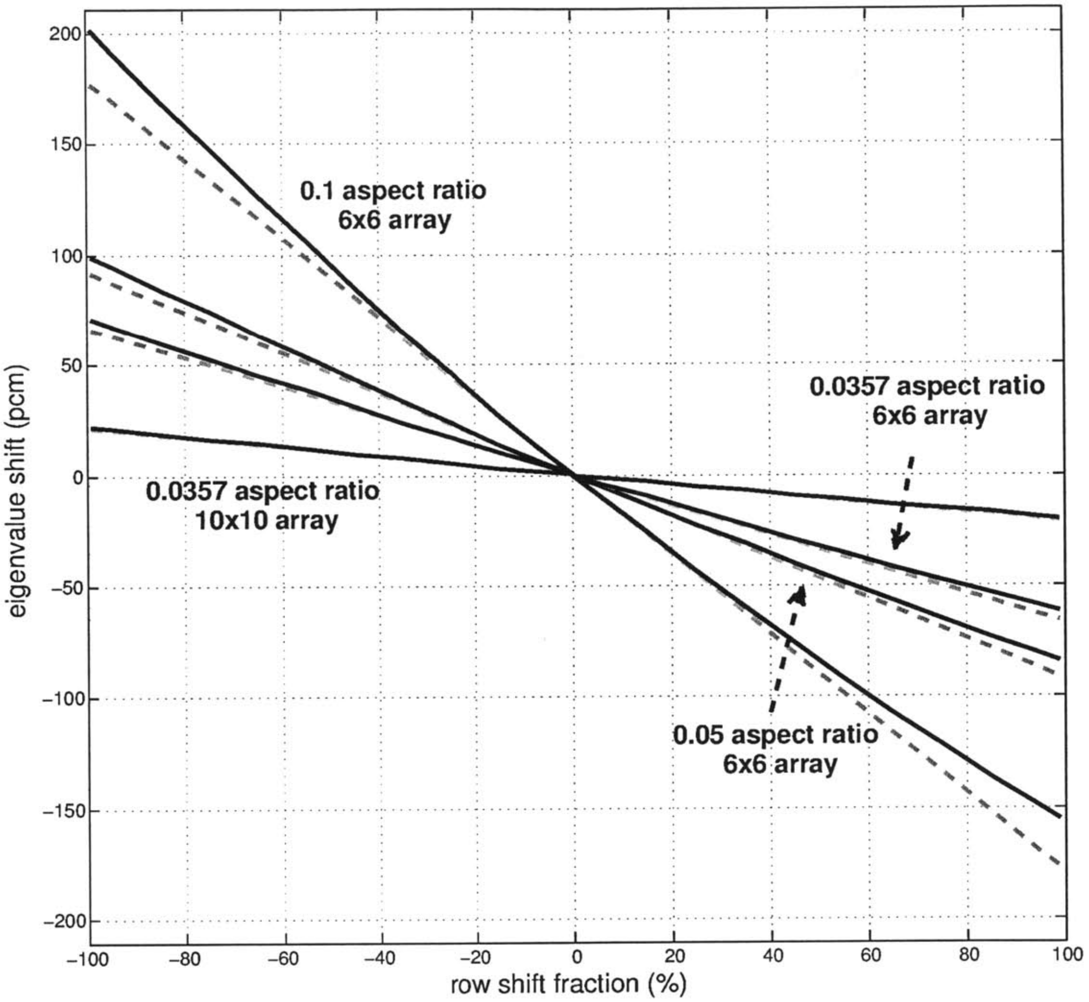

Figure 7.11: The reactivity due to a single row of assemblies shifting to the right and left. Here the "row shift fraction" is the fraction by which the assembly row shifts over to touch the neighboring assembly row. When this fraction is positive, the assembly row shifts to the right. The solid black lines represent the reference cases, while the dashed red lines represent the linear reactivity coefficients computed via "virtual density". The reactivity coefficients agree to within 0.1%. Here the "aspect ratio" is a/b.

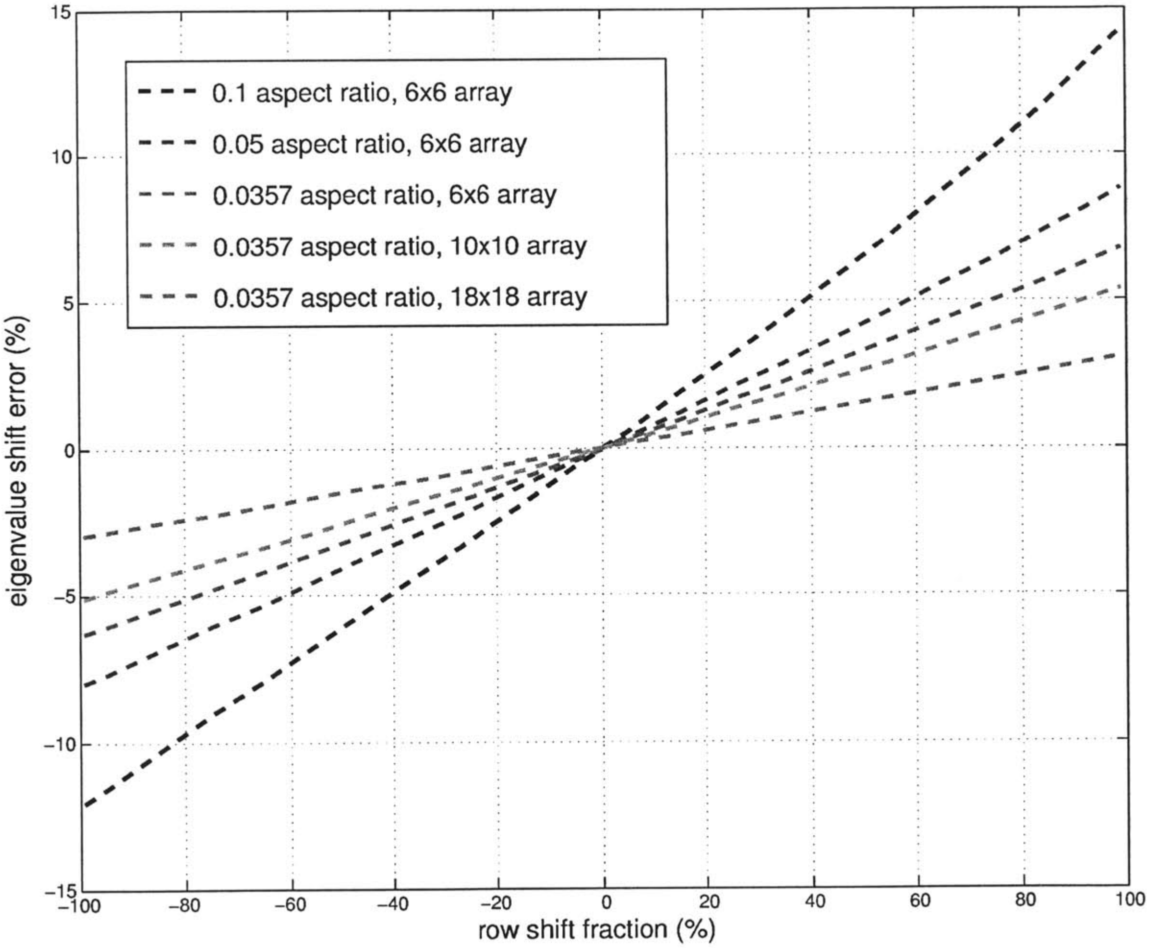

Figure 7.12: The "virtual density" error magnitudes for the four cases in Figure 7.11 plus one additional case: an aspect ratio $a/b = 0.0357$ in an 18x18 assembly array.

7.7.3 Arbitrary Motion of a Single Assembly

Now let us consider arbitrary motion of a *single* assembly in the same 2-D Cartesian geometry. This time we will stick to our original 6x6 array. See Figure 7.13 for an illustration of this scenario. We divide a into 4 mesh cells and b into 5 mesh cells. Note that we could just as easily choose a very coarse mesh, because mesh resolution does not strongly affect the accuracy of "virtual density" perturbation theory.

Unlike the cases of assembly row shifting and assembly ring flowering, we can *not* construct a true reference case for this scenario. The expanding and contracting coolant zones on either side of the moving assembly do *not* constitute a through-core slice, and so any reference case will have misaligned mesh cells. Thus, we must employ the "virtual mesh" method to validate "virtual density" perturbation theory for this case.

Table 7.8 shows the results. We construct "virtual reference" cases for expansion/contraction fractions of $f = 1.01$, 1.001, 1.0001, and 1.00001. If dx in the expanding coolant region multiplies by a factor of f, then dx in the contracting coolant region multiplies by $2 - f$. Here the MaPS eigenvalue convergence is 10^{-20}. As expected, the "virtual density" reactivity coefficient is exact, and the "virtual reference" reactivity coefficient approaches it as f approaches 1. Why the error increases so quickly with f is not clear, but this should serve as a warning to always select f as close as possible to 1 in any "virtual reference" case. This rapid increase in error with respect to the "virtual reference" does *not* necessary reflect the true error, because the "virtual reference" is only truly valid in the limit of zero perturbation magnitude.

Regardless, "virtual density" perturbation theory clearly predicts the reactivity coefficient very precisely. We can easily expand and contract the two interstitial zones above and below the same assembly to obtain the reactivity coefficient for vertical movement. Once we have separate coefficients for horizontal and vertical movement, we can easily compute the coefficient due to movement in any arbitrary direction. For example, if a given assembly moves a distance r the θ direction ($0 < \theta < 2\pi$), then the reactivity coefficient for that movement is

$$\frac{d\rho}{dr} = \frac{d\rho}{dx}\cos\theta + \frac{d\rho}{dy}\sin\theta \tag{7.19}$$

Here $d\rho/dx$ and $d\rho/dy$ are the reactivity coefficients with respect to movement in the x and y directions, and they are trivial to obtain from $d\rho/df$. Thus, we can now compute reactivity coefficients due to any combination of arbitrary assembly motions. This is the gateway to neutronic seismic analysis!

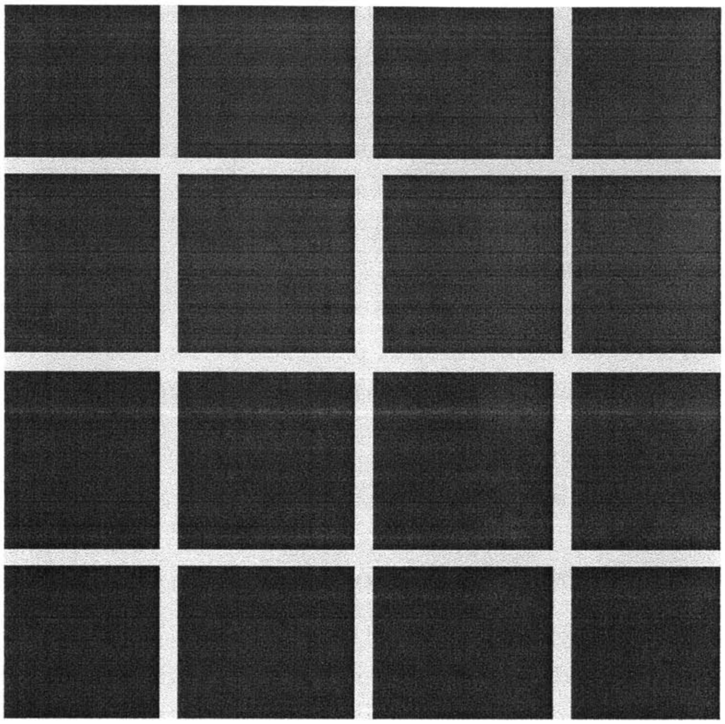

Figure 7.13: An illustration of single assembly motion to the right. The vertically-directed interstitial zone to the right of this row narrows, while the interstitial zone to the left of it widens.

Table 7.8: Shift of a Single Assembly in 2-D Cartesian Geometry

	k_{eff}	pcm/%	error (%)
unperturbed	1.07116770839566		
VirDenT		-0.180820212394	
"virtual reference" 0.001%	1.07116770632100	-0.180813519712	0.003701
"virtual reference" 0.01%	1.07116768765537	-0.180758982227	0.03386
"virtual reference" 0.1%	1.07116750162353	-0.180209275724	0.3379
"virtual reference" 1.0%	1.07116570374087	-0.174713098344	3.377

Table 7.9: Radial Assembly Flowering with Axial Dependence in 3-D Cartesian Geometry

	k_{eff} virtual ref.	pcm/% virtual ref.	pcm/% VirDenT	error (%)
unperturbed	1.025845055			
0.1% flat	1.025795779	-46.82647027	-46.83187358	-0.011539008
0.1% cosine	1.025803335	-39.6453831	-39.63532896	0.025360179
0.1% linear	1.025819163	-24.60432957	-24.58673361	0.071515706
0.1% quadratic	1.025829865	-14.43470344	-14.4209074	0.0955755

7.8 Numeric Results for 3-D Cartesian Assembly Distortions

Now we turn to 3-D Cartesian cases. We will keep using the same basic 6x6 assembly array shown in Figures 6.7 and 6.8, but now it is 3-D again.

7.8.1 3-D Cartesian Assembly Flowering

In Section 7.7.1, we studied various flowering scenarios in the 2-D model. Now we will examine the radial flowering of all three assembly "rings" in 3-D, but now the flowering magnitude has an axial dependence that could be any arbitrary function. We can directly compare the "virtual density" perturbation theory result to a "virtual reference" case for a variety of axial form functions.

If $f(z)$ is the fractional widening of all interstitial sodium zones as a function of axial position $0 < z < z_{\text{top}}$, then we can define various axial flowering profiles:

$$f(z) = \begin{cases} 1 + 0.01 & \text{flat} \\ 1 + 0.01\left(\frac{z}{z_{\text{top}}}\right) & \text{linear} \\ 1 + 0.01\left(\frac{z}{z_{\text{top}}}\right)^2 & \text{quadratic} \\ 1 + 0.01\cos\left(\pi\left(\frac{z}{z_{\text{top}}} - \frac{1}{2}\right)\right) & \text{cosine} \end{cases} \tag{7.20}$$

Here we have scaled $f(z)$ so that its maximum value (on any axial plane) is 1.01, a 1% expansion of the interstitial zones. So when we state a reactivity coefficient as "per percent", this "percent" is the *maximum* swelling percent in the core.

Table 7.9 shows results for each $f(z)$ form. The non-uniform "virtual density" perturbation theory predicts all reactivity coefficients to within 0.1%.

Table 7.10: "Swing" of an Assembly Row in 3-D Cartesian Geometry

	k_{eff}	pcm/%	error (%)
unperturbed	1.006183941228		
VirDenT		-0.989498173787	
"virtual reference" 0.1%	1.006183841052	-0.989490243075	0.0008015

7.8.2 3-D Cartesian Assembly Row "Swing"

We now return to the assembly row "swing" case that we used to compare a "virtual reference" to Monte Carlo in Section 6.4.4. Previously, we compared the "virtual reference" to Monte Carlo; now we compare "virtual density" perturbation theory to the "virtual reference". Table 7.10 shows agreement to less than 0.001%.

7.8.3 3-D Cartesian Assembly Row "S-Shape" Bow

The assembly row "swing" is nothing more than assembly row shifting with a linear axial dependence. Now let us consider a true "bowing" scenario. Since we have already examined flowering cases with cosine axial shapes, we will now examine bowing with an "s-shape" axial dependence. Let Δx be the displacement of an assembly row in the x direction.

$$\Delta x(z) = \epsilon \sin\left[\frac{\pi\left(z - 2z_{\text{top}}/3\right)}{2z_{\text{top}}/3}\right] \tag{7.21}$$

Here ϵ is a small perturbation distance. In our "virtual reference" case, we set it to $a/1000$ (0.1% of the interstitial sodium zone width). Note that this sinusoidal curve is *not* axially symmetric - its zero-points occur at $z = 0$ and $z = (2/3)z_{\text{top}}$. This is important, because core symmetry would cause any axially symmetric bowing to produce a linear reactivity coefficient of zero.

Table 7.11 shows the results for this "s-shaped" row bow. Here we define a 1% bowing as $\epsilon/a = 0.01$. The "virtual density" perturbation theory and its "virtual reference" agree to 0.002%.

7.9 Numeric Results for Full-Core 3-D Distortions in Hexagonal-Z Geometry

Now that we have exhausted various distortion scenarios for simple Cartesian geometries, let us now turn to realistic fast reactor models: the full-core 3-D hex-z FFTF and Jōyō benchmarks. We already validated the whole-core uniform "virtual density" theory for these

Table 7.11: "S-Shape" Bowing of an Assembly Row in 3-D Cartesian Geometry

	k_{eff}	pcm/%	error (%)
unperturbed	1.006183941228		
VirDenT		-0.908252341874	
"virtual reference" 0.1%	1.006183849278	-0.908233319663	0.002094

benchmarks, but now we return to them for validation of the non-uniform "virtual density" theory.

7.9.1 Spatial Distributions of Reactivity due to Anisotropic Swelling

Before we analyze specific distortion cases, we can study the axial and radial distributions of the short-hand reactivity quantities: S, L_z, L_r, Γ_z, and Γ_r. We can also study the axial and radial distributions of their sums: axial swelling, radial swelling, isotropic swelling, and density reduction. Figures 7.14 and 7.15 show axial distributions for Jōyō. Figures 7.16 - 7.23 show radial distributions for Jōyō. Figures 7.24 and 7.25 show axial distributions for FFTF. Figures 7.26 - 7.31 show radial distributions for FFTF. Note that we skip over the isotropic swelling and density reduction reactivity coefficients for FFTF so not to be too redundant. Many of these distributions are intuitive, and they are all instructive. Read each caption for detailed observations, and see Appendices J and K for Jōyō and FFTF benchmark descriptions and core maps.

Note that the axial distributions of Γ_r are zero, because Γ_r integrated over any through-core axial slice is always zero. Similarly, the radial distributions of Γ_z are zero, because Γ_z integrated over any whole assembly is always zero.

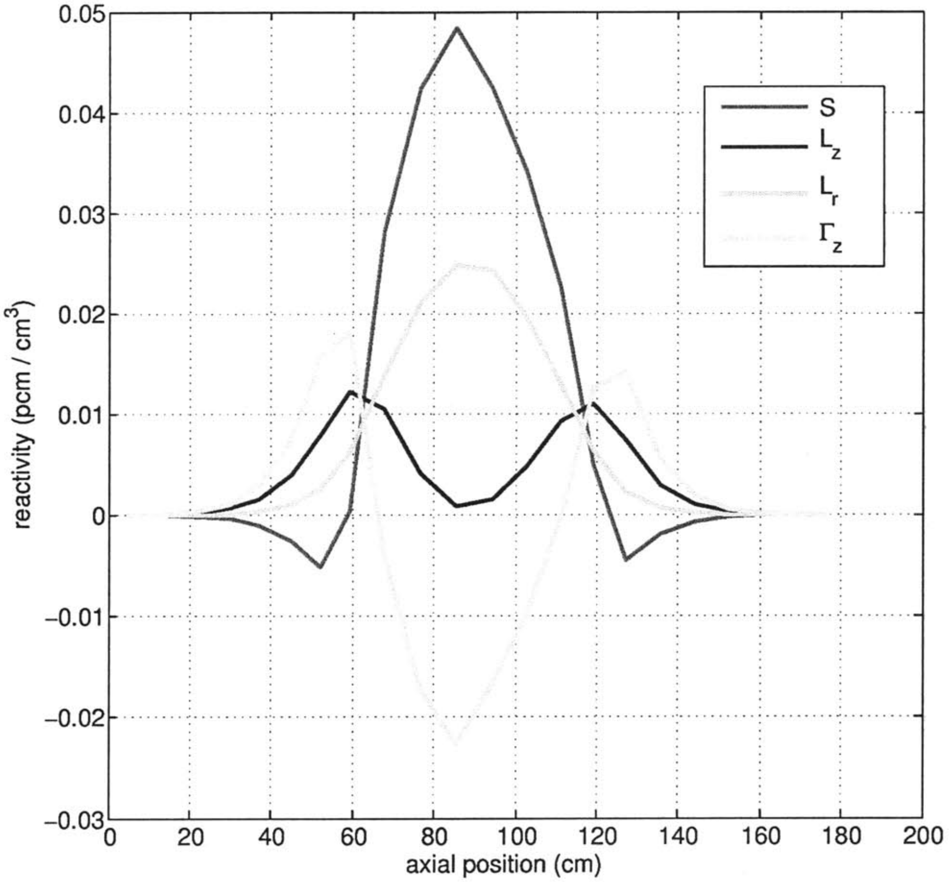

Figure 7.14: Axial distributions of S, L_z, L_r, and Γ_z in Jōyō. Note that Γ_r is not shown, because it is zero for each through-core axial slice. Adding these distributions yields S - L_z - L_r + $\Gamma_z = 0$. The spectral distribution S is positive in the fueled region and negative in the axial reflectors and shields. L_r and L_z are always positive, while Γ_z can be either positive or negative.

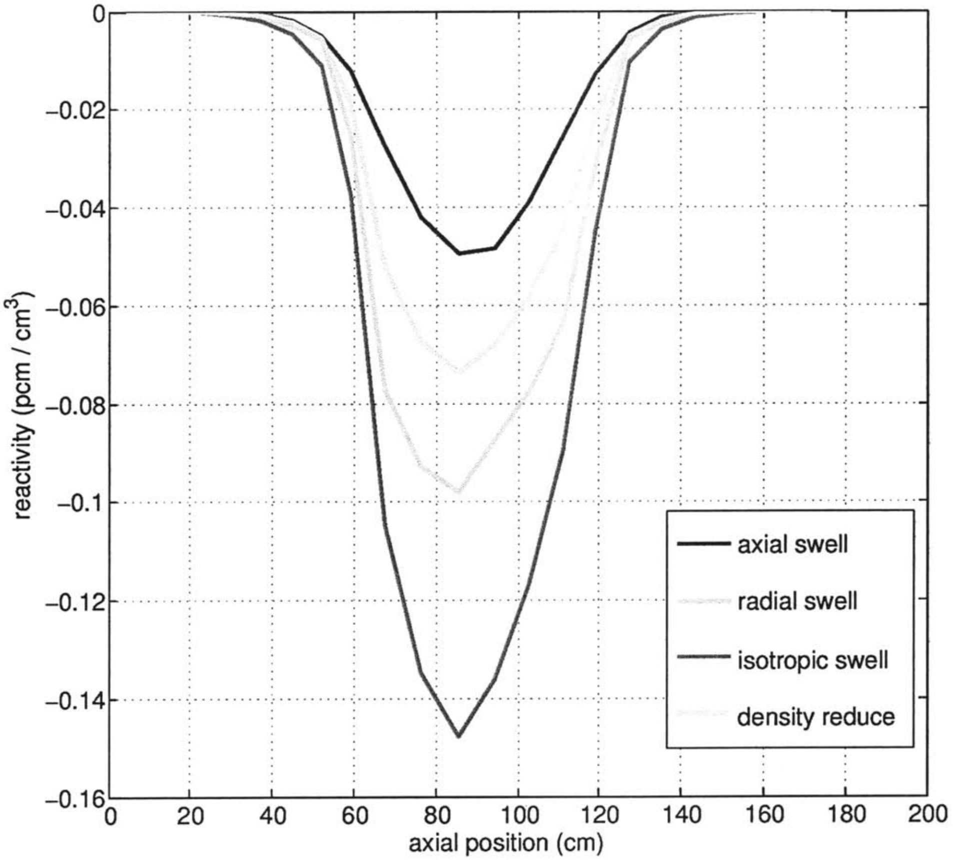

Figure 7.15: Axial distributions of axial, radial, and isotropic swelling (as well as density reduction) in Jōyō. Note that axial and radial swelling add to precisely equal isotropic swelling, which itself is twice a density reduction. Thus, a density reduction is the average of axial and radial swelling.

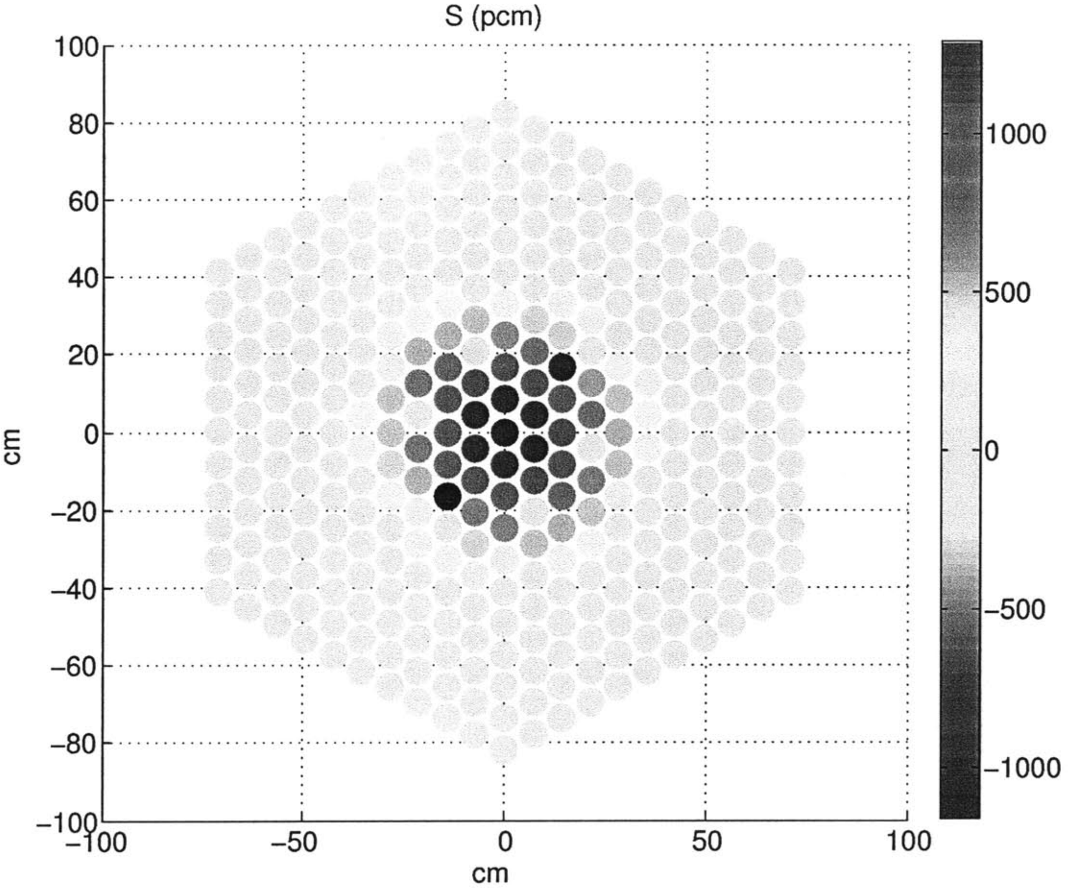

Figure 7.16: Spectral reactivity S in each assembly in Jōyō. This is very positive in the fuel assemblies, very negative in the "regulation" control assemblies, and slightly negative in the reflector and shield assemblies.

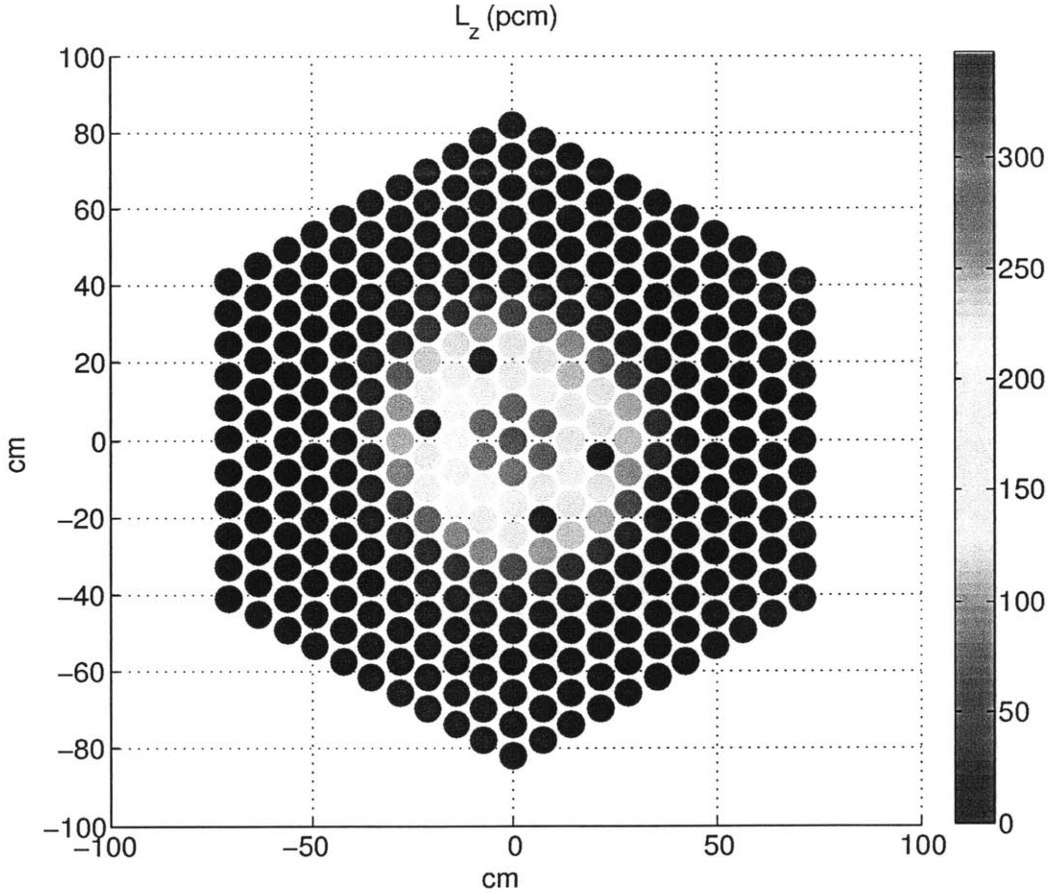

Figure 7.17: Axial leakage reactivity L_z in each assembly in Jōyō. This is always positive but especially positive in the four "safety" control assemblies.

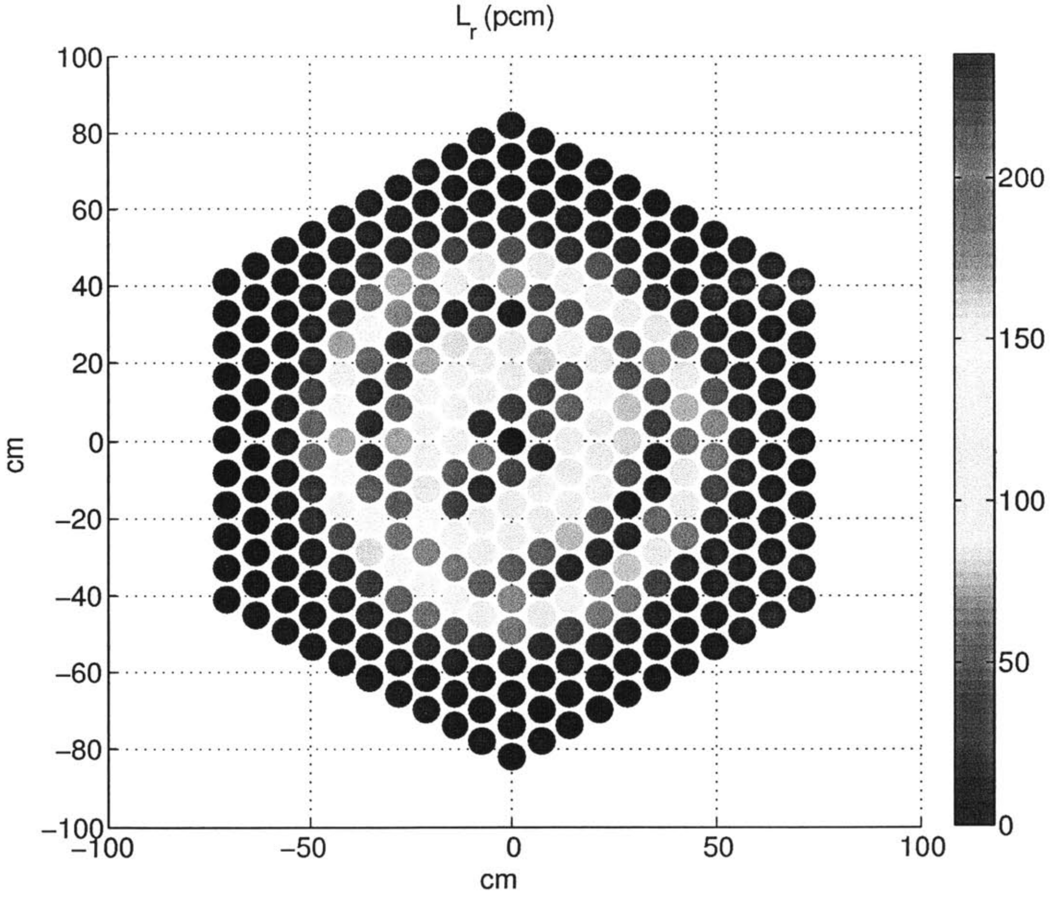

Figure 7.18: Radial leakage reactivity L_r in each assembly in Jōyō. This is always positive and has an annular distribution.

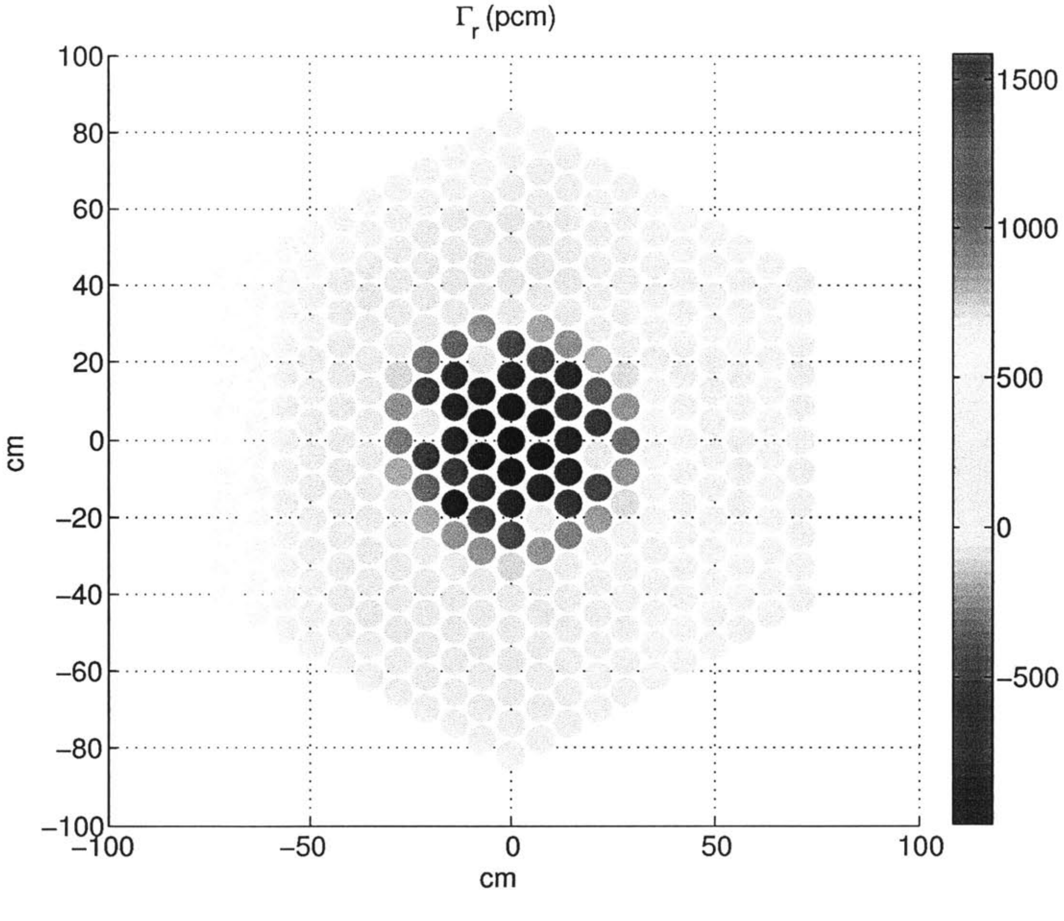

Figure 7.19: Radial surface leakage reactivity Γ_r in each assembly in Jōyō. Note that Γ_z is zero within each whole assembly. This is negative throughout most of the fuel, very positive in the "regulation" control assemblies, less positive in the "safety" control assemblies, and slightly positive in the reflector and shield assemblies.

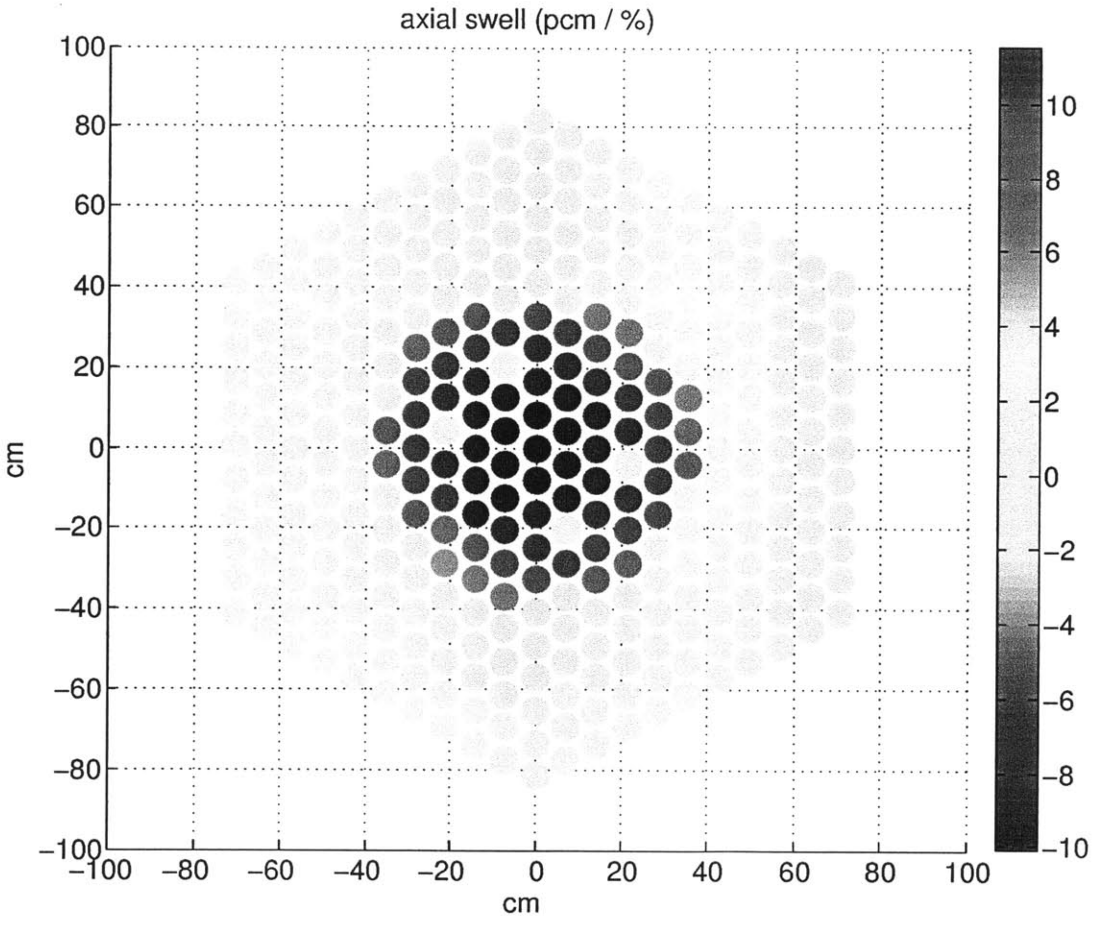

Figure 7.20: Axial swelling reactivity in each assembly of Jōyō. This is strongly negative in fuel assemblies and strongly positive in control assemblies.

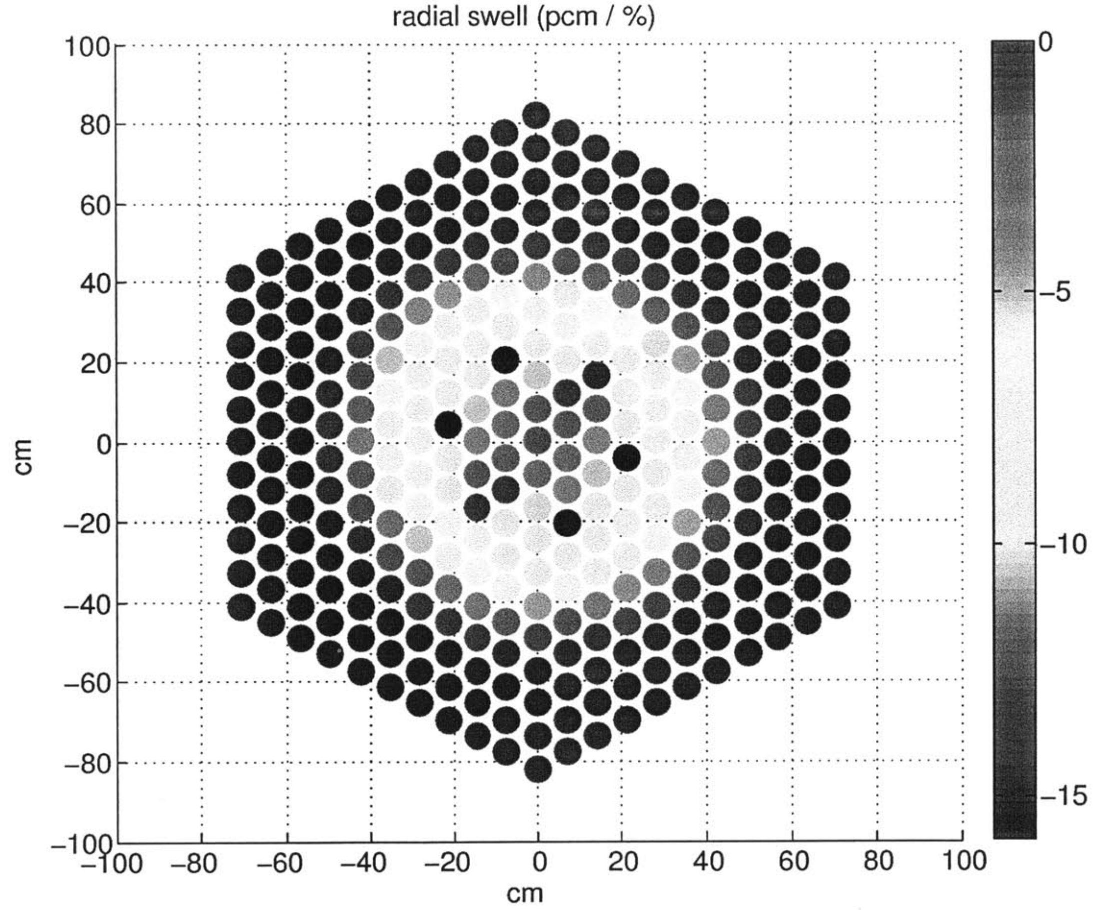

Figure 7.21: Radial swelling reactivity in each assembly of Jōyō. Note that radial swelling of a single assembly will cause it to unphysically "overlap" with neighboring assemblies, but this distribution may still be useful for analyzing core flowering scenarios. It is negative everywhere but especially so in the control assemblies. It is almost precisely zero in the central assembly.

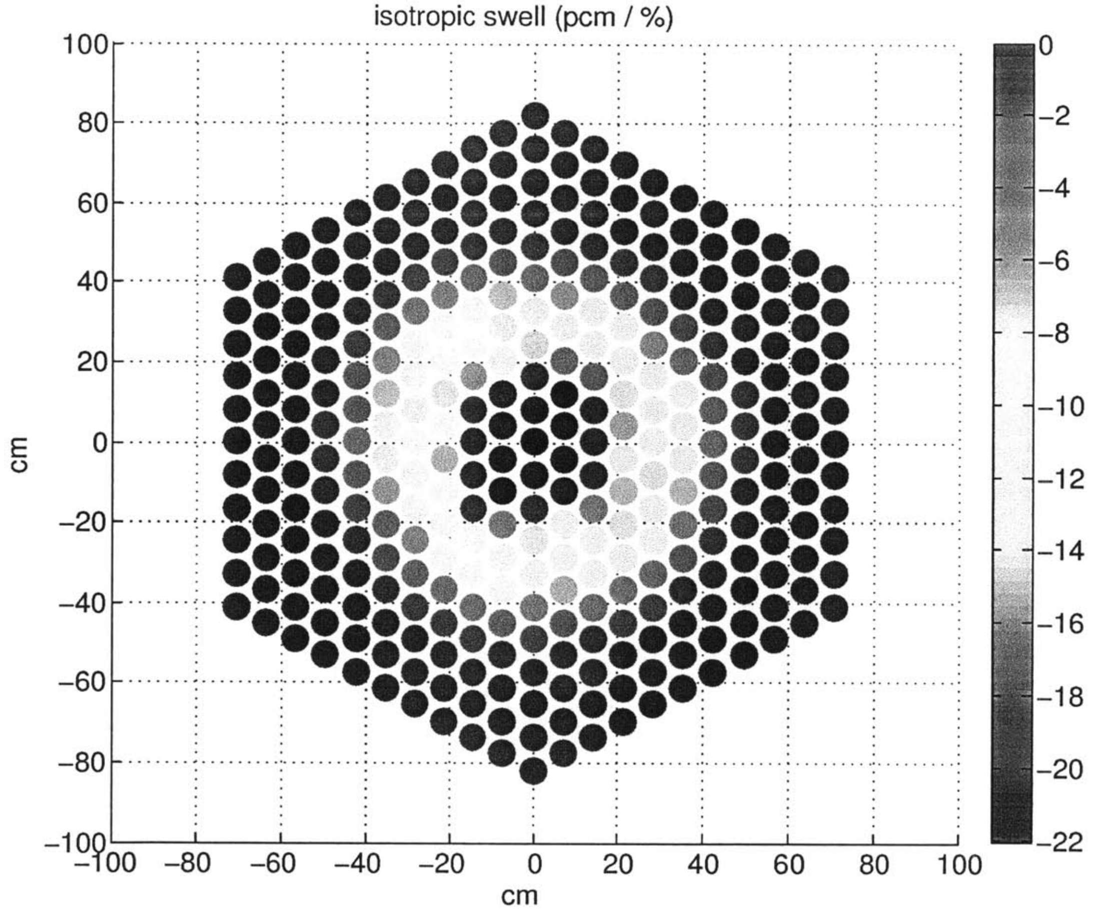

Figure 7.22: Isotropic swelling reactivity in each assembly of Jōyō. Note that isotropic swelling of a single assembly will cause it to unphysically "overlap" with neighboring assemblies. This distribution is the simple sum of the axial and radial swelling distributions in Figures 7.20 and 7.21, respectively.

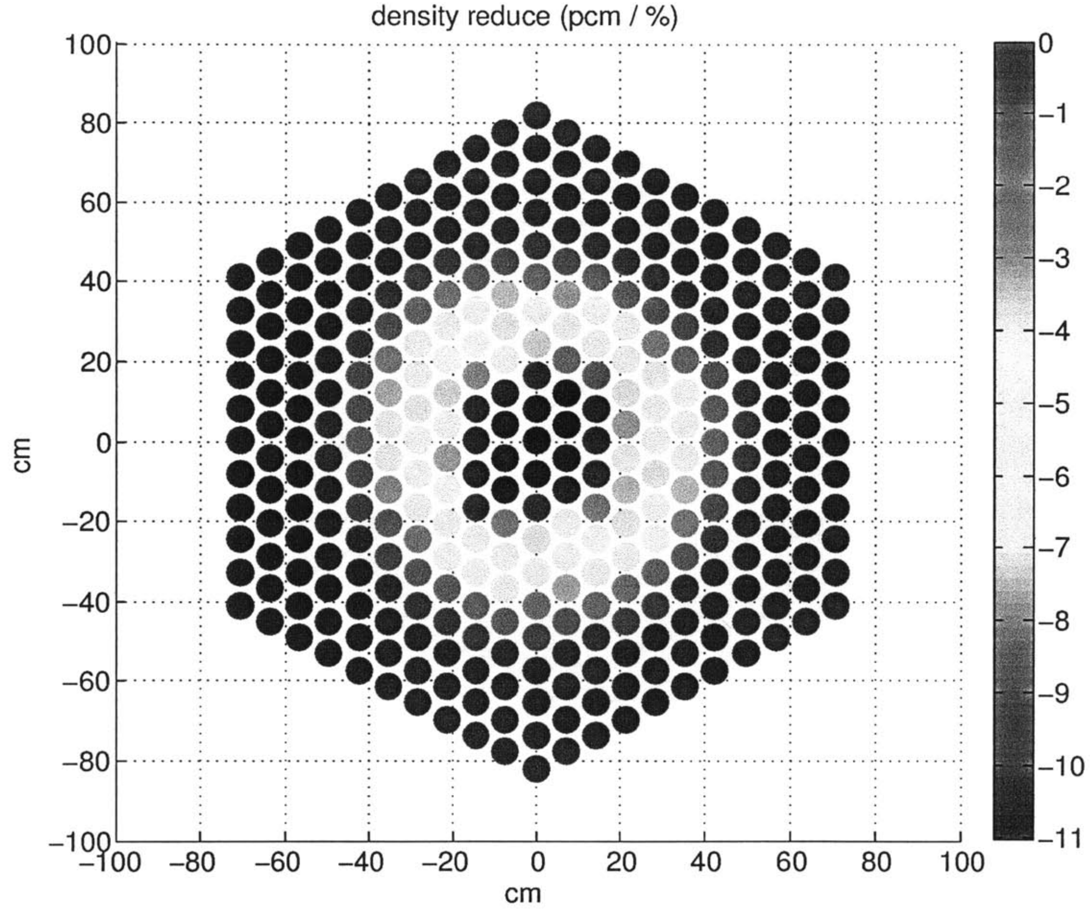

Figure 7.23: Density reduction reactivity in each assembly of Jōyō. This is precisely half the isotropic swelling distribution in Figure 7.22.

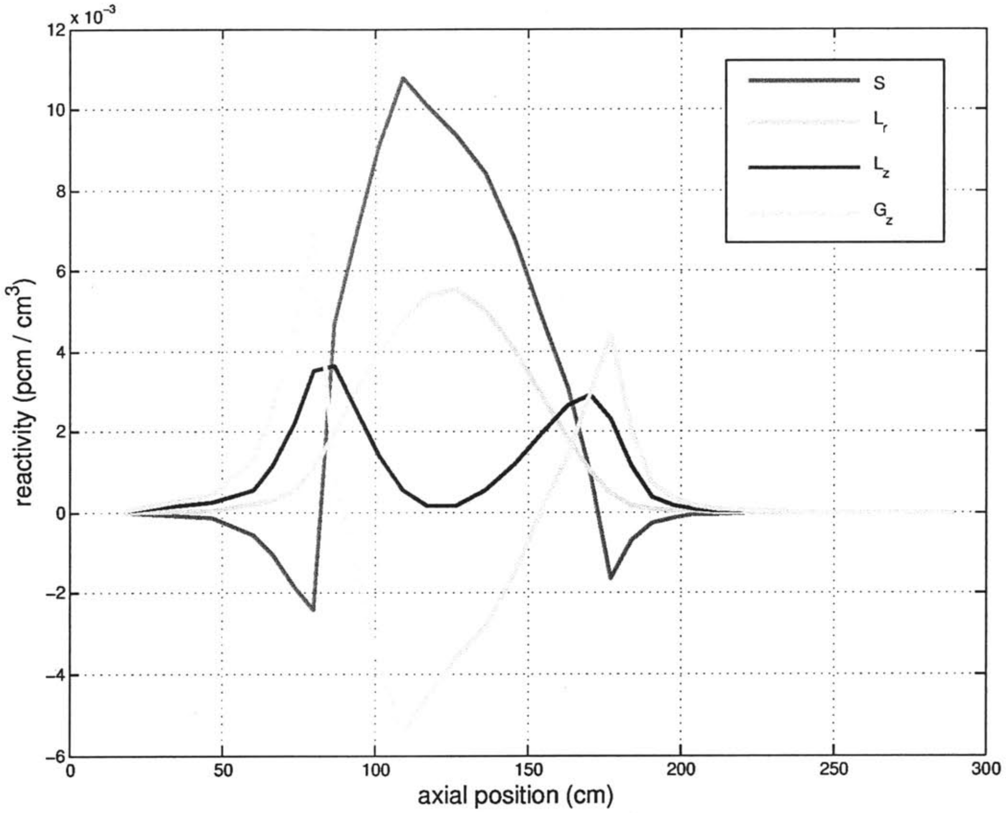

Figure 7.24: Axial distributions of S, L_z, L_r, and Γ_z in FFTF. Note that Γ_r is not shown, because it is zero for each through-core axial slice. Adding these distributions yields S - L_z - $L_r + \Gamma_z = 0$. The spectral distribution S is positive in the fueled region and negative in the axial reflectors and shields. L_r and L_z are always positive, while Γ_z can be either positive or negative.

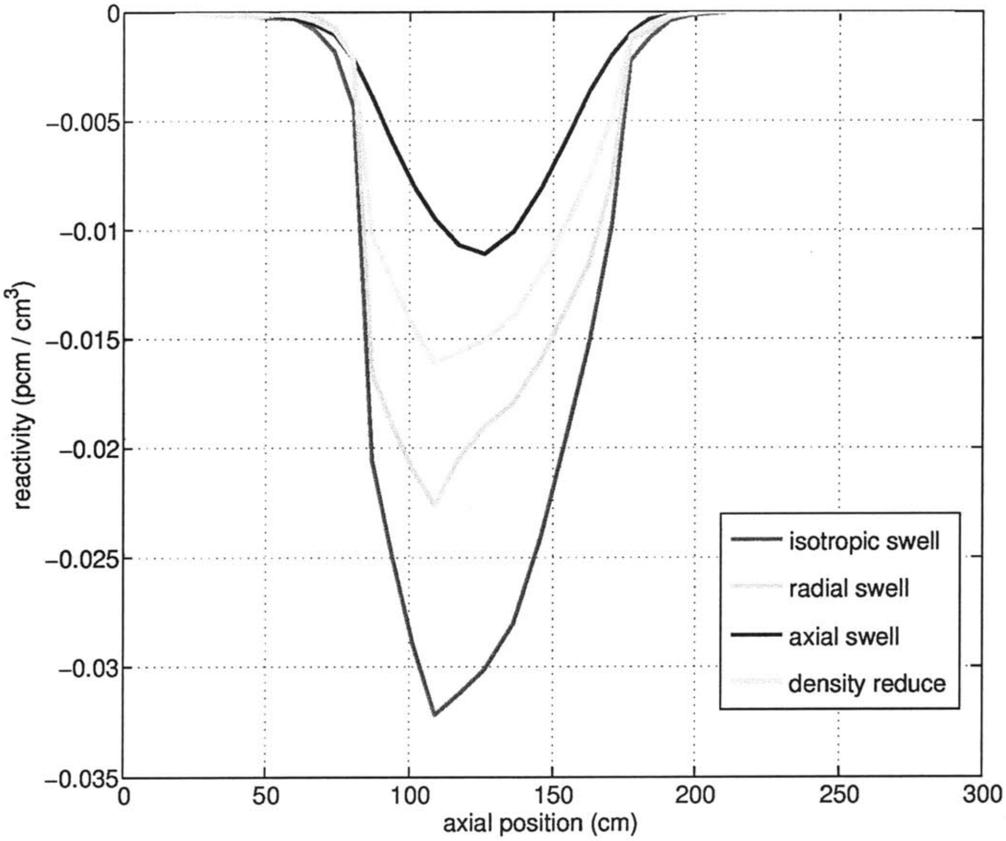

Figure 7.25: Axial distributions of axial, radial, and isotropic swelling (as well as density reduction) in FFTF. Note that axial and radial swelling add to precisely equal isotropic swelling, which itself is twice a density reduction. Thus, a density reduction is the average of axial and radial swelling.

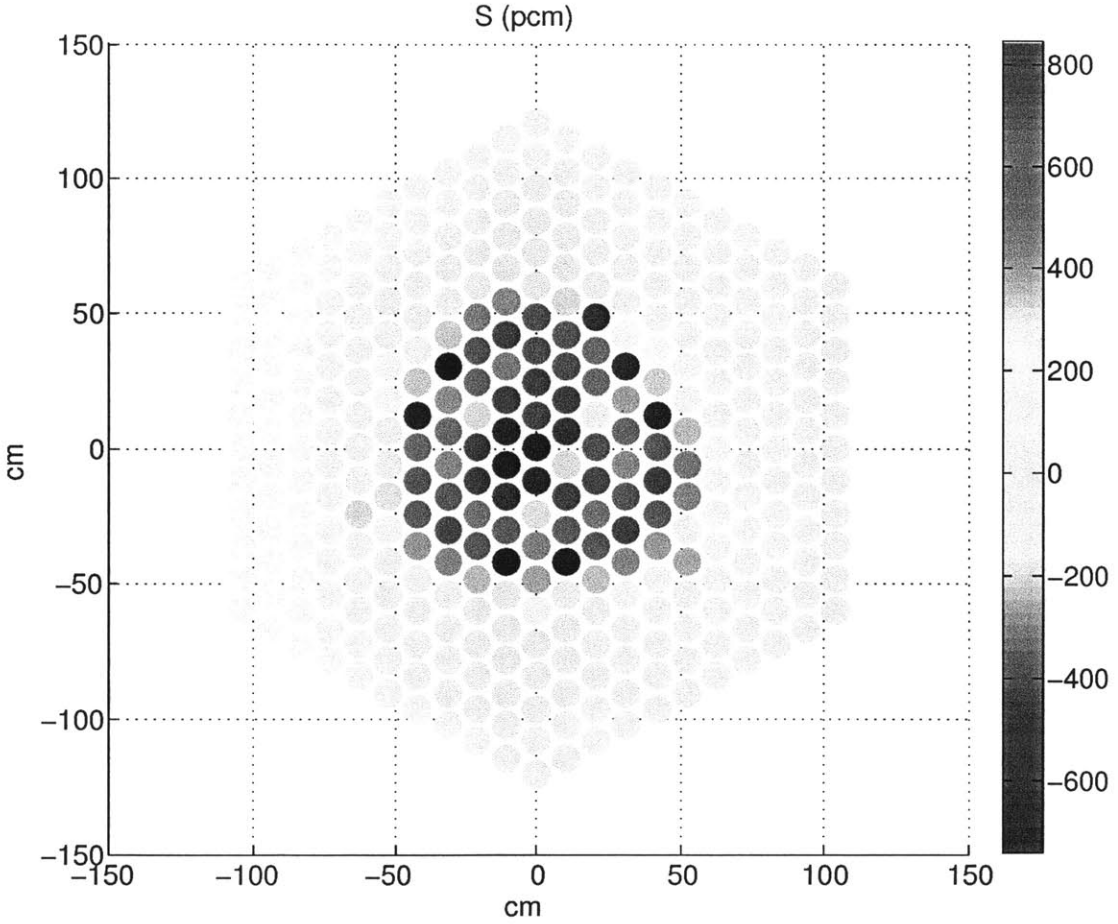

Figure 7.26: The spectral reactivity S in each assembly in FFTF. This is strongly positive in fuel assemblies, strongly negative in control assemblies, and slightly negative in "shim" and reflector assemblies.

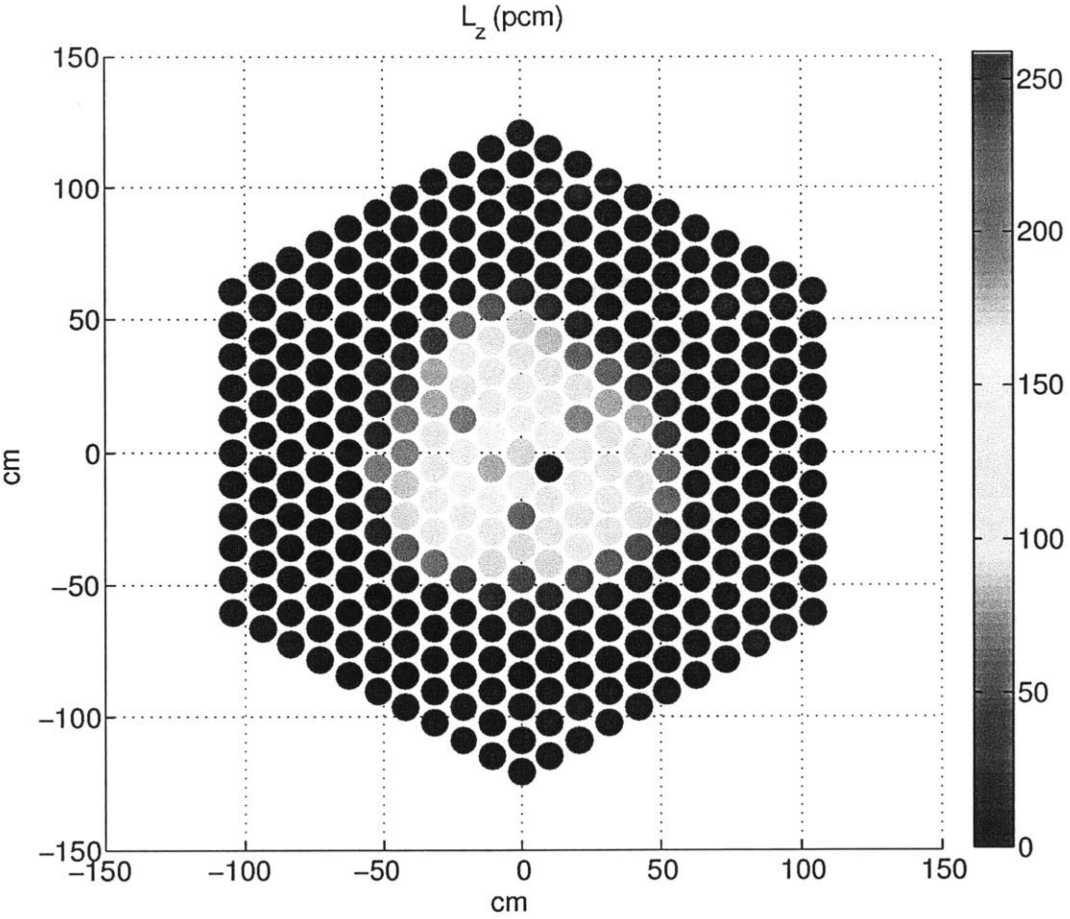

Figure 7.27: Axial leakage reactivity L_z in each assembly in FFTF. This is positive everywhere, especially in the primary control assemblies and the "in-reactor thimble".

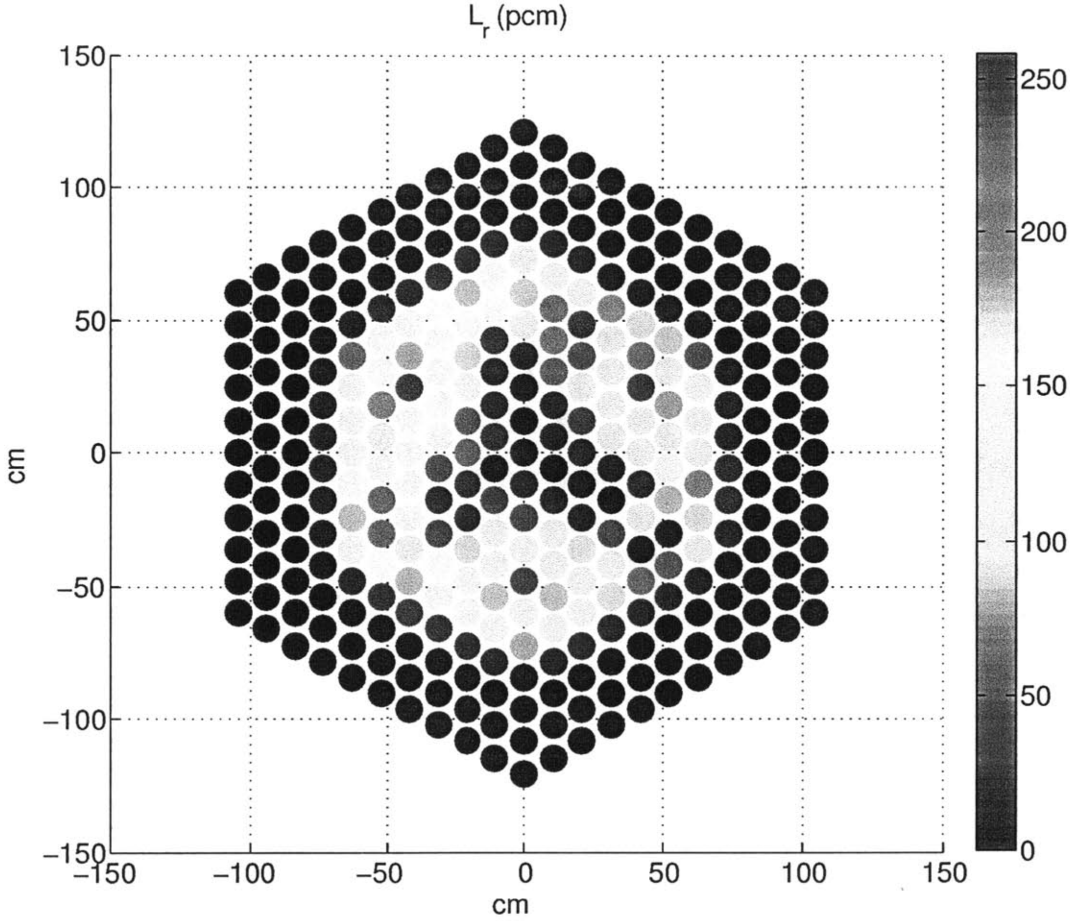

Figure 7.28: Radial leakage reactivity L_r in each assembly in FFTF. This is positive everywhere, but roughly "lobed annular" in shape.

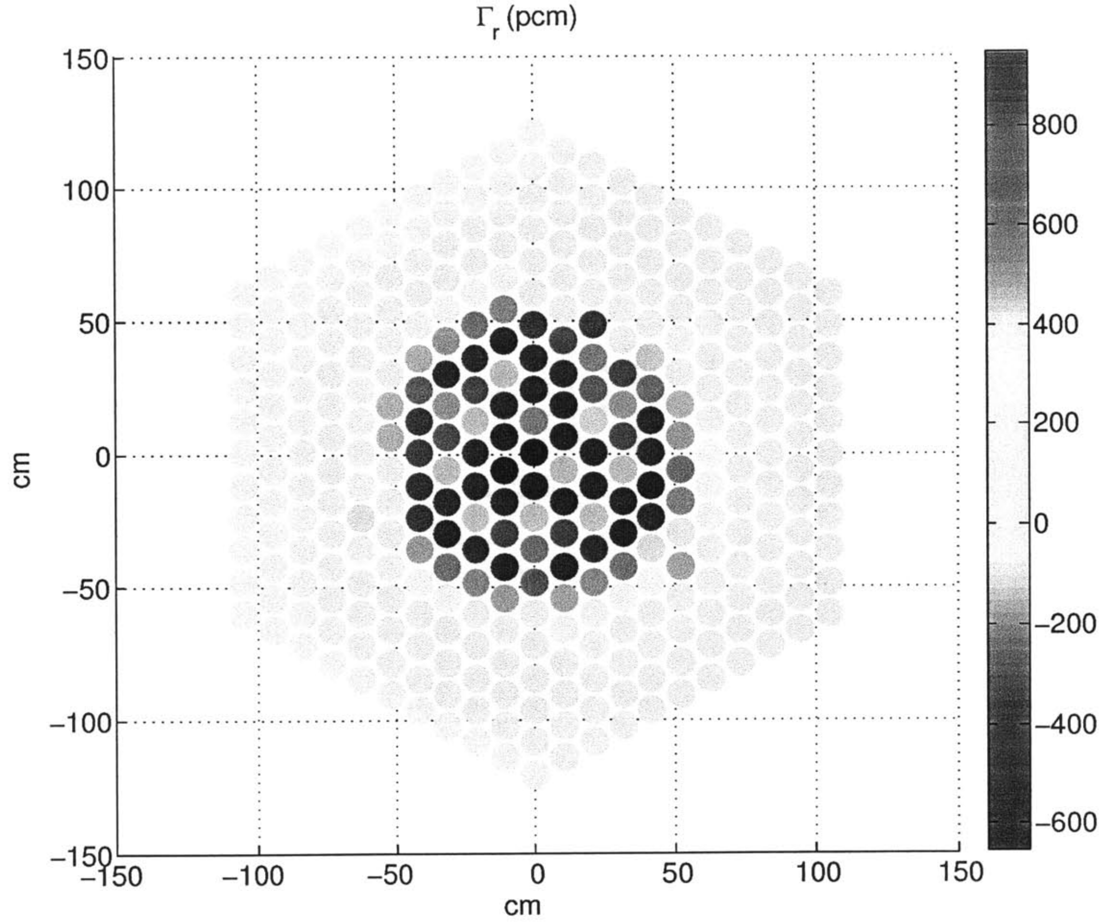

Figure 7.29: Radial surface leakage reactivity Γ_r in each assembly in FFTF. Note that Γ_z is zero within each whole assembly. This is negative in fuel assemblies, positive in control assemblies, and less positive in test and reflector assemblies.

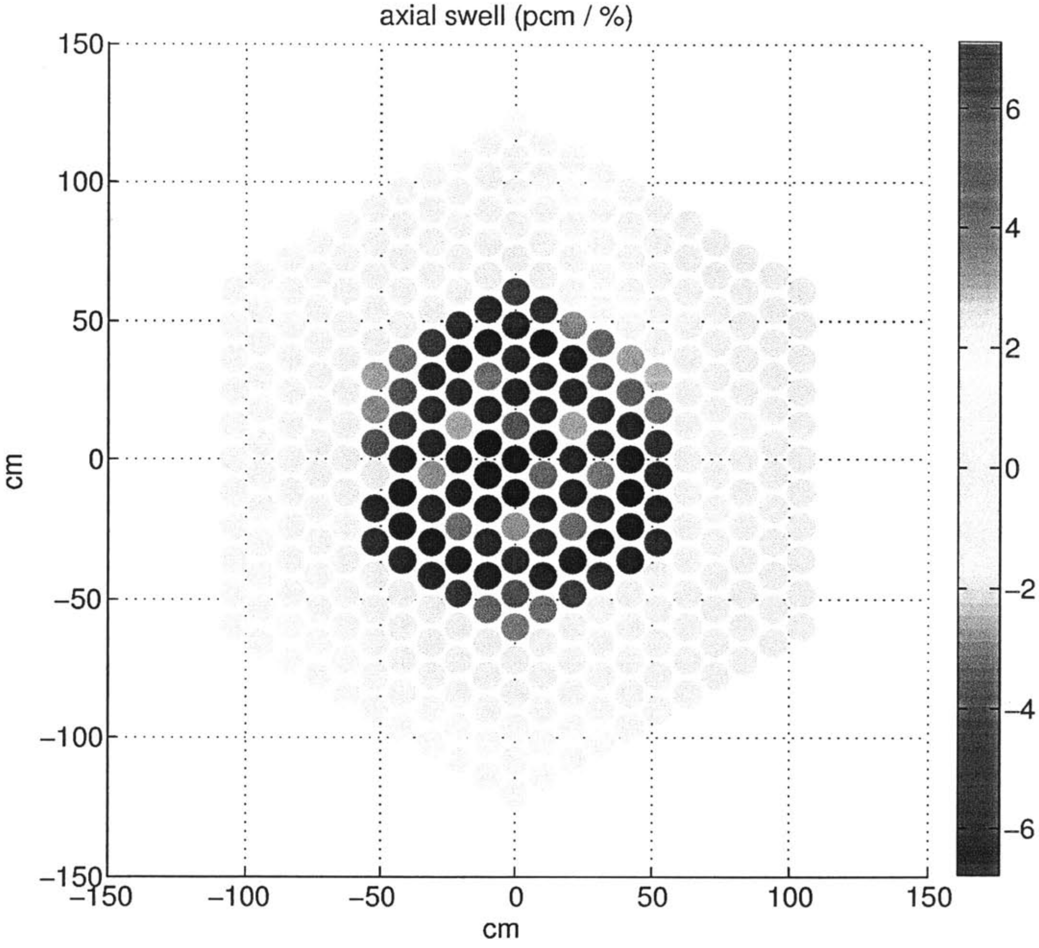

Figure 7.30: Axial swelling reactivity in each assembly of FFTF. This is negative in the fuel and positive in control and test assemblies.

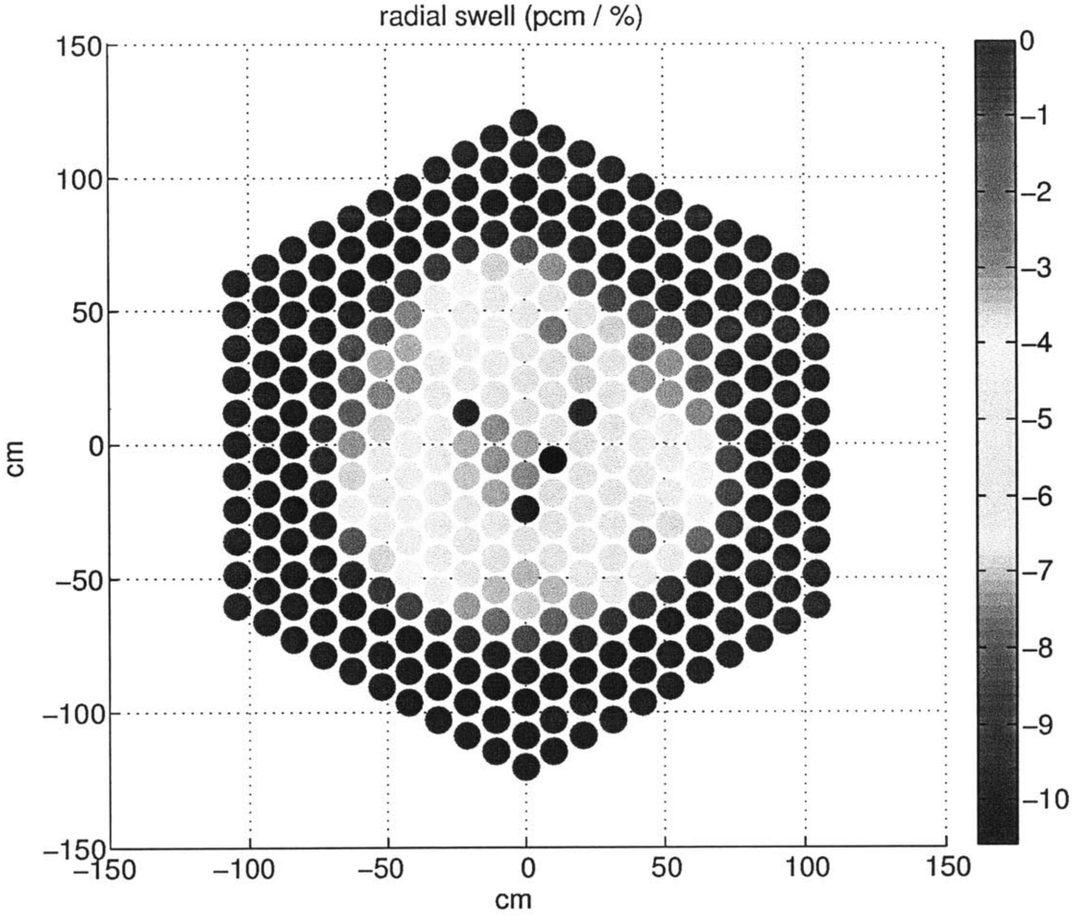

Figure 7.31: Radial swelling reactivity in each assembly of FFTF. Note that radial swelling of a single assembly will cause it to unphysically "overlap" with neighboring assemblies, but this distribution may still be useful for core flowering scenarios. This distribution is always negative, especially so in control assemblies.

Table 7.12: "Virtual Density" Axially Non-Uniform Axial Swelling for Jōyō

Jōyō	k_{eff} ref.	pcm/% ref.	pcm/% VirDenT	error (%)
unperturbed	0.973187327			
z = 10-15	0.972846003	-360.5174435	-360.9862082	0.130%
z = 12-19	0.972985511	-213.1340199	-213.5042931	0.170%
z = 1-8	0.973159281	-29.61341677	-29.68877639	0.250%
z = 16-22	0.973177772	-10.0886773	-10.12323552	0.340%
z = 16-17	0.973178469	-9.35277827	-9.381242643	0.304%
z = 1-8, 16-17	0.973150422	-38.96822254	-39.07001903	0.261%
z = 11-12, 14-15	0.972974942	-224.2983084	-224.5937528	0.132%
z = 8-9, 15-16	0.973099677	-92.55462961	-92.7119564	0.170%

7.9.2 Axially Non-Uniform Axial Swelling

Now we can begin to apply "virtual density" perturbation theory to practical swelling scenarios. Here we use the triangle-z finite difference equations derived in Section 7.3.2.

First consider *axial* swelling that is non-uniform in the *axial* direction. This constitutes axially swelling an entire through-core axial slice as illustrated in Figure 7.32. Fortunately, constructing a reference case for this scenario requires no "virtual mesh". Table 7.12 shows the results, in which we swell various sets of axial core slices by 1%. The errors are greater than 0.1% merely because we chose a 1% swelling reference case - they would be smaller had we chosen a 0.1% swelling reference case.

Throughout these non-uniform triangular-z and hexagonal-z analyses, we will employ the coordinate system (i, j, z) to identify a single axial segment z of a given assembly (i, j). The axial position is $z \geq 1$, the assembly ring index is $i \geq 1$, and the assembly index within each ring (clockwise from 30 degrees) is $j \geq 1$.

We compute the "virtual density" perturbation theory coefficients in Table 7.12 using Equation 5.45. In these through-core axial slices, Γ_r is always zero. Thus, the reactivity coefficients are simply

$$\left[\frac{d\rho}{df}\right]_{\text{swell,ax}} = -2L_r \tag{7.22}$$

This is a prime example of why the "infinitude of equalities" can be expeditious. We can now compute the reactivity coefficient via only one term!

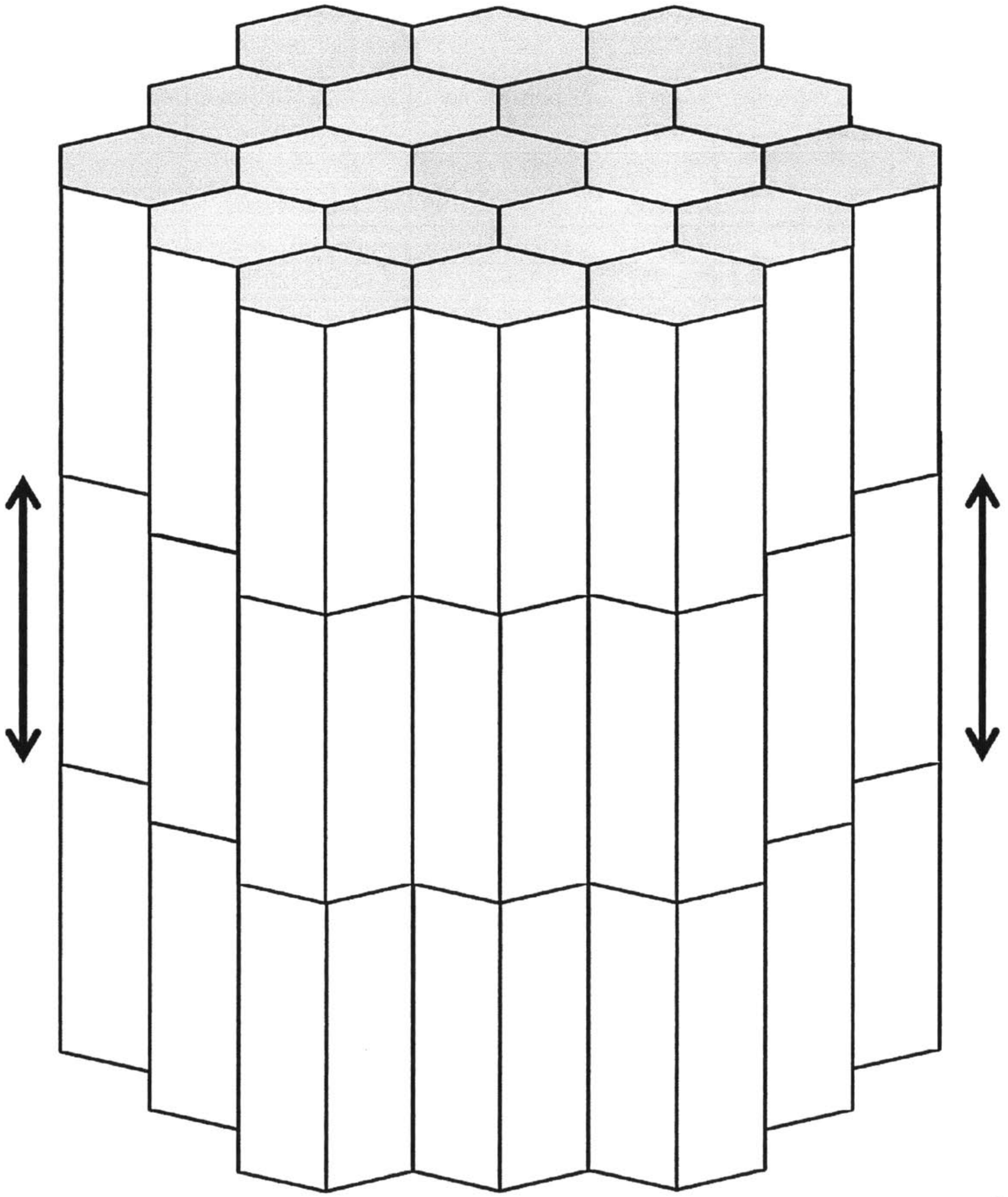

Figure 7.32: Axially non-uniform axial swelling of a through-core axial slice. Here one axial slice of the core swells in the axial direction, increasing the total core height. This is one of few non-uniform distortion cases for which we can easily construct a true reference case (not a "virtual reference").

Table 7.13: "Virtual Density" Radially Non-Uniform Axial Swelling for Jōyō

Jōyō	k_{eff} ref.	pcm/% virtual ref.	pcm/% VirDenT	error (%)
unperturbed	0.973187327			
i = 1-2	0.973123114	-67.80409977	-67.87510217	0.105%
i = 3-4	0.973044661	-150.6577384	-150.7823601	0.083%
f for $i = 5$, f^2 for $j = 1$	0.972992614	-205.6311762	-205.8215998	0.093%
$f(i,j) \propto$ assembly power	0.972873912	-331.0295589	-331.0559652	0.008%

7.9.3 Radially Non-Uniform Axial Swelling

Now consider *axial* swelling that is non-uniform in the *radial* direction. This amounts to different assemblies swelling axially by different factors. Figure 7.33 illustrates assemblies swelling axially by different factors such that the innermost assemblies swell more (perhaps proportional to power or temperature). Figure 7.34 illustrates assemblies swelling axially by random factors.

Unlike *axially* non-uniform axial swelling, *radially* non-uniform axial swelling requires a "virtual reference". Table 7.13 shows the results for four different scenarios:

- Assembly rings 1 and 2 (i = 1-2) swelling axially. These are the 7 innermost assemblies.
- Assembly rings 3 and 4 (i = 3-4) swelling axially. This is an annular group of 30 assemblies.
- Assembly ring $i = 5$ swelling axially by a factor f in tandem with all $j = 1$ assemblies (a "radial row" of assemblies) swelling axially by a factor f^2. Thus, the single assembly (i,j) = (5,1) constituting the intersection of these regions swells axially by a factor f^3.
- Each assembly swelling axially by a factor $f(i,j)$ in proportion to its total power.

We compute the "virtual density" perturbation theory coefficients in Table 7.13 using Equation 5.44. In whole assemblies, Γ_z is always zero. Thus, the reactivity coefficients are simply

$$\left[\frac{d\rho}{df}\right]_{\text{swell,ax}} = -S + L_z - L_r \tag{7.23}$$

So we have once again utilized the "infinitude of equalities" to simplify the computation, because we can simply choose whichever equivalent expression results in zero surface terms!

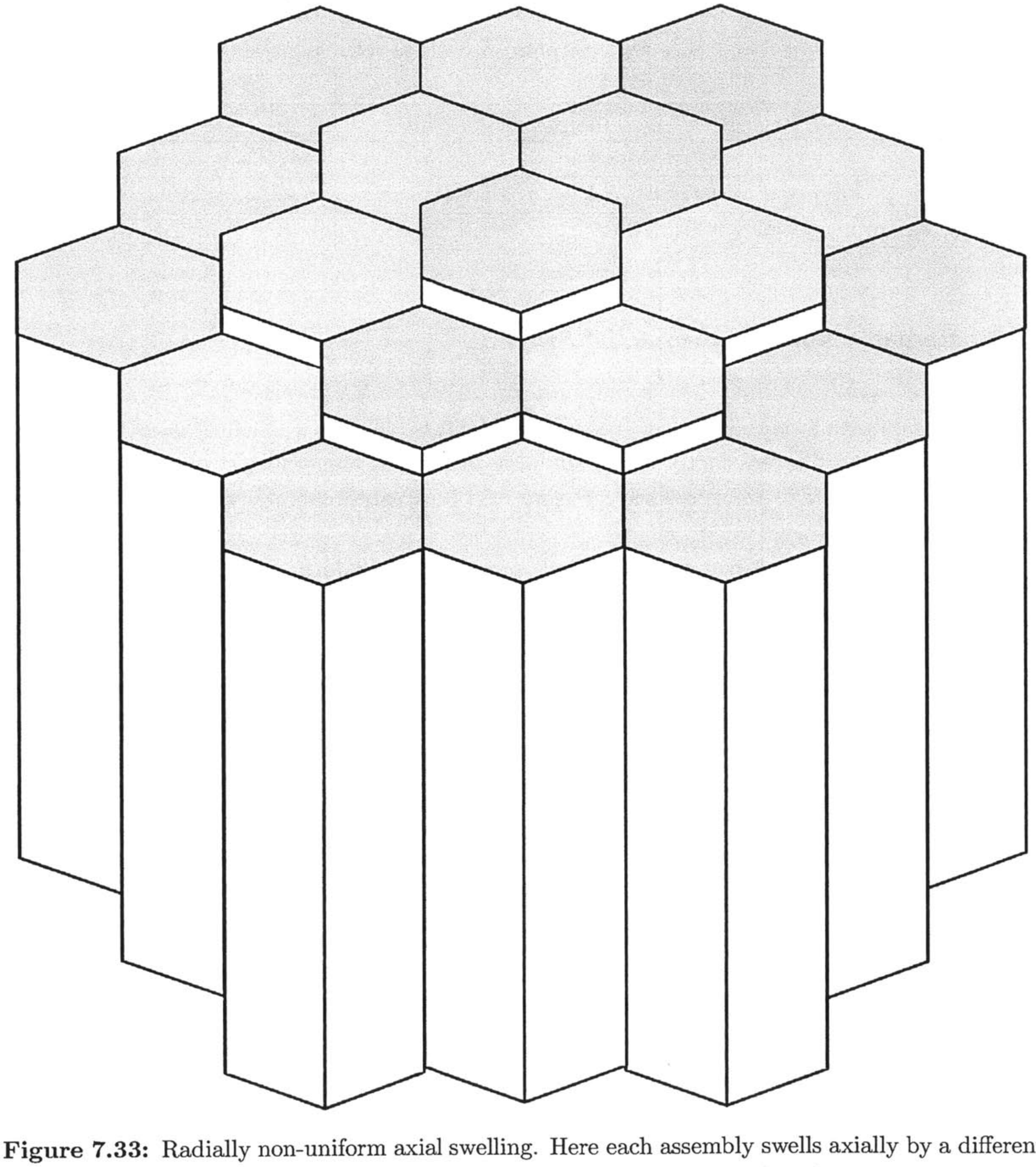

Figure 7.33: Radially non-uniform axial swelling. Here each assembly swells axially by a different factor, perhaps proportional to power or temperature. Validating cases like this requires a "virtual reference".

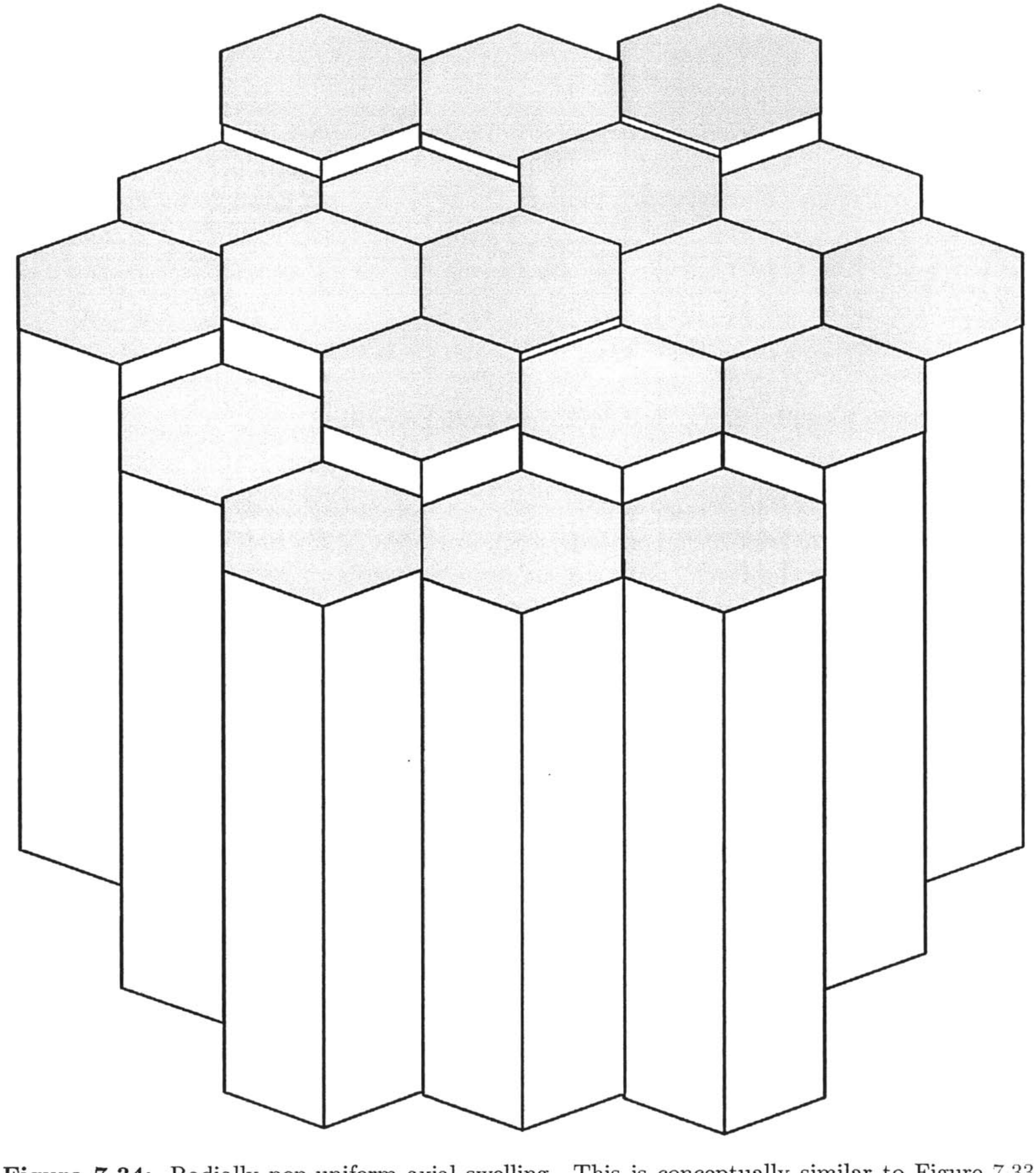

Figure 7.34: Radially non-uniform axial swelling. This is conceptually similar to Figure 7.33, except now each assembly swells axially by a randomly-selected factor.

Table 7.14: "Virtual Density" Axially and Radially Non-Uniform Axial Swelling for Jōyō

Jōyō	k_{eff} ref.	pcm/% virtual ref.	pcm/% VirDenT	error (%)
unperturbed	0.973187327			
i = 3-5, z = 8-11	0.973053911	-140.8879898	-140.9419228	0.038%
i = 4-6, z = 9-15	0.972974044	-225.2468139	-225.3048945	0.026%
i = 3-5, z = 11-13	0.973051788	-143.1303975	-143.1722496	0.029%
i = 1-2, z = 10-13	0.973142535	-47.29663122	-47.3381405	0.088%
i = 1-4, z = 8-12	0.973048321	-146.7922989	-146.8807177	0.060%
$f(i,j,z) \propto$ power	0.972927112	-274.8249477	-274.8326436	0.003%

7.9.4 Axially and Radially Non-Uniform Axial Swelling

Now consider *axial* swelling that is non-uniform in the *axial and radial* directions. This is simply a combination of the previous two scenarios. Now each axial segment of each assembly can swell axially by a different factor. Refer back to Figure 7.4 for an illustration.

Table 7.14 shows the results. We arbitrarily select various axial segments of various assembly groups. We also perform one case in which each (i, j, z) zone swells axially in proportion to its fission power.

We could compute the "virtual density" perturbation theory coefficients in Table 7.14 using any expression among Equations 5.41 - 5.47. Both Γ_z and Γ_r cannot be zero for an arbitrary triangular mesh cell, so we cannot avoid computation of at least one surface term. However, even though we cannot avoid computation of *either* surface term, we *can* avoid computation of *both* surface terms. Choosing Equation 5.45 avoids computation of Γ_z, while choosing Equation 5.44 avoids computation of Γ_r.

Table 7.15: "Virtual Density" Axially Non-Uniform Radial Swelling for Jōyō

Jōyō	k_{eff} ref.	pcm/% virtual ref.	pcm/% VirDenT	error (%)
unperturbed	0.973187327			
z = 12-14	0.972836558	-370.4969511	-371.0293602	0.144%
z = 15-19	0.973127562	-63.10773091	-63.21006741	0.162%
z = 11	0.973026313	-170.0362912	-170.3488787	0.184%
z = 1-8	0.973136617	-53.54554599	-53.66346039	0.220%
$f(z)$ linear	0.972722422	-491.1101011	-491.1816011	0.015%
$f(z)$ quadratic	0.970705725	-262.6928069	-262.4647498	0.087%
$f(z)$ cosine	0.972327156	-909.0252531	-908.8938323	0.015%
$f(z)^2 \propto$ power	0.97245447	-774.3787431	-774.4514445	0.009%

7.9.5 Axially Non-Uniform Radial Swelling

Now consider *radial* swelling that is non-uniform in the *axial* direction. Figure 7.35 illustrates one through-core axial slice swelling radially outward. Figure 7.36 shows numerous axial slices swelling by different magnitudes to reflect a cosine flowering scenario. This may occur when a core is constrained at its top and bottom but permitted to "bulge" outward around its center.

Table 7.15 shows the results for a variety of radial swelling cases. First we swell single axial slices (defined by a range of z values). Then we swell every axial slice with linear, quadratic, and cosine axial distributions. The cosine distribution reflects the "restrained" illustration in Figure 7.36, while the linear or quadratic distributions reflect the "unrestrained" illustration in Figure 1.3. Finally, we swell each axial slice so that f^2 (the fractional volume increase) is proportional to the slice's total fission power.

In order to construct these "virtual reference" cases, note that if the hexagonal assembly pitch in an axial slice increases by the factor f, then all material densities within that slice decrease by the factor f^2.

We compute the "virtual density" perturbation theory coefficients in Table 7.15 using Equation 5.52. Remember that Γ_r is zero in through-core axial slices, so the reactivity coefficients are simply

$$\left[\frac{d\rho}{df}\right]_{\text{swell,rad}} = -2S - 2L_z \tag{7.24}$$

Again, we have chosen an equivalent expression to avoid computation of the surface terms.

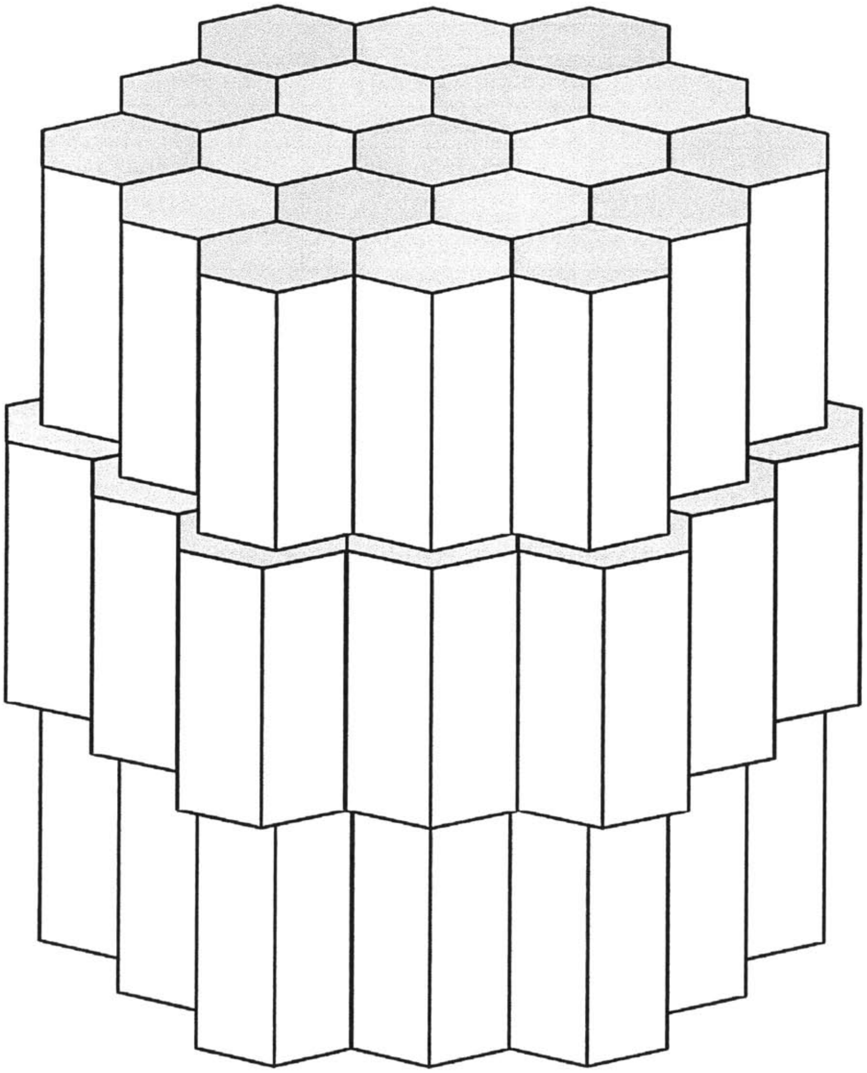

Figure 7.35: Axially non-uniform radial swelling. Here a through-core axial slice swells *radially* outward, increasing core diameter but not affecting core height. Validating cases like this requires a "virtual reference".

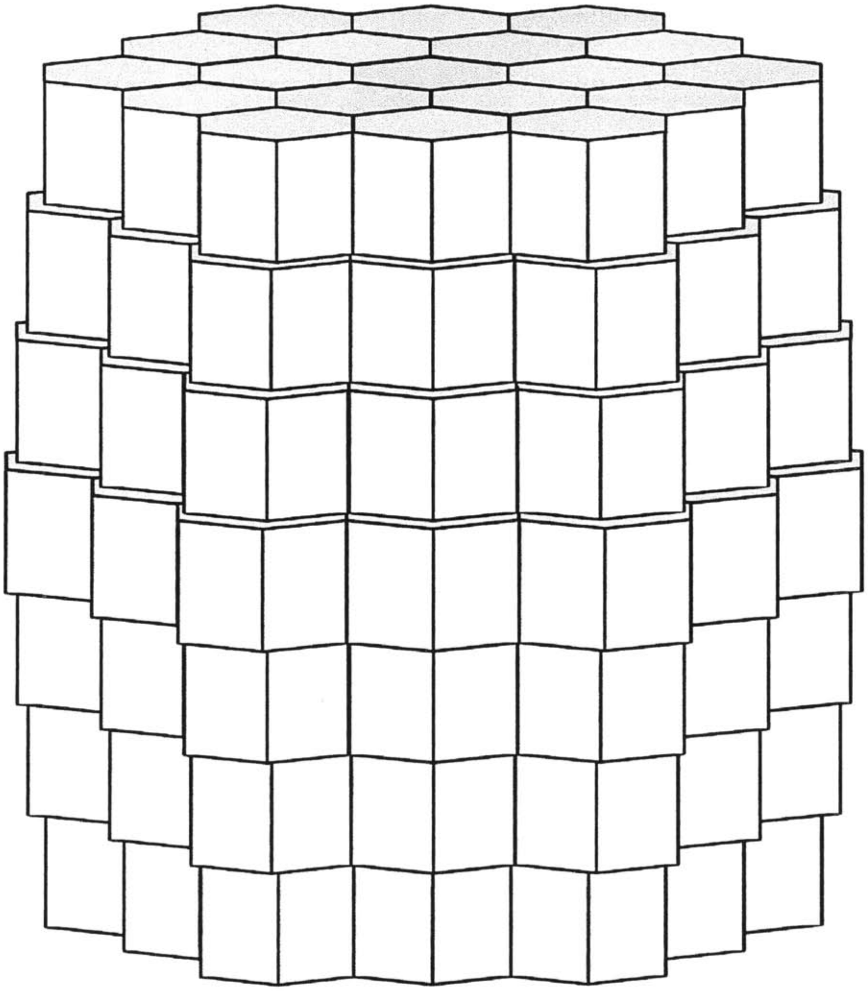

Figure 7.36: Axially non-uniform radial swelling with an axial profile. Here the core top and bottom do not radially swell, but the core middle "bulges" outward. This could be a model for a core that is restrained at only its top and bottom.

7.9.6 The Enigma of Radially Non-Uniform Radial Swelling

Now we have covered three fundamentally distinct swelling types in triangular-z or hexagonal-z geometry:

- *axial* swelling that is non-uniform in the *axial* direction
- *axial* swelling that is non-uniform in the *radial* direction
- *radial* swelling that is non-uniform in the *axial* direction

Of course, there is one swelling type missing here: *radial* swelling that is non-uniform in the *radial* direction. This swelling type is particularly cumbersome, because an individual assembly cannot swell radially without unphysically "overlapping" its neighboring assemblies. We can compute radial swelling for a whole core or a through-core axial slice, but can we compute it for a single assembly?

In reality, individual assemblies are separated by interstitial coolant zones, so a given assembly can move (slightly) radially outward without violating geometry. However, this is a localized heterogeneous effect, and so how can we model it on a coarse mesh of homogenized assemblies? We will devote much of Chapter 11 to solving this enigma.

7.10 The Exact Precision of Non-Uniform Anisotropic "Virtual Density" Perturbation Theory

Throughout all this numeric verification of the non-uniform anisotropic "virtual density" formalism, we have shown very small errors between perturbation theory and the (virtual or actual) reference cases - usually less than 0.1% and oftentimes much smaller.

It is crucial to understand that these errors originate not from the reactivity coefficient computed via "virtual density", but from the reference cases. All the reference cases contain curvature in reactivity, because the reactivity has non-zero first *and* second derivatives. The "virtual density" formalism predicts the first derivative *exactly*; all the errors we have shown here originate from the non-zero second derivative in the reference cases.

We now return to the case in which each axial segment of each assembly in Jōyō swells axially in proportion to its power. This computation appears in Table 7.14. We can compute the axial swelling reactivity coefficient via "virtual density" perturbation theory and compare it to a "virtual reference" case. As the swelling fraction f approaches 1 (and $f-1$ approaches 0), the error between the "virtual density" and "virtual reference" reactivity coefficients approaches 0. We illustrate this in Figure 7.37.

The reason for this decrease in error is *not* anything in "virtual density" perturbation theory, which predicts the *same* linear coefficient regardless of the swelling magnitude. Only the reference reactivity coefficient changes (converges to the correct answer) as the swelling magnitude approaches zero. Thus, we can say with confidence that the non-uniform anisotropic "virtual density" perturbation theory formalism is precise.

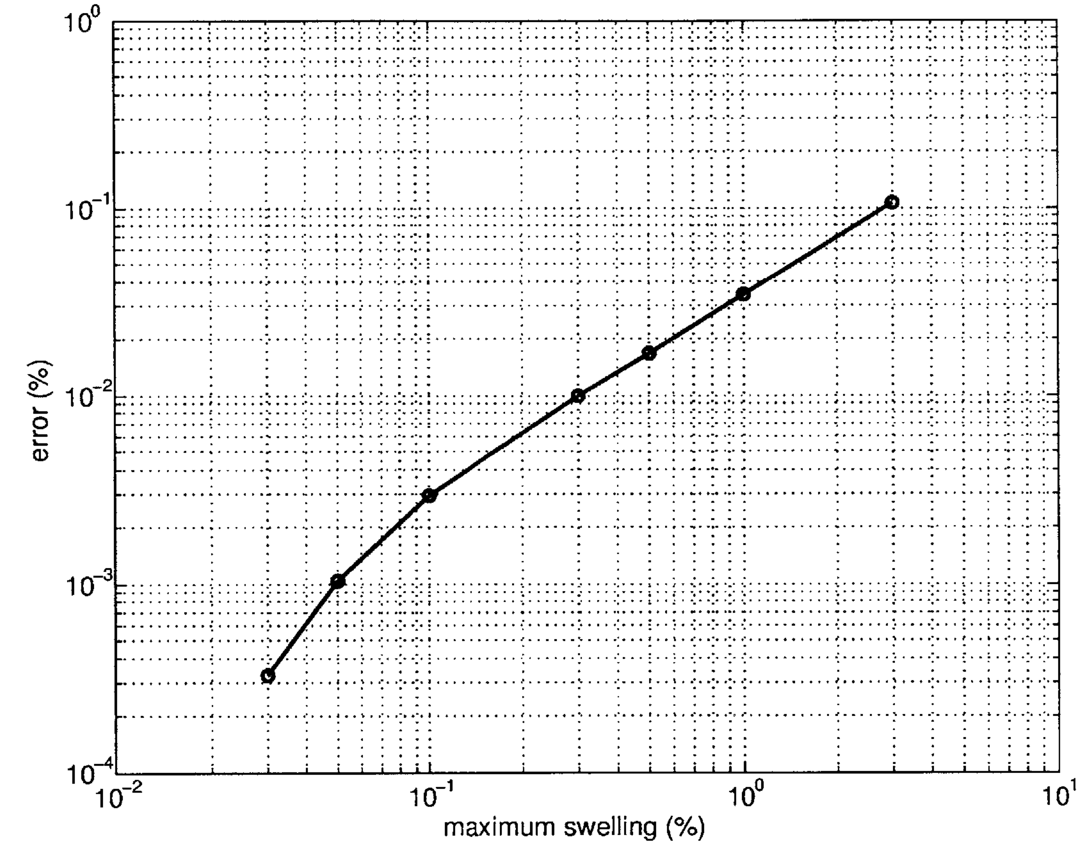

Figure 7.37: The error magnitude (%) between "virtual density" perturbation theory and a "virtual reference" eigenvalue calculation for the case in which each axial segment of each assembly in Jōyō swells axially in proportion to its power. Here the "maximum swelling" is the maximum value of $f - 1$ throughout the core. Clearly, the error approaches zero as the swelling approaches zero.

7.11 Summary

We have numerically validated the non-uniform anisotropic "virtual density" theory for a plethora of swelling cases in simple 2-D and 3-D Cartesian geometries as well as in full-core

3-D hexagonal-z benchmarks. In all cases, we find that reactivity coefficients agree precisely with diffusion reference cases. Errors are usually below 0.1% and often below 0.01%. We showed that these errors arise entirely from non-zero second derivatives of reactivity in the reference eigenvalue calculations, not the perturbation theory itself. Thus, the perturbation theory predicts reactivity coefficients that are *exact* in diffusion theory.

Moving forward, we write the "Virtual Density" Theory (VirDenT) distortion code, which computes reactivity coefficients due to non-uniform anisotropic swelling scenarios in fast reactors.

8 Performance of "Virtual Density" Perturbation Theory Relative to Traditional Boundary Perturbation Theory

8.1 Chapter Abstract

We compare and contrast "virtual density" perturbation theory with the traditional boundary perturbation theory developed by Pomraning, Larsen, and Rahnema. First, we mathematically prove that "virtual density" perturbations and traditional boundary perturbations are precisely equivalent for arbitrary 1-D problems, which constitute non-uniform isotropic expansions. We also mathematically prove that these two perturbation theories are equivalent for 2-D boundary shift problems, which constitute non-uniform *an*isotropic expansions. Extension of this proof to swellings or 3-D problems is straightforward. We compare the two theories numerically for a series of alternating uranium and sodium 1-D slabs in finite difference diffusion, and we show that "virtual density" theory predicts reactivities much more accurately and efficiently than traditional boundary perturbation theory. Boundary perturbation theory is often very inaccurate on a coarse mesh but converges to the "virtual density" solution as the mesh becomes finer. We also compare the two theories for axial assembly swelling in an abbreviated FFTF benchmark with a coarse mesh. Here we find that reactivity coefficients obtained via "virtual density" perturbation theory agree with reference solutions to less than 0.1%, while those obtained via boundary perturbation theory exhibit sporadic accuracy - sometimes in the range of 1-5% error, more frequently in the range 5-20% error, and occasionally well over 100% error in control rod assemblies. We conclude that although "virtual density" perturbation theory and boundary perturbation theory are analytically equivalent, boundary perturbations are often thwarted in coarse mesh finite difference solutions due to inaccurate flux gradients along mesh cell surfaces in heterogeneous cores.

8.2 Introduction

Now that we have numerically validated non-uniform anisotropic "virtual density" theory against "virtual mesh" diffusion reference cases, we seek to ascertain how well"virtual density" theory performs relative to traditional boundary perturbation theory. Showing that our new theory works is nice, but showing how it compares to previous methods could "seal the deal".

8.3 Analytic Comparison for Uniform Isotropic Expansions

First, we can easily show that "virtual density" perturbation theory and boundary perturbation theory are equivalent for uniform isotropic expansions of 1-D slabs and spheres. Here we use the traditional boundary perturbation theory as derived by Pomraning, Larsen, and Rahnema in the early 1980s [42,43,45]. For simplicity, we assume zero-flux vacuum boundary conditions.

Pomraning and Larsen derived the following one-group expression for the reactivity due to an external boundary perturbation.

$$\Delta\rho = \frac{\int dS|\vec{r'} - \vec{r}|D\left(\hat{n}\cdot\nabla\phi(\vec{r})\right)^2}{\int dV\nu\Sigma_f\phi(\vec{r})^2} \tag{8.1}$$

Here $\vec{r}$ and $\vec{r'}$ are vectors pointing to the unperturbed and perturbed surfaces, respectively. $\hat{n}$ is the unit normal vector of the unperturbed surface S.

8.3.1 1-D Slab Expansion

Let us return to the infinite homogenous 1-D slab of thickness a that we studied in Section 3.5.1. Here the domain is $-a/2 \leq x \leq a/2$. Let the perturbed thickness be $a + 2\alpha$ so that

$$\frac{a'}{2} = \frac{a}{2} + \alpha \tag{8.2}$$

The equivalent expansion fraction f is

$$f = 1 + \frac{\alpha}{a/2} \tag{8.3}$$

We can usefully express the relationship between f and α as a derivative.

$$\frac{df}{d\alpha} = \frac{2}{a} \tag{8.4}$$

Now the simplified Pomraning expression for the first order reactivity in one group is

$$\Delta\rho = \frac{\int dS\alpha D\left(\nabla\phi(\vec{r})\right)^2}{\int dV\nu\Sigma_f\phi(\vec{r})^2} \tag{8.5}$$

This expression assumes that the surface perturbation vector $\vec{r'} - \vec{r} = \alpha$ is always perpendicular to the normal unit vector $\hat{n}$ of surface S. Now expressing the first derivative of the eigenvalue with respect to the perturbation magnitude α is simple:

$$\frac{d(\Delta\rho)}{d\alpha} = \frac{\int dSD\left(\nabla\phi(\vec{r})\right)^2}{\int dV\nu\Sigma_f\phi(\vec{r})^2} \tag{8.6}$$

Considering our particular 1-D infinite slab geometry and evaluating this derivative at α = 0 yields

$$\left[\frac{d(\Delta\rho)}{d\alpha}\right]_{\alpha=0} = \frac{2\left[D\left(\nabla\phi(x)\right)^2\right]_{x=a/2}}{\int_{a/2}^{a'/2} dx\, \nu\Sigma_f \phi(x)^2} \tag{8.7}$$

The factor of 2 here simply accounts for the fact that the infinite 1-D slab has two surfaces, each perturbed by the same distance α. Performing the integral leaves

$$\left[\frac{d(\Delta\rho)}{d\alpha}\right]_{\alpha=0} = \frac{2D(\pi/a)^2}{\nu\Sigma_f(a/2)} \tag{8.8}$$

Now in order to compare with our "virtual density" results, we must evaluate the derivative with respect to f. We can do this by multiplying our result by the derivative $d\alpha/df$.

$$\left[\frac{d(\Delta\rho)}{df}\right]_{f=1} = \frac{d\alpha}{df}\left[\frac{d(\Delta\rho)}{d\alpha}\right]_{\alpha=0} = \frac{2D}{\nu\Sigma_f}\left(\frac{\pi}{a}\right)^2 \tag{8.9}$$

This result is precisely consistent with our "virtual density" result shown in Equation 3.54.

8.3.2 1-D Sphere Expansion

Now let us perform the same comparative analysis for the case of a homogenous sphere bounded by vacuum. We previously studied this in Section 3.5.2. We perturb its radius by a distance α.

$$R' = R + \alpha \tag{8.10}$$

The corresponding expansion fraction we use in "virtual density" theory is

$$f = 1 + \frac{\alpha}{R} \tag{8.11}$$

The differential relationship between f and α is

$$\frac{df}{d\alpha} = \frac{1}{R} \tag{8.12}$$

Now the reactivity per unit α is

$$\left[\frac{d(\Delta\rho)}{d\alpha}\right]_{\alpha=0} = \frac{\left[D\left(\nabla\phi(r)\right)^2\right]_{r=R}}{\int_0^R 4\pi r^2 dr\, \nu\Sigma_f\left(\phi(r)\right)^2} = \frac{4\pi R^2 D\left(\frac{\pi^2}{R^4}\right)}{4\pi\nu\Sigma_f\frac{R}{2}} \tag{8.13}$$

The reactivity per unit f is

$$\left[\frac{d(\Delta\rho)}{df}\right]_{f=1} = \frac{d\alpha}{df}\left[\frac{d(\Delta\rho)}{d\alpha}\right]_{\alpha=0} = \frac{2D}{\nu\Sigma_f}\left(\frac{\pi}{R}\right)^2 \tag{8.14}$$

Again, this result is precisely consistent with our "virtual density" theory result in Equation 3.68.

8.4 Analytic Equivalence for Non-Uniform Isotropic Expansions in 1-D

We have shown that "virtual density" perturbation theory is equivalent to boundary perturbation theory for the two very simple cases of uniform isotropic expansion of homogeneous slabs and spheres. Now we will mathematically prove this equivalence for any geometry change in any arbitrary 1-D Cartesian geometry.

Consider the single internal boundary shift in an array of 1-D slabs. As illustrated in Figure 8.1, an internal boundary shifts slightly to the right by a distance Δx. The "virtual density" theory transforms this boundary shift into two material density changes: a factor of $-\Delta x/a$ density change in the right slab and a factor of $\Delta x/a$ density change in the left slab. In one group, we can express the uniform isotropic "virtual density" perturbation numerator as

$$\Delta\rho\left\langle\phi^\dagger\middle|\hat{F}\phi\right\rangle = \tfrac{\Delta x}{a}\int_{-a}^{0}dx\ \left[\phi^\dagger(x)\left(\tfrac{1}{k}\nu\Sigma_f - \Sigma_a\right)_L\phi(x) + D_L\nabla\phi^\dagger(x)\cdot\nabla\phi(x)\right] \tag{8.15}$$

$$-\tfrac{\Delta x}{a}\int_{0}^{a}dx\ \left[\phi^\dagger(x)\left(\tfrac{1}{k}\nu\Sigma_f - \Sigma_a\right)_R\phi(x) + D_R\nabla\phi^\dagger(x)\cdot\nabla\phi(x)\right] \tag{8.16}$$

Here the subscripts L and R denote material properties in the left and right slabs, respectively. Although there is obviously no fission in the sodium slab, we leave the fission term there for completeness. These expressions are precisely equivalent to the isotropic S and L quantities defined in Section 3.4.3.

However, if we consider the conceptual underpinnings of the "virtual density" theory - the basic reasoning we employed to conceive it - there is no reason why we must alter material densities throughout the whole two slabs. Instead, we could alter material densities in only half of each slab - the half adjacent to the shifting boundary. Now the perturbation numerator will look like this:

$$\Delta\rho\left\langle\phi^\dagger\middle|\hat{F}\phi\right\rangle = \tfrac{2\Delta x}{a}\int_{-a/2}^{0}dx\ \left[\phi^\dagger(x)\left(\tfrac{1}{k}\nu\Sigma_f - \Sigma_a\right)_L\phi(x) + D_L\nabla\phi^\dagger(x)\cdot\nabla\phi(x)\right] \tag{8.17}$$

$$-\tfrac{2\Delta x}{a}\int_{0}^{a/2}dx\ \left[\phi^\dagger(x)\left(\tfrac{1}{k}\nu\Sigma_f - \Sigma_a\right)_R\phi(x) + D_R\nabla\phi^\dagger(x)\cdot\nabla\phi(x)\right] \tag{8.18}$$

Believe it or not, Equations 8.17 and 8.18 are *precisely* equivalent to Equations 8.15 and 8.16. No approximation is involved. We can even take this further - if we can integrate over a half slab instead of a whole slab, why can we not integrate over only a very small slice of the slab immediately adjacent to the shifting boundary? In fact, we can. We now choose to integrate over a small slice in each slab with thickness Δx, the same thickness as the volume displaced by the shifting surface.

$$\Delta\rho\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle = \int_{-\Delta x}^{0} dx\, \left[\phi^\dagger(x)\left(\tfrac{1}{k}\nu\Sigma_f - \Sigma_a\right)_L \phi(x) + D_L \nabla\phi^\dagger(x)\cdot\nabla\phi(x)\right] \tag{8.19}$$

$$- \int_{0}^{\Delta x} dx\, \left[\phi^\dagger(x)\left(\tfrac{1}{k}\nu\Sigma_f - \Sigma_a\right)_R \phi(x) + D_R \nabla\phi^\dagger(x)\cdot\nabla\phi(x)\right] \tag{8.20}$$

Again, Equations 8.19 and 8.20 are *precisely* equivalent to Equations 8.15 and 8.16.

Now, assuming that $\Delta x << a$, we can transform each of the volume integrals in Equations 8.19 and 8.20 into a surface integral over the unperturbed surface S at $x = 0$. This is the same volume-to-surface transformation performed by Rahnema and Pomraning when describing the diffusion perturbation "anomaly" [43]. Note that we now add L and R subscripts to $\nabla\phi$, because flux gradients are discontinuous across material discontinuities.

$$\Delta\rho\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle = \Delta x \int_S dS \left[\phi^\dagger(0)\left(\tfrac{1}{k}\nu\Sigma_f - \Sigma_a\right)_L \phi(0) + D_L \nabla\phi_L^\dagger(0)\cdot\nabla\phi_L(0)\right] \tag{8.21}$$

$$- \Delta x \int_S dS \left[\phi^\dagger(0)\left(\tfrac{1}{k}\nu\Sigma_f - \Sigma_a\right)_R \phi(0) + D_R \nabla\phi_R^\dagger(0)\cdot\nabla\phi_R(0)\right] \tag{8.22}$$

Now we notice that both integrals (for R and L quantities) have the same domain - the unperturbed surface. Thus, we can easily combine them into a single surface integral. We now denote $\phi(0)$ as simply ϕ.

$$\Delta x \int_S dS \left[\phi^\dagger \left(\left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_L - \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_R\right)\phi + D_L\nabla\phi_L^\dagger\cdot\nabla\phi_L - D_R\nabla\phi_R^\dagger\cdot\nabla\phi_R\right] \tag{8.23}$$

Now consider the difference in leakage terms in Equation 8.23. We know that the real and adjoint currents are conserved across the surface S due to Fick's Law:

$$- D_L\nabla\phi_L = -D_R\nabla\phi_R \tag{8.24}$$

$$D_L\nabla\phi_L^\dagger = D_R\nabla\phi_R^\dagger \tag{8.25}$$

Now we can cleverly apply Equations 8.24 and 8.25 to the leakage difference in Equation 8.23 to obtain

$$\Delta x \int_S dS \left[\phi^\dagger \left(\left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_L - \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_R\right)\phi + \left(D_R - D_L\right)\nabla\phi_R^\dagger\cdot\nabla\phi_L\right] \tag{8.26}$$

Note that we could alternatively (and equivalently) express the leakage term in Equation 8.26 using the left adjoint flux and the right real flux:

$$\nabla\phi_R^\dagger \cdot \nabla\phi_L = \nabla\phi_L^\dagger \cdot \nabla\phi_R \tag{8.27}$$

Equation 8.26 is equivalent to Equation 2.43, which Rahnema and Pomraning derived to correct for the diffusion perturbation "anomaly" [43]. Thus, we have successfully re-derived the diffusion internal boundary perturbation proposed by Rahnema and Pomraning from scratch based on different fundamental principles.

Rahnema and Pomraning describe this "anomaly" as an unexpected failure of the usual first order diffusion perturbation formula for internal boundary shifts. Boundary shifts are an order 1 change in cross-sections over an order $\epsilon << 1$ volume, while density changes are an order $\epsilon << 1$ change in cross-sections over an order 1 volume. Pomraning shows that the "classic" perturbation formula fails and re-derives the correct revised formula [45]. However, we have shown here that no such "anomaly" occurs via the "virtual density" theory - the classic first order perturbation formula is correct if we only define the perturbation as a "virtual" density change rather than as a boundary shift. Thus, this "anomaly" is really not so anomalous.

Although we have only mathematically proven this for a single boundary shift, it is obvious that 1-D geometry perturbation can be decomposed into individual boundary shifts. Thus, we have shown that "virtual density" perturbation theory is equivalent to traditional boundary perturbation theory for any arbitrary geometry perturbation in 1-D problems.

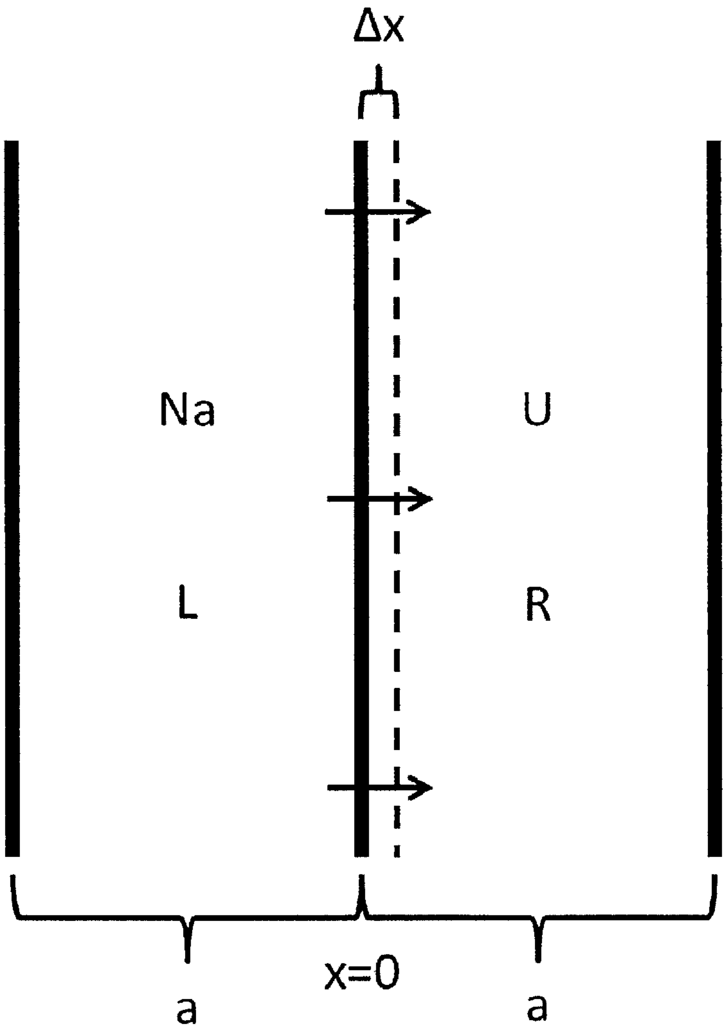

Figure 8.1: An illustration of a single internal boundary shift. The boundary shifts rightward by a small distance Δx so that fuel mass is lost. Each slab has unperturbed width a. Let $x = 0$ correspond to the position of the unperturbed boundary. Quantities in the right slab are denoted by the subscript R, while quantities in left slab are denoted by the subscript L.

8.5 Analytic Equivalence for Non-Uniform Anisotropic Expansions in 2-D

In the previous section, we proved the analytic equivalence of isotropic "virtual density" and traditional boundary perturbation theory in 1-D slab geometry. Here we will walk through a similar proof for a boundary shift in 2-D Cartesian geometry. Although this is all one-group, extension to multigroup is trivial. Multigroup first order perturbations are always the simple sum of perturbations in each group.

As illustrated in Figure 8.2, an internal boundary (between two mesh cells) shifts slightly to the right by a distance Δx. The "virtual density" theory transforms this boundary shift into two material density changes: a factor of $-\Delta x/a$ density change in the right slab and a factor of $\Delta x/a$ density change in the left slab. In shorthand notation, the "virtual density" eigenvalue perturbation is

$$\Delta\rho = \frac{\Delta x}{a}\left[S + L_x - L_y + \Gamma_y\right]_L - \frac{\Delta x}{b}\left[S + L_x - L_y + \Gamma_y\right]_R \tag{8.28}$$

Here we have chosen to use Equation 5.33, because it contains the spectral reactivity S with a "+1" coefficient.

In one group, we can express the full spatial integrals in the perturbation numerators of this shorthand notation as

$$\Delta\rho\left\langle\phi^\dagger\middle|\,\hat{F}\phi\right\rangle = \tag{8.29}$$

$$\frac{\Delta x}{a}\int_{-a}^{0} dx \int_{0}^{c} dy \left[\phi^\dagger(x,y)\left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_L \phi(x,y)\right] \tag{8.30}$$

$$+\frac{\Delta x}{a}\int_{-a}^{0} dx \int_{0}^{c} dy \left[D_L\left[\nabla\phi^\dagger(x,y)\right]_x\left[\nabla\phi(x,y)\right]_x - D_L\left[\nabla\phi^\dagger(x,y)\right]_y\left[\nabla\phi(x,y)\right]_y\right] \tag{8.31}$$

$$+\frac{\Delta x}{a}\int_{-a}^{0} dx \left[D_L\phi^\dagger(x,c)\left[\nabla\phi(x,c)\right]_y - D_L\phi^\dagger(x,0)\left[\nabla\phi(x,0)\right]_y\right] \tag{8.32}$$

$$-\frac{\Delta x}{b}\int_{0}^{b} dx \int_{0}^{c} dy \left[\phi^\dagger(x,y)\left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_R \phi(x,y)\right] \tag{8.33}$$

$$-\frac{\Delta x}{b}\int_{0}^{b} dx \int_{0}^{c} dy \left[D_R\left[\nabla\phi^\dagger(x,y)\right]_x\left[\nabla\phi(x,y)\right]_x - D_R\left[\nabla\phi^\dagger(x,y)\right]_y\left[\nabla\phi(x,y)\right]_y\right] \tag{8.34}$$

$$-\frac{\Delta x}{b}\int_{0}^{b} dx \left[D_R\phi^\dagger(x,c)\left[\nabla\phi(x,c)\right]_y - D_R\phi^\dagger(x,0)\left[\nabla\phi(x,0)\right]_y\right] \tag{8.35}$$

Here the subscripts L and R denote material properties in the left and right slabs, respectively. The subscripts x and y denote the x and y components of the real or adjoint flux gradients. The S, L_x, and L_y quantities are volume integrals, while the Γ_y quantities are surface integrals.

However, if we consider the conceptual underpinnings of the "virtual density" theory - the basic reasoning we employed to conceive it - there is no reason why we must alter material densities throughout the whole two slabs. Instead, we could alter material densities in only half of each slab - the half adjacent to the shifting boundary. Now the same perturbation numerator looks like this:

$$\frac{2\Delta x}{a}\int_{-a/2}^{0} dx \int_{0}^{c} dy \left[\phi^{\dagger}(x,y)\left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_L \phi(x,y)\right] \tag{8.36}$$

$$+\frac{2\Delta x}{a}\int_{-a/2}^{0} dx \int_{0}^{c} dy \left[D_L\left[\nabla\phi^{\dagger}(x,y)\right]_x\left[\nabla\phi(x,y)\right]_x - D_L\left[\nabla\phi^{\dagger}(x,y)\right]_y\left[\nabla\phi(x,y)\right]_y\right] \tag{8.37}$$

$$+\frac{2\Delta x}{a}\int_{-a/2}^{0} dx \left[D_L\phi^{\dagger}(x,c)\left[\nabla\phi(x,c)\right]_y - D_L\phi^{\dagger}(x,0)\left[\nabla\phi(x,0)\right]_y\right] \tag{8.38}$$

$$-\frac{2\Delta x}{b}\int_{0}^{b/2} dx \int_{0}^{c} dy \left[\phi^{\dagger}(x,y)\left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_R \phi(x,y)\right] \tag{8.39}$$

$$-\frac{2\Delta x}{b}\int_{0}^{b/2} dx \int_{0}^{c} dy \left[D_R\left[\nabla\phi^{\dagger}(x,y)\right]_x\left[\nabla\phi(x,y)\right]_x - D_R\left[\nabla\phi^{\dagger}(x,y)\right]_y\left[\nabla\phi(x,y)\right]_y\right] \tag{8.40}$$

$$-\frac{2\Delta x}{b}\int_{0}^{b/2} dx \left[D_R\phi^{\dagger}(x,c)\left[\nabla\phi(x,c)\right]_y - D_R\phi^{\dagger}(x,0)\left[\nabla\phi(x,0)\right]_y\right] \tag{8.41}$$

Believe it or not, the expressions in Equations 8.36 - 8.41 are *precisely* equivalent to those in Equations 8.30 - 8.35. No approximation is involved. We can even take this further - if we can integrate over a half slab instead of a whole slab, why can we not integrate over only a very small slice of the slab immediately adjacent to the shifting boundary? In fact, we can. We now choose to integrate over a small slice in each slab with thickness Δx, the same thickness as the volume displaced by the shifting surface.

$$\int_{-\Delta x}^{0} dx \int_{0}^{c} dy \left[\phi^\dagger(x,y)\left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_L \phi(x,y)\right] \tag{8.42}$$

$$+\int_{-\Delta x}^{0} dx \int_{0}^{c} dy \left[D_L \left[\nabla\phi^\dagger(x,y)\right]_x \left[\nabla\phi(x,y)\right]_x - D_L \left[\nabla\phi^\dagger(x,y)\right]_y \left[\nabla\phi(x,y)\right]_y\right] \tag{8.43}$$

$$+\int_{-\Delta x}^{0} dx \left[D_L\phi^\dagger(x,c)\left[\nabla\phi(x,c)\right]_y - D_L\phi^\dagger(x,0)\left[\nabla\phi(x,0)\right]_y\right] \tag{8.44}$$

$$-\int_{0}^{\Delta x} dx \int_{0}^{c} dy \left[\phi^\dagger(x,y)\left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_R \phi(x,y)\right] \tag{8.45}$$

$$-\int_{0}^{\Delta x} dx \int_{0}^{c} dy \left[D_R \left[\nabla\phi^\dagger(x,y)\right]_x \left[\nabla\phi(x,y)\right]_x - D_R \left[\nabla\phi^\dagger(x,y)\right]_y \left[\nabla\phi(x,y)\right]_y\right] \tag{8.46}$$

$$-\int_{0}^{\Delta x} dx \left[D_R\phi^\dagger(x,c)\left[\nabla\phi(x,c)\right]_y - D_R\phi^\dagger(x,0)\left[\nabla\phi(x,0)\right]_y\right] \tag{8.47}$$

Again, Equations 8.42 - 8.47 are *precisely* equivalent to Equations 8.30 - 8.35. No approximation is made.

Now, assuming that Δx is extremely small, we can transform each of the volume integrals in Equations 8.42 - 8.47 into surface integrals over the unperturbed surface $\vec{S}$ at $x = 0$. This is the same volume-to-surface transformation performed by Rahnema and Pomraning when describing the diffusion perturbation "anomaly" [43]. Note that we now add L and R subscripts to $\nabla\phi$, because flux gradients are discontinuous across material discontinuities. Also, we denote $\phi(0)$ as simply ϕ. The surface $\vec{S}$ is vertical, and so the dot product $d\vec{S}\cdot\hat{x}$ will have a magnitude of 1.

$$\Delta x \int_S \left(d\vec{S}\cdot\hat{x}\right) \left[\phi^\dagger \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_L \phi\right] \tag{8.48}$$

$$+\Delta x \int_S \left(d\vec{S}\cdot\hat{x}\right) \left[D_L \left(\nabla\phi^\dagger\right)_{x,L} \left(\nabla\phi\right)_{x,L} - D_L \left(\nabla\phi^\dagger\right)_{y,L} \left(\nabla\phi\right)_{y,L}\right] \tag{8.49}$$

$$-\Delta x \int_S \left(d\vec{S}\cdot\hat{x}\right) \left[\phi^\dagger \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_R \phi\right] \tag{8.50}$$

$$-\Delta x \int_S \left(d\vec{S}\cdot\hat{x}\right) \left[D_R \left(\nabla\phi^\dagger\right)_{x,R} \left(\nabla\phi\right)_{x,R} - D_R \left(\nabla\phi^\dagger\right)_{y,R} \left(\nabla\phi\right)_{y,R}\right] \tag{8.51}$$

Note that the surface integral Γ_y terms have disappeared. When we converted the volume integrals to surface integrals, we also (implicitly) converted the surface integrals to *zero.* The Γ_y terms are only non-zero at a *single point* on the surface $\vec{S}$, and that single point has zero area. Thus, the integral of Γ_y over the surface area is zero.

Now we notice that the right-hand and left-hand integrals both have the same domain - the unperturbed surface. Thus, we can easily combine them into a single surface integral.

$$\Delta x \int_S \left(d\vec{S} \cdot \hat{x}\right) \left[\phi^\dagger \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_L \phi - \phi^\dagger \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_R \phi\right] \tag{8.52}$$

$$+\Delta x \int_S \left(d\vec{S} \cdot \hat{x}\right) \left[D_L \left(\nabla\phi^\dagger\right)_{x,L} (\nabla\phi)_{x,L} - D_R \left(\nabla\phi^\dagger\right)_{x,R} (\nabla\phi)_{x,R}\right] \tag{8.53}$$

$$-\Delta x \int_S \left(d\vec{S} \cdot \hat{x}\right) \left[D_L \left(\nabla\phi^\dagger\right)_{y,L} (\nabla\phi)_{y,L} - D_R \left(\nabla\phi^\dagger\right)_{y,R} (\nabla\phi)_{y,R}\right] \tag{8.54}$$

Now consider the difference in leakage terms in Equations 8.53 and 8.54. We know that the real and adjoint currents perpendicular to the surface $\vec{S}$ are conserved across $\vec{S}$ due to Fick's Law:

$$-D_L \nabla\phi_{x,L} = -D_R \nabla\phi_{x,R} \tag{8.55}$$

$$D_L \nabla\phi^\dagger_{x,L} = D_R \nabla\phi^\dagger_{x,R} \tag{8.56}$$

We also know that the the real and adjoint gradients *parallel* to the surface $\vec{S}$ must be equal on either side of $\vec{S}$. Were these gradients not equal, the flux magnitude would not be continuous along $\vec{S}$.

$$\nabla\phi_{y,L} = \nabla\phi_{y,R} \tag{8.57}$$

$$\nabla\phi^\dagger_{y,L} = \nabla\phi^\dagger_{y,R} \tag{8.58}$$

Now we can cleverly apply Equations 8.55 - 8.58 to the leakage differences in Equations 8.53 and 8.54 to obtain

$$\Delta x \int_S \left(d\vec{S} \cdot \hat{x}\right) \left[\phi^\dagger \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_L \phi - \phi^\dagger \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_R \phi\right] \tag{8.59}$$

$$+\,\Delta x \int_S \left(d\vec{S} \cdot \hat{x}\right) \left[(D_R - D_L) \left(\nabla\phi^\dagger\right)_{x,R} (\nabla\phi)_{x,L} + (D_R - D_L) \left(\nabla\phi^\dagger\right)_{y,R} (\nabla\phi)_{y,L}\right] \tag{8.60}$$

Now the leakage terms in Equation 8.60 are nothing but $D_R - D_L$ multiplied by the dot product of the real and adjoint flux gradients! So now the full surface integral simplifies to

$$\Delta x \int_S \left(d\vec{S} \cdot \hat{x}\right) \left[\phi^\dagger \left(\left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_L - \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_R\right) \phi + (D_R - D_L) \nabla\phi^\dagger_R \cdot \nabla\phi_L\right] \tag{8.61}$$

Note that we could alternatively (and equivalently) express the leakage term in Equation 8.61 using the left adjoint flux and the right real flux:

$$\nabla\phi_R^\dagger \cdot \nabla\phi_L = \nabla\phi_L^\dagger \cdot \nabla\phi_R \tag{8.62}$$

Equation 8.61 is equivalent to Equation 2.43, which Rahnema and Pomraning derived to correct for the diffusion perturbation "anomaly" [43]. Thus, we have successfully rederived the diffusion internal boundary perturbation proposed by Rahnema and Pomraning from scratch based on different fundamental principles. We did this in Section 8.4 for the isotropic case, and now we have done the same for the fully general anisotropic case.

Although we have only mathematically proven this for a single 2-D boundary shift, any 2-D geometry perturbation can (obviously) be decomposed into individual 2-D boundary shifts. Figure 8.2 could represent two adjacent mesh cells within a core, and so any 2-D perturbation can be expressed as a superposition of many mesh cells behaving similarly.

Thus, we have shown that "virtual density" perturbation theory is equivalent to traditional boundary perturbation theory for any arbitrary geometry perturbation in 2-D problems. Extension to 3-D is trivial given what we have shown in this section - it merely requires the addition of L_z and Γ_z terms, which one could carry through the proof in precisely the same manner as L_y and Γ_y.

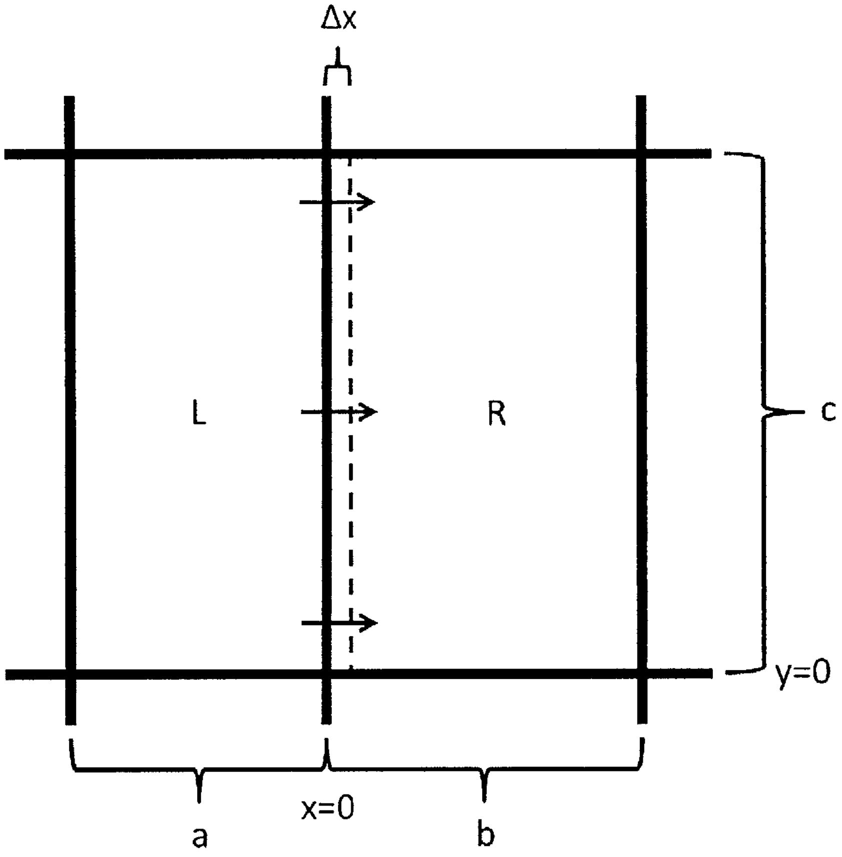

Figure 8.2: An illustration of a single internal boundary shift in Cartesian 2D geometry. The boundary shifts rightward by a small distance Δx without changing material densities, so mass is *not* conserved. The region to the left has unperturbed width a, while the region to the right has unperturbed width b. Both regions have the same height c. Let $x = 0$ correspond to the position of the unperturbed boundary. Quantities in the right slab are denoted by the subscript R, while quantities in left slab are denoted by the subscript L.

8.6 Numeric Comparison for Arbitrary 1-D Perturbations

Now that we have proven the *analytic* equivalence of "virtual density" perturbation theory and boundary perturbation theory, we will compare them *numerically.* Since a proper comparison will require a range of coarse to very fine meshes, we begin with a 1-D slab problem that can be solved very quickly.

Any 1-D Cartesian problem can be modeled as a series of infinite 1-D slabs. Even though shifting one slab is often a very non-uniform and asymmetric distortion, we can still apply the isotropic "virtual density" theory to it. When an infinite 1-D slab is expanded or contracted, it remains an infinite 1-D slab - it retains its "shape". Thus, in this section, we demonstrate that the isotropic "virtual density" theory is applicable to *any* arbitrary geometry perturbation in 1-D problems.

Consider the test geometry shown in Figure 8.3. Six uranium metal slabs interspersed with seven sodium slabs comprise a 1-D reactor model. In this model, we can define *any* geometry perturbation in either of two paradigms: (1) as shifting inter-slab boundaries or (2) as expansions or contractions of slabs. Paradigm (1) corresponds to traditional boundary perturbation theory, while paradigm (2) corresponds to "virtual density" perturbation theory.

We model this 1-D test geometry in MaPS. We employ mesh-centered finite difference, and we can vary the number of mesh cells per slab. We always use a strict zero-flux boundary condition without extrapolation. Figure 8.4 shows the spatial distribution of the real and adjoint fluxes in this 1-D test geometry at four different energies. Refer back to Figure 2.1 for the real and adjoint flux spectra within the central sodium slab.

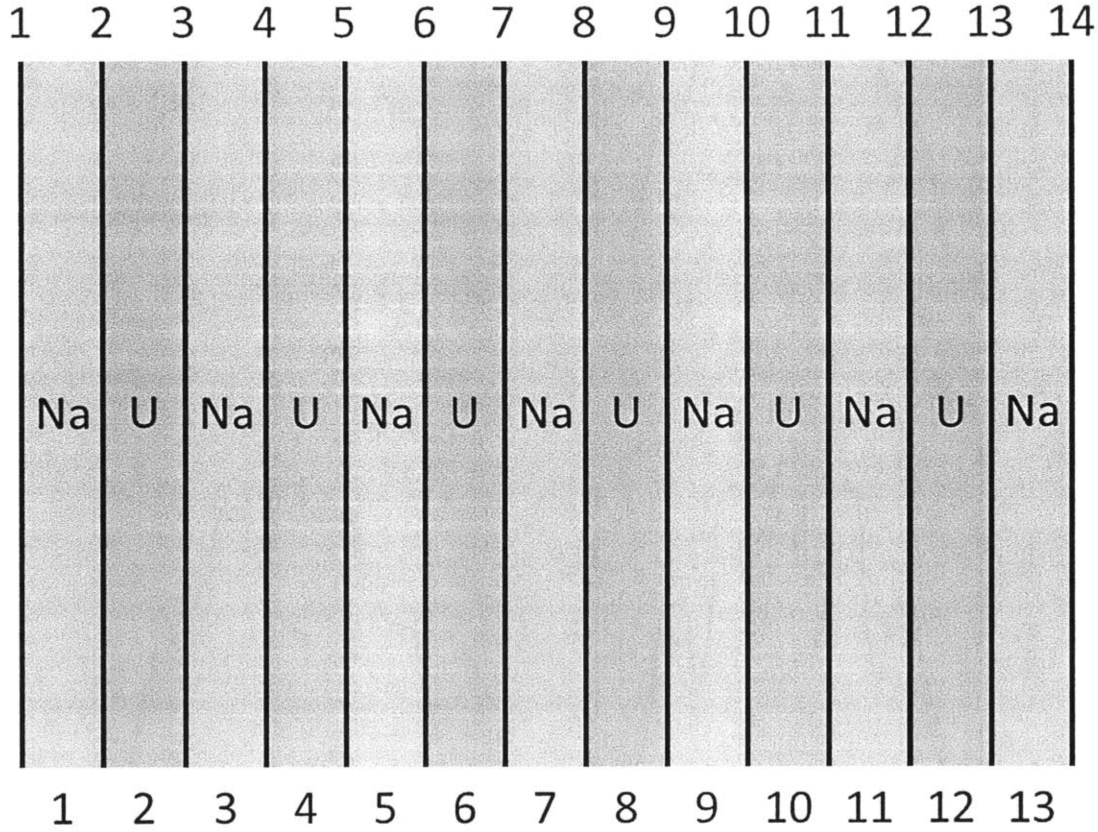

Figure 8.3: Six uranium metal slabs interspersed with seven pure sodium slabs. The uranium is enriched to 11%. All slabs are 6 cm thick with vacuum boundary conditions on the outer sodium slabs. We have numbered the slabs 1-13 and the surfaces 1-14.

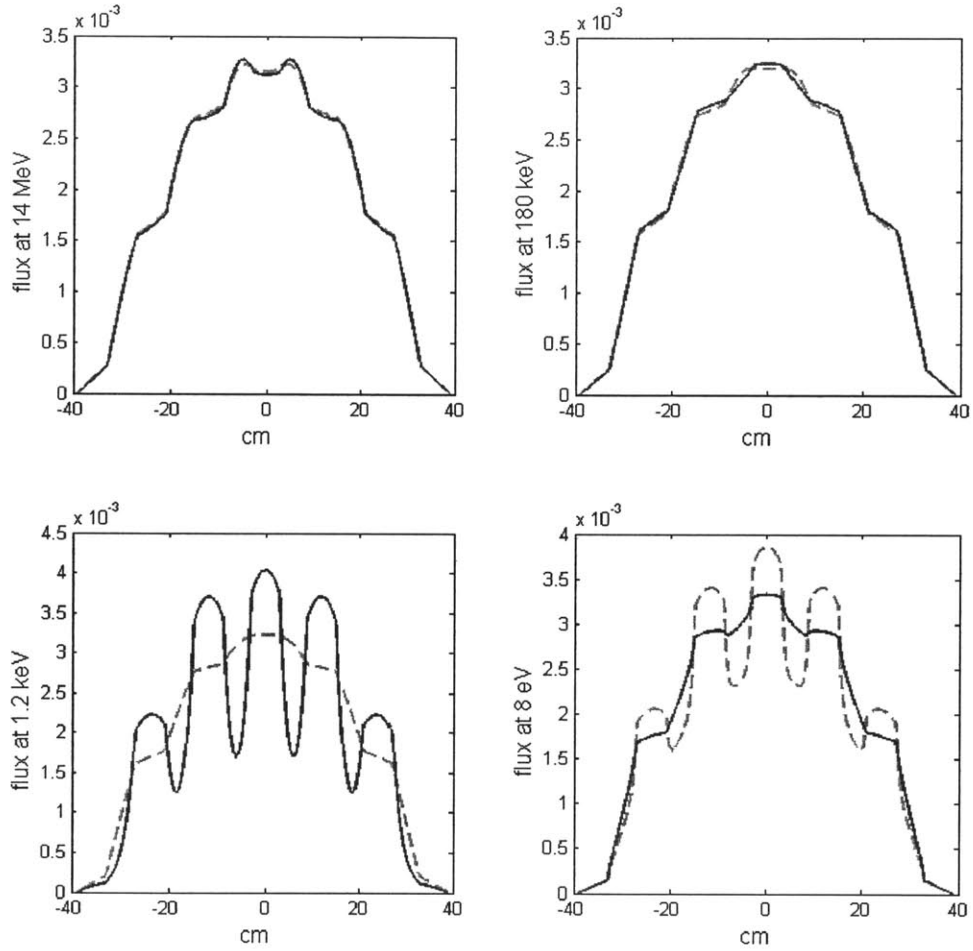

Figure 8.4: Spatial distributions of the real and adjoint fluxes in the alternating uranium-sodium slab geometry shown in Figure 8.3. At high energies, both real and adjoint fluxes tend to be spatially smooth. In the keV range, the real flux peaks sharply in the sodium slabs, because neutrons are strongly absorbed in the uranium slabs. In the eV range, very few neutrons exist. However, the adjoint flux peaks sharply in the sodium slabs, because neutrons are more "important" when they scatter down to lower energies.

8.6.1 Evaluation of Surface Currents

Pomraning and Rahnema formulated traditional boundary perturbation theory for internal boundary shifts in diffusion theory, correcting a curious "anomaly" that has no analog in transport theory [43].

$$\Delta\rho = \frac{\int dS|\vec{r'} - \vec{r}| \left[(D_R - D_L) \nabla\phi_R^\dagger \cdot \nabla\phi_L + \phi^\dagger \left(\left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_L - \left(\frac{1}{k}\nu\Sigma_f - \Sigma_a\right)_R \right) \phi \right]}{\int dV \phi^\dagger \nu\Sigma_f \phi} \tag{8.63}$$

This applies to an internal interface shift *to the right.* The scalar $|\vec{r'} - \vec{r}|$ is the distance between the unperturbed surface S and the perturbed surface. The subscripts R and L denote quantities evaluated immediately to the right or left of the boundary (the unperturbed surface S). Note that although the flux gradient is discontinuous across the interface, the flux itself is always continuous. Although this expression is only one-group, extension to multigroup requires nothing more than summing over all energy groups and adding scattering terms.

Equation 8.63 requires precise knowledge of the real and adjoint flux gradients to the right and left of each inter-slab boundary. Although finite difference solutions produce volume-averaged fluxes, they also provide implicit knowledge of surface fluxes. Thus, we can estimate surface fluxes and flux gradients on either side of a boundary with some simple math.

As we are implementing methods developed by others, it is crucial to define our implementation precisely. We have found no finite difference implementation of Equation 8.63 in the literature. We strive to implement Equation 8.63 in the most accurate way possible, but others may propose different implementations.

See Figure 8.5 for an illustration. Let ϕ_R denote the flux in the mesh cell immediately to the right of the boundary, and let ϕ_L denote the flux in the mesh cell immediately to the left of that same boundary. Let ϕ_0 represent the flux magnitude directly on the boundary. The right mesh cell has thickness Δx_R and diffusion coefficient D_R, while the left mesh cell has thickness Δx_L and diffusion coefficient D_L.

We can begin by expressing the right and left flux gradients as simple finite differences including ϕ_0.

$$\nabla\phi_L = \frac{\phi_0 - \phi_L}{\Delta x_L/2} \tag{8.64}$$

$$\nabla\phi_R = \frac{\phi_R - \phi_0}{\Delta x_R/2} \tag{8.65}$$

Now let us conserve current across the boundary and then substitute in the finite differences above.

$$D_L \nabla \phi_L = D_R \nabla \phi_R \tag{8.66}$$

$$D_L \frac{\phi_0 - \phi_L}{\Delta x_L / 2} = D_R \frac{\phi_R - \phi_0}{\Delta x_R / 2} \tag{8.67}$$

Now solving for ϕ_0 is trivial:

$$\phi_0 = \frac{\frac{D_L}{\Delta x_L}\phi_L + \frac{D_R}{\Delta x_R}\phi_R}{\frac{D_L}{\Delta x_L} + \frac{D_R}{\Delta x_R}} \tag{8.68}$$

Once we know ϕ_0, then we may return to Equations 8.64 and 8.65 to obtain $\nabla\phi_R$ and $\nabla\phi_L$. We can then repeat the identical process to obtain $\nabla\phi_R^\dagger$ and $\nabla\phi_L^\dagger$.

As for external boundary shifts, Larsen and Pomraning derived a similar expression for external reactor boundaries [42].

$$\Delta\rho = \frac{\int dS |\vec{r'} - \vec{r}| D \left(\hat{n} \cdot \nabla\phi^\dagger\right) \left(\hat{n} \cdot \nabla\phi\right)}{\int dV \phi^\dagger \nu \Sigma_f \phi} \tag{8.69}$$

Since we have chosen a strict zero-flux boundary condition, we can express $\nabla\phi$ at the outer boundary as $-\phi/(\Delta x/2)$, where ϕ is the flux in the mesh cell adjacent to the boundary and Δx is the width of that mesh cell.

These difference techniques will not provide perfect surface currents, but they are perhaps the best one can do with finite difference solutions.

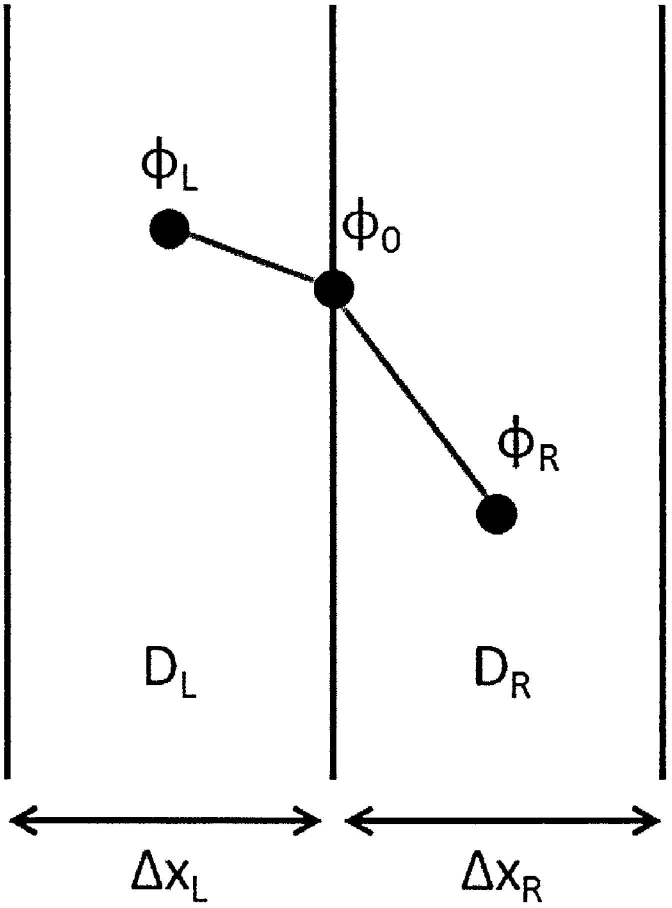

Figure 8.5: Estimating flux gradients on either side of a boundary from a mesh-centered finite difference solution. Let ϕ_R and ϕ_R denote the fluxes in the mesh cells immediately to the right and left of the boundary. ϕ_0 represents the flux directly on the boundary. Δx_R, D_R, Δx_L, and D_L are the thicknesses and diffusion coefficients in the right and left mesh cells.

8.6.2 Single Internal Boundary Shifting

Now let us compare "virtual density" perturbation theory and traditional boundary perturbation theory. We begin with the simplest of all geometry perturbations - shifting a single internal boundary slightly to the right. We shall arbitrarily choose surface 8 in Figure 8.3. We will not change the densities of slabs 7 or 8, so surface 8's movement to the right will decrease the mass of uranium and increase the mass of sodium. Thus, we should expect a large decrease in reactivity.

In traditional boundary perturbation theory, we can simply apply Equation 8.63 to surface 8 to obtain the first order reactivity. This is a surface integral over surface 8, which is equivalent to a volume integral in the small region between the unperturbed surface 8 and the perturbed surface 8. This corresponds to the "actual" movement of surface 8.

In "virtual density" perturbation theory, we perform no surface integrals at all. Instead of actually moving surface 8, we *increase* the neutron mean free path in slab 8 and *decrease* the neutron mean free path in slab 7. This is equivalent to *decreasing* the uranium density in slab 8 and *increasing* the sodium density in slab 7.

Figure 8.6 compares these two approaches to the exact reactivity. Here the "boundary shift fraction" is equal to dx/a, where dx is the distance by which the boundary shifts and a is the slab width. As expected, this surface shift causes a large decrease in reactivity, primarily due to loss of fuel mass.

Figure 8.6 gives the impression that boundary perturbation theory is more accurate than "virtual density" perturbation theory in this case, but that impression is actually false. Figure 8.7 shows the percent error between the perturbation theory predictions and the exact reactivity. Now it is clear that "virtual density" perturbation theory predicts the first derivative of eigenvalue *exactly* for the smallest perturbations, while boundary perturbation theory is in error by over 2%. In this particular case, boundary perturbation theory is more accurate for larger perturbations, but that accuracy is due to coincidence rather than any virtue of the method. The virtue of a first order perturbation method should be judged by how well it predicts the *first derivative* of eigenvalue for infinitesimally small perturbation magnitudes, and Figure 8.7 shows that "virtual density" perturbation theory predicts that first derivative precisely.

Figure 8.7 motivates us to directly compare how well "virtual density" perturbation theory and boundary perturbation theory predict the first derivatives of reactivities. We already did this in Figure 8.7, which has 4 mesh cells per slab. Now let us vary the number of mesh cells per slab and shrink the "boundary shift fraction" down to 0.0001 so that each perturbation method can best capture the first derivative of eigenvalue - the reactivity "coefficient" for this boundary shift.

Figure 8.8 compares the perturbation theory errors as a function of mesh cells per slab. While the accuracy of "virtual density" perturbation theory is completely independent of the mesh resolution, the accuracy of traditional boundary perturbation theory depends *strongly* on mesh resolution. This should not be terribly surprising, because the "virtual density" formulation allows one to express the perturbation in the same domain as the unperturbed solution - a simple difference of diffusion operators. In contrast, the boundary perturbation formulation can *not* be expressed in terms of the unperturbed diffusion operators - it requires accurate surface currents. Since this is a finite difference model, the surface current accuracy depends strongly upon the mesh resolution. Thus, while traditional boundary perturbation theory is only accurate for very fine meshes, "virtual density" perturbation theory is accurate for even very coarse meshes.

The solid black line in Figure 8.8 shows how well the direct eigenvalue solution (the "exact value") at each mesh resolution agrees with its value computed on the most hyperfine mesh. Since "virtual density" perturbation theory agrees precisely with this "exact value" at each mesh resolution, this black line also represents how well "virtual density" perturbation theory predicts the hyperfine "exact value". Thus, "virtual density" perturbation theory is always correct on a given mesh, but it can never be more accurate than that mesh itself. Figure 8.8 does *not* imply that boundary perturbation theory agrees more closely with the hyperfine "exact value", because the errors are actually of opposite sign (despite being the same order of magnitude).

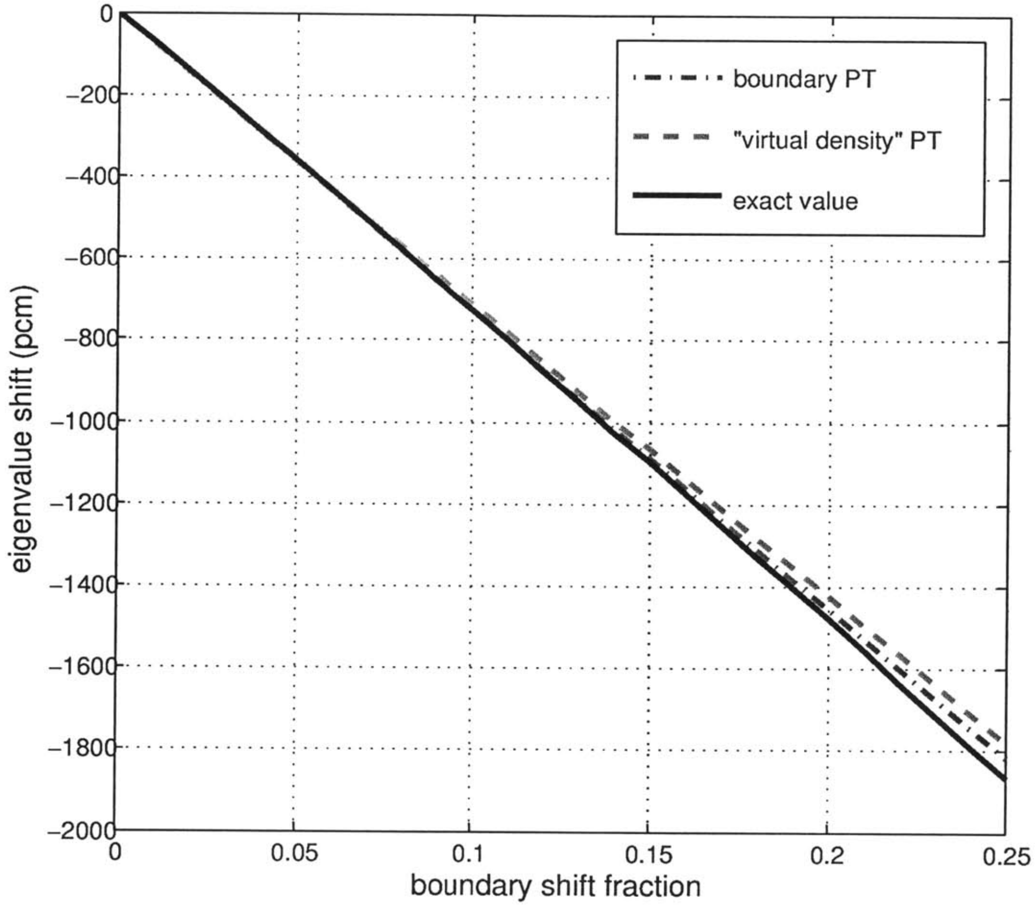

Figure 8.6: Reactivity due to movement of a single internal surface - surface 8 in Figure 8.3. The exact value (computed via finite difference diffusion) is compared to that predicted by "virtual density" and traditional boundary perturbation theory. Here the "boundary shift fraction" is equal to dx/a, where dx is the distance by which the boundary shifts and a is the slab width. Here each slab contains 4 finite difference mesh cells.

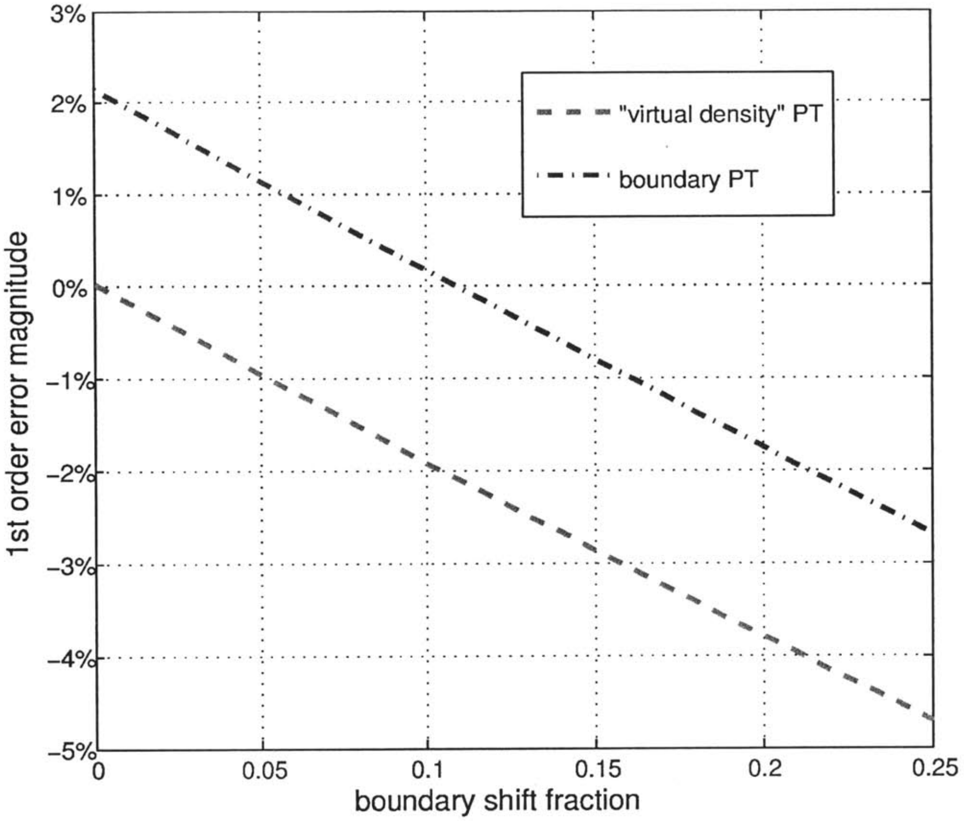

Figure 8.7: Perturbation theory error in reactivity due to movement of a single internal surface - surface 8 in Figure 8.3. Here the "boundary shift fraction" is equal to dx/a, where dx is the distance by which the boundary shifts and a is the slab width. Here each slab contains 4 finite difference mesh cells.

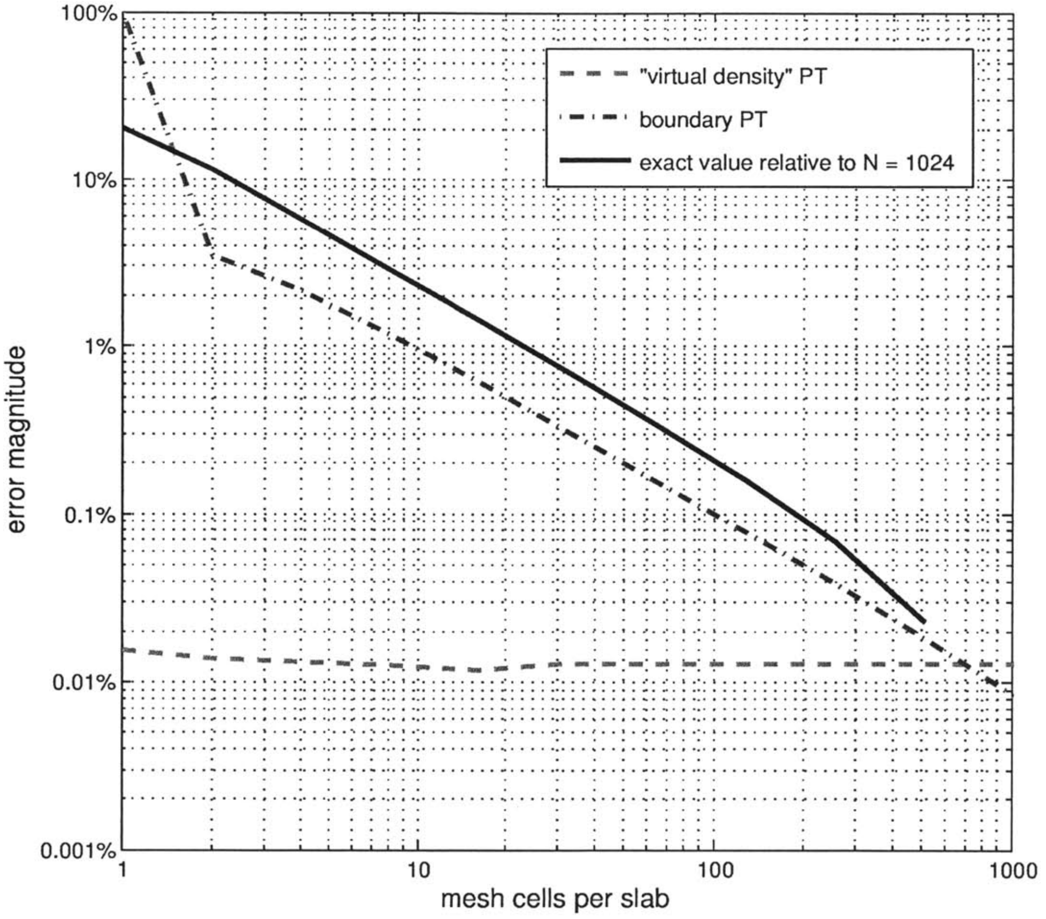

Figure 8.8: Perturbation theory error in eigenvalue derivative with respect to movement of a single internal surface (surface 8 in Figure 8.3) as a function of mesh cells per slab. The solid black line represents the error between the diffusion reference eigenvalue at each mesh resolution and at the most hyperfine mesh.

8.6.3 Single External Boundary Shifting

Now let us try shifting an *external* boundary - surface 14 in Figure 8.3. In boundary perturbation theory, we actually move surface 14 to the right by applying Equation 8.69. In "virtual density" perturbation theory, we simply *decrease* the neutron mean free path in slab 13 by *increasing* the sodium density in that slab. Since this perturbation amounts to thickening a reflector, we should expect reactivity to increase.

Figure 8.9 compares these two approaches to the exact reactivity. Here the "slab expansion fraction" is equal to dx/a, where dx is the distance by which the boundary shifts and a is the slab width. As expected, this surface shift causes an increase in reactivity, because the outer sodium slab (which acts as a reflector) is thickening. Clearly, Figure 8.9 shows that "virtual density" perturbation theory is much more accurate than boundary perturbation theory, which is in error by approximately 20%.

Again, we compare "virtual density" perturbation theory and boundary perturbation theory by how well they predict the first derivative of eigenvalue. We shrink the "slab expansion fraction" down to 0.0001 so that each perturbation can best capture that first derivative, and then we compare accuracy as a function of mesh cells per slab. Figure 8.10 shows the result. Clearly, "virtual density" perturbation theory is extremely accurate regardless of mesh coarseness, while traditional boundary perturbation theory is only accurate for very fine meshes.

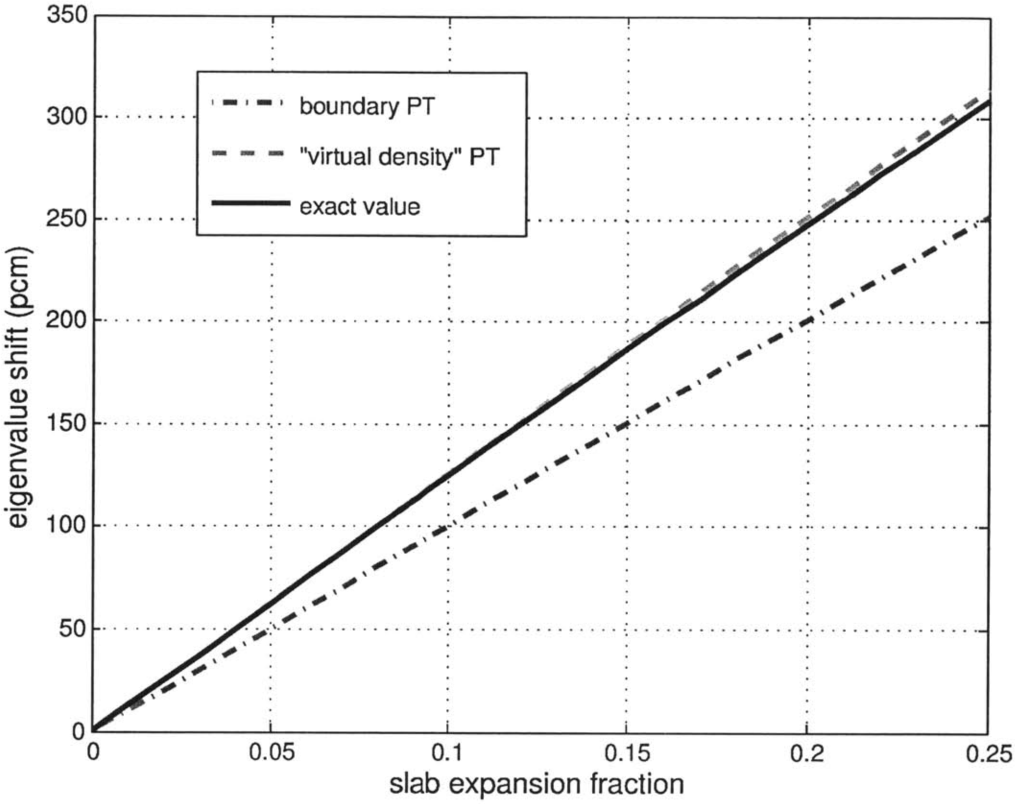

Figure 8.9: Reactivity due to movement of a single external surface - surface 14 in Figure 8.3. The exact value (computed via finite difference diffusion) is compared to that predicted by "virtual density" and traditional boundary perturbation theory. Here each slab contains 4 finite difference mesh cells.

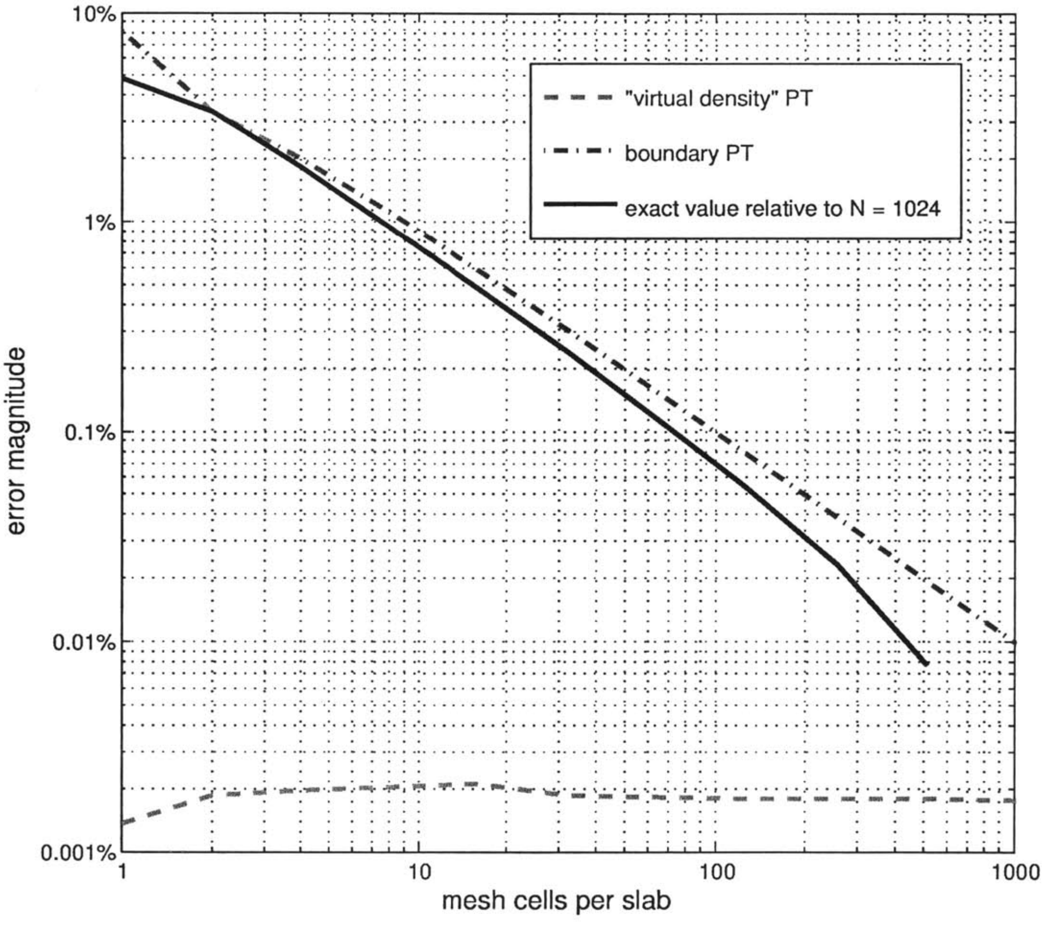

Figure 8.10: Perturbation theory error in eigenvalue derivative with respect to movement of a single external surface (surface 14 in Figure 8.3) as a function of mesh cells per slab. The solid black line represents the error between the diffusion reference eigenvalue at each mesh resolution and at the most hyperfine mesh.

8.6.4 Single Slab Shifting

Now let us consider a more practical geometry distortion. The external boundary shift does not conserve coolant mass, and the internal boundary shift conserves neither coolant nor fuel mass. Let us now suppose that a single fuel slab - slab 8 in Figure 8.3 - moves rightward. This conserves both coolant and fuel mass - it merely displaces sodium from slab 9 into slab 7.

In boundary perturbation theory, we actually move this whole slab by integrating along surfaces 8 and 9. In "virtual density" perturbation theory, we simply *increase* the sodium density in slab 7 and *decrease* the sodium density in slab 9.

Figure 8.11 compares these two approaches to the exact reactivity. The "slab shift fraction" is the fraction by which fuel slab 8 moves rightward toward fuel slab 10. A "slab shift fraction" of 1 corresponds to fuel slab 8 moving rightward until it touches fuel slab 10. Equivalently, the "slab shift fraction" is equal to dx/a, where dx is the distance by which the slab shifts and a is the slab width. Clearly, Figure 8.11 shows that "virtual density" perturbation theory is much more accurate than boundary perturbation theory, which is in error by approximately 80%.

Again, we compare "virtual density" perturbation theory and boundary perturbation theory by how well they predict the first derivative of eigenvalue. We shrink the "slab shift fraction" down to 0.0001 so that each perturbation can best capture that first derivative, and then we compare accuracy as a function of mesh cells per slab. Figure 8.12 shows the result. Clearly, "virtual density" perturbation theory is extremely accurate regardless of mesh resolution, while traditional boundary perturbation theory is only accurate for very fine meshes.

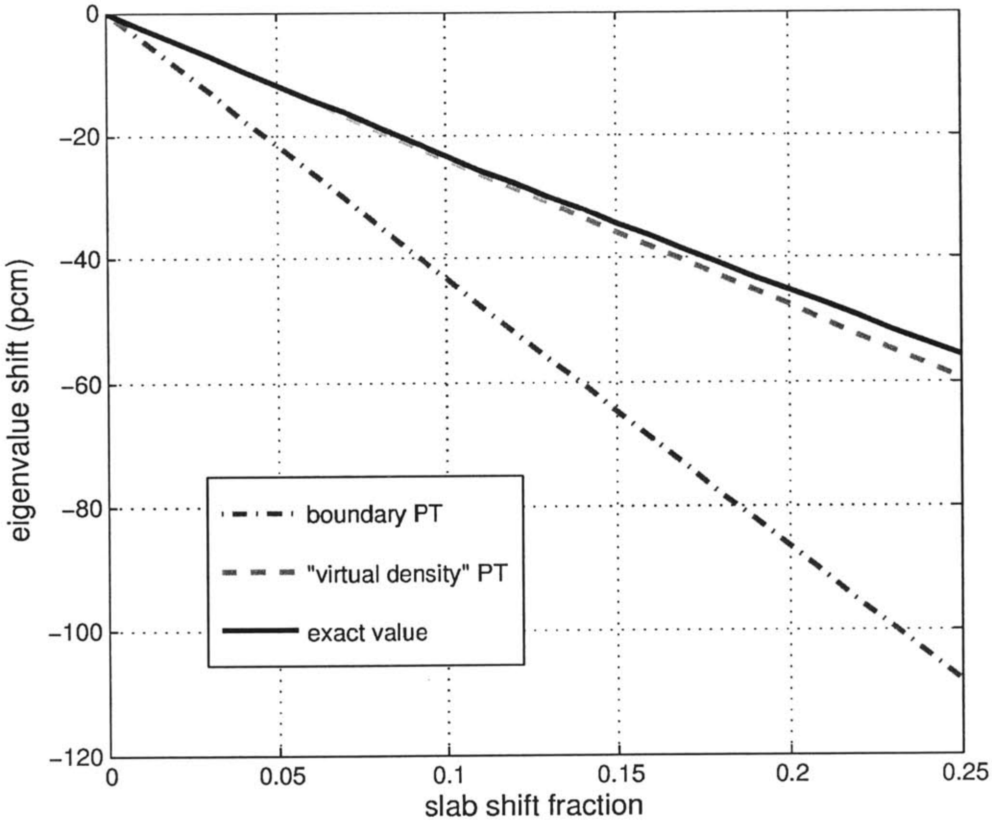

Figure 8.11: Reactivity due to movement of a single uranium slab - slab 8 in Figure 8.3. The exact value (computed via finite difference diffusion) is compared to that predicted by "virtual density" and traditional boundary perturbation theory. Here each slab contains 4 finite difference mesh cells.

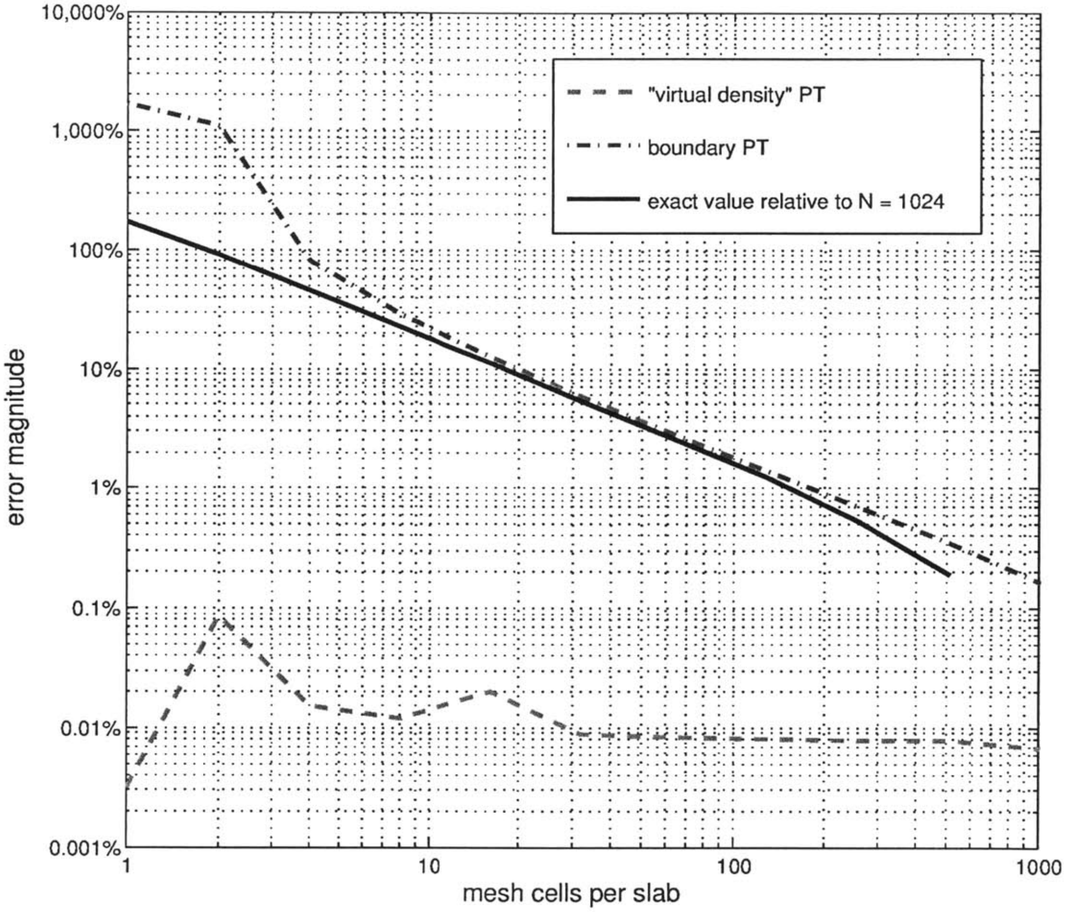

Figure 8.12: Perturbation theory error in eigenvalue derivative with respect to movement of a single fuel slab (slab 8 in Figure 8.3) as a function of mesh cells per slab. The solid black line represents the error between the diffusion reference eigenvalue at each mesh resolution and at the most hyperfine mesh.

8.6.5 Half Core Shifting

Now let us examine a final scenario. Suppose that slab 7 (sodium) in Figure 8.3 expands so that the entire right half of the core moves rightward. In boundary perturbation theory, we actually move half the core by integrating along surfaces 8, 9, 10, 11, 12, 13, and 14. In "virtual density" perturbation theory, we simply increase the sodium density in slab 7, and *that's it.* Thus, the "virtual density" theory converts seven surface integrals (spread throughout half the core) into one volume integral (localized in the core center).

Figure 8.13 compares these two approaches to the exact reactivity. The "slab expansion fraction" is equal to dx/a, where dx is the distance by which boundaries 8-14 shifts and a is the slab width. Clearly, Figure 8.13 shows that "virtual density" perturbation theory is vastly more accurate than boundary perturbation theory, which is in error by approximately 30%.

Again, we compare "virtual density" perturbation theory and boundary perturbation theory by how well they predict the first derivative of eigenvalue. We shrink the "slab expansion fraction" down to 0.0001 so that each perturbation can best capture that first derivative, and then we compare accuracy as a function of mesh cells per slab. Figure 8.14 shows the result. Clearly, "virtual density" perturbation theory is extremely accurate regardless of mesh resolution, while traditional boundary perturbation theory is only accurate for very fine meshes.

Through examining these four cases (internal boundary shift, external boundary shift, single slab shift, and half core shift) we have seen that the "virtual density" theory allows us to convert boundary perturbations into density perturbations, which in turn allows us to express perturbations as simple differences in the diffusion operators - the same operators we used to obtain the unperturbed solution. Thus, "virtual density" perturbation theory will be nearly equally accurate for coarse and fine meshes. In sharp contrast, traditional boundary perturbation theory can only be accurate for very fine meshes, because it requires accurate surface currents. Thus, this "virtual density" method makes geometric perturbation theory reliable in coarse mesh finite difference.

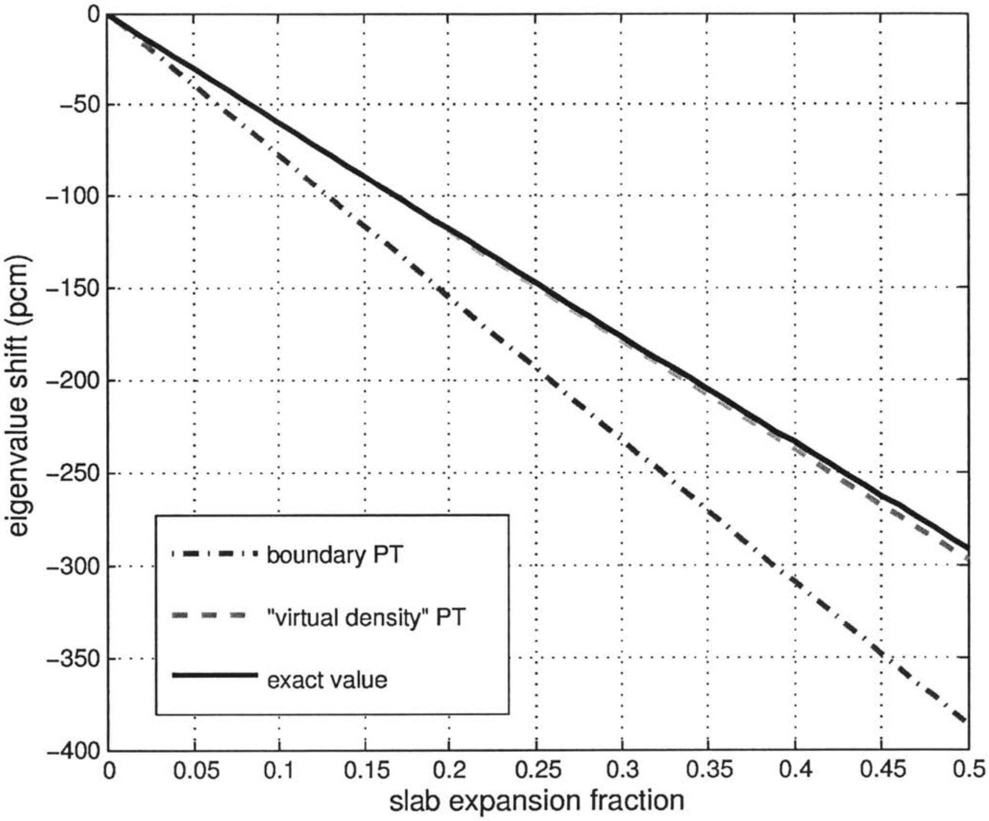

Figure 8.13: reactivity due to movement of half of the six-slab core. In Figure 8.3, surfaces 8-14 all move rightward by the same small distance. Here each slab contains 4 finite difference mesh cells.

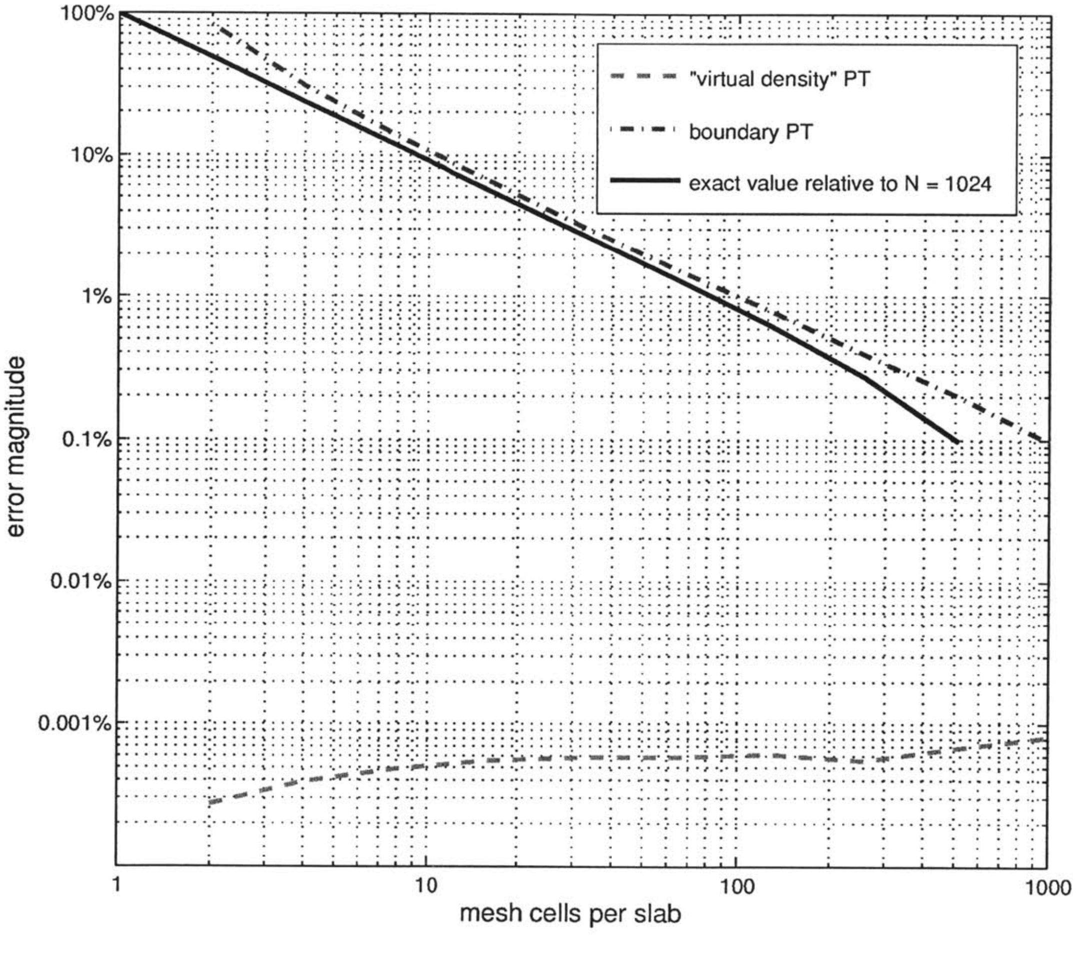

Figure 8.14: Perturbation theory error in eigenvalue derivative with respect to movement of half of the six-slab core as a function of mesh cells per slab. The solid black line represents the error between the diffusion reference eigenvalue at each mesh resolution and at the most hyperfine mesh.

8.6.6 Computational Advantages of "Virtual Density"

In addition to greatly enhanced accuracy on coarse meshes, "virtual density" perturbation theory has dramatic computational advantages in some cases. Consider the "half core shift" scenario once again. While traditional boundary perturbation theory requires seven separate surface integrals (and is not so accurate), "virtual density" perturbation theory requires only one volume integral (and is very accurate).

What if we were to increase the number of fuel slabs in this 1-D reactor? The reactor in Figure 8.3 has six fuel slabs, but what if it had ten, twenty, or one hundred fuel slabs? How would the computation times of boundary perturbation theory and "virtual density" perturbation theory scale with the number of slabs in this 1-D reactor? If the reactor has N slabs, boundary perturbation theory would require $N + 1$ surface integrals. However, "virtual density" perturbation theory would still only require *one* slab integral.

Figure 8.15 compares the perturbation theory computation time for these two methods as a function of N. While the "virtual density" perturbation theory computation time remains roughly constant, the boundary perturbation theory computation time is roughly proportional to N. Thus, for ***this case only***, we can say that "virtual density" perturbation theory is $O(1)$ while boundary perturbation theory is $O(N)$.

One might contend that this "half core shift" scenario is completely unrealistic, but that is not so. It is actually very similar to a scenario posed by Favorite for the Zeus benchmark problem, in which a gap near the reactor center opens up and shifts about 40% of the core downward [63]. Whenever large swaths of the core move together, this $O(1)$ versus $O(N)$ comparison is valid. However, for the most generic case of each individual fuel slab rattling around arbitrarily, both methods are $O(N)$.

Figure 8.16 compares the error magnitudes of "virtual density" perturbation theory and boundary perturbation theory as a function of N. Neither method displays any strong trend here, but it is clear that "virtual density" perturbation theory is much more accurate in addition to being much more computationally efficient.

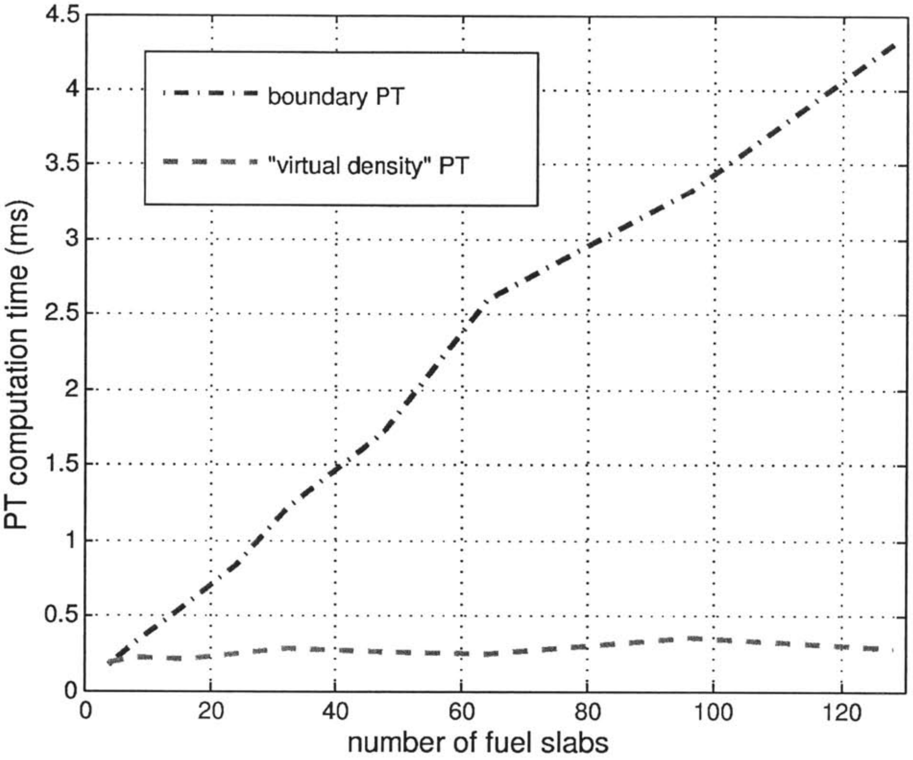

Figure 8.15: Perturbation theory computation time due to movement of half of an N-slab core. The computation time for the "virtual density" method is compared to that for the traditional boundary perturbation theory method. Computation times for the first order reactivity (the derivative of eigenvalue with respect to the perturbation magnitude) are shown as a function of N. Here each slab contains 8 finite difference mesh cells.

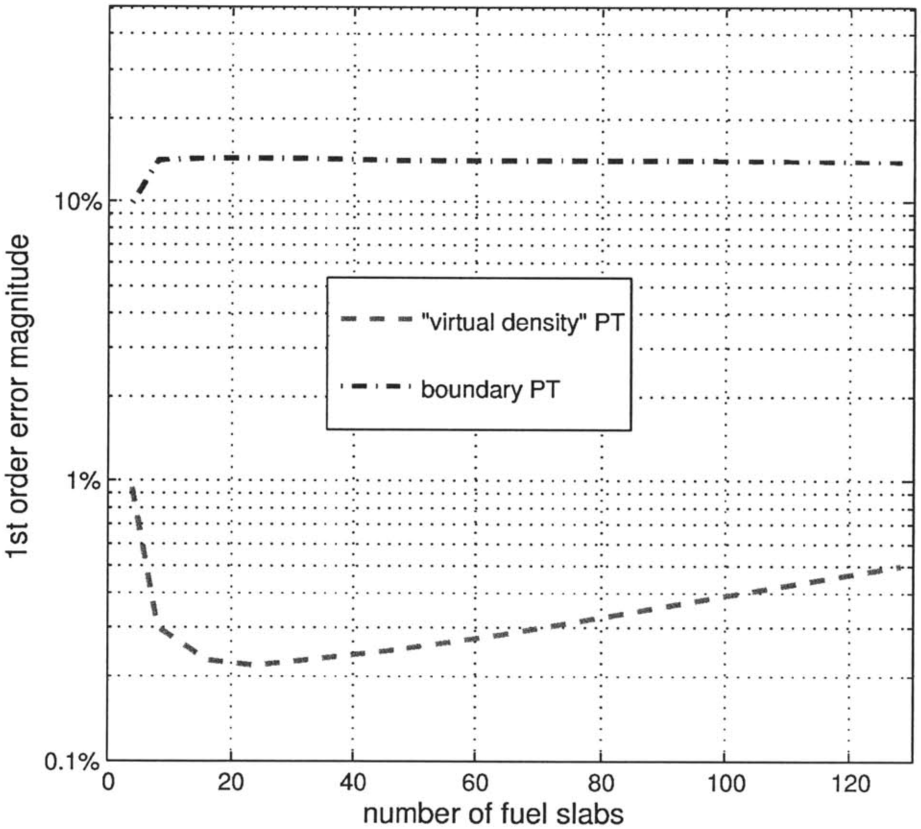

Figure 8.16: Perturbation theory error in reactivity due to movement of half of an N-slab core as a function of N. Here each slab contains 8 finite difference mesh cells.

8.6.7 Higher Order "Virtual Density"

Until this point, we have only examined first order perturbation theory. This is adequate for obtaining the first derivative of reactivity with respect to some perturbation - a reactivity coefficient. However, if we wish to estimate reactivities for larger perturbations, we need second order perturbation theory. First order perturbation theory predictions are always linear, and although many eigenvalue responses to perturbations are in fact quite linear themselves, linear predictions will always be limited to small perturbation magnitudes.

Unfortunately, our literature review concluded that second order perturbation theory has not been applied to *internal* surface perturbations in diffusion theory. All the literature (mostly by Rahnema and his various colleagues) is centered around transport theory, and the sparse diffusion treatment is exclusively for *external* boundaries. One might suppose that applying the transport methods to diffusion would be simple, but that is not necessarily true. The boundary perturbation "anomaly" discussed by Pomraning and Rahnema complicates diffusion but does not appear in transport theory [43]. Many variational functionals could certainly be applied to internal boundary shifts in diffusion, but there is no guarantee that any will work - functionals are finicky!

Fortunately, if we have already converted a boundary perturbation into a "virtual density" perturbation, then we can simply apply the well-known functional developed by Stacey [33]. We discussed this functional in our literature review (see Equations 2.27 and 2.28 in Section 2.3.5), and it is universally applicable to all material density perturbations - it is tried and true. Thus, the "virtual density" theory allows for reliable application of second order methods to diffusion, while traditional boundary perturbation theory does not.

Figure 8.17 compares boundary perturbation theory with first *and* second order "virtual density" perturbation theory for the "half core shift" problem. Here the "slab expansion fraction" has the same definition, but now we follow it out to 3. This means that the central sodium slab thickness is quadrupling! While first order "virtual density" correctly predicts the first derivative of the reactivity, second order "virtual density" correctly predicts the second derivative of the reactivity - it captures the curvature.

Figure 8.18 shows the error magnitudes in Figure 8.17. Second order "virtual density" perturbation theory predicts the reactivity to within 4% when the slab quadruples in thickness.

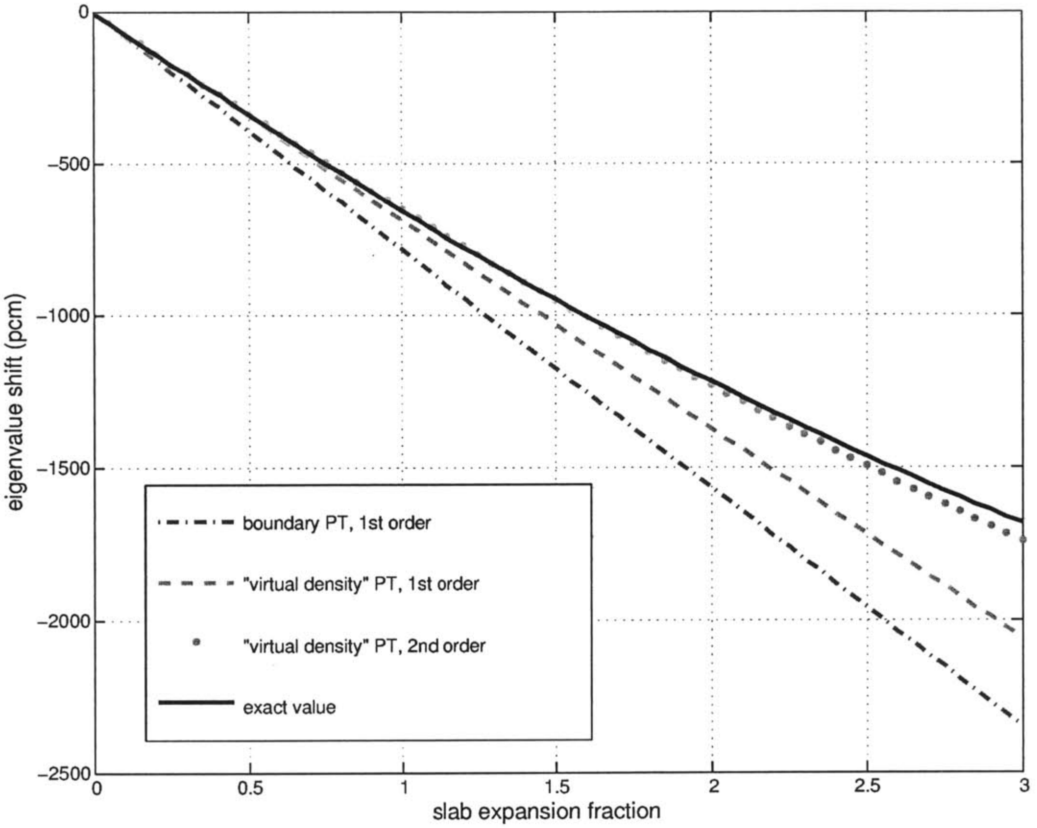

Figure 8.17: First and second order "virtual density" perturbation theory predictions of reactivity due to movement of half of the six-slab core. In Figure 8.3, surfaces 8-14 all move rightward by the same small distance. Here each slab contains 8 finite difference mesh cells.

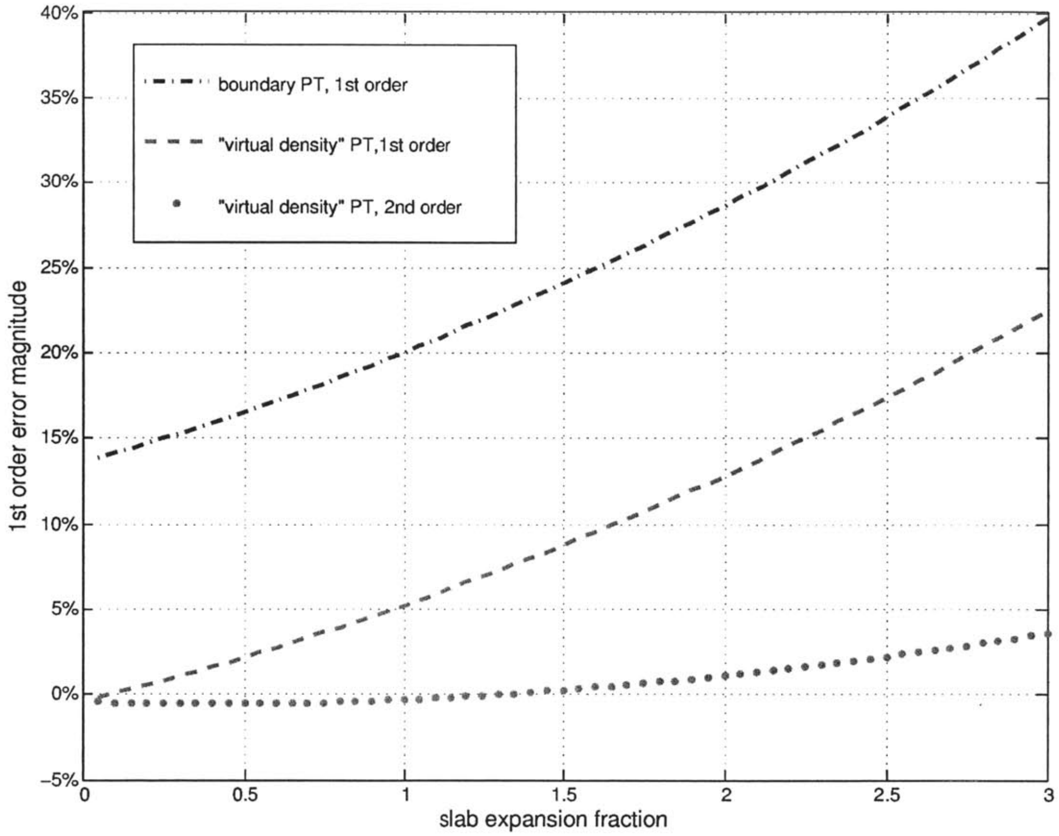

Figure 8.18: First and second order "virtual density" perturbation theory error in reactivity due to movement of half of the six-slab core. Here each slab contains 8 finite difference mesh cells.

8.7 Interacting Effects due to Multiple Surface Perturbations?

As we discussed in our literature review, Favorite attempted to predict the first derivative of eigenvalue with respect to moving the entire lower 40% of the Zeus benchmark problem [63]. He found that although first order boundary perturbation theory predicted the first derivatives very well for *single* interface shifts, it failed for multiple interfaces shifting together. Favorite even states that "the first order theory has no way to account for interacting effects of multiple perturbations" and that the derivative of eigenvalue "with respect to the location of the Zeus lower core is not the simple sum of the derivatives with respect to the location of the individual surfaces".

However, our analytic derivation in Section 8.4 shows that, at least for arbitrary 1-D problems, the boundary perturbation method is (1) precisely equivalent to our "virtual density" method and (2) precisely additive for multiple interfaces. So in analytic test problems and derivations, first order boundary perturbation theory *is* additive. The problem arises not in analytic cases, but in numeric cases. As we demonstrated throughout Section 8.6, boundary perturbation theory is very inaccurate on coarse meshes, because the surface currents are not well-converged. We also demonstrated that boundary perturbation theory tends to be much more accurate for single internal interface shifts than for multiple internal interface shifts. Compare Figures 8.6 and 8.8 to Figures 8.11 and 8.12. With 4 mesh cells per slab, boundary perturbation theory yields 2% error for a *single* interface shift but 30% error for *two* interface shifts (a single slab shift). As the mesh becomes more fine, boundary perturbation theory always converges to the correct answer, but it converges much more slowly for multiple interfaces than it does for a single interface.

The explanation for this enigma is actually quite simple. We know that boundary perturbation theory is inaccurate on coarse meshes due to poorly-converged surface currents in the leakage term. Thus, boundary perturbation theory is relatively accurate when the leakage term comprises a smaller fraction of the total reactivity and relatively *in*accurate when the leakage term comprises a larger fraction of the total reactivity. When we shift a *single* interface between a fuel region and a non-fuel region, we either gain or lose fuel mass. Intuitively, this gain or loss of fissile material causes most of the reactivity, because the fission term is huge. Thus, leakage contributes only a small amount to the total reactivity and can introduce only small errors. In contrast, shifting multiple interfaces (such as in the "single slab shifting" and "half core shifting" cases) does *not* add or remove net fuel mass. Thus, the fission term is zero. Leakage comprises a much larger portion of the total reactivity and introduces much larger errors.

Although Favorite's work on Zeus employs deterministic transport (S_N, to be specific), his results are very similar to our own coarse mesh diffusion results in 1-D slab geometry. Regardless of whether one works in transport or diffusion, surface currents in the leakage term will likely always produce most of the error simply because they are not directly used to obtain the unperturbed eigenvalue. Furthermore, the fission term will likely always dominate contribution to the reactivity, unless there *is* no fission term.

Thus, individual interface shifts within Zeus yield small errors because their fission terms dwarf their leakage terms, while multiple concurrent interface shifts yield large errors because they have a net zero fission term. These large errors in the first derivative of eigenvalue do *not* arise from interacting effects between surface perturbations. Such effects do not exist, because surface perturbations are *always* linearly independent to first order.

8.8 The Uniqueness of "Virtual Density" Perturbations

In Section 8.4, we showed that a single boundary surface integral is equivalent to two slab volume integrals. However, when multiple boundaries move together, many of these slab volume integrals counteract one another. Consider once again the "half core shift" problem, in which the *right* half of the core (surfaces 8-14) shifts *right*ward by an amount Δx. Each of the seven surface integrals is equivalent to two slab integrals, but all save one of those slab integrals counteract one another, and we are left with only one non-zero slab integral (slab 7).

Now suppose that the *left* half of the core (surfaces 1-7) shifts *left*ward by an amount Δx. Intuitively, we know that this produces the same reactivity as the *right* half of the core shifting *right*ward. This is *not* a consequence of core symmetry - it would be true regardless of core composition. Now consider a third perturbation in which the *left* half of the core shifts *left*ward by an amount $\alpha\Delta x$ while the *right* half of the core shifts *right*ward by an amount $(1 - \alpha)\Delta x$. Here α is an arbitrary number between 0 and 1. Intuition tells us that this will also produce the same reactivity for all values of α. In boundary perturbation theory, these three equivalent perturbations would require surface integrals over three different sets of surfaces: 1-7, 8-14, or 1-14. However, the equivalent "virtual density" perturbation would still consist of the same single volume integral over slab 7.

Now consider the "single internal boundary shift" case, in which surface 8 shifts *right*ward. Boundary perturbation theory requires a single integral over surface 8, while the corresponding "virtual density" perturbation requires integrals over slabs 7 and 8.

Now suppose that surface 8 remains fixed, but all other surfaces shift *left*ward. Intuitively, we know this perturbation is equivalent to only surface 8 shifting *right*ward. Applying boundary perturbation theory to this equivalent perturbation requires surface integrals over all surfaces *except* surface 8. However, the corresponding "virtual density" perturbation still requires integrals over only slabs 7 and 8.

One can perform more thought experiments in this vein, but the conclusion is that each "virtual density" perturbation can only be defined one unique way. Boundary perturbations, on the other hand, can always be defined at least two different ways (and sometimes multiple different ways).

8.9 Numeric Comparison for Non-Uniform Anisotropic Swellings in 3-D Hexagonal-Z

So far, we have shown that "virtual density" perturbation theory (VirDenT) and boundary perturbation theory (BPT) are analytically equivalent but *not* numerically equivalent in finite difference diffusion - especially not for coarse mesh solutions.

Now we demonstrate the same finding in a practical core model. We take the same FFTF benchmark (described in Appendix G) that we employed throughout Chapter 7. In this case, we wish to rapidly run numerous reference calculations in quick succession, so we abbreviate the FFTF benchmark by removing the outermost 4 assembly rings. This leaves a 7-ring core containing 1 reflector ring encircling 6 rings of fuel interspersed with control and test assemblies. Each assembly contains six triangular mesh cells, and 22 axial mesh cells. k_{eff} is 0.9486.

8.9.1 Estimation of Boundary Flux Gradients

Before displaying results, we must again define precisely how we compute the 3-D flux gradients in Equation 8.63. Consider Figure 8.19, which displays a hexagonal assembly divided into axial zones with six triangular mesh cells per hexagon. Axially swelling even one axial zone causes *all* the surfaces above it to shift upward, so one must know the real and adjoint flux gradients (in all three directions) integrated over each triangular surface. Although the unperturbed finite difference solution provides gradients perpendicular to these surfaces, it does *not* provide gradients parallel to these surfaces. We must estimate these parallel gradients ourselves.

The arrows on surface 3 in Figure 8.19 represent the axial flux gradients (perpendicular to the surface), and we compute these just as we did for the 1-D case in Section 8.6.1. The red arrows on surface 3 represent the radial (or lateral) flux gradients (parallel to the surface), and obtaining these is less straightforward. Since we already know the volume-averaged radial flux gradients in cells 2 and 3, we can linearly interpolate these to estimate the gradient values on surface 3. If the axial heights of cells 2 and 3 are dz_2 and dz_3, then we could interpolate like this:

$$\nabla\phi_{s3} = \frac{\frac{\nabla\phi_2}{dz_2} + \frac{\nabla\phi_3}{dz_3}}{\frac{1}{dz_2} + \frac{1}{dz_3}} \tag{8.70}$$

Here $\nabla\phi_{s3}$ is the *radial* flux gradient averaged over surface 3, while $\nabla\phi_2$ and $\nabla\phi_3$ are the *radial* flux gradients in cells 2 and 3. We try this interpolation along with many others, including

$$\nabla\phi_{s3} = \frac{\frac{D_2\nabla\phi_2}{dz_2} + \frac{D_3\nabla\phi_3}{dz_3}}{\frac{D_2}{dz_2} + \frac{D_3}{dz_3}} \tag{8.71}$$

Here D_2 and D_3 are the diffusion coefficients in cells 2 and 3. Neither Equation 8.70 nor Equation 8.71 yield significantly better accuracy than the other.

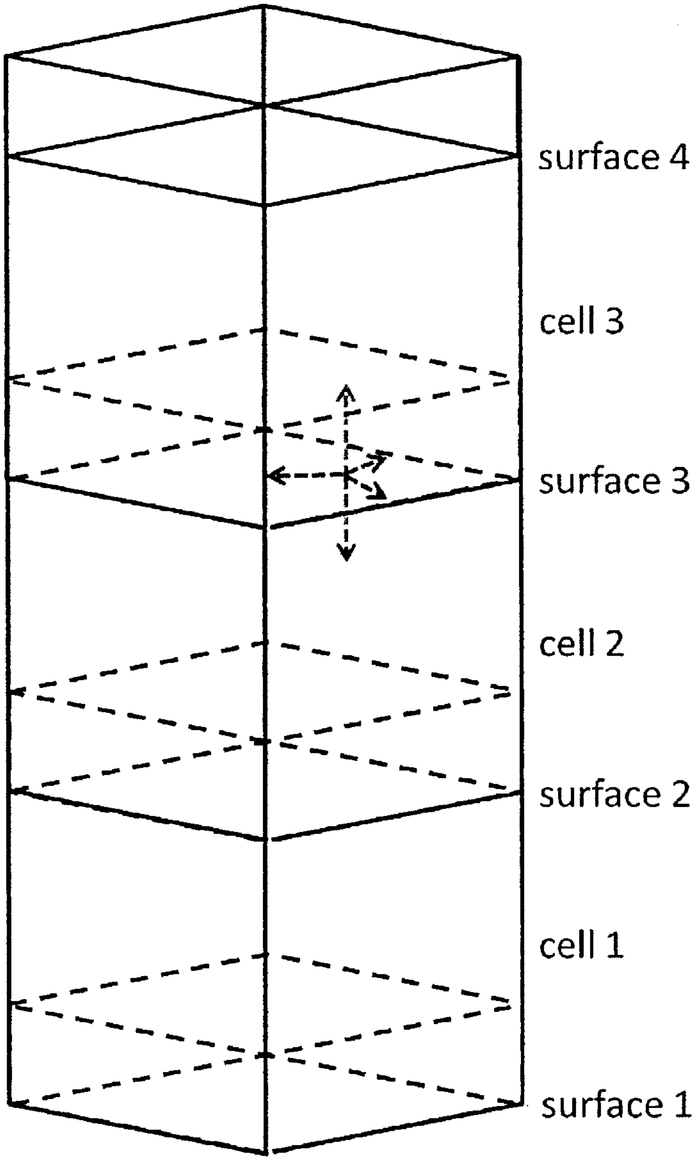

Figure 8.19: Boundary flux gradients for axial swelling of a hexagonal assembly in a triangular-z finite difference mesh. The arrows represent the axial flux gradient averaged over one triangular surface, while the red arrows represent the lateral (or radial) flux gradients averaged over the same triangular surface.

8.9.2 Numeric Results

We compare VirDenT and BPT for two types of axial swelling scenarios: (1) axial swelling of each mesh cell in proportion to its fission power and (2) uniform axial swelling of each assembly.

In scenario (1), we determine the fission power of each hexagonal axial segment of each fuel assembly, and we swell each of those axial segments in proportion to its power. We set the maximum swelling of any segment within each assembly to 0.1%. We compute one "virtual mesh" reference case for such swelling of *each* fuel assembly within the first five assembly rings. We also rapidly compute the "virtual density" reactivity coefficient for each assembly swelling. Figure 8.20 shows the magnitude of the reactivity coefficient error between the "virtual density" perturbation and the "virtual mesh" reference. They always agree to less than 0.1%, and the errors are remarkably uniform. Remember that in Section 7.10, we concluded that these errors arise from the reference, not the perturbation. Thus, reducing the maximum swelling magnitude in each assembly to below 0.1% would cause the errors shown in Figure 8.20 to shrink even further and eventually approach zero.

Figure 8.21 shows the same scenario (1), but now we compare BPT to the "virtual mesh" reference. Errors range as high as 5% with an average between 2% and 3%.

Figure 8.22 shows scenario (1) with a twist: instead of swelling axially, the assembly zones *expand* axially. Thus, their axial dimensions increase without a decrease in density. Although this case is not physically possible, it serves to elucidate the source of error in BPT. In order to compute a swelling in BPT, one must perturb the mesh cell boundaries *and* decrease cell densities in a separate perturbation. Since a density reduction is simple and precise to compute, we know its error must be small - on the same order as the "virtual density" errors. Thus, most error in BPT must arise from the boundary perturbations alone, and we can isolate this fact with an axial expansion case. Figure 8.22 shows that the errors are much larger, even over 20%. This confirms our expectation that errors in BPT arise almost entirely from the boundary perturbations themselves, not density perturbations added on top of them. The "virtual density" theory has "virtually" no error, because it converts the boundary perturbations into "density" perturbations!

Now let us turn to scenario (2), which is uniform axial swelling of each assembly. This allows us to swell fuel and non-fuel assemblies alike. We can rest assured that "virtual density" perturbation theory will agree with the "virtual reference" precisely, as we already performed this comparison in Chapter 7. Thus, here we show only the BPT comparison to a "virtual mesh" reference.

Figure 8.23 shows the results. Comparing this to Figure G.1 in Appendix G reveals which positions are fuel, control, and test assemblies. Although most fuel assemblies have errors well below 50%, control assemblies exhibit no pretense of accuracy. BPT predicts axial swelling reactivity coefficients that are wrong by a factor of 4 in secondary control assemblies and a factor of 3 in primary control assemblies. Evidently, large intra-assembly flux gradients within these control assemblies completely ruin interpolation of the boundary flux gradients.

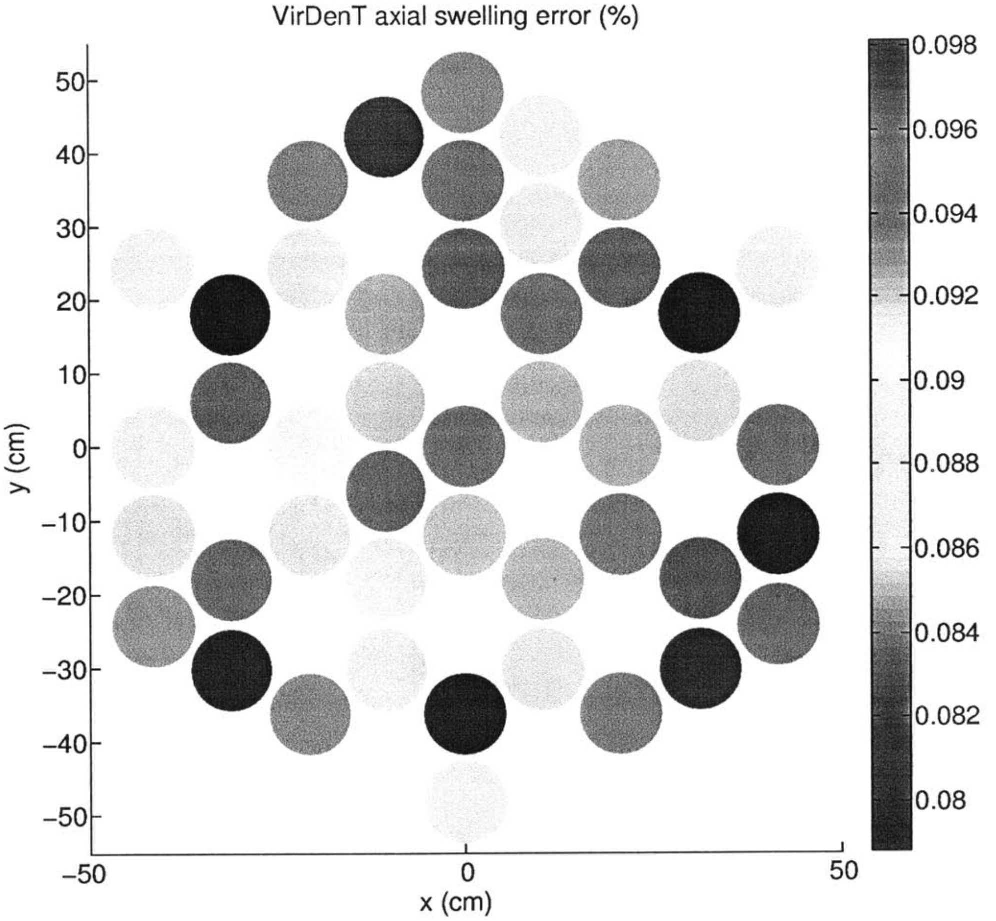

Figure 8.20: The VirDenT reactivity coefficient error (for axial swelling proportional to power) relative to a "virtual mesh" diffusion reference case in the FFTF benchmark. Here each axial mesh cell of each fuel assembly swells axially in proportion to its power. The cumulative reactivity coefficient for each assembly is calculated via a direct "virtual mesh" eigenvalue solution and via VirDenT. The two calculations always agree to less than 0.1%.

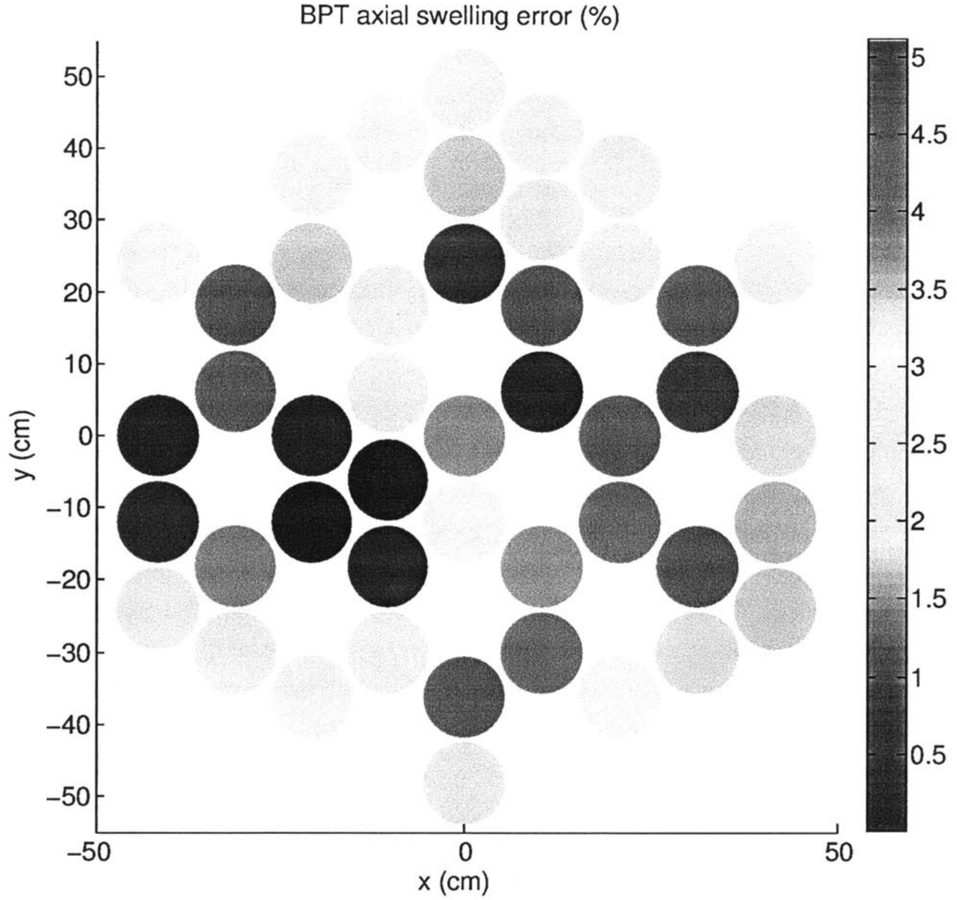

Figure 8.21: The BPT reactivity coefficient error (for axial swelling proportional to power) relative to a "virtual mesh" diffusion reference case in the FFTF benchmark. Here each axial mesh cell of each fuel assembly swells axially in proportion to its power. The cumulative reactivity coefficient for each assembly is calculated via a direct "virtual mesh" eigenvalue solution and via BPT. The two calculations agree to less than about 5%.

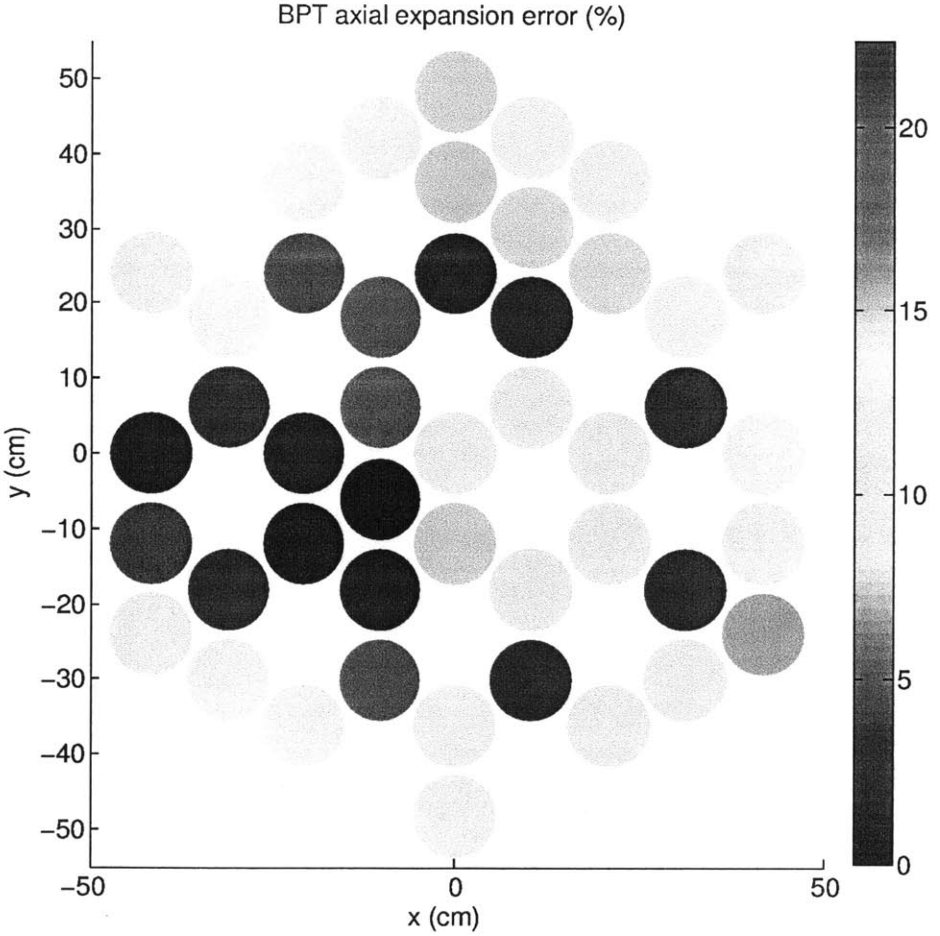

Figure 8.22: The BPT reactivity coefficient error (for axial expansion proportional to power) relative to a "virtual mesh" diffusion reference case in the FFTF benchmark. Here each axial mesh cell of each fuel assembly expands axially in proportion to its power. The cumulative reactivity coefficient for each assembly is calculated via a direct "virtual mesh" eigenvalue solution and via BPT. Errors fall anywhere between 1% and 22%.

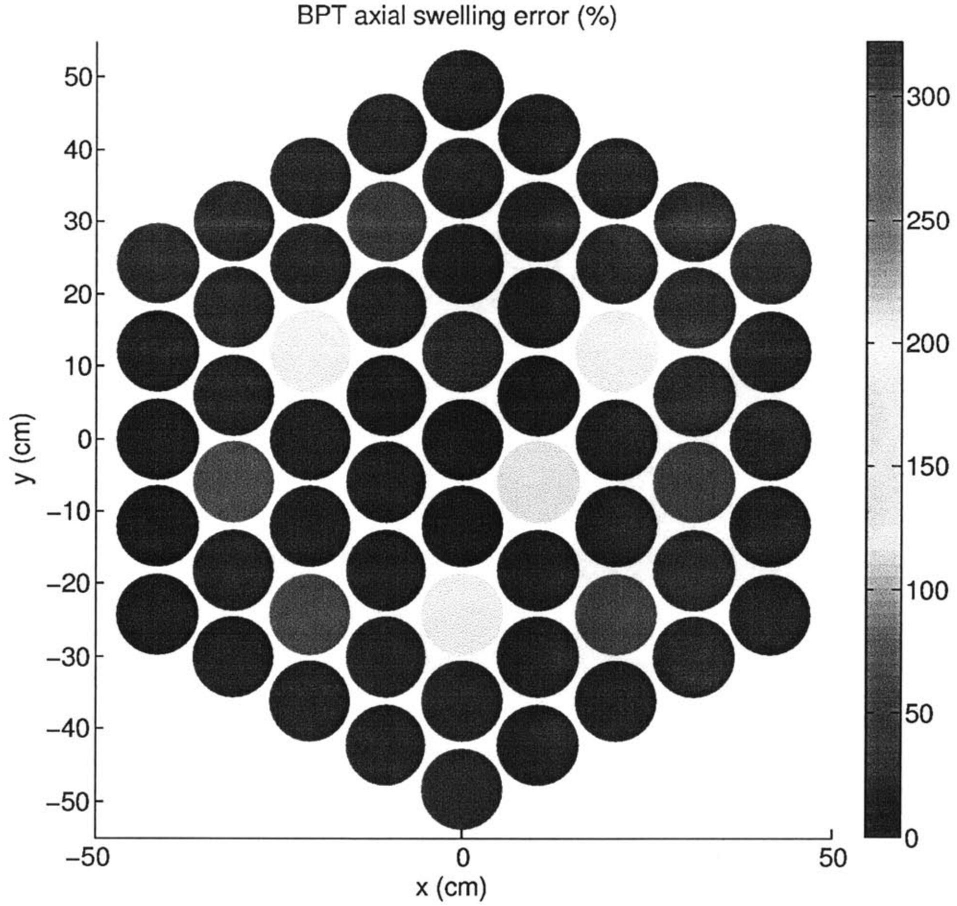

Figure 8.23: The BPT reactivity coefficient error (ffor uniform axial swelling within each assembly) relative to a "virtual mesh" diffusion reference case in the FFTF benchmark. The cumulative reactivity coefficient for each assembly is calculated via a direct "virtual mesh" eigenvalue solution and via BPT. Errors are relatively small in fuel assemblies but gargantuan in control assemblies.

8.10 Summary

Traditional boundary perturbation methods require highly accurate surface currents, rendering them impotent in coarse mesh finite difference. In contrast, “virtual density” theory can express these boundary perturbations as simple differences of the volume-averaged unperturbed fluxes in each mesh cell, allowing for much more precise prediction of reactivities in coarse mesh problems. In many of these problems, “virtual density” perturbation theory is orders of magnitude more precise than traditional boundary perturbation theory. We have demonstrated this fact in hexagonal-z geometry in an abbreviated FFTF benchmark, and we have shown in 1-D Cartesian geometry that reactivity coefficients computed via boundary perturbation theory converge to those computed via “virtual density” theory as the mesh becomes finer.

As far as we know, boundary perturbation theory has not been used for industry application, and this analysis provides a strong indication for why this is so. The diffusion boundary perturbation formula is certainly correct analytically, but its implementation in coarse mesh finite difference can be unreliable. Thus, our “virtual density” perturbation theory constitutes the first successful application of geometric perturbation theory to full-core models in diffusion theory.

9 "Virtual Density" and Point Kinetics: Reactivity Safety via the PseuSeiAnimator

9.1 Chapter Abstract

We insert reactivities computed via the non-uniform anisotropic "virtual density" theory into standard point kinetic theory. In this way, we determine the time-dependent neutronic behavior of a small 3-D Cartesian sodium fast reactor core with a low dominance ratio. We demonstrate the most generic distortion scenario - random motion of assemblies in the radial plane, which we dub "pseudo-seismic". We find that this pseudo-seismic assembly motion induces a time-dependent reactivity with maximums well below β and a time-average value of zero. Oscillatory pseudo-seismic motion of *individual* assemblies induces little power evolution, while addition of *aggregate* assembly motion (oscillatory radial flowering) inserts negative reactivity and causes core power to slowly fall. We create the Pseudo-Seismic (PseuSei) Animator as an instructive tool to show assembly motion and power evolution in real time.

9.2 Introduction

So far, we have obtained reactivity coefficients via the "virtual density" theory - *instantaneous* reactivities inserted by *instantaneous* geometry distortions. Of course, the next step is to determine the *time-dependent* neutronic behavior induced by *time-dependent* reactivities inserted by *time-dependent* geometry distortions. In this chapter, we will show how to insert "virtual density" reactivities into reactor point kinetic theory.

9.3 Point Kinetics via the Prompt Jump Approximation

Neutrons don't live very long. In a typical light water reactor (LWR), the mean time it takes for a neutron to be born at fast energies, slow down via moderation, and induce thermal fission is on the order of one millisecond. In fast reactors, which skip the moderating step, neutrons live on the order of a *micro*second. This "prompt" neutron lifetime represents the time scale at which one might expect a reactor power level to rise or fall. Of course, this is far quicker than any human operator can react. So if this were the whole story, reactors could never operate safely. They would need to be balanced *precisely* at $k_{\text{eff}} = 1$, and even the *slightest* nudge above this would cause the power level to run away precipitously.

Fortunately, not all neutrons are born promptly from fission. A small fraction β are delayed - they emerge from the daughter fission products on the order of *seconds* after the main fission event. Thus, as long as the inserted reactivity is less than β, the reactor power level will rise or fall on the order of *seconds* - slow enough for an operator to control. This is what makes reactor control possible!

Delayed neutrons constitute anywhere from 0.20% to 2.03% of all neutrons for common fissionable isotopes. Refer back to Table 1.1 for a list. ^{235}U and ^{239}Pu have delayed neutron fractions of $\beta = 0.64\%$ and 0.20%, respectively. Thus, a breed-and-burn fast reactor initially fueled by uranium will experience a gradual fall in β throughout its lifetime. The β value will begin a bit above 0.64% and fall to a bit above 0.20%. The small portion of fission on ^{238}U ($\beta = 1.62\%$) holds the core-average β above what its value would be for pure fissile material.

The most accurate way to model reactor kinetics in diffusion theory is "spatial kinetics". This involves directly solving the time-dependent multigroup diffusion equation for $\phi_g(\vec{r}, t)$ in each energy group g.

$$\frac{1}{v_g}\frac{d\phi_g}{dt} - \nabla \cdot D_g \nabla \phi_g + \Sigma_{r,g}\phi_g = \sum_{g'=1}^{g-1} \Sigma_{s,g' \to g}\phi_{g'} + \chi_g \sum_{g'=1}^{G} \nu_{g'}\Sigma_{f,g'}\phi_{g'} \tag{9.1}$$

This is identical to the steady-state diffusion equation in Equation 2.6, except now there is a time-dependent term. Here v_g is the velocity of neutrons in group g.

Since solving the full equation is an arduous task, nuclear engineers frequently use "point kinetics" as an approximation to "spatial kinetics". Point kinetic theory assumes that the whole reactor flux and power distribution behaves as a "point" - it can rise or fall in time, but its shape does not change. This approximation is only accurate when the time scale in which a neutron travels across a reactor (spatially) is not large relative to the time scale at which reactor power rises or falls during a transient.

A good indicator of this is the "dominance ratio", which is the ratio of the first eigenvalue k_1 to the fundamental eigenvalue $k_0 = k_{\text{eff}}$. In LWRs, which have thermal spectra, dominance ratios are quite high - often above 0.99. This is because neutrons move around the reactor slowly enough (due to low speeds and short mean free paths) that one core region can experience a transient before another core region "feels" its effects. Thus, this relatively slow spatial propagation of transient effects causes the flux and power distributions to *change shape* during transients. So point kinetic theory is not very accurate, and spatial kinetics is necessary. Now consider a fast reactor, which has high neutron speeds and long mean free paths. The dominance ratio is low - only 0.81 for the FFTF benchmark. Fast neutrons can travel throughout the core relatively quickly, and so transient effects can spatially propagate quickly - even more quickly than the transient time scale itself. Thus, the flux and power shapes will not change significantly, and point kinetic theory is accurate. We should also note that cores with spatial "lobes" impede propagation of transient effects, causing higher dominance ratios and decreasing the validity of point kinetics - the power of each "lobe" can behave differently during a transient. Wrapping up this digression, we can confidently state that point kinetics is more valid for low dominance ratios and less valid for high dominance ratios.

Numerous variations on point kinetic theory exist, and we will choose the simplest - the "prompt jump" approximation. While more complex point kinetic formulations "group" delayed neutrons by the lengths of their decay constants, the prompt jump model assumes *one* delayed neutron group with an aggregate lifetime λ. Rather than model the extremely rapid power "jump" due to prompt neutrons (before the delayed neutrons "catch up"), we can simply model this "prompt jump" as an instantaneous rise in power from P_1 to P_2 [157]:

$$\frac{P_2}{P_1} = \frac{\beta - \rho_1}{\beta - \rho_2} \tag{9.2}$$

Here an instantaneous reactivity insertion from ρ_1 to ρ_2 causes the power to instantaneously "jump" from P_1 to P_2. Now consider a continuous variation of reactivity $\rho(t)$. Duderstadt's iconic textbook gives this expression for the power $P(t)$ [157]:

$$P(t) = P_0 e^{A(t)} \tag{9.3}$$

The power varies exponentially from initial value P_0, and the exponent $A(t)$ also varies in time [157]:

$$A(t) = \int_0^t d\tau \left[\frac{\dot{\rho}(\tau) + \lambda\rho(\tau)}{\beta - \rho(\tau)} \right] \tag{9.4}$$

Here $\rho(\tau)$ is the instantaneous reactivity insertion, and $\dot{\rho}(\tau)$ is its first derivative in time. Note that Equation 9.4 integrates the entire history of the reactivity (and its derivative) from the beginning of the transient up until the current time t.

If we are given a set of geometry distortions as a function of time, we can use "virtual density" theory to compute $\rho(t)$ and $\dot{\rho}(t)$. Then we can simply numerically integrate Equations 9.3 and 9.4 through time to obtain the power evolution $P(t)$.

9.4 Numeric Demonstration: Pseudo-Seismic Motion

Since we have no authentic seismic distortion data, we must make do with approximations. Consider the 3-D 6x6 array of Cartesian assemblies studied in Chapter 6 and illustrated in Figure 6.7. We can conjure up "pseudo-seismic" motion by making each assembly position oscillate sinusoidally in space and time with randomly-generated period, amplitude, and angular direction. Let η be a random number between 0 and 1. We can sample the angular direction θ of each assembly motion:

$$\theta = 2\pi\eta \tag{9.5}$$

Now we choose maximum and minimum oscillation amplitudes $r_{\max}$ and $r_{\min}$ so that assemblies do not "overlap" one another, and we randomly select the oscillation magnitude r of each assembly:

$$r = (r_{\max} - r_{\min})\eta + r_{\min} \tag{9.6}$$

We sample the oscillation period τ in the same manner:

$$\tau = (\tau_{\max} - \tau_{\min})\eta + \tau_{\min} \tag{9.7}$$

Now the $x(t)$ and $y(t)$ coordinates of each assembly center as functions of time are simple to define. Here x_0 and y_0 are the initial (unperturbed) assembly coordinates.

$$x(t) = x_0 + r\cos\theta\sin\left(\frac{2\pi t}{\tau}\right) \tag{9.8}$$

$$y(t) = y_0 + r\sin\theta\sin\left(\frac{2\pi t}{\tau}\right) \tag{9.9}$$

Now we have each assembly oscillating in a random direction with random amplitude and period. The time-averaged assembly positions are (x_0,y_0), so there is no enlargement or contraction of the whole core (no flowering). However, this does not really capture what might happen in a true seismic distortion case. In reality, as assemblies rattle around and bump into each other, they will tend to spread out such that the aggregate core size increases slightly. We can model this effect with an extra "flowering term" in both $x(t)$ and $y(t)$.

Let R_0 be the distance from each assembly center to the core center.

$$R_0^2 = x_0^2 + y_0^2 \tag{9.10}$$

Now suppose that $R(t)$ varies in time like this:

$$\frac{R(t)}{R_0} = 1 + f\left[\frac{1}{\alpha} + \sin\left(\frac{2\pi t}{\tau_f}\right)\right] \tag{9.11}$$

Here f is the maximum radial swelling factor; it would be 0.01 for a 1% increase in radial position. The parameter α can range from 1 to ∞. When $\alpha = 1$, the core flowers only radially outward and never inward. When $\alpha > 1$, the core oscillates between flowering radially outward and inward. As $\alpha \to \infty$, the core flowers only radially inward. The flowering period τ_f is randomly sampled in the same way as τ, except τ_f is the same for every assembly (τ is sampled separately for each assembly).

Now the full expressions for $x(t)$ and $y(t)$ are

$$x(t) = x_0 + r\cos\theta\sin\left(\frac{2\pi t}{\tau}\right) + x_0 f\left[\frac{1}{\alpha} + \sin\left(\frac{2\pi t}{\tau_f}\right)\right] \tag{9.12}$$

$$y(t) = y_0 + r\sin\theta\sin\left(\frac{2\pi t}{\tau}\right) + y_0 f\left[\frac{1}{\alpha} + \sin\left(\frac{2\pi t}{\tau_f}\right)\right] \tag{9.13}$$

So if we want to model pseudo-seismic motion of each assembly independently, we use Equations 9.8 and 9.9. If we wish to include oscillatory flowering in the pseudo-seismic motion, we use Equations 9.12 and 9.13.

9.4.1 The Pseudo-Seismic (PseuSei) Animator

We develop a graphical user interface (GUI) in MATLAB [210] to visualize geometry distortions and their point kinetic effects. Currently, this tool supports only pseudo-seismic assembly motion in Cartesian geometry, and so we dub it **PseuSei**. See Figure 9.1 for a screenshot of PseuSei, which animates a 2-D axial slice of 3-D assembly motion and plots the resulting $\rho(t)$ and $P(t)$. PseuSei numerically integrates Equations 9.3 and 9.4. Figure 9.1 is merely an exaggerated hypothetical example of pseudo-seismic motion. The interstitial coolant zones are wider than usual, and the assembly displacements are correspondingly larger than usual. A pre-recorded PseuSei animation is on YouTube.

PseuSei rapidly computes and plots assembly displacements, $\rho(t)$, and $P(t)$ in real time. Pre-computation involves solving the unperturbed real and adjoint flux distributions via MaPS and then computing the "virtual density" quantities L_x and Γ_x for the vertically-directed interstitial coolant zones and L_y and Γ_y for the horizontally-directed interstitial coolant zones. We can add up these quantities using Equation 5.32 to obtain the reactivity coefficients $d\rho/dx$ and $d\rho/dy$ for each assembly. Refer back to our validation of single assembly motion in Section 7.7.3. This concludes the pre-computation effort. Then, we can generate assembly displacements and compute reactivities (for arbitrary assembly motion) extremely rapidly in real time using Equation 9.14 (identical to Equation 7.19).

$$\frac{d\rho}{dr} = \frac{d\rho}{dx}\cos\theta + \frac{d\rho}{dy}\sin\theta \tag{9.14}$$

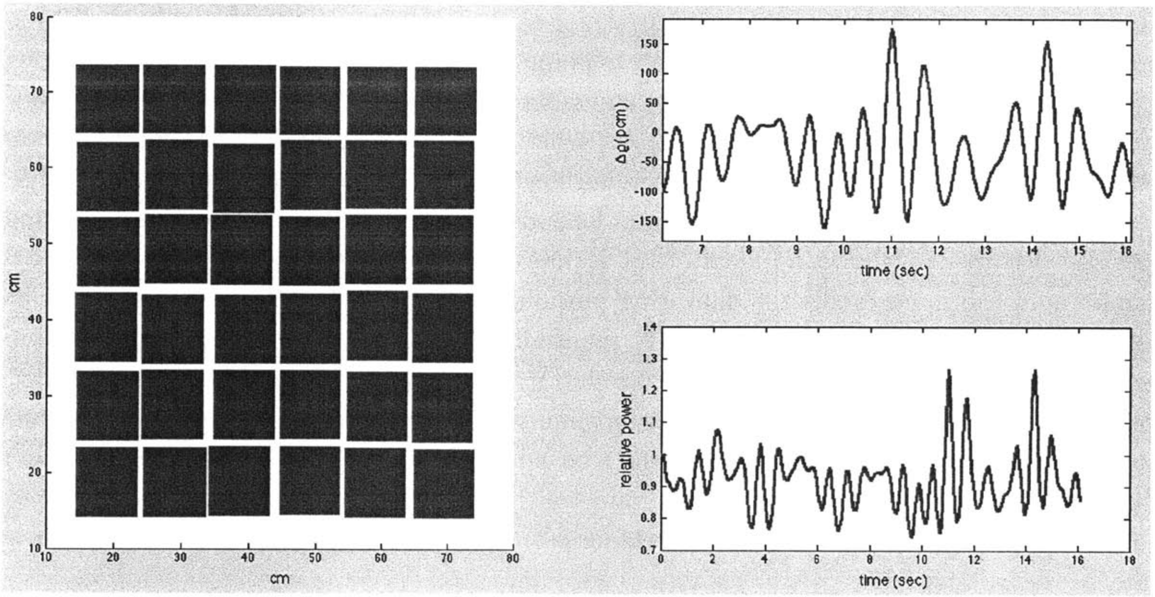

Figure 9.1: A screenshot of the PseuSei visualization tool. The left screen shows the "pseudo-seismic" motion of a 6x6 assembly array. The upper right screen shows the reactivity $\rho(t)$ in pcm. The lower right screen shows the point kinetic power evolution $P(t)/P_0$.

9.4.2 3-D Cartesian Results

We stick with the same 3-D 6x6 array of Cartesian assemblies studied in Chapter 6 and illustrated in Figure 6.7. Let the assembly side length be 10 cm and the interstitial coolant width be 0.3 cm. As we showed in Section 7.7, this interstitial gap is narrow enough to ensure very linear reactivities due to assembly motion. The unperturbed k_{eff} is 1.04345 with a dominance ratio of 0.95106, which is actually quite low relative to dominance ratios in LWRs and other thermal reactors. Thus, we will assume that point kinetics is valid. We assume $\beta = 0.68\%$ to be consistent with enriched uranium fuel, and we choose an approximate fast neutron lifetime of $\lambda = 0.4353$ s^{-1}.

Figures 9.2 and 9.3 show $d\rho/dx$ and $d\rho/dy$ for each assembly in units of pcm/cm. We can apply Equation 9.14 to obtain $d\rho/dr$ when r points in the $\theta = \pi/4$ direction. This is shown in Figure 9.4. Note that the six diagonal assemblies have zero reactivity coefficients due to core symmetry. In Figure 9.4, we plot $d\rho/dr$ when r points radially outward from the core center. Thus, we have broken down the radial flowering reactivity coefficient by assembly.

Now we can use PseuSei to plot time-dependent behavior for various pseudo-seismic motions. Figure 9.6 shows application of Equations 9.8 and 9.9, which have oscillatory seismic motion but no oscillatory flowering motion. The reactivity $\rho(t)$ has a time-average of zero, but, intriguingly, the power $P(t)$ gradually drifts *upward*. At first consideration, one might suppose this is because the "prompt jump" is larger than the "prompt drop". However, decreasing the numeric integration time step dt shows that the $P(t)$ trend flattens out as $dt \to 0$. Thus, the upward drift may be no more than a numeric specter, and then $P(t)$ simply oscillates around its initial value with no upward or downward trend. Since analytic solutions to Equation 9.4 for periodic $\rho(t)$ are not readily attainable, we leave it at that.

Figure 9.7 shows oscillatory seismic motion *and* oscillatory flowering motion as defined in Equations 9.12 and 9.13 with $\alpha = 1$ and $f = 0.005$. This means that the core flowering is always outward, so the flowering reactivity is always negative. Thus, the net reactivity is almost always negative, because the flowering reactivity is large relative to the seismic reactivity. The core power dwindles to half its initial value within 10 seconds.

Figure 9.8 shows oscillatory seismic motion *and* oscillatory flowering motion as defined in Equations 9.12 and 9.13 with $\alpha = 4$ and $f = 0.005$. This means that the core flowering is mostly outward but sometimes inward. Thus, the net reactivity has a negative time-averaged value but is frequently positive. The power oscillates with a downward trend and drops to half its initial value in about 30 seconds.

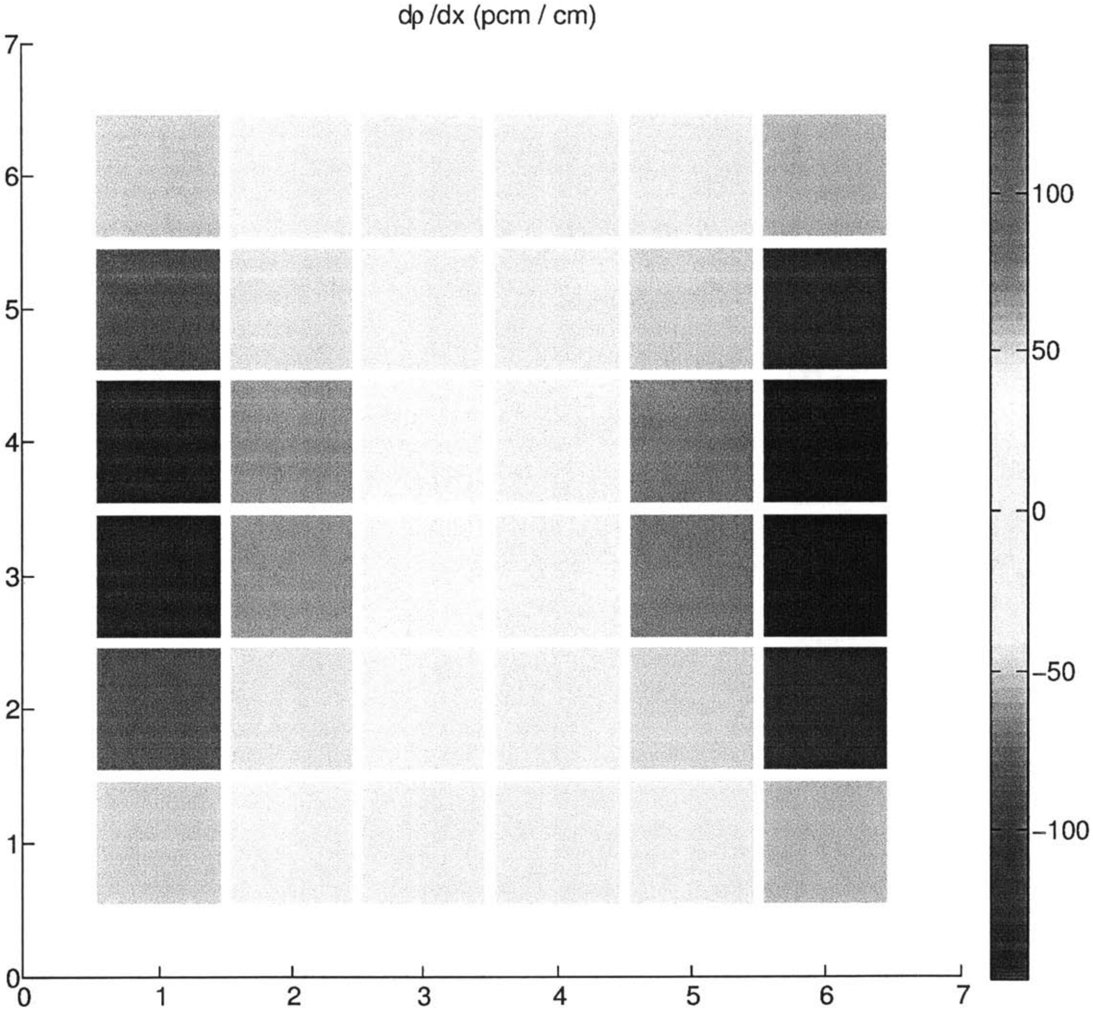

Figure 9.2: The reactivity coefficient $d\rho/dx$ for each assembly.

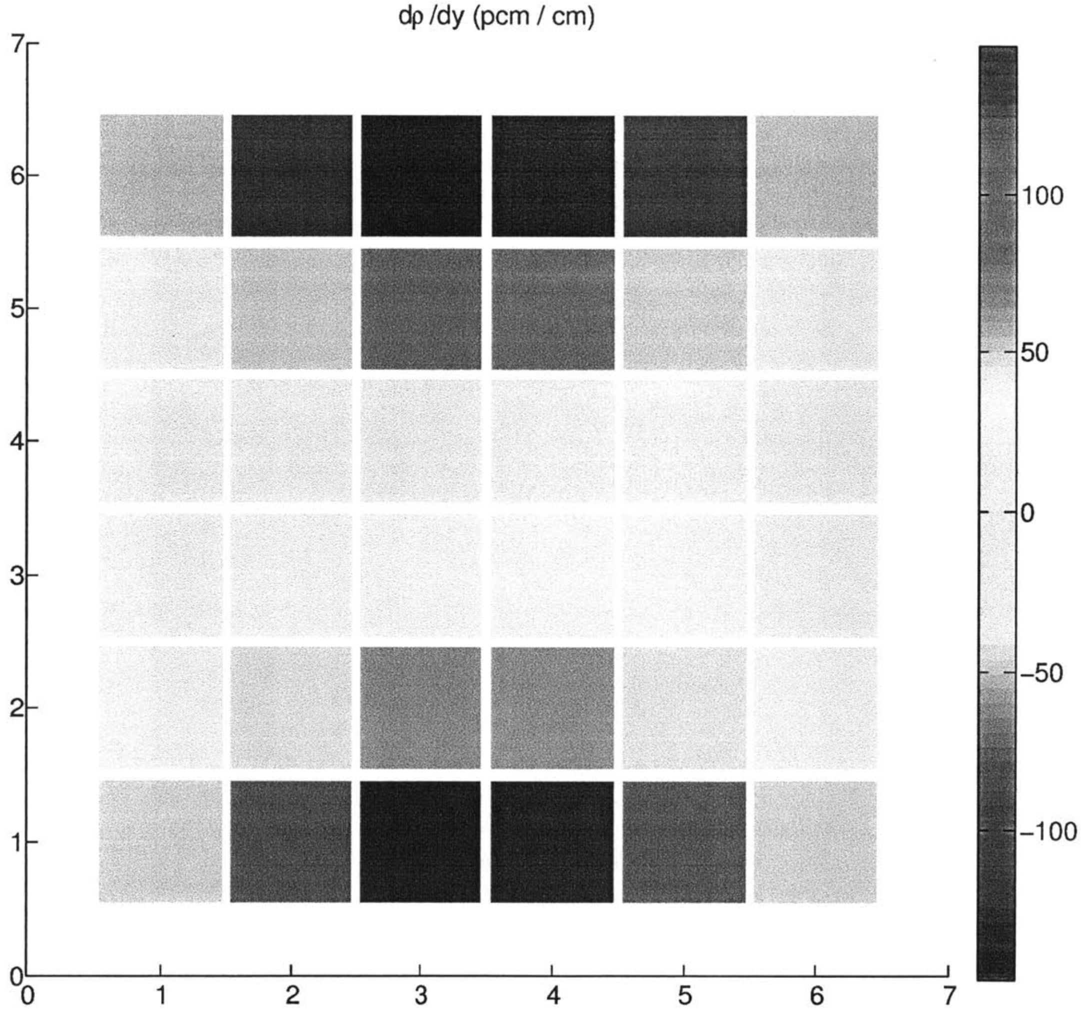

Figure 9.3: The reactivity coefficient $d\rho/dy$ for each assembly.

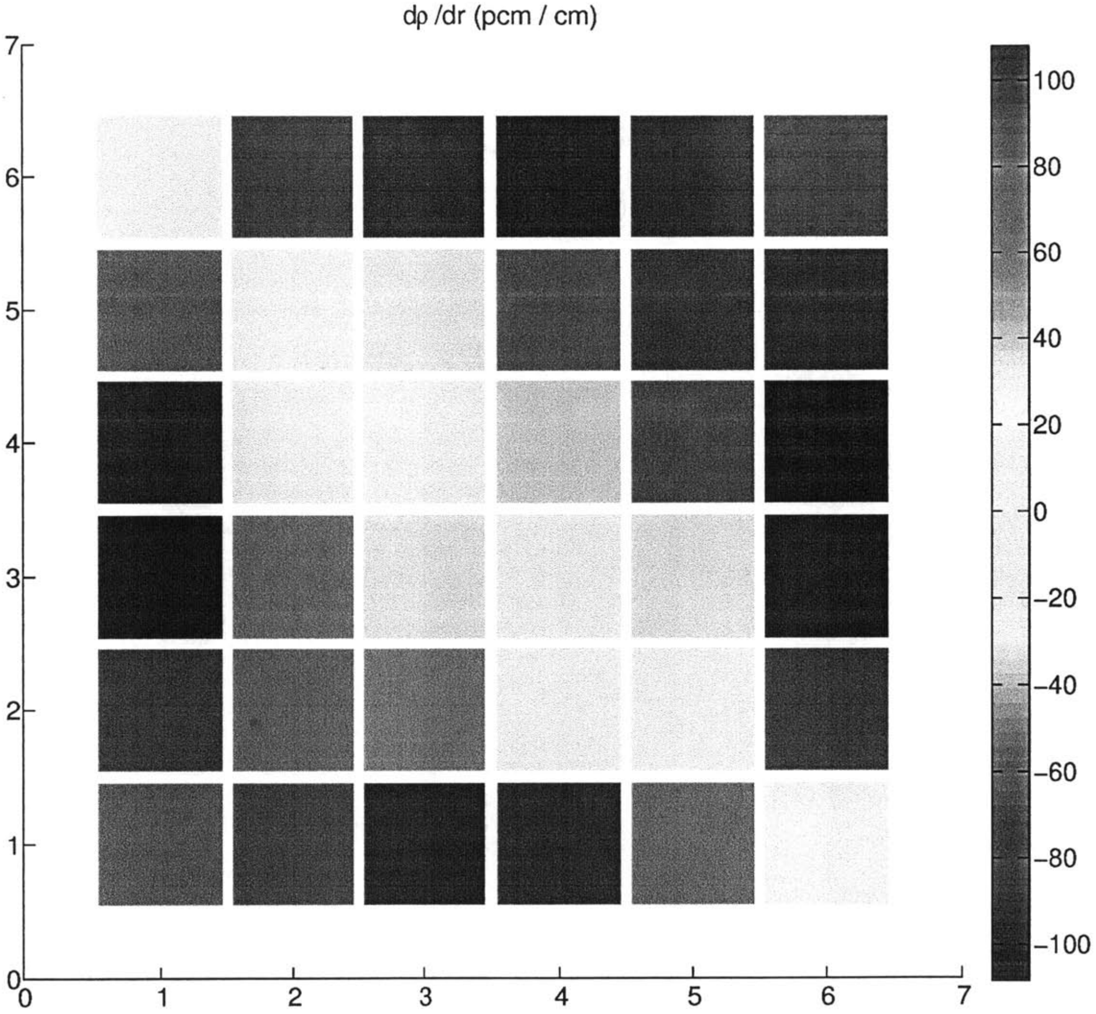

Figure 9.4: The reactivity coefficient $d\rho/dr$ for each assembly, where r points in the $\theta = \pi/4$ direction.

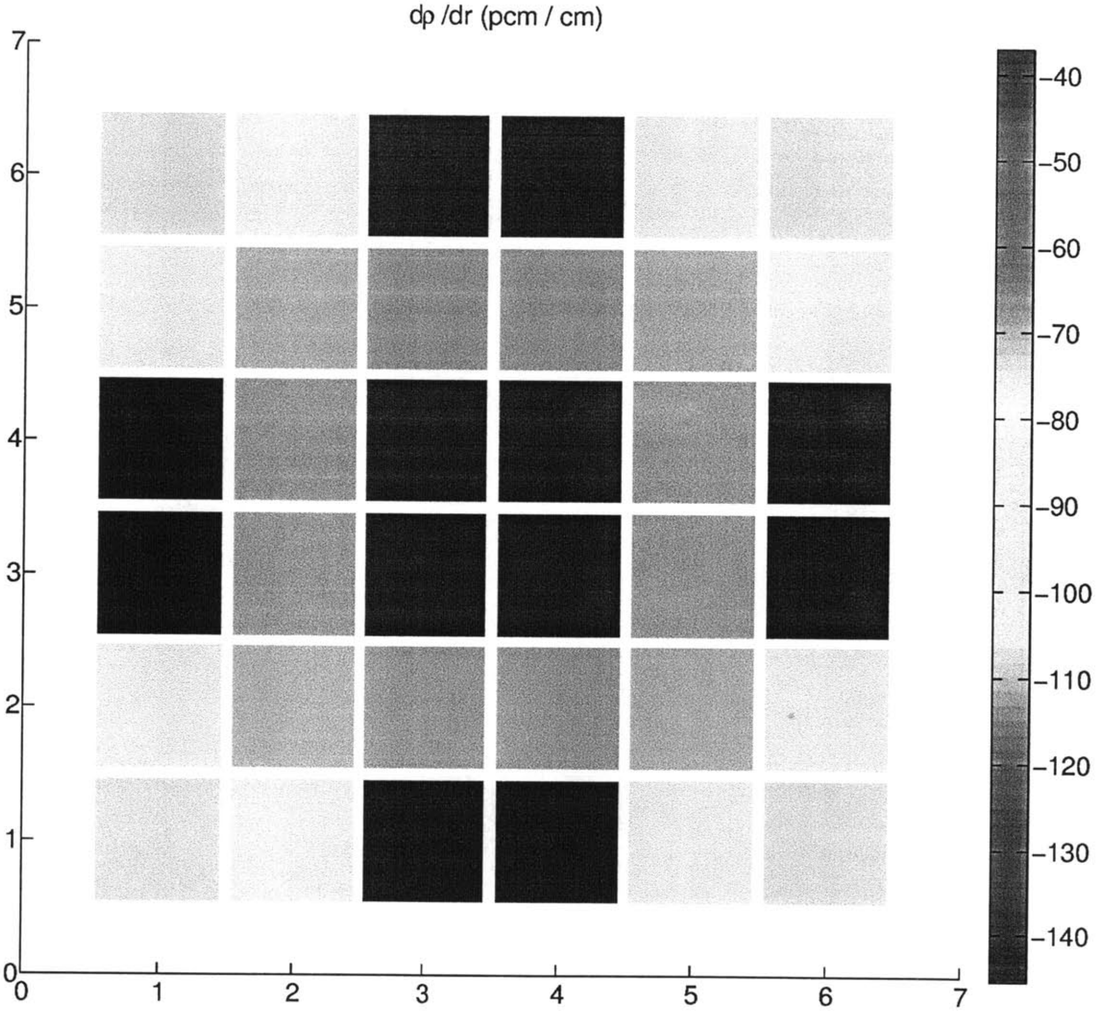

Figure 9.5: The reactivity coefficient $d\rho/dr$ for each assembly, where r points radially outward from the core center.

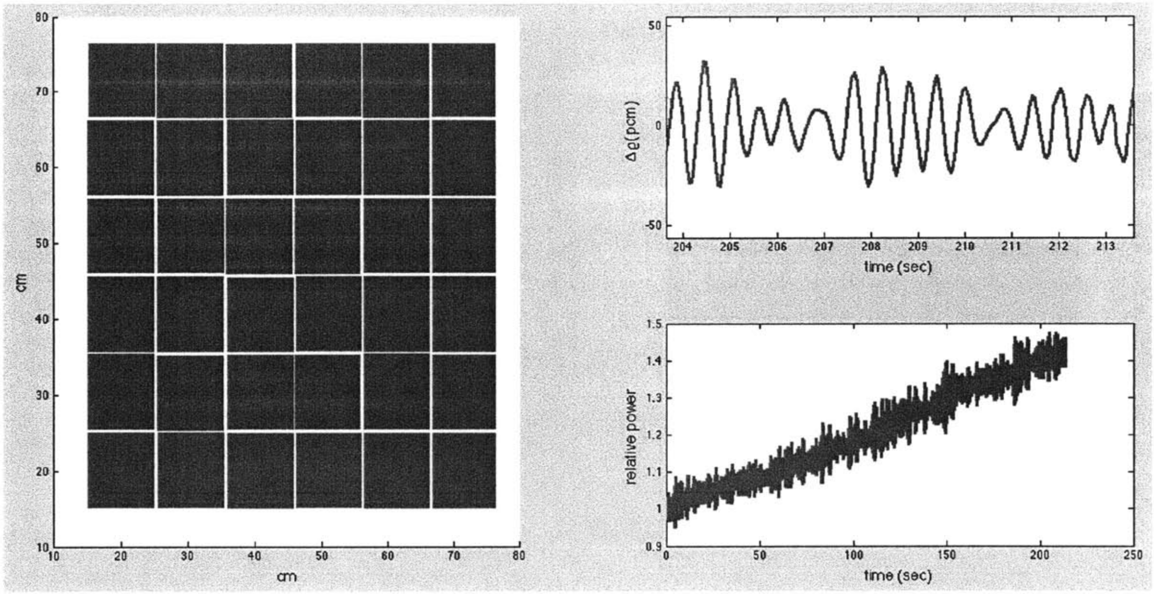

Figure 9.6: The PseuSei visualization tool for oscillatory seismic motion. The left screen shows the "pseudo-seismic" motion of a 6x6 assembly array. The upper right screen shows the reactivity $\rho(t)$ in pcm. The lower right screen shows the point kinetic power evolution $P(t)/P_0$.

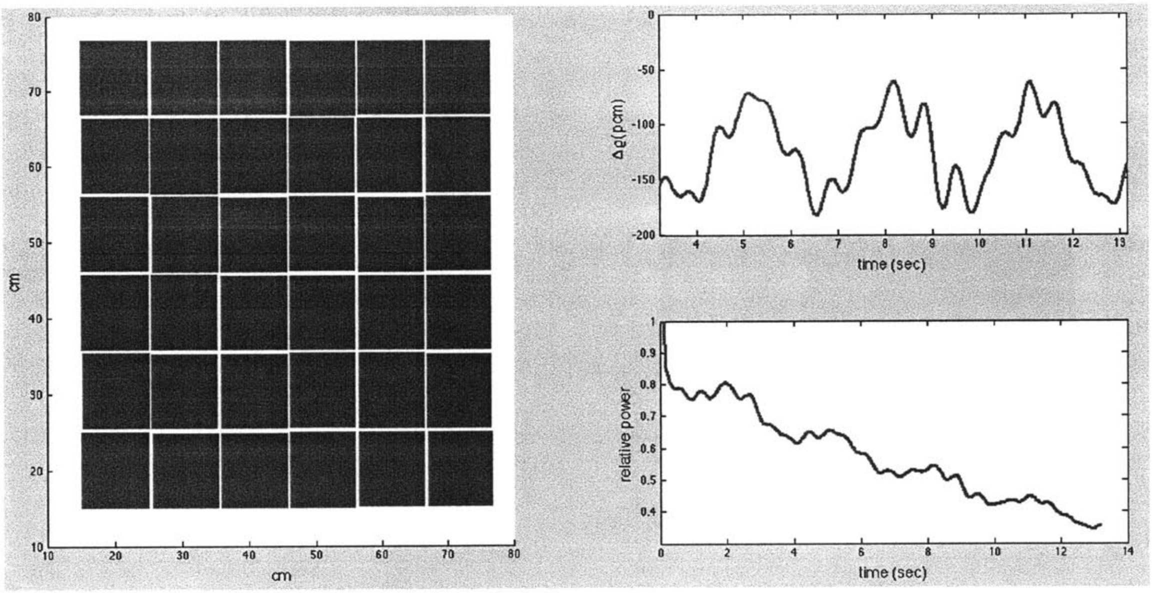

Figure 9.7: The PseuSei visualization tool for oscillatory seismic and flowering motion. We use a value $\alpha = 1$ in Equations 9.11 - 9.13. The left screen shows the "pseudo-seismic" motion of a 6x6 assembly array. The upper right screen shows the reactivity $\rho(t)$ in pcm. The lower right screen shows the point kinetic power evolution $P(t)/P_0$.

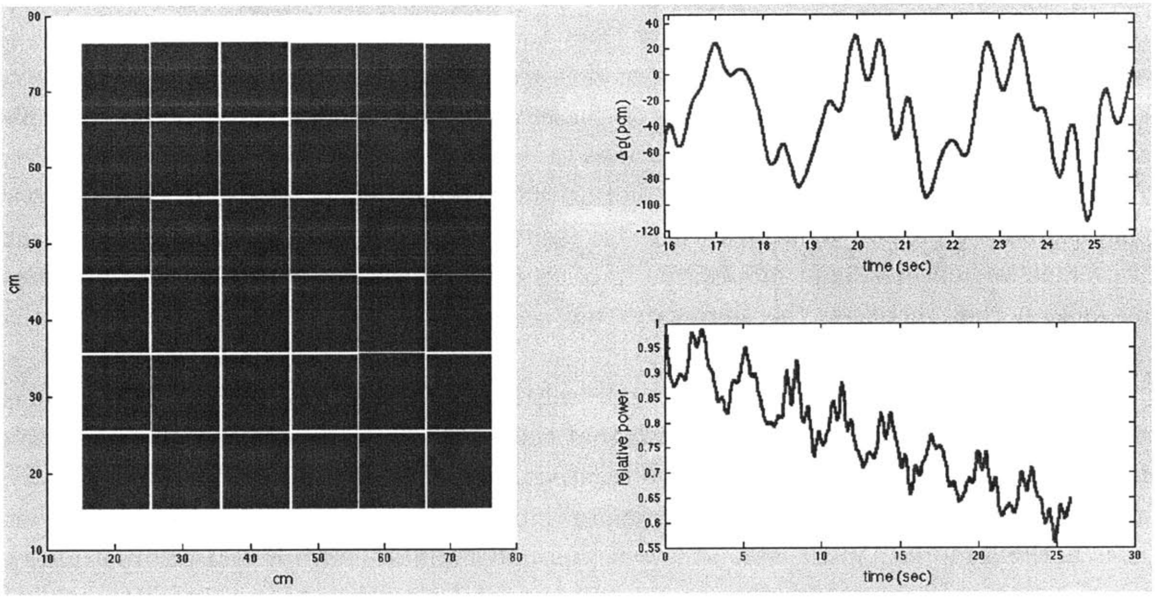

Figure 9.8: The PseuSei visualization tool for oscillatory seismic and flowering motion. We use a value $\alpha = 4$ in Equations 9.11 - 9.13. The left screen shows the "pseudo-seismic" motion of a 6x6 assembly array. The upper right screen shows the reactivity $\rho(t)$ in pcm. The lower right screen shows the point kinetic power evolution $P(t)/P_0$.

9.5 Summary

Although inserting material density perturbation reactivities into point kinetic theory has been common practice for decades, we have not seen this done for *geometry* perturbation reactivities. The PseuSei Animator serves as an instructive tool allowing one to see assembly motion with power evolution in real time.

In terms of safety, all PseuSei simulations show that reactivities never approach β and that prompt-criticality is therefore not a concern. As we showed in Section 7.7, enlarging the core to include more assemblies would only decrease the reactivity coefficient of each individual assembly movement. Thus, performing this calculation with such a small core produces reactivities that are much larger than one would see in a realistic core with many more assemblies. However, realistic core models may have much smaller values of β due to fission of ^{239}Pu and/or ^{233}U. Thus, based on these results, we cannot claim categorically that realistic cores would be "safer".

If we consider only oscillatory seismic motion of individual assemblies, then the core power oscillates about its initial value with either no trend or perhaps a very gradual upward drift. However, if we consider aggregate seismic motion that induces oscillatory flowering motion (any motion that increases the aggregate core radius), this typically inserts net negative reactivity and causes core power to fall.

One potential danger could be aggregate seismic motion in which multiple assemblies flower or bow toward the core center (such that the aggregate core radius decreases); such a scenario would insert sizable net positive reactivity. Fortunately, assemblies almost always tend to "spread out" rather than become more tightly clustered during distortion scenarios. This is nothing more than the second law of thermodynamics - entropy and disorder always tend to increase. However, whether or not any inward flowering or bowing might occur is a problem for mechanical analysis.

10 The VirDenT-MEPT Method: Distorted Flux and Power Distributions

10.1 Chapter Abstract

We seek to compute flux and power distributions due to geometry changes (distortions) in reactor cores. Directly solving a diffusion eigenvalue problem for non-uniform anisotropic "virtual density" swellings is not straightforward. Thus, we develop a new method for computing distorted flux distributions indirectly via perturbation theory. This method integrates non-uniform anisotropic "virtual density" theory (VirDenT) into modal (harmonic) expansion techniques. Previously, modal expansion perturbation theory (MEPT) has been applied only to material density changes. Fortunately and conveniently, the mathematical formulations of VirDenT and MEPT are remarkably compatible - the MEPT formalism fits the VirDenT formalism "like a glove". Here we lay out the theory for this hybrid VirDenT-MEPT method and provide numerous numeric examples for a large, highly-heterogeneous sodium fast reactor core in Cartesian geometry.

10.2 Introduction

Previously, we developed the non-uniform anisotropic "virtual density" theory (VirDenT) from fundamental conceptual reasoning. However, we have only applied it to compute reactivity (eigenvalue change), which is a whole-core integral quantity. In order to fully exploit the capabilities of VirDenT, we wish to obtain *spatial distributions* of various quantities - flux, power, and reaction rates.

We could compute these directly via flux tallies in Monte Carlo, but such an approach is prohibitively expensive. Building a hyperfine distorted mesh in diffusion or deterministic transport would be more reasonable but still quite expensive - both in terms of CPU time and development time necessary to build the mesh. One might *think* we could directly solve diffusion on a relatively coarse mesh using non-uniform anisotropic "virtual density", but doing so is not straightforward. There is no simple way to incorporate the surface leakage Γ_i terms into a direct diffusion solution.

10.3 The Challenge of VirDenT in Traditional Second Order (Generalized) Perturbation Theory

Traditional generalized perturbation theory for material density changes typically employs Stacey's well-known functional [155]:

$$\left(\tfrac{1}{k}\hat{F} - \hat{M}\right) d\phi = \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi + (\Delta\rho)\,\hat{F}\phi \tag{10.1}$$

Refer back to Section 2.3.5 for a derivation and definitions of variables. Here Equation 10.1 is identical to Equation 2.41.

As we discussed in Sections 2.3.5 and 2.7, Equation 10.1 is nothing more than a fixed source problem for which the flux shift $d\phi$ is the solution. Essentially, this requires a computational expense comparable to a single outer iteration in a generalized eigenvalue problem. This is a very large computation time for each perturbation, especially when fast reactor eigenvalue problems often require only a few outer iterations. If one wishes to perform hundreds or thousands of small geometry perturbations, the fixed source problems become expensive.

Furthermore, it is not immediately clear how to incorporate the anisotropic VirDenT formalism into Equation 10.1. The operator $d\hat{M}$ will be broken up into directional components, and there is no straightforward way to accomplish this. Although we can evaluate the adjoint-weighted anisotropic leakage quantities $\left\langle \phi^{\dagger} \middle| \hat{L}_i\phi \right\rangle$ without fuss, the vector quantities $\hat{L}_i\phi$ are not straightforward to evaluate within the VirDenT formalism.

Thus, we seek a more efficient and convenient method for obtaining perturbed flux and power distributions.

10.4 Modal Expansion Perturbation Theory (MEPT) Formalism

Serendipitously, the modal expansion perturbation theory (MEPT) formalism fits our "virtual density" theory (VirDenT) formalism "like a glove". See Section 2.2.3 for a conceptual explanation and literature review of modal expansion techniques. Mitani originally developed modal (or harmonic) expansion techniques in the 1970s [34]. In 2012, Touran and Lee revisited the topic and derived a new set of equations that allow for the contribution of each mode to be computed independently of the other modes, thus enabling convenient parallelization [39,40]. Let us now review the MEPT formalism of Touran and Lee.

Let the perturbed real and adjoint fluxes be

$$\phi_0' = \phi_0 + d\phi \tag{10.2}$$

$$\phi_0^{\dagger\prime} = \phi_0^{\dagger} + d\phi^{\dagger} \tag{10.3}$$

Here the subscript 0 denotes the fundamental mode. The real and adjoint diffusion neutron balance equations for each mode n are

$$\hat{M}\phi_n = \frac{1}{k_n}\hat{F}\phi_n \tag{10.4}$$

$$\hat{M}^{\dagger}\phi_n^{\dagger} = \frac{1}{k_n}\hat{F}^{\dagger}\phi_n^{\dagger} \tag{10.5}$$

These real and adjoint problems have the same eigenvalues but different eigenvectors. Now the expression for the flux shift derived by Touran and Lee is [39,40]

$$d\phi = \sum_{n=1}^{\infty} a_n \phi_n \tag{10.6}$$

The coefficients a_n reflect the magnitude of the contribution of each mode n. In the context of traditional density perturbations, we can express them in terms of the perturbed operators $d\hat{F}$ and $d\hat{M}$.

$$a_n = -\frac{\left\langle \phi_n^{\dagger} \middle| \left(\frac{1}{k_0}d\hat{F} - d\hat{M}\right)\phi_0 \right\rangle}{\left\langle \phi_n^{\dagger} \middle| \left(\frac{1}{k_0}\hat{F} - \hat{M}\right)\phi_n \right\rangle} = \frac{\left\langle \phi_n^{\dagger} \middle| \left(\frac{1}{k_0}d\hat{F} - d\hat{M}\right)\phi_0 \right\rangle}{\left(\frac{1}{k_n} - \frac{1}{k_0}\right)\left\langle \phi_n^{\dagger} \middle| \hat{F}\phi_n \right\rangle} \tag{10.7}$$

Note that $a_0 = 0$ by convention. We can write similar expressions for the perturbed adjoint flux.

$$d\phi^{\dagger} = \sum_{n=1}^{\infty} a_n^{\dagger} \phi_n^{\dagger} \tag{10.8}$$

$$a_n^{\dagger} = -\frac{\left\langle \phi_n \middle| \left(\frac{1}{k_0}d\hat{F}^{\dagger} - d\hat{M}^{\dagger}\right)\phi_0^{\dagger} \right\rangle}{\left\langle \phi_n \middle| \left(\frac{1}{k_0}\hat{F}^{\dagger} - \hat{M}^{\dagger}\right)\phi_n^{\dagger} \right\rangle} = \frac{\left\langle \phi_n \middle| \left(\frac{1}{k_0}d\hat{F}^{\dagger} - d\hat{M}^{\dagger}\right)\phi_0^{\dagger} \right\rangle}{\left(\frac{1}{k_n} - \frac{1}{k_0}\right)\left\langle \phi_n \middle| \hat{F}^{\dagger}\phi_n^{\dagger} \right\rangle} \tag{10.9}$$

This constitutes the MEPT formalism as it currently stands. We can apply it to any material density perturbation as shown by Touran and Lee. However, what we *really* desire is to apply it to geometry changes.

10.5 The Elegant Union of "Virtual Density" Perturbation Theory (VirDenT) and Modal Expansion Perturbation Theory (MEPT)

Conveniently, this MEPT formalism fits our VirDenT formalism like a cocoon its caterpillar. We will dub this symbiotic union the **VirDenT-MEPT method**.

10.5.1 VirDenT-MEPT Shorthand Notation

First we define some shorthand notation similar to what we defined for non-uniform anisotropic VirDenT in Sections 4.4.2 and 5.3.2. Let S_n be the spectral contribution of each mode n. Let $L_{n,i}$ and $\Gamma_{n,i}$ be the leakage contributions of mode n in direction i.

$$S_n = \frac{\left\langle \phi_n^\dagger \middle| \left(\frac{1}{k_0}\hat{F} - \hat{A} - \hat{T} \right) \phi_0 \right\rangle}{\left(\frac{1}{k_n} - \frac{1}{k_0} \right) \left\langle \phi_n^\dagger \middle| \hat{F}\phi_n \right\rangle} \tag{10.10}$$

$$L_{n,i} = \frac{\left\langle \phi_n^\dagger \middle| \hat{L}_i\phi_0 \right\rangle}{\left(\frac{1}{k_n} - \frac{1}{k_0} \right) \left\langle \phi_n^\dagger \middle| \hat{F}\phi_n \right\rangle} \tag{10.11}$$

$$\Gamma_{n,i} = \frac{\left\langle \phi_n^\dagger \middle| \hat{\Gamma}_i\phi_0 \right\rangle}{\left(\frac{1}{k_n} - \frac{1}{k_0} \right) \left\langle \phi_n^\dagger \middle| \hat{F}\phi_n \right\rangle} \tag{10.12}$$

The explicit integrals in the numerators of S_n, L_n, and Γ_n are

$$\left\langle \phi_n^\dagger \middle| \hat{L}_i\phi_0 \right\rangle = \int dV \left[\sum_{g=1}^{G} \left(\nabla\phi_{n,g}^\dagger \right)_i \left(\nabla\phi_{0,g} \right)_i D_g \right] \tag{10.13}$$

$$\left\langle \phi_n^\dagger \middle| \hat{\Gamma}_i\phi_0 \right\rangle = \int \left(d\vec{S} \cdot \hat{i} \right) \left[\sum_{g=1}^{G} \phi_{n,g}^\dagger \left(\nabla\phi_{0,g} \right)_i D_g \right] \tag{10.14}$$

$$\left\langle \phi_n^\dagger \middle| \hat{F}\phi_0 \right\rangle = \int dV \left[\sum_{g'=1}^{G} \chi_{g'}\phi_{n,g'}^\dagger \sum_{g=1}^{G} \nu_g \Sigma_{f,g} \phi_{0,g} \right] \tag{10.15}$$

We can define corresponding adjoint quantities $S_n^\dagger$, $L_{n,i}^\dagger$, and $\Gamma_{n,i}^\dagger$.

$$S_n^\dagger = \frac{\left\langle \phi_n \middle| \left(\frac{1}{k_0}\hat{F}^\dagger - \hat{A}^\dagger - \hat{T}^\dagger \right) \phi_0^\dagger \right\rangle}{\left(\frac{1}{k_n} - \frac{1}{k_0} \right) \left\langle \phi_n \middle| \hat{F}^\dagger\phi_n^\dagger \right\rangle} \tag{10.16}$$

$$L_{n,i}^\dagger = \frac{\left\langle \phi_n \middle| \hat{L}_i^\dagger\phi_0^\dagger \right\rangle}{\left(\frac{1}{k_n} - \frac{1}{k_0} \right) \left\langle \phi_n \middle| \hat{F}\phi_n^\dagger \right\rangle} \tag{10.17}$$

$$\Gamma_{n,i}^\dagger = \frac{\left\langle \phi_n \middle| \hat{\Gamma}_i^\dagger\phi_0^\dagger \right\rangle}{\left(\frac{1}{k_n} - \frac{1}{k_0} \right) \left\langle \phi_n \middle| \hat{F}\phi_n^\dagger \right\rangle} \tag{10.18}$$

The explicit integrals in these adjoint numerators are

$$\left\langle \phi_n \middle| \hat{L}_i^\dagger \phi_0^\dagger \right\rangle = \int dV \left[\sum_{g=1}^{G} (\nabla \phi_{n,g})_i \left(\nabla \phi_{0,g}^\dagger \right)_i D_g \right] \tag{10.19}$$

$$\left\langle \phi_n \middle| \hat{\Gamma}_i^\dagger \phi_0^\dagger \right\rangle = \int \left(d\vec{S} \cdot \hat{i} \right) \left[\sum_{g=1}^{G} \phi_{n,g} \left(\nabla \phi_{0,g}^\dagger \right)_i D_g \right] \tag{10.20}$$

$$\left\langle \phi_n \middle| \hat{F}^\dagger \phi_0^\dagger \right\rangle = \int dV \left[\sum_{g=1}^{G} \phi_{n,g} \nu_g \Sigma_{f,g} \sum_{g'=1}^{G} \chi_{g'} \phi_{0,g'}^\dagger \right] \tag{10.21}$$

Since we have not previously defined the adjoint absorption and scattering inner products, they are

$$\left\langle \phi_n \middle| \hat{A}^\dagger \phi_0^\dagger \right\rangle = \int dV \left[\sum_{g=1}^{G} \phi_{n,g} \Sigma_{a,g} \phi_{0,g}^\dagger \right] \tag{10.22}$$

$$\left\langle \phi_n \middle| \hat{T}^\dagger \phi_0^\dagger \right\rangle = \int dV \left[\sum_{g=1}^{G} \sum_{g'=g+1}^{G} \left(\phi_{n,g'} - \phi_{n,g} \right) \Sigma_{s,g \to g'} \phi_{0,g'}^\dagger \right] \tag{10.23}$$

Note that the operators in the numerator of S_n contain the fundamental eigenvalue k_0 while operating on the fundamental mode ϕ_0. Thus, even though the adjoint weighting in S_n is *not* the fundamental mode, we can still write the equality

$$S_n = \sum_i \left(L_{n,i} - \Gamma_{n,i} \right) \tag{10.24}$$

In general, this equality holds when the operators in the numerator contain the eigenvalue corresponding to the mode they operate on (the mode in the right-hand "ket" of the inner product). The weighting quantity (the mode in the left-hand "bra" of the inner product) can be absolutely any vector quantity, as long as it is consistent for S_n, $L_{n,i}$, and $\Gamma_{n,i}$. It need not even be an eigenvector solution at all! Similarly, the denominator can be absolutely any scalar quantity, as long as it is consistent.

Obviously, can write the same equality for the adjoint quantities:

$$S_n^\dagger = \sum_i \left(L_{n,i}^\dagger - \Gamma_{n,i}^\dagger \right) \tag{10.25}$$

10.5.2 Flux Shift Coefficients and Power Shift Coefficients

If we already know the VirDenT formalism and the MEPT formalism, fusing them is straightforward. The derivative of the flux distribution with respect to the swelling fraction f is

$$\frac{d\phi}{df} = \frac{d\phi_0(\vec{r}, E)}{df} \tag{10.26}$$

There is no ambiguity in expressing this as simply $d\phi/df$, because we only ever care about flux shifts for the fundamental mode ($n = 0$). The derivative of the power distribution with respect to the swelling fraction f is

$$\frac{dP}{df} = \frac{dP(\vec{r})}{df} = \sum_{g=1}^{G} \Sigma_{f,g}(\vec{r}) \frac{d\phi_{0,g}(\vec{r})}{df} \tag{10.27}$$

We will express this simply as dP/df.

These two coefficients - the flux and power shifts - encapsulate everything we can know from VirDenT-MEPT. Once we globally know the first order change in the multigroup flux, we can globally know the first order change in any reaction rate of interest.

10.5.3 VirDenT-MEPT Formalism

Suppose that an arbitrary interior zone within a reactor swells anisotropically in direction i by a factor f_i. We can express the flux shift coefficient as

$$\frac{d\phi}{df_i} = \sum_{n=1}^{\infty} \frac{da_n}{df_i} \phi_n \tag{10.28}$$

Note that a_n has a non-zero derivative with respect to f_i, but the unperturbed ϕ_n does not. The derivative da_n/df_i is simple to express via the VirDenT formalism.

$$\frac{da_n}{df_i} = \sum_{j \neq i} (-2L_{n,j} + \Gamma_{n,j}) \tag{10.29}$$

Here we have done nothing more than take the VirDenT formalism (Equation 5.9) and plug it straight into the MEPT formalism (Equation 10.7) via the shorthand notation (Equations 10.11 and 10.12). So the flux shift coefficient in one expression is

$$\frac{d\phi}{df_i} = \sum_{n=1}^{\infty} \left[\sum_{j \neq i} (-2L_{n,j} + \Gamma_{n,j}) \right] \phi_n \tag{10.30}$$

The *magnitude* of the flux shift for an arbitrary swelling of ϵ_i in each direction i is

$$d\phi = \sum_{n=1}^{\infty} \left[\sum_{i} \epsilon_i \sum_{j \neq i} (-2L_{n,j} + \Gamma_{n,j}) \right] \phi_n \tag{10.31}$$

The flux shift coefficient due to an anisotropic *expansion* in direction i is

$$\frac{d\phi}{df_i} = \sum_{n=1}^{\infty}\left[S_n + L_{n,i} + \sum_{j\neq i}\left(-L_{n,j} + \Gamma_{n,j}\right)\right]\phi_n \tag{10.32}$$

The magnitude of the flux shift for an arbitrary expansion is

$$d\phi = \sum_{n=1}^{\infty}\left[\sum_{i}\epsilon_i\left(S_n + L_{n,i} + \sum_{j\neq i}\left(-L_{n,j} + \Gamma_{n,j}\right)\right)\right]\phi_n \tag{10.33}$$

Now let us briefly consider hexagonal-Z or R-Z geometry. The flux shift coefficient due to non-uniform axial swelling is

$$\frac{d\phi}{df_z} = \sum_{n=1}^{\infty}\left(-2L_{n,r} + \Gamma_{n,r}\right)\phi_n \tag{10.34}$$

The flux shift coefficient due to non-uniform *radial* swelling is

$$\frac{d\phi}{df_r} = \sum_{n=1}^{\infty}\left(-4L_{n,z} + 2\Gamma_{n,z} - 2L_{n,r} + \Gamma_{n,r}\right)\phi_n \tag{10.35}$$

This constitutes the basic VirDenT-MEPT formalism. Once we have obtained these flux shift coefficients, the power shift coefficients are straightforward to obtain via Equation 10.27. We could write down many, *many* more expressions, but what we have shown here is sufficient to give the reader a feel for the theory. If one understands the concept, the expressions are obvious.

10.6 Numeric Self-Consistency of VirDenT-MEPT

Now that we have laid out the VirDenT-MEPT formalism, we aim to validate it for non-uniform anisotropic swellings in a realistic core model.

10.6.1 A Large, Asymmetric, and Highly-Heterogeneous Cartesian Core

Most past applications of MEPT for material density perturbations have consisted of either (1) nearly homogeneous cores or (2) small segments of cores. The main demonstration case studied by Touran and Lee is a bare homogeneous sodium fast reactor in hexagonal-z geometry [39,40]. However, Touran's thesis *does* include modal analysis with a core that contains control and shield assemblies [39].

In order to eviscerate any doubts regarding the veracity of MEPT in heterogeneous cores, we opt to construct a reference core that is both large and very heterogeneous. See Figure 10.1, which displays the macroscopic fission cross-section through an axial slice of this core. This Cartesian sodium fast reactor core has a 10x10 array of square assemblies (each 20 cm on a side) surrounded by a thick U-238 reflector. The enrichment distribution is "checkerboard" to ensure a high degree of heterogeneity and radially-increasing to introduce some radial power flattening. The core geometry is constant through its axial height. The core height has 10 axial mesh cells, and each axial segment of each assembly is a single mesh cell. Thus, the fueled core is 10 x 10 x 10 mesh cells, and the whole core is 16 x 16 x 10 mesh cells. There are no axial reflectors and no explicit coolant gaps. We will use this test core for all VirDenT-MEPT validation throughout this chapter.

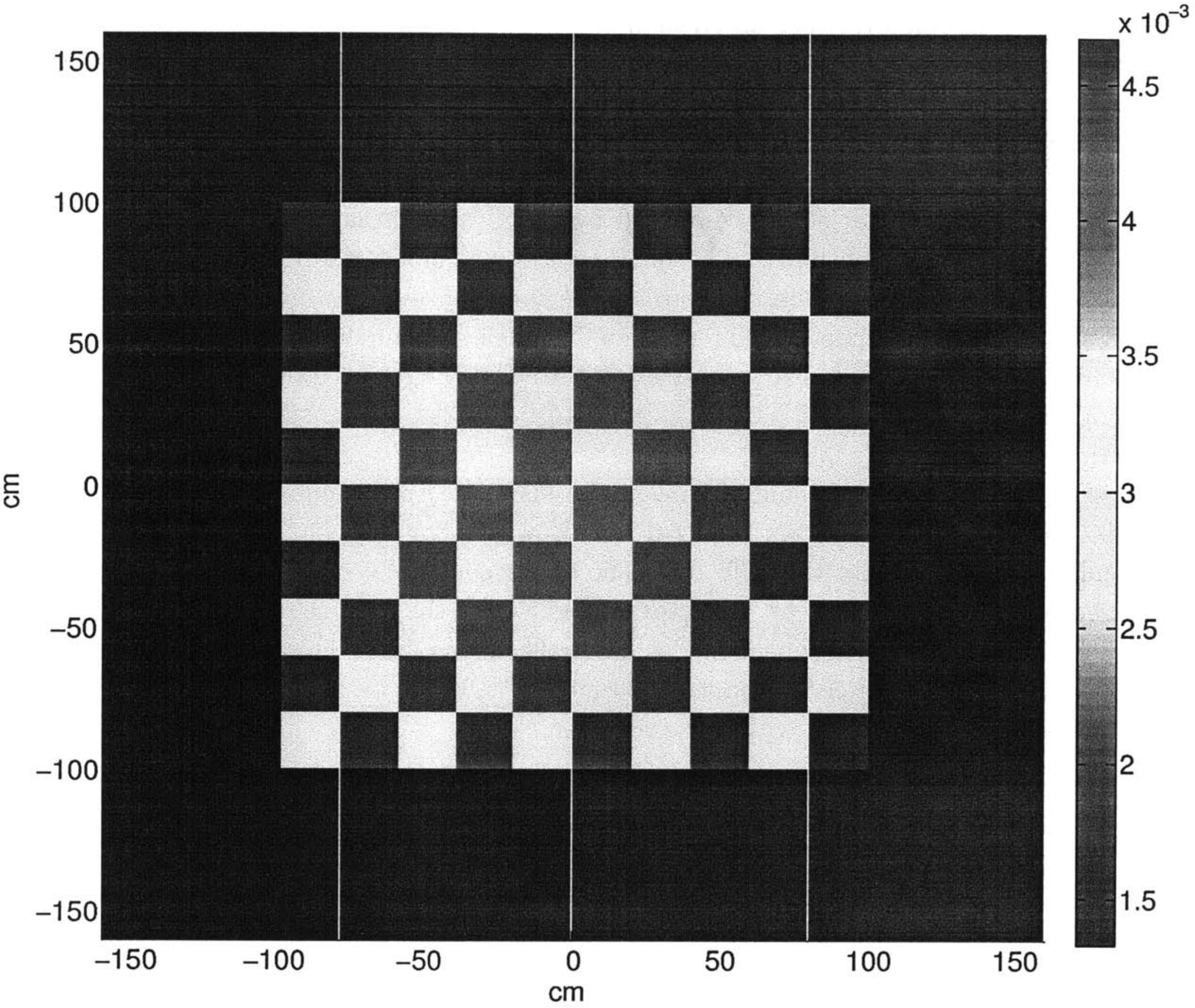

Figure 10.1: The homogenized macroscopic fission cross-section (units of 1/cm) at 1.7 MeV in each 20 cm x 20 cm assembly of our VirDenT-MEPT validation core. A U-238 metal reflector (three assemblies thick) blankets the core radially. The fueled region consists of 100 assemblies containing uranium metal and sodium with a coolant volume fraction of 0.38. A checkerboard enrichment pattern ensures strong heterogeneity, and radially-increasing enrichment flattens the power distribution.

10.6.2 The Harmonics and Their Eigenvalues

We construct this test core geometry in the MATLAB-PETSc-SLEPc (MaPS) solver (see Appendix A). We employ either an Arnoldi or a Krylov-Schur eigenvalue solver to obtain any arbitrary number of flux harmonics. We use the same MC**2 [187] 33-group microscopic cross-section set used for the Cartesian cores in Chapter 7. This test core has a total of 16 x 16 x 10 = 2560 mesh cells, which is the maximum number of harmonics obtainable.

Figure 10.2 shows eigenvalues for the first 2501 modes. Figures 10.3 and 10.4 show the first 6 real and adjoint harmonics computed by MaPS and displayed in axial mesh cell 5 (mesh cell 1 is the core bottom). The fundamental real mode (the real flux) shows that the checkerboard enrichment pattern pushes the flux slightly away from the core center, toward the upper-right direction. The harmonics often display checkerboard-like patterns themselves, as the mesh cells size is large enough to allow large variations in fluxes between neighboring cells. The adjoint harmonics tend to be spatially "smoother" than their real brethren.

Figure 10.5 shows the real and adjoint modes 965-967. An intriguing trend begins with n = 966. All previous modes are localized mainly within the fuel, but modes 966 and 967 are localized mainly within the reflector. The subsequent $\approx$ 50 modes alternate between these two types - "fuel modes" or "reflector modes". Once the total number of "fuel modes" has reached $\approx$ 1020, all subsequent modes are "reflector modes". This is because the 10 x 10 x 10 = 1000 fuel mesh cells produce 1000 modes. After those have been exhausted, all remaining modes are localized within the reflector.

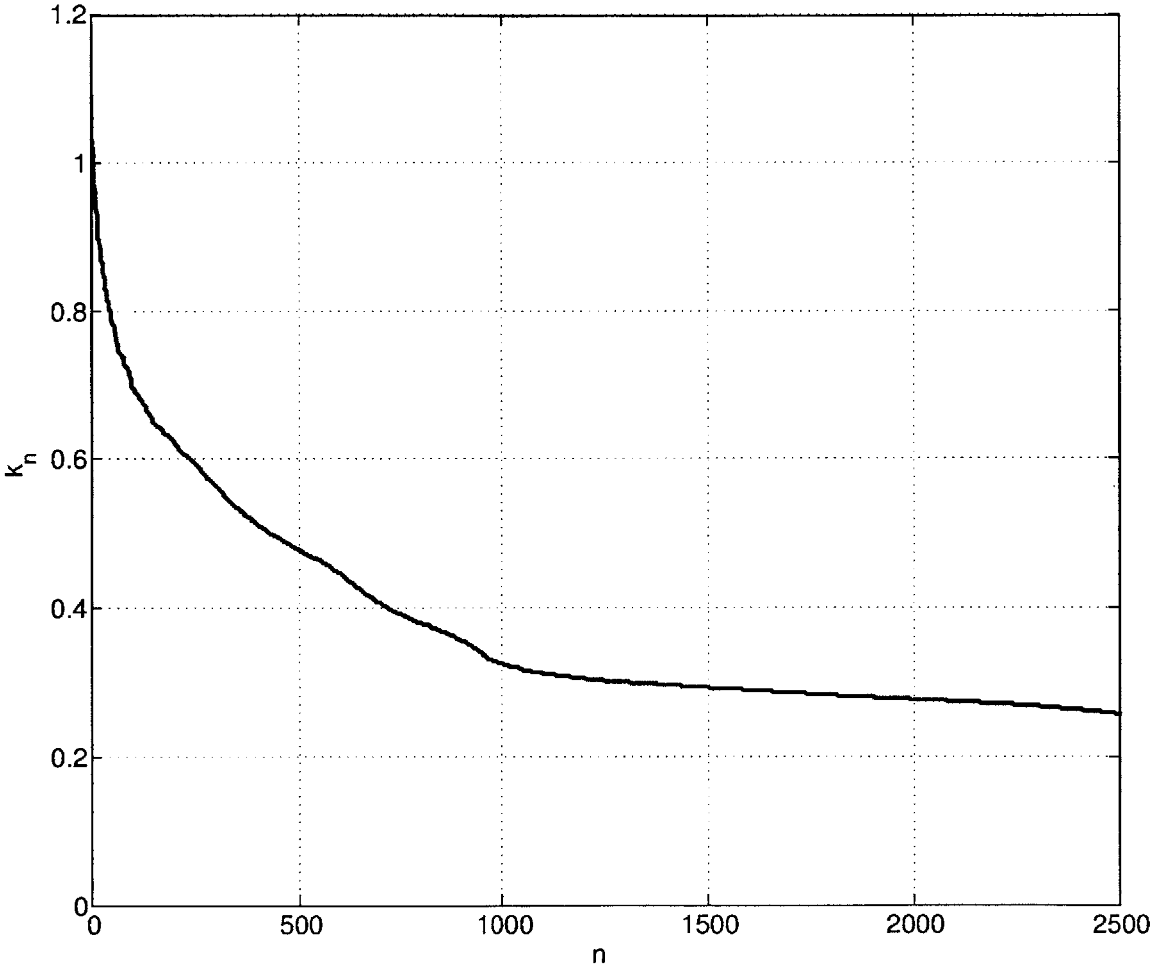

Figure 10.2: The eigenvalue k_n for the first 2501 modes ($n = 0$ to 2500). Here the dominance ratio is $k_1/k_0 = 0.9618$.

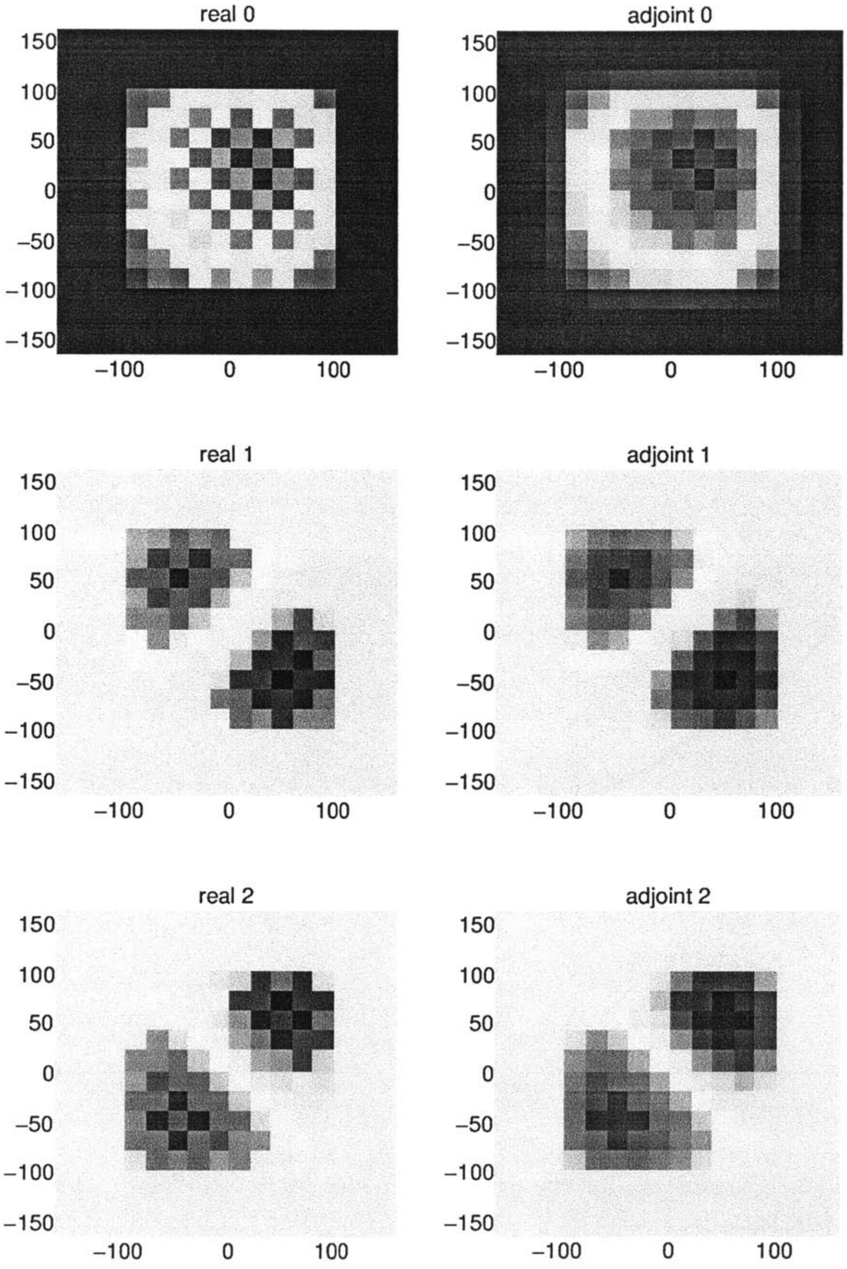

Figure 10.3: The first 3 real and adjoint modes (n = 0,1,2) in axial mesh cell 5 at approximately 1.7 MeV.

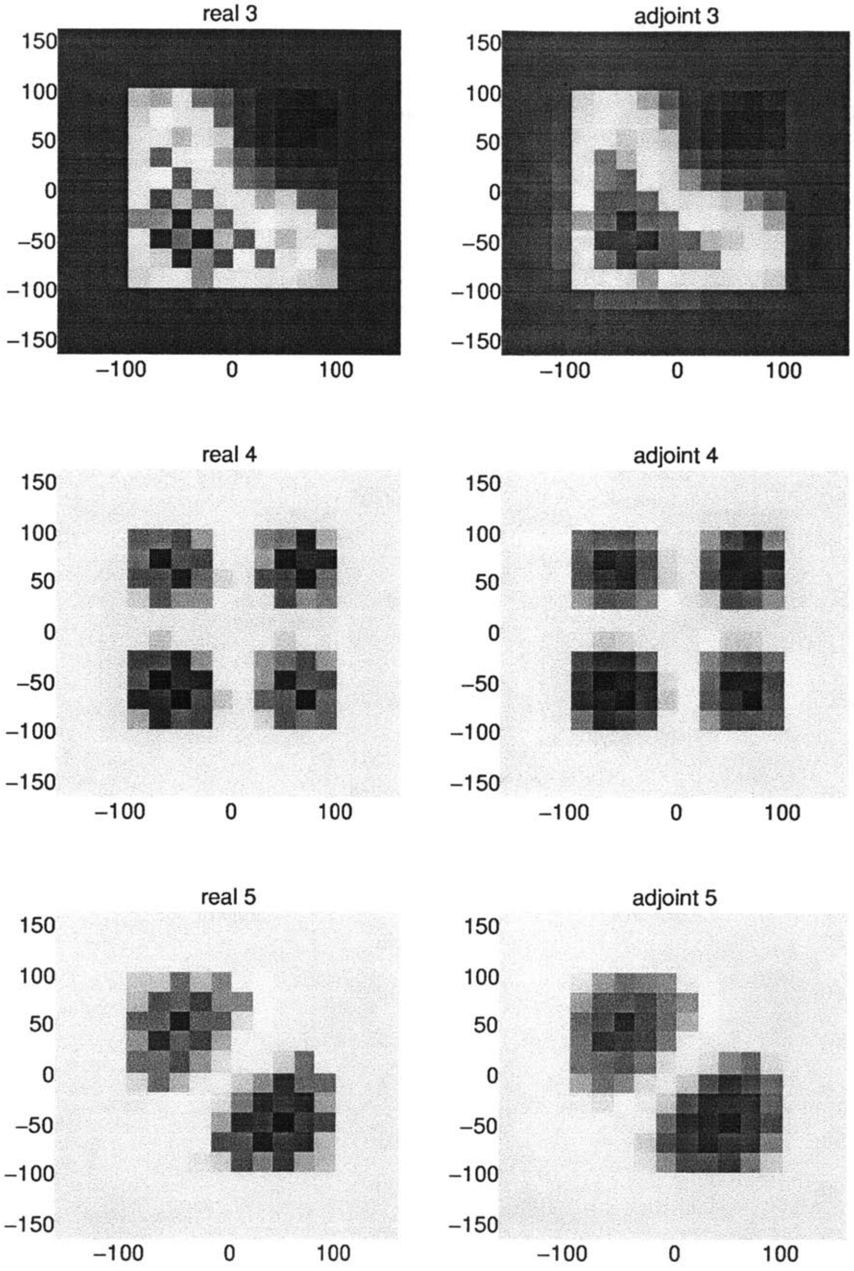

Figure 10.4: The second 3 real and adjoint modes (n = 3,4,5) in axial mesh cell 5 at approximately 1.7 MeV.

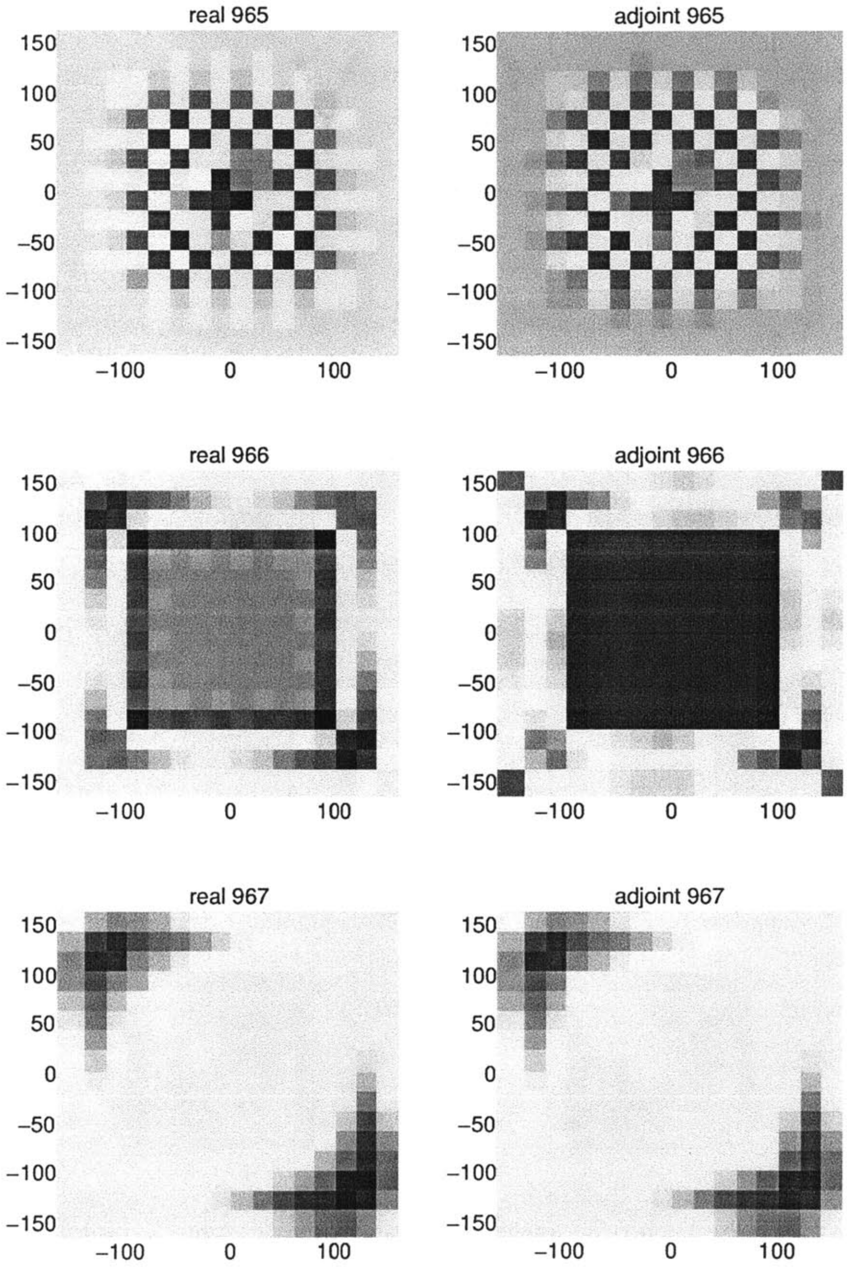

Figure 10.5: Real and adjoint modes for n = 965,966,967 in axial mesh cell 5 at approximately 1.7 MeV. The "reflector modes" begin at n = 966.

10.6.3 Verification of Adjoint Operator Construction

The veracity of MEPT is extremely sensitive to inaccuracies in operator construction. We detail the MaPS real and adjoint operator construction in Appendix A. We can ensure that the real and adjoint $\hat{M}$ and $\hat{F}$ operators are properly constructed and self-consistent with a few simple tests. The mathematical definition of any adjoint operator $\hat{Q}^\dagger$ requires

$$\left\langle \phi_n^\dagger \middle| \hat{Q}\phi_n \right\rangle = \left\langle \hat{Q}^\dagger \phi_n^\dagger \middle| \phi_n \right\rangle \tag{10.36}$$

So we can construct two sets of "error quantities" $\xi_{\mathrm{M,n}}$ and $\xi_{\mathrm{F,n}}$ to test the accuracy of adjoint construction for $\hat{M}$ and $\hat{F}$, respectively. We evaluate $\xi_{\mathrm{M,n}}$ and $\xi_{\mathrm{F,n}}$ for each mode:

$$\xi_{\mathrm{F,n}} = 1 - \frac{\left\langle \phi_n^\dagger \middle| \hat{F}\phi_n \right\rangle}{\left\langle \hat{F}^\dagger \phi_n^\dagger \middle| \phi_n \right\rangle} \tag{10.37}$$

$$\xi_{\mathrm{M,n}} = 1 - \frac{\left\langle \phi_n^\dagger \middle| \hat{M}\phi_n \right\rangle}{\left\langle \hat{M}^\dagger \phi_n^\dagger \middle| \phi_n \right\rangle} \tag{10.38}$$

Figure 10.6 shows $\xi_{\mathrm{M,n}}$ and $\xi_{\mathrm{F,n}}$ for the first 2500 modes computed with 10^{-12} eigenvalue convergence. These "error quantities" never exceed 2×10^{-13} in magnitude, leaving to doubt that our adjoint operator construction is mathematically precise. Note that these error magnitudes increase slightly near $n \approx 950$. This signifies where the "reflector modes" begin.

However, Equation 10.36 is not nearly the strongest form of the adjoint condition. This condition holds not only for vectors ϕ_n and $\phi_n^\dagger$, but also for *any arbitrary* vectors η_1 and η_2.

$$\left\langle \eta_1 \middle| \hat{Q}\eta_2 \right\rangle = \left\langle \hat{Q}^\dagger \eta_1 \middle| \eta_2 \right\rangle \tag{10.39}$$

We generate arbitrary η_1 and η_2 by obtaining each element from the MATLAB rand function[210]. Now we define two more "error quantities" $\xi_{\mathrm{M,rand}}$ and $\xi_{\mathrm{F,rand}}$:

$$\xi_{\mathrm{F,rand}} = 1 - \frac{\left\langle \eta_1 \middle| \hat{F}\eta_2 \right\rangle}{\left\langle \hat{F}^\dagger \eta_1 \middle| \eta_2 \right\rangle} = \tag{10.40}$$

$$\xi_{\mathrm{M,rand}} = 1 - \frac{\left\langle \eta_1 \middle| \hat{M}\eta_2 \right\rangle}{\left\langle \hat{M}^\dagger \eta_1 \middle| \eta_2 \right\rangle} \tag{10.41}$$

We gladly report that both $\xi_{\mathrm{M,rand}}$ and $\xi_{\mathrm{F,rand}}$ are less than 10^{-12}.

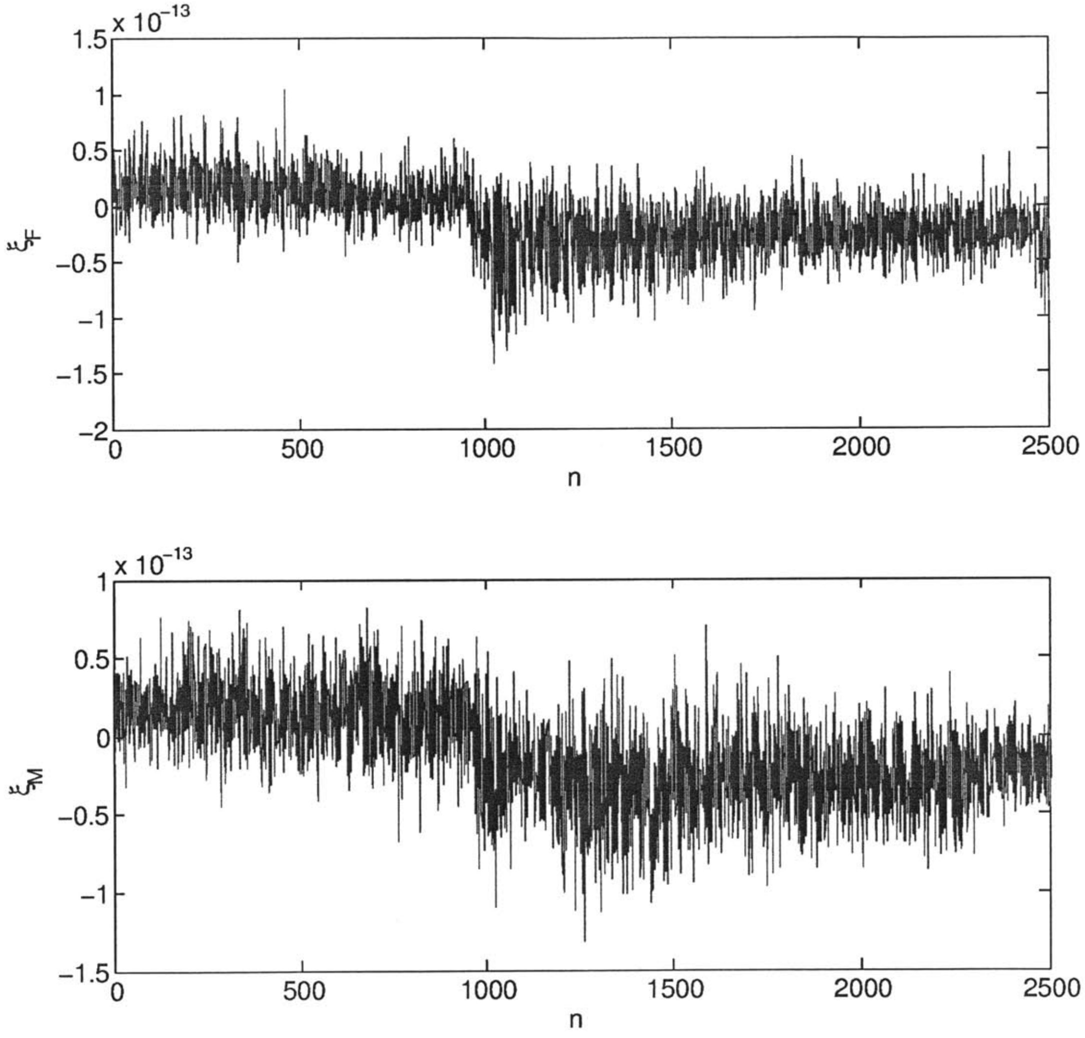

Figure 10.6: The errors ξ_F and ξ_M for the first 2500 modes. Here ξ_F is the fractional error between $\left\langle \phi_n^\dagger \middle| \hat{F}\phi_n \right\rangle$ and $\left\langle \hat{F}^\dagger\phi_n^\dagger \middle| \phi_n \right\rangle$, while ξ_M is the fractional error between $\left\langle \phi_n^\dagger \middle| \hat{M}\phi_n \right\rangle$ and $\left\langle \hat{M}^\dagger\phi_n^\dagger \middle| \phi_n \right\rangle$.

10.6.4 Harmonic Bi-Orthogonality and a Warning

In the previous section, we verified that $\hat{M}^\dagger$ and $\hat{F}^\dagger$ are the *mathematical* adjoints of $\hat{M}$ and $\hat{F}$. However, there is a significant distinction between a *mathematical* adjoint and a *physical* adjoint. There are actually multiple ways in which one can construct a mathematical adjoint but only *one* way in which one can construct a physical adjoint. All physical adjoints are mathematical adjoints, but not all mathematical adjoints are physical adjoints.

For example, simply transposing all elements in the $\hat{M}$ operator will yield a mathematical adjoint that satisfies Equations 10.36 and 10.39. However, doing so will *not* yield the physical adjoint - the mathematical adjoint produced will not represent neutron "importance". Constructing the physical adjoint requires transposing the scattering terms while leaving the leakage terms in place. This is because the diffusion leakage terms are self-adjoint, while the multigroup scattering terms are not.

One test for the physical adjoint is *bi-orthogonality.* The bi-orthogonality condition is

$$\left\langle \phi_m^\dagger \middle| \, \phi_n \right\rangle = C_n \delta_{mn} \tag{10.42}$$

Here m and n represent any two harmonic indices, and C_n is an arbitrary constant. This condition holds *only* for the physical adjoint - *not* any mathematical adjoint. If the harmonics are *bi-orthonormal,* then $C_n = 1$ for all n. Converting an bi-orthogonal set into a bi-orthonormal set requires only simple normalization. However, this normalization is unnecessary for MEPT, because the harmonic magnitudes cancel out when combining Equations 10.6 and 10.7.

One might say that when bi-orthogonal, the harmonics are "in harmony". Unfortunately, this "harmony" is sometimes arduous to achieve. Orthogonality and bi-orthogonality are notorious as numeric "stumbling blocks" across many fields of computation. Although bi-orthogonality is certainly a mathematical requirement, numeric eigenvalue solvers often introduce small errors that disturb the "harmony". We encountered two potential numeric pitfalls along these lines:

Potential Pitfall 1: Cores with any symmetry (axial, radial, or rotational) will produce some *degenerate* harmonics. Two harmonics are degenerate when they share the same eigenvalue but not necessarily the same eigenvector (spatial and energy distributions). As one might expect, degenerate modes often become entangled and therefore unusable. Instead of producing two degenerate eigenvectors, a solver might produce two linear superpositions of those eigenvectors. These linear superpositions will have correct eigenvalues but wrong spatial and energy distributions, and they will corrupt the modal expansion.

Solution 1: We introduce *slight* asymmetries into the core composition to ensure that degenerate harmonics do not occur. In our VirDenT-MEPT test core, the checkerboard enrichment pattern eliminates any possibility of radial, lateral, or rotational symmetry. Additionally, we introduce a *slight* linear enrichment increase along the core height to avoid any axial symmetry. These asymmetries need not be pronounced at all - they could be as minute as an 0.1% enrichment variation. Techniques for disentangling degenerate modes do exist, but for now we choose to bypass them via asymmetry.

Potential Pitfall 2: Many preconditioners and Krylov subspace methods employ various *diagonalization* techniques. These are convenient, because leakage, scattering, and fission terms in the $\hat{M}$ and $\hat{F}$ matrices typically lie along a relatively small number of diagonals. For example, an Q-D Cartesian reactor with G energy groups and no up-scattering will have $2Q + G + 1$ diagonals in its $\hat{M}$ matrix and $2Q$ in its $\hat{F}$ matrix. All non-zero elements will lie along these diagonals. However, non-Cartesian geometries introduce more ambiguity in terms of how one chooses to index the spatial mesh cells. We chose a non-conventional indexing convention for triangle-z geometry (shown in Figures A.1 and A.2), which causes leakage terms to form exotic patterns (see Figures A.7 and A.8) instead of neat and tidy diagonals. This is perfectly fine from a mathematical standpoint - all indexing conventions are equally valid and *should* work equally well. Unfortunately, indexing conventions affect *spectral radius*, which can affect the quality of preconditioning.

Solution 2: We have three choices. First, we can comb through all available preconditioner to find one that we know will be accurate for our particular indexing convention. Second, we can re-index all triangle-z mesh cells such that non-zero leakage terms will lie on only a few diagonals. This is how the code DIF3D indexes triangles [193]. Third, we can simply choose to implement VirDenT-MEPT in a core with Cartesian geometry, for which diagonal leakage terms are always most convenient.

Figure 10.7 shows the bi-orthogonality condition in Equation 10.42 for the first 100 harmonics computed via MEPT for the Cartesian test core in Figure 10.1. Clearly, inner products in which $m = n$ dwarf those in which $m \neq n$. Figure 10.8 shows the same bi-orthogonality condition for harmonics 2001-2100. Again, the condition appears met.

However, we must note that mere visual inspection of Figures 10.7 and 10.8 is not sufficient to conclude that bi-orthogonality has been achieved, because MEPT is *extremely* sensitive to the slightest errors in bi-orthogonality. The entire derivation by Touran and Lee hinges upon precise bi-orthogonality.

The reality is that even in Cartesian geometry, MaPS is plagued by tiny bi-orthogonality errors. Although the bi-orthogonality appears very clean in Figure 10.7, a very small number of $m \neq n$ terms actually approach (or even exceed) 1% of their corresponding $m = n$ terms. This 1% may seem small, but it becomes surprisingly consequential when computing the a_n coefficients and converging $d\phi$.

Despite this bi-orthogonality plague, we are comfortable enough with maximum bi-orthogonality errors of 10^{-2} (and mean errors of 10^{-6} or 10^{-7}) to proceed. Nevertheless, we acknowledge that these errors may limit VirDenT-MEPT accuracy, and we caution all MEPT users to beware of slight bi-orthogonality errors.

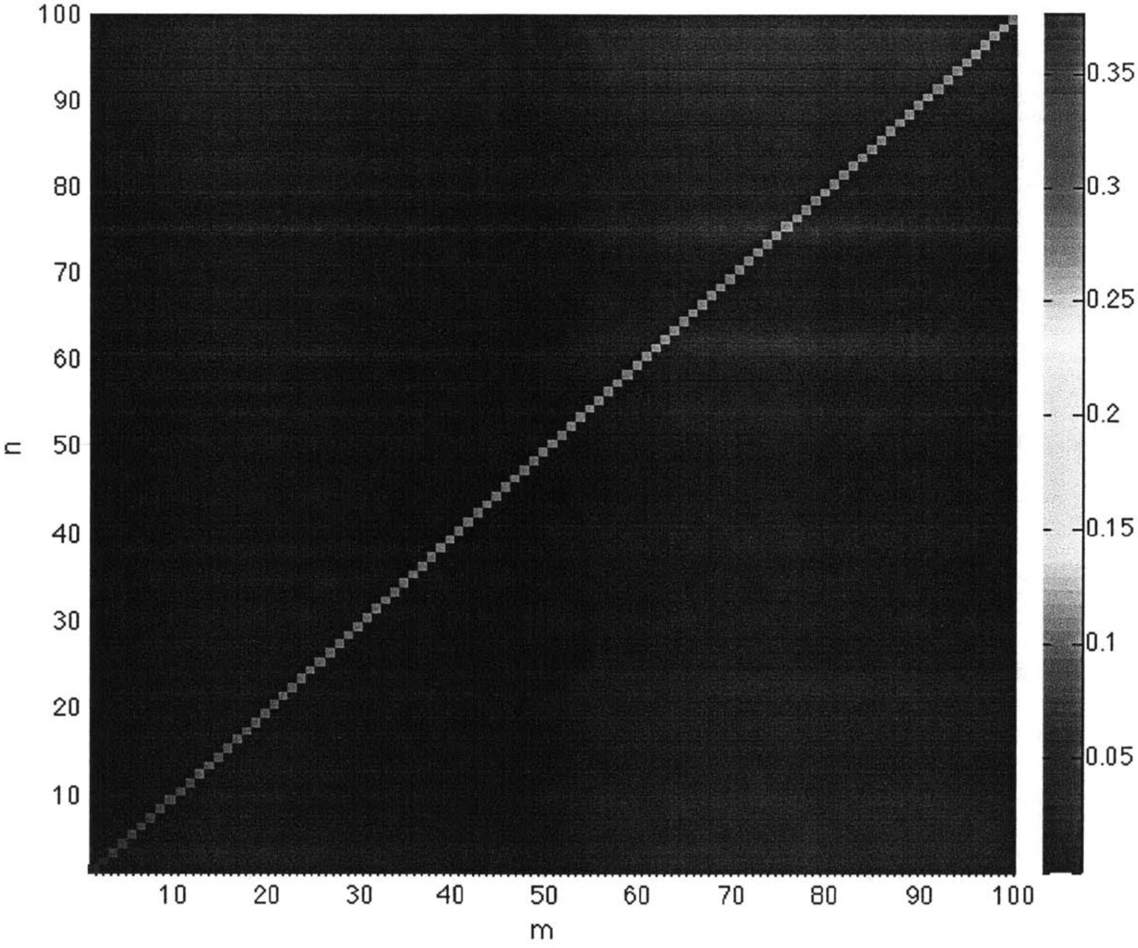

Figure 10.7: All values of the inner product $\left\langle \phi_m^\dagger \middle| \phi_n \right\rangle$ for m and n ranging from 0 to 99. This product is very close to zero except when $m = n$, which verifies the bi-orthogonality condition.

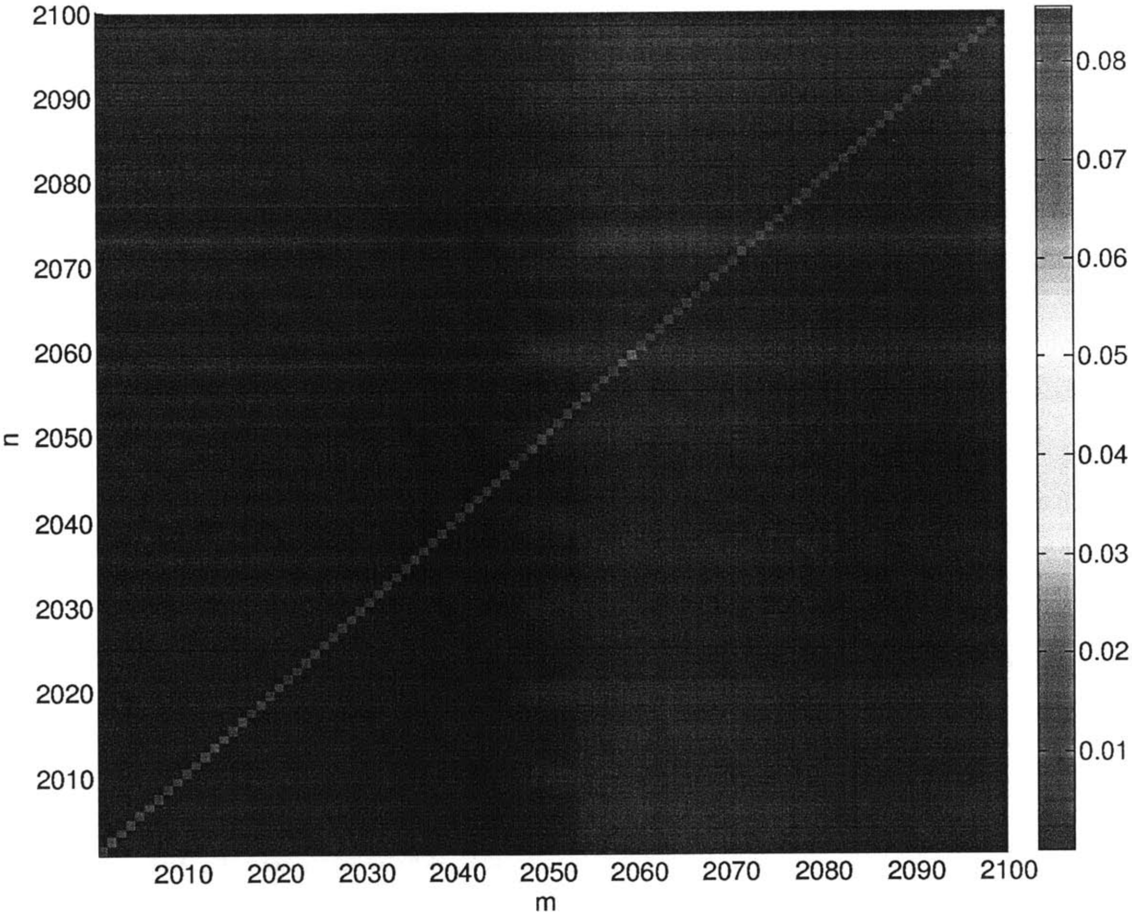

Figure 10.8: All values of the inner product $\left\langle \phi_m^\dagger \middle| \phi_n \right\rangle$ for m and n ranging from 2000 to 2099. This product is very close to zero except when $m = n$, which verifies the bi-orthogonality condition for these higher modes.

10.6.5 Numeric Self-Consistency of the VirDenT-MEPT Formalism

Before proceeding on to numeric validation of distortion scenarios, we are prudent to check the self-consistency of the VirDenT-MEPT formalism derived in Sections 10.5.1 and 10.5.3. This formalism hinges upon the equality in Equation 10.24. We can test this equality for each harmonic by defining the "error quantity" $\xi_{\mathrm{VirDenT},n}$:

$$\xi_{\mathrm{VirDenT},n} = 1 - \frac{L_n - \Gamma_n}{S_n} \tag{10.43}$$

This quantity should be as close as possible to zero for every harmonic. Figure 10.9 shows that it is always less than 4×10^{-4} for the first 2500 harmonics. Its mean is 2.7×10^{-6}, and its standard deviation is 3.2×10^{-5}. We are satisfied with this level of accuracy for the VirDenT-MEPT calculations.

Note that the magnitudes of the $\xi_{\mathrm{VirDenT},n}$ peaks increase substantially just before $n =$ 1000. This is because the "reflector modes" are a bit less well converged than the "fuel modes".

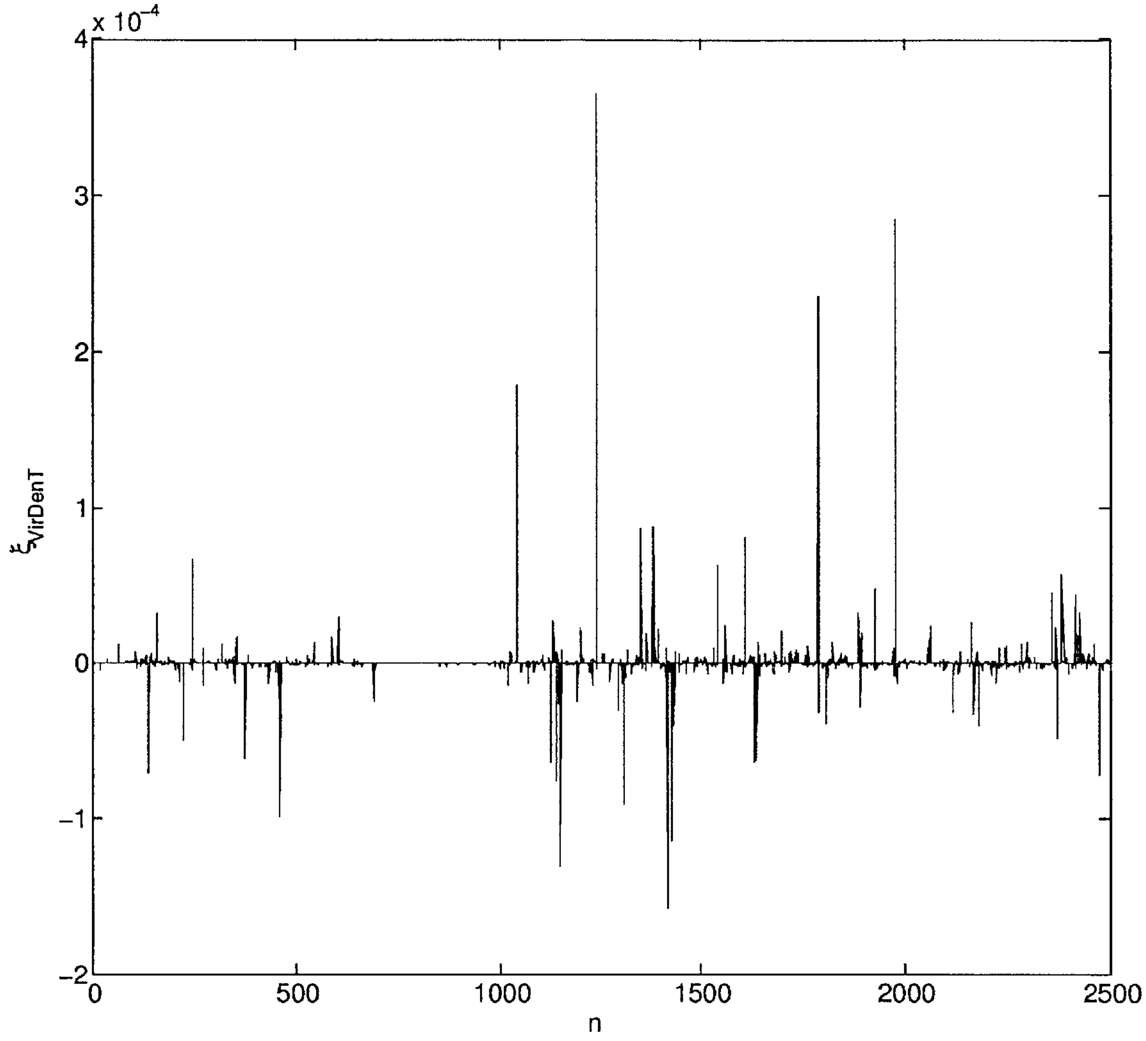

Figure 10.9: The error ξ_{VirDenT} for the first 2500 modes. Here ξ_{VirDenT} is the core-wide volume-averaged fractional error between S_n and $L_n - \Gamma_n$. Despite the occasional large spike, the *magnitude* of ξ_{VirDenT} has a mean of 2.7×10^{-6} and a standard deviation of 3.2×10^{-5}.

10.7 Numeric Validation of VirDenT-MEPT

Touran and Lee have already validated MEPT for material density perturbations, around which the whole history of modal expansion revolves[39,40]. Now we aim to validate our new VirDenT-MEPT formalism to show how MEPT can also work for geometry perturbations.

We use the same Cartesian test core discussed in Section 10.6.1 and illustrated in Figure 10.1. In order to construct a reference for $d\phi$, we obtain the perturbed and unperturbed fluxes from MaPS and normalize them both to the *same power*. MEPT is designed to compute the change in flux *shape*, not flux magnitude.

Although Touran and Lee chose to compute errors between the reference and MEPT using $\phi' = \phi + d\phi$, we choose to compute errors in $d\phi$ alone. The addition of the unperturbed flux ϕ, which we know precisely, only artificially deflates the error magnitude. Just as we computed errors in reactivity coefficients (not reactivity magnitudes), we now compute errors in flux shift coefficients (not flux magnitudes).

Throughout this section, all reference case swelling magnitudes are 1%. We use all 2500 computed modes in all cases.

10.7.1 Uniform Radial Swelling

First, consider uniform radial core swelling. Constructing a reference case in MaPS is simple; we increase the x and y dimensions of each mesh cell by f and reduce all material densities by f^2.

We can use Equation 10.47 to easily derive a VirDenT-MEPT expression for this scenario. Simply replacing $L_{n,r}$ with $L_{n,x} + L_{n,y}$ and $\Gamma_{n,r}$ with $\Gamma_{n,x} + \Gamma_{n,y}$ yields

$$\frac{d\phi}{df} = \sum_{n=1}^{\infty} \left(-4L_{n,z} + 2\Gamma_{n,z} - 2L_{n,x} - 2L_{n,y} + \Gamma_{n,x} + \Gamma_{n,y}\right) \phi_n \tag{10.44}$$

Of course, since this is a *uniform* case, the Γ terms are zero. So we are left with

$$\frac{d\phi}{df} = \sum_{n=1}^{\infty} \left(-4L_{n,z} - 2L_{n,x} - 2L_{n,y}\right) \phi_n \tag{10.45}$$

Figure 10.10 shows the results. The $d\phi/df$ distributions are nearly indistinguishable to the eye. Errors are typically below 3% but peak sharply in mesh cells where the flux shift crosses through zero. When trying to converge near-zero values with a modal sum, fractional error are expected to be quite large. Touran and Lee did not observe these zero-shift error peaks, because they computed errors in the perturbed flux ϕ' instead of in the flux shift $d\phi$ [39].

In Figures 3.7 and 3.8 of Touran's thesis, he shows that the deviation between ϕ and ϕ' (computed via a reference calculation) lies in the range 10-12% while the deviation between

ϕ and $\phi + d\phi$ (computed via MEPT) lies in the range 0.45-0.65%. This implies a $d\phi$ error in the neighborhood of 5%. The error magnitudes we see in Figure 10.10 are within this range (barring the zero-shift peaks). Of course, his and our perturbations are completely different, but at least we can rest assured that we achieve the same "ballpark" error range.

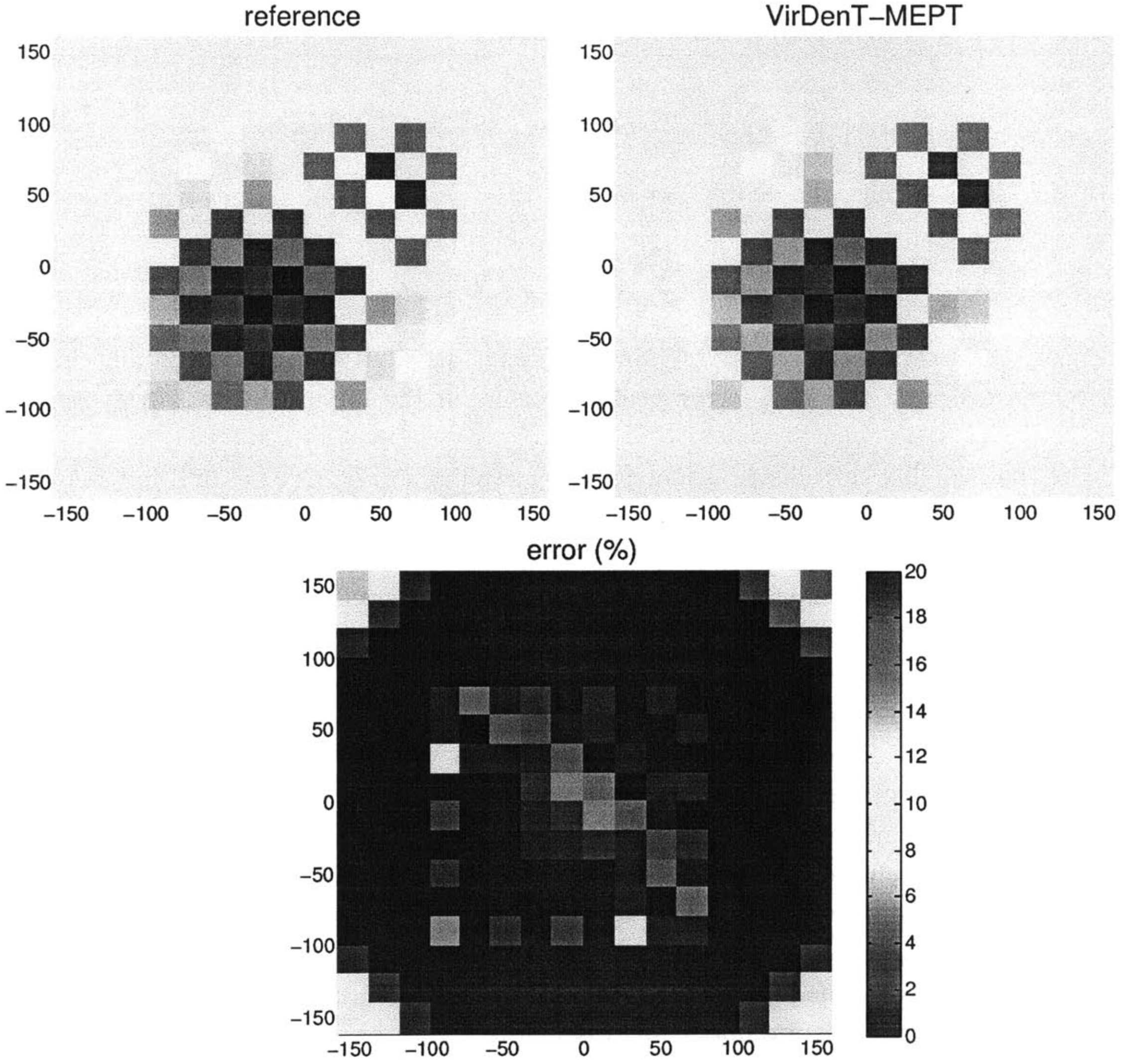

Figure 10.10: The flux shift coefficient due to uniform radial swelling, displayed in axial zone 5 at approximately 1.7 MeV. The reference calculation is compared with VirDenT-MEPT.

10.7.2 Uniform Axial Swelling

Now consider uniform axial swelling. We construct the reference case by increasing the z dimension of each mesh cell by f and decreasing all material densities by f.

The VirDenT-MEPT expression for non-uniform axial swelling is

$$\frac{d\phi}{df} = \sum_{n=1}^{\infty} \left(-2L_{n,x} - 2L_{n,y} + \Gamma_{n,x} + \Gamma_{n,y}\right) \phi_n \tag{10.46}$$

In this uniform case, the expression reduces to

$$\frac{d\phi}{df} = \sum_{n=1}^{\infty} \left(-2L_{n,x} - 2L_{n,y}\right) \phi_n \tag{10.47}$$

Figure 10.11 shows the results. Again, the $d\phi/df$ distributions are nearly indistinguishable to the eye, and the same error peaking occurs in the mesh cells with near-zero flux shift.

Note that the flux shift distributions for axial swelling in Figure 10.11 and radial swelling in Figure 10.10 are very similar. They are certainly not *exactly* the same, however. Notice that the zero flux shift in the reflector is for axial swelling than for radial swelling. Considering the color scaling, this means that for axial swelling, the negative flux shift (blue) is larger relative to the positive flux shift (red). Conversely, the negative flux shift in radial swelling is slightly smaller relative to the positive flux shift. So although the localized "“checkerboard” patterns appear nearly identical, the global shapes are noticeably different.

We can see this difference clearly by plotting the VirDenT-MEPT da_n/df coefficients in Figure 10.12. These coefficients have significant non-zero values for n approaching 1300! They fall into two main clusters. The first cluster, in the range $n < 200$, defines the approximate global shape of the flux shift. The second cluster, in the range $600 < n < 1300$ defines the more localized flux shift induced by the core heterogeneity. While the second cluster of da_n/df values is nearly identical for radial and axial swelling, the first cluster is not. This explains why the localized flux shifts appear so similar in Figures 10.11 and 10.10 while the global flux shifts are noticeably distinct.

Touran and Lee made plots similar to Figure 10.12, and they observed only the first cluster of da_n/df coefficients[39]. This is because their primary test core is nearly homogeneous, and so the whole flux shift is a global shape - there *are* no localized patterns due to heterogeneity. Computing only a few hundred harmonics can work for a nearly homogeneous core, but any realistic heterogeneous core requires computing at least as many harmonics as their are *fuel* mesh cells. One can probably ignore the “reflector modes”, but no “fuel modes” can be neglected in a heterogeneous core. Contrary to popular belief, excitation of higher modes does *not* only arise from localized perturbations - it can also arise from uniform global perturbations in a heterogeneous core.

Figure 10.13 shows the harmonics corresponding to the 6 largest magnitudes of da_n/df for uniform axial swelling. Comparing this with Figure 10.11 reveals that harmonic $n = 2$ captures the most basic global shape, while much higher modes (such as $n = 820$ and 722) capture the localized effects.

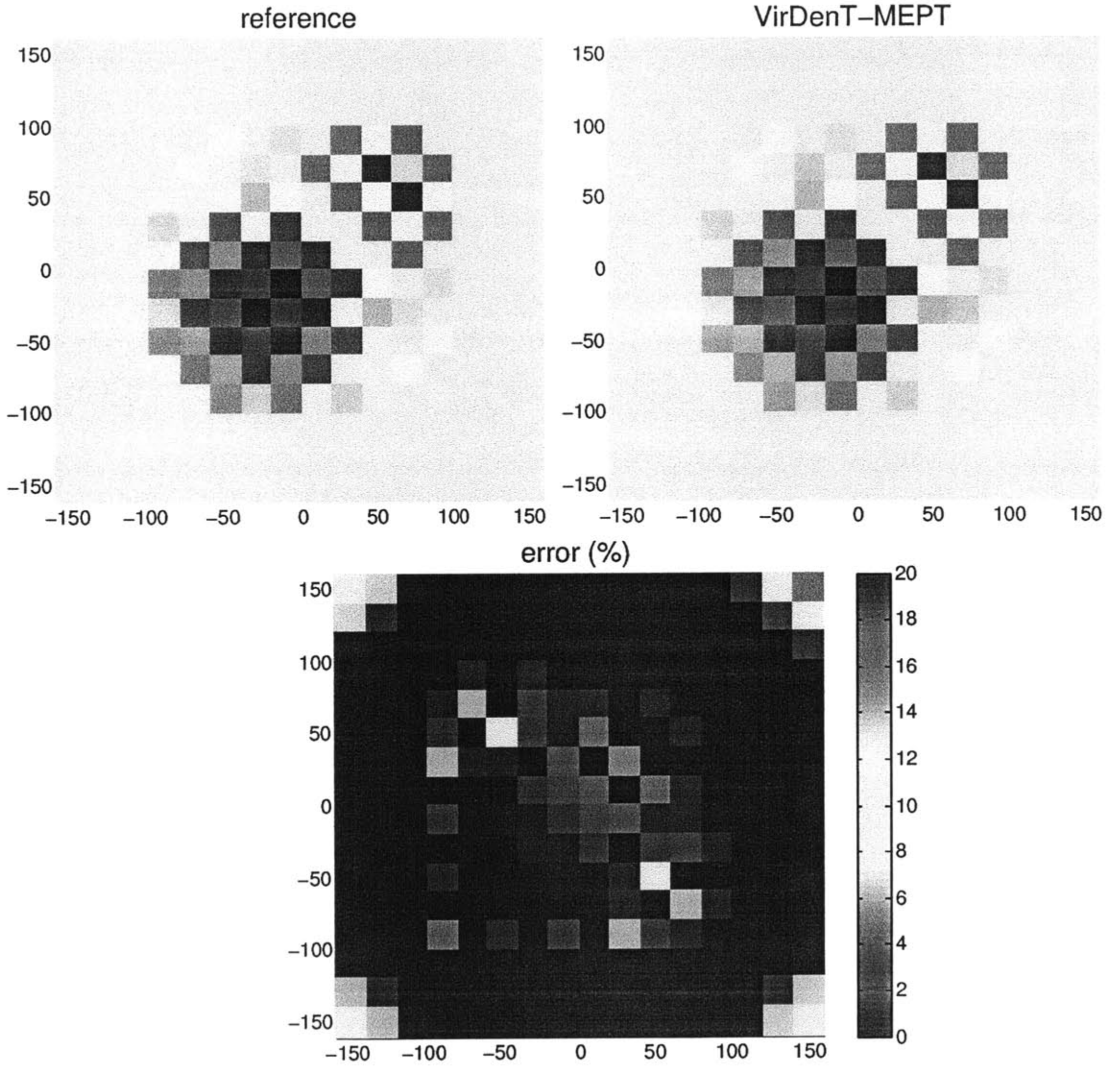

Figure 10.11: The flux shift coefficient due to uniform axial swelling, displayed in axial zone 5 at approximately 1.7 MeV. The reference calculation is compared with VirDenT-MEPT.

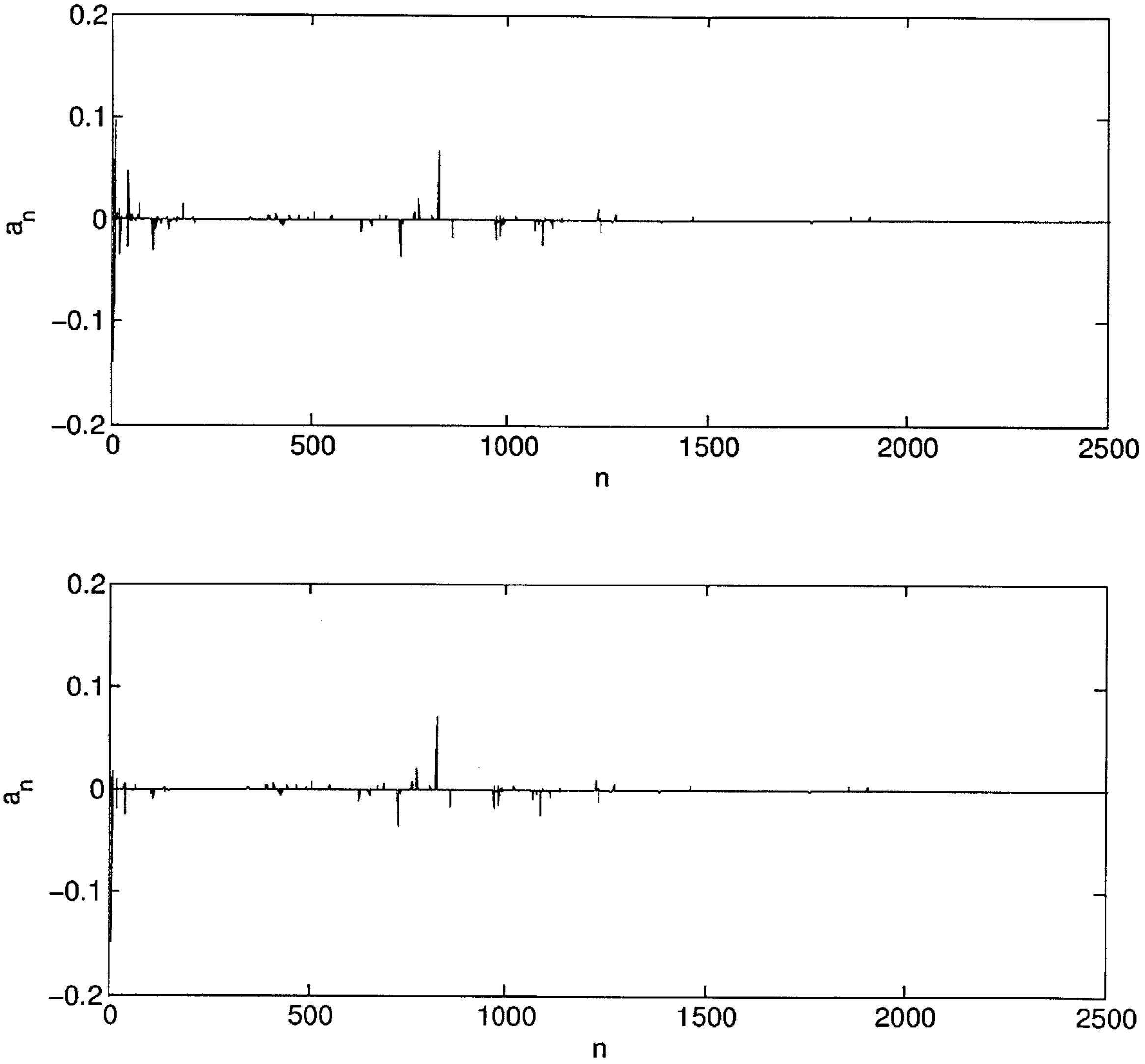

Figure 10.12: The VirDenT-MEPT da_n/df coefficients for uniform radial (top) and axial (bottom) swelling.

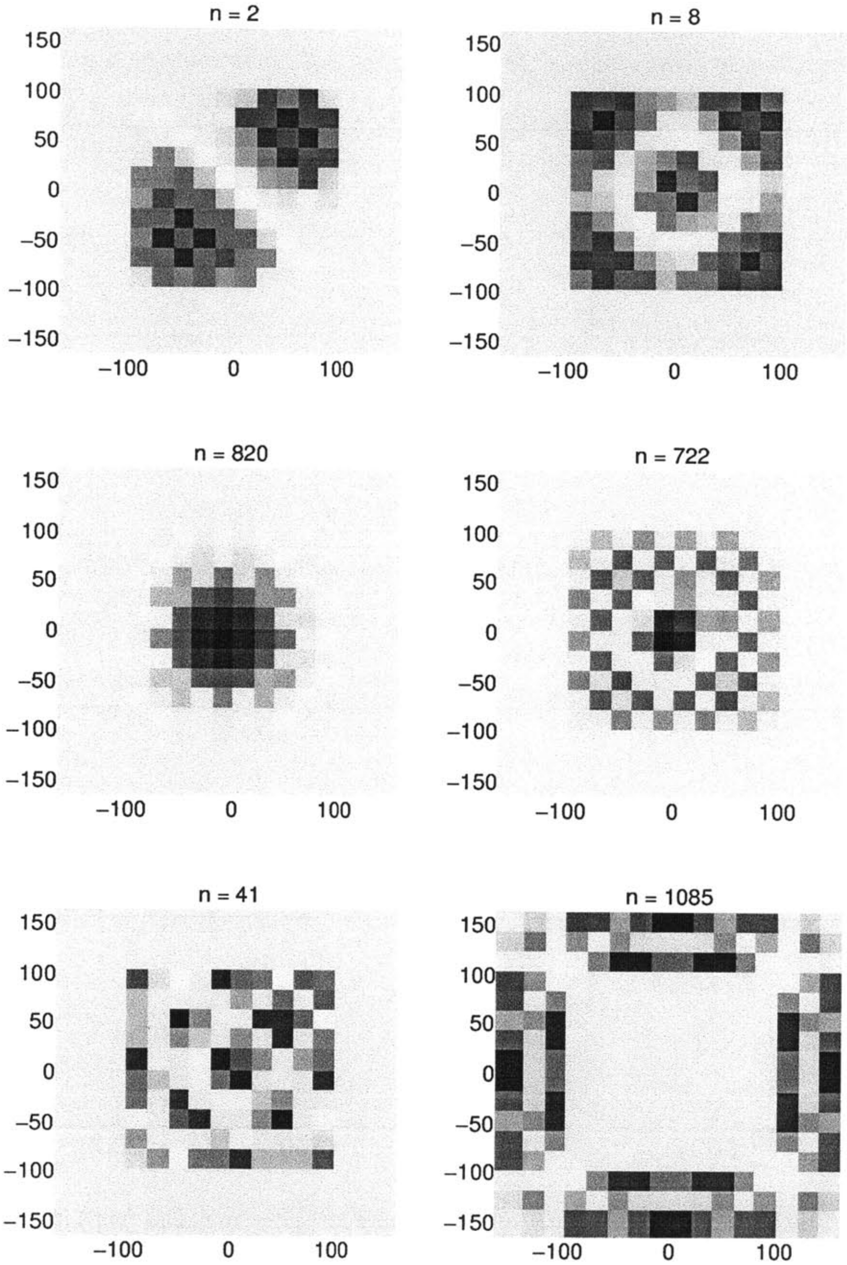

Figure 10.13: The modes corresponding to the 6 largest VirDenT-MEPT da_n/df coefficients for uniform axial swelling.

10.7.3 Axially Non-Uniform Axial Swelling

Now we turn to non-uniform swelling cases. First consider axial swelling that is non-uniform in the axial direction. This amounts to axial swelling of through-core axial slices. This test core model has 10 such axial slices (axial mesh cells), each with the same material composition except that the fuel enrichment increases linearly throughout core height such that it is 10% higher at the top.

Consider axial zones 6, 7, and 8 swelling axially while all other zones remain unchanged. Figures 10.14, 10.15, and 10.16 show the $d\phi$ results in zones 7, 9, and 3, respectively. In zone 7, which swells, the flux shift $d\phi$ is strongly heterogeneous with error magnitudes in the same range as the uniform radial and axial swelling cases. In zone 9, while lies above the swelling region, the flux shift distribution is a bit smoother, and errors are smaller. In zone 3, which lies far below the swelling region, errors are extremely small - never exceeding 0.7% and mostly well below 0.1%!

Intriguingly, it seems that VirDenT-MEPT is much more robust when evaluating perturbation flux shifts in regions of the core far removed from those perturbations. Although this may seem surprising, it is actually very intuitive. As the flux shift propagates away from its causal perturbation, it becomes smoother and less localized. Thus, the modes can more readily capture its shape.

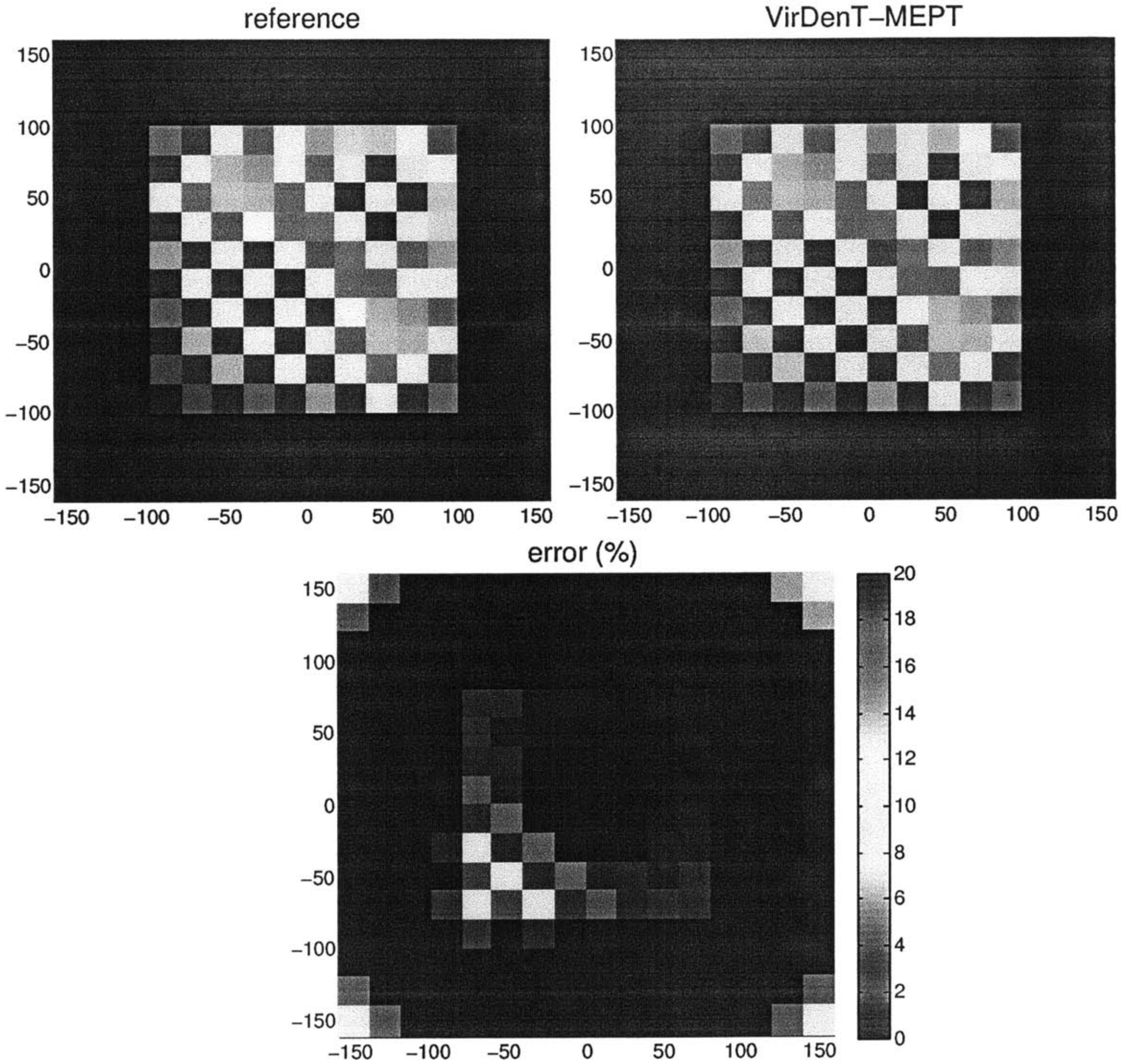

Figure 10.14: The flux shift coefficient due to axial swelling in axial zones 6-8, displayed in zone 7 at approximately 1.7 MeV. The reference calculation is compared with VirDenT-MEPT.

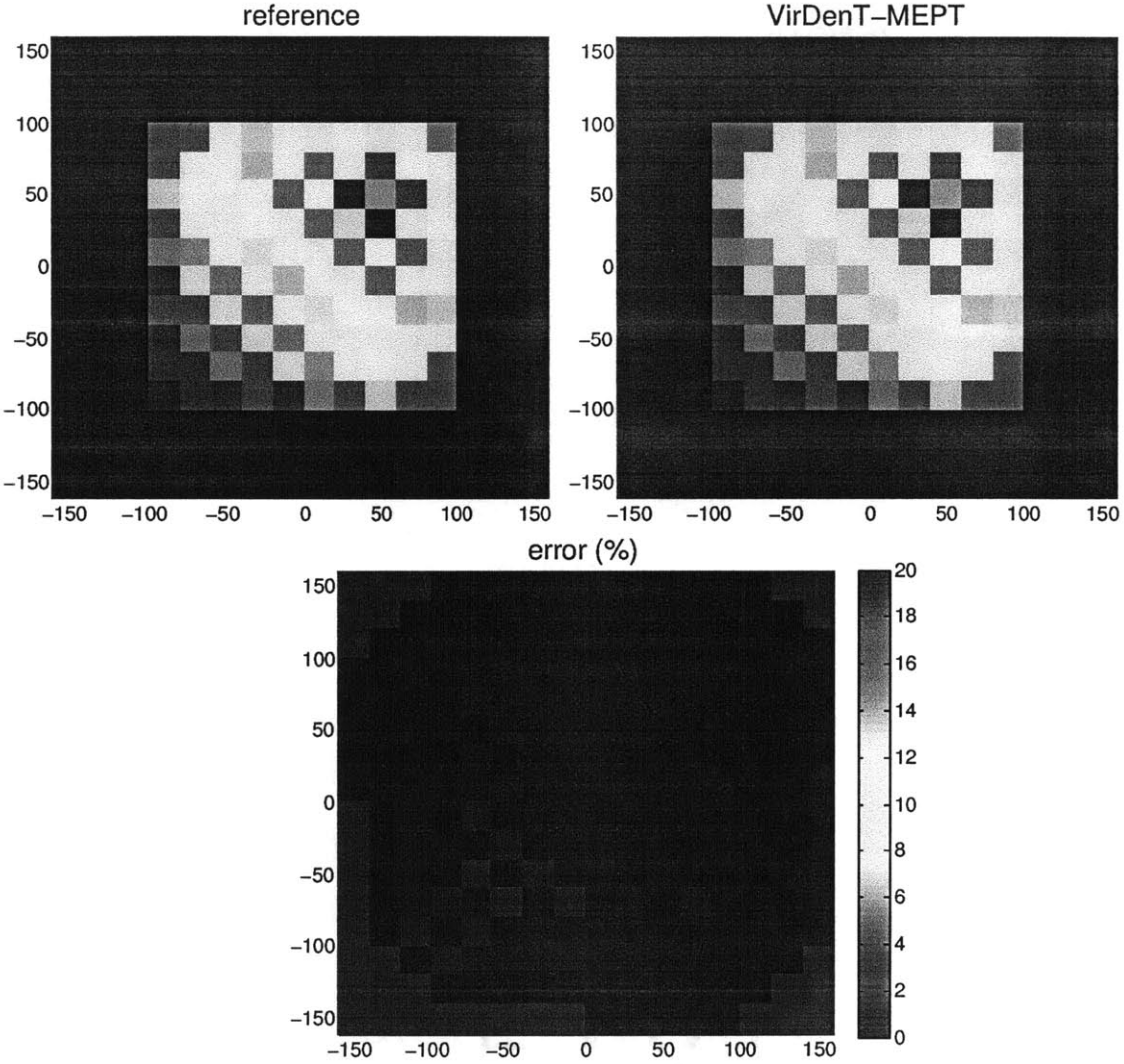

Figure 10.15: The flux shift coefficient due to axial swelling in axial zones 6-8, displayed in zone 9 at approximately 1.7 MeV. The reference calculation is compared with VirDenT-MEPT.

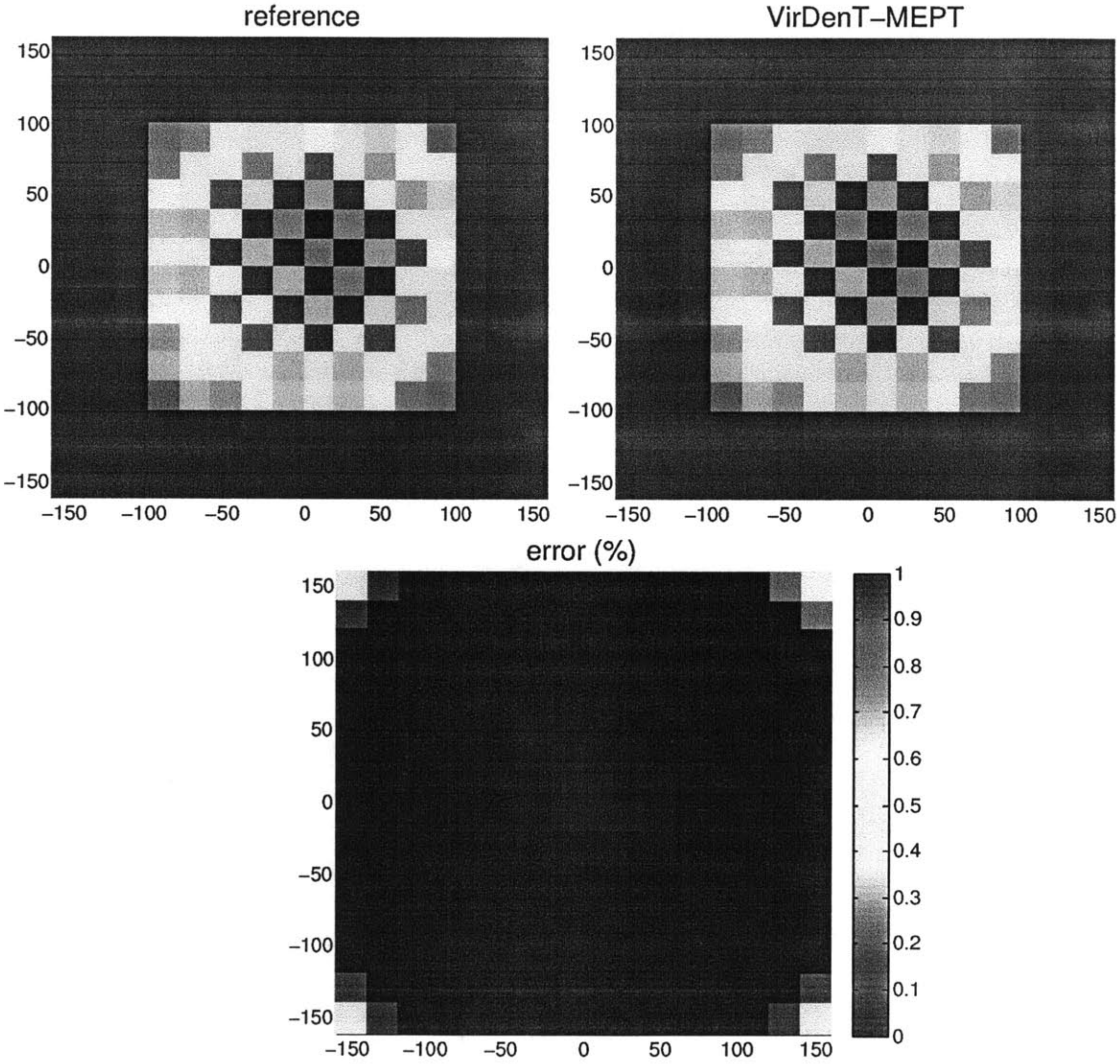

Figure 10.16: The flux shift coefficient due to axial swelling in axial zones 6-8, displayed in zone 3 at approximately 1.7 MeV. The reference calculation is compared with VirDenT-MEPT.

10.7.4 The "Virtual Mesh" Method Applied to Flux Distributions

So far, we have considered only swelling cases that can be validated against a "true" reference. In order to analyze any further swelling cases, we must rely upon the "virtual mesh" method studied in Chapter 6 and used to validate VirDenT reactivity coefficients in Chapter 7. In Chapter 6, we showed mathematically how a "virtual mesh" can accurately predict first order quantities, and we verified this fact with Monte Carlo.

Although we only verified this for first order reactivities, it is also true for first order flux distributions. The mathematical reasoning is precisely the same as we laid out in Section 6.3. We stated that it is valid for a first order change in any quantity, which includes flux distributions. Rigorously validating the "virtual mesh" method for flux shifts would require well-converged global flux tallies in Monte Carlo, but we are confident enough in the "virtual mesh" mathematical reasoning to postpone such an endeavor.

10.7.5 Radially Non-Uniform Axial Swelling

Consider axial swelling of an arbitrary set of assemblies. Here we define the set as a "ring" including all assemblies in the central 6x6 array *except* those in the central 2x2 array. So this ring is 2 assemblies thick and contains a total of 32 assemblies.

Figure 10.17 shows the results. Again, errors are quite low except for large peaks in mesh cells where the flux shift crosses zero.

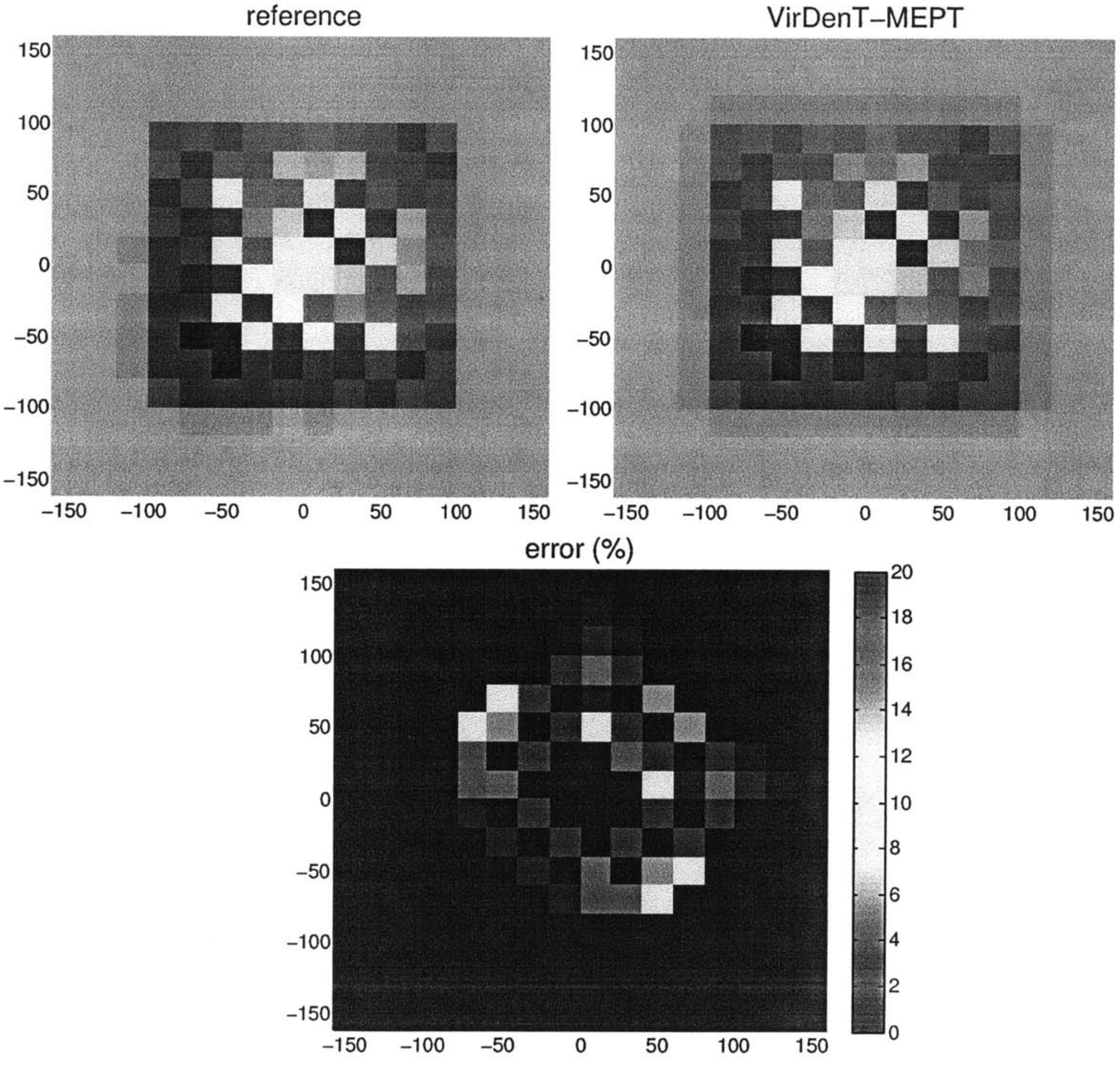

Figure 10.17: The flux shift coefficient due to axial swelling of an assembly "ring", displayed in zone 5 at approximately 1.7 MeV. The reference calculation is compared with VirDenT-MEPT.

10.7.6 Axially Non-Uniform Radial Swelling

Now consider *radial* swelling in axial zones 6-8 while all other zones remain unchanged. Figures 10.18 and 10.19 show the resulting flux shift in zones 5 and 4, respectively. In zone 5, just below the swelling region, we see errors in the fuel below 3% but large errors in the reflector. In zone 4, further removed from the swelling region, we see errors less than 1.2% throughout the core.

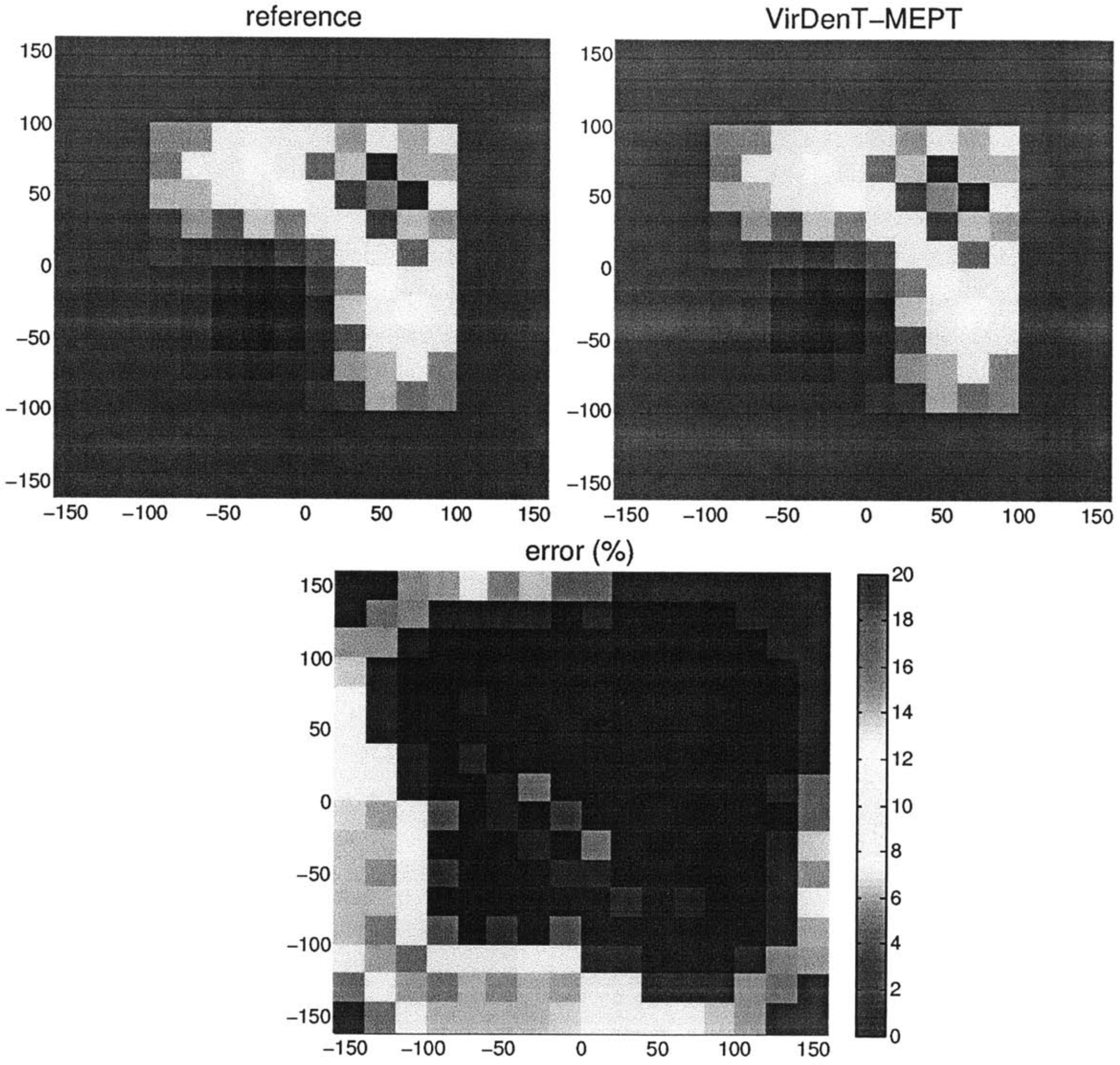

Figure 10.18: The flux shift coefficient due to radial swelling in axial zones 6-8, displayed in zone 5 at approximately 1.7 MeV. The reference calculation is compared with VirDenT-MEPT.

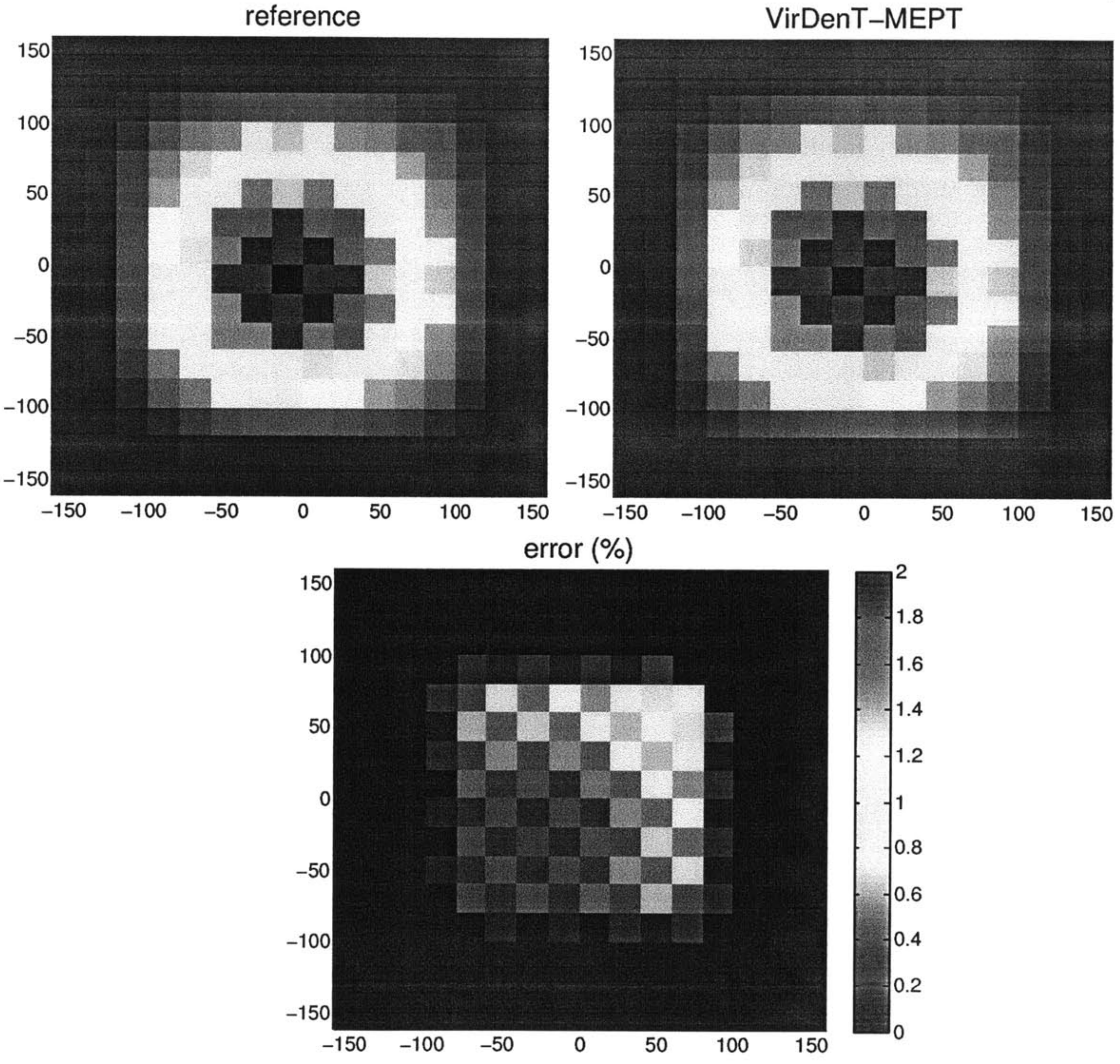

Figure 10.19: The flux shift coefficient due to radial swelling in axial zones 6-8, displayed in zone 4 at approximately 0.6 MeV. The reference calculation is compared with VirDenT-MEPT.

10.7.7 Axially Non-Uniform X-Direction Swelling

Now consider anisotropic swelling in only the x direction within axial zones 6-8. Although this scenario is not terribly realistic, it is at least instructive and a valid test for VirDenT-MEPT.

Figures 10.20 and 10.21 show the result in zones 7 and 3, respectively. The flux shift distribution peaks toward the positive x side of the core. The error in zone 7, which swells, is less than 8% throughout the fuel, while the error in zone 3, far below the swelling region, remains below 0.25% throughout the core.

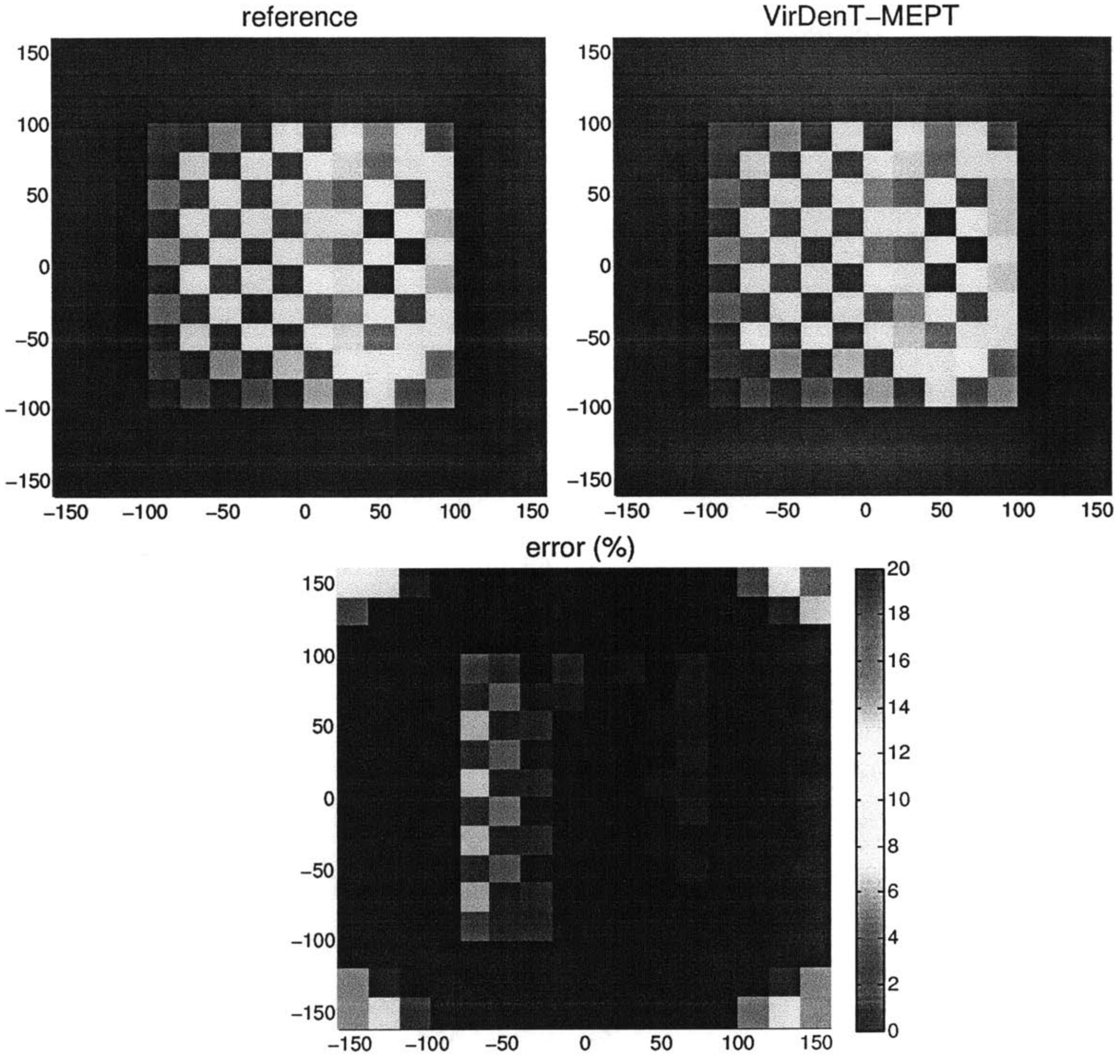

Figure 10.20: The flux shift coefficient due to anisotropic swelling in the x direction in zones 6-8, displayed in axial zone 7 at approximately 1.7 MeV. The reference calculation is compared with VirDenT-MEPT.

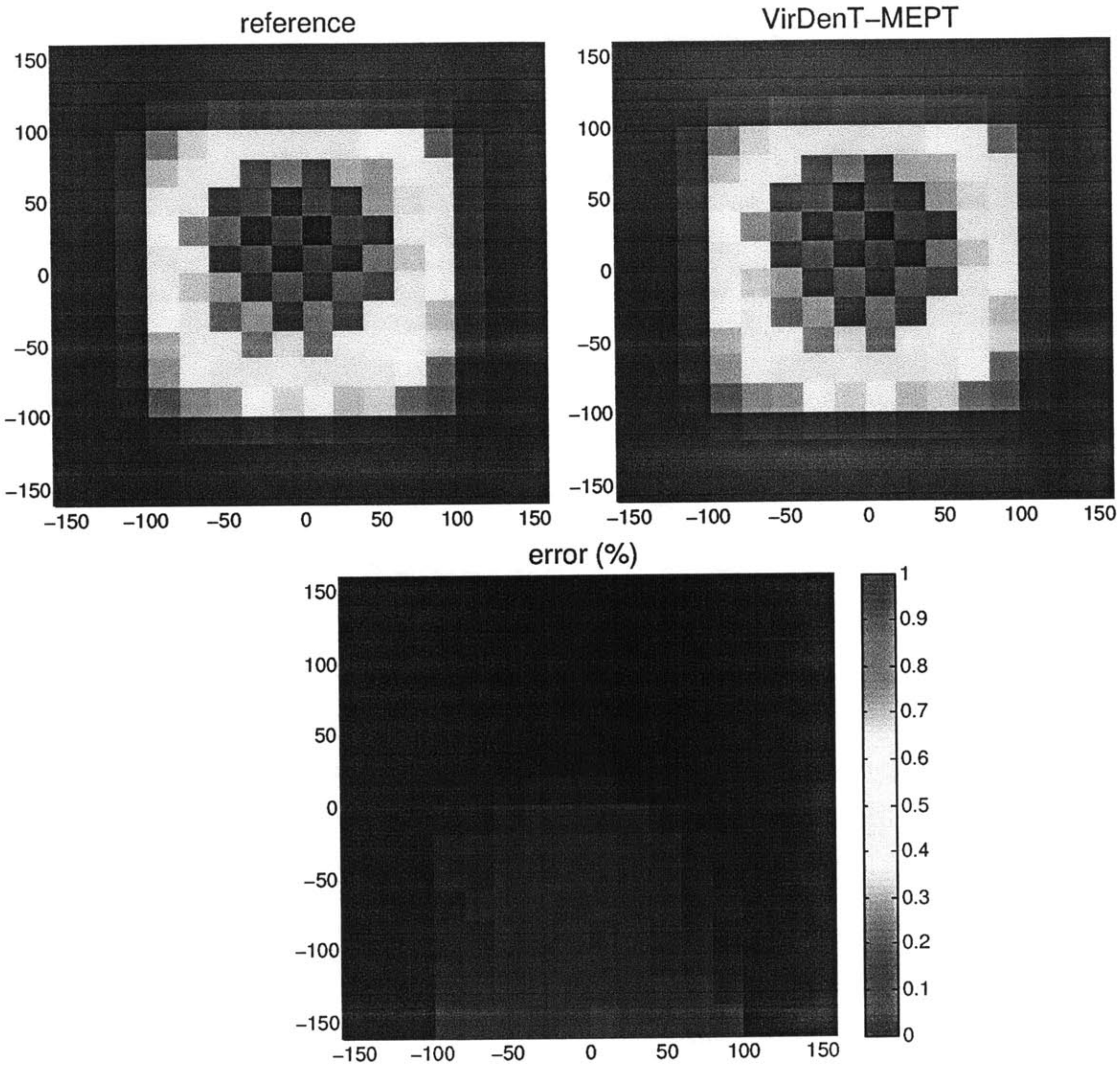

Figure 10.21: The flux shift coefficient due to anisotropic swelling in the x direction in zones 6-8, displayed in zone 3 at approximately 0.6 MeV. The reference calculation is compared with VirDenT-MEPT.

10.7.8 Radially and Axially Non-Uniform Axial Swelling

Now we test the most complex swelling scenario. Suppose that each individual mesh cell swells axially by a factor f proportional to its fission power. We scale down f throughout the core so that its maximum value (in the cell with maximum power) is 1%.

Figure 10.22 shows the result in zone 5. Flux shift errors are always under 6% except in the zero-shift peaks and some regions of the reflector. These error magnitudes are roughly consistent with what we found for the uniform radial and axial swelling cases.

So far, we have only shown *flux* shifts in one energy group at a time. Typically, MEPT is more robust in the high-energy groups. This is because the low-energy groups have shorter mean free paths and more spatial variation, hampering the harmonic convergence (or at least requiring *more* harmonics). In order to allay any concerns regarding this effect, we will show the *power* shift distribution for this most complex swelling scenario. We compute the power shift coefficient using Equation 10.27.

Figure 10.23 shows the power shift in axial zone 5 for a very large 10% axial swelling magnitude. We show the power shift coefficient error in terms of both a percentage and a magnitude. Note that we cannot solve the diffusion eigenvalue problem directly for this non-uniform axial swelling scenario, and so we simply perform a "virtual mesh" eigenvalue calculation for a very small swelling magnitude and extrapolate the solution out to 10%. Thus, these errors do not include effects beyond first order. As Figure 10.23 shows, the power shift error *percentage* distribution exhibits the same zero-shift peaks evident in the flux shift error distributions. The power shift error *magnitude* distribution, however, does *not* exhibit these peaks.

Note that errors in Figure 10.23 are no larger than errors in the high-energy flux shown in Figure 10.22. Although the flux may not be well-converged at very low energies, it also barely exists at very low energies. Thus, the low-energy contribution to power is quite small. As shown in Figure 1.1, about 95% of power in a typical sodium fast reactor is produced by neutrons above 1 keV.

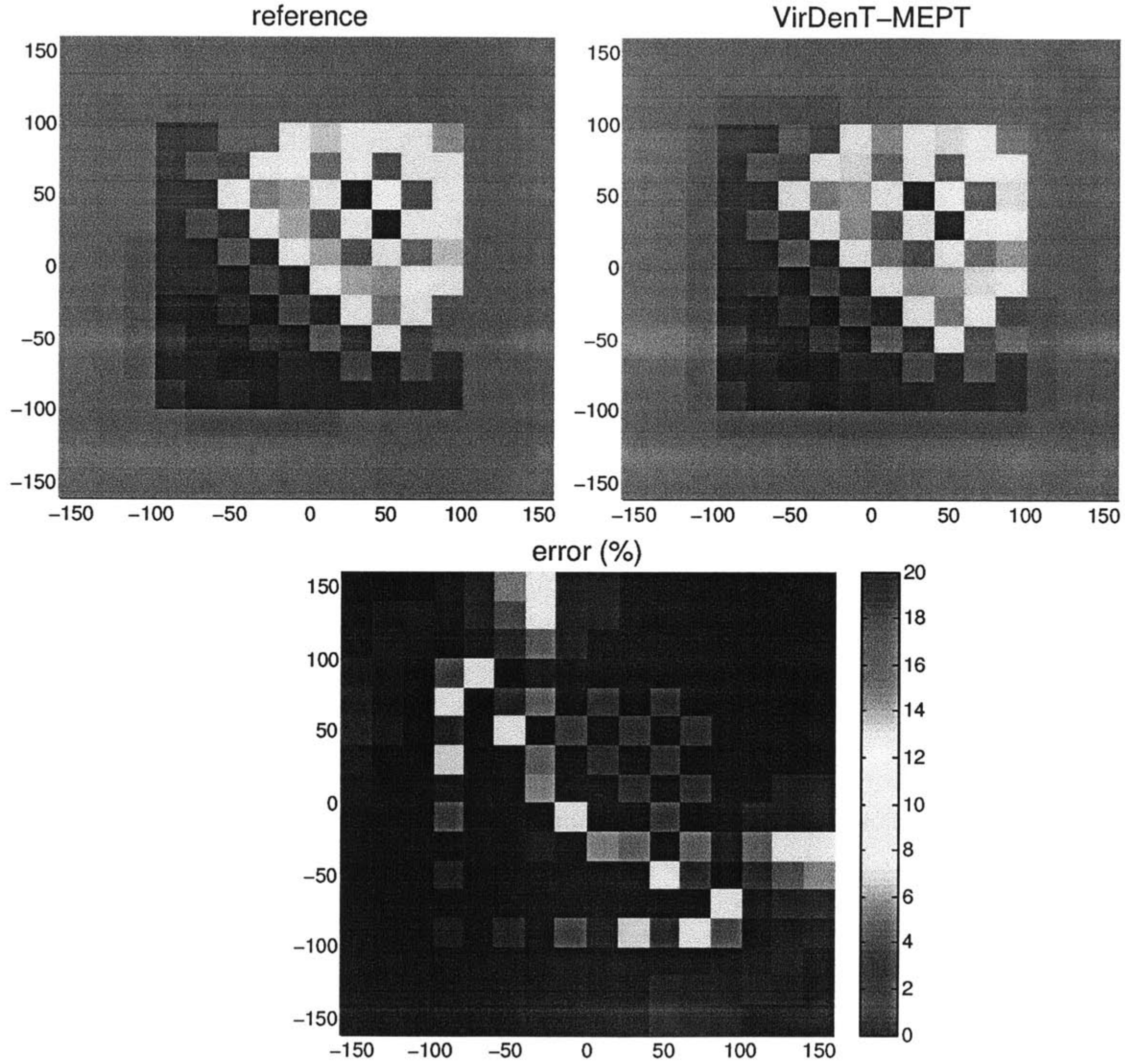

Figure 10.22: The flux shift coefficient due to axial swelling of each mesh cell in proportion to its fission power, displayed in zone 5 at approximately 1.7 MeV. The reference calculation is compared with VirDenT-MEPT.

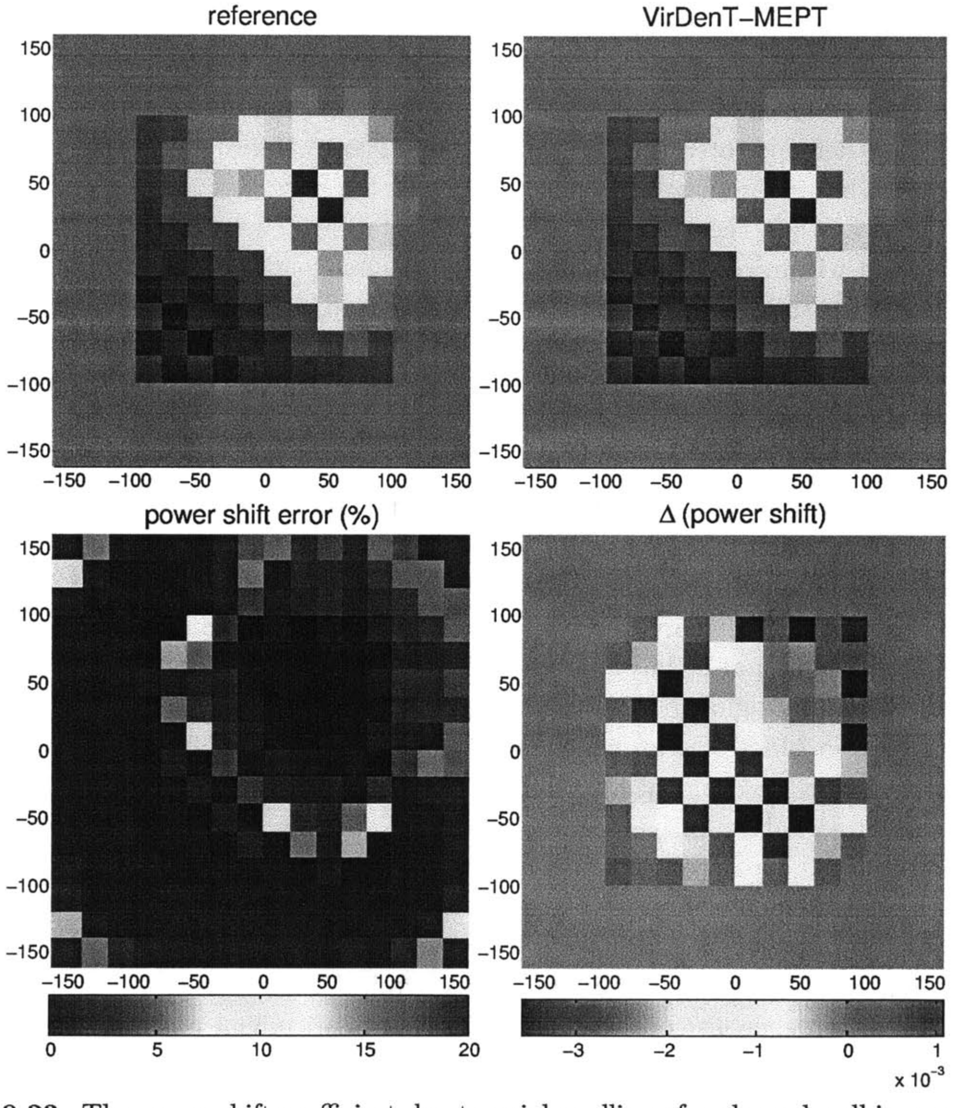

Figure 10.23: The power shift coefficient due to axial swelling of each mesh cell in proportion to its fission power, displayed in axial zone 5. The reference calculation is compared with VirDenT-MEPT. We display the error as both a percentage and a magnitude (given that the maximum flux shift is normalized to 1)

10.8 Summary

Traditionally, modal expansion has been used exclusively for material density changes and has met only middling success for large, heterogeneous cores. The main limitation was that solving hundreds (or thousands) of harmonics was not practical with computers that existed in the "harmonic heyday" of the 1970s. Very recently, Touran and Lee revisited the topic and demonstrated that modern Arnoldi/Krylov techniques can easily compute thousands of modes, and they demonstrated the accuracy of MEPT for material density perturbations in small cores with limited heterogeneity[39,40].

Unfortunately, while the vast increase of computation power since the 1970s has put modal expansion within reach, it has also, paradoxically, "outmoded" modal expansion. If nodal methods can solve a full core diffusion problem in seconds, why would one pursue modal expansion? It still certainly is more computationally efficient when evaluating very large numbers of perturbations, but if a full core can be solved directly in seconds, why should one spend manhours on modal analysis?

The answer is geometry perturbations. One can always rerun an eigenvalue calculation for any material density change in seconds - just change the input file and press "go" again. However, geometry perturbations are not so simple. One cannot simply rerun the code, because doing so would require building an entirely new type of mesh that few (if any) existing diffusion codes have. Futhermore, the surface leakage terms in non-uniform anisotropic "virtual density" theory do not meld well with direct diffusion calculations. Currently, the only reliable way to compute global flux distributions due to geometry distortions is via multigroup flux tallies in Monte Carlo, which is extremely computationally expensive. Tallying just several well-converged global flux shifts in Monte Carlo would cost CPU decades! So although computational power has rendered modal expansions obsolete for material density perturbations, modal expansions are still quite expeditious for geometry perturbations.

In this chapter, we demonstrated this for several non-uniform anisotropic swelling scenarios by combining our VirDenT formalism with the MEPT formalism developed by Touran and Lee[39,40]. In order to test the limits of this hybrid VirDenT-MEPT method, we chose a large, highly-heterogeneous core with a checkerboard enrichment pattern. We found that if one computes at least as many harmonics as fuel mesh cells, VirDenT-MEPT performs about as well as pure MEPT. Heterogeneity does not stymie MEPT; it only requires more harmonics. Even in our highly-heterogeneous test core, VirDenT-MEPT flux shifts generally agree with reference cases to within a few percent.

11 The "Virtual Density" Theory (VirDenT) Code: Distortion Reactivity for Fast Reactor Design

11.1 Abstract

We introduce the VirDenT code, which computes reactivities for arbitrary distortion scenarios but also contains modules for classic reactivity coefficients, flux reconstruction, and pin-detail visualization. We extend conventional fast reactor flux reconstruction techniques in hexagonal-z geometry to include vector fields and adjoint-weighted quantities. In particular, we reconstruct intra-assembly analytic representations for the real and adjoint neutron net currents, the fission power gradient, and the reactivity worth of an arbitrary nuclide. In order to visualize core-wide, pin-level distributions of these reconstructed (scalar and vector) quantities, we develop a suite of visualization tools dubbed PyPinPlot. Furthermore, we unearth an intriguing physics scenario in which *adding* material actually *increases* leakage. We demonstrate how to estimate sodium void reactivity coefficients on the subchannel level. We validate classic reactivity coefficients for fuel, structure, and coolant density as well as uniform axial and radial swelling. All coefficients agree with DIF3D eigenvalue references to within 1%. In order to validate distortion reactivities, we compare VirDenT (which operates on a homogenized assembly DIF3D finite difference solution) to heterogeneous pin-detail Monte Carlo.

11.2 Introduction

The purpose of neutronic methods is to design and build reactors. Thus, any neutronic method is pointless unless it can ultimately contribute to the analysis and design of pragmatic cores. If we cannot apply "virtual density" theory to full-core 3-D heterogeneous distortion scenarios, all this work is ultimately in vain.

In Chapter 5, we successfully applied "virtual density" theory to homogenized assemblies in the FFTF and Jōyō benchmarks. However, even these scenarios were not entirely realistic, because we assumed that *all* homogenized material densities changed by the same factor. In reality, fuel, structure, and coolant materials swell by different factors, and any useful design work must account for this. Furthermore, we did not cover the most arbitrary case of radially non-uniform radial swelling, because such cases are not possible to validate via homogenized assembly diffusion (even with "virtual mesh" techniques).

This chapter describes the VirDenT code, which models arbitrary assembly distortions with axial dependence - axially and radially non-uniform axial and radial swelling. We validate VirDenT reactivities via continuous energy heterogeneous Monte Carlo, the highest-fidelity reference we can obtain. First, however, we will describe the auxiliary features of

VirDenT, which include pin-level visualization tools, flux and power reconstruction, and classic reactivity coefficient calculation.

11.3 VirDenT within ARMI

We implement VirDenT within the Advanced Reactor Modeling Interface (ARMI), a Python-based modeling framework that loosely couples nuclear reactor simulations to provide high-fidelity systems analysis [142,211]. It wraps and couples numerous stand-alone codes but also contains a number of modules written entirely in Python, such as depletion and subchannel analysis.

We compose VirDenT as a Python module within ARMI, relying heavily upon the `NumPy` scientific computing package and the `MPI4Py` parallelization package. Figure 11.1 shows the interaction between VirDenT and other modules or wrapped codes within ARMI. VirDenT takes neutronic input from DIF3D [193] and REBUS [189] and computes pin powers via flux reconstruction. These pin powers are essential for subchannel analysis (COBRA [191]) and computational fluid dynamics (CFD) in order to perform pin-level and subchannel-level thermal hydraulic analysis. Pin powers are also necessary to determine peak displacements per atom (DPA) for materials analysis.

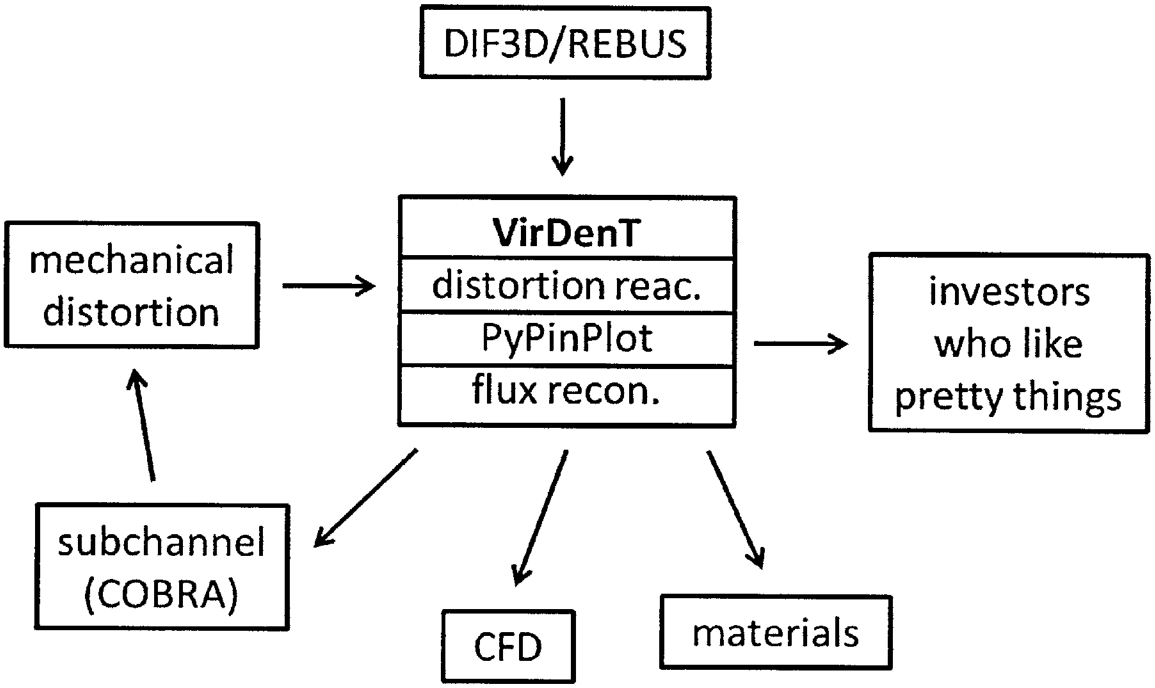

Figure 11.1: VirDenT within ARMI. VirDenT takes input from DIF3D and REBUS, and its flux reconstruction module produces power distributions for subchannel analysis, computational fluid dynamics (CFD), and materials analysis. Subchannel analysis provides temperature data for mechanical analysis, which in turn feeds assembly distortions back into VirDenT.

11.4 PyPinPlot: A Pin-Level Visualization Tool for Fast Reactors in Hex-Z Geometry

In order to visualize various reconstructed pin-level quantities in VirDenT, we develop a compact visualization tool utilizing the Python packages `matplotlib` and `mpatches` [204]. We dub this PyPinPlot. This tool can produce core-wide, pin-level, and high-resolution maps of reconstructed scalar or vector quantities. PyPinPlot displays a specified quantity on each pin throughout a complete core slice in the (x, y) plane at one fixed axial position, and it can also generate 1-D plots of scalar quantities as a function of axial position in one assembly.

Examples of scalar quantities include multigroup flux, multigroup adjoint flux, power, fission source, burnup, and DPA. Examples of vector quantities include current, adjoint current, and the power gradient.

PyPinPlot can handle any arbitrary pin configuration, and it will automatically generate a pin configuration given a number of full pin rings. Pin diameters reflect the actual outer cladding diameters, and hexagonal shell thicknesses reflect actual assembly duct thicknesses. Furthermore, PyPinPlot can illustrate some mechanical distortions in the form of (x,y) assembly translations.

See Figure 11.2 for an example of scalar and vector plots. Via `matplotlib`, PyPinPlot can generate plots as Portable Document Format (PDF), Portable Network Graphics (PNG), or Joint Photographic Experts Group (JPEG) files. The PDF files can be enlarged up to 640x with zero reduction of image quality, as shown in Figure 11.2.

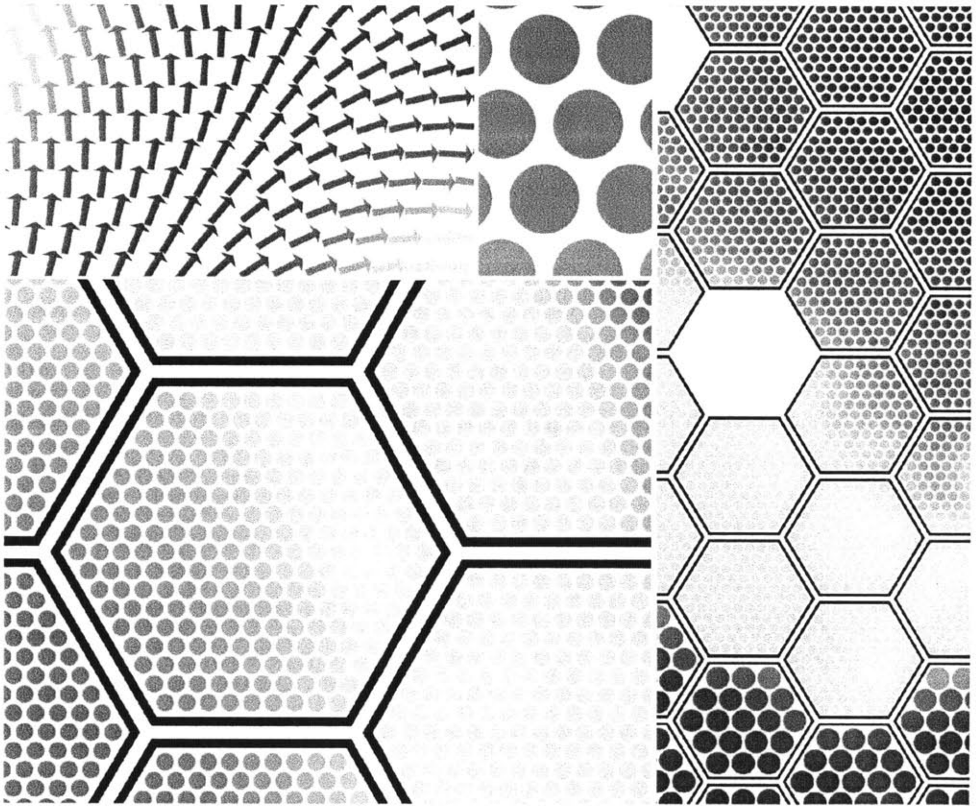

Figure 11.2: Examples of the high resolution of PyPinPlot scalar and vector plots.

11.5 Pin-Level Flux and Power Reconstruction

Fast neutrons have long mean free paths. Thus, reactor cores with fast neutron spectra can be modeled on coarse meshes, often with only one mesh cell per hexagonal assembly block. Such coarse meshes yield fairly accurate k_{eff} eigenvalues and core-wide flux distributions. Unfortunately, they are not well suited for detailed intra-assembly neutronic or thermal hydraulic analysis. While coarse assembly-level meshes are adequate for scoping analyses, build-ready design and regulatory standards require knowledge of neutronic quantities on the pin level. One could take the "brute force" route by solving the neutron diffusion (or transport) equation on a hyperfine pin-level mesh, or one could reconstruct the intra-assembly multigroup flux distributions from coarse mesh quantities.

Techniques for multigroup flux, power, and burnup reconstruction in fast reactors have been well studied and applied in hexagonal geometry. However, we extend these techniques to reconstruct other neutronic properties, including current vector fields and adjoint-weighted quantities.

11.5.1 Theory

A coarse assembly-level nodal diffusion solution produces cell-averaged fluxes and surface-averaged partial currents. We follow the procedure outlined by Yang in hexagonal-z geometry [113,114]. Within each hexagonal prism block, the multigroup real flux ϕ is assumed to be separable in the axial (z) and radial (x,y) directions. In the radial direction, the reconstructed flux in each energy group preserves six surface-averaged net currents, six surface-averaged fluxes, six corner-point fluxes, and one volume-averaged flux. In the axial direction, the flux reconstruction in each energy group preserves two surface-averaged net currents, two surface-averaged fluxes, and one volume-averaged flux. Thus, there are a total of five axial constraints and nineteen radial constraints [113]. Equation 11.1 shows the complete group flux expression; the flux reconstruction process requires computing the $C_{i,g}$ and $C_{j,g}$ coefficients.

$$\phi_g(x,y,z) = \left[\sum_{i=1}^{19} C_{i,g} x^{m_i} y^{n_i}\right] \left[\sum_{j=1}^{5} C_{j,g} z^{j-1}\right] \tag{11.1}$$

We can easily represent the power density P as a sum of the same monomials with different weights:

$$P(x,y,z) = \left[\sum_{i=1}^{19} C_i x^{m_i} y^{n_i}\right] \left[\sum_{j=1}^{5} C_j z^{j-1}\right] \tag{11.2}$$

$$C_i = \gamma \sum_{g=1}^{G} \Sigma_{f,g} C_{i,g} \tag{11.3}$$

Here γ represents the energy release per fission event. The C_j values are normalized such that the average of the separable z polynomial is 1.

We have benchmarked this reconstruction approach against a hyperfine mesh finite difference diffusion solution for the Fast Flux Test Facility (FFTF) [146] in MaPS. In this hyperfine mesh, we subdivide each hexagonal assembly into either 150 or 294 equilateral triangles (the number of triangles per hexagon is always six times a perfect square). We solve the eigenvalue problem on this mesh to obtain the reference flux and power distributions. Then we perform volume and surface integrals on these hyperfine reference flux distributions to obtain volume-averaged fluxes, surface-averaged fluxes, and surface-averaged currents for each assembly block. We apply the flux reconstruction techniques to these integrated quantities to obtain the $C_{i,g}$ and $C_{j,g}$ coefficients in Equation 11.1 for each assembly block. Now we evaluate Equations 11.1 and 11.2 for (x, y) values corresponding to a fictitious set of pins located at the centers of the hyperfine triangles. This allows for direct comparison to the reference flux and power distributions. Core-averaged reconstructed pin power error magnitudes are less than 0.4% for the FFTF benchmark and less than 0.05% for a homogenous core with FFTF fuel.

See Appendix B for a more thorough discussion of the flux reconstruction theory and its validation process.

11.5.2 Implementation

The ARMI neutronics module includes the diffusion solver DIF3D [193] and its embedded transport option called VARIational Anisotropic Nodal Transport (VARIANT) [199]. We use 33-group cross-section sets generated by MC**2 [187].

We calculate the surface-averaged real and adjoint partial currents via the DIF3D binary output files NHFLUX and NAFLUX, respectively [193]. However, the DIF3D nodal diffusion adjoint solver is unmaintained, so we must employ the VARIANT v10.0 nodal transport adjoint solver to obtain a usable NAFLUX file. The use of VARIANT v10.0 is crucial, as v8.0 will produce an NHFLUX/NAFLUX file format that is exceptionally cumbrous to read.

We develop an ARMI Python module to read surface-averaged partial currents from NHFLUX and/or NAFLUX, generate the 19 radial and 5 axial constraints, and solve matrix equations to produce the $C_{i,g}$ and $C_{j,g}$ values shown in Equation 11.1. This module provides pin power input to thermal hydraulic, materials, and mechanical analysis. Furthermore, we couple it to a transmutation module to perform pin-level depletion and explore the effects of assembly rotation during shuffling.

Figures 11.3 and 11.4 show reconstructed pin powers for FFTF and Jōyō on their axial midplanes. These high-resolution images are produced via PyPinPlot. Only assemblies containing at least some fissionable material are reconstructed - control and test assemblies are left "blank". Figure 11.5 shows reconstructed pin powers for the Generic Metal-Fueled (GMF) SFR test core, described in Appendix I. Figure 11.6 shows pin powers in a small homogeneous core with FFTF mixed oxide (MOX) fuel. In this case, the flux reconstruction produces a smooth, cosine-like radial power profile.

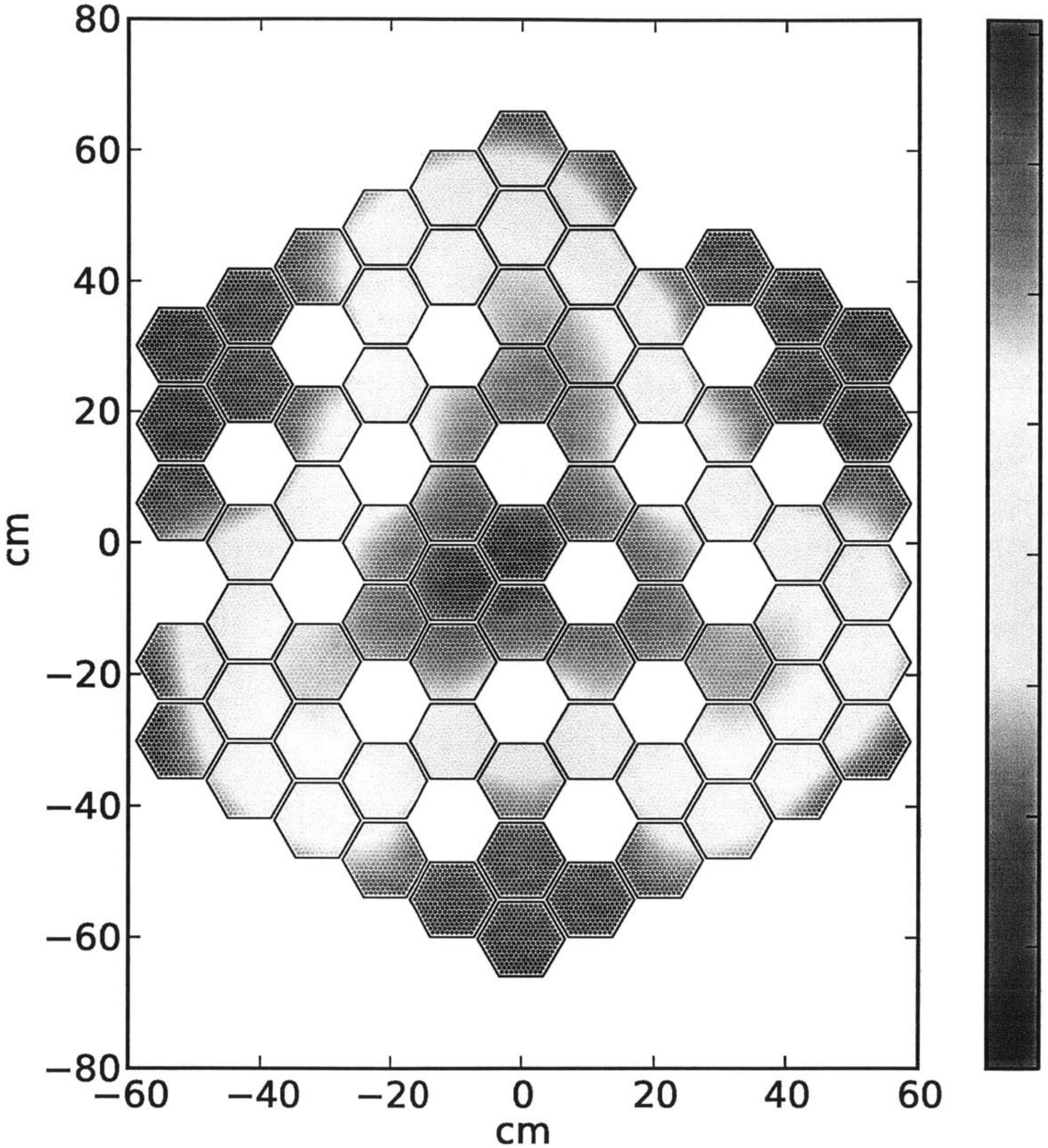

Figure 11.3: Pin power reconstruction in the FFTF sodium fast reactor benchmark displayed via PyPinPlot [146]. The core consists of two enrichment zones. The blank hex locations represent withdrawn control assemblies and test assemblies interspersed throughout the core.

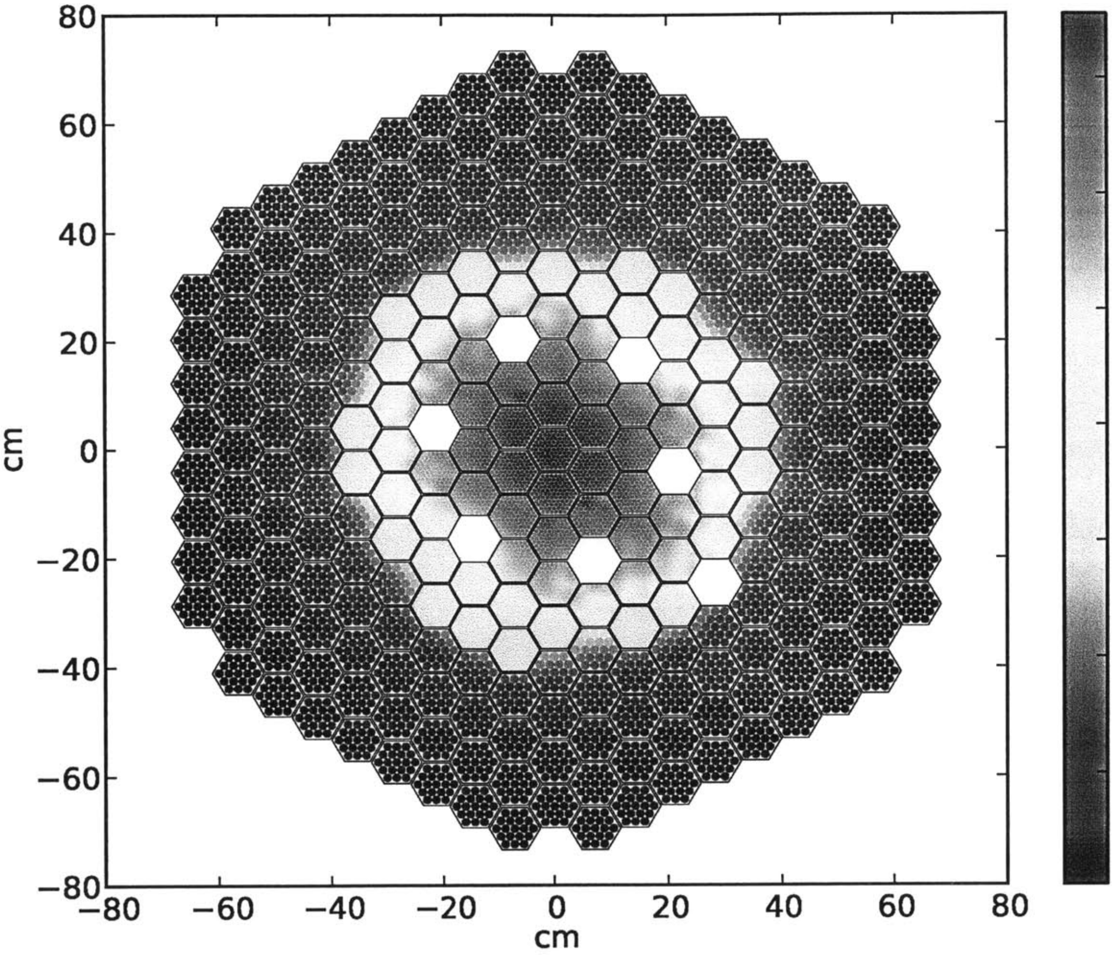

Figure 11.4: Pin power reconstruction in the Jōyō sodium fast reactor benchmark displayed via PyPinPlot [147]. An inner core of 91-pin fuel assemblies is surrounded by a thick blanket of 19-pin breeder assemblies. The blank hex locations represent withdrawn control assemblies.

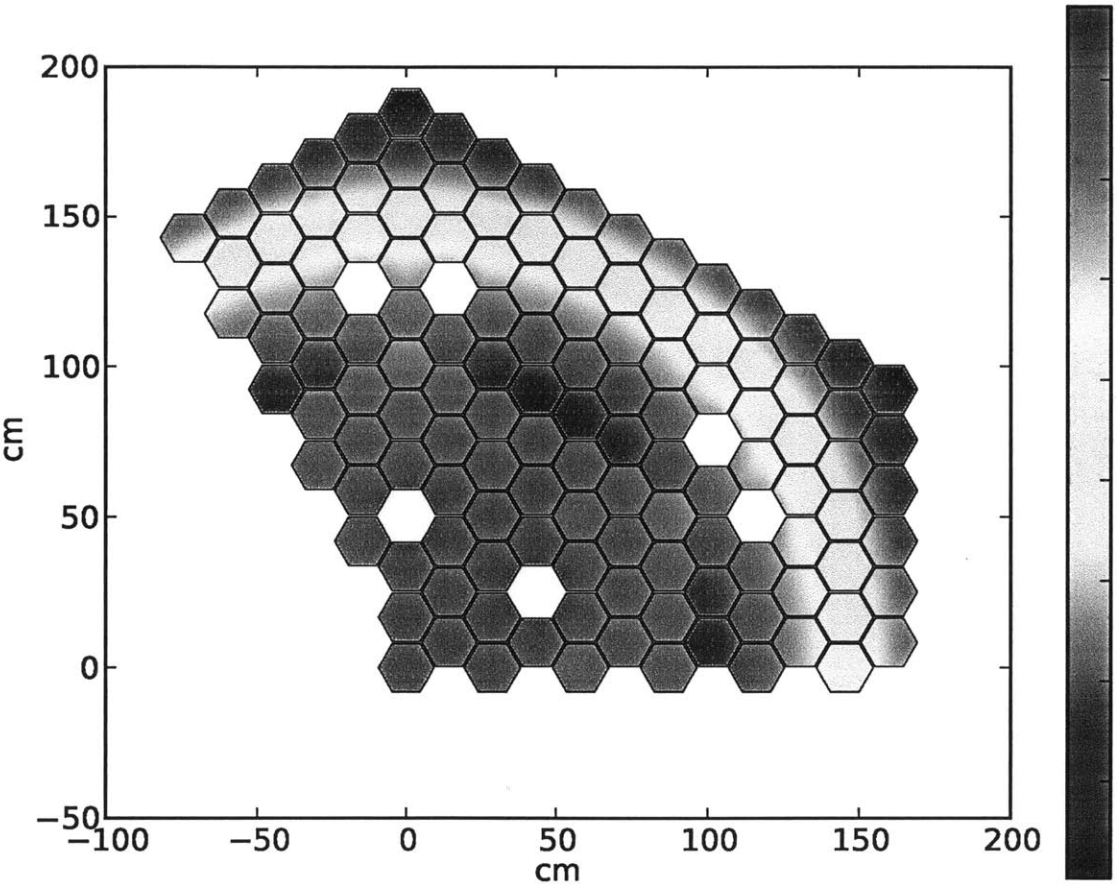

Figure 11.5: Pin power reconstruction in the GMF SFR benchmark displayed via PyPinPlot [147]. The blank hex locations represent withdrawn control assemblies.

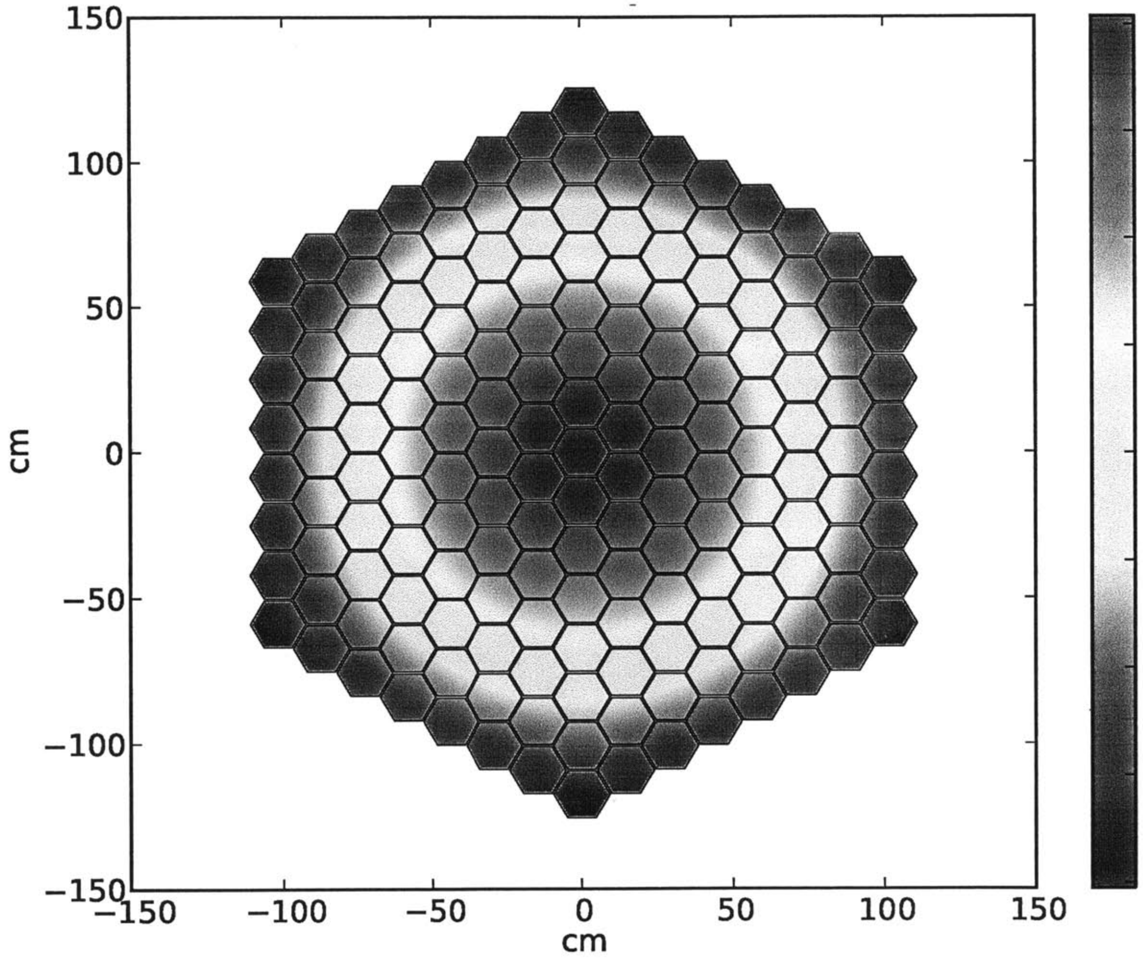

Figure 11.6: Pin power reconstruction of a hypothetical homogenous core (fully loaded with FFTF fuel) displayed via PyPinPlot.

11.5.3 Pin-Level Adjoint Flux Reconstruction

Consider a fission event. Mathematically, neutrons from a certain group flux ϕ_g are transferred to another group flux $\phi_{g'}$. The neutron source S in group g' is simple to express:

$$S_{g'} = \chi_{g'} \sum_{g=1}^{G} \nu_g \Sigma_{f,g} \phi_g \tag{11.4}$$

Here ν_g is the average number of neutrons born from a fission event induced by a neutron in group g, while $\chi_{g'}$ is the fraction of those neutrons born into group g'. Σ_f is the macroscopic fission cross-section. See Sections 2.3.2 and 2.3.3 for an introduction to the multigroup diffusion formalism.

Now consider a fission event in the context of adjoint flux. Mathematically, neutrons from a certain adjoint group flux $\phi^\dagger_{g'}$ are transferred to another adjoint group flux $\phi^\dagger_g$. This is why the adjoint flux is often referred to as the "backward" flux. The adjoint neutron source in group g is thus

$$S^\dagger_g = \nu_g \Sigma_{f,g} \sum_{g'=1}^{G} \chi_{g'} \phi^\dagger_{g'} \tag{11.5}$$

Now let us consider how to define the total fission neutron source. Remember that the sum of χ_g over all g is 1.

$$S = \sum_{g'=1}^{G} S_{g'} = \sum_{g=1}^{G} \nu_g \Sigma_{f,g} \phi_g \tag{11.6}$$

Now let us similarly define the total fission adjoint neutron source.

$$S^\dagger = \sum_{g=1}^{G} S^\dagger_g \tag{11.7}$$

While the real source represents neutrons *produced by* fission events that were *caused by* the real flux, the adjoint source represents neutrons *that caused* fission events *that produced* the adjoint flux. Computing the real source requires tracing the real flux *forward* to determine which fission events it causes. In contrast, computing the adjoint source requires tracing the adjoint flux *backward* to determine which fission events produced it.

Previously, we discussed how to reconstruct the adjoint multigroup flux from a coarse mesh DIF3D-VARIANT solution. Once the real and adjoint multigroup fluxes are known, computing the real and adjoint sources is trivial.

Figure 11.7 compares the real and adjoint pin sources for a subregion of a small sodium fast reactor test core. Blank hexagon cells represent withdrawn control assemblies. The real and adjoint sources have similar distributions *except* in the immediate vicinities of these withdrawn control assemblies, which are flooded with sodium. The real source peaks near these withdrawn control assemblies, because sodium softens the spectrum. In contrast, the adjoint source depresses near these withdrawn control assemblies, because neutrons that enter these fuel-less assemblies are less likely to induce fission and thus altogether less “important”.

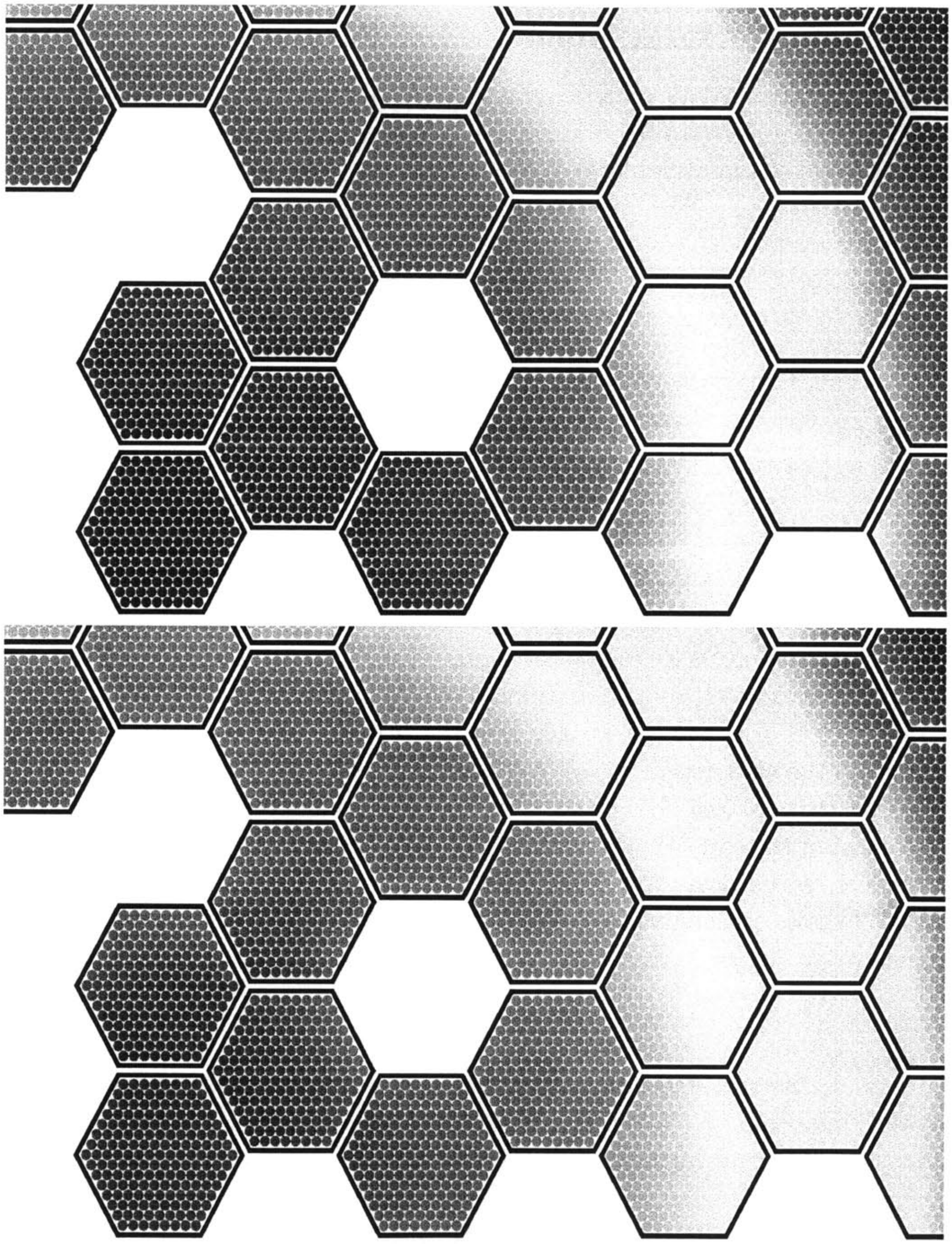

Figure 11.7: Pin scalar quantities in a small 1/3 test core with reflective boundary conditions displayed via PyPinPlot. The upper image shows the real source, while the lower image shows the adjoint source. Note that the real fluxes tend to peak near the withdrawn control assemblies (due to a softened spectrum), while the adjoint fluxes tend to depress in the same areas (reflecting lower neutron importance). In these unique areas, the real and adjoint currents are roughly parallel, while they are roughly antiparallel throughout most of the core.

11.5.4 Vector Field Reconstruction: Neutron Currents and Power Gradients

Since we have reconstructed an analytic representation for the multigroup flux in each hexagonal block, it is straightforward to obtain an analytic representation for the (real and adjoint) net currents ($\vec{J}$ and $\vec{J}^{\dagger}$) in each block:

$$\vec{J}_g = -D_g \nabla \phi_g \tag{11.8}$$

$$\vec{J}^{\dagger}_g = D_g \nabla \phi^{\dagger}_g \tag{11.9}$$

Here D is the diffusion coefficient. For example, the x coordinate of the group flux gradient is

$$[\nabla \phi_g]_x = \left[\sum_{i=1}^{19} C_{i,g} m_i x^{m_i - 1} y^{n_i} \delta_{m_i > 0} \right] \left[\sum_{j=1}^{5} C_{j,g} z^{j-1} \right] \tag{11.10}$$

where δ is the Dirac delta function. We can plot these currents as vector fields via PyPinPlot. Figure 11.8 compares the real and adjoint net neutron currents (for one energy group) in a small sodium fast reactor test core. Note that the neutron current field lines bend toward the withdrawn control assemblies, which are weak neutron sinks.

Referring back to Figure 11.7, we saw that the real and adjoint source distributions are similar except in the vicinities of withdrawn control assemblies. Figure 11.8 further confirms this. Note that the real and adjoint currents flow in opposite directions (countercurrently) throughout most of the core. However, they flow in the same direction (concurrently) on the "leeward" side of withdrawn control assemblies.

Figure 11.9 shows the angle θ between the real and adjoint vector fields throughout the same core shown in Figure 11.8.

$$\vec{J}^{\dagger} \cdot \vec{J} = |\vec{J}^{\dagger}||\vec{J}| \cos\theta \tag{11.11}$$

When $\theta > \pi$, the real and adjoint currents flow countercurrently. Real neutrons flow toward lower "importance". When $\theta < \pi$, the real and adjoint currents flow concurrently. Real neutrons flow toward higher "importance".

$$\vec{J}^{\dagger}_g \cdot \vec{J}_g > 0 \quad \text{neutrons diffusing toward higher importance} \tag{11.12}$$

$$\vec{J}^{\dagger}_g \cdot \vec{J}_g < 0 \quad \text{neutrons diffusing toward lower importance} \tag{11.13}$$

We can also construct an analytic representation for the power density gradient, assuming that the fission power density (a scalar function) is continuously distributed within each block (and not discretized into pin locations). For example, the y-component of the power density gradient is

$$[\nabla P]_y = \left[\sum_{i=1}^{19} C_i n_i x^{m_i} y^{n_i - 1} \delta_{n_i > 0}\right] \left[\sum_{j=1}^{5} C_{j,g} z^{j-1}\right] \tag{11.14}$$

Figure 11.10 exemplifies this for the same test core shown in Figures 11.8 and 11.9. The highest power gradients tend to occur on the "leeward" sides of withdrawn control assemblies. Core-wide pin-level power gradient maps can be useful for mechanical and structural analysis.

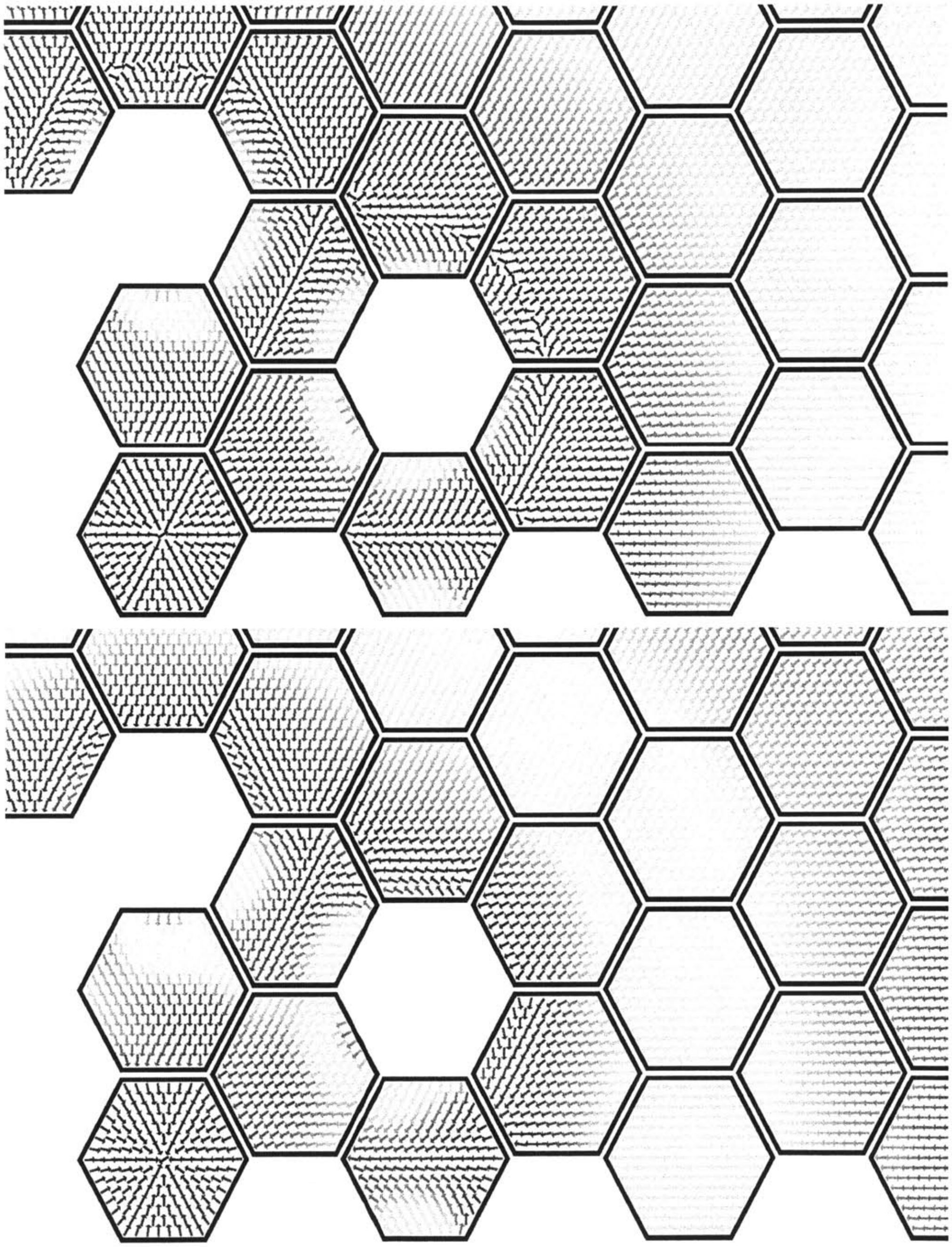

Figure 11.8: The real (top) and adjoint (bottom) net neutron currents in group 5 (1.35 - 2.23 MeV) displayed via PyPinPlot for a small 1/3 test core model. Colors represent relative magnitude, while arrows show direction. The real and adjoint currents are nearly always antiparallel, except on the "leeward" side of withdrawn control assemblies. In these unique regions, the real and adjoint currents are roughly parallel, indicating that neutrons flow toward higher importance. Thus, removing material will decrease leakage.

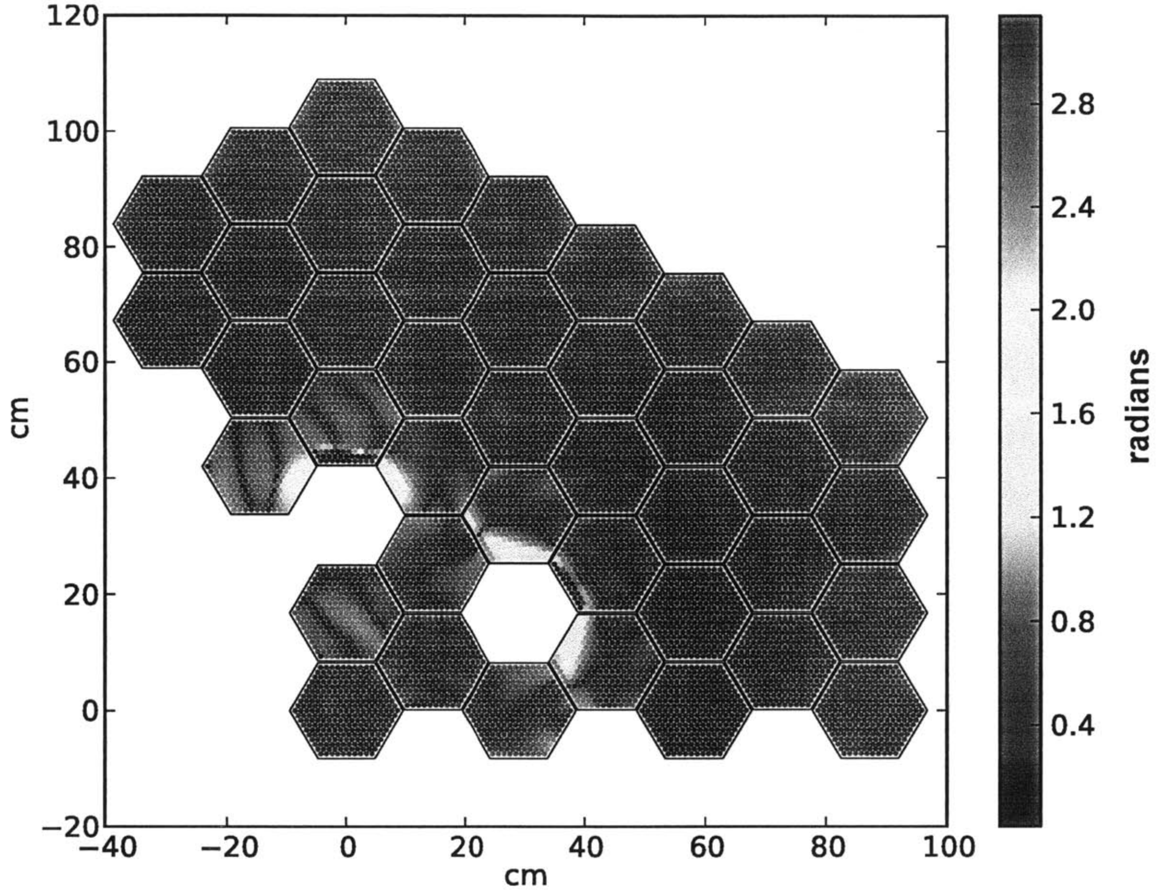

Figure 11.9: The angle (radians) between the total real net current and the total adjoint net current. Note that Figure 11.8 shows the real and adjoint net currents for this same core in energy group 5. Throughout most of the core, the real and adjoint currents flow in opposite directions (countercurrent). However, on the "leeward" sides of withdrawn control assemblies, the real and adjoint currents flow in the same direction (concurrent). This implies that, in these unique regions, neutrons flow toward higher importance.

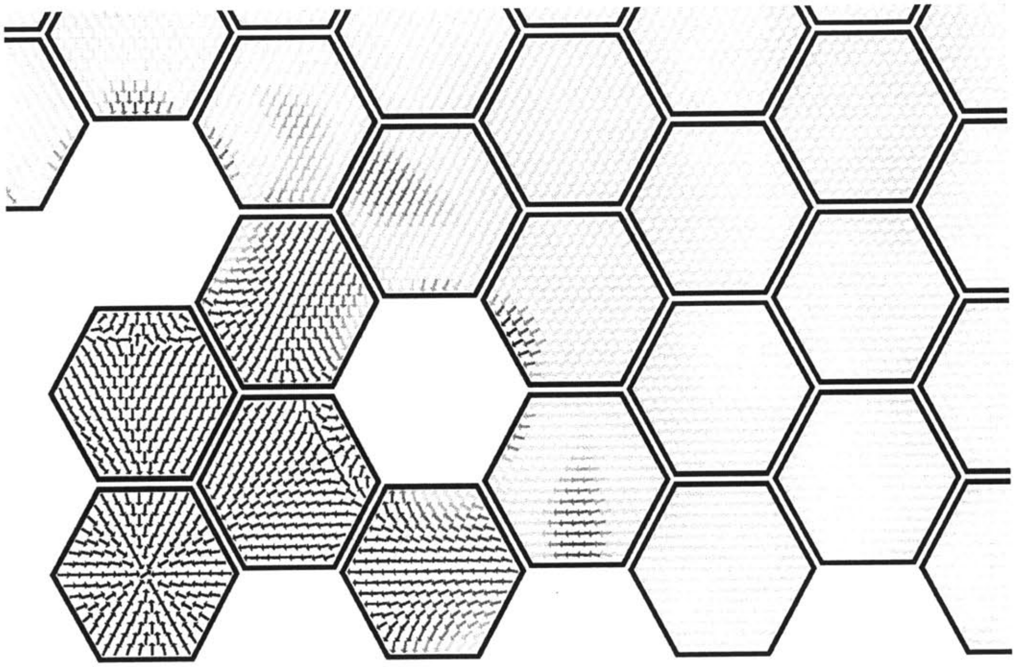

Figure 11.10: The fission power gradient displayed via PyPinPlot for a small 1/3 test core model. Colors represent relative magnitude, while arrows show direction. The highest power gradients occur on the "leeward" side of withdrawn control assemblies. Such core-wide power gradient maps are elucidating for thermal hydraulic and mechanical analysis, as they often reflect temperature gradients.

11.5.5 Adjoint-Weighted Quantity Reconstruction: When Adding Material Increases Leakage

Since we have reconstructed analytic representations for the real and adjoint fluxes as well as for the real and adjoint flux gradients, it is straightforward (albeit computationally cumbersome) to obtain an analytic representation for the reactivity worths of various materials. The first order reactivity $\Delta\rho$ due to small perturbations in the standard multigroup diffusion operators dF and dM is

$$\Delta\rho = \frac{\left\langle \phi^\dagger \middle| \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi \right\rangle}{\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle} \tag{11.15}$$

If we wish to compute a value that represents the "reactivity worth" of voiding sodium, then we can simply evaluate $\phi^\dagger(dM\phi)$. If we already have analytic representations of the flux, the adjoint flux, and their gradients in each block, then an analytic function proportional to the sodium "reactivity worth" per unit mass is

$$\phi^\dagger(-dM\phi) = -\sum_{g=1}^{G}\left[\phi_g^\dagger d\Sigma_{a,g}\phi_g - \left(\nabla\phi_g^\dagger \cdot \nabla\phi_g\right)\frac{d\Sigma_{tr,g}}{3\Sigma_{tr,g}^2} + \sum_{g'=1}^{G}\left(\phi_g^\dagger - \phi_{g'}^\dagger\right) d\Sigma_{s,g\to g'}\phi_g\right] \tag{11.16}$$

Here Σ_a, Σ_{tr}, and Σ_s are the absorption, transport, and scattering macroscopic cross-sections, respectively. We can also write the leakage term in terms of a dot product of the real and adjoint neutron currents by using Equations 10.7 and 10.8.

$$\langle\phi^\dagger| -dL\phi\rangle = -\int dV\left[\sum_{g=1}^{G}\left(\vec{J}_g^\dagger \cdot \vec{J}_g\right)3d\Sigma_{tr,g}\right] \tag{11.17}$$

Note that Equation 11.17 is always positive when the real and adjoint currents are antiparallel and always negative when the real and adjoint currents are parallel. Thus, adding material will always increase the leakage component of reactivity (by decreasing leakage) *except* in the special case when the real and adjoint currents are parallel. In that special case, adding material will actually decrease the leakage component of reactivity (by increasing leakage).

This result seems counter-intuitive, but considering the "importance" interpretation of the adjoint flux can be elucidating. When the real and adjoint currents are parallel, real neutrons flow toward higher "importance". Thus, adding material obstructs the flow of neutrons toward higher "importance", which effectively increases leakage. Thus, the leakage component of reactivity decreases. It follows that removing material allows neutrons to more readily flow toward higher "importance", which decreases leakage and increases reactivity.

Figure 11.11 compares the parallel and antiparallel scenarios conceptually. This intriguing phenomenon occurs only in small core subregions as illustrated in Figures 11.8 and 11.9 - it is a heterogeneous effect. Nevertheless, designing cores with larger such regions could be favorable for neutron economy as well as create some intriguing physics scenarios.

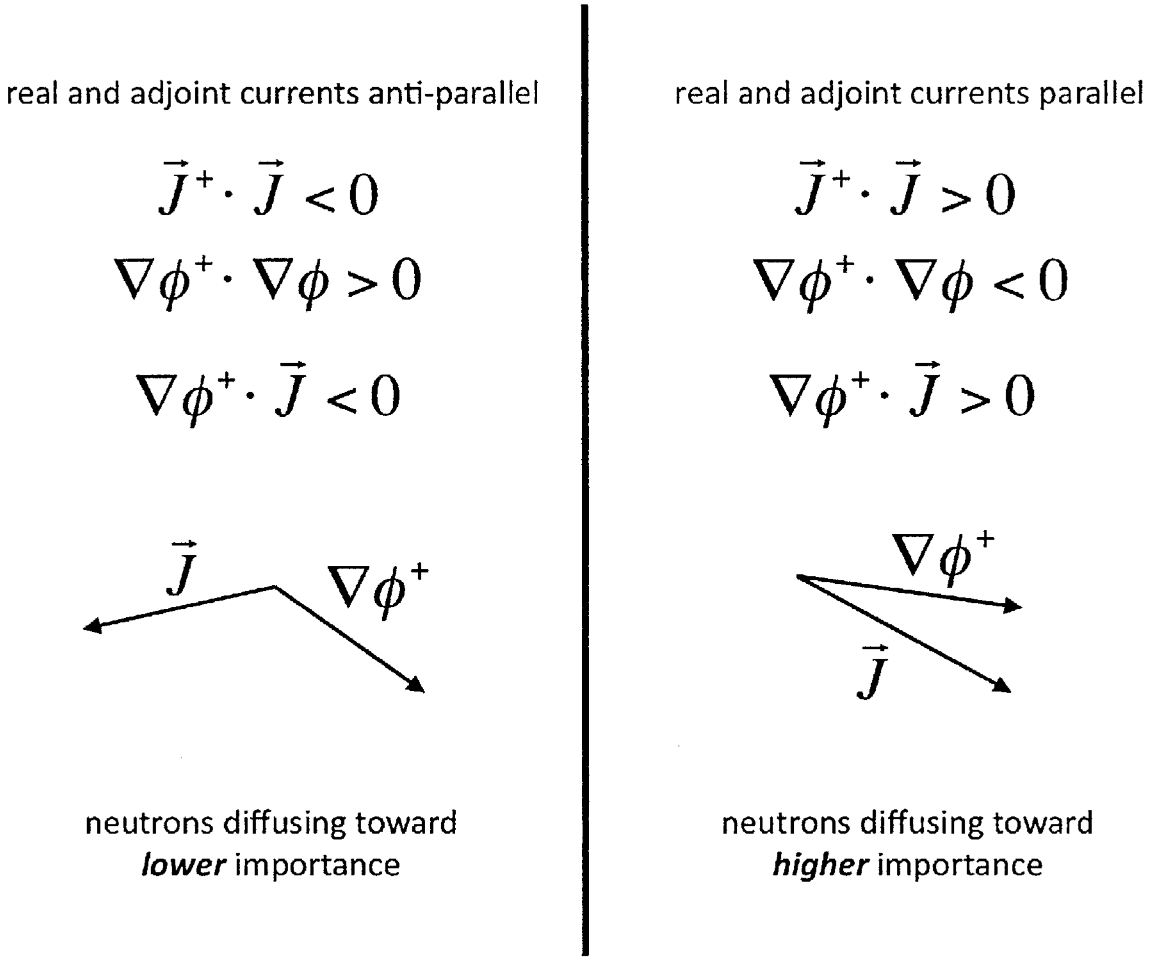
real and adjoint currents anti-parallel
$\vec{J}^{+} \cdot \vec{J} < 0$
$\nabla\phi^{+} \cdot \nabla\phi > 0$
$\nabla\phi^{+} \cdot \vec{J} < 0$
$\vec{J}$
$\nabla\phi^{+}$
neutrons diffusing toward *lower* importance
real and adjoint currents parallel
$\vec{J}^{+} \cdot \vec{J} > 0$
$\nabla\phi^{+} \cdot \nabla\phi < 0$
$\nabla\phi^{+} \cdot \vec{J} > 0$
$\nabla\phi^{+}$
$\vec{J}$
neutrons diffusing toward ***higher*** importance

Figure 11.11: The conceptual interpretation of real and adjoint currents as anti-parallel (as they usually are) and parallel (as they sometimes are in localized areas). Where the real and adjoint currents are parallel, neutrons diffuse toward higher importance.

In order to reconstruct an analytic representation for Equation 11.16, we must compute the product of two fluxes (or two components of a flux gradient). Equation 11.18 shows how to simply represent the product of two monomial sums as a single monomial sum.

$$\left[\sum_{i=1}^{I} C_{i,g} x^{m_i} y^{n_i}\right] \left[\sum_{j=1}^{J} C_{j,g} x^{m_j} y^{n_j}\right] = \sum_{i=1}^{I} \sum_{j=1}^{J} C_{i,g} C_{j,g} x^{m_i+m_j} y^{n_i+n_j} \tag{11.18}$$

Figure 11.12 shows the spatial distribution of this reconstructed "reactivity worth" in Equation 11.16 evaluated for a uniform sodium density perturbation. It is positive throughout the core except in regions of high flux gradients (high leakage), where it is negative. It is well-known that cores with high leakage have low sodium void worth. Cores with "pancake" or "parfait" (axially multilayered) geometries have been shown to mitigate the void worth problem [124,132].

Figure 11.12 also shows that sodium worth is also greatly reduced in regions immediately adjacent to withdrawn control rods, where the spectrum tends to be softer. Spectrum softening is also known to reduce sodium void worth. Strategically inserting small quantities of moderating material into a large SFR core can soften the neutron spectrum enough such that sodium voiding does not significantly harden the spectrum or increase reactivity [125,128].

Computing and visualizing adjoint-weighted quantities (such as sodium void worth or other reactivity coefficients) on the pin level or subchannel level can elucidate spatial effects in transients. It may also elucidate potential strategies for reducing whole-core reactivity coefficients.

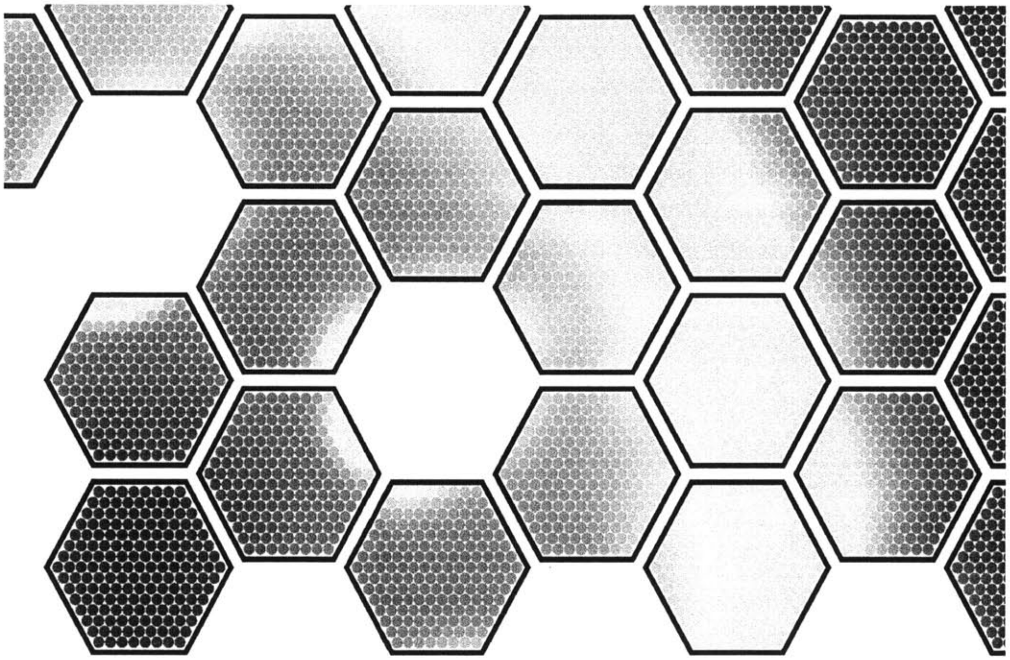

Figure 11.12: The relative sodium "reactivity worth" distribution for a small 1/3 test core model. This "reactivity worth" (which is proportional to the local sodium void reactivity coefficient per unit mass) is positive throughout most of the core, but it is significantly reduced in regions with (1) high leakage and (2) softened spectra. One can see an example of reduced void worth in case (2) surrounding withdrawn control assemblies. The magnitudes in this plot are relative. Although computing the product $\phi\phi^\dagger$ tends to amplify small errors in ϕ and $\phi^\dagger$, the general spatial trends will be correct.

11.5.6 Additional Considerations

Although we have unearthed an intriguing heterogeneous effect in which adding material increases leakage, we have only demonstrated its existence in reconstructed nodal diffusion theory.

One might suggest that this effect arises from errors in the flux reconstruction techniques. Figure B.6 shows that the flux reconstruction error magnitude does *not* peak in the vicinity of withdrawn control assemblies (or test assemblies) - the same regions in which this effect occurs. Refer back to Figures 11.7, 11.8, and 11.9. Thus, it is extremely unlikely that this effect is an artifact of the flux reconstruction techniques.

One also might suggest that this intriguing effect arises entirely from the diffusion approximation. If this were true, it would not occur in transport theory. However, we have already explained the origin of the effect intuitively. As shown in Figure 11.7, the effect arises from the fact that the real flux peaks in the vicinity of withdrawn control assemblies while the adjoint flux depresses in the same regions. We can easily explain this in terms of physics. The real flux peaks, because the extra sodium replacing the withdrawn control assembly weakly moderates the spectrum, causing the power in adjacent fuel pins to peak. The adjoint flux (the importance function) depresses, because neutrons heading into a fuel-less assembly are less likely to induce fission. Nothing about these physical arguments is tied to the diffusion approximation - they are equally valid when considering transport. Thus, it likely that a transport solution would also exhibit this effect.

Furthermore, note that when we say "adding material increases leakage", we define "leakage" as the leakage *term* in reactivity worth. Of course, there are also capture, scattering, and fission terms. It would be intriguing to find a scenario in which this leakage effect causes the *total* reactivity worth to be positive instead of negative (or vice versa).

11.6 Modeling Distortions via Monte Carlo

Now that we have discussed the flux reconstruction and PyPinPlot components of VirDenT, let us turn to the distortion reactivity. Consider arbitrary assembly displacements in the (x,y) plane with axial dependence. We cannot compute this directly via homogenized assembly diffusion. Thus, in order to validate it, we must turn to heterogeneous Monte Carlo with pin detail.

We choose to use MCNP [202]. ARMI can generate MCNP input files, which we can visualize via `mplot`. Figure 11.13 shows homogenized assembly MCNP. Essentially, this has the same homogenized geometry as a DIF3D model, except we solve it with Monte Carlo transport instead of diffusion. However, we cannot model arbitrary assembly distortions with this model, because we cannot move a single assembly without it overlapping its neighbors.

Figure 11.14 shows the same assemblies as Figure 11.13 with a heterogeneous MCNP model. This has full pin, clad, duct, and interstitial coolant detail. This allows us to move a single assembly within its hexagonal cell without overlapping its neighbors or losing mass. Figure 11.15 shows the same assemblies as Figure 11.14, except that the upper three assemblies have shifted radially outward by a couple of millimeters. Toggling back and forth between Figures 11.14 and 11.15 makes this evident.

Figure 11.16 zooms in on a distorted interstitial coolant zone between three neighboring assemblies. The orange , , and zones that meet in the center are all interstitial coolant, and one can see that the upper-right assembly has shifted slightly away from its two neighbors.

Figure 11.17 illustrates an exaggerated uniform radial swelling scenario in the same heterogeneous MCNP model. The assemblies do not swell, but they simply move further apart and open up larger interstitial coolant zones.

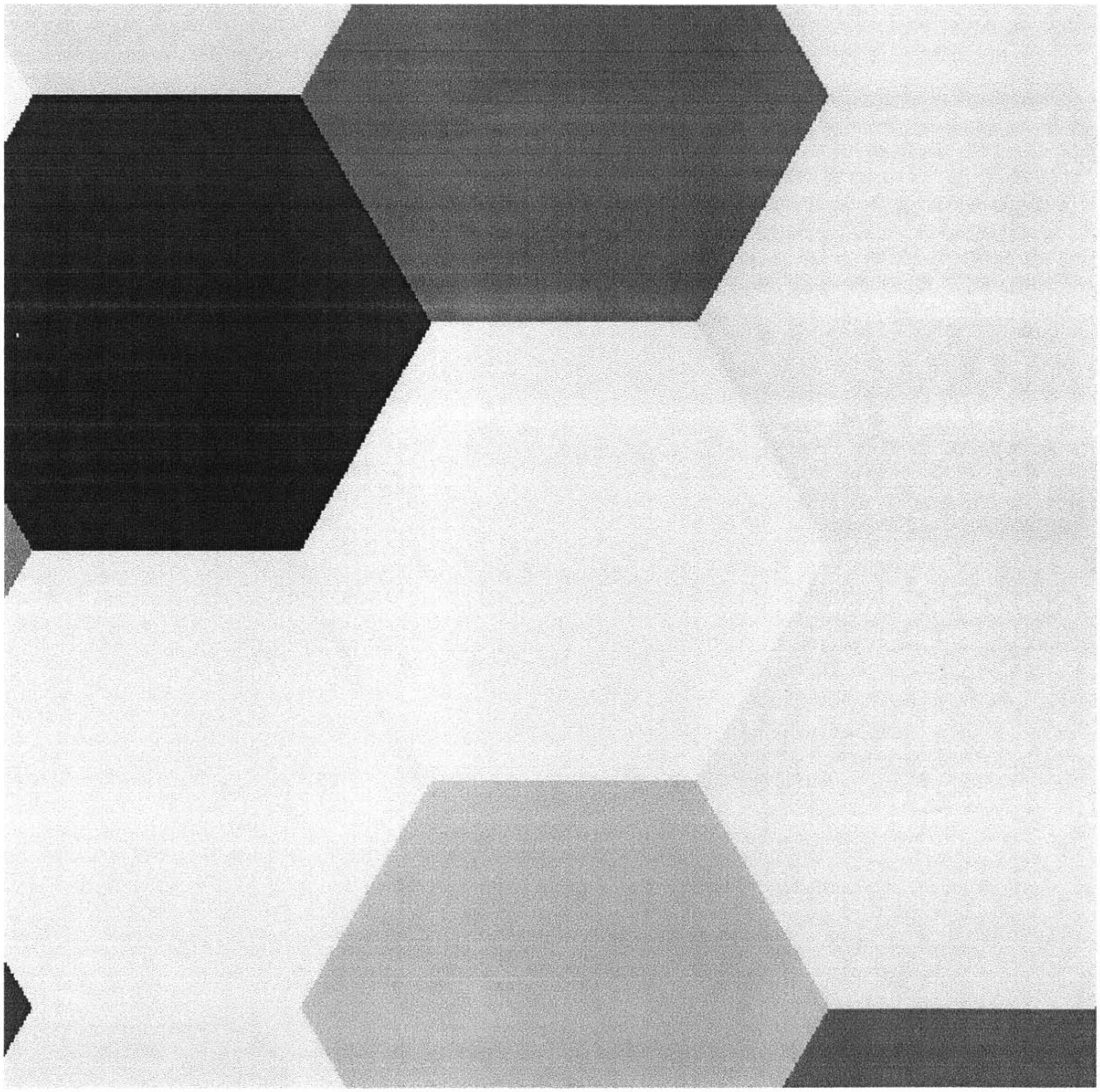

Figure 11.13: Homogenized assembly Monte Carlo displayed via `mplot` in MCNP.

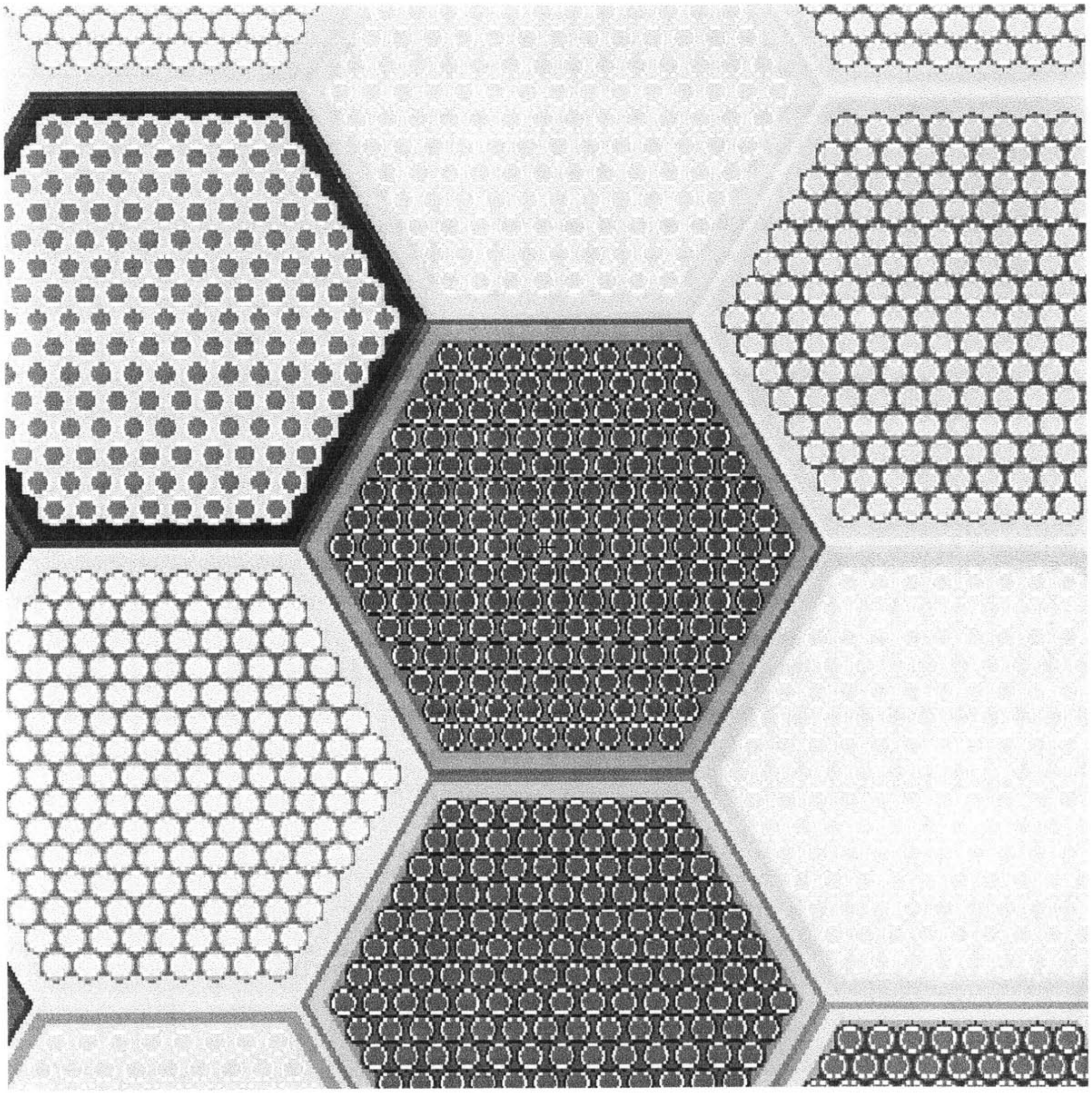

Figure 11.14: Heterogeneous pin-detail Monte Carlo without distortions displayed via `mplot` in MCNP.

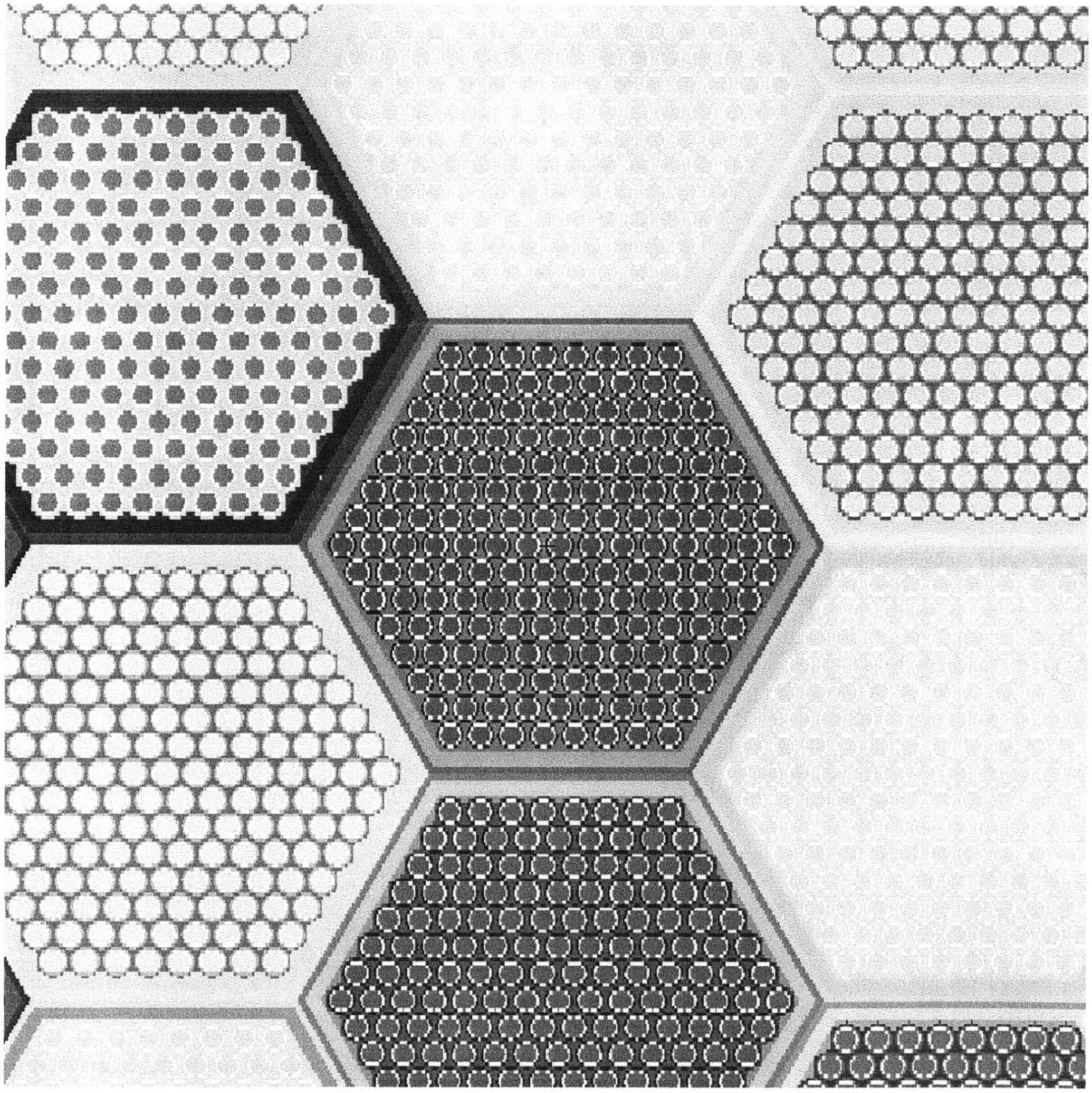

Figure 11.15: Heterogeneous pin-detail Monte Carlo with small assembly distortions displayed via `mplot` in MCNP.

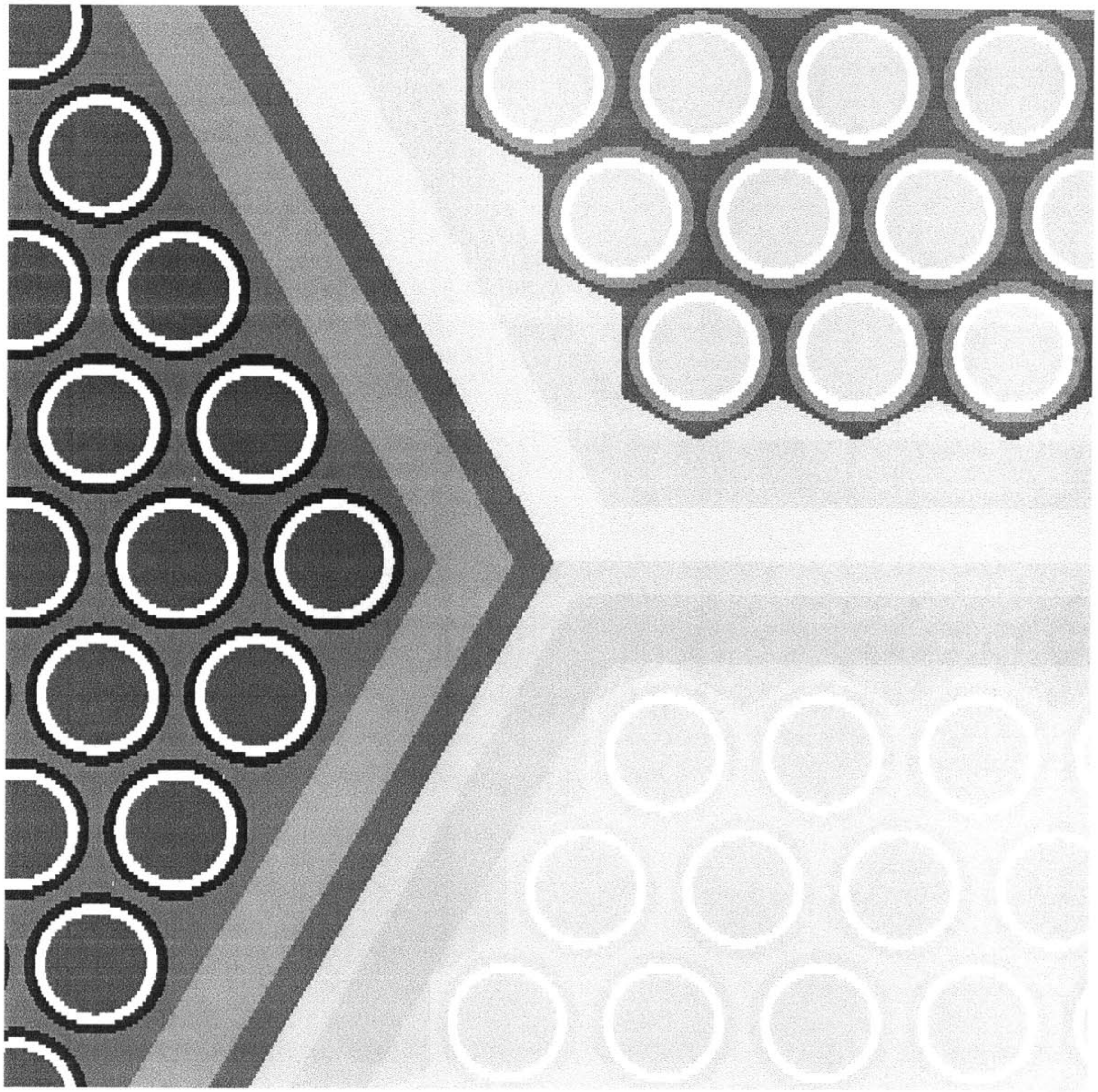

Figure 11.16: Zoomed-in heterogeneous pin-detail Monte Carlo displayed via `mplot` in MCNP.

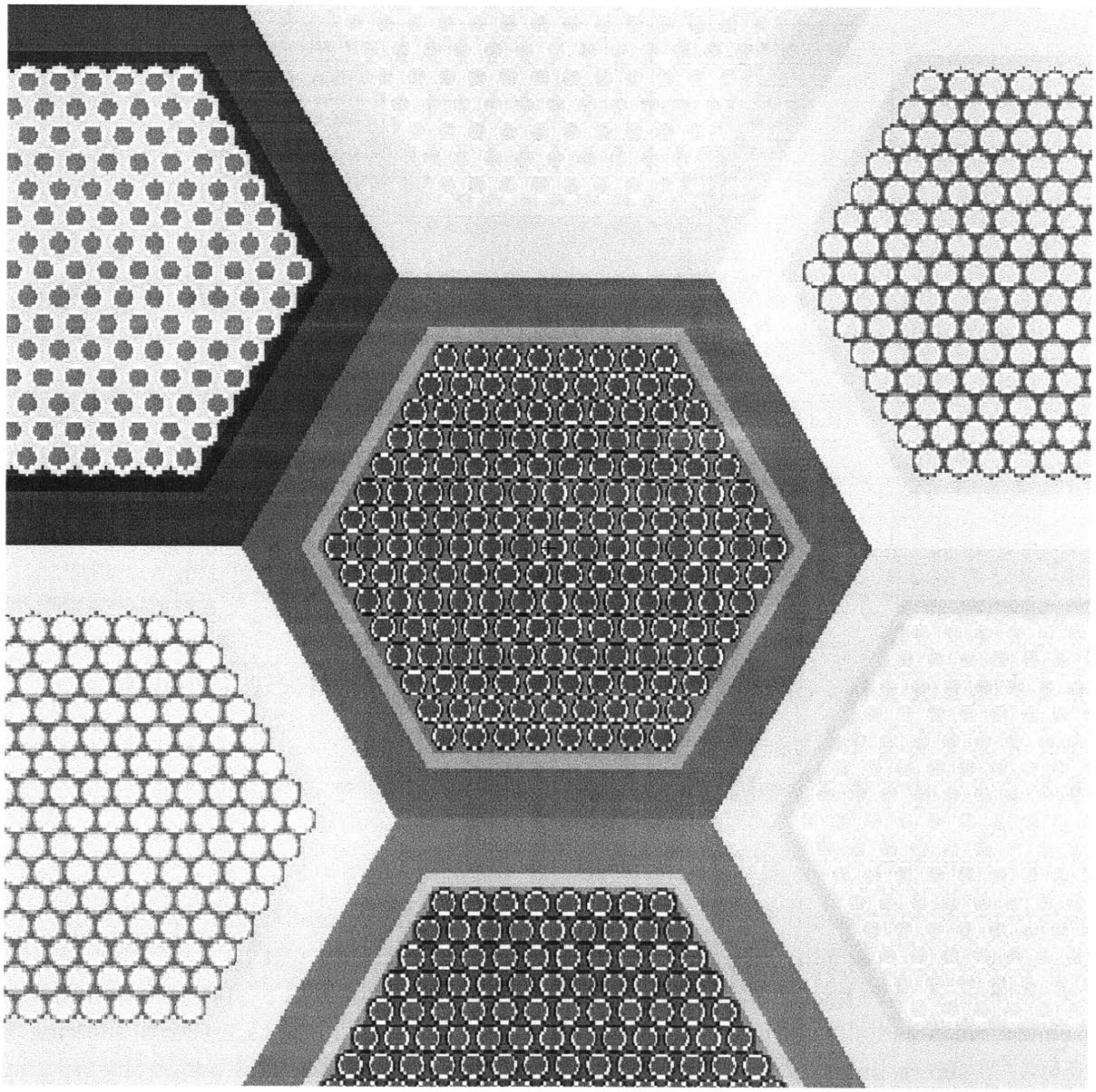

Figure 11.17: Uniform radial flowering in heterogeneous pin-detail Monte Carlo displayed via `mplot` in MCNP.

Table 11.1: The Linearity of Monte Carlo Reactivities for FFTF Ring 6 Shifting (1 pcm convergence). The unperturbed k_{eff} is 1.00555.

ring 6 shift (mm)	k_{eff}	$\Delta\rho$ (pcm)	pcm/mm
0.45	1.00530	-25 ± 2	-55.6 ± 4.4
0.90	1.00506	-49 ± 2	-54.4 ± 2.2
1.35	1.00484	-71 ± 2	-52.6 ± 1.5
1.80	1.00461	-94 ± 2	-52.2 ± 1.1

Table 11.2: The Linearity of Monte Carlo Reactivities for Test Core Rings 10,11,12 Shifting (1 pcm convergence). The unperturbed k_{eff} is 1.07887.

rings 10,11,12 shift (mm)	k_{eff}	$\Delta\rho$ (pcm)	pcm/mm
0.5	1.07878	-9 ± 2	-18.0 ± 4.0
1.0	1.07868	-19 ± 2	-19.0 ± 2.0
1.5	1.07860	-27 ± 2	-18.0 ± 1.3
2.0	1.07852	-35 ± 2	-17.5 ± 1.0

11.7 Properties of Assembly Distortion Reactivity in SFRs

Before commencing the VirDenT validation process, it is useful to assure ourselves of two properties of arbitrary assembly distortions: linearity and linear independence.

11.7.1 Linearity

Distortion reactivities are *linear* if reactivities are roughly proportional to assembly displacement. If an assembly moves 2 mm, the resulting reactivity will be twice that for a 1 mm movement. We generate heterogeneous MCNP models of FFTF and move individual assembly rings radially outwards by tiny amounts. Table 11.1 shows linearity for ring 6. Table 11.2 shows linearity for rings 10-12 moving together. Table 11.3 shows linearity for ring 10 alone. All these results clearly demonstrate that linearity holds for small distortion magnitudes to within 1σ uncertainty.

11.7.2 Linear Independence

The second property of arbitrary assembly distortions is *linear independence.* This means that the reactivity due to moving assemblies A and B together is the sum of that due to

Table 11.3: The Linearity of Monte Carlo Reactivities for Test Core Ring 10 Shifting (1 pcm convergence). The unperturbed k_{eff} is 1.07887.

ring 10 shift (mm)	k_{eff}	$\Delta\rho$ (pcm)	pcm/mm
1.0	1.07881	-6 ± 2	-6 ± 2
2.0	1.07875	-12 ± 2	-6 ± 1

Table 11.4: The Linear Independence of Monte Carlo Reactivities for FFTF Ring Shifting (1 pcm convergence). The unperturbed k_{eff} is 1.00555. Each ring moves radially outward by 1.8 mm.

ring(s)	k_{eff}	$\Delta\rho$ (pcm)	sum of $\Delta\rho$ (pcm) for each ring	error (pcm)
ring 4	1.00525	-30 ± 2	N/A	N/A
ring 5	1.00519	-36 ± 2	N/A	N/A
ring 6	1.00461	-94 ± 2	N/A	N/A
ring 7	1.00560	5 ± 2	N/A	N/A
rings 4,5	1.00491	-64 ± 2	-66 ± 4	-2 ± 6
rings 5,6	1.00424	-131 ± 2	-130 ± 4	1 ± 6
rings 4,5,6	1.00398	-156 ± 2	-160 ± 6	-4 ± 8

moving A and B separately. Table 11.4 demonstrates this for assembly ring motion in the same FFTF benchmark.

11.8 VirDenT Distortion Implementation

We attempt two distinct approaches to applying non-uniform anisotropic "virtual density" theory to arbitrary assembly displacements: interstitial coolant integration and homogenized triangle integration.

11.8.1 Interstitial Coolant Integration

Throughout Chapters 7 and 9, we studied Cartesian SFR cases in which the finite difference mesh contained explicit coolant gaps. These explicit gaps made computing arbitrary assembly motion quite easy; we simply applied "virtual density" theory to them. However, we now attempt to model arbitrary hexagonal assembly motion with a DIF3D mesh that lacks explicit gaps. We obtain the homogenized real and adjoint currents through the lateral assembly surfaces via DIF3D nodal, and then we use those currents to compute S, L_i, and Γ_i within the thin heterogeneous interstitial coolant zones along those surfaces.

Figure 11.18 illustrates the x, y, and z directions within one of these interstitial zones. In order to accurately apply non-uniform anisotropic "virtual density" to these zones, we must know the real and adjoint flux gradients accurately in all these directions. Unfortunately, this de-homogenization problem is not straightforward given fluxes computed on a homogenized mesh. Thus, we seek a different approach.

11.8.2 Homogenized Triangle Integration

The second method is to compute S, L_i, and Γ_i for each DIF3D finite difference triangle, which is straightforward. Then we can anisotropically swell each regular triangle to become an irregular triangle such that the original mesh of regular triangles now becomes a distorted mesh of irregular triangles. Figure 11.19 illustrates this process.

11.9 VirDenT Distortion Validation

We compute the "virtual density" S, L_i, and Γ_i quantities (using the DIF3D finite difference real and adjoint solutions) in the same way we did in Chapter 7.

11.9.1 Reactivity Coefficients: Fuel, Structure, Coolant, Radial Swelling, Axial Swelling

We compute the reactivity coefficients for fuel, structure, and coolant material densities as well as for uniform radial and axial swelling. Table 11.5 compares these coefficients computed via VirDenT to DIF3D eigenvalue references for the GMF SFR. All coefficients agree to within 1%, and similar results are found for any ARMI core.

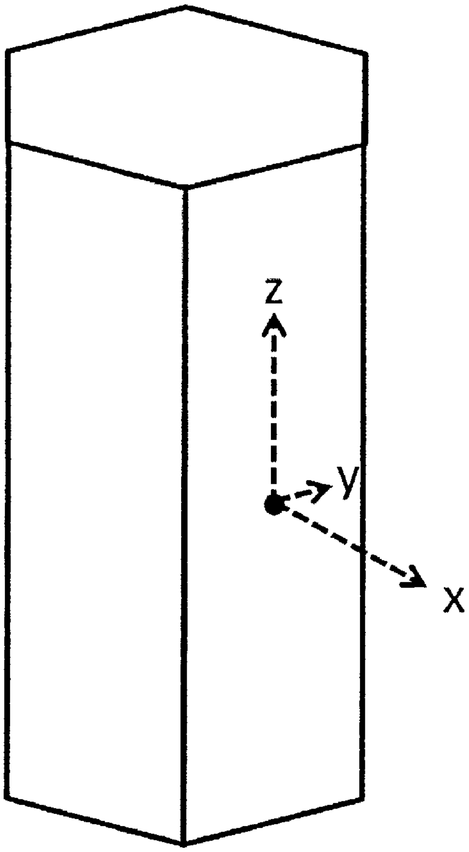

Figure 11.18: x, y, and z coordinates on the lateral surface of a hexagonal assembly, which is an interstitial coolant zone.

11.9.2 Arbitrary Distortions

NUBOW-3D [192] generates realistic 3-D assembly distortions in the form of (x,y) assembly displacements with axial dependence. VirDenT reads these distortions via ARMI. We generate heterogeneous Monte Carlo models for these displacements as illustrated by Figures 11.14 and 11.15. Converging these MCNP simulations for large SFR cores requires approximately 1 CPU year. Computing an unperturbed DIF3D finite difference solution with VirDenT pre-computation requires a couple CPU minutes, and computing arbitrary VirDenT reactivities from that solution requires a few CPU seconds.

Currently, we have obtained validation results only for proprietary cores, but we can say that VirDenT agrees with heterogeneous MCNP to within 1.5σ uncertainty for all of several distortion scenarios we have tested. The validation process is ongoing, and distortion scenarios in non-proprietary cores will be published in the open literature.

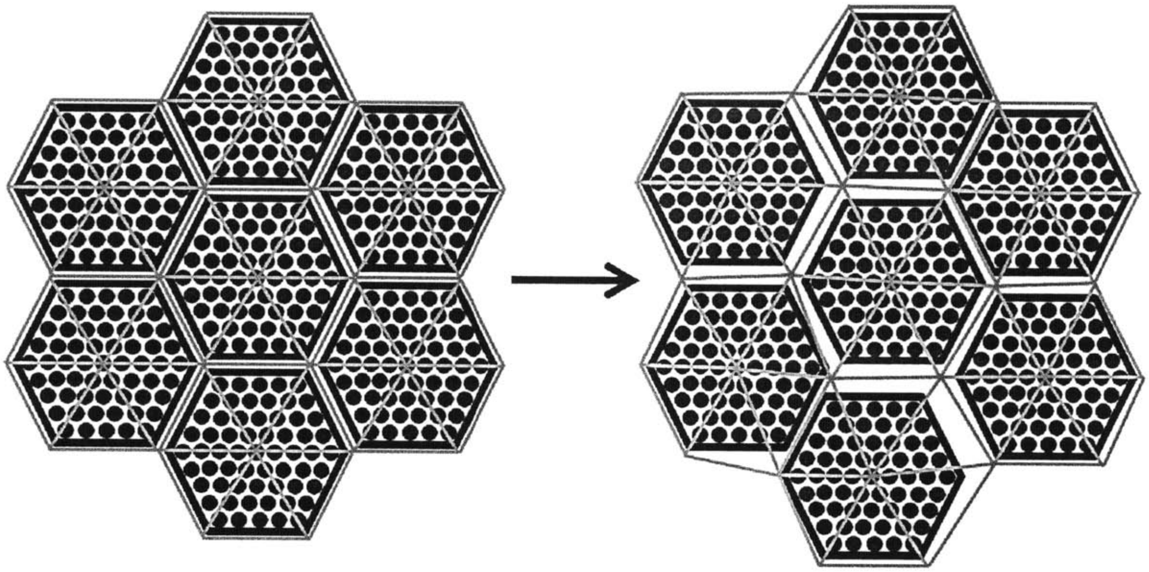

Figure 11.19: Transforming an un-distorted regular triangular mesh into a distorted irregular triangular mesh via non-uniform anisotropic swelling.

Table 11.5: Reactivity Coefficient Validation for the Generic Metal-Fueled (GMF) Test Core

	DIF3D/ARMI (pcm/%)	VirDenT (pcm/%)	error (%)
fuel density	-241.1477	-240.4323	0.297
structure density	-51.6302	-51.5873	0.083
coolant density	6.8586	6.8543	0.063
radial swelling	-343.7806	-345.1656	0.403
axial swelling	-124.1347	-124.3875	0.204

11.10 Summary

We have extended conventional flux reconstruction techniques to include vector fields and adjoint-weighted quantities. It is not unreasonable to surmise that an intra-assembly analytic representation of any neutronic quantity could be obtained. We have also created the suite of visualization tools dubbed PyPinPlot. In particular, the fission power gradient core maps are useful for thermal hydraulic and mechanical analysis, and the sodium worth core maps can potentially elucidate techniques for reducing core-wide (or at least localized) void worth. Furthermore, we have unearthed an intriguing physics scenario in which removing material actually decreases leakage.

Most significantly, VirDenT computes reactivity coefficients due to non-uniform anisotropic swelling of assembly zones (which direct diffusion theory cannot compute) in CPU seconds, while Monte Carlo (currently the only high-fidelity approach) requires CPU years to do the same. Preliminary results indicate that VirDenT agrees well with Monte Carlo, although the validation process is ongoing.

12 Conclusions and Future Work

12.1 Introduction

Here we summarize the major accomplishments of this work, and we ruminate over future, wider applications of "virtual density" theory.

12.2 Major Contributions and Ramifications

At our outset, we sought to develop a new method to accurately quantify negative reactivity feedback due to arbitrary geometry distortions in fast reactors. Previously, no method existed that was generic, efficient, and accurate.

We developed the "virtual density" theory of neutronics, which alters material density (isotropically or anisotropically) instead of explicitly changing geometry. While geometry is discretized, material densities occupy a continuous domain; this allows density changes to obviate the greatest computational challenges of geometry changes. Although primitive forms of this theory exist in Soviet literature, they are only applicable to cases in which entire cores swell uniformly. Thus, we conceived a much more general and pragmatic form of "virtual density" theory to model non-uniform and localized geometry distortions via perturbation theory. This process required inventing new "surface leakage" perturbation quantities and distinguishing them from existing "volume leakage" quantities.

In order to efficiently validate "virtual density" perturbation theory, we conceived the "virtual mesh" method for diffusion theory. This new method involves constructing a slightly perturbed "fake" mesh that produces correct first-order reactivity and flux shifts due to anisotropic swelling or expansion of individual mesh cells. First order reactivities computed on a "virtual mesh" agree with continuous energy Monte Carlo to within 1σ uncertainty.

We validated "virtual density" theory via the "virtual mesh" method in 3-D coarse mesh models of the Fast Flux Test Facility (FFTF) and Jōyō benchmarks using the MATLAB-PETSc-SLEPc (MaPS) multigroup finite difference diffusion code, which we developed for this purpose. We modeled a panoply of non-uniform anisotropic swelling scenarios, including axial swelling of individual assemblies, axial swelling of each mesh cell in proportion to its fission power, and radial core flowering with arbitrary axial dependence. In 3-D coarse mesh Cartesian cores with explicit coolant gaps, we modeled individual assembly motion, assembly row motion with arbitrary axial dependence, and assembly row "s-shape" bowing. In all cases, we found that "virtual density" perturbation theory predicts reactivity coefficients that agree with "virtual mesh" reference cases to within 0.01%. These reactivity coefficients are two to four orders of magnitude more accurate than those computed via boundary perturbation theory. We also developed the Pseudo-Seismic (PseuSei) Animator within MaPS to explore point-kinetic effects of arbitrary assembly motion for 3-D coarse mesh

Cartesian cases. In general, this "virtual density" perturbation method can precisely predict reactivity coefficients due to anisotropic swelling or expansion of any core region in *any* direction.

Furthermore, we computed flux and power shift distributions due to geometry distortions. We found that our "virtual density" formalism couples seamlessly with existing modal expansion perturbation theory (MEPT) formalism, and we used the resulting new hybrid method to compute flux and power shifts due to arbitrary anisotropic swelling of arbitrary core regions. We tested this new method for a large, highly-heterogeneous Cartesian core, and we found that predicted (global and local) flux and power shift distributions typically agree with "virtual mesh" reference cases to within a few percent.

Development of the "Virtual Density" Theory (VirDenT) industry code constituted the culmination of this work. This parallelized Python code computes "virtual density" reactivity coefficients given a DIF3D flux solution as input. VirDenT contains a flux reconstruction module that computes individual pin powers from a homogenized nodal diffusion solution. It also contains PyPinPlot, a high-resolution visualization tool for pin-level powers, fluxes, and current vector fields. Most importantly, VirDenT computes reactivity coefficients due to local anisotropic swelling of assembly zones (which direct diffusion theory cannot compute) in CPU seconds, while Monte Carlo (currently the only high-fidelity approach) requires CPU years to do the same. Preliminary results indicate that VirDenT agrees well with continuous energy, pin-detail heterogeneous Monte Carlo, although the validation process is ongoing.

In conclusion, we successfully conceived a new method based on new fundamental neutronic principles and guided it from academic research to practical industry application. This method is generic, accurate, and efficient. While most of the neutronics community pursues hyperfine spatial resolution in order to model distortions, "virtual density" theory allows one to model distortions with relatively coarse spatial resolution. Most importantly, it allows the fast reactor community to more accurately quantify the neutronic effects (especially negative reactivity feedback) due to geometry distortions. Accurately knowing this feedback magnitude can enable SFR designers to justify a positive sodium void reactivity to regulators - an important step toward breaking past the light water paradigm.

12.3 Future Work

We intend the "virtual density" theory of neutronics to be seminal. It is readily extensible not only throughout neutronics, but perhaps also beyond neutronics. Here we will ponder over some potential future applications.

12.3.1 Spatial Kinetics with Distortions: Incorporating "Virtual Density" Theory into Direct Diffusion Solutions

People perform spatial kinetics for fast reactors quite often. However, none of these spatial kinetics simulations include geometry distortions. The code DIF3D-K does even include uniform radial core swelling[194]. Fast reactor modeling at Argonne National Laboratory includes the Wigeland reactivity models[105,106,107,108] obtained via SASSYS[196], but this is a single whole-core value that does not include non-uniformity of radial swelling. The Wigeland model has no spatial distribution, and so it does not allow for spatial kinetics. Argonne also sometimes employs the Knutson reactivity models [101,102] obtained via NUBOW-3D[192], but these do not agree well with diffusion references for uniform radial swelling. Refer back to Sections 2.6.1 and 2.6.2 for more detailed explanation of the Knutson and Wigeland models. In summary, spatial kinetics have apparently never included arbitrary geometry distortions. This is a major limitation. If distortions constitute the dominant negative reactivity feedback (as FFTF experimental data has shown[106]), then how can we say spatial kinetics is correct without distortions?

Currently, the only options for spatial kinetics with distortions are (1) flux tallying via Monte Carlo at each time step and (2) diffusion or deterministic transport calculations on a hyperfine mesh. Option (1) is prohibitively expensive, and option (2) is still quite expensive and is complicated by the need to re-mesh at each time step. One can only eliminate re-meshing effects by employing meshes that are much more fine than the distortions themselves, which causes computational expense to exponentially increase.

We could attempt spatial kinetics with distortions using the VirDenT-MEPT method described in Chapter 10. Modal expansion was originally intended to model transients, and Duderstadt's textbook uses modes to perform analytical transient analysis for a homogeneous 1-D slab[157]. However, the neutronics community has long since bypassed modes as a means for spatial kinetics in favor of directly solving the fundamental flux distribution at each time step. Although we have shown that VirDenT-MEPT can work for heterogeneous cores, the number of required modes could be discouraging to potential users.

Accordingly, a less intimidating (and thus more popular) approach might be solving the diffusion equation directly using anisotropic "virtual density". Although the focus of this thesis is perturbation theory, finding a way to solve the eigenvalue problem directly for distorted geometries on the same mesh might be very advantageous. Adjusting material densities and anisotropic diffusion coefficients to directly solve a distorted geometry is trivial *as long as* that distortion can be expressed using only S and L_i. These include all uniform swellings as well as any non-uniform swelling that extends through the whole core in at least one dimension (such as a through-core axial slice or an entire assembly). However, any "virtual density" perturbation that requires Γ_i terms (such as an arbitrary interior core zone) cannot be solved directly via standard diffusion theory. In order to incorporate Γ_i

terms into a direct eigenvalue solution (rather than perturbation theory), we would need to invent "surface discontinuity" or "surface slipping" factors for diffusion theory. This effort may be the most salient future application of "virtual density", as it would enable direct simulation of time-dependent distortion effects on a coarse mesh without re-meshing.

12.3.2 "Virtual Density" Theory in Transport Theory

Although pure diffusion theory is accepted as adequate for fast reactors with long neutron mean free paths, adequately modeling pin lattice effects in thermal reactors (such as LWRs) requires at least some transport theory. One cannot simply homogenize pin detail in LWRs as in SFRs without any additional pin-detail computation.

One could implement "virtual density" theory in deterministic transport theory. Although transport theory is undoubtedly more complex than diffusion theory (it has the additional angular variable), "virtual density" theory *might* actually be easier to implement in certain types of deterministic transport. Unlike diffusion, transport contains explicit angular dependence of the flux, which might ease anisotropic "virtual density" implementation.

Method of characteristics (MOC) might afford the most straightforward implementation. Unlike in diffusion, we need not separate spatial and spectral terms or differentiate between volume and surface leakage. Formulating the quantities S, L_i, and Γ_i constituted most of our research time and effort in diffusion, but these quantities are not separate in transport. Consider the differential transport equation defined along a single characteristic (a line or "track" through space):

$$
\begin{aligned}
\hat{\Omega}\cdot\nabla\psi(\vec{r},\hat{\Omega},E) &= -\Sigma_t(\vec{r},E)\psi(\vec{r},\hat{\Omega},E) && (12.1)\\
&+ \int_0^\infty dE'\int_0^{4\pi} d\hat{\Omega}'\Sigma_s(\vec{r},\hat{\Omega}\cdot\hat{\Omega}',E'\rightarrow E)\psi(\vec{r},\hat{\Omega}',E') && (12.2)\\
&+ \frac{\chi(\vec{r},E)}{4\pi k}\int_0^\infty dE'\nu(\vec{r},E')\Sigma_f(\vec{r},E')\int_0^{4\pi} d\hat{\Omega}'\psi(\vec{r},\hat{\Omega}',E') && (12.3)
\end{aligned}
$$

We can easily see that increasing material density along a track while simultaneously decreasing the *length* of that track will have no effect on the equation. It follows that we can replicate the effects of a track length change with a material density change. Instead of making a track segment longer, we simply increase all material densities along that track segment. Although we numerically demonstrated uniform isotropic "virtual density" for MOC in Section 3.3.5, we now see that non-uniform anisotropic "virtual density" for MOC is almost as simple. The only complication is defining material densities for each track segment rather than for each spatial region. Different tracks in the same spatial region must "see" different (or "virtual") material densities.

Now consider moving an LWR fuel pin slightly in the 2-D plane. Instead of "actually" moving the pin, we can "virtually" change material densities along track segments on either side of the pin. Track segments within the pin itself will see no density change. Although one could certainly "actually" move a pin and obtain a result, ensuring accuracy of that result might require hyperfine track spacing (or even re-tracking). In contrast, "virtual density" MOC could obtain an accurate result with a relatively coarse track spacing without re-tracking.

Although MOC "virtual density" could be implemented in terms of perturbation theory (given real and adjoint fluxes along each track), more people might be interested a direct eigenvalue solution implementation. In fact, a direct "virtual density" eigenvalue solution for MOC avoids the Γ_i term obstacle in diffusion.

This same concept is likely extensible to any deterministic transport method with explicit tracks, such as S_N. Methods without explicit tracks, such as P_N, could pose greater challenges.

12.3.3 "Virtual Density" Theory for LWRs

Our derivation of "virtual density" theory assumed nothing about spectrum or mean free path length. It must be equally valid for reactors with thermal spectra and short mean free paths. Thus, assuming that we successfully implement "virtual density" in deterministic transport, we can readily apply "virtual density" to LWRs. There are two potential applications:

- The Consortium for Advanced Simulation of LWRs (CASL) has flagged fuel assembly distortions (FADs) as a potential priority. Most interest in FADs regards determining the *mechanical* behavior of LWR fuel assemblies, and many people assume that the neutronic response can be "brute forced" via Monte Carlo or via deterministic methods with exotic hyperfine meshes or track spacings. However, direct "virtual density" eigenvalue solutions could obviate the need for exotic hyperfineness.

- The neutronic seismic behavior of LWRs (or any reactor) has not been well quantified. One could use direct "virtual density" eigenvalue solutions in MOC to compute the neutronic effects of arbitrary pin and assembly movements in the 2-D plane. Furthermore, one could compute the MOC adjoint solution and use "virtual density" perturbation theory to rapidly compute the reactivity coefficients $d\rho/dx$ and $d\rho/dy$ for individual pin or assembly movements in each direction.

12.3.4 "Virtual Density" Theory for Detectors and Security

If we can implement "virtual density" theory in deterministic transport, it immediately becomes applicable to fixed source detector problems. In fact, much of the most recent boundary perturbation theory work by Favorite is applied to detectors [60,65] for security applications, sometimes for photon transport. Detection and imaging transport problems routinely involve highly-complex geometries, such as objects passing through border security. Application of "virtual density" theory to these problems could provide two advantages:

- One could simplify complex geometries (or complex geometry changes) by applying "virtual density" to direct fixed source solutions. Some geometries may be tortuously difficult to express in terms of code input, but material densities are always easy to express.
- In critical reactors, the adjoint represents "importance" with respect to the fission reaction rate, and perturbation theory yields the sensitivity of eigenvalue. In fixed source detector problems, one can define the adjoint to represent "importance" with respect to a given detector response, and perturbation theory yields the sensitivity of that detector response. Thus, implementing "virtual density" perturbation theory in a fixed source detector code could rapidly determine sensitivity of a detector response with respect to various complex geometry changes that might be toilsome to solve directly. Various boundary perturbation theory functionals are more or less accurate for certain cases, but "virtual density" is generically accurate.

12.3.5 "Virtual Planck" Theory in Quantum Mechanics

One could apply the mathematics of "virtual density" theory to similar partial differential equations (PDEs) in other fields. First consider the Schrödinger equation, which is very similar in form to the neutron diffusion equation. It has a potential V in place of cross-sections and energy eigenvalues in place of k eigenvalues. The steady-state Schrödinger equation is

$$-\frac{\hbar^2}{2m}\nabla^2\psi_n(\vec{r}) + V(\vec{r})\psi_n(\vec{r}) = E_n\psi_n(\vec{r}) \tag{12.4}$$

Here the subscript n indexes the infinite number of eigenfunctions ψ_n and their corresponding eigenvalues E_n. $\hbar$ and m are Planck's constant and particle mass. We can define the Hamiltonian operator $\hat{H}$ in order to express this as an eigenvalue problem with linear operators:

$$\hat{H}\psi_n = E_n\psi_n \tag{12.5}$$

We can also define an equivalent adjoint eigenvalue problem, just as in neutron diffusion.

$$\hat{H}^\dagger \psi_n^\dagger = E_n \psi_n^\dagger \tag{12.6}$$

The ψ_n wavefunctions are always normalized such that $\left\langle \psi_n^\dagger \middle| \psi_n \right\rangle = 1$. This is consistent with the physical interpretation that $\psi_n^\dagger \psi_n = |\psi_n|^2$ is a probability distribution for a particle. Thus, the real and adjoint ψ_n are not only orthogonal, but also orthonormal.

$$\left\langle \psi_m^\dagger \middle| \psi_n \right\rangle = \delta_{mn} \tag{12.7}$$

These days, perturbation theory is much more widely used in physics and quantum engineering than in nuclear engineering. Since the Schrödinger equation has few precise analytical solutions, quantum physicists solve the system for an approximate potential and then employ perturbation theory to obtain a solution closer to that for the true potential.

Consider perturbing the potential V so that there is a perturbed Hamiltonian $\hat{H}'$.

$$\hat{H}' = \hat{H} + d\hat{H} \tag{12.8}$$

This Hamiltonian perturbation induces perturbations in the wavefunctions and eigenvalues.

$$\psi_0' = \psi_0 + d\psi_0 \tag{12.9}$$

$$E_0' = E_0 + \Delta E_0 \tag{12.10}$$

Now the first order eigenvalue change is

$$\Delta E_0 = \left\langle \psi_0^\dagger \middle| d\hat{H} \psi_0 \right\rangle \tag{12.11}$$

This expression is ubiquitous throughout quantum mechanics. It is mathematically equivalent to classic first order perturbation theory (see Equation 2.23) in neutronics except for the lack of a perturbation denominator, because the wavefunctions have already been normalized to probability.

Modal expansion techniques are also ubiquitous throughout quantum mechanics. One can express first order changes in wavefunction distribution as an infinite weighted sum of wavefunction modes.

$$d\psi_0 = \sum_{n=1}^{\infty} a_n \psi_n \tag{12.12}$$

The a_n coefficients are

$$a_n = \frac{\left\langle \psi_n^\dagger \middle| d\hat{H}\psi_0 \right\rangle}{E_0 - E_n} \tag{12.13}$$

Compare Equations 12.12 and 12.13 (Schrödinger) to Equations 10.6 and 10.7 (neutron diffusion). Again, these are mathematically equivalent save for the fission normalization in neutron diffusion. Although seldom used in modern neutronics, modal expansion is widely used throughout physics and is standard in many introductory quantum physics texts [162,161].

Now consider how to perform *geometry* perturbations in quantum mechanics, which amount to changing the domain of V or the positions of "step changes" in V. Currently, physicists use Equations 12.11, 12.12, and 12.13. Consider a step change (an instantaneous rise or drop) in potential that slightly shifts position. This is a large potential change within a small volume, just like a large change in cross-sections within a small volume when a material boundary shifts position in neutronics. Instead of "actually" moving this potential step change, we can "virtually" adjust the potential magnitude on either side of the step.

Of course, we would need some way of changing this potential anisotropically. We can accomplish this by breaking up Planck's constant into separate directions. Now the spatial term in the Schrödinger equation looks like this:

$$-\frac{\hbar^2}{2m}\nabla^2\psi = -\frac{\hbar}{2m}\left(\frac{d}{dx}\hbar_x\frac{d\psi}{dx} + \frac{d}{dy}\hbar_y\frac{d\psi}{dy} + \frac{d}{dz}\hbar_z\frac{d\psi}{dz}\right) \tag{12.14}$$

Of course, every physicist knows that Planck's constant is not anisotropic - it is identical for particle travel in every direction. It is a universal physical constant, and tampering with it is taboo! However, Equation 12.14 introduces a "virtual Planck" $\hbar_i$ in each direction i. Thus, instead of performing a large potential perturbation within a small volume, we could perform small "virtual Planck" perturbations spread out over larger volumes. We suspect the advantages of "virtual Planck" perturbation theory would be comparable to those of "virtual density" perturbation theory in neutronics, but further investigation is required.

12.3.6 "Virtual Conductivity" and "Virtual Viscosity" in Thermal Hydraulics

Beyond eigenvalue problems and perturbation theory, we can consider incorporation of "virtual" quantities within a wider range of PDEs. First consider the heat equation, which governs temperature distributions in solid fuel elements and structural materials within reactor cores. Although not an eigenvalue problem, the heat equation *is* the diffusion equation.

$$\rho(\vec{r})c_p(\vec{r})\frac{dT(\vec{r},t)}{dt} = \nabla \cdot k(\vec{r})\nabla T(\vec{r},t) + \dot{Q}(\vec{r},t) \tag{12.15}$$

Here ρ is material density, c_p is specific heat capacity, and k is thermal conductivity. $\dot{Q}$ is a volumetric heat source, and temperature T is the quantity we solve for. In the steady-state case, this is mathematically similar to a one-group fixed source problem in neutron diffusion.

Consider modeling geometry changes in fuel elements. Instead of "actually" swelling or deforming a fuel pin, we can "virtually" alter the thermal conductivity anisotropically. Consider breaking up k into directional components:

$$\nabla \cdot k\nabla T = \frac{d}{dx}k_x\frac{dT}{dx} + \frac{d}{dy}k_y\frac{dT}{dy} + \frac{d}{dz}k_z\frac{dT}{dz} \tag{12.16}$$

Now the anisotropic thermal conductivities serve the same function as directional diffusion coefficients in neutronics. In this way, we have "virtual conductivity" in heat diffusion just as we have "virtual density" in neutron diffusion. One could explore coupling these two methods to incorporate fuel temperature feedback in "virtual density" simulations.

However, fully coupling neutronics with thermal hydraulics requires not only conduction within fuel elements, but also coolant flow. The Navier-Stokes equation for incompressible fluid flow is

$$\rho\left(\frac{d\vec{v}}{dt} + \vec{v}\cdot\nabla\vec{v}\right) = -\nabla P + \mu\nabla^2\vec{v} + \vec{F} \tag{12.17}$$

Here ρ is fluid density, P is a pressure distribution, μ is viscosity, and $\vec{F}$ is any external body force (such as gravity). The goal is to solve for the distribution of the velocity vector $\vec{v}$. The convection term $\vec{v}\cdot\nabla\vec{v}$ is non-linear, but that does not necessary preclude a "virtual" principle. It may be possible to introduce an anisotropic "virtual viscosity" into the viscous term. Of course, we would also need to introduce artificial anisotropy into the convection term. More investigation is needed to ascertain the viability of these ideas. However, even if this works, people only solve Equation 12.17 directly for *laminar* flow. Solving *turbulent* flow usually involves time-averaging Equation 12.17 for computational fluid dynamics (CFD). It is unclear whether this time-averaging would ruin the "virtual" principle. Also, correlations used in subchannel analysis may or may not be well-disposed to a "virtual" principle.

12.3.7 Toward a Pristine "Virtual Reality"

We have outlined many potential future applications of "virtual density" theory beyond neutronics. Some applications, such as the Schrödinger equation, are straightforward low-risk endeavors, while others, such as the Navier-Stokes equations, are substantially less straightforward and may not work well at all.

Nevertheless, "virtual density" theory and its non-neutronic cousins represent a departure from current conventional wisdom. While most worldwide computation efforts focus on simulating "actual reality" as closely as possible to maximize fidelity, we take a different approach. We reject the necessity of tortuously modeling a reality that is distorted, crooked, and disfigured. Instead, we transform that disfigured reality into a simpler, smoother, and more pristine "virtual reality".

A The MATLAB-PETSc-SLEPc (MaPS) Finite Difference Diffusion Code

We develop a multigroup finite difference diffusion code to experiment with and validate various geometric perturbation theory techniques. We cannot simply use existing diffusion codes, because they do not allow us to solve the "virtual mesh" cases described in Chapter 6. Also, developing a diffusion code with a MATLAB interface allows for rapid trial and analysis of numerous cases.

A.1 Overview

This code is written primarily in MATLAB [210], but it is coupled with the PETSc [200] and SLEPc [203] linear algebra packages, which are written in C and C++. The preexisting MATLAB interface for SLEPc was modified to allow MATLAB to access more SLEPc functionality. The MATLAB code constructs the multigroup diffusion matrix operators for a given geometry with vacuum, reflective, or albedo boundary conditions. Once the diffusion operators are constructed, the MATLAB code feeds them to the PETSc-SLEPc solver, which returns the eigenvalue and eigenvector for a generalized eigenvalue problem. We refer to this MATLAB-PETSc-SLEPc diffusion solver as MaPS.

We prefer a Krylov-Schur outer iteration, which we find to be most efficient for our cases, but we also sometimes use Arnoldi or "power" outer iterations. Note that power iterations can only obtain the fundamental real and adjoint flux distributions, not any of the higher modes/harmonics. We employ a "gmres" solver for the inner eigenvalue iterations, an "ilu" preconditioner, and a "shift" spectral transform. Due to the low dominance ratios of fast reactor eigenvalue problems, this typically results in $\sim 10^{-10}$ eigenvalue convergence after only one or two Krylov-Schur or Arnoldi iterations.

MaPS takes arbitrary microscopic or macroscopic group cross-sections as input. Throughout this work, we use MC**2 [187] to generate all multigroup cross-sections. We can either (1) run MC**2 independently and read the microscopic cross-sections into MATLAB via Python for Nuclear Engineering (PyNE) [212] or (2) export macroscopic cross-sections (derived from MC**2 microscopic cross-sections) from the Advanced Reactor Modeling Interface (ARMI) [211] to a text file format that MATLAB can read.

A.2 Geometry Options

MaPS contains a number of distinct geometry options:

1. 1-D Cartesian

2. 2-D Cartesian
3. 3-D Cartesian
4. 2-D Hexagonal (each hexagon subdivided into 6 triangles)
5. 3-D Hexagonal-Z (each hexagon subdivided into 6 triangles)
6. 2-D Hyperfine Hexagonal (each hexagon subdivided into an arbitrarily large number of triangles)

MaPS does not contain a 3-D hyperfine hexagonal-z option, because the computational performance would be poor. The purpose of hyperfine hexagonal-z is validation of flux reconstruction methods, for which we deem 2-D to be sufficient.

Figure A.1 illustrates the subdivision of each hexagon into 6 triangles. If the indices (i, j) specify a unique hexagonal assembly, then the indices (i, j, k) specify a unique triangle within a unique assembly. The index k ranges from 1 to 6 and specifies corners and surfaces of an assembly, as well as one of the 6 triangles within that assembly. Like the ARMI [211] assembly ordering and ARMI pin ordering described in Appendix B, the k index runs counter-clockwise. $k = 1$ corresponds to the assembly surface and triangle at 2 o'clock, as well as the assembly corner at 1 o'clock. Compare Figures A.1 and ?? for a very clear picture of this.

Figure A.2 illustrates the subdivision of a triangle (one of 6 within a single hexagon) into an arbitrarily large number of hyperfine triangles. These hyperfine triangles are indexed by (k, u, v) within a single hex as shown, and they can be indexed by (i, j, k, u, v) within an entire 2-D core.

The maximum value of u fully determines how many hyperfine triangles fit within a single hexagon. For example, $u_{\text{max}} = 5$ will subdivide a hexagon into 150 triangles, while $u_{\text{max}} = 7$ will subdivide it into 294 triangles. So a hexagon is subdivided into $6u_{\text{max}}^2$ triangles. Thus, the number of equilateral triangles within a hexagon is always six times a perfect square.

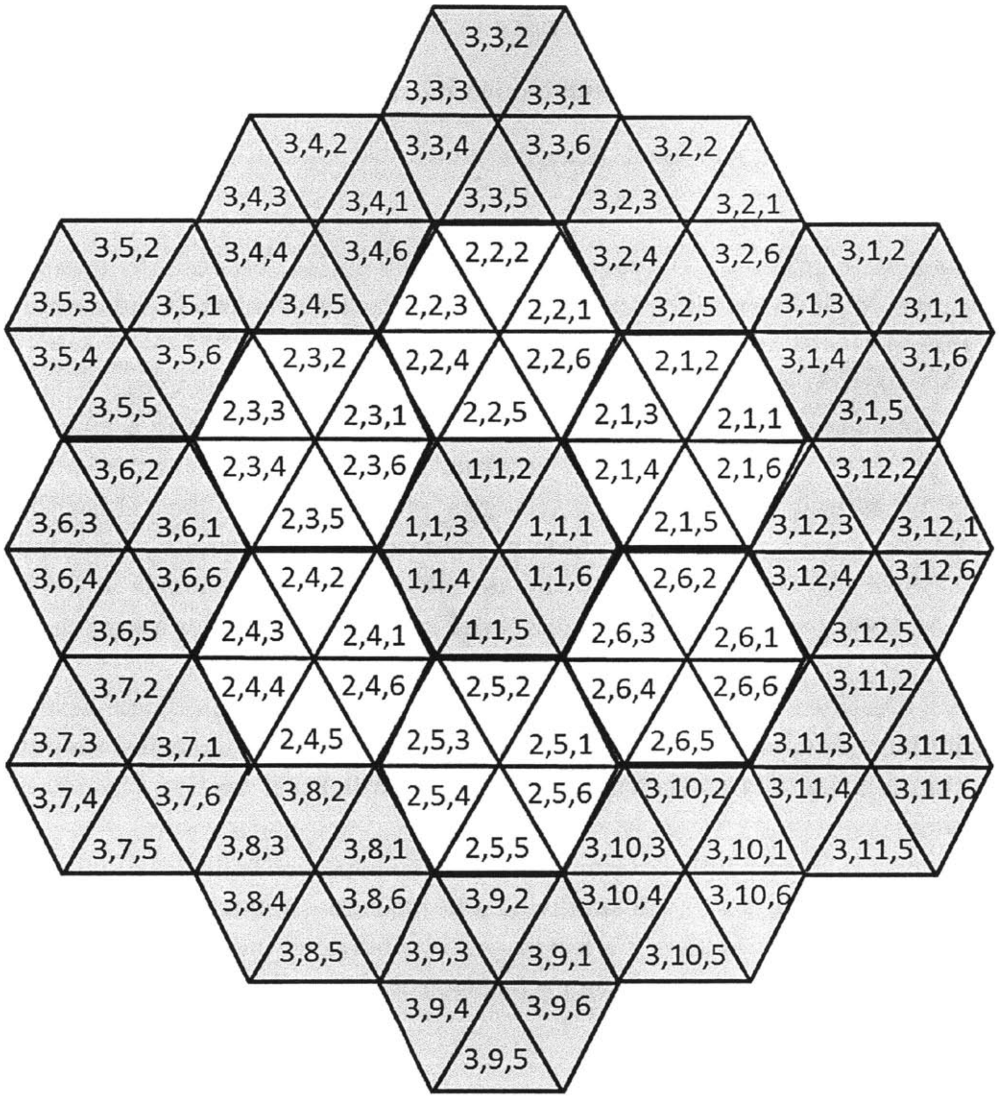

Figure A.1: The subdivision and indexing of six triangles within each hexagonal assembly in MaPS. Each triangle is specified by three indices (i, j, k). The first two (i, j) specify a unique hexagon, while all three together specify a unique triangle within a unique hexagon. Here hexagonal rings are alternately colored white and grey.

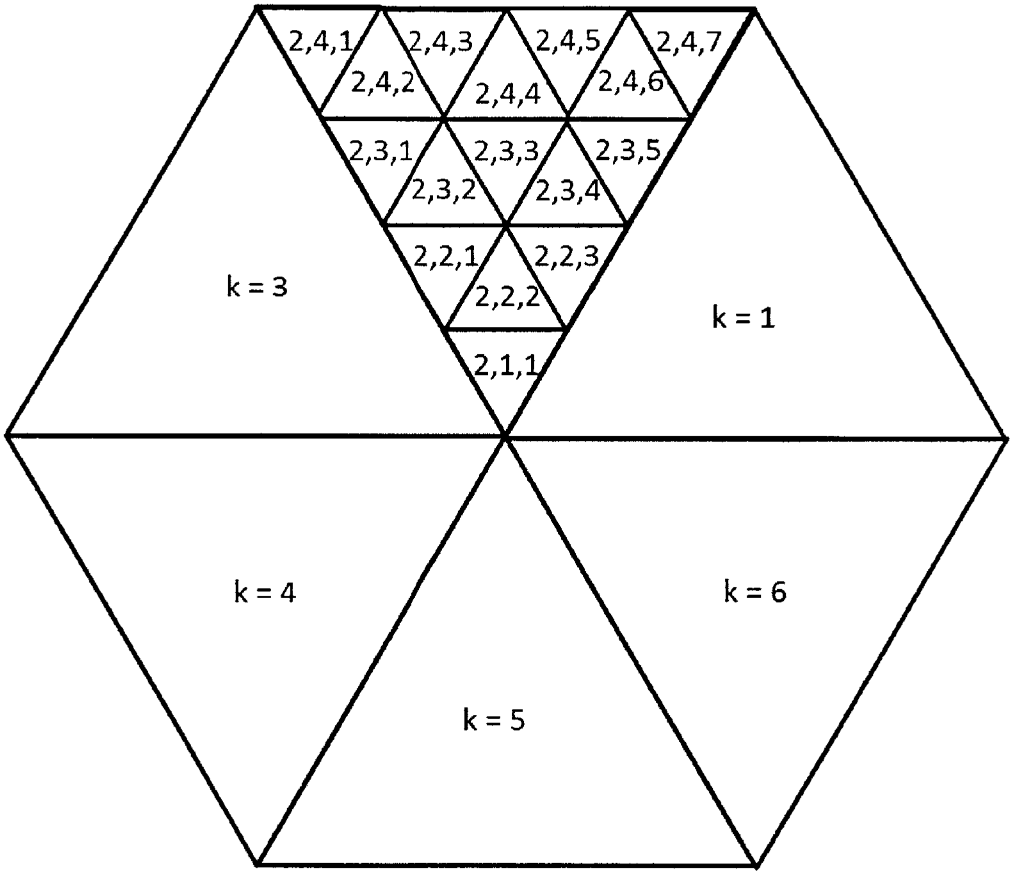

Figure A.2: The subdivision and indexing of a hyperfine triangular mesh within one hexagon. Each of the six triangles (specified by (i, j, k) indices) composing a hexagon is subdivided into a number of smaller triangles. These smaller triangles within each (i, j, k) triangle are indexed by "row" and "position", which we call u and v. The indices (k, u, v) thus specify a unique hyperfine triangle within a single hex, while the indices (i, j, k, u, v) fully specify a hyperfine triangle within an entire 2-D core.

A.3 Finite Difference Equations

We derive finite difference equations for MaPS that are precisely consistent with those in DIF3D [193]. Laying these out here allows for quick comparison with the "virtual density" finite difference equations shown in Section 7.3. First refer back to the multigroup diffusion equation that we wrote down in Equation 2.6:

$$-\nabla \cdot D_g \nabla \phi_g + \Sigma_{r,g}\phi_g = \sum_{g'=1}^{g-1} \Sigma_{s,g'\to g}\phi_{g'} + \chi_g \sum_{g'=1}^{G} \nu_{g'}\Sigma_{f,g'}\phi_{g'} \tag{A.1}$$

The left-hand side represents neutron loss in group g, while the right-hand side represents neutron gain in group g. These losses and gains are in units of neutrons per unit time per unit volume. Now consider how to discretize the leakage term, which represents the net flow of neutrons out of a given cell. Let the mesh cell (a polyhedron) have N flat surfaces indexed by n, each with area A_n. Let $J_{g,n}$ be the net current outward through surface n, and let the cell have volume V. Now we can simply express the net rate of neutrons passing outward through the cell surfaces per unit cell volume.

$$\frac{1}{V}\sum_{n=1}^{N} J_{g,n}A_n + \Sigma_{r,g}\phi_g = \sum_{g'=1}^{g-1} \Sigma_{s,g'\to g}\phi_{g'} + \chi_g \sum_{g'=1}^{G} \nu_{g'}\Sigma_{f,g'}\phi_{g'} \tag{A.2}$$

Of course, since this is finite difference, we must express $J_{g,n}$ in terms of flux differences. Consider Figure A.3, which illustrates a mesh cell and its neighbor n. Let the flux and diffusion coefficient in n be $\phi_{g,n}$ and $D_{g,n}$, and let the flux magnitude on the surface n be $\phi_{s,n}$. Also let the distances between the cell centroids and the surface be Δ and Δ_n. Note that we consider only cell configurations in which a line drawn between neighboring cell centroids is perpendicular to the surface between those two cells. This is true of both Cartesian and triangle-z geometries. We can obtain $\phi_{g,s}$ from a simple application of Fick's law:

$$\phi_{g,s} = \frac{\frac{D_g\phi_g}{\Delta} + \frac{D_{g,n}\phi_{g,n}}{\Delta_n}}{\frac{D_g}{\Delta} + \frac{D_{g,n}}{\Delta_n}} \tag{A.3}$$

Now some algebraic manipulation yields a clean expression for $J_{g,n}$, which is proportional to the difference between ϕ_g and $\phi_{g,n}$.

$$J_{g,n} = -D_g\left(\frac{\phi_{g,s} - \phi_g}{\Delta}\right) = \frac{\phi_g - \phi_{g,n}}{\frac{\Delta}{D_g} + \frac{\Delta_n}{D_{g,n}}} \tag{A.4}$$

For convenience, we define the variable $\gamma_{g,n}$:

$$\gamma_{g,n} = \frac{1}{\frac{\Delta}{D_g} + \frac{\Delta_n}{D_{g,n}}} \tag{A.5}$$

When a neighboring mesh cell position n is vacant (a vacuum boundary), we set $\phi_{g,n} = 0$ and define $\gamma_{g,n}$ as

$$\gamma_{g,n} = \frac{1}{\frac{\Delta}{D_g} + \frac{1}{\alpha/\beta}} \tag{A.6}$$

Here we typically set $\alpha/\beta = 0.46920$ for consistency with DIF3D [193]. Now the discretized multigroup diffusion equation is

$$\frac{1}{V}\sum_{n=1}^{N} A_n\gamma_{g,n}\left(\phi_g - \phi_{g,n}\right) + \Sigma_{r,g}\phi_g = \sum_{g'=1}^{g-1}\Sigma_{s,g'\to g}\phi_{g'} + \chi_g\sum_{g'=1}^{G}\nu_{g'}\Sigma_{f,g'}\phi_{g'} \tag{A.7}$$

Writing this down for ϕ_g in every mesh cell provides a system of *linear* equations. We can organize this multitude of linear equations into a single matrix equation.

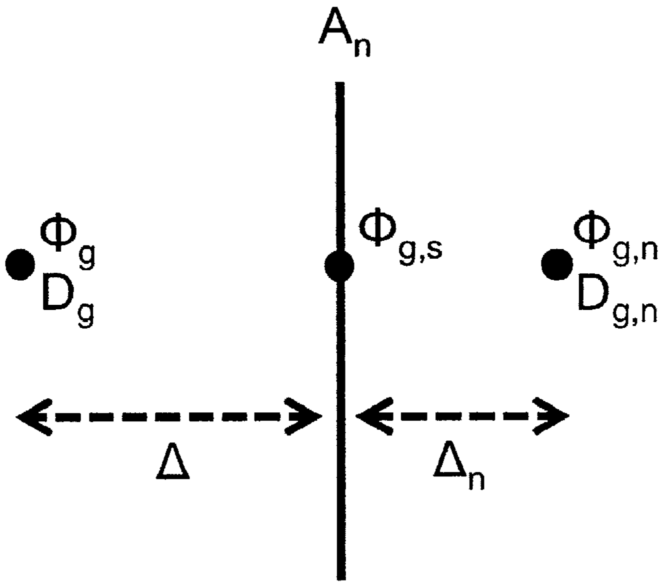

Figure A.3: Setting up Fick's Law between two adjacent finite difference mesh cells. The cell centroid is a distance Δ from the interface, and its neighbor's centroid is a distance Δ_n from the same interface.

A.4 Matrix Construction

We now construct two large, square, and sparse matrices $\hat{M}$ and $\hat{F}$ as described in Section 2.3.2. We can use these to solve the generalized eigenvalue problem.

$$\hat{M}\phi = \frac{1}{k}\hat{F}\phi \tag{A.8}$$

Note that the eigenvector ϕ above is 1-D and has a length equal to the product of the number of energy groups and the number of mesh cells. Thus, we must transform our geometry indexing conventions into a 1-D indexing system.

Let the Cartesian indices x, y, and z range from 1 to X, Y, and Z. Then the 1-D eigenvector index q_{cart} of flux ϕ_g in mesh cell (x,y,z) is

$$q_{\text{cart}} = ZYX(g-1) + ZY(x-1) + Z(y-1) + z \tag{A.9}$$

Now consider triangle-z geometry in which each hexagonal assembly is broken into 6 triangles. We can define a unique triangle with the indices (i,j,k,z). See Figure A.1 for an illustration. Let i be the assembly ring index and j be the assembly position within each ring. I is the total number of rings in the core, and $J(i) = 6(i-1)$ is the total number of assemblies in each ring (except for the first ring, which has 1 assembly). k is the triangle index within each assembly, and its maximum K is always 6. So the 1-D eigenvector index is

$$q_{\text{hex-z}} = ZK\left[\sum_{i'=1}^{I} J(i')\right](g-1) + ZK\left[\sum_{i'=1}^{i-1} J(i')\right] + ZK(j-1) + Z(k-1) + z \tag{A.10}$$

Now consider triangle-z geometry in which each of the 6 triangles within an assembly can be subdivided into an arbitrarily large number of sub-triangles. Now u and v index a sub-triangle within a triangle. See Figure A.1 for an illustration. If $6U^2$ sub-triangles form a triangle, then the index u ranges from 1 to U. The index v ranges from 1 to $V(u) = 2u - 1$. So the 1-D eigenvector index is

$$q_{\text{hex-z,fine}} = Z\left[\sum_{u'=1}^{U} V(u')\right] K \left[\sum_{i'=1}^{I} J(i')\right](g-1) \tag{A.11}$$

$$+Z\left[\sum_{u'=1}^{U} V(u')\right] K \left[\sum_{i'=1}^{i-1} J(i')\right] + Z\left[\sum_{u'=1}^{U} V(u')\right] K(j-1) \tag{A.12}$$

$$+Z\left[\sum_{u'=1}^{U} V(u')\right](k-1) + Z\left[\sum_{u'=1}^{u-1} V(u')\right] + Z(v-1) + z \tag{A.13}$$

Now that we have established a convention for transforming multivariable indexing into 1-D indexing, we can construct the $\hat{M}$ and $\hat{F}$ matrices. If q represents an eigenvector index ranging from 1 to Q, then each matrix is $Q \times Q$ and can be indexed by a (q_1,q_2) pair. Now let q_g be the 1-D index of ϕ_g in a given mesh cell, and let $q_{g,n}$ be the index of $\phi_{g,n}$ in the neighboring mesh cell n. If the matrix elements of $\hat{F}$ are b, then

$$b(q_g, q_{g'}) = \chi_g \nu_{g'} \Sigma_{f,g'} \tag{A.14}$$

This constitutes the entire $\hat{F}$ matrix. The $\hat{M}$ matrix is a bit more complex, as it has three types of terms. First, the scattering terms are

$$a(q_g, q_{g'}) = -\Sigma_{s,g' \to g} \tag{A.15}$$

Now the main diagonal elements in $\hat{M}$ are

$$a(q_g, q_g) = \Sigma_{r,g} + \frac{1}{V} \sum_{n=1}^{N} A_n \gamma_{g,n} \tag{A.16}$$

The off-diagonal elements in $\hat{M}$ are

$$a(q_g, q_{g,n}) = -\frac{A_n \gamma_n}{V} \tag{A.17}$$

In R-D Cartesian problems, $\hat{M}$ will have a main diagonal as well as $2R$ off-diagonals. See Figure A.4, which displays non-zero elements of $\hat{F}$ for the FFTF benchmark problem in Appendix G. Figure A.5 zooms in on the non-zero elements of $\hat{M}$ for a simple 3-D Cartesian problem.

In triangular or triangular-z problems, $\hat{M}$ will have a main diagonal but no well-defined off-diagonals. The off-diagonal terms will form pretty patterns instead of straight lines. Figures A.6 and A.7 show non-zero elements of $\hat{M}$ for the FFTF benchmark problem in Appendix G. Figure A.8 zooms in on the non-zero off-diagonal leakage terms in 2-D hyperfine triangular geometry.

Note that we have chosen to arrange matrix elements by energy group first and spatial region second. Although it is well-known that arranging elements by spatial region first increases efficiency, we have chosen not to pursue that route.

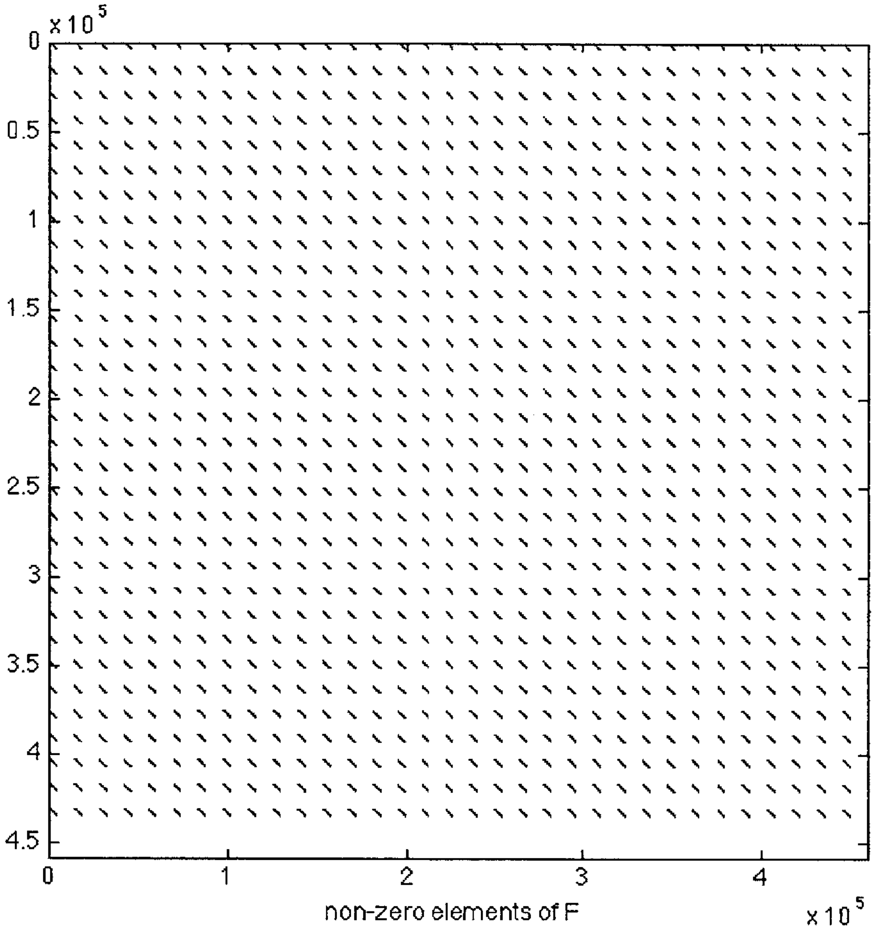

Figure A.4: Non-zero elements in the fission matrix $\hat{F}$ for the FFTF benchmark. If every mesh cell in the core contained fuel, each diagonal would be completely filled, but the gaps represent spatial regions without fuel - shields, reflectors, and control rods.

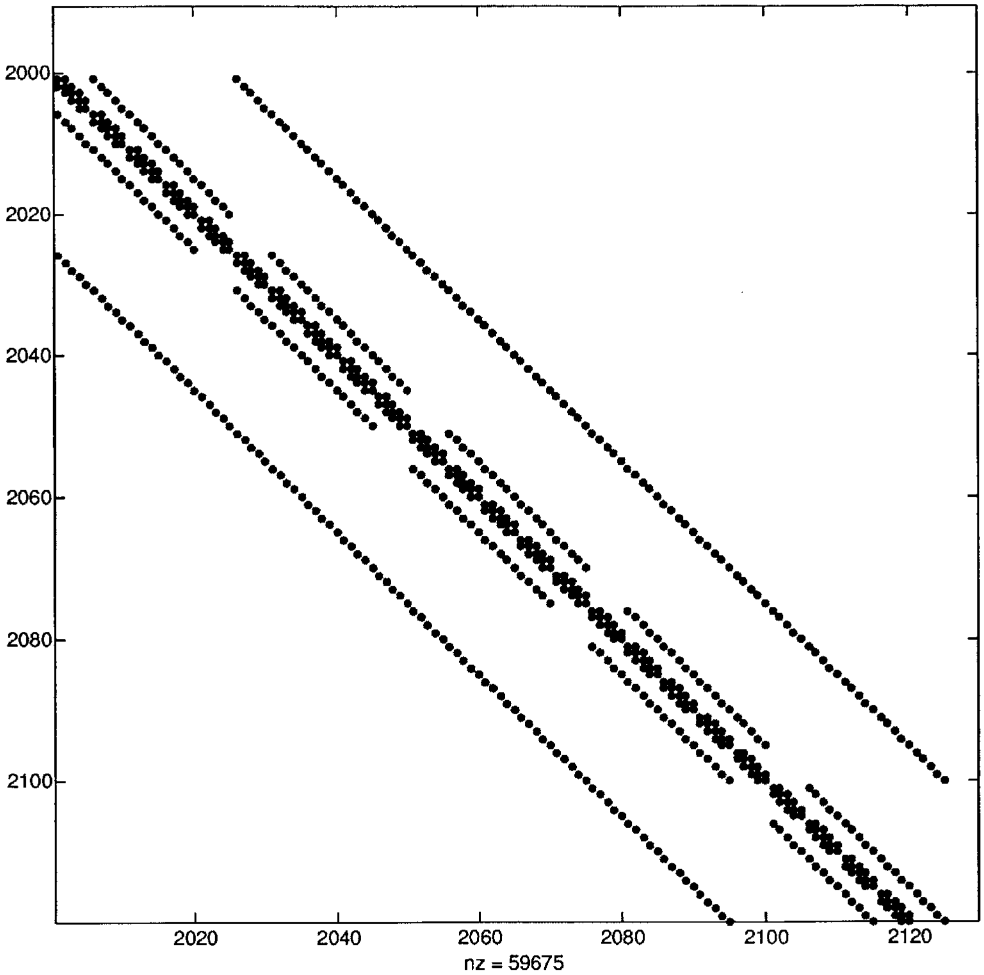

Figure A.5: Non-zero elements in the $\hat{M}$ matrix for a small 3-D Cartesian problem. The seven central diagonals represent absorption in each mesh cell and leakage into its six neighbors.

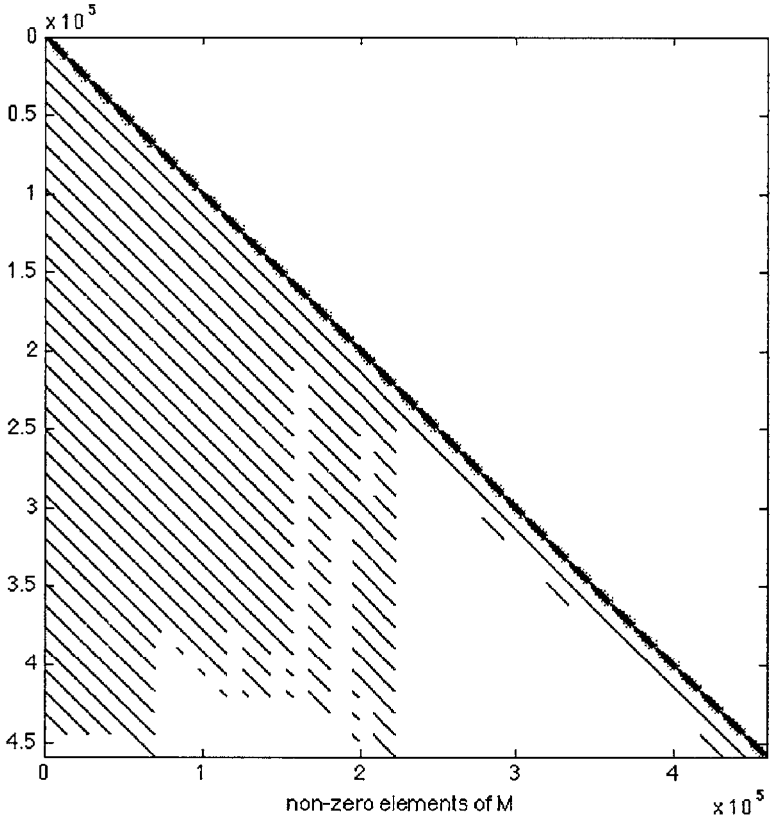

Figure A.6: Non-zero elements in the $\hat{M}$ matrix for the FFTF benchmark in triangle-z geometry. Obviously, we do not consider up-scattering.

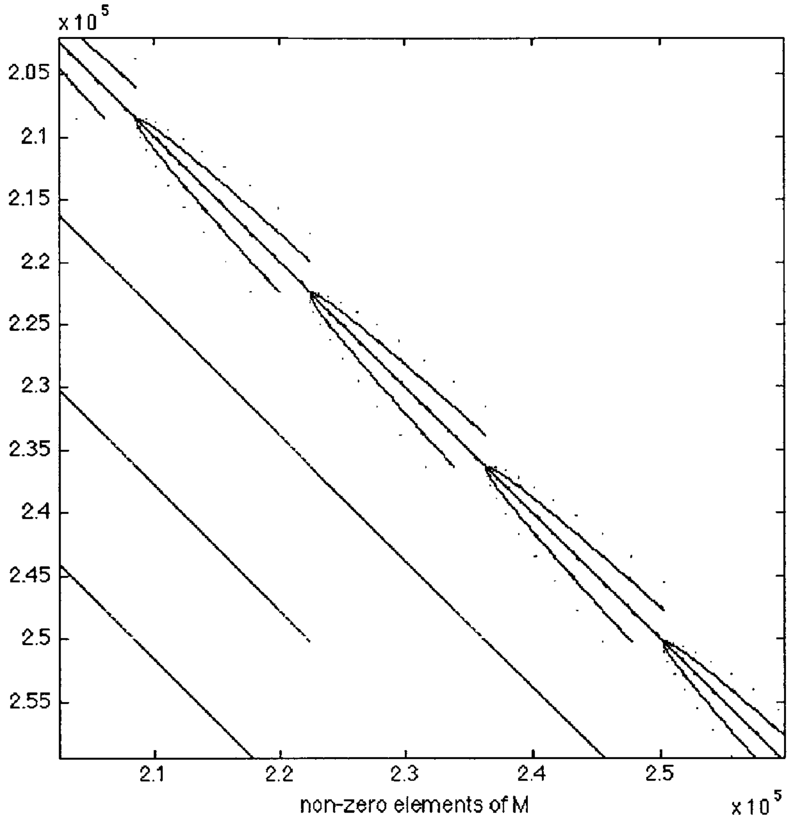

Figure A.7: Non-zero elements near the central diagonal in the $\hat{M}$ matrix for the FFTF benchmark in triangle-z geometry. This is a zoomed-in section of Figure A.6. The central diagonal represents absorption in each mesh cell. The two straight diagonals immediately on either side (not visually distinguishable here) represent leakage into the cell's two axial neighbors. The terms that form arrow-like projections off the central diagonals represent leakage into the cell's three radial/lateral neighbors.

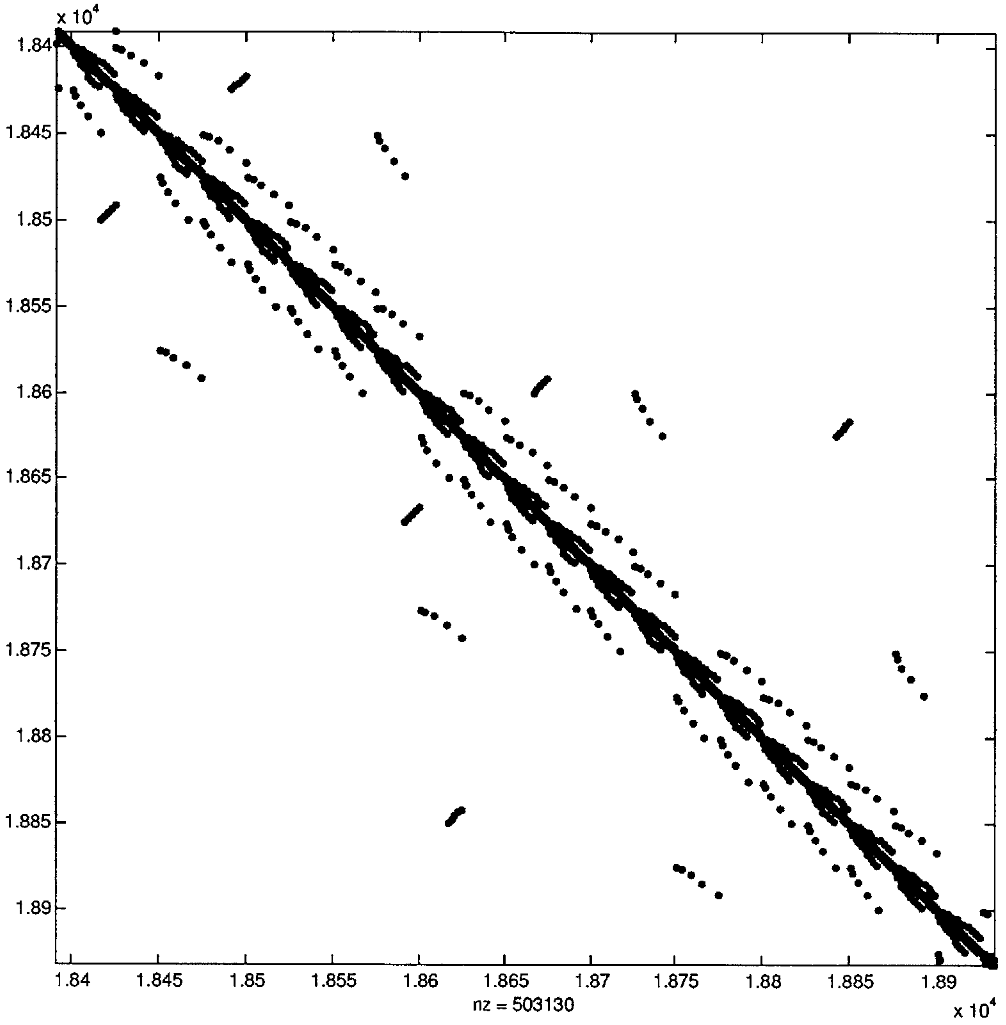

Figure A.8: Non-zero elements near the central diagonal in the $\hat{M}$ matrix for a 2-D hyperfine triangle core. This is zoomed-in to make the off-diagonal leakage terms visible. In this case, $U = 5$ so that each hexagon is subdivided into 150 triangles. See Figure A.2 for an illustration. The leakage terms form various projections off the central diagonal.

A.5 Adjoint Matrix Construction

Many mathematical texts define an adjoint operator as the "complex conjugate of a transpose". However, one can *not* obtain the adjoint flux by simply transposing the entire $\hat{F}$ and $\hat{M}$ matrices. In reality, we transpose only the energy components of $\hat{F}$ and $\hat{M}$. The spatial components are unchanged. This makes sense if we remember that the one-group diffusion equation is self-adjoint. The real and adjoint fluxes (and operators) are identical without energy dependence.

Thus, in order to convert $\hat{F}$ and $\hat{M}$ into $\hat{F}^\dagger$ and $\hat{M}^\dagger$, we need only transpose the energy group indices of each non-zero matrix element. All scattering and fission terms (in each mesh cell) representing neutron transfer from group g to group g' must be moved so that they represent "backward" neutron transfer from group g' to g. The absorption and leakage terms within $\hat{M}$ are left untouched.

A.6 Benchmarking

Before anyone will believe any results from MaPS, we must benchmark it. We accomplish this with both finite difference diffusion, with which we expect very precise agreement, and continuous energy Monte Carlo, with which we expect approximate agreement.

A.6.1 Multigroup Finite Difference Diffusion

We compare MaPS to two finite difference diffusion benchmarks. First, we compare MaPS to LABAN-PEL, a 2-D multigroup diffusion code from Oak Ridge National Laboratory [197]. We verify that MaPS and LABAN-PEL produce *exactly* the same k_{eff} eigenvalue for the 2-group, 2-D Cartesian ZION-1 benchmark problem [198]. Cross-sections are given in the ZION-1 description.

Second, we compare MaPS to DIF3D, a hexagonal-z multigroup diffusion code from Argonne National Laboratory [193]. We build the full-core Jōyō benchmark (see the description and core map in Appendix H), which has 331 hexagonal assemblies, 6 triangular mesh cells per assembly, 66 axial zones, and 33 energy groups. We use identical MC**2 cross-sections [187]. In this problem, k_{eff} values in MaPS and DIF3D are 0.974266044 and 0.974222662, respectively. This is a discrepancy of less than 5 pcm.

A.6.2 Continuous Energy Monte Carlo

Multigroup diffusion and continuous energy transport will never agree precisely; we expect both energy discretization and anisotropic scattering/leakage effects to introduce errors in the few hundred pcm range.

We have benchmarked MaPS with the Monte Carlo N-Particle Transport Code (MCNP) [202]. Typically, MaPS agrees with continuous energy MCNP to about 100 pcm for 1-D problems and to about 500 pcm for high-leakage 2-D problems. In high-leakage 3-D problems with fuel adjacent to vacuum boundaries, errors can approach and exceed 1000 pcm due to anisotropic leakage effects.

A.7 Speed and Efficiency

MaPS is intended to be convenient, not necessarily fast. It is not an optimized production code, so we sacrifice some efficiency in favor of usability. All MaPS calculations are performed on a MacBook Pro laptop with a 2.9 GHz processor and 8 GB of RAM. MaPS is not parallelizable, so all calculations are without MPI.

MaPS can solve a Q x Q generalized eigenvalue problem in about five minutes for Q = 1,835,064, where Q is the product of the number of spatial regions times the number of energy groups. This corresponds to a Jōyō benchmark with 331 assemblies, 6 triangles per assembly, 28 axial zones, and 33 energy groups. An FFTF benchmark of the same size solves even more quickly due to fewer fuel assemblies (the depleted uranium in Jōyō greatly enlarges the fission source). See Appendices J and K for descriptions and core maps of FFTF and Jōyō. MaPS can solve a triangular-z problem with $Q \approx$ 125,000 in less than 25 seconds.

MaPS tends to slow down markedly as the non-zero matrix element storage approaches the system memory limit. Fortunately, both FFTF and Jōyō benchmarks are not large enough to induce this slowdown, which typically occurs for Q in the range of several million for triangle-z problems.

Overall, we find MaPS to be extremely expeditious for the trial-and-error process necessary to develop both perturbation theory and flux reconstruction methods.

B Pin Power and Multigroup Flux Reconstruction Methods

Here we detail the flux reconstruction methods employed within VirDenT as described in Chapter 11.

In fast reactors, the neutron mean free path is long enough that a grid structure of homogenized assemblies is sufficiently accurate to perform scoping analysis of steady-state and burnup scenarios. However, when designing an actual reactor, it is always necessary to reconstruct pin-level power and temperature distributions from the homogenized values.

We closely follow the reputable multigroup flux and pin power reconstruction methods developed and published by Yang at Argonne National Laboratory [113]. For each energy group in each axial block in each hex assembly, we construct a continuous (x, y) flux distribution to preserve the volume-averaged flux, six surface-averaged fluxes, six surface-averaged net currents, and six corner-point fluxes. In the axial direction (z), which is separable, we construct a continuous 1-D polynomial to preserve the volume-averaged flux, two surface-averaged fluxes, and two surface-averaged net currents. We then compute pin linear powers in each axial block of each assembly by summing fission reaction rates over all energy groups. Additionally, we automatically generate pin configurations, and we compute assembly-averaged pin power peaking factors for input to COBRA [191], a subchannel thermal hydraulics code.

We integrate this pin power reconstruction code into VirDenT and the Advanced Reactor Modeling Interface (ARMI) [211] as a Python module, which includes both the reconstruction methods and scripts to read the DIF3D output files.

In order to verify the accuracy of these reconstruction methods, we analyze two reference cases with the MATLAB-PETSc-SLEPc (MaPS) finite difference diffusion code (described in Appendix A). These cases are the Fast Flux Test Facility (FFTF) and a small homogenous core with FFTF fuel (MOX) [146]. For each of these cases, we compare two solutions: (1) an exact hyperfine mesh solution with hundreds of triangular mesh cells per hex and (2) a solution reconstructed from coarse whole-hex volume and surface quantities, which we obtain by integrating the exact hyperfine solution quantities over the hex volume and hex surfaces.

This Appendix B details (1) the reconstruction methods employed, (2) crucial details of the integration of these methods into ARMI, and (3) results from the two reference cases.

B.1 Multigroup Flux Reconstruction Methodology

We assume that the axial (z) and hexagonal (x, y) flux distributions are separable, so we can reconstruct one at a time. We implement the pin power reconstruction methodology developed by Yang at Argonne National Laboratory [113,114,115,116].

B.1.1 The Hexagonal Flux Distribution

The aim of this task is to construct a continuous function in (x,y) to represent the flux within the boundaries of each hexagonal assembly. We do this by representing the flux as a weighted sum of monomials (a monomial is any single term $x^m y^n$).

$$\phi(x,y) = \sum_{m,n} C_{m,n} x^m y^n \tag{B.1}$$

Determining the monomial weights (the constants $C_{m,n}$ in front of each $x^m y^n$ term), we must introduce a number of constraints equal to the number of monomials. One constraint, which ARMI already employs, is the volume-averaged flux in the hexagonal cell. However, we can introduce more constraints by requiring the monomial sum (the full x-y polynomial) to preserve six surface-averaged net currents, six surface-averaged fluxes, and six corner-point fluxes. That brings us to 19 constraints, so our x-y polynomial can have 19 distinct monomial terms. It is also possible to require the flux to preserve the three planar moments, which brings our total to 22 constraints or monomials.

We require each energy group of the flux to satisfy all these constraints. Thus, we must solve a $Q \times Q$ matrix for each energy group in each hexagonal ARMI block, where Q is usually 19 or 22.

$$\begin{bmatrix} L_1(m_1,n_1) & L_1(m_2,n_2) & L_1(m_3,n_3) & \cdots & L_1(m_{Q-1},n_{Q-1}) & L_1(m_Q,n_Q) \\ L_2(m_1,n_1) & L_2(m_2,n_2) & L_2(m_3,n_3) & \cdots & L_2(m_{Q-1},n_{Q-1}) & L_2(m_Q,n_Q) \\ L_2(m_1,n_1) & L_2(m_2,n_2) & L_2(m_3,n_3) & \cdots & L_3(m_{Q-1},n_{Q-1}) & L_3(m_Q,n_Q) \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ L_{Q-1}(m_1,n_1) & L_{Q-1}(m_2,n_2) & L_{Q-1}(m_3,n_3) & \cdots & \ddots & \vdots \\ L_Q(m_1,n_1) & L_Q(m_2,n_2) & L_Q(m_3,n_3) & \cdots & \cdots & L_Q(m_Q,n_Q) \end{bmatrix} \begin{bmatrix} C_{m_1,n_1} \\ C_{m_2,n_2} \\ C_{m_3,n_3} \\ \vdots \\ \vdots \\ C_{m_Q,n_Q} \end{bmatrix} = \begin{bmatrix} \Phi_1 \\ \Phi_2 \\ \Phi_3 \\ \vdots \\ \vdots \\ \Phi_Q \end{bmatrix} \tag{B.2}$$

Here Φ_q represents the q^{th} constraint (a volume-averaged flux, one of six surface-averaged currents, etc.). C_{m_q,n_q} represents the constant associated with the q^{th} monomial $x^{m_q} y^{n_q}$. There are a total of Q constraints and Q monomials. The $Q \times Q$ matrix elements $L_q(m_q,n_q)$ represent the q^{th} monomial $x^{m_q} y^{n_q}$ evaluated to meet the q^{th} constraint Φ_q.

For example, if $q = 1$ corresponds to the volume-averaged flux constraint, then the following two statements are true:

1. Φ_1 is equal to the volume-averaged flux in this hexagonal cell.

2. $L_1(m_q,n_q)$ is equal to the monomial $x^{m_q} y^{n_q}$ averaged over the hexagon area.

Figure B.1 below shows the x-y coordinate system as well as the surface and corner numbering system used in this reconstruction. For a given hex assembly, the surfaces and corners are specified by the index k, which ranges from 0 to 5. Figure B.1 shows surfaces and corners labeled as S_{k+1} and C_{k+1}. Note that we have drawn the hexagon resting on its side (as it does within the ARMI coordinate system), even though the x-y coordinates shown are oblique. This is because the L_q functions derived (and listed below) assume the oblique coordinate. This discrepancy is acceptable, as long as we rotate the x-y coordinate system by $\pi/6$ after reconstruction and before exportation to other ARMI modules.

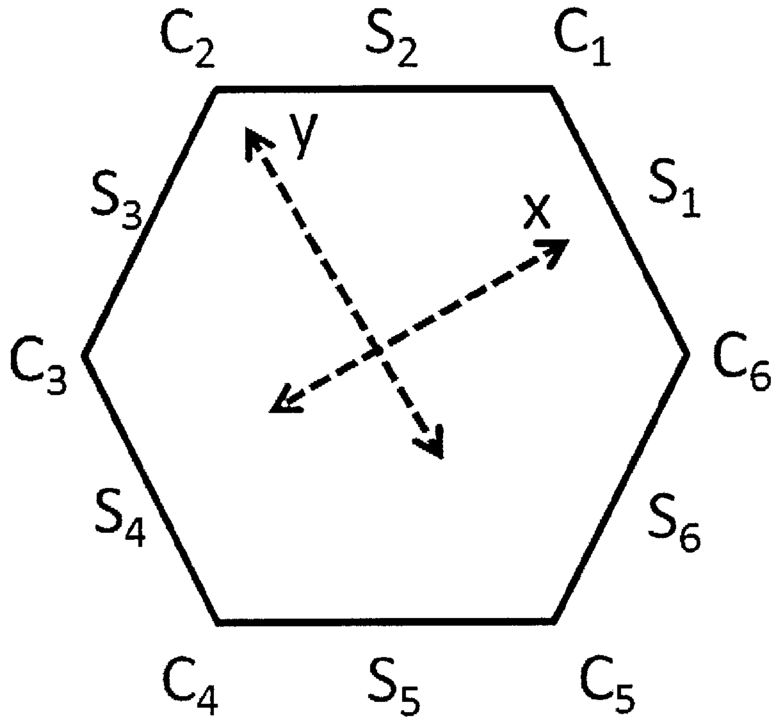

Figure B.1: The orientation and numbering of hex surfaces and corners that we choose for a continuous flux reconstruction problem.

For convenience, we derive expressions for each of the Q values of L_q as functions of m and n. We do this for a regular hexagon of flat-to-flat pitch 2 cm. The equations below show the derived $L_q(m, n)$ functions for $Q = 22$. These correspond to one volume-averaged flux, six surface-averaged fluxes, six surface-averaged net currents, six corner-point fluxes, and three planar flux moments.

$x^m y^n$ averaged over the hex area

$$L_1(m,n) = \frac{[1+(-1)^m][1+(-1)^n]}{4(n+1)} \left(\frac{2}{\sqrt{3}}\right)^{n+2} Z(m,n+1) \tag{B.3}$$

$x^m y^n$ averaged over the hex surface S_1

$$L_2(m,n) = \frac{1+(-1)^n}{2(n+1)} \left(\frac{1}{\sqrt{3}}\right)^{n} \tag{B.4}$$

$x^m y^n$ averaged over the hex surface S_2

$$L_3(m,n) = \left(\frac{2}{\sqrt{3}}\right)^{n} Z(m,n) \tag{B.5}$$

$x^m y^n$ averaged over the hex surface S_3

$$L_4(m,n) = (-1)^m L_3(m,n) \tag{B.6}$$

$x^m y^n$ averaged over the hex surface S_4

$$L_5(m,n) = (-1)^m L_2(m,n) \tag{B.7}$$

$x^m y^n$ averaged over the hex surface S_5

$$L_6(m,n) = (-1)^{m+n} L_3(m,n) \tag{B.8}$$

$x^m y^n$ averaged over the hex surface S_6

$$L_7(m,n) = (-1)^n L_3(m,n) \tag{B.9}$$

$x^m y^n$ evaluated at the hex corner C_1

$$L_8(m,n) = \left(\frac{1}{\sqrt{3}}\right)^{n} \tag{B.10}$$

$x^m y^n$ evaluated at the hex corner C_2

$$L_9(m,n) = \left(\frac{2}{\sqrt{3}}\right)^{n} \delta_{m0} \tag{B.11}$$

$x^m y^n$ evaluated at the hex corner C_3

$$L_{10}(m,n) = (-1)^m L_8(m,n) \tag{B.12}$$

$x^m y^n$ evaluated at the hex corner C_4

$$L_{11}(m,n) = (-1)^{m+n} L_8(m,n) \tag{B.13}$$

$x^m y^n$ evaluated at the hex corner C_5

$$L_{12}(m,n) = (-1)^n L_9(m,n) \tag{B.14}$$

$x^m y^n$ evaluated at the hex corner C_6

$$L_{13}(m,n) = (-1)^n L_8(m,n) \tag{B.15}$$

The normal derivative of $x^m y^n$ multiplied by D and averaged over surface S_1

$$L_{14}(m,n) = -mDL_2(m,n) \tag{B.16}$$

The normal derivative of $x^m y^n$ multiplied by D and averaged over surface S_2

$$L_{15}(m,n) = -\frac{mD}{2} L_3(m-1,n) - \frac{\sqrt{3}nD}{2} L_3(m,n-1) \tag{B.17}$$

The normal derivative of $x^m y^n$ multiplied by D and averaged over surface S_3

$$L_{16}(m,n) = (-1)^m L_{15}(m,n) \tag{B.18}$$

The normal derivative of $x^m y^n$ multiplied by D and averaged over surface S_4

$$L_{17}(m,n) = (-1)^m L_{14}(m,n) \tag{B.19}$$

The normal derivative of $x^m y^n$ multiplied by D and averaged over surface S_5

$$L_{18}(m,n) = (-1)^{m+n} L_1 5(m,n) \tag{B.20}$$

The normal derivative of $x^m y^n$ multiplied by D and averaged over surface S_6

$$L_{19}(m,n) = (-1)^n L_{15}(m,n) \tag{B.21}$$

The planar moment of $x^m y^n$ in the direction x

$$L_{20}(m,n) = \frac{[1+(-1)^{m+1}][1+(-1)^n]}{4(n+1)} \left(\frac{2}{\sqrt{3}}\right)^{n+2} Z(m,n+1) \tag{B.22}$$

The planar moment of $x^m y^n$ in the direction $u = y + x/\sqrt{3}$

$$L_{21}(m,n) = \frac{[1+(-1)^m][1+(-1)^{n+1}]}{4(n+1)} \left(\frac{2}{\sqrt{3}}\right)^{n+2} Z(m,n+1) - \frac{(-1)^m - (-1)^n}{(n+1)(n+m+2)} \left(\frac{1}{\sqrt{3}}\right)^{n+2} \tag{B.23}$$

The planar moment of $x^m y^n$ in the direction $v = y - x/\sqrt{3}$

$$L_{22}(m,n) = \frac{[1+(-1)^m][1+(-1)^{n+1}]}{4(n+1)} \left(\frac{2}{\sqrt{3}}\right)^{n+2} Z(m,n+1) - \frac{1+(-1)^{m+n+1}}{(n+1)(n+m+2)} \left(\frac{1}{\sqrt{3}}\right)^{n+2} \tag{B.24}$$

$Z(m,n)$ in all the previous functions

$$Z(m,n) = \sum_{i=0}^{n} \frac{n!}{(m+i+1)(n-i)!i!} \left(\frac{-1}{2}\right)^{i} \tag{B.25}$$

Given a set of Q constraints, choosing the most accurate set of Q monomials (a set of Q m,n pairs) can be difficult. There are three hard requirements:

1. The number of monomials equals the number of constraints.
2. The monomial $(m,n) = (0,0)$ must be included as a constant flux term.
3. At least one monomial must have an order $K = m + n$ such that the total possible number of monomials of order K is greater than or equal to the number of constraints.

Regarding requirement (c), the total possible number of monomials of order $K = m + n$ is equal to $(K+1)(K+2)/2$. So if we choose 13 constraints, at least one monomial must have degree $K = m + n = 4$. There are $(K+1)(K+2)/2 = 15$ total monomials of degree 4, while there are only 10 total monomials of degree 3.

By default, the ARMI flux reconstruction module includes $Q = 19$ constraints (all except the three planar flux moments). The default set of monomials is:

1. All monomials with $m + n \leq 4$, including $(m,n) = (0,0)$. These constitute 15 monomials.
2. The remaining 4 monomials are $(m,n) = (5,0)$, $(4,1)$, $(3,2)$, $(4,2)$.

One can very easily alter this default set of $Q = 19$ monomials in the flux reconstruction module. Some sets of monomials can be more or less accurate than other sets.

B.1.2 The Axial Flux Distribution

Reconstructing the axial multigroup flux is much simpler. We choose our default constraints to be the volume-average flux, the two surface-averaged fluxes, and the two surface-averaged net currents. Thus, we construct a simple 4th order polynomial in z with 5 arbitrary constants:

$$\phi(z) = Az^4 + Bz^3 + Cz^2 + Dz + E \tag{B.26}$$

The matrix elements are simple to derive, and there is neither ambiguity nor flexibility in the choice of polynomial terms.

In order to obtain the full flux distribution in a given ARMI block, we simply multiply the hexagonal and axial flux distributions and then divide by the volume-averaged flux (so that our final units are that of flux, not $(\text{flux})^2$).

B.2 Obtaining Nodal Quantities from DIF3D

In order to use volume-averaged fluxes, surface-averaged net currents, surface-averaged fluxes, corner-point fluxes, and planar flux moments as constraints, we must first know them! ARMI already uses the volume-averaged fluxes in each DIF3D diffusion mesh cell. However, in order to obtain the other 21 quantities, we must read the NHFLUX binary file output by the DIF3D nodal option. NHFLUX contains three important sets of information for the flux reconstruction module:

1. Multigroup surface-averaged partial currents for each nodal mesh surface.

2. Five multigroup planar flux moments for each nodal mesh cell.

3. A hexagonal indexing map to obtain the proper ordering of assemblies in NHFLUX.

Using these three sets of information, we can compute all 19 constraints for each energy group in each ARMI block.

B.2.1 Hex Surface-Averaged Fluxes and Net Currents

The outgoing net current averaged over hex surface S with area A is

$$J_{\text{net}} = \frac{D}{A} \int ds\, \hat{n} \cdot \nabla\phi\left(x(s), y(s)\right) \tag{B.27}$$

where s is a parameter that represents position on the surface, and $\hat{n}$ is the unit vector normal to the surface.

The DIF3D binary file NHFLUX contains six outgoing partial currents for each hex assembly in each (x, y) plane of nodal mesh cells. In order to compute the net outgoing current on each surface, we must obtain and subtract the incoming partial current (which is stored as an outgoing partial current of a neighboring hex assembly).

$$J_{\text{net}} = J^{\dagger} - J^{-} \tag{B.28}$$

Furthermore, NHFLUX contains one incoming partial current on each external boundary of the reactor (usually vacuum). Due to the diffusion flux extrapolation boundary condition, these partial currents are not zero. Unfortunately, since they are not outgoing partial currents attached to any particular hex assembly (as all of the internal surface partial currents are), these external surface incoming partial currents are indexed separately from hex assemblies. They are lumped together at the end of the hex surface current data structure.

Once we have correctly paired partial surface currents of neighboring assemblies, we can easily compute the total surface-averaged flux like this:

$$\phi = 2(J^{\dagger} + J^{-}) \tag{B.29}$$

This expression is an approximation that assumes an isotropic flux distribution. Since this is diffusion, not transport, we are already ignoring angular dependence. The factor of 2 arises from the integral of $\cos\theta$ from $-\pi/2$ to $\pi/2$.

B.2.2 Hex Corner-Point Fluxes

Three cells and three surfaces join at each hex corner. Thus, we can interpolate between three volume-averaged fluxes ϕ_v and three-surface averaged fluxes ϕ_s to estimate a corner-point flux. We use a reputable interpolation formula:

$$\phi_C = \frac{\sum_n D_n \left[17(\phi_{s,n} + \phi_{s,n-1}) - 8\phi_{v,n}\right]}{26 \sum_n D_n} \tag{B.30}$$

Here both summations are for $n = 1,2,3$. D_n is the diffusion constant in the n^{th} hex cell. Figure B.2 illustrates the relationship between these quantities. Near vacuum boundaries, we assume that any "ghost" hex cell (occupied by vacuum) has zero flux. Any "ghost" hex surface (bounding two "ghost" vacuum hex cells) has zero flux. However, hex surfaces that separate an existing hex from a vacuum hex do have surface fluxes due to the diffusion flux extrapolation at vacuum boundaries.

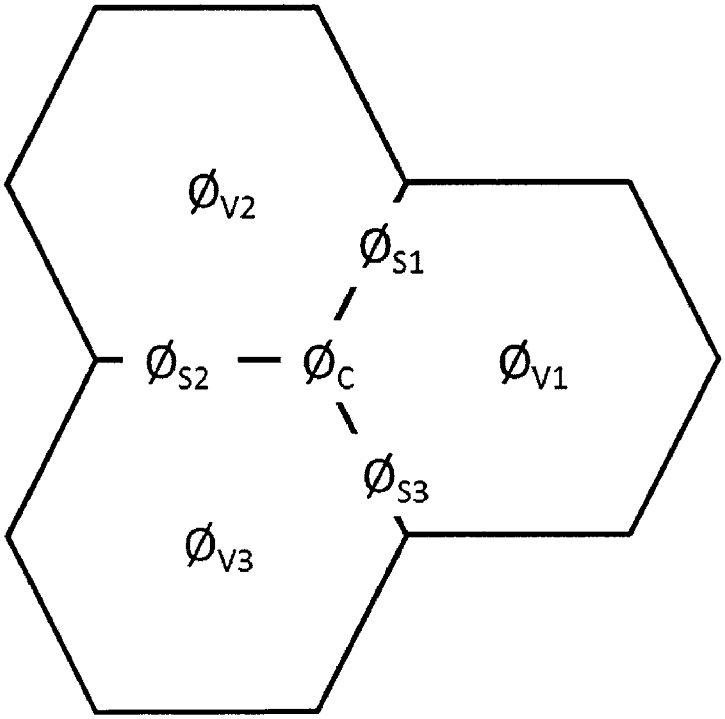

Figure B.2: Each corner-point flux can be estimated using an interpolation between the three adjoining surface-averaged fluxes and the three adjoining volume-averaged fluxes.

B.2.3 Axial Surface-Averaged Fluxes and Net Currents

The process for computing quantities on axial surfaces is similar to that for computing those same quantities on hexagonal surfaces, except that the storage convention is different. NHFLUX stores axial partial currents by mesh surface, not by mesh node. Each axial surface has two partial currents: an upward current and a downward current.

B.2.4 Hex Planar Flux Moments

Obtaining planar flux moments is relatively straightforward. There are five per hexagonal mesh cell, three in the x-y plane and two in the axial directions. A planar flux moment is the difference between the volume-averaged flux in one half of a region and the volume-averaged flux in the other half of that same region. In the case of a hexagon, there are three flux moments corresponding to three ways to divide the hex in half (by drawing a line between opposite corners). For example, the hexagonal flux moment in the x-direction is

$$\frac{1}{A}\int_A \mathrm{sgn}(x)\phi(x,y)dxdy \tag{B.31}$$

Here A is the full hexagonal area. We use the coordinate system shown in Figure B.1, so that $x = 0$ is a line between the top and bottom hex corners. The function sgn(x) is equal to -1 for $x < 0$ and 1 for $x > 0$. Essentially, we are subtracting the volume-averaged flux in one half-hex ($x < 0$) from that in the other half-hex ($x > 0$). There are two other hexagonal flux moments, one in the direction $u = y + x/\sqrt{3}$ and the other in the direction $v = y - x/\sqrt{v}3$.

B.3 Pin Power Peaking

The flux reconstruction module also constructs whole-assembly pin power peaking factors for the subchannel thermal hydraulic analysis code COBRA [191,209]. For each pin in each assembly, we sum the whole-pin power (through all axial blocks) to compute pin power peaking factors. We store this data for later export to COBRA. See Figure B.3 for a PyPinPlot display of intra-assembly pin power peaking factors in a small homogenous core.

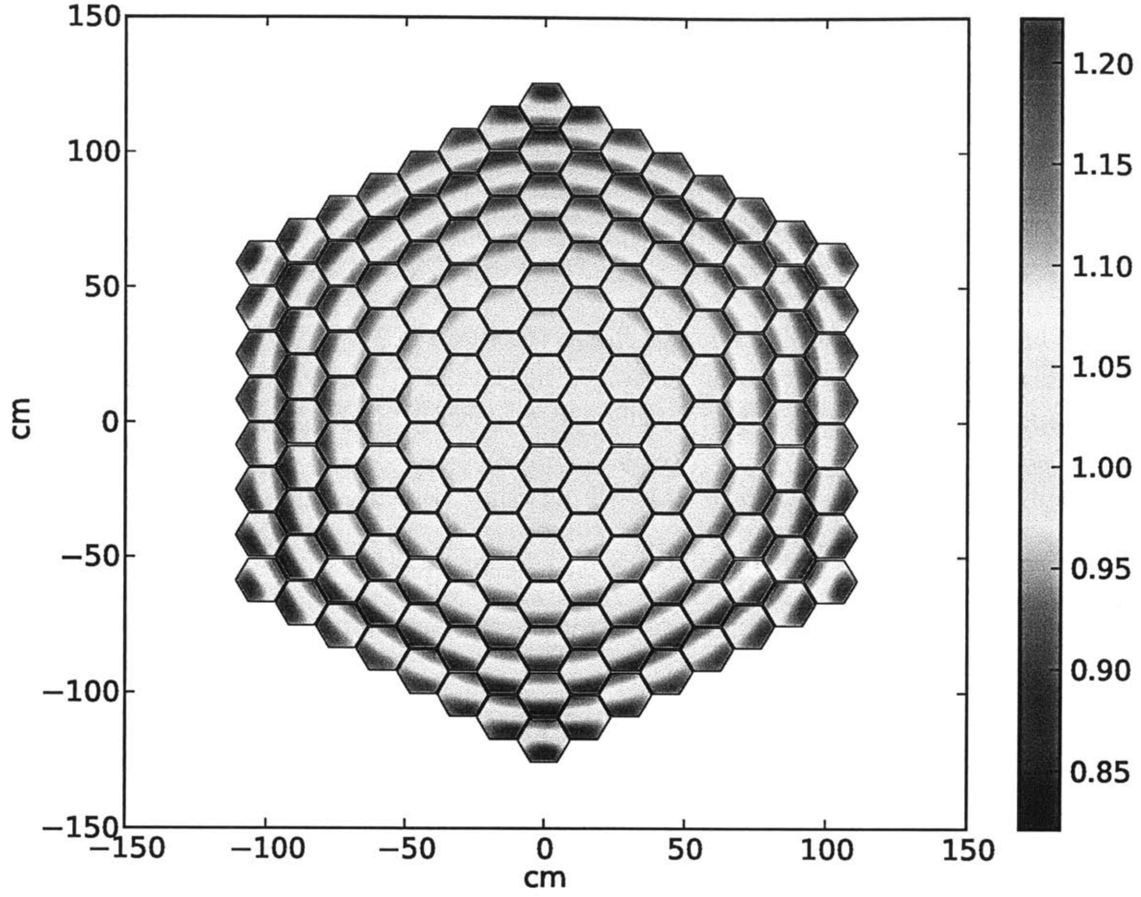

Figure B.3: An example of the ARMI pin peaking factor plotting tool for a small homogenous test core with 8 full hex rings. This demonstrates the plotting capabilities for full core models.

B.4 Reference Cases

In order to verify the accuracy of these pin power reconstruction methods, we construct three reference (or benchmark) cases. For each reference case, we validate the methods using these steps:

1. We run these reference cores with hyperfine geometry (150 or 294 triangles per hex) in MaPS to obtain fine (and highly accurate) multigroup flux and power solutions.

2. We integrate the hyperfine triangle fluxes to obtain hex-averaged fluxes, hex surface-averaged fluxes, and hex surface-averaged net currents. The surface quantities involve computing the simple sums and differences between neighboring hyperfine triangles in neighboring hexes (hyperfine triangles on opposite sides of hex surfaces). Then we simply average those quantities over all hyperfine triangles along a given hex surface.

3. We compute the hex corner-point fluxes in the usual way.

4. We perform the continuous multigroup flux reconstruction using our computed volume, surface, and corner quantities.

5. We construct a fictitious set of pin (x, y) coordinates so that pin centers correspond to hyperfine triangle centers.

6. We compute the power of each of those pins.

7. We compare the reconstructed pin powers to the directly-computed hyperfine triangle powers.

This process isolates the source of error in our pin power reconstruction. Comparing a hyperfine mesh solution to powers reconstructed from a course mesh solution would benchmark nothing, because the errors due to the difference in meshes would dwarf the errors due to the actual reconstruction methods.

We complete the process outlined above for two reference cases: the FFTF core and a homogenous core with FFTF (inner core) MOX fuel.

B.4.1 The Fast Flux Test Facility (FFTF)

We construct the FFTF benchmark in MaPS as shown and described in Appendix G. This is a sodium-cooled fast reactor with mixed oxide (MOX) fuel. We neglect some nuclides that occur only in trace amounts, and we fill out the outermost hex assembly ring with 18 extra "outer radial shield" assemblies. We take a 2-D core slice at the axial midplane, so our model is supercritical with $k_{\text{eff}} \approx 1.1$. Each hex contains 294 triangles.

Figures B.4 and B.5 show FFTF power density distributions derived from the exact hyperfine mesh solution and the flux reconstruction, respectively. These two distributions are virtually indistinguishable.

Figure B.6 shows the error magnitudes between these two distributions, which are 0.33% on average. However, the error can spike to a few percent (even has high as 5%) near the hex corner-points at the edge of the fueled region. This is due to error in the corner-point flux interpolation, which is less accurate adjacent to abrupt changes in fuel composition and flux spectrum. The corner-points in the internal fueled region do not produce such errors.

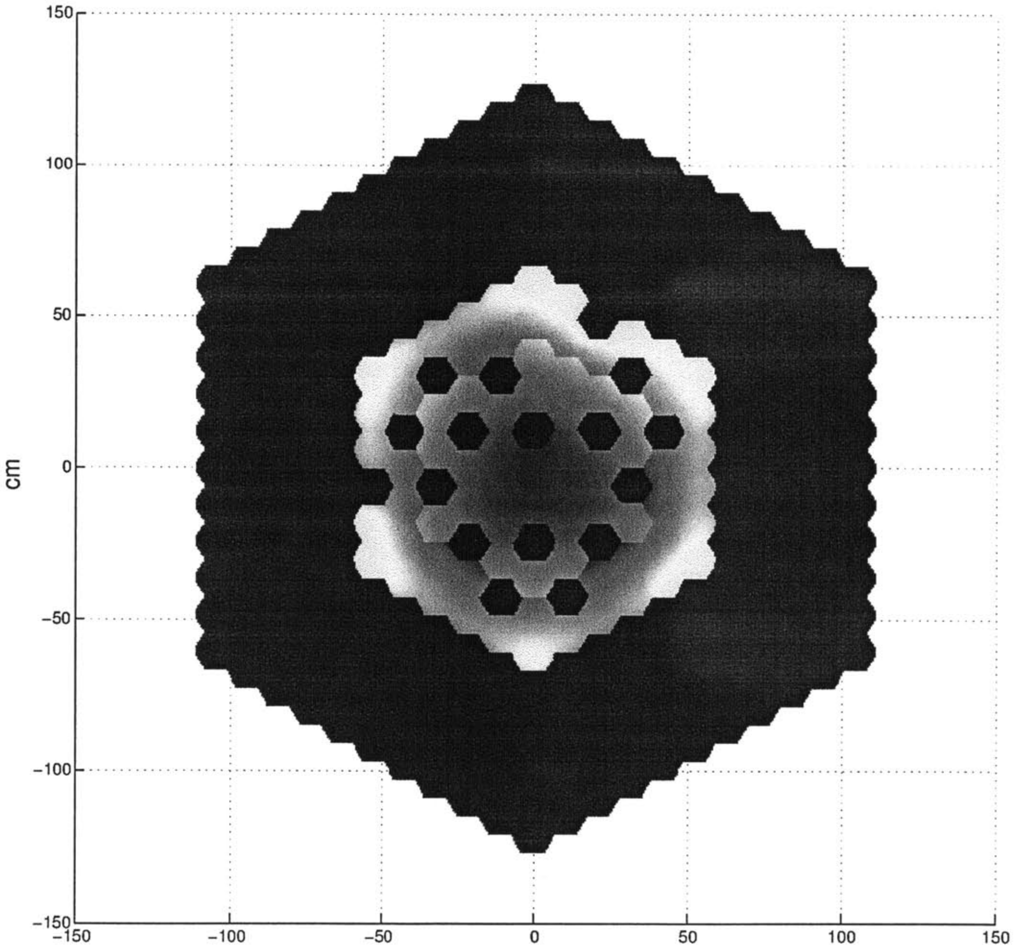

Figure B.4: A hyperfine MaPS solution for power density in the FFTF benchmark. Each hex assembly contains 294 triangles.

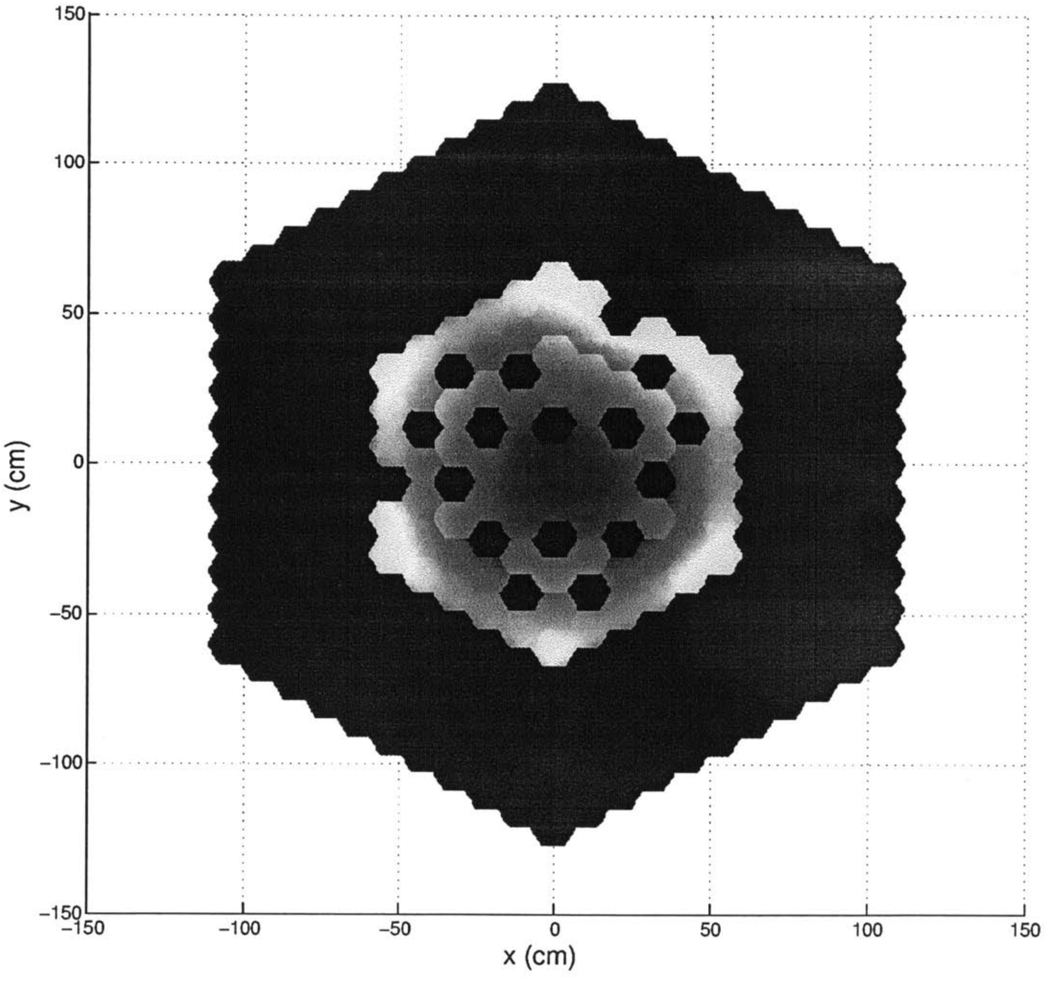

Figure B.5: Reconstruction of power density in the FFTF benchmark. Compare this to Figure B.4.

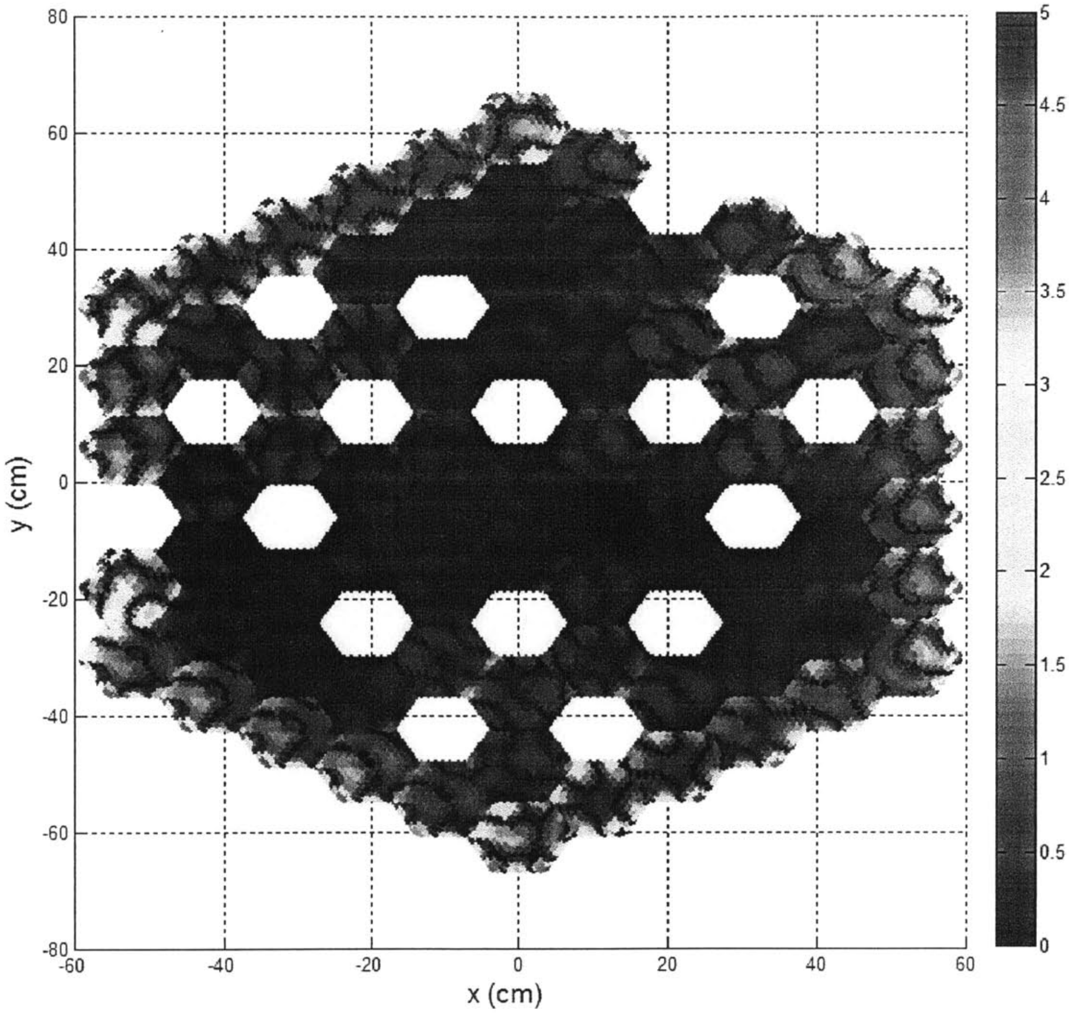

Figure B.6: The percent error magnitude between the reconstructed power density (Figure B.5) and the exact hyperfine mesh solution (Figure B.4) for the FFTF benchmark. The average error magnitude is 0.33% with a standard deviation of 0.53%.

B.4.2 A Homogenous Core with FFTF Fuel

This is a homogenous core with 7 full hex assembly rings in which all assemblies are identical to the FFTF "driver fuel" assemblies in the "inner enrichment zone". Homogenous cores are excellent test cases for power reconstruction, as they produce very smooth and continuous flux and power distributions. Pin power reconstruction errors should thus be very small.

Figures B.7 and B.8 show the homogenous core power density distributions derived from the exact hyperfine mesh solution and the flux reconstruction, respectively. These two distributions are completely indistinguishable to the naked eye.

Figure B.9 shows the error between these two distributions. The error is extremely low throughout the core internal, but it spikes to above 5% in fuel assemblies that neighbor the vacuum boundary. The low flux magnitudes adjacent to the vacuum boundary cause this error, which affects those entire assemblies. Fortunately, no practical reactor designs would expose fuel to vacuum without intermediate reflectors or shields, and so this is not a problem. If we neglect these edge assemblies, then the core-wide mean error magnitude is 0.03% with a standard deviation of 0.06%.

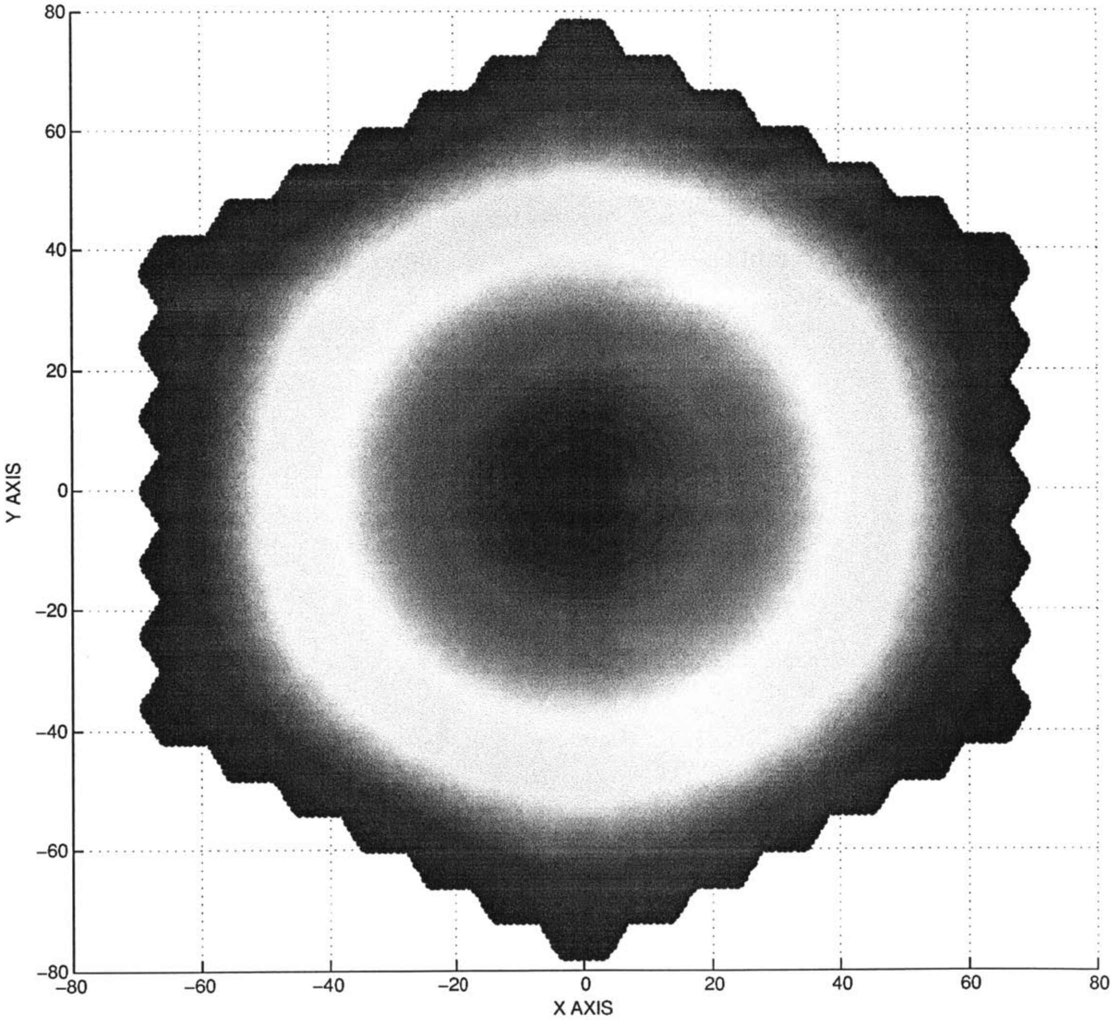

Figure B.7: A hyperfine MaPS solution for power density in a small homogenous core with FFTF MOX fuel. Each hex assembly contains 294 triangles.

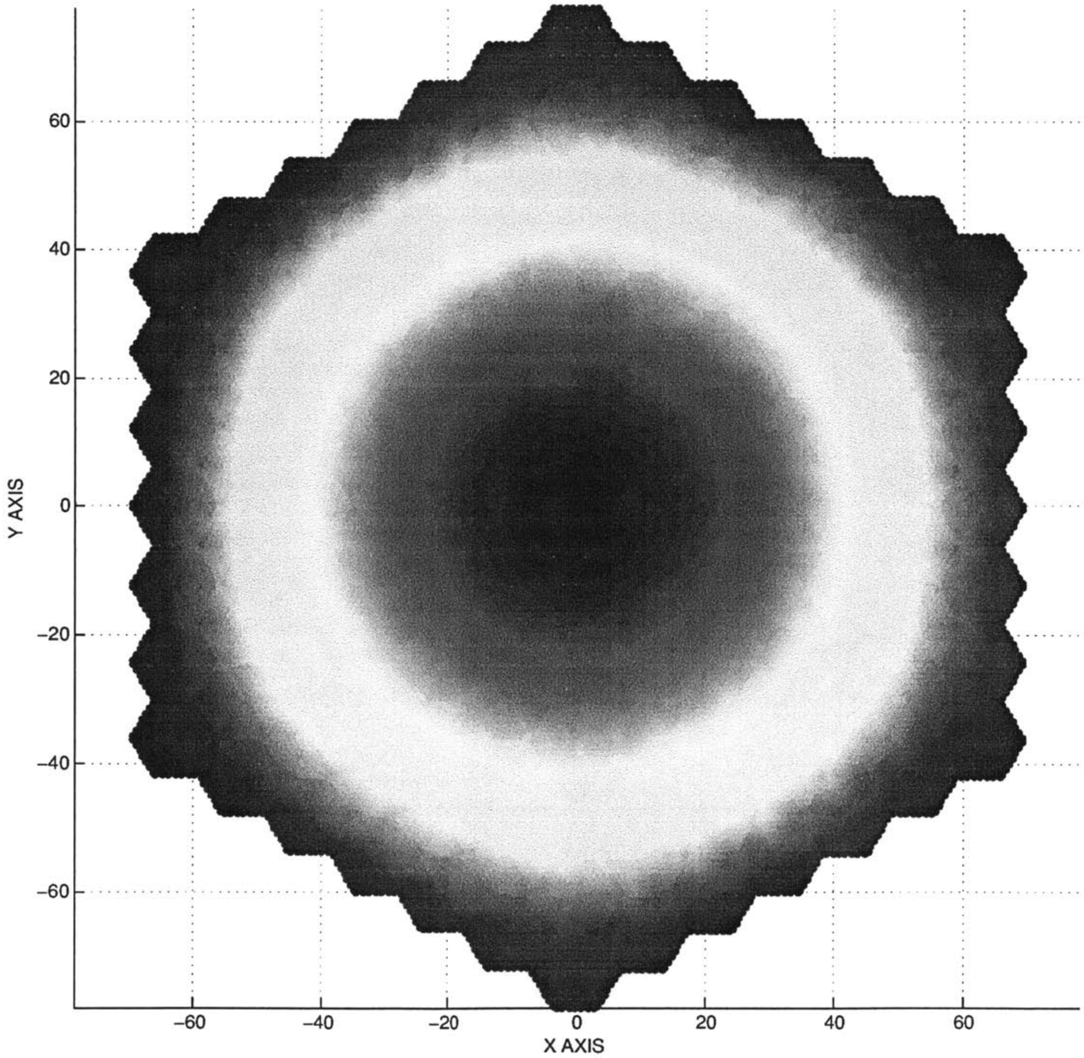

Figure B.8: Reconstruction of power density in a small homogenous core with FFTF MOX fuel. Compare this to Figure B.7. Can you tell the difference?

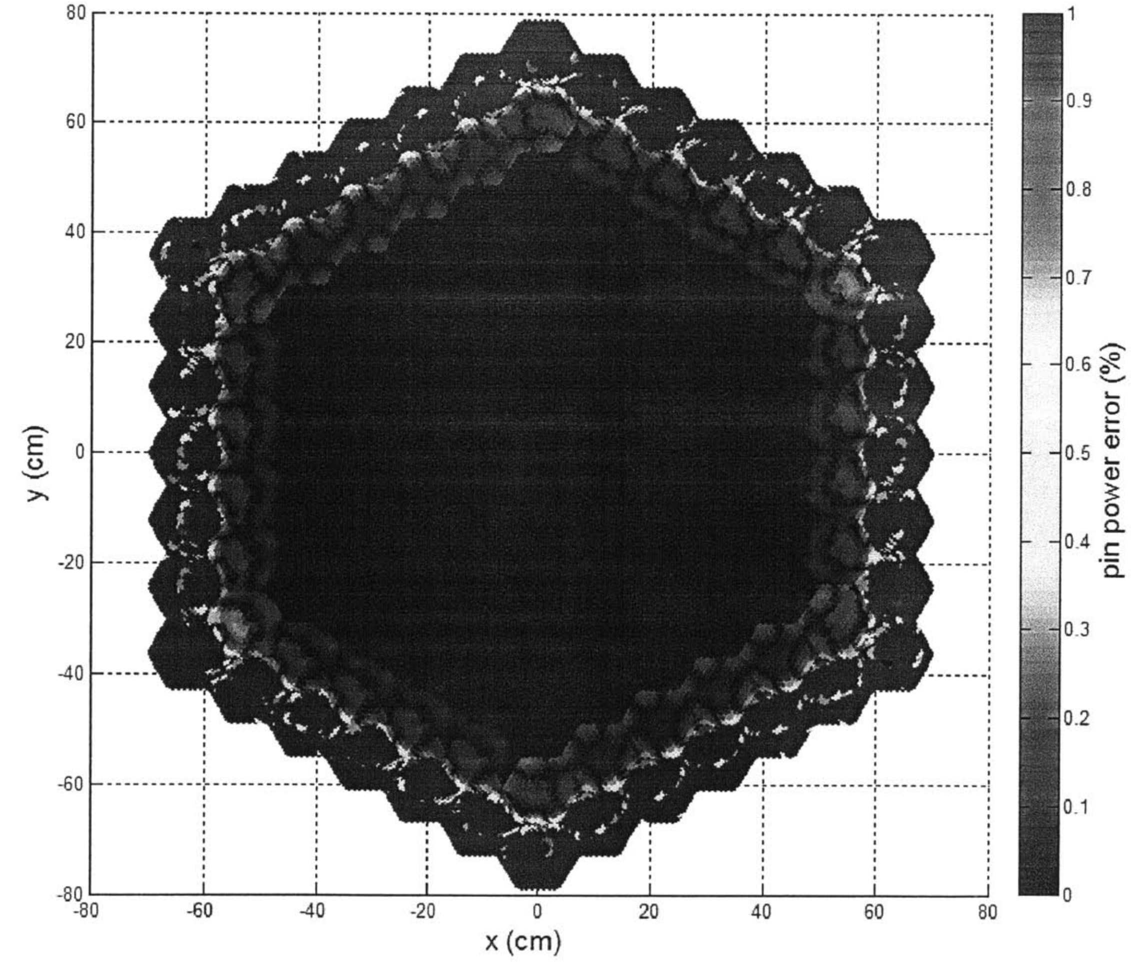

Figure B.9: The percent error magnitude between the reconstructed power density (Figure B.8) and the exact hyperfine mesh solution (Figure B.7) for the homogenous core with FFTF MOX fuel. Fuel assemblies adjacent to the vacuum boundary yield large errors, but if those assemblies are excluded, the core-averaged mean error magnitude is 0.03% with a standard deviation of 0.06%. Yes, the errors really are that small!

Table B.1: Core-Wide Mean Pin Power Errors

	error magnitude (%)	standard deviation (%)
FFTF	0.33	0.53
Homogenous core	0.03	0.06

B.4.3 Summary of Reference Case Findings

Table B.1 shows mean pin power (or power density) error magnitudes and their standard deviations in the two references cases we have analyzed. Most of the pin power error is concentrated in the outer assemblies, which are relatively unimportant for thermal hydraulic and mechanical distortion effects.

Figure B.10 shows cumulative distribution functions (CDFs) of error magnitudes for the same two cases shown in the table above.

We can draw some general conclusions from our work with these two reference cases that should guide anyone who uses this pin power reconstruction code for arbitrary core designs:

1. High flux gradients do *not* correlate with high error magnitudes in pin power reconstruction.

2. Higher order flux derivatives, such as changes in flux curvature within a single assembly, sometimes *do* correlate with relatively high error magnitudes in pin power reconstruction.

3. Low flux and power density magnitudes strongly correlate with high error magnitudes in power reconstruction. This effect is much stronger than the effect in (2).

4. When fuel is adjacent to a vacuum boundary, pin power reconstruction is not suitably accurate.

5. Corner-point fluxes tend to introduce error in pin power reconstruction when those corners are located on large discontinuities in material composition or flux spectrum.

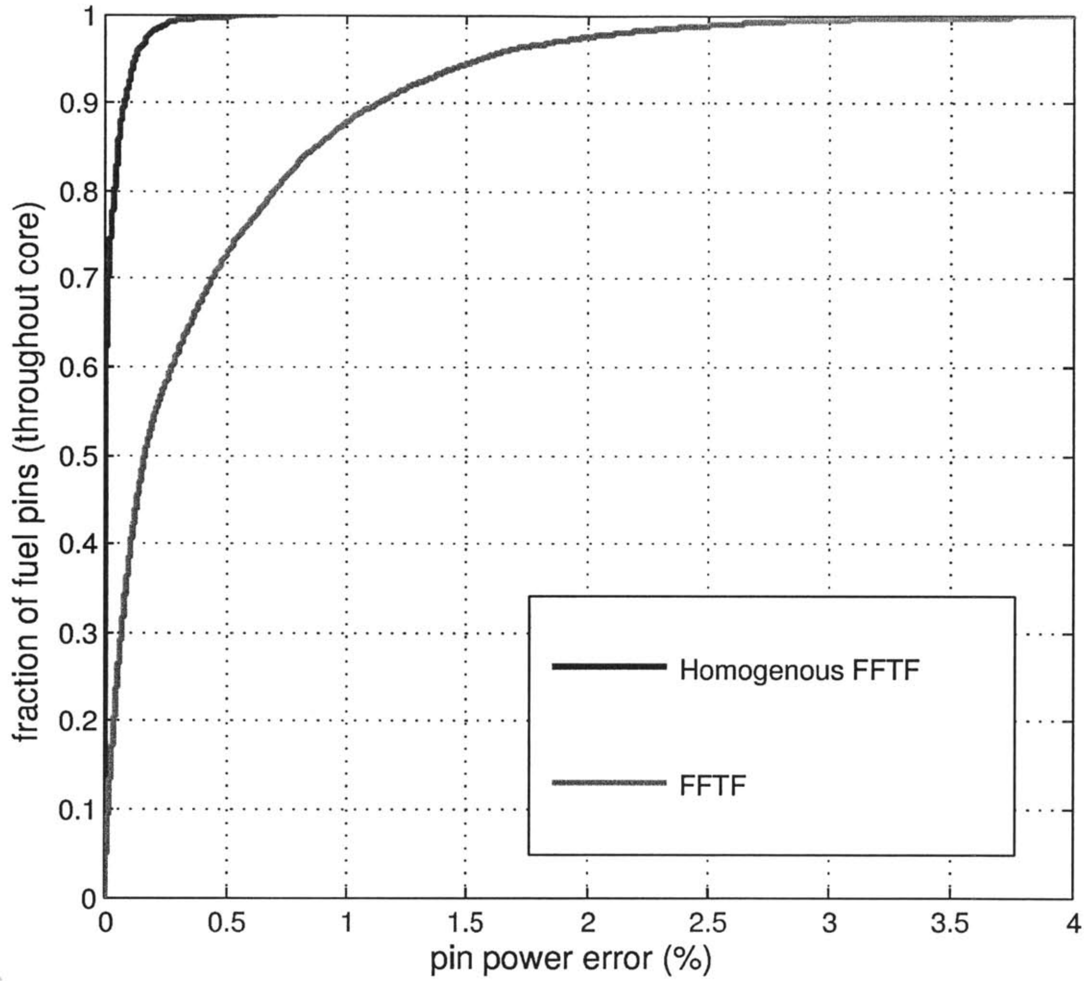

Figure B.10: Cumulative distribution functions (CDFs) for pin power error magnitudes between hyperfine mesh solutions and reconstructed solutions. Here the homogenous core CDF neglects the outer ring of fuel assemblies adjacent to the vacuum boundary.

C Hex Assembly Ordering and Orientation

Hexagons are not squares. Hexagonal lattices are not Cartesian. Consequently, people often confuse themselves over how to enumerate hexagons. There is no standard way to index or order hexagons within a hexagonal lattice, so each code developer conjures up his or her own way. Unlike the honeybees in Figure C.1, humans have no standard, innate method for this.

Figure C.1: Honeybees construct honeycomb as hexagonal lattices. While keeping track of hexagons is innate and intuitive for these bees, we humans are not so sophisticated.

This Appendix C and Figures C.3 - C.8 describe and illustrate each hex ordering employed to transfer data from the DIF3D [193] binary output file NHFLUX to ARMI [211]. This is a Python module embedded within ARMI. There is some overlap in the descriptions of each ordering convention, but we must be very clear in defining these orderings and how they are distinct.

ARMI indexes hex assemblies by the index pair (i, j) = (ring, position). We will call this "ARMI nodal" ordering. See Figures C.3 and C.4 for diagrams of "ARMI nodal" ordering.

Unfortunately, the DIF3D NHFLUX file stores hex assembly data in "DIF3D 'four color' nodal" ordering, which is a mosaic of four interwoven (ring, position) "ARMI nodal" orderings. Although this ordering appears bizarre and purposeless, it actually minimizes the "spectral radius" of the nodal calculation. See Figure C.8 for a diagram of "DIF3D 'four color' nodal" ordering.

Fortunately, NHFLUX contains a data structure called ITRMAP, which maps "DIF3D 'four color' nodal" ordering to what we call "DIF3D GEODST" ordering. GEODST hex ordering is based on an oblique (or bent) set of x-y axes. One could consider it to be pseudo-Cartesian. Indices i and j correspond to these pseudo-Cartesian axes. See Figures C.6 and C.7 for diagrams of "DIF3D GEODST" ordering.

ARMI already contains functionality that converts "ARMI nodal" ordering into "MCNP GEODST" ordering, which is similar to "DIF3D GEODST" ordering except that one of the oblique axes is bent in a different direction. See Figure C.5 for a diagram of "MCNP GEODST" ordering.

Since converting between "DIF3D GEODST" ordering and "MCNP GEODST" ordering is trivial, that is where we make the final (very indirect) link between "ARMI nodal" and "DIF3D 'four color' nodal". This final link, between "DIF3D GEODST" and "MCNP GEODST", is simple:

$$j_{\text{dif3d}} = j_{\text{mcnp}} + i_{\text{mcnp}} \tag{C.1}$$

$$i_{\text{dif3d}} = j_{\text{mcnp}} \tag{C.2}$$

Figure C.2 illustrates the three-step process (involving four hex orderings) of transferring data from DIF3D NHFLUX to ARMI. After one is aware of this process (the DIF3D manual does not describe it in detail), executing it is not particularly difficult. ITRMAP converts between "DIF3D 'four color' nodal" and "DIF3D GEODST", while Python code converts between "MCNP GEODST" and "ARMI nodal".

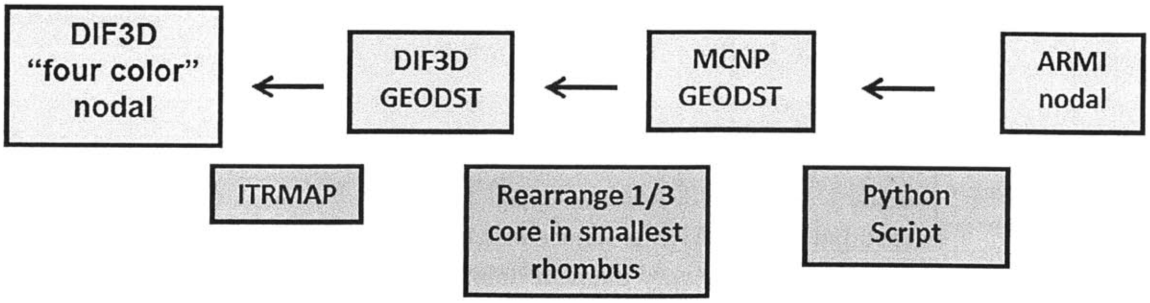

Figure C.2: Converting hex ordering schemes from ARMI to DIF3D NHFLUX.

What is much less trivial, however, is the caveat - that DIF3D not only re-orders hex assemblies, but also re-arranges them. DIF3D arranges all existing assemblies to fit within the smallest possible rhombus. For an example of this, compare Figures C.5 and C.6, which show the "MCNP GEODST" and "DIF3D GEODST" orderings, respectively. In order to fit all hex assemblies within the smallest possible rhombus, DIF3D moves some assemblies to the locations of their "symmetric identical" assemblies. This is particularly important for a 1/3 core symmetry model, in which each assembly (except the central assembly) has two "symmetric identical" positions.

Of course, when a hex assembly is moved to one of its "symmetric identical" positions, it is also rotated. Thus, we must keep track of each hex assembly's orientation, which may change. Orientation is essential in terms of obtaining the correct surface partial current data (and especially for computation of corner-point fluxes). This issue is even further complicated when one must add or subtract the partial surface currents of two neighboring hex assemblies, and only one of those assemblies has been moved or rotated. Consequently, the usual relationship between indices of adjoining hex surfaces does not always apply. The flux reconstruction module tediously resolves all of this.

C.1 Conventional Nodal Ordering

ARMI indexes hex assemblies by the index pair (i, j) = (ring, position). We will call this "ARMI nodal" ordering. This ordering can be indexed with the two variables (i, j) as in Figure C.3, or it can be indexed with one variable as in Figure C.4. When indexed with two variables, the indexing includes non-existing "symmetric identical" assembly positions (such as the "missing" 2/3 of assemblies in a 1/3 core model). When indexed with one variable, the indexing does not include the "symmetric identical" assembly positions - it skips over them.

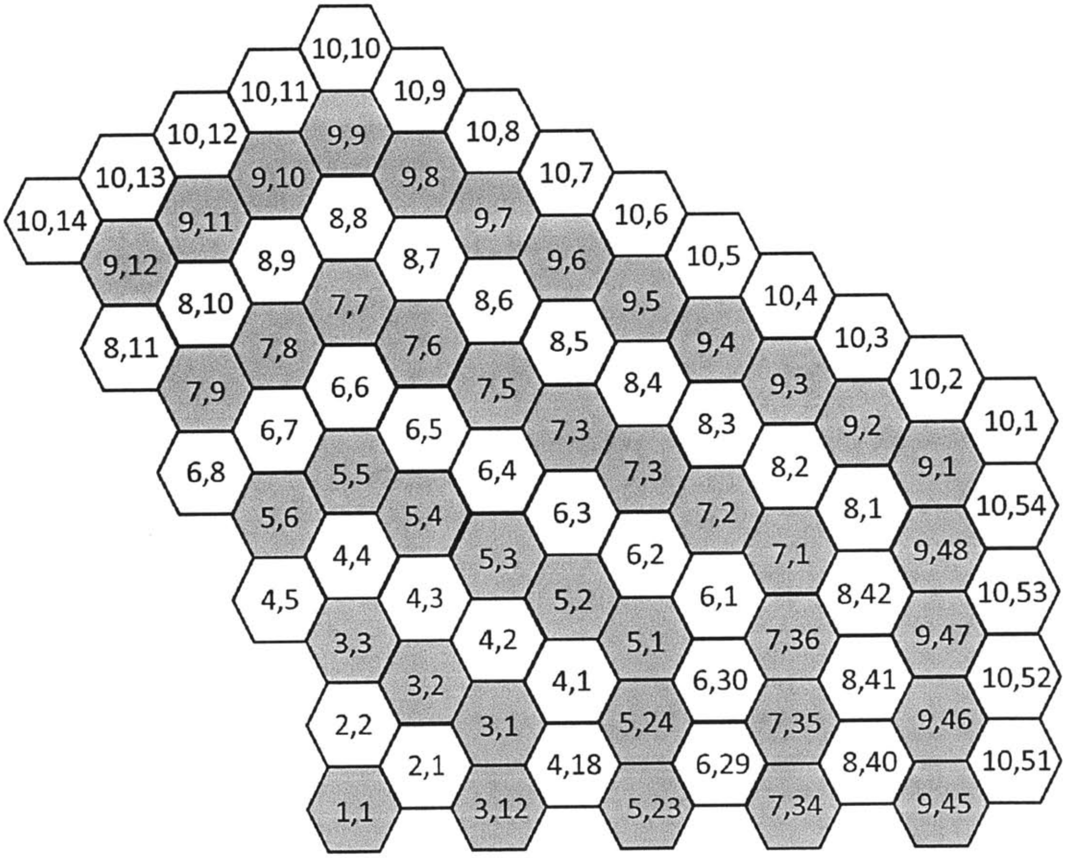

Figure C.3: 'ARMI nodal" hex ordering in (ring, position) format for a 1/3 core model with 10 full rings. Rings are alternately colored white and grey. For 1/3 and 1/6 core models, indices include the empty "symmetric identical" assembly positions.

C.2 MCNP GEODST Ordering

GEODST hex ordering is based on an oblique (or bent) set of x-y axes. One could consider it to be pseudo-Cartesian. Indices i and j correspond to these pseudo-Cartesian axes. The flux reconstruction module employs two types of GEODST ordering: "MCNP GEODST" and "DIF3D GEODST". These are essentially the same, except that the pseudo-Cartesian x-y axes are "bent" in different directions. Compare Figures C.5 and C.6.

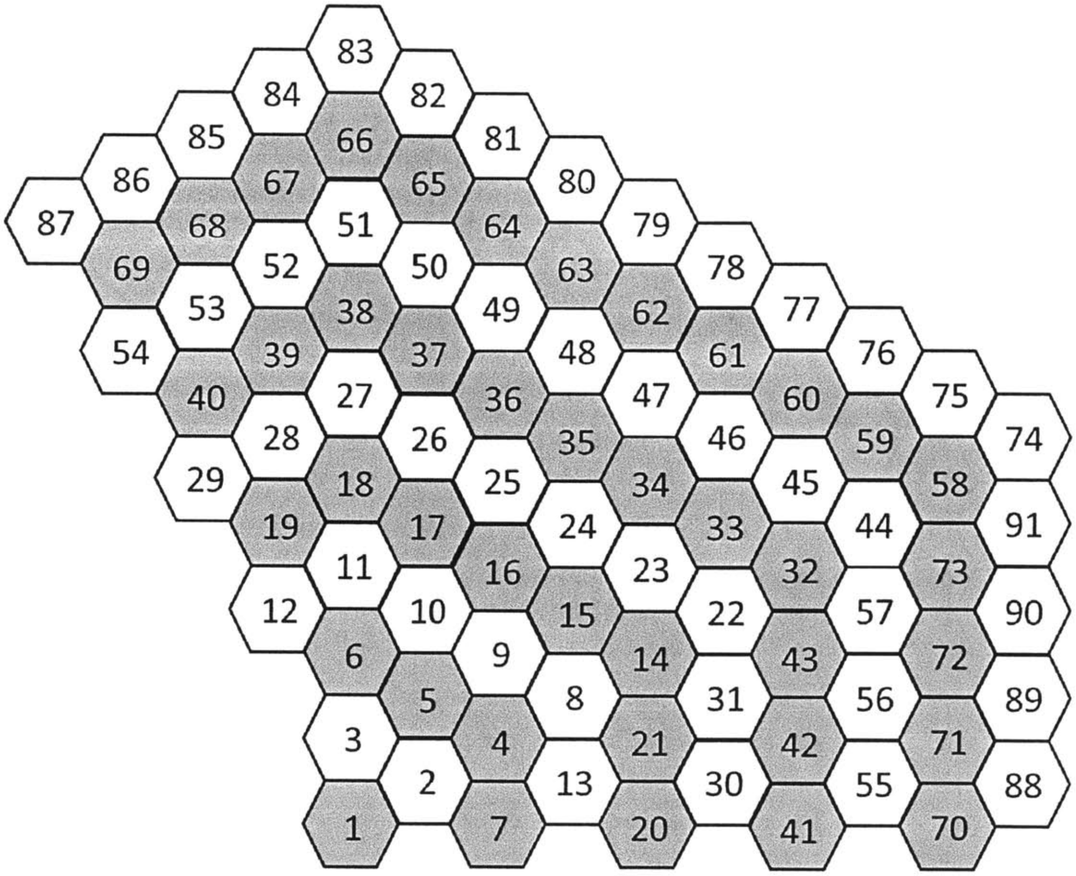

Figure C.4: "ARMI nodal" hex ordering in 1-D format for a 1/3 core model with 10 full rings. Note the obvious correspondence between this and Figure C.3. Rings are alternately colored white and grey. Here, the single index does not include the empty "symmetric identical" positions. The index values are always consecutive without any "gaps".

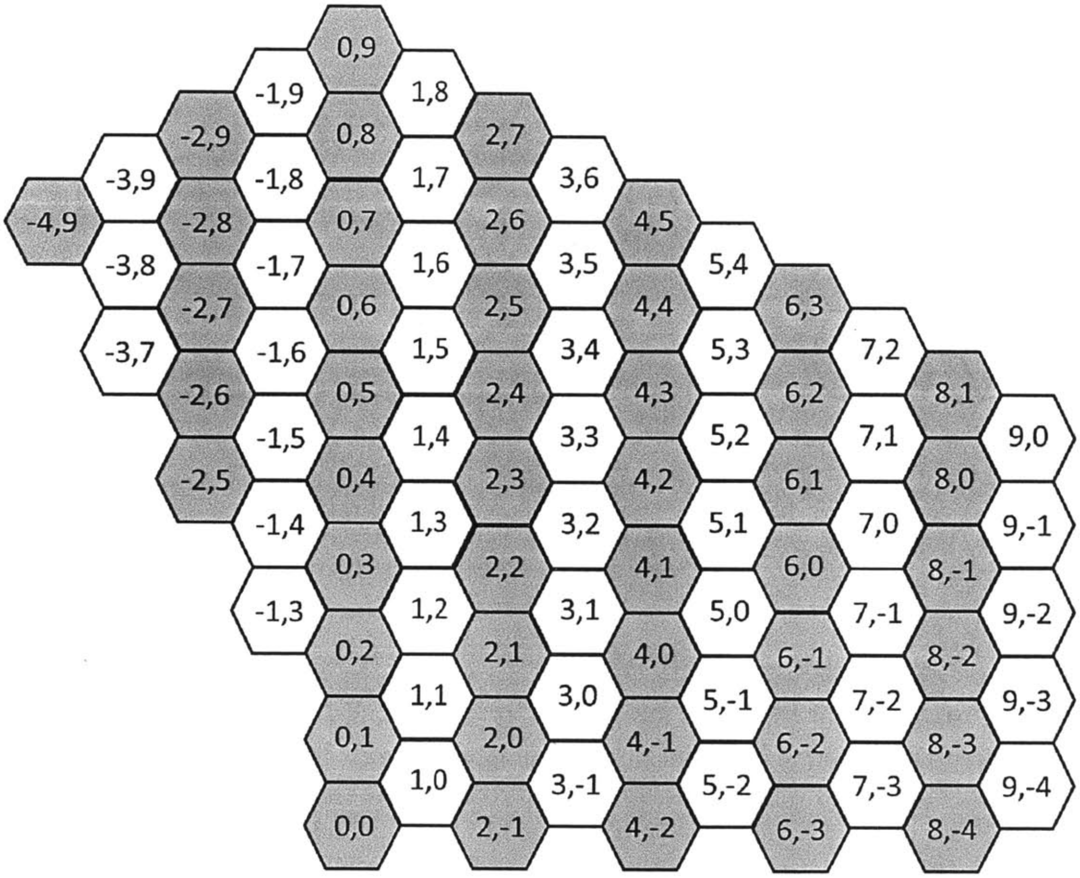

Figure C.5: "MCNP GEODST" hex ordering in $(i, j) = (x, y)$ format for a 1/3 core model with 10 full rings. Here rows of constant i are alternately colored white and grey.

C.3 DIF3D GEODST Ordering

"DIF3D GEODST" ordering is similar to "MCNP GEODST" ordering except that one of the pseudo-Cartesian axes is bent in a different direction. One can easily convert indices between these two orderings using Equations C.1 and C.2.

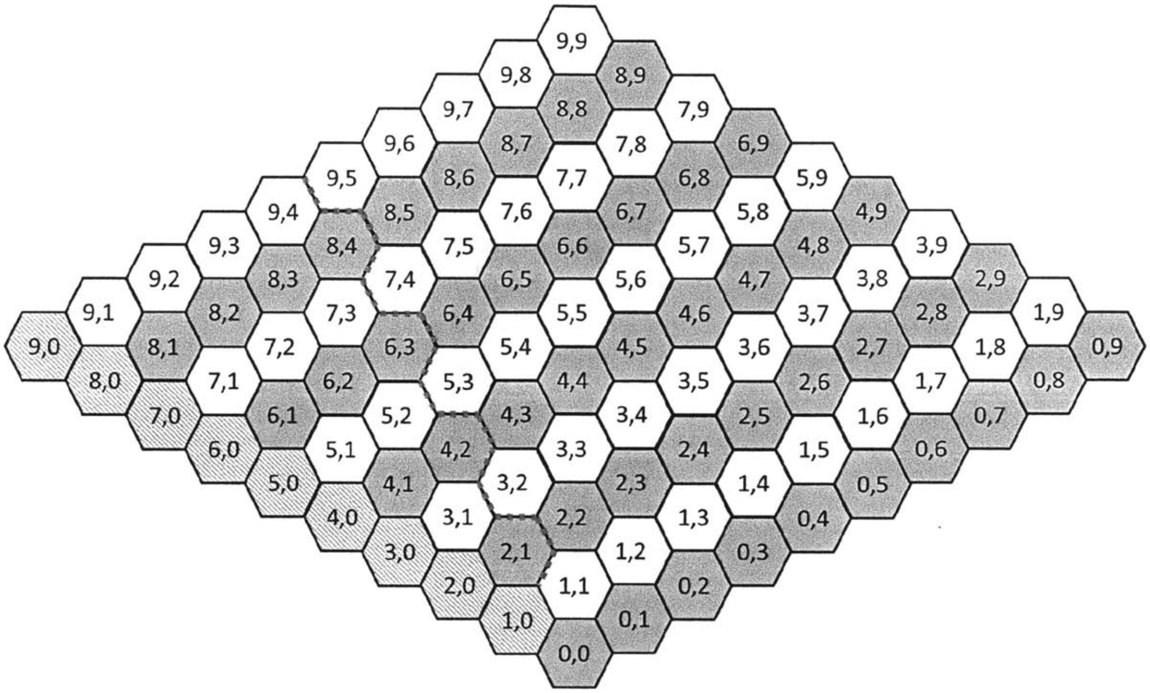

Figure C.6: "DIF3D GEODST" hex ordering in $(i, j) = (x, y)$ format for a 1/3 core model with 10 full rings. Here rows of constant i are alternately colored white and grey. DIF3D re-arranges hex assemblies to fit within the smallest possible rhombus. The empty "ghost" positions (which fill out the rhombus) are filled with stripes. In order to convert to the standard ARMI 1/3 core model, all assemblies left of the red line must be transferred to their "symmetric identical" positions (and rotated).

Although converting between "DIF3D GEODST" and "MCNP GEODST" indexing is trivial, the re-arrangement of hex assembly positions is significantly less trivial. As described previously, DIF3D re-arranges all hex assemblies to fit within the smallest possible rhombus (so that the i and j GEODST indices have the same minimum and maximum values).

C.4 DIF3D "Four Color" Nodal Ordering

Unfortunately, the DIF3D NHFLUX file stores hex assembly data in "DIF3D 'four color' nodal" ordering, which is a mosaic of four interwoven (ring, position) "ARMI nodal" or-

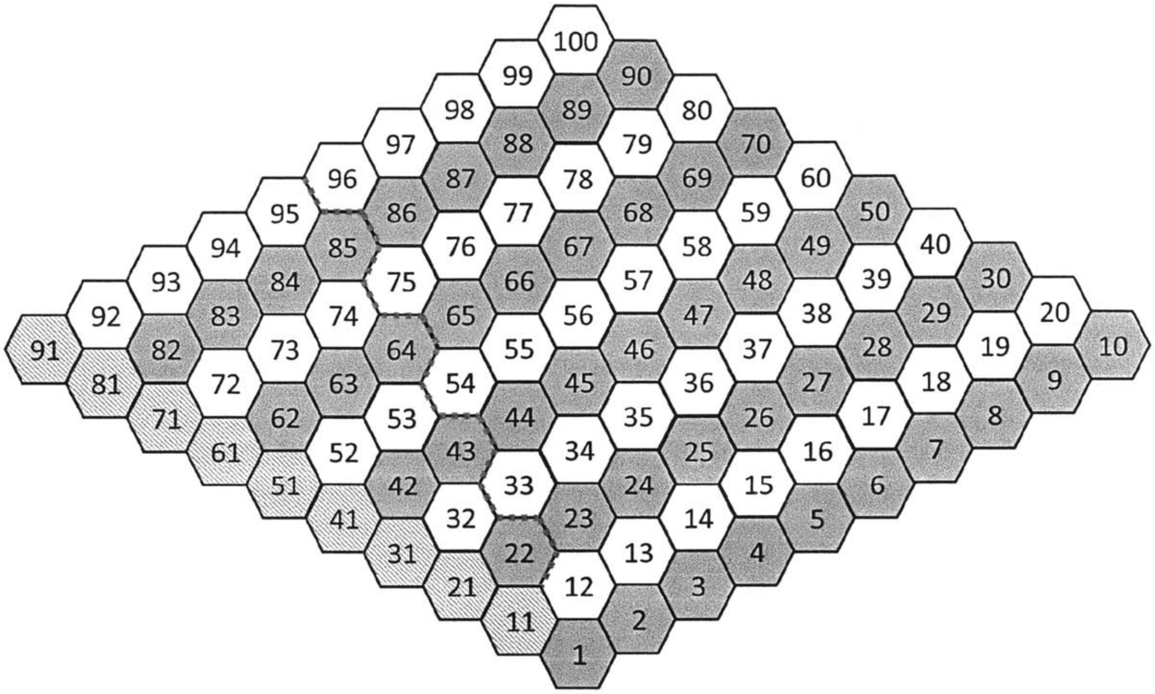

Figure C.7: "DIF3D GEODST" ordering in 1-D format for a 1/3 core model with 10 full rings. Here rows of constant j are alternately colored white and grey. DIF3D re-arranges hex assemblies to fit within the smallest possible rhombus. The empty "ghost" positions (which fill out the rhombus) are filled with stripes. Note that the 1-D index includes these "ghost" positions. In order to convert to the standard ARMI 1/3 core model, all assemblies left of the red line must be transferred to their "symmetric identical" positions (and rotated).

derings. Although this ordering appears bizarre and purposeless, it actually minimizes the "spectral radius" of the nodal calculation.

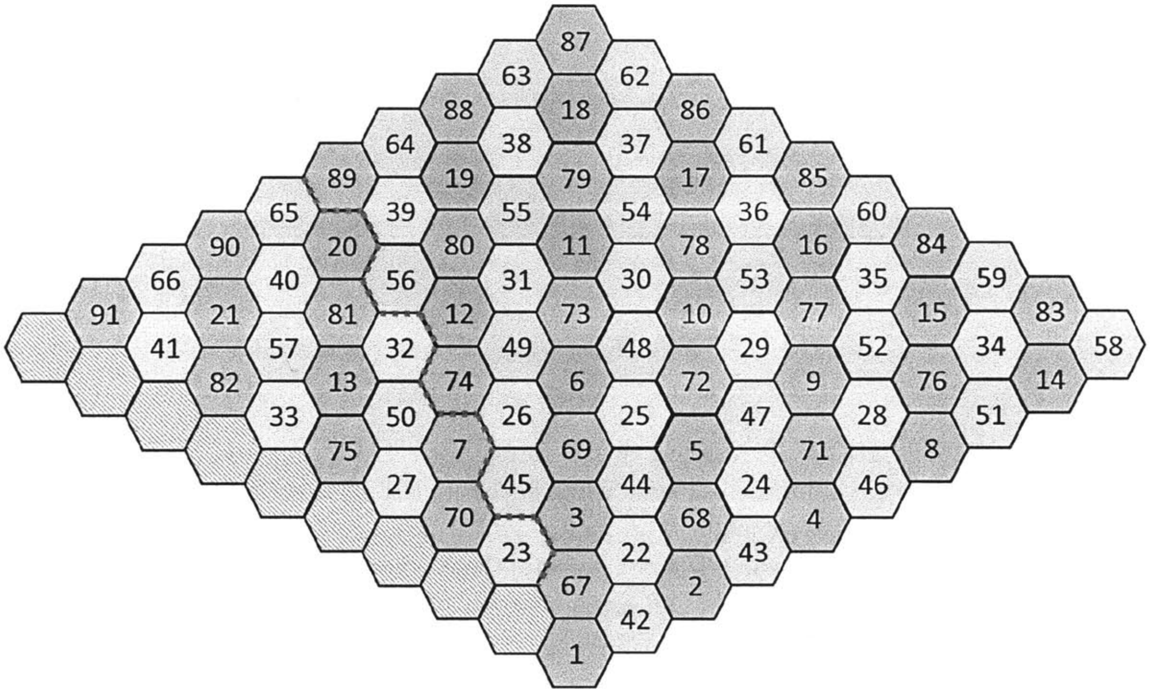

Figure C.8: "DIF3D 'four color' nodal" ordering in 1-D format for a 1/3 core model with 10 full hex rings. Note that this is really a mosaic of four interwoven (ring, position) orderings. Each color represents a distinct (ring, position) ordering. DIF3D re-arranges hex assemblies to fit within the smallest possible rhombus. The empty "ghost" positions (which fill out the rhombus) are filled with stripes. Note that the 1-D index does not include these "ghost" positions. In order to convert to the standard ARMI 1/3 core model, all assemblies left of the red line must be transferred to their "symmetric identical" positions (and rotated).

C.4.1 DIF3D Full Core Orderings

Now that we have thoroughly elucidated the wonders of hex assembly ordering for 1/3 core models, let's consider full core models. The "ARMI nodal" and "MCNP GEODST" full core orderings are simple extensions of their 1/3 core counterparts, so we will not bother to display them.

Figure C.9 shows the full core "DIF3D 'four color' nodal" ordering for a small core with 6 full hex rings. This is similar to the 1/3 core model, except that the rhombus is arranged differently around the core in this case.

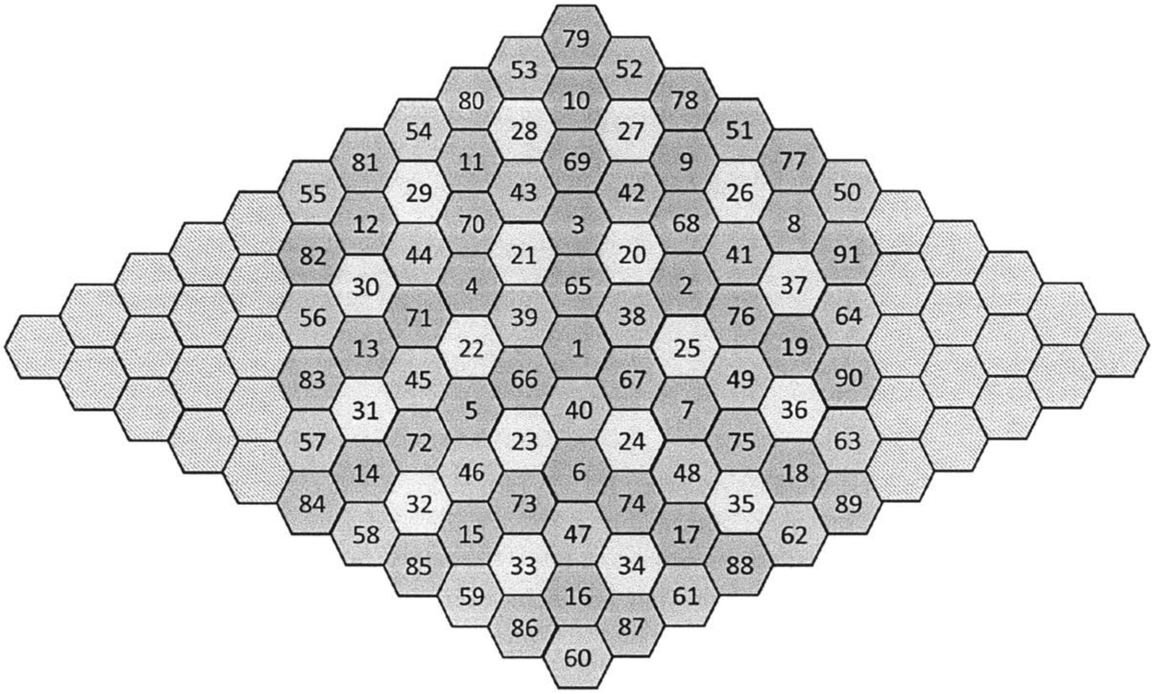

Figure C.9: "DIF3D 'four color' nodal" ordering in 1-D format for a full core model with 6 full hex rings. Note that this is really a mosaic of four interwoven (ring, position) orderings. Each color represents a distinct (ring, position) ordering. DIF3D re-arranges hex assemblies to fit within the smallest possible rhombus. The empty "ghost" positions (which fill out the rhombus) are filled with stripes, but they are not included in the numbering. Compare this to Figure C.8, which shows a 1/3 core model.

Figure C.10 shows the full core "DIF3D GEODST" ordering for the same small core with 6 full hex rings. This is distinctly different than its 1/3 core counterpart shown in Figure C.7. The central hex is not "number 1" here. The rhombus is arranged differently around the core model. However, this is much simpler to manipulate than its 1/3 core counterpart, because no assemblies must change position (or orientation) when converting to "ARMI nodal" ordering.

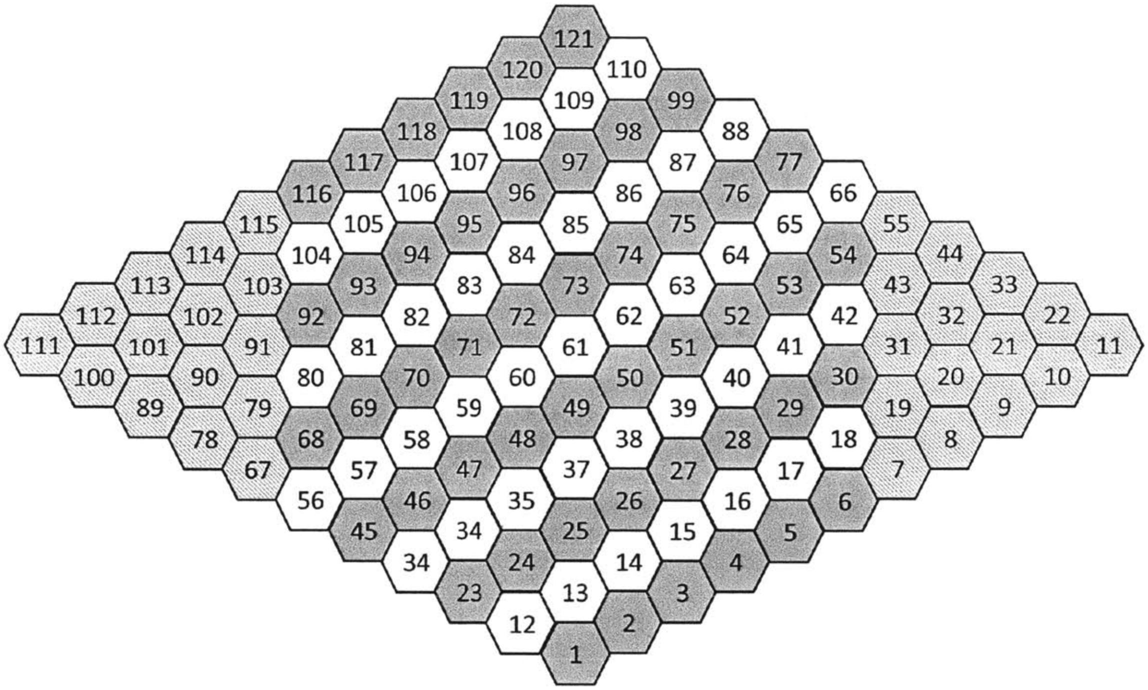

Figure C.10: "DIF3D GEODST" ordering in 1-D format for a full core model with 6 full hex rings. Here rows of constant j are alternately colored white and grey. DIF3D re-arranges hex assemblies to fit within the smallest possible rhombus. The empty "ghost" positions (which fill out the rhombus) are filled with stripes, and they are included in the numbering. This hex ordering is unique in that the central hex is not "number 1". Compare this to Figure C.7, which shows a 1/3 core model.

D Improvements to Traditional Boundary Perturbation Theory via Interpolation Techniques

Here we describe a few methods for interpolating the classic perturbation theory formula between finite difference mesh cells. We experimented with thesis methods early on in this thesis work, before beginning work on "virtual density" theory.

Evaluating perturbation theory requires expressing the core as a grid structure of discrete cells. For example, consider evaluating the first order perturbation result.

$$\Delta\rho' = \frac{\left\langle \phi^{\dagger} \middle| \left(\frac{1}{k} d\hat{F} - d\hat{M}\right)\phi \right\rangle}{\left\langle \phi^{\dagger} \middle| \hat{F}\phi \right\rangle} \tag{D.1}$$

In the denominator, F is nonzero in every region throughout the core that contains fissionable material. The product $\phi^{\dagger}F\phi$ has a certain value in each specified cell (with homogenized materials), and it is these values that are weighted by cell volume and summed up to obtain the whole-core inner product $\left\langle \phi^{\dagger} \middle| \hat{F}\phi \right\rangle$.

In the numerator, dF and dM are nonzero only in the locality of the perturbation. Thus, if a geometric distortion occurs *entirely within* a single cell, the homogenized dF and dM will still be zero in that cell. Consequently, in order to capture the effects of a geometric perturbation, the cell dimensions in the immediate vicinity of that perturbation must be at least as small as the perturbation displacement. In order to change dF or dM, a geometric perturbation must alter the homogenized materials in neighboring cells.

The problem with this is that many geometry distortions result in only very small displacements. Were we to employ a fixed grid of cells throughout the entire core, we would have an exceedingly large number of cells. We would obtain the desired resolution of $\left\langle \phi^{\dagger} \middle| \left(\frac{1}{k} d\hat{F} - d\hat{M}\right)\phi \right\rangle$, but our computation of $\left\langle \phi^{\dagger} \middle| \hat{F}\phi \right\rangle$ would be extremely inefficient. The best approach might actually be to temporarily refine the cell structure in the vicinity of the perturbation.

Of course, Larsen and Pomraning showed in their very first paper on boundary perturbation theory that this conundrum can be circumvented by transforming the volume integral into a surface integral over the unperturbed surface [42]. However, in this section we explore possible interpolation techniques to evaluate the volume integral directly over small spatial regions.

D.1 Linear Interpolation of Unperturbed Fluxes

D.1.1 1-D Linear Interpolation

The numerator in the expression for a first order perturbation is

$$\left\langle \phi^\dagger \middle| \left(\frac{1}{k} d\hat{F} - d\hat{M}\right) \phi \right\rangle \tag{D.2}$$

In 1-D, the volume integration is typically performed by assuming that the unperturbed flux ϕ_i is constant in each region i. Thus, integrating by volume amounts to computing the energy group dot product in each region i and then multiplying the result by the 1-D width Δx. So the contribution to the total 1-D perturbation for a *single* 1-D region is

$$\Delta x \phi_i^\dagger \left(\frac{1}{k} d\hat{F} - d\hat{M}\right) \phi_i \tag{D.3}$$

For a very fine mesh, this is adequate. However, for unperturbed fluxes computed on more coarse meshes, it is more accurate to linearly interpolate the unperturbed fluxes so that the contribution of each 1-D region i becomes

$$\Delta x \phi_i^\dagger \left(\frac{1}{k} d\hat{F} - d\hat{M}\right) \phi_i \rightarrow \int dx\, \phi^\dagger(x) \left(\frac{1}{k} d\hat{F} - d\hat{M}\right) \phi(x) \tag{D.4}$$

Here we will assume that both $\phi(x)$ and $\phi^\dagger(x)$ are linear functions whose averages are preserved at ϕ_i and $\phi_i^\dagger$. If we define $x = 0$ as the center of each mesh-centered 1-D region i, then

$$\phi(x) = Ax + \phi_i \tag{D.5}$$

It makes sense to define the constant A as the average slope (or 1-D gradient) of ϕ in the region i, which is

$$A = \frac{1}{2}\left(\frac{\phi_{i+1} - \phi_i}{\Delta x} + \frac{\phi_i - \phi_{i-1}}{\Delta x}\right) = \frac{\phi_{i+1} - \phi_{i-1}}{2\Delta x} \tag{D.6}$$

Now the explicit $\phi(x)$ is

$$\phi(x) = \frac{\phi_{i+1} - \phi_{i-1}}{2\Delta x} x + \phi_i \tag{D.7}$$

Note that this will cause $\phi(x)$ to be discontinuous at the boundaries of adjacent mesh regions, but we don't really care. Now we can express the contribution of region i as

$$\int_{-\Delta x/2}^{\Delta x/2} dx \left(\frac{\phi_{i+1}^\dagger - \phi_{i-1}^\dagger}{2\Delta x} x + \phi_i^\dagger\right) \left(\frac{1}{k} d\hat{F} - d\hat{M}\right) \left(\frac{\phi_{i+1} - \phi_{i-1}}{2\Delta x} x + \phi_i\right) \tag{D.8}$$

Remember we assumed that $x = 0$ corresponds to the center of the mesh region so that the region of width Δx extends from $-\Delta x/2$ to $\Delta x/2$. We can expand this so that each

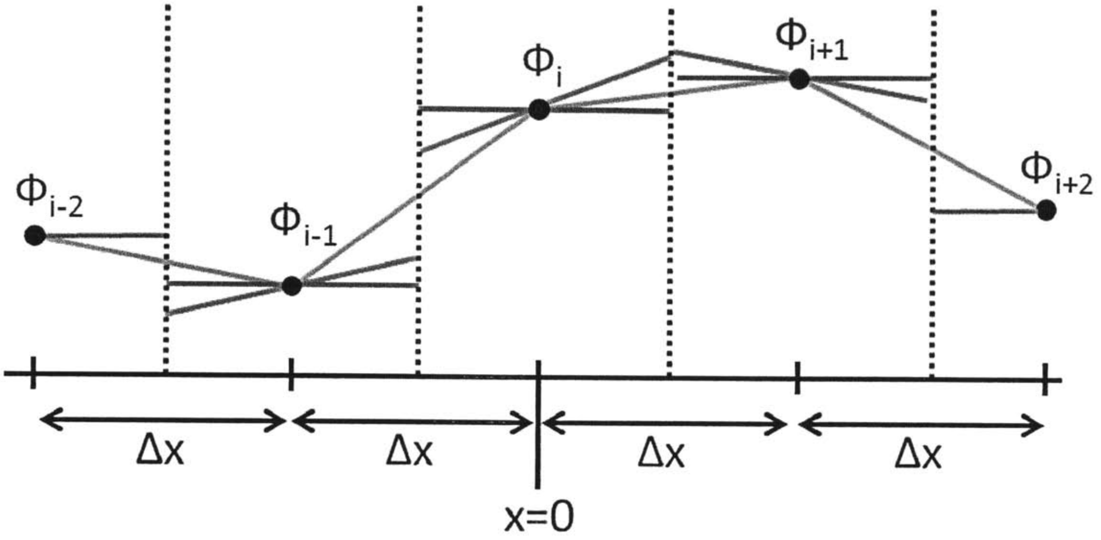

Figure D.1: Three methods of flux interpolation between 1-D mesh-centered points. The red and blue lines represent the constant flux approximation, which nearly all traditional perturbation theory methods employ. The green lines represent linear interpolation fixed such that the flux is continuous at mesh boundaries. Note that this requires two piecewise linear flux functions in each mesh cell. The red lines represent a single linear function in each mesh cell chosen so that its slope corresponds to the average flux slope. This is our method of choice. Note that the cell spacing Δx is constant.

term in the integrand contains an energy group dot product of only one adjoint flux and one real flux.

Now the integration is fairly straightforward. All the x terms are zero due to symmetry. The x^2 terms and the constant terms are simple.

$$\int_{-\Delta x/2}^{\Delta x/2} dx = \Delta x \tag{D.9}$$

$$\int_{-\Delta x/2}^{\Delta x/2} dx\, x = 0 \tag{D.10}$$

$$\int_{-\Delta x/2}^{\Delta x/2} dx\, x^2 = \frac{(\Delta x)^3}{12} \tag{D.11}$$

Once the integration is performed, the expression becomes

$$\Delta x \left[\phi_i^\dagger \left(\frac{1}{k} d\hat{F} - d\hat{M} \right) \phi_i \right. \tag{D.12}$$

$$+ \frac{1}{48} \phi_{i+1}^\dagger \left(\frac{1}{k} d\hat{F} - d\hat{M} \right) \phi_{i+1} + \frac{1}{48} \phi_{i-1}^\dagger \left(\frac{1}{k} d\hat{F} - d\hat{M} \right) \phi_{i-1} \tag{D.13}$$

$$\left. - \frac{1}{48} \phi_{i-1}^\dagger \left(\frac{1}{k} d\hat{F} - d\hat{M} \right) \phi_{i+1} - \frac{1}{48} \phi_{i+1}^\dagger \left(\frac{1}{k} d\hat{F} - d\hat{M} \right) \phi_{i-1} \right] \tag{D.14}$$

The first term is identical to the constant flux implementation of perturbation theory, and we can thus interpret the subsequent four terms as smaller corrections due to the linear interpolation. However, we can combine these four "correction" terms into a single term that is more intuitive:

$$\Delta x \left[\phi_i^\dagger \left(\frac{1}{k} d\hat{F} - d\hat{M} \right) \phi_i + \frac{1}{48} \left(\phi_{i+1}^\dagger - \phi_{i-1}^\dagger \right) \left(\frac{1}{k} d\hat{F} - d\hat{M} \right) \left(\phi_{i+1} - \phi_{i-1} \right) \right] \tag{D.15}$$

D.1.2 2-D Linear Interpolation

The process for deriving the 2-D implementation of linear interpolation is the similar to that for 1-D. Instead of simply multiplying the constant flux energy group product by each region's area A, we integrate the linearly interpolated energy group product over that same area.

$$A\phi_i^\dagger \left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi_i \rightarrow \int\int dA\, \phi^\dagger(x,y)\left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\phi(x,y) \tag{D.16}$$

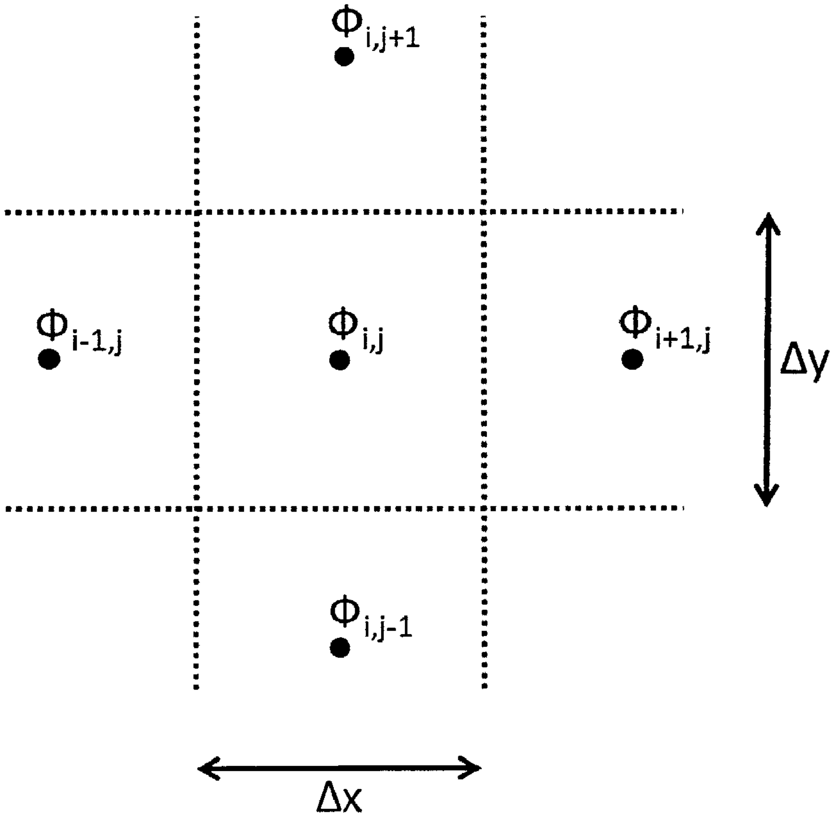

Figure D.2: Here are discrete flux values in mesh-centered cells in 2-D Cartesian geometry. The cell spacings in each direction Δx and Δy are constant. Each discrete flux is indexed with a i, j index pair. Our linear interpolation technique requires expressing the flux as a continuous linear function in x and y within each rectangular cell. The flux need not be continuous on cell boundaries.

Here $\phi(x, y)$ is a linear function in x and y. We work in Cartesian geometry so that each 2-D region is a rectangle that can be assigned a unique i, j index pair. Again, we assume that $(x, y) = (0, 0)$ corresponds to the center of the area so that the average flux is preserved as $\phi_{i,j}$.

$$\phi(x, y) = Ax + By + \phi_{i,j} \tag{D.17}$$

The constants A and B correspond to the average x and y gradients:

$$A = \frac{\phi_{i+1,j} - \phi_{i-1,j}}{2\Delta x} \tag{D.18}$$

$$B = \frac{\phi_{i,j+1} - \phi_{i,j-1}}{2\Delta y} \tag{D.19}$$

So the explicit $\phi(x, y)$ is

$$\phi(x, y) = \frac{\phi_{i+1,j} - \phi_{i-1,j}}{2\Delta x}x + \frac{\phi_{i,j+1} - \phi_{i,j-1}}{2\Delta y}y + \phi_{i,j} \tag{D.20}$$

Now the contribution of region i, j to the first order perturbation is

$$\int_{-\Delta x/2}^{\Delta x/2} dx \int_{-\Delta y/2}^{\Delta y/2} dy \left(\frac{\phi^\dagger_{i+1,j} - \phi^\dagger_{i-1,j}}{2\Delta x}x + \frac{\phi^\dagger_{i,j+1} - \phi^\dagger_{i,j-1}}{2\Delta y}y + \phi^\dagger_{i,j} \right) \tag{D.21}$$

$$\times \left(\frac{1}{k} d\hat{F} - d\hat{M} \right) \left(\frac{\phi_{i+1,j} - \phi_{i-1,j}}{2\Delta x}x + \frac{\phi_{i,j+1} - \phi_{i,j-1}}{2\Delta y}y + \phi_{i,j} \right) \tag{D.22}$$

Just as in 1-D, we can expand the terms. All terms with x, y or xy will be zero due to symmetry.

$$\int_{-\Delta x/2}^{\Delta x/2} dx \int_{-\Delta y/2}^{\Delta y/2} dy = \Delta x \Delta y \tag{D.23}$$

$$\int_{-\Delta x/2}^{\Delta x/2} dx \int_{-\Delta y/2}^{\Delta y/2} dy\, x = 0 \tag{D.24}$$

$$\int_{-\Delta x/2}^{\Delta x/2} dx \int_{-\Delta y/2}^{\Delta y/2} dy\, y = 0 \tag{D.25}$$

$$\int_{-\Delta x/2}^{\Delta x/2} dx \int_{-\Delta y/2}^{\Delta y/2} dy\, xy = 0 \tag{D.26}$$

$$\int_{-\Delta x/2}^{\Delta x/2} dx \int_{-\Delta y/2}^{\Delta y/2} dy\, x^2 = \frac{(\Delta x)^3}{12} \Delta y \tag{D.27}$$

$$\int_{-\Delta x/2}^{\Delta x/2} dx \int_{-\Delta y/2}^{\Delta y/2} dy\, y^2 = \Delta x \frac{(\Delta y)^3}{12} \tag{D.28}$$

Performing the integration yields a single term identical to the constant flux implementation plus *eight* correction terms - four for the x gradient and four for the y gradient. Just as in 1-D, we can condense the each set of four terms into a single more intuitive term:

$$\Delta x \Delta y \left[\phi_{i,j}^{\dagger} \left(\frac{1}{k} d\hat{F} - d\hat{M} \right) \phi_{i,j} \right. \tag{D.29}$$

$$+ \frac{1}{48} \left(\phi_{i+1,j}^{\dagger} - \phi_{i-1,j}^{\dagger} \right) \left(\frac{1}{k} d\hat{F} - d\hat{M} \right) \left(\phi_{i+1,j} - \phi_{i-1,j} \right) \tag{D.30}$$

$$\left. + \frac{1}{48} \left(\phi_{i,j+1}^{\dagger} - \phi_{i,j-1}^{\dagger} \right) \left(\frac{1}{k} d\hat{F} - d\hat{M} \right) \left(\phi_{i,j+1} - \phi_{i,j-1} \right) \right] \tag{D.31}$$

D.1.3 3-D Linear Interpolation

A 3-D linear interpolation follows the same process. We integrate a linearly interpolated flux $\phi(x, y, z)$ over a 3-D Cartesian volume V.

$$V\phi_i^\dagger \left(\frac{1}{k}d\hat{F} - d\hat{M}\right) \phi_i \to \int \int \int dV \, \phi^\dagger(x, y, z) \left(\frac{1}{k}d\hat{F} - d\hat{M}\right) \phi(x, y, z) \tag{D.32}$$

We can explicitly express $\phi(x, y, z)$ in terms of x, y, an d z gradients as

$$\phi(x, y, z) = Ax + By + +Cz + \phi_{i,j,k} \tag{D.33}$$

$$= \frac{\phi_{i+1,j,k} - \phi_{i-1,j,k}}{2\Delta x}x + \frac{\phi_{i,j+1,k} - \phi_{i,j-1,k}}{2\Delta y}y + \frac{\phi_{i,j,k+1} - \phi_{i,j,k-1}}{2\Delta z}z + \phi_{i,j,k} \tag{D.34}$$

We can index each Cartesian volume with a unique i, j, k index triplet. Note that $(x, y, z) = (0, 0, 0)$ corresponds to the center of the Cartesian 3-D volume so that the average flux $\phi_{i,j,k}$ is preserved. Performing the expansion, integration, and condensation of terms in the same manner as in 1-D and 2-D yields a single term identical to the constant flux implementation plus *twelve* correction terms - four for each Cartesian direction of the flux gradient. Condensing each set of four terms into a single term yields

$$\Delta x \Delta y \Delta z \left[\phi_{i,j,k}^\dagger \left(\frac{1}{k}d\hat{F} - d\hat{M}\right) \phi_{i,j,k}\right. \tag{D.35}$$

$$+\frac{1}{48}\left(\phi_{i+1,j,k}^\dagger - \phi_{i-1,j,k}^\dagger\right)\left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\left(\phi_{i+1,j,k} - \phi_{i-1,j,k}\right) \tag{D.36}$$

$$+\frac{1}{48}\left(\phi_{i,j+1,k}^\dagger - \phi_{i,j-1,k}^\dagger\right)\left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\left(\phi_{i,j+1,k} - \phi_{i,j-1,k}\right) \tag{D.37}$$

$$\left.+\frac{1}{48}\left(\phi_{i,j,k+1}^\dagger - \phi_{i,j,k-1}^\dagger\right)\left(\frac{1}{k}d\hat{F} - d\hat{M}\right)\left(\phi_{i,j,k+1} - \phi_{i,j,k-1}\right)\right] \tag{D.38}$$

D.2 2-D Assembly Shifting

Now we examine a very simple geometry distortion in 2-D, the shifting of one fuel assembly outward into a sodium blanket. We construct a 5x5 array of homogenized uranium metal square assemblies surrounded by a sodium blanket. We employ a flexible Cartesian mesh that we can adjust to be hyperfine in the vicinity of the distortion while remaining coarse elsewhere. See Figure D.3 for an illustration.

Figure D.4 shows the reactivity effects due to this 2-D assembly shifting. We use the hyperfine mesh shown in Figure D.3 to compute the actual diffusion eigenvalue solution. When we apply perturbation theory using the unperturbed fluxes from the same hyperfine mesh, we can predict the reactivity effects quite accurately up to a 20% assembly shift fraction. However, when we apply perturbation theory using unperturbed fluxes from a coarse mesh (several mesh cells per assembly), the result is poor. Linear interpolation techniques help somewhat, but not nearly enough.

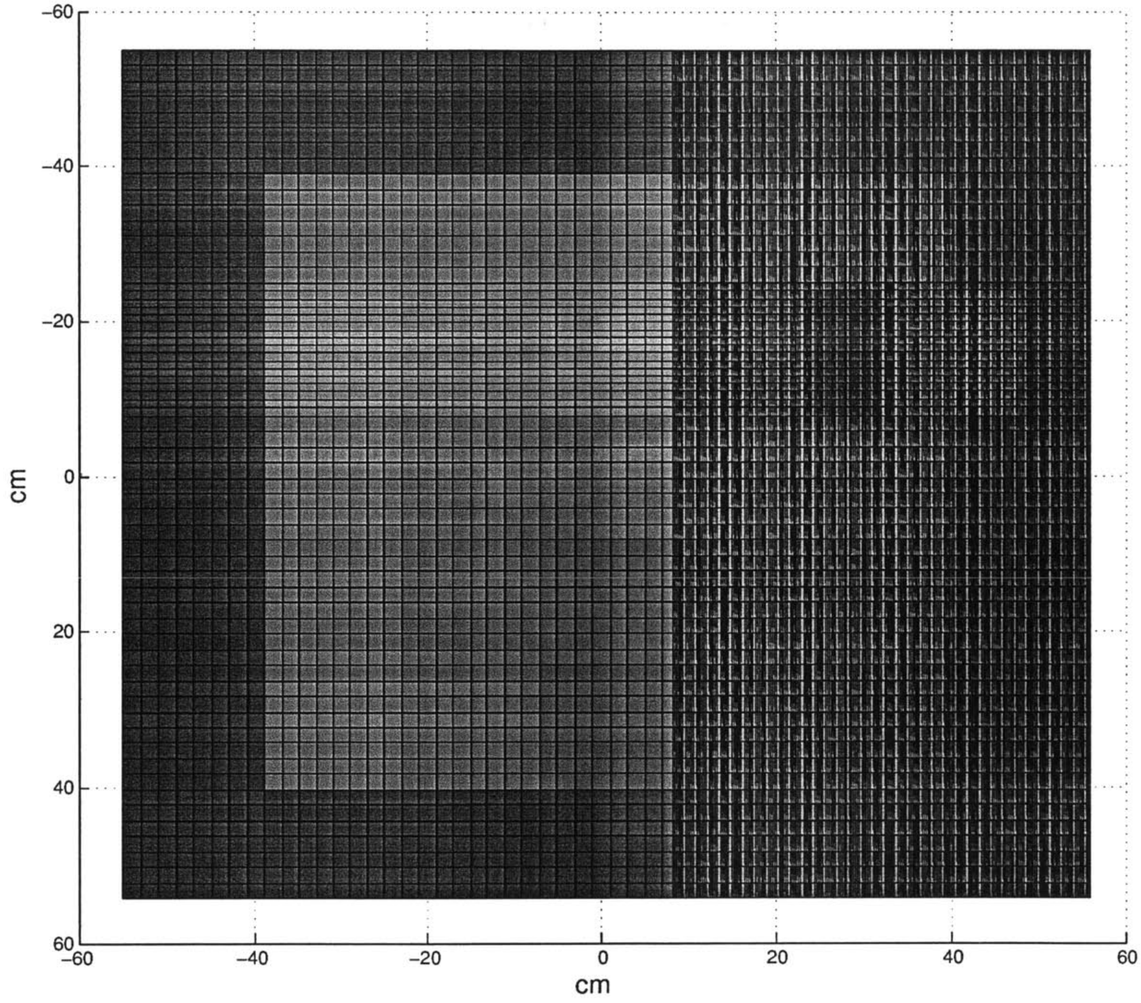

Figure D.3: The grid structure of our 2-D assembly shifting model. The red mesh cells represent an array of 5x5 homogenized assemblies, while the blue cells represent a pure sodium blanket. Slightly different shades of red represent different enrichments. Here one assembly has shifted outward by a "fractional assembly shift" of 0.5 so that it is halfway out into the blanket. Note that we have made the mesh hyperfine in the vicinity of the assembly shift perturbation in order to resolve the value of the reactivity via a diffusion eigenvalue solution.

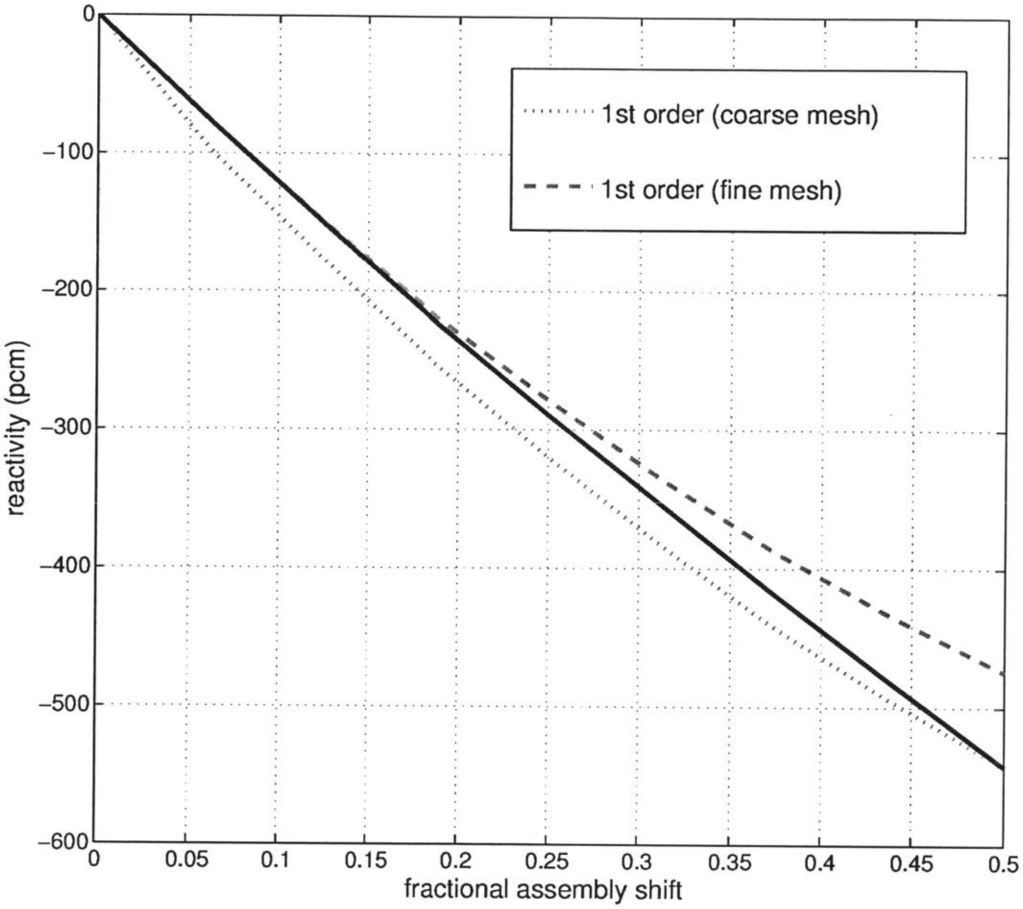

Figure D.4: Reactivity as a function of 2-D assembly shift fraction for the distortion shown in Figure D.3. The solid black line shows the diffusion eigenvalue result using the hyperfine mesh. The two red dotted lines show first order perturbation theory estimates using the same hyperfine mesh and a coarse mesh (several mesh cells per assembly) with linear flux interpolation.

E The Energy Dependence of Sodium Void Worth

We can employ standard first order perturbation theory to show the contribution of each energy group to sodium void worth. Figure E.1 shows the relative contribution of each energy group to the first order sodium density worth per mass of sodium. When sodium is added to a reactor, the neutron spectrum shifts to slightly lower energy, causing a reduction in reactivity at high energies and a simultaneous increase in reactivity at lower energies. However, the reactivity reduction at high energies outweighs the reactivity increase at lower energies, and so the net sodium worth is negative (equivalently, the net sodium *void* worth is positive).

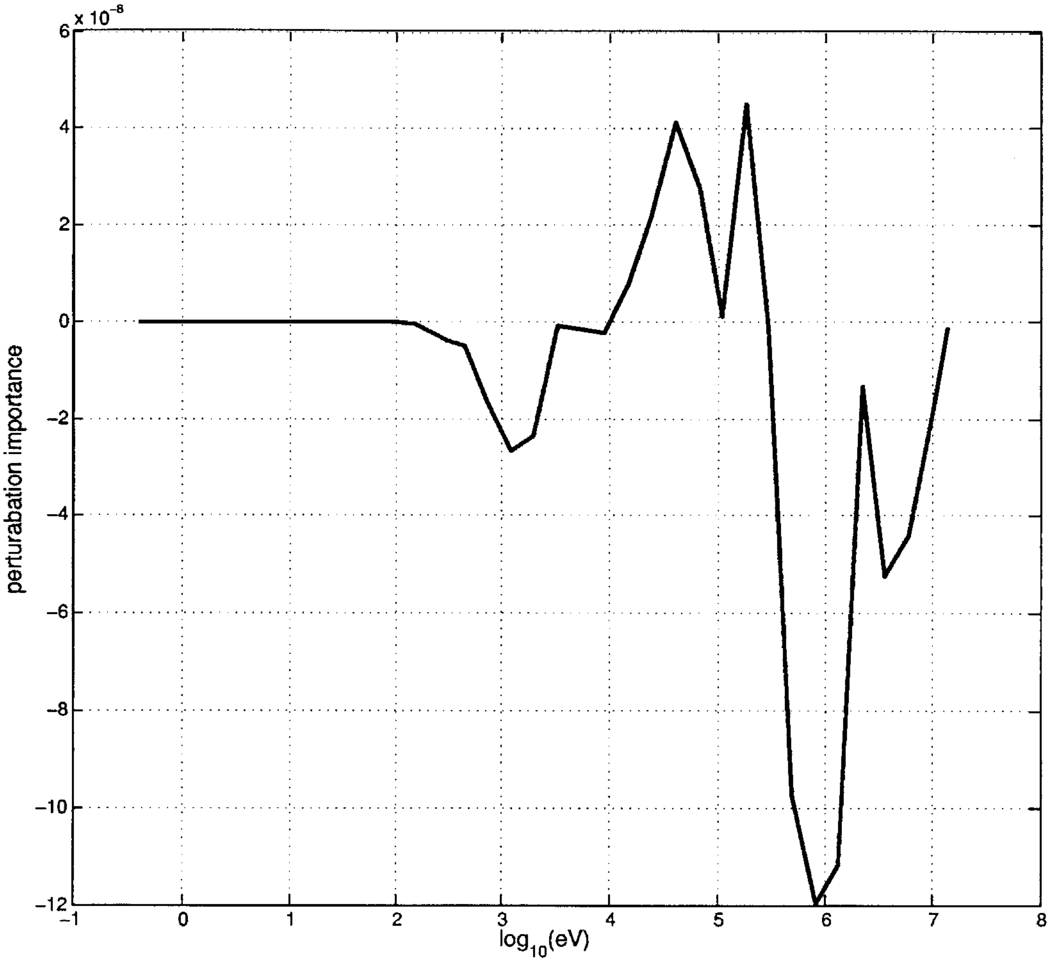

Figure E.1: The relative contribution of each energy group to the first order perturbation theory estimate of sodium void worth (per unit mass).

F Successive Approximations for Stacey's Variational Functional Applied to Material Densities

In Section 2.2.3, we discussed the variational functional derived by Stacey for evaluating second order material density perturbations [155]. This functional can correctly predict the perturbed eigenvalue to second order and the perturbed flux distribution to first order. Favorite and Stacey have successfully applied it to numerous scenarios [29].

$$\frac{\Delta\rho''}{\Delta\rho'} = 1 + \left\langle \psi^\dagger \middle| \left(\tfrac{1}{k} d\hat{F} - d\hat{M}\right)\phi \right\rangle \tag{F.1}$$

$$\left(\tfrac{1}{k}F^\dagger - M^\dagger\right)\psi^\dagger = -\frac{\left(\tfrac{1}{k}d\hat{F}^\dagger - d\hat{M}^\dagger\right)\phi^\dagger}{\left\langle \phi^\dagger \middle| \left(\tfrac{1}{k}d\hat{F} - d\hat{M}\right)\phi \right\rangle} + \frac{\hat{F}^\dagger\phi^\dagger}{\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle} \tag{F.2}$$

Note that Equation F.2 amounts to a fixed adjoint source problem - solving for the adjoint flux (left hand side) given a fixed adjoint source (right hand side). However, since this is perturbation theory, we have already solved the eigenvalue problem $\left(\frac{1}{k}F^\dagger - M^\dagger\right)\phi^\dagger = 0$ to obtain the unperturbed adjoint flux $\phi^\dagger$. Thus, solving the fixed-source problem in Equation F.2 will yield a solution for $\psi^\dagger$ that is a superposition of a *general* solution and a *particular* solution. The *general* solution represents the case of zero source, so it is proportional to $\phi^\dagger$. The *particular* solution represents the case of the source specified as in Equation F.2, and it is this *particular* solution that we seek to isolate and assign to $\psi^\dagger$. Unfortunately, disentangling this desired particular solution from the undesired general solution is not so simple - one must "sweep out" the general solution via "successive approximations" outlined by Favorite and Stacey [29].

First let us define the function $\xi_0^\dagger$, the solution to the fixed-source problem *without* the fission operator - a fission-less fixed-source problem.

$$-M^\dagger\xi_0^\dagger = -\frac{\left(\tfrac{1}{k}d\hat{F}^\dagger - d\hat{M}^\dagger\right)\phi^\dagger}{\left\langle \phi^\dagger \middle| \left(\tfrac{1}{k}d\hat{F} - d\hat{M}\right)\phi \right\rangle} + \frac{\hat{F}^\dagger\phi^\dagger}{\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle} \tag{F.3}$$

Now let us define two infinite series of functions: $\xi_n^\dagger$ and $\psi_n^\dagger$, each for $n = 0$ to ∞. Given the initial function $\xi_0^\dagger$ above, we can build up these two series recursively using Equations F.4 and F.5:

$$\psi_n^\dagger = \xi_n^\dagger - \frac{\left\langle \xi_n^\dagger \middle| F\phi \right\rangle}{\left\langle \phi^\dagger \middle| \hat{F}\phi \right\rangle}\phi^\dagger \tag{F.4}$$

$$M^\dagger\xi_n^\dagger = \frac{1}{k}\hat{F}^\dagger\psi_{n-1}^\dagger \tag{F.5}$$

The purpose of the second term in the right side of Equation F.4 is to "sweep out" the general solution. We can evaluate as many functions in these two series as we like, but we find that $n = 10$ is typically sufficient to ensure a well-converged solution.

Finally, let the generalized adjoint function $\psi^\dagger$ (the *particular* solution to the fixed-source problem) be the sum of all its components $\psi_n^\dagger$.

$$\psi^\dagger = \sum_{n=0}^{\infty} \psi_n^\dagger \tag{F.6}$$

This completes our explanation for how we isolate the *particular* solution $\psi^\dagger$ to Stacey's variational functional, which is effectively a fixed-source problem.

G The Fast Flux Test Facility (FFTF) Benchmark

The Fast Flux Test Facility (FFTF) is a deactivated sodium fast reactor (SFR) with mixed oxide (MOX) fuel. The FFTF is located on the historic Hanford site along the Columbia River near Richland, Washington. It operated as a research reactor from 1980 until 1992. In 1993, it was deactivated and gradually disassembled throughout the 1990s. In 2005, the core was drained of its sodium coolant but backfilled with argon to thwart corrosion.

See the core map in Figure G.1, and see a fuel pin axial profile in Figure G.2. The FFTF contains 11 rings of hexagonal assemblies (313 total) immersed in sodium. Two enrichment zones occupy assembly rings 1-4 and 5-6. This fueled inner core contains various primary and secondary control assemblies as well as shim assemblies, a thimble, and a vibration open test assembly. Three reflector assembly rings and two shield assembly rings surround the fuel. The flat-to-flat hexagonal assembly pitch is 12.051 cm. Each fuel assembly contains 271 pins (9 full rings), each with an outer diameter of 0.5842 cm and helically wrapped with wire. The active (fueled) core is 91.44 cm long. The interstitial sodium-filled regions (the thin spaces between neighboring assembly ducts) are 0.43 cm thick. Both cladding and duct are stainless steel (SS-316). See the formal benchmark documentation for detailed descriptions of all components, including homogenized number densities [146].

In order to model this core in finite difference diffusion, we homogenize each axial segment of each hexagonal assembly. The fine mesh benchmark has 112 axial zones, but we typically coarsen this to 28 zones for efficiency. We also “fill in” the outermost assembly ring with 18 additional shield assemblies.

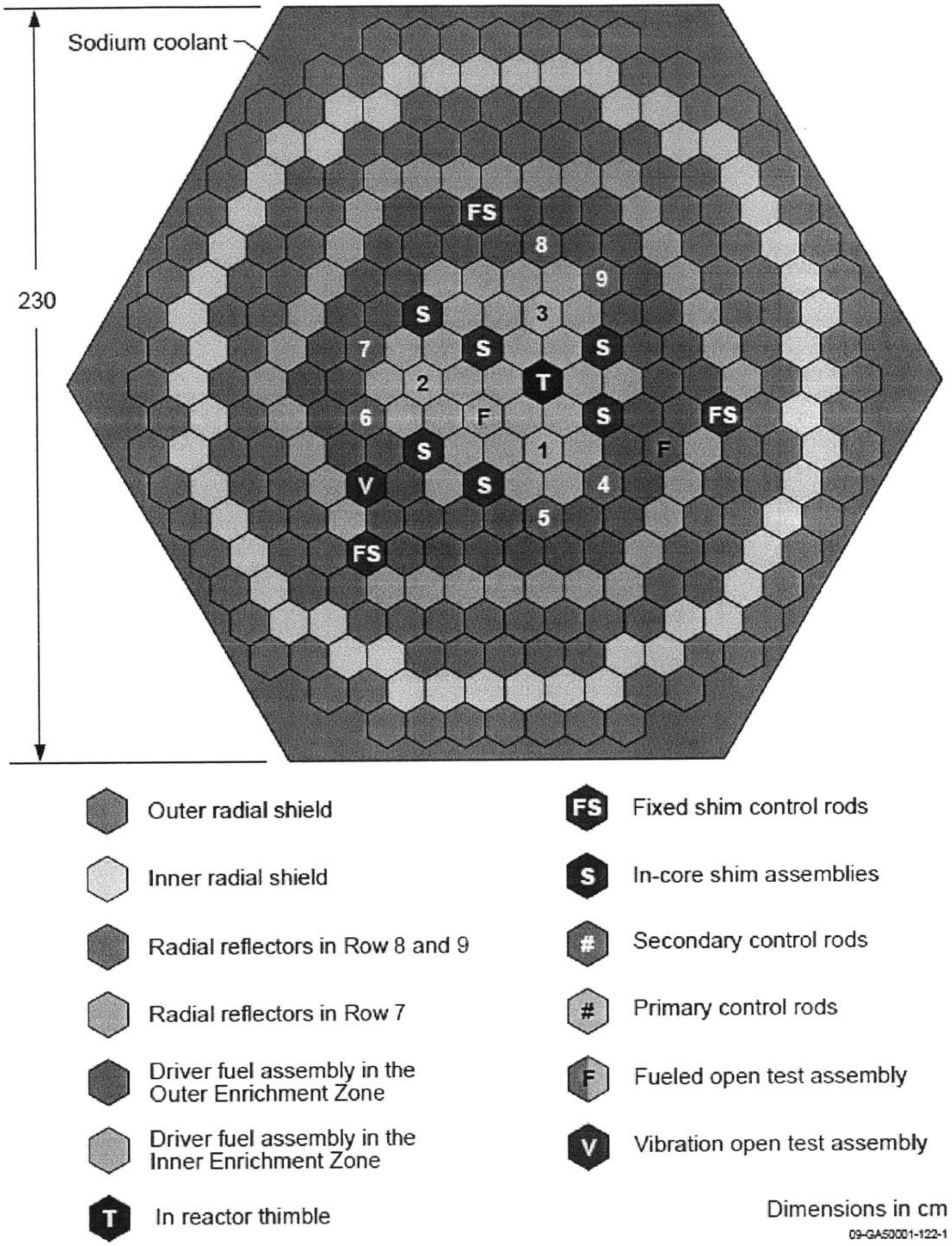
Sodium coolant
230
FS
8
9
S
3
7
S
S
2
T
6
F
S
FS
1
F
S
V
S
4
5
FS
Outer radial shield
Inner radial shield
Radial reflectors in Row 8 and 9
Radial reflectors in Row 7
Driver fuel assembly in the Outer Enrichment Zone
Driver fuel assembly in the Inner Enrichment Zone
T In reactor thimble
FS Fixed shim control rods
S In-core shim assemblies
# Secondary control rods
# Primary control rods
F Fueled open test assembly
V Vibration open test assembly
Dimensions in cm
09-GA50001-122-1

Figure G.1: The FFTF benchmark core map. This is a sodium fast reactor (SFR) with mixed oxide (MOX) fuel. Two rings of shield assemblies and three rings of reflector assemblies surround six inner rings of fuel assemblies interspersed with various control and test assemblies [146].

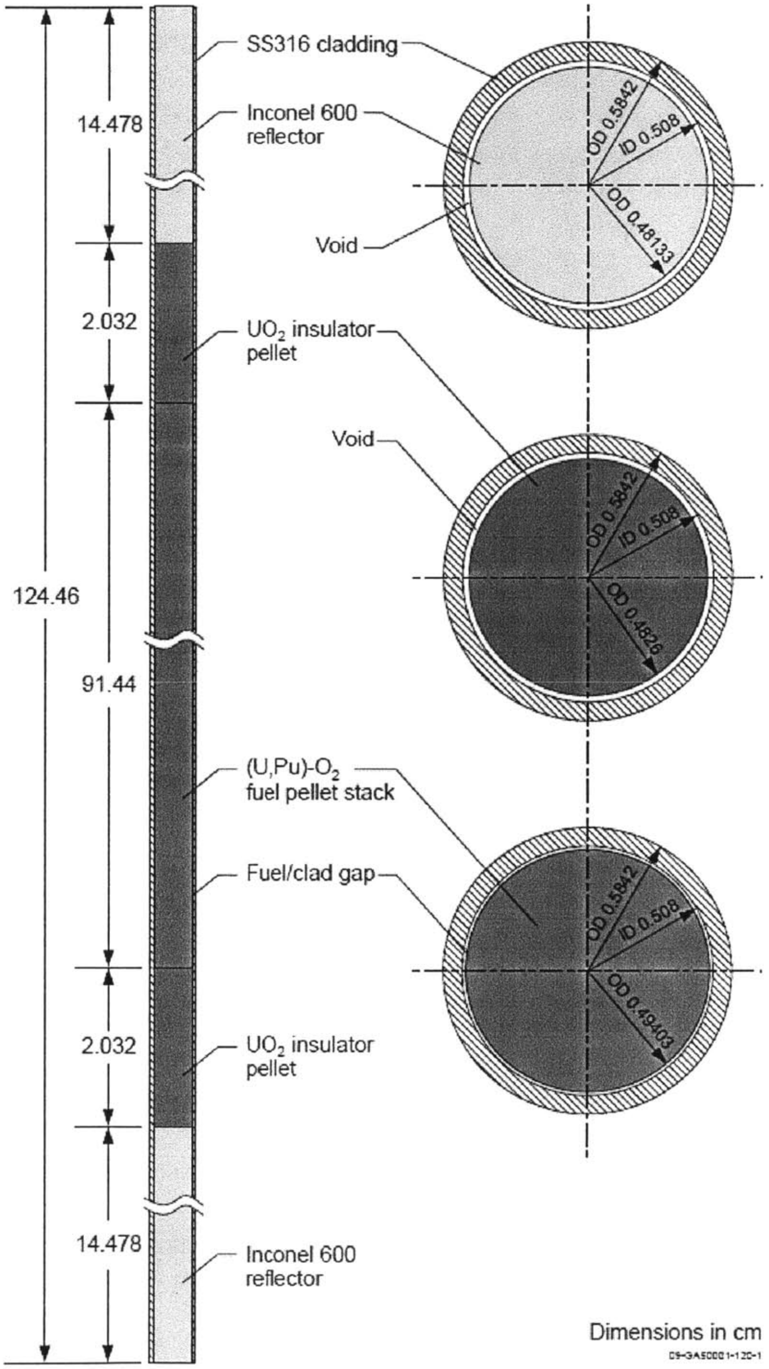

Figure G.2: The fuel pin axial profile in the FFTF benchmark. This is a sodium fast reactor (SFR) with mixed oxide (MOX) fuel [146]. All dimensions are cm.

H The Jōyō Benchmark

Jōyō is a sodium fast reactor (SFR) on the Pacific coast of Japan near Ōarai in Ibaraki Prefecture - roughly halfway between Tokyo and the Fukushima reactors. It has operated since 1977 and weathered the 2011 earthquake and tsunami without incident.

See the core map in Figure H.1, and see axial profiles of different assembly types in Figure H.2. Jōyō contains 11 rings of hexagonal assemblies (313 total) immersed in sodium. Uniformly-enriched fuel occupies assembly rings 1-5 and part of ring 6. This inner core contains 6 control assemblies and one assembly acting as a neutron source. Four or five rings of 19-pin depleted uranium assemblies surround the inner core for breeding purposes. Shield assemblies occupy the outermost ring. The flat-to-flat hexagonal assembly pitch is 8.1817 cm. Each fuel assembly contains 91 pins (6 full rings), each with an outer diameter of 0.5614 cm and helically wrapped with wire. The depleted uranium assemblies contain 19 pins, which are much bulkier at 1.38 cm. The active (fueled) core is 60 cm long. The interstitial sodium-filled regions (the thin spaces between neighboring assembly ducts) are 0.30 cm thick. Both cladding and wire wrap are stainless steel (SS-316). See the formal benchmark documentation for detailed descriptions of all components [147].

In order to model this core in finite difference diffusion, we homogenize each axial segment of each hexagonal assembly. The fine mesh benchmark has 66 axial zones, but we typically coarsen this to 22 zones for efficiency. We also "fill in" the outermost assembly ring with 18 additional shield assemblies.

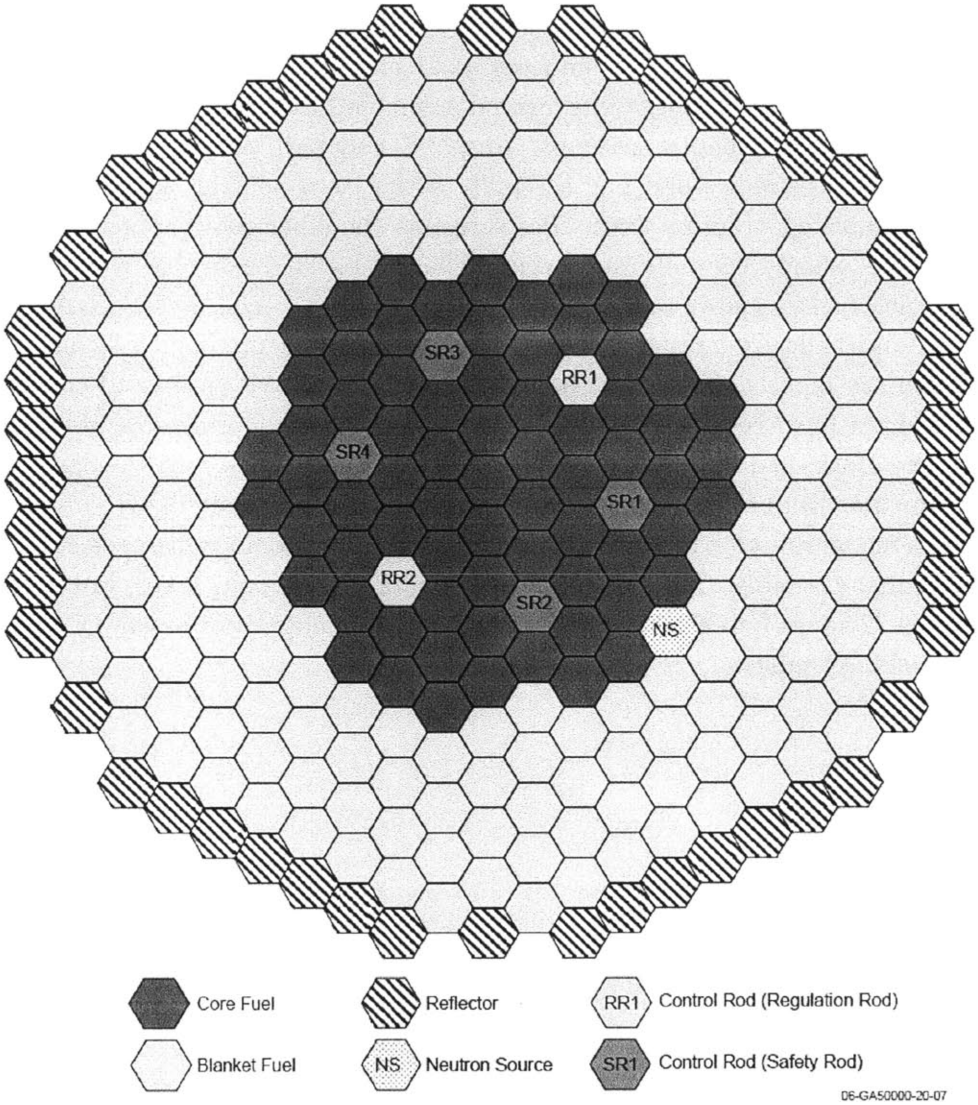

Figure H.1: The Jōyō benchmark core map. This is a sodium fast reactor (SFR) with mixed oxide (MOX) fuel. A thick blanket of depleted uranium assemblies (for breeding) surrounds an inner core region [147].

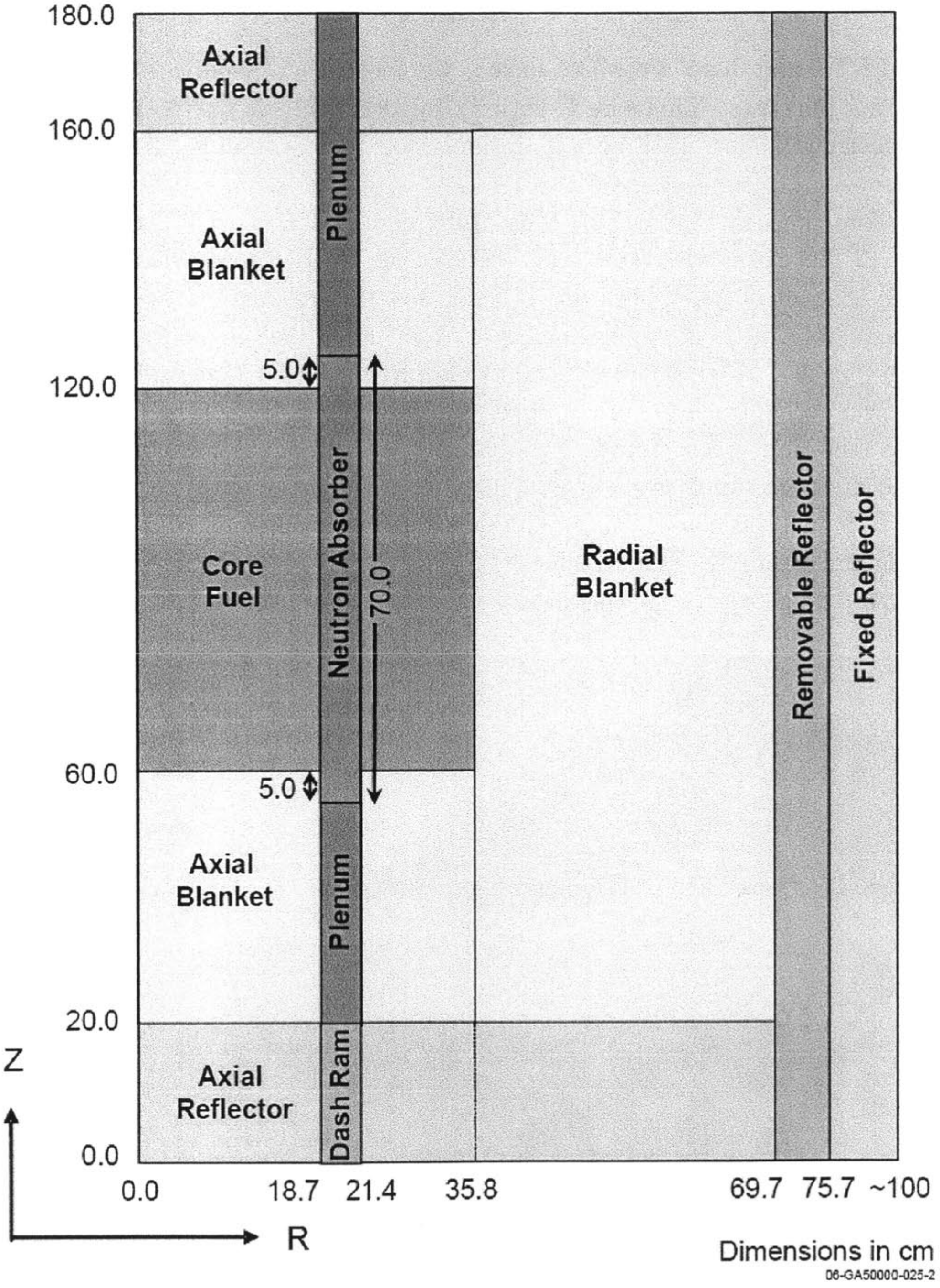

Figure H.2: Axial profiles of various assembly types in the Jōyō benchmark. This is a sodium fast reactor (SFR) with mixed oxide (MOX) fuel [147].

I The Generic Metal-Fueled (GMF) Test Core Model

Since both FFTF and Jōyō are MOX-fueled, we construct a generic metal-fueled (GMF) core for testing purposes. This core is entirely hypothetical - it has been neither built nor previously designed. This GMF core is sodium-cooled and contains enriched uranium but no plutonium.

See the core map in Figure I.1. The GMF core contains 16 full rings of hexagonal assemblies: 12 fuel rings with 2 different enrichment levels, three reflector rings, and one shield ring. 18 control assemblies are interspersed throughout the fuel. The flat-to-flat hexagonal assembly pitch is approximately 15 cm. Each fuel assembly contains 331 pins (11 full rings), each with an outer diameter of approximately 0.6 cm and helically wrapped with wire. The active (fueled) core is approximately 80 cm long. The interstitial sodium-filled regions (the thin spaces between neighboring assembly ducts) are 0.59 cm thick. Both cladding and duct are stainless steel (SS-316).

In order to model this core in finite difference diffusion, we homogenize each axial segment of each hexagonal assembly. The axial mesh benchmark has 15 axial zones. Due to symmetry, we choose 1/3 core geometry for this model.

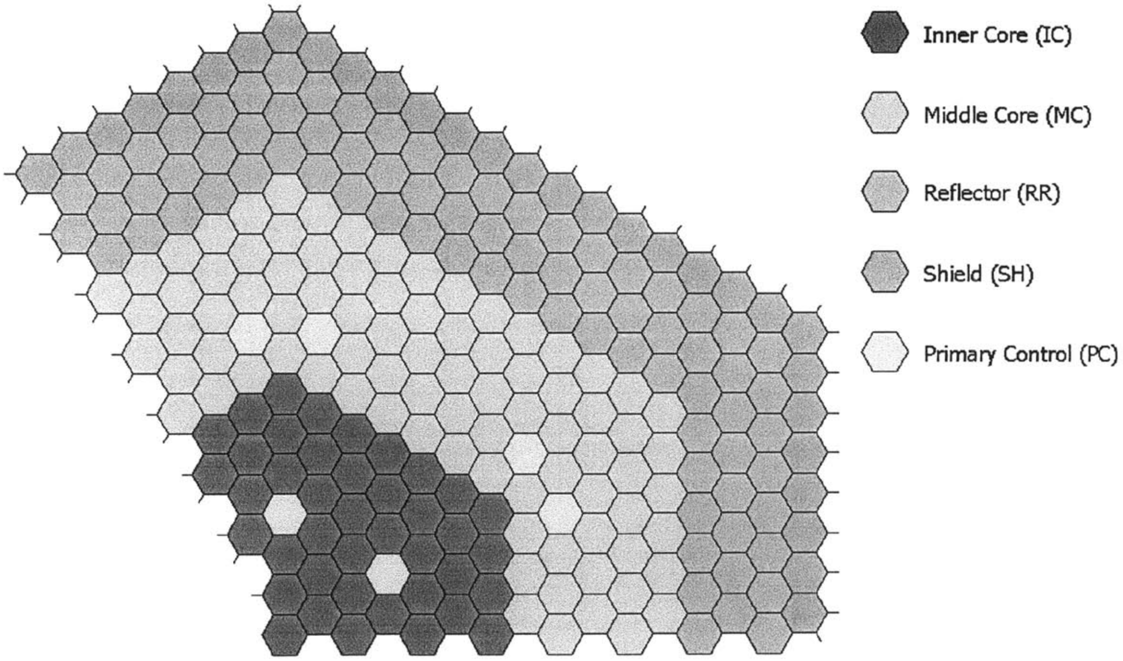

Figure I.1: The generic metal-fueled (GMF) test core map. This is a sodium fast reactor (SFR) with enriched uranium metal fuel. Shield and reflector assemblies encase a large inner core region.

References

Generalized Perturbation Theory

[1] P. Roussopoulos. "Méthodes Variationnelles en Théorie des Collisions". *C. R. Acad. Sci. Paris* 236: 1858-1860 (1953).

[2] N. C. Francis, J. C. Stewart, L. S. Bohl, T. J. Krieger. *Variational Solutions of the Transport Equation.* A/CONF, 15/P/627. Knolls Atomic Power Laboratory, Schenectady, NY (1958).

[3] L. N. Usachev. "Perturbation Theory for the Breeding Ratio and Other Number Ratios Pertaining to Various Reactor Processes". *Journal of Nuclear Energy, Parts A/B: Reactor Science and Technology* 18(10): 571-583 (1964).

[4] J. Lewins. *Importance: The Adjoint Function.* Pergamon Press, New York (1965).

[5] G. C. Pomraning. "A Variational Principle for Linear Systems". *Journal of the Society for Industrial and Applied Mathematics* 13(2): 511-519 (1965).

[6] J. Lewins. "A Variational Principle for Ratios in Critical Systems". *Journal of Nuclear Energy, Part A/B: Reactor Science and Technology* 20(2): 141-143 (1966).

[7] G. C. Pomraning. "A Variational Description of Dissipative Processes". *Journal of Nuclear Energy, Part A/B: Reactor Science and Technology* 20(8): 617-634 (1966).

[8] G. C. Pomraning. "Variational Principles for Eigenvalue Equations". *Journal of Mathematical Physics* 8: 149 (1967).

[9] G. C. Pomraning. "A Derivation of Variational Principles for Inhomogenous Equations". *Nuclear Science and Engineering* 29: 220 (1967).

[10] A. Gandini. "A Generalized Perturbation Method for Bilinear Functionals of the Real and Adjoint Neutron Fluxes". *Journal of Nuclear Energy* 21(10): 755-765 (1967).

[11] G. C. Pomraning. "The Calculation of Ratios in Critical Systems". *Journal of Nuclear Energy* 21(3): 285-291 (1967).

[12] G. C. Pomraning. "Generalized Variational Principles for Reactor Analysis". No. CNM-R-2 (vol. 1), Brookhaven National Laboratory, Upton, NY (1969).

[13] G. I. Bell, S. Glasstone. *Nuclear Reactor Theory*, Chap. 6, Van Nostrand Reinhold, New York (1970).

[14] J. Devooght. "Higher-Order Variational Principles and Iterative Processes". *Nuclear Science and Engineering* 41: 399 (1970).

[15] W. M. Stacey. "Variational Estimates of Reactivity Worths and Reaction Rate Ratios in Critical Systems". *Nuclear Science and Engineering* 48: 444 (1972).

[16] W. M. Stacey. "Variational Estimates and Generalized Perturbation Theory for Ratios of Linear and Bilinear Functionals". *Journal of Mathematical Physics* 13: 1119 (1972).

[17] W. M. Stacey. "An Improved Reactivity Table Model for Liquid Metal Fast Breeder Reactor Dynamics". *Nuclear Science and Engineering* 49: 213 (1972).

[18] H. Henryson, et. al. "Variational Sensitivity Analysis - Theory and Application". Proceedings of the ANS Topical Meeting on Advanced Reactors; Physics, Design and Economics, Atlanta, GA, September 8-11 (1974).

[19] M. Weston. *Variational Methods in Nuclear Reactor Physics* vol. 10, Academic Press (1974).

[20] A. Gandini. "Time-Dependent Generalized Perturbation Methods for Burnup Analysis". RT/FI 75: 4, Comitato Nazionale per l'Energia Nucleare, Roma (1975).

[21] E. Greenspan. "Developments in Perturbation Theory". *Advances in Nuclear Science and Technology* 9: 181-268 (1976).

[22] D. G. Cacuci, C. F. Weber, E. M. Oblow, J. H. Marable. "Sensitivity Theory for General Systems of Nonlinear Equations". *Nuclear Science and Engineering* 75: 88-110 (1980).

[23] D. G. Cacuci. "Sensitivity Theory for Nonlinear Systems". *Journal of Mathematical Physics* 22: 2794-2803 (1981).

[24] T. A. Taiwo, A. F. Henry. "Perturbation Theory Based on a Nodal Model". *Nuclear Science and Engineering* 92(1): 34 (1986).

[25] M. L. Williams. "Perturbation Theory for Nuclear Reactor Applications". *CRC Handbook of Nuclear Reactor Calculations* 3: 63-188 (1986).

[26] A. Gandini. "Generalized Perturbation Theory (GPT) Methods: A Heuristic Approach". *Advances in Nuclear Science and Technology* 19: 205-380 (1987).

[27] W. S. Yang, T. J. Downar. "Generalized Perturbation Theory for Constant Power Core Depletion". *Nuclear Science and Engineering* 99: 353-366 (1988).

[28] T. J. Downar. "Depletion Perturbation Theory for Burnup Dependent Microscopic Cross-Sections". *Annals of Nuclear Energy* 19(1): 27-37 (1992).

[29] J. A. Favorite, W. M. Stacey. "Variational Estimates of Point-Kinetics Parameters". *Nuclear Science and Engineering* 121(3): 353-370 (1995).

[30] K. F. Laurin-Kovitz, E. E. Lewis. "Solution of the Mathematical Adjoint Equations for an Interface Current Nodal Formulation". *Nuclear Science and Engineering* 123: 369 (1996).

[31] W. M. Stacey, J. A. Favorite. "Variational Reactivity Estimates". Joint International Conference on Mathematical Methods and Supercomputing for Nuclear Applications I, American Nuclear Society, La Grange Park, Illinois (1997).

[32] S. Pelloni, P. Coddington. "On Using Alternative Perturbation Theory Methodologies in the Evaluation of Reactivity Effects in ADS Cores". *Annals of Nuclear Energy* 33(17): 1360-1367 (2006).

[33] J. A. Favorite. "Variational Reactivity Estimates: New Analyses and New Results". International Conference on Advances in Mathematics, Computational Methods, and Reactor Physics. Saratoga Springs, NY, May 3-7 (2010).

Modal Expansion Perturbation Theory (MEPT)

[34] H. Mitani. "Higher Order Perturbation Methods in Reactor Calculations". *Nuclear Science and Engineering* 51(2): 180-188 (1973).

[35] A. Gandini. "Implicit and Explicit Higher Order Perturbation Methods for Nuclear Reactor Analysis". *Nuclear Science and Engineering* 67(3): 347 (1978).

[36] G. Palmiotti. "Use of the Explicit High-Order Perturbation Formulation". *Nuclear Science and Engineering* 83(2): 281 (1983).

[37] J. Moreira, J. C. Lee. "Space-Time Analysis of Reactor Control Rod Worth Measurements". *Nuclear Science and Engineering* 86: 91-105 (1984).

[38] J. Moreira, J. C. Lee. "Accuracy of the Modal-Local Method for Reactivity Determination". *Nuclear Science and Engineering* 98(3): 244 (1988).

[39] N. Touran, J. C. Lee. "A Modal Expansion Equilibrium Cycle Perturbation Method for Optimizing High Burnup Fast Reactors". Doctoral Thesis, Department of Nuclear Engineering and Radiological Sciences, University of Michigan (2012).

[40] N. Touran, J. C. Lee. "A Modal Expansion Equilibrium Cycle Perturbation Method for Optimizing Fast Reactors". *Transactions of the American Nuclear Society* 109(1): 793-795 (2013).

Geometric Perturbation Theory

[41] M. Komata. "Generalized Perturbation Theory Applicable to Reactor Boundary Changes". *Nuclear Science and Engineering* 64: 811 (1977).

[42] E. W. Larsen, G. C. Pomraning. "Boundary Perturbation Theory". *Nuclear Science and Engineering* 77: 415-425 (1981).

[43] F. Rahnema, G. C. Pomraning. "An Anomaly in the Use of Perturbation Theory". *Nuclear Science and Engineering* 78: 393-437 (1981).

[44] F. Rahnema, S. A. Ahmad, W. E. Kastenberg, G. C. Pomraning. "Recriticality Considerations in Disrupted Gas Cooled Fast Reactor Core Geometry". *Nuclear Technology* 59: 246-255 (1982).

[45] G. C. Pomraning. "Perturbation Theory in the Diffusion Approximation". *Nuclear Science and Engineering* 83: 72-74 (1983).

[46] F. Rahnema, G. C. Pomraning. "Boundary Perturbation Theory for Inhomogenous Transport Equations". *Nuclear Science and Engineering* 84: 313-319 (1983).

[47] G. C. Pomraning. "Boundary Perturbation Theory for Nonanalytic Perturbations". *Nuclear Science and Engineering* 85: 188-196 (1983).

[48] F. Rahnema, G. C. Pomraning. "External boundary perturbations in neutron transport theory". *Journal of Mathematical Physics* 24: 687-701 (1983).

[49] F. Rahnema. "Internal Interface Perturbations in Neutron Transport Theory". *Nuclear Science and Engineering* 86: 76-90 (1984).

[50] F. Rahnema. "On the Internal Surface Perturbation". *Annals of Nuclear Energy* 23: 1401-1406 (1996).

[51] H. N. M. Gheorghiu, F. Rahnema. "Generalized Rayleigh Quotients for Eigenvalue Estimates of Neutron Balance Equations". *Nuclear Science and Engineering* 125: 314-323 (1997).

[52] H. N. M. Gheorghiu, F. Rahnema. "Variational Principles for Steady-State Neutron Flux Functionals". *Transport Theory and Statistical Physics* 27: 67-95 (1998).

[53] F. Rahnema, P. Ravetto. "On the Equivalence of Boundary and Boundary Condition Perturbations in Transport Theory". *Nuclear Science and Engineering* 128: 209-223 (1998).

[54] M. S. McKinley, F. Rahnema. "High-Order Boundary Condition Perturbation Theory for the Diffusion Approximation". *Nuclear Science and Engineering* 136: 15-33 (2000).

[55] M. S. McKinley, F. Rahnema. "High-Order Boundary Condition Perturbation Theory for the Neutron Transport Equation". Lawrence Livermore National Laboratory, UCRL-ID-155132 (2001).

[56] D. Ilas, F. Rahnema. "A Heterogeneous Coarse Mesh Transport Method". *Transport Theory and Statistical Physics* 32:445-471 (2003).

[57] S. W. Mosher, F. Rahnema. "An Intra-Nodal Flux Expansion for a Heterogeneous Coarse Mesh Discrete Ordinates Method". Nuclear Mathematical and Computational Sciences, Gatlinburg, TN, April 6-11 (2003).

[58] S. W. Mosher, F. Rahnema. "The Incident Flux Response Expansion Method for Heterogeneous Coarse Mesh Transport Problems". *Transport Theory and Statistical Physics* 35:55-86 (2006).

[59] J. A. Favorite. "Variational Estimates of Internal Interface Perturbations and a New Variational Functional for Inhomogeneous Transport Problems". *Nuclear Science and Engineering* 152: 180-196 (2006).

[60] J. A. Favorite. "Variational Estimates of Neutron-Induced Gamma Line Leakages and Ratios for Internal Interface Perturbations". *Nuclear Science and Engineering* 155: 321-329 (2007).

[61] J. A. Favorite, K. C. Bledsoe. "Eigenvalue Sensitivity to System Dimensions". *Annals of Nuclear Energy* 37: 522-528 (2010).

[62] J. A. Favorite. "Adjoint-Based Eigenvalue Sensitivity to Geometry Perturbations". Transactions of the American Nuclear Society 102: 240-243 (2010).

[63] J. A. Favorite. "Adjoint-Based Eigenvalue Sensitivity to Geometry Perturbations, and a Warning". Transactions of the American Nuclear Society 103: 393-395 (2010).

[64] J. A. Favorite. "Revisiting Boundary Perturbation Theory for Inhomogeneous Transport Equations". Transactions of the American Nuclear Society 106: 433-436 (2012).

[65] J. A. Favorite, J. C. Armstrong, P. A. Vaquer. "Solving Inverse Detection Problems Using Passive Radiation Signatures". LA-UR-12-24142, Los Alamos National Laboratory (2012).

[66] J. A. Favorite. "Nonspherical Perturbations of Spherical Geometries in Transport Theory". *Nuclear Science and Engineering* 175: 1-26 (2013).

Anisotropic Diffusion

[67] D. J. Behrens. "The Effect of Holes in a Reacting Material on the Passage of Neutrons". *Proc. Phys. Soc.* 62: 607 (1949).

[68] D. J. Behrens. "The Effect of Holes in a Reacting Material on the Passage of Neutrons, with Special Reference to the Critical Dimensions of a Reactor". A.E.R.E. T/R 103 (1954).

[69] P. Benoist. "A General Formalism of the Diffusion Coefficient in a Heterogeneous Medium Which May Contain Cavities". S.P.M. 522 (1959).

[70] C. Carter. "Streaming due to holes in a reactor". *Journal of Nuclear Energy, Parts A/B: Reactor Science and Technology* 15: 76-80 (1961).

[71] P. Benoist. "Streaming effects and collision probabilities in lattices". *Nuclear Science and Engineering* 34: 285-307 (1968).

[72] M. Nakagawa, H. Takano, S. Katsuragi. "Analysis of Sodium Void Reactivity Measured in FCA B-1 Assembly". *Nuclear Science and Technology* 10: 419-427 (1973).

[73] D. C. Wade, E. M. Gelbard. "Neutron Streaming in Plate Criticals". American Nuclear Society Topical Meeting on "Advanced Reactors; Physics, Design, and Economics", Atlanta, Georgia, September 8-11 (1974).

[74] T. Yoshida, S. Iijima. "numeric Study on Applicability of Benoist's Diffusion Coefficient to Sodium Void Reactivity Analysis". *Journal of Nuclear Science and Technology* 13: 464-467 (1976).

[75] T. Yoshida. "Improved Treatment of the Neutron Streaming Through Control Rod Followers in a Sodium-Cooled Fast Reactor".*Nuclear Science and Engineering* 72: 361-365 (1979).

[76] J. L. Rowlands, C. R. Eaton. "Effective Diffusion Coefficients for Low Density Cylindrical Channels". *Nuclear Science and Engineering* 76: 336 (1980).

[77] P. Benoist. "A simple model for the calculation of the sodium-voiding effects on neutron leakages in a fast reactor lattice, I. Formalism". *Nuclear Science and Engineering* 86: 22-40 (1984).

[78] P. Benoist. "A simple model for the calculation of the sodium-voiding effects on neutron leakages in a fast reactor lattice, II. Numeric Results for Hexagonal Lattices". *Nuclear Science and Engineering* 86: 41-46 (1984).

[79] S. M. Lee, S. Aoki, T. Takeda. "Diffusion Calculation Model for Streaming Effects in Low Density Channels in LMFBRs". *Journal of Nuclear Science and Technology* 26: 712-720 (1989).

[80] D. Mathews, V. Thibulsky, R. Chawla. "Anisotropic Diffusion Effects in Deterministic Pebble-Bed Lattices". Transactions of the American Nuclear Society 68: 438 (1993).

[81] I. Petrovic, P. Benoist. "B_N Theory: Advances and New models for Neutron Leakage Calculation". *Advances in Nuclear Science and Technology* 24: 223-282 (2002).

[82] T. S. Poveschenko, N. I. Laletin. "About Calculation of Axial Diffusion Coefficient in Nuclear reactor Cells". Proceedings of the International Conference PHYSOR, Chicago, Illinois, April 25-29 (2004).

[83] E. W. Larsen, T. J. Trahan. "2-D Anisotropic Diffusion in Optically Thin Channels". Transactions of the American Nuclear Society 101: 387 (2009).

[84] T. J. Trahan, E. W. Larsen. "3-D Anisotropic Neutron Diffusion in Optically Thick Media with Optically Thin Channels". Transactions of the International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering (M&C 2011), Rio de Janiero, Brazil, May 8-12 (2011).

[85] W. F. G. Van Rooijen, G. Chiba. "Diffusion coefficients for LMFBR cells calculated with MOC and Monte Carlo methods". *Annals of Nuclear Energy* 38: 133-144 (2011).

The "Virtual Density" Theory

[86] H. Serber. "The Los Alamos Primer". Los Alamos Scientific Library Report LA-1 (1943).

[87] S. B. Shikhov. "Calculating the effect of a change in dimensions on the critical mass of a fast reactor using the perturbation theory". *Atomic Energy* 6(2): 162-168 (1958).

[88] S. B. Shikhov. "The Effect of Dimensional Changes on the Critical Mass of a Fast Reactor, Calculated by Perturbation Theory". *Journal of Nuclear Energy, Part A: Reactor Science* 11(2): 214-219 (1960).

[89] W. R. Stratton. "Criticality Data and Factors Affecting Criticality of Single Homogenous Units". Los Alamos Scientific Library Report LA-3612 (1967).

[90] S. B. Shikhov, S. M. Shmelev. "Taking account of the influence of an arbitrary change in the dimensions on the critical mass of a fast reactor using perturbation theory". *Physics of Nuclear Reactors*, Atomizdat, Moscow, p. 67-85 (1968).

[91] V. V. Orlov, V. I. Matveev, A. V. Zhukov, A. I. Ivanov, G. M. Ishakin. "The Main Principles of Selection of the Methods of Regulation of Fast Reactors and Temperature and Power Effects in Fast Reactors". *Kernergie*, Heft, Berlin, Deutsche Demokratische Republik (DDR), p. 112-124 (1969).

[92] S. B. Shikhov, V. B. Troyanski. *Nuclear Reactor Theory*, Chap. 12: "Application of Small Perturbation Theory to the Reactor Problems". Moscow, Energoatomizdat (1983).

[93] Y. V. Petrov, E. G. Sakhnovsky. "On the Boundary Perturbation Theory as Applied to Nuclear Reactors". *Nuclear Science and Engineering* 90: 1-12 (1985).

[94] B. D. Abramov. "Calculation of Reactivity Effects in Deformed Zones of Nuclear Reactors". *Atomic Energy* 85(6): 853-859 (1998).

[95] "Transient and Accident Analysis of a BN-800 Type LMFR with Near Zero Void Effect". IAEA-TECDOC-1139, p. 15-16, International Atomic Energy Agency (2000).

[96] D. E. Cullen. "Mass and Density, Criticality Relationships". Lawrence Livermore National Laboratory, UCRL-ID-143496 (2001).

[97] D. E. Cullen. "Mass and Density, Criticality Relationships, Generalized". Lawrence Livermore National Laboratory, UCRL-TR-204988 (2004).

[98] M. Reed, K. Smith, B. Forget. "The 'Virtual Density' Principle of Neutronics and Its Application to Geometric Perturbation Theory". Transactions of the 2012 ANS Winter Meeting and Nuclear Technology Expo in San Diego, CA (2012).

[99] M. Reed, K. Smith, B. Forget. "The 'Virtual Density' Principle of Neutronics: Applying Geometric Perturbation Theory to Practical Core Distortion Scenarios". Transactions of the International Topical Meeting on Mathematics and Computation (MC) Applied to Nuclear Science and Engineering in Sun Valley, ID (2013).

Fast Reactor Distortion Models and Empirical Data

[100] H. F. Donahue, R. W. Keaton. "Fuel Rod Bowing in the SRE". Atomics International Division of North American Aviation, Inc., NAA-SR-6878 (1962).

[101] B. J. Knutson, D. M. Lucoff, R. A. Harris, S. L. Hecht. "Reactivity Analysis of Core Distortion Effects in FFTF". American Nuclear Society. Miami Beach, Florida, June 7-12 (1981).

[102] B. J. Knutson. "Reactivity Analysis of Core Distortion Effects in FFTF". Master's Thesis in Nuclear Engineering, University of Washington (1982).

[103] S. A. Kamal, Y. Orechwa. "Transient Bowing of Core Assemblies in Advanced Liquid Metal Fast Reactors". Transactions of the American Nuclear Society in Reno, NV, June 15 (1986).

[104] S. A. Kamal, Y. Orechwa. "Bowing of Core Assemblies in Advanced Liquid Metal Fast Reactors". ASME/ANS Bi-Annual Nuclear Power Conference, Philadelphia, PA, July 20-23 (1986).

[105] R. A. Wigeland. "Effect of a Detailed Radial Core Expansion Reactivity Feedback Model on ATWS Calculations Using SASSYS/SAS4A". Transactions of the American Nuclear Society 53: 303 A (1986).

[106] R. A. Wigeland. "Comparison of the SASSYS/SAS4A Radial Core Expansion Reactivity Feedback Model and the Empirical Correlation for the FFTF". Transactions of the American Nuclear Society in Los Angeles, CA (1987).

[107] R. A. Wigeland, T. J. Moran. "Radial Core Expansion Reactivity Feedback in Advanced LMRs: Uncertainties and their Effects on Inherent Safety". Argonne National Laboratory, CONF-8804155-1 (1988).

[108] R. A. Wigeland, D. J. Hill. "Validation of the SASSYS Core Radial Expansion Reactivity Feedback Model". Transactions of the American Nuclear Society, San Diego, CA (1988).

[109] R. W. Schaefer. "Critical Experiment Tests of Bowing and Expansion Reactivity Calculations for Liquid Metal Cooled Fast Reactors".*Nuclear Science and Engineering* 103(2): 196-209 (1989).

[110] B. Fontaine, G. Prulhiere, A. Vasile, P. Masoni, P. Barret, D. Rochwerger, J. Gros, R. Dupraz, N. Moussallam, M. Chassignet. "Description and Preliminary Results of PHENIX Core Flowering Test". *Nuclear Engineering and Design* 241: 4143-4151 (2011).

[111] K. Mikityuk, S. Pelloni, P. Coddington, E. Bubelis, R. Chawla. "FAST: An Advanced Code System for Fast Reactor Transient Analysis". *Annals of Nuclear Energy* 32: 1613-1631 (2005).

Pin Power Reconstruction

[112] K. R. Rempe, K. S. Smith, A. F. Henry. "SIMULATE-3 Pin Power Reconstruction: Methodology and Benchmarking". *Nuclear Science and Engineering* 103: 334-342 (1989).

[113] W. S. Yang, P. J. Finck, H. S. Khalil. "Reconstruction of Pin Burnup Characteristics from Nodal Calculations in Hexagonal Geometry". International conference of the physics of reactors: operation, design, and computation. Marseilles, France, April 23-26 (1990).

[114] W. S. Yang, P. J. Finck, H. S. Khalil. "Reconstruction of Pin Power and Burnup Characteristics from Nodal Calculations in Hexagonal-Z Geometry". *Nuclear Science and Engineering* 111: 21-33 (1992).

[115] H. Zhang, Rizwan-Uddin, J. Dorning. "Systematic homogenization and self-consistent flux and pin power reconstruction for nodal diffusion methods. I: Diffusion equation-based theory". *Nuclear Science and Engineering* 121: 226-244 (1995).

[116] H. S. Lee, W. S. Yang. "Comparison of Interpolation Methods for Reconstructing Pin-wise Power Distribution in Hexagonal Geometry". *Journal of the Korean Nuclear Society* 31: No. 3: 303-13 (1999).

[117] M. Reed, N. Touran, K. Smith, B. Forget. "Pin-Level Reconstruction of Various Neutronic Quantities in Fast Reactors: Enhanced Physical Insight via Visualization Tools". Transactions of the 2013 ANS Annual Meeting in Atlanta, GA (2013).

Reactor Design

[118] S. M. Feinberg. "Discussion Comment". Rec. of Proc. Session B-10, ICPUAE, United Nations, Geneva, Switzerland (1958).

[119] L. M. Lidsky. "Fission-Fusion Systems: Hybrid, Symbiotic, and Augean". *Nuclear Fusion* 15:151-173 (1975).

[120] M. Sakagami, S. Takeda. "Spatial Mesh Effects in Three-Dimensional Diffusion Calculations for Fast Reactors" *Nuclear Science and Technology* 15:5, 365-375 (1978).

[121] M. J. Driscoll, B. Atefi, D. D. Lanning. "An Evaluation of the Breed/Burn Fast Reactor Concept". MITNE-229 (1979).

[122] L. M. Lidsky. "The Trouble with Fusion". MIT Technology Review (1983).

[123] H. P. Planchon, J. I. Sackett, G. H. Golden, R. H. Sevy. "Implications of the EBR-II Inherent Safety Demonstration Test". *Nuclear Engineering and Design* 101: 75-90 (1987).

[124] T. Kamei, M. Yamaoka. "An Axially Multilayered Low Void Worth Liquid-Metal Fast Breeder Reactor Core Concept". *Nuclear Technology* 97: 264-271 (1992).

[125] Y. Oka, T. Jevremovic. "Negative Coolant Void Reactivity in Large Fast Breeder Reactors with Hydrogenous Moderator Layer". *Annals of Nuclear Energy* 23: 1105-1115 (1996).

[126] C. E. Till, Y. I. Chang, W. H. Hannum. "The Integral Fast Reactor - An Overview". *Progress in Nuclear Energy* 31: 3-11 (1997).

[127] G. L. Hofman, L. C. Walters, T. H. Bauer. "Metallic Fast Reactor Fuels". *Progress in Nuclear Energy* 31: 83-110 (1997).

[128] K. Tsujimoto, T. Iwasaki, N. Hirakawa, T. Osugi, S. Okajima, M. Andoh. "Improvement of Reactivity Coefficients of Metallic Fuel LMFBR by Adding Moderating Material". *Annals of Nuclear Energy* 28: 831-855 (2001).

[129] M. J. Driscoll, P. Hejzlar. "Reactor Physics Challenges in Gen-IV Reactor Design". *Nuclear Engineering and Technology* 37 (2005).

[130] C. W. Forsberg. "Molten Salt Reactor Technology Gaps". Proceedings of ICAPP, Reno, NV (2006).

[131] J. Gilleland, C. Ahlfeld, D. Dadiomov, R. Hyde, Y. Ishikawa, D. McAlees, J. McWhirter, N. Myhrvold, J. Nuckolls, A. Odedra, K. Weaver, C. Whitmer, L. Wood, G. Zimmerman. "Novel Reactor Designs to Burn Non-Fissile Fuel". Proceedings of the 2008 International Congress on Advances in Nuclear Power Plants (ICAPP). American Nuclear Society, Anaheim, California. Paper 8319 (2008).

[132] N. Todreas, P. Hejzlar, A. Nikiforova, R. Petroski, E. Shwageraus, C. Fong, M. Driscoll, M. Elliott, G. Apostolakis. "Flexible Conversion Ratio Fast Reactors: Overview". *Nuclear Engineering and Design* 239: 2582-2595 (2009).

[133] L. Wood, T. Ellis, N. Myhrvold, R. Petroski. "Exploring the Italian Navigator's New World: Toward Economic, Full-Scale, Low Carbon, Conveniently-Available, Proliferation-Robust, Renewable Energy Resources". 2nd Session of the Erice International Seminars on Planetary Emergencies, Erice, Italy, 19024 (2009).

[134] J. Freidberg, et. al. "Research Needs for Fusion-Fission Hybrid Systems". Report of the Research Needs Workshop (ReNeW), Gaithersburg, MD (2009).

[135] T. Ellis, R. Petroski, P. Hejzlar, G. Zimmerman, D. McAlees, C. Whitmer, N. Touran, J. Hejzlar, K. Weaver, J. C. Walter, J. McWhirter, C. Ahfeld, T. Burke, A. Odedra, R. Hyde, J. Gilleland, Y. Ishikawa, L. Wood, N. Myhrvold, W. H. Gates. "Traveling-Wave Reactors: A Truly Sustainable and Full-Scale Resource for Global Energy Needs". Proceedings of the 2010 International Congress on Advances in Nuclear Power Plants (ICAPP), San Diego, CA. June 13-17 (2010).

[136] R. Petroski, B. Forget. "General Analysis of Breed-and-Burn Reactors and Limited-Separations Fuel Cycles". Doctoral Thesis, Department of Nuclear Science and Engineering, Massachusetts Institute of Technology (2010).

[137] D. Tenchine. "Some Thermal Hydraulic Challenges in Sodium Cooled Fast Reactors". *Nuclear Engineering and Design* 240(5): 1195-1217 (2010).

[138] M. Reed, R. Parker, B. Forget. "A Fission-Fusion Hybrid Reactor in L-Mode Tokamak Configuration with Natural Uranium". Fusion for Neutrons and Subcritical Nuclear Fission (FUNFI), American Institute of Physics (AIP) Conference Proceedings 1442: 224-231 (September 2011).

[139] F. Heidet, E. Greenspan. "Neutron Balance Analysis for Sustainability of Breed-and-Burn Reactors". *Nuclear Science and Engineering* 171: 13-31 (2012).

[140] F. Heidet, E. Greenspan. "Feasibility of Lead-Cooled breed-and-burn Reactors". *Progress in Nuclear Energy* 54: 75-80 (2012).

[141] N. Todreas. "Thermal Hydraulic Challenges in Fast Reactor Design". 12th International Topical Meeting on Nuclear Reactor Thermal Hydraulics (NURETH-12), Pittsburgh, PA (2007).

[142] J. Cheatham, N. Touran, B. Truong, M. Reed, R. Petroski. "Fast Reactor Design Using the Advanced Reactor Modeling Interface (ARMI)". Proceedings of the 21st International Conference on Nuclear Engineering (ICONE21) in Chengdu, China (2013).

[143] R. Petroski, B. Forget, C. Forsberg. "Evaluation of Core Compositions for Use in breed-and-burn Reactors and Limited-Separations Fuel Cycles". *Annals of Nuclear Energy* 55: 151-168 (2013).

[144] M. P. Short, R. G. Ballinger, H. E. Hanninen. "Corrosion Resistance of Alloys F91 and Fe-12Cr-2Si in Lead-Bismuth Eutectic up to 715 C". *Journal of Nuclear Materials* 434: 259-281 (2013).

Reactor Benchmarks

[145] R. D. Mosteller, P. J. Jaegers. "Detailed Analysis of the Second Zeus Critical Experiment with MCNP". LA-UR-01-1610, Submitted for Presentation at the International Meeting on Mathematical Methods for Nuclear Applications, September 9-13 (2001).

[146] J. D. Bess. "Evaluation of the Initial Isothermal Physics Measurements at the Fast Flux Test Facility, a Prototypic Liquid Metal Fast Breeder Reactor". Idaho National Laboratory, FFTF-LMFR-RESR-001 (2006).

[147] K. Yokoyama, A. Shono. "Japan's Experimental Fast Reactor Jōy MK-I Core: Sodium-cooled Uranium-Plutonium Mixed Oxide Fueled Fast Core Surrounded by UO2 Blanket". JOYO-LMFR-RESR-001, NEA/NSC/DOC (2006).

[148] R. D. Mosteller, R. W. Brewer, J. Sapir. "The Initial Set of Zeus Experiments: Intermediate-Spectrum Critical Assemblies with a Graphite-HEU Core Surrounded by a Copper Reflector". HEU-MET-INTER-006 in *International Handbook of Evaluated Criticality Safety Benchmark Experiments*, NEA/NSC/DOC(95)03 (2004).

Seismic Analysis

[149] A. Preumont, A. Pay, A. Decauwers. "The Seismic Analysis of a Free-Standing FBR Core". *Nuclear Engineering and Design* 103: 199-210 (1987).

[150] "Intercomparison of liquid metal fast reactor seismic analysis codes Volume 3: Comparison of observed effects with computer simulated effects on reactor cores from seismic disturbances". International Atomic Energy Agency (IAEA). Proceedings of a final research coordination meeting held in Bologna, Italy (1995).

[151] B. C. Johnson, G. E. Apostolakis. "Application of the Technology-Neutral Framework to Sodium Cooled Fast Reactors". Doctoral Thesis, Department of Nuclear Science and Engineering, Massachusetts Institute of Technology (2010).

[152] B. C. Johnson, G. E. Apostolakis. "Seismic Risk Evaluation within the Technology Neutral Framework". *Nuclear Engineering and Design* 242: 341-352 (2012).

[153] Global Earthquake Search, United States Geological Survey (USGS). URL: http://earthquake.usgs.gov/earthquakes/eqarchives/epic/epic_global.php (2011).

Textbooks

[154] A. M. Weinberg, E. P. Wigner *The Physical Theory of Neutron Chain Reactions.* University of Chicago Press (1958).

[155] W. M. Stacey. *Variational Methods in Nuclear Reactor Physics* Academic Press, New York, NY (1974).

[156] G. I. Bell, S. Glasstone. *Nuclear Reactor Theory.* Van Nostrand Reinhold Company (1979).

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

[158] S. Nakamura. *Computational Methods in Engineering and Science.* Krieger Publishing Company (1986).

[159] J. Le Clercq *The Nuclear Age.* (1986).

[160] J. R. Lamarsh, A. J. Baratta. *Introduction to Nuclear Engineering*, Ed. 3. Prentice Hall, Upper Saddle River, New Jersey (2001).

[161] S. Gasiorowicz. *Quantum Physics*, Ed. 3. John Wiley Sons (2003).

[162] D. J. Griffiths. *Introduction to Quantum Mechanics*, Ed. 2. Pearson Prentice Hall (2005).

[163] W. M. Stacey. *Nuclear Reactor Physics*, Ed. 2. Wiley-VCH (2007).

[164] A. Hebert. *Applied Reactor Physics.* Presses Internationales Polytechnique (2009).

Historical Sources

[165] J. Chadwick. "Possible Existence of a Neutron". *Nature* 193: 312 (1932).

[166] L. Meitner and O. Frisch. "Disintegration of Uranium by Neutrons: A New Type of Nuclear Reaction". *Nature* 143: 239-240 (1939).

[167] E. Fermi. "The Development of the First Chain Reacting Pile". Symposium on Atomic Energy and its Implications. Proceedings of the American Philosophical Society 90 (1946).

[168] Atomic Energy Act of 1946 (Public Law 585, 79th Congress) http://www.osti.gov/atomicenergyact.pdf (1946)

[169] T. Rockwell. *The Rickover Effect: How One Man Made a Difference.* U.S. Naval Institute Press (1992).

[170] A. M. Weinberg. *The First Nuclear Era: The Life and Times of a Technological Fixer.* American Institute of Physics, New York (1994).

[171] F. Duncan. *Rickover: The Struggle for Excellence. Naval Institute Press.* Annapolis, MD (2001).

[172] History of the U.S. Nuclear Regulatory Commission. http://www.nrc.gov/about-nrc/emerg-preparedness/history.html

[173] 2005 Energy Policy Act: http://www.gpo.gov/fdsys/pkg/PLAW-109publ58/content-detail.html (2005).

Popular Literature

[174] "World List of Nuclear Power Plants". *Nuclear News*, 13th Annual Reference Issue (2011).

[175] M. Reed. "Crises Are Not Opportunities". Guest Opinion Column, The Tech (April 2011).

[176] M. Reed. "Violent Nascence: Raw Genius Made Deadly Power". Fortnight Journal, Ed. 3 (December 2011).

[177] M. Reed. "From War to Peace: Buck Bureaucracy, Spark Innovation". Fortnight Journal, Ed. 3 (January 2012).

[178] M. Reed. "Nuclear Waste and Nuclear Medicine: Politicians Exploit Science to Create Fear". Fortnight Journal, Ed. 3 (January 2012).

[179] M. Reed. "Nuclear Atrophy: Accidents Choked Nuclear Progress". Fortnight Journal, Ed. 3 (February 2012).

[180] M. Reed. "Nuclear Renaissance: After a Lost Generation, Young Minds Reengage". Fortnight Journal, Ed. 3 (February 2012).

[181] M. Reed. "Fukushima Future: In Crisis, Expertise Falters". Fortnight Journal, Ed. 3 (March 2012).

[182] E. P. Loewen. "The Science (and Art) of Science Communication for Nuclear Technology: A Matter of Perspective", Speech for "The Science of Science Communication", Arthur M. Sackler Colloquia of the National Academy of Sciences, Washington, D.C. (2012).

[183] M. Reed. "Creative Writing on the History of Nuclear Technology". Transactions of the 2012 ANS Winter Meeting and Nuclear Technology Expo in San Diego, CA (November 2012).

[184] "Developing a New Nuclear Workforce". *Nuclear News*, 56(1): p. 40 (2013).

[185] R. Graber, T. Retson. "Will Low Natural Gas Prices Eliminate the Nuclear Option in the U.S.?". 5th Annual Nuclear Construction Summit, Charlotte, NC (2013).

Codes and Algorithms

[186] W. E. Arnoldi. "The Principle of Minimized Iterations in the Solution of the Matrix Eigenvalue Problem". *Quarterly of Applied Mathematics* 9(1): 17-29 (1951).

[187] B. J. Toppel, A. L. Rago, D. M. O'Shea. "MC2: A Code to Calculate Multigroup Cross-Sections". ANL-7318 (1967).

[188] R. W. Hardie, W. W. Little, Jr. "PERT-V, A Two-Dimensional Perturbation Code for Fast Reactor Analysis". Reactor and Plant Technology Department, FFTF Project, Battelle Memorial Institute, Pacific Northwest National Laboratory (1969).

[189] A. P. Olson, G. K. Leaf, D. A. Meneley, P. M. Walker. "The Fuel Cycle Analysis System, REBUS". *Nuclear Science and Engineering* 45: 53 (1971).

[190] W. M. Stacey, J. P. Regis. "VARI-1D: A One-Dimensional Variational Sensitivity Code". FRA Technical Memorandum No. 57, FRA-TM-57, Argonne National Laboratory (1973).

[191] C. L. Wheeler, C. W. Stewart, R. J. Cena, D. S. Rowe, A. M. Sutey. "COBRA-IV-I: an interim version of COBRA for thermal-hydraulic analysis of rod bundle nuclear fuel elements and cores". Battelle Pacific Northwest National Labs, Richland, WA. BNWL-1962 (1976).

[192] G. A. McLennan. "NUBOW-3D (inelastic): A FORTRAN Program for the Static Three-Dimensional Analysis of Bowed Reactor Cores, Including Irradiation Creep and Swelling". Transactions of the American Nuclear Society, Washington, D.C., November 12 (1978).

[193] K. L. Derstine. "DIF3D: a code to solve one-, two-, and three-dimensional finite-difference diffusion theory problems". Argonne National Laboratory, ANL-82-64 (1984).

[194] T. A. Taiwo. "DIF3D-K: A Nodal Kinetics Code for Solving the Time-Dependent Diffusion Equation in Hexagonal-Z Geometry". Argonne National Laboratory (1992).

[195] A. M. Tentner, et. al. "The SAS4A LMFBR Whole Core Accident Code". Proceedings of the International Meeting on Fast Reactor Safety, p. 989, Knoxville, Tennessee (1985).

[196] F. E. Dunn, et. al. "The SASSYS-1 LMFBR Systems Analysis Code". Proceedings of the International Meeting on Fast Reactor Safety, p. 999, Knoxville, Tennessee (1985).

[197] E. Z. Mueller. "LABAN-PEL: A Two-Dimensional, Multigroup Diffusion, High-Order Response Matrix Code". PEL-309 (1991).

[198] E. Z. Mueller, Z. J. Weiss. "Benchmarking with the Multigroup Diffusion High-Order Response Matrix Method". *Annals of Nuclear Energy* 18:9,535-544 (1991).

[199] G. Palmiotti, et. al. "VARIANT: VARIational Anisotropic Nodal Transport for Multidimensional Cartesian and Hexagonal Geometry Calculation". Argonne National Laboratory, ANL-95/40 (1995).

[200] S. Balay, W. Gropp, L. C. McInnes, B. Smith. "PETSc - The Portable, Extensible Toolkit for Scientific Computation". Argonne National Laboratory (1998).

[201] G. W. Stewart. "A Krylov-Schur Algorithm for Large Eigenproblems". *SIAM Journal on Matrix Analysis and Applications* 23(3): 601-614 (2002).

[202] F. B. Brown. "MCNP - A General Monte Carlo N-Particle Transport Code, Version 5". LA-UR-03-1987, Los Alamos National Laboratory (2003).

[203] V. Hernandez, J. E. Roman, V. Vidal. "SLEPc: Scalable Library for Eigenvalue Problem Computations" High Performance Computing for Computational Science, VECPAR 2002. Lecture Notes in Computer Science 2563: 377-391 (2003).

[204] J. D. Hunter. "Matplotlib: A 2-D Graphics Environment". *Computing in Science and Engineering* 9, No. 3, 90-95 (2007).

[205] C. H. Adams. "Specifications for VARI3D - A Multidimensional Reactor Design Sensitivity Code". Argonne National Laboratory.

[206] "VARI3D: A Generalized Perturbation Theory Capability for DIF3D". Argonne National Laboratory (2008).

[207] M. Lutz. *Programming Python.* O'Reilly Media, Inc. (2010).

[208] "DIF3D10.0: Code System Using Variational Nodal Methods and Finite Difference Methods to Solve Neutron Diffusion and Transport Theory Problems". RSICC Compute Code Collection, Oak Ridge National Laboratory. CCC-784.(2011).

[209] J. W. Fricano, J. Buongiorno. "COBRA4i-MIT: An Updated Sub-Channel Analysis Code for Sodium Fast Reactor Design". Proceedings of ICAPP, Nice, France. American Nuclear Society (2011).

[210] MATLAB version R2012a. Natick, Massachusetts: The MathWorks, Inc. (2012).

[211] N. Touran, J. Cheatham, R. Petroski. "Model Biases in High-Burnup Fast Reactor Simulations". PHYSOR 2012, April 17, Knoxville, Tennessee (2012).

[212] A. Scopatz, P. K. Romano, P. P. H. Wilson, K. D. Huff. "PyNE: Python for Nuclear Engineering". *Transactions of the American Nuclear Society* Winter Meeting (2012).

[213] W. Boyd, K. Smith, B. Forget. "A Massively Parallel Method of Characteristics Neutral Particle Transport Code for GPUs". Mathematics and Computation, Sun Valley, Idaho (2013).