SENSITIVITY AND UNCERTAINTY ANALYSIS OF MULTIPHYSICS NUCLEAR REACTOR CORE DEPLETION

by

Andrew Scott Bielen

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Nuclear Engineering and Radiological Sciences)
in the University of Michigan
2015

Doctoral Committee:

Professor Thomas J. Downar , Co-Chair
Associate Professor Annalisa Manera, Co-Chair
Associate Professor Krzysztof J. Fidkowski
Professor John C. Lee
Joseph L. Staudenmeier, US Nuclear Regulatory Commission

For my wife, Lisa

AKNOWLEDGEMENTS

I would like to acknowledge the contributions of the many that helped and guided me through this Ph.D. process. First and foremost are my co-advisors Prof. Tom Downar and Prof. Annalisa Manera, whose guidance and support were instrumental in the completion of this dissertation. I would also like to thank my committee members, Prof. John Lee and Prof. Krzysztof Fidkowski, and Dr. Joe Staudenmeier of the US Nuclear Regulatory Commission, whose feedback on this dissertation was invaluable. Additionally, I thank my branch chief at the NRC, Dr. Chris Hoxie, for his patience, understanding, and support during the completion of this work.

In addition to these members, I must also thank the following: Dr. Patrick Raynaud of the US NRC and Ken Geelhood of Pacific Northwest National Laboratory, for conversations and guidance on working with and developing the FRAPCON fuel performance code for coupled neutronics and uncertainty analysis was crucial; Dr. Andrew Ward of the University of Michigan, for assisting me with the necessary PARCS/PATHS development and cross section generation using HELIOS; Dr. Tim Drzewiecki of the US NRC for assistance in providing the computer resources required to complete the sensitivity and uncertainty portion of this thesis; and Michael Rose and Thomas Saller at the University of Michigan for supporting me on my visits back to Ann Arbor.

Lastly, but most importantly, I thank my wife Lisa for supporting my desire to earn my doctorate, and her willingness to engage in considerable personal sacrifice to ensure that it happened. Without her emotional support, this work would not have been accomplished.

This dissertation was completed with the support of the US NRC Graduate Fellowship Program from 2011-2015.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

ABSTRACT

Nuclear reactor simulation is a complex multiphysics process which involves solution to equations describing the neutronic, thermal-hydraulic, and fuel thermo-mechanical behavior of the core components. Typically in current generation reactor physics analysis these three component areas are given separate consideration or are at best loosely coupled. Within this work, a methodology for tightly coupling the core neutronics code PARCS, thermal-hydraulics code PATHS, and fuel rod simulator code FRAPCON was developed. This coupled code package, referred to as FRAPARCS, was applied to two fuel depletion problems: a pin cell and a 5x5 assembly mini-core. The results of the depletion calculations indicate that standalone PARCS does not adequately capture the evolution of fuel rod behavior which influences the Doppler fuel temperature used in cross section evaluation, and as a result significant differences in computed core performance can be seen. In particular, the behavior of the fuel-cladding gap and associated temperature drop was found to be important.

FRAPARCS was then applied to the pin cell calculation to evaluate the uncertainty and sensitivity of the nuclear performance of the core due to the influence of fuel thermo-mechanical models available for manipulation in FRAPCON. A sensitivity study was conducted to determine which fuel models were influential on the neutronics outputs; we determined that fuel thermal conductivity, fuel thermal expansion, cladding creep, and fuel swelling had an important influence on the core Doppler temperature and reactivity. Additionally, the heat transfer coefficient was found to be important. Then, FRAPARCS was integrated within the DAKOTA uncertainty package. Two varieties of sampling-based methods (Random and Latin Hypercube Sampling) and two stochastic expansion methods (Polynomial Chaos Expansion (PCE) and Stochastic Collocation SC)) were used to evaluate the uncertainty in the nuclear parameters throughout core depletion. We found that the uncertainty in the core reactivity relative to the unbiased case was approximately 60 pcm at the beginning of depletion, reducing to approximately 15 pcm by the end of life, due to the dampening effect of plutonium buildup

reducing the importance of fuel uncertainty on high-burnup fuel. We found that the response statistics could be well-estimated by a first-order tensor-product PCE expansion, only requiring 32 calculations.

Using variance-based decomposition, we found that initially the most important models contributing to nuclear variance were the thermal conductivity and fuel thermal expansion models. Once the fuel-clad gap closes, however, fuel thermal conductivity uncertainty dominates the overall variance in the output. We also found that the importance of input interactions on overall variance is negligible; at worst case, interaction effects contribute ~3% of the overall variance in both Doppler temperature and reactivity.

CHAPTER 1

Introduction

1.1 Motivation

Nuclear reactor core analysis requires accurate, reliable and efficient simulation software to predict the behavior of reactor systems under normal operation and transient scenarios. This need exists because the expense, regulatory hurdles, and inherent potential safety issues that exist in the construction and operation of nuclear facilities preclude the ability to construct full-scale experimental facilities to evaluate system behavior under the full range of postulated conditions. In the nuclear industry, computer simulations are used (validated with smaller-scale separate and integral effects tests subject to applicability and scaling analysis) to quantify the behavior of nuclear systems and the margins that exist to failure under all plausible scenarios.

The purpose of this research is to investigate methods to quantify the uncertainty in nuclear design computations that arise from uncertainties in fuel performance during the depletion of the reactor fuel. One of the primary goals of this research is to provide these estimates using current-generation reactor design and analysis tools rather than some of the more advanced methods currently under development (for example, consider the suite of methods being developed under the auspices of the Department of Energy's CASL program [1]). The primary reason for not using advanced codes is they do not yet have as extensive validation base as the current generation of codes. Furthermore, current generation codes are fast running, which is advantageous for performing uncertainty analysis.

The principal problem being addressed in this work is core depletion, in which the coupled neutron/nuclide field in the core is evaluated as the fuel undergoes irradiation in the power production process. There are both economic and safety requirements that must be met by

the core throughout the depletion cycle. The economic requirements include energy production over a specified time period and efficient fuel utilization are examples; while the safety requirements include maintaining the required shutdown margin as well as thermal and material limits within regulatory criteria. By far the most common tool used in the nuclear industry to compute such quantities is a two energy group, nodal diffusion-based core simulator which includes neutronics modules, depletion modules, and simplified thermal-hydraulics modules. Common examples of such software include PARCS, SIMULATE, and ANC ([2], [3], [4], respectively).

The following codes will be used in the work to evaluate the effect of the thermo-mechanical fuel performance parameters on the fuel during depletion:

- FRAPCON-3.4 is used to evaluate the evolution of nuclear fuel rods throughout the depletion process, including the computation of temperature distribution and mechanical and chemical changes within the fuel pellets and cladding [5].
- PARCS is used to compute the neutronics performance of the core, including core reactivity and power distributions throughout the fuel burnup cycle [2].
- PATHS is used to obtain the thermal-hydraulic behavior of the coolant within the reactor core [6].
- The DAKOTA code package is used to perform uncertainty analysis, including estimates of correlation parameters between the fuel performance input and core neutronics/thermal-hydraulics output [7].

The physics codes above were developed under contract for the US Nuclear Regulatory Commission (NRC) and are representative of methods currently used by the nuclear industry for licensing basis design calculations. PATHS is provided as a library within PARCS and is the default method for computing the thermal hydraulics of the core during depletion. The first step in the completion of this thesis was to develop a method to couple FRAPCON to PARCS/PATHS which was achieved using the scripting tool Python [8]. Then, the coupled codes (referred to as FRAPARCS) were implemented within the DAKOTA framework, and a number of different uncertainty approaches available in the code package were used to investigate the range of correlation between the fuel parameters and the resulting neutronics/thermal-hydraulics output. Further details are provided in the appropriate chapters.

The purpose of this introductory Chapter is to provide a brief description of the ways that nuclear reactors are modeled and to provide a demonstration of the impact of fuel performance on reactor depletion to help establish the motivation for the research here. This was achieved by

depleting a core model of a Pressurized Water Reactor (PWR) using PARCS/PATHS over a range of possible gap conductances. The differences in core reactivity, fuel temperatures, and power distribution are then discussed. Finally, the introductory chapter ends with an outline and discussion of the rest of the thesis.

1.2 Reactor Analysis Methods

A principal challenge in relying on computer simulations for reasonable assurance that nuclear power plants are operated safely (e.g. there is no release of fission products from the fuel rods) is that reactor simulation is a complex multi-physics problem. Thus, methods have been developed over the last several decades of reactor analysis to simplify the equations and methods in order to insure that calculations are tractable on the computer resources available at the time, while simultaneously maintaining assurance that the accuracy of simulation results was still "acceptable." In the past, one of the key methods used to reduce the complexity of the models is to perform operator splitting, in which the different classes of physics that represent reactor behavior are divided and simulated individually, with simplified feedback between them [9]. With appropriately conservative treatment of these interactions, margin between failure criteria and the simulation results is assured. The downside is that excessive conservatism can lead to inefficient plant operation and economic penalties. As noted below, advanced methods are currently under development to remove some of the conservatism.

1.2.1 Simulation Needs and Approaches

There are at least three classes of physics that are important in modeling the behavior of a reactor core: the neutronics, which is concerned with determining the distribution of neutrons and nuclides within the core, which therefore determines the power source; the thermal-hydraulics, which is concerned with the mass, momentum, and energy distribution of the reactor coolant, providing the sink for energy generated within the core; and the reactor fuel thermo-mechanics, which is concerned with the temperature distribution within the reactor fuel rods and the mechanical and chemical changes that are induced within it under irradiation. There is some overlap between these different areas; the power distribution obtained from the neutronics solution drives the temperature and material changes in the fuel thermo-mechanics and the fluid conditions computed by the thermal-hydraulics, which then in turn influence the temperatures

used in cross section evaluations to determine the power distribution and reactivity. For the sake of simplicity, these interactions are often simplified or treated conservatively; however, this treatment can introduce unwelcome ambiguity depending on the particular situation being evaluated.

To illustrate this point, consider that within the reactor core analysis process, the first step might be to generate power distributions using the core simulator with very limited treatment of fuel changes with burnup. The range of power distributions experienced by all the rods in the reactor core might then be passed to a fuel thermo-mechanical code to evaluate their behavior against the design criteria. There are two primary competing effects driving fuel temperature changes with burnup: first, the cladding creeps down and the fuel swells, which eventually causes contact between the fuel and cladding and tends to drive the fuel temperature down; however, the fuel thermal conductivity degrades with burnup, which for a fixed power level would tend to drive the fuel temperature up. In either case, by imposing a power distribution computed by a code with limited knowledge of these effects in the thermo-mechanical calculation, the actual fuel temperatures computed may be either higher (which would be conservative) or lower (which would be non-conservative) than what would be actually experienced. In either case, neglecting this interaction can result in fuel temperatures which are not consistent between the different codes used to model the same physical system. This ambiguity is understandably undesirable.

Each of these physics classes provides its own unique set of challenges. For example, within the neutronics domain there has been considerable research over the last 50 years which has established confidence in the predictions of the coupled neutron/nuclide field. However, the material structure and complexity of the energy dependence of the cross sections involved make detailed full core calculations extremely expensive. The challenges within the thermal-hydraulics area are primarily the representation of the stress tensor within the Navier-Stokes equations, as well as the inherent non-linearity of the fluid field equations. In the thermo-mechanical domain, the material models used are often empirical or semi-empirical due to the complexity of representing material behavior under irradiation; it is very difficult to represent material behavior at a macro-scale based entirely on first principles.

For Light Water Reactors (LWRs), the common method for dealing with the material and cross section complexity in neutronics calculations is to perform the analysis in multiple stages:

the process begins with cross section library generation at the pin cell level, which is very basic in spatial detail and very fine in energy detail in order to reduce some of the explicit energy dependence while still maintaining its effects. Then, lattice physics calculations are performed at the fuel assembly lattice level to generate few (usually two) group cross sections that spatially homogenize the detailed assembly geometry. The spatial and angular dependence of the neutron flux at the lattice physics stage is computed using detailed transport theory. These few group cross sections are generated at a number of reactor conditions for use within the core simulator to cover the operating range. The core simulator, then, typically uses some low order neutron transport approximation such as diffusion theory with these pre-computed cross sections to calculate the reactor power distribution and criticality. A summary work which describes this process can be found in [10].

The process described above requires considerable computation on the front end to generate few group cross sections over the anticipated range of temperature and fluid conditions, but once computed, reactor core depletion can be simulated in a matter of minutes to a few hours using a core simulator such as PARCS. Despite the fact that detailed assembly geometry is "smeared" or homogenized into uniform blocks during the calculation, detailed pin-level information can be extracted using pin-power reconstruction methods, should this be of interest in a particular analysis. Because it is well-understood, relatively inexpensive, and sufficiently accurate for economical operation, power reactors in the United States all use some variation of this design process to perform their depletion calculations.

As computer resources have grown in capability, there has been a commensurate growth in the complexity of the models (removal of approximations and therefore a more faithful representation of the physics involved) and a decrease in the acceptable difference between the computed result and the failure criteria (the closer the reactor is operated to the margins, the more productive and efficient it is). An illustration of the first point is contained within the DOE CASL program; one of the program's goals is to develop integrated methods that can simultaneously simulate a full reactor core with pin-level, transport-based neutronics, detailed, finite-element based fuel performance, and subchannel-level thermal-hydraulics [11]. This is an enormous problem with billions of degrees of freedom, but with a sufficiently large number of computer processors, it is technically feasible. However, the number of facilities across the world that have access to such computational resources is very limited, and therefore for the types of

routine design calculations performed regularly in the commercial industry, the current methods will continue to be used for the foreseeable future.

1.2.2 Uncertainty Analysis and Multiphysics Simulation

Related to the advancement of the state of knowledge within reactor technology and the desire to recover lost margin in reactor operation has been the movement towards analysis methods based on a best-estimate, rather than conservative, approach. The traditional approach, particularly for thermal analysis, was to treat uncertainty in particular parameters by biasing initial conditions and models in the worst possible direction in order to maximize the consequences of a given scenario. The classic example of this philosophy is Appendix K Loss of Coolant Accident (LOCA) analysis [12], which dates from the early days of commercial nuclear power. The "bounding" approach was based on the belief that, if a reactor simulation predicted acceptable results even for this worst case of conditions, then it would survive any potential scenario that might be encountered during actual operation. This was the accepted state of the art in the nuclear industry for many years.

The primary difficulty with the bounding approach is that, while in general (but not always) adequate margin is assured to failure criteria, it can lead to excessive margin, resulting in economic penalties with little real additional safety benefit. Therefore, as the understanding of the underlying physics of nuclear power plants and the complexity of the analysis methods has improved, more sophisticated "best estimate" methods for margin evaluation were developed. The NRC provided guidance to the nuclear industry for the development of best-estimate methods for LOCA analysis in the late 1980's ([13], [14]), which then led to nuclear fuel suppliers receiving approval for the use of such methods for licensing analysis [15], [16]. The availability of these techniques played a large role in the implementation of thermal power uprates for power reactors from the 1990's through today.

Since the best-estimate LOCA methodologies matured and were implemented in licensing-basis calculations, there have been efforts both to provide for more complete physics representation in nuclear reactor system analysis and to better quantify the margins which exist within these methods. Beginning in the late 1990's, the Nuclear Energy Agency (NEA) of the Organisation for Economic Cooperation and Development (OECD) sponsored efforts to couple system-level thermal-hydraulics codes with transient core simulators for realistic Main Steam

Line Break (MSLB) accident analysis [17]. OECD/NEA has also been active in the area of neutronics margin quantification, as evidenced by their support of the Light Water Reactor Uncertainty Analysis in Modeling (LWR UAM) benchmark problems [18]. Part of this benchmarking effort is devoted to determining the influence of fuel thermo-mechanics on neutronics behavior (Phase II Part 1), which is one area where the research here intends to make a contribution.

1.3 Demonstration Case: PWR Depletion

In order to assess the potential effect that fuel performance can have on PWR core depletion, a full-core depletion was analyzed, based on a model developed to validate PARCS against measurements taken during the first cycle of depletion for Surry Power Station Unit 1 ([19], [20]).

1.3.1 PARCS Model Description

The base model described the Surry nuclear reactor with parameters consistent with measured values at different points throughout the cycle. Surry is a 157 assembly, 3-loop Westinghouse PWR operating in Southern Virginia. For the initial cycle, it used a 15x15 fuel assembly design and operated at a thermal power of 2441 MW. The core layout including initial assembly enrichments is given in Figure 1.3.1:

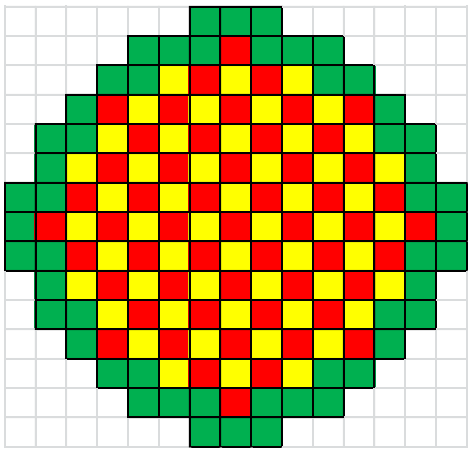


Figure 1.3.1: Surry cycle 1 core map (red: 3.117 wt/o; yellow: 2.572 wt/o; green: 1.868 wt/o)

For this sensitivity study, the following simplifications were made in order to provide a full-core model that simulated the effect of depletion for a time period characteristic of fresh-to-fully burned fuel:

- The number of timesteps was modified from values taken from the measurements to 47 one-month intervals, in order to simulate the effect of depleting the fuel out near its licensing limit (allowable peak nodal burnup is 62 GWd/MTU)
- The core power throughout each depletion step was set to 100%, rather than the measured values
- The boron concentration used through each depletion step was set to zero, because the interest for this case was to generate core reactivity vs. time curves, rather than matching measured reactivities. Also, the control rods bank positions were modified to be consistent with their full power values reported in [19]
- The core inlet temperature and flow rate was set to the full power values reported in [19]

In PARCS, the fuel treatment is simplified with material properties and dimensions taken as their fresh, beginning-of-cycle values. In addition, the gap conductance (which captures the value of the temperature drop across the fuel-cladding gap) is considered to be fixed across the core for each assembly type throughout the depletion. This value can have a strong influence on the calculated fuel temperatures, which are then used in the PARCS cross section lookup routines during calculation of the neutron flux shape. The limitation in the current PARCS treatment of fuel behavior is that it treats as static many quantities (such as fuel thermal conductivity) that in fact evolve considerably over the course of depletion. This could therefore have a significant impact on the prediction of fuel temperatures, which could result in a misprediction of the core power shape or core reactivity. The direct evaluation of these effects is precluded with standalone PARCS, however, so for the demonstration problem, the only fuel thermo-mechanically-related parameter easily available to the user (the gap conductance) was used.

1.3.2 Gap Conductance Calculation with FRAPCON

In order to perform a sensitivity study on gap conductance a FRAPCON calculation was first performed for a pin at core-average conditions (i.e., core-average Linear Heat Generation Rate (LHGR) and channel inlet mass flow rate and temperature) and axial power distributions representative of a PWR during a 60 GWd/MTU depletion. In this calculation, 9 axial nodes were used to represent the fuel rod, which was deemed sufficient for the scoping computation

performed here. One of the many outputs from such a calculation is the nodal gap conductance (a heat transfer coefficient-like quantity that represents the heat flow out of the fuel region into the cladding). Figure 1.3.2 shows the variation of gap conductance from this calculation for each of the 9 axial nodes during the irradiation of the fuel:

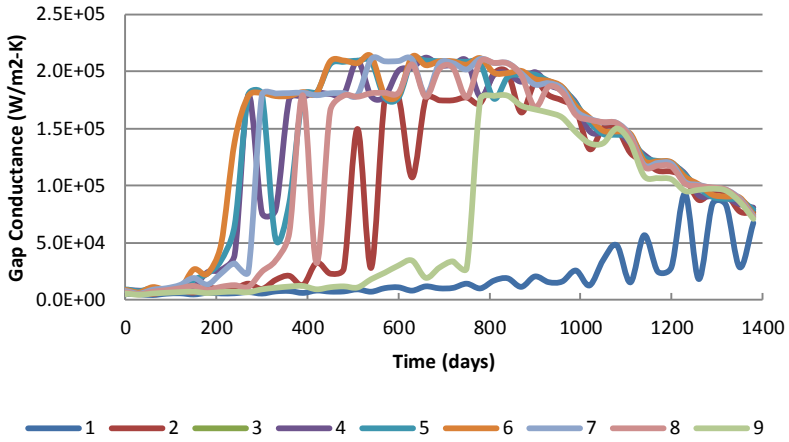


Figure 1.3.2: FRAPCON prediction of nodal gap conductance for Surry-1 depletion

From this figure, the behavior of the gap is clearly evident. Early in life (prior to about 200 days), the gap is open between the fuel and cladding for all nodes; this corresponds with a gap conductance on the order of 5000 W/m^2-K. As burnup accrues, the fuel begins to swell outward and the cladding creeps down onto the fuel; this causes the gap to close and the conductance to increase by several orders of magnitude (up to around 200,000 W/m^2-K), staying there for a large portion of the depletion. The oscillation seen for certain nodes is due to the evolution of the axial power shape with depletion, which causes the fuel and cladding to strain in and out of contact. At approximately 800 days, the gap conductance begins to drop, and by end of life is roughly half of the maximum; this is due to fission gas release late in life, which reduces the gas gap conductivity (xenon and krypton, in particular, have a much lower conductivity than the helium gas fuel rods are initially filled with) and causes a resulting decline in the gap conductance. This figure provides the possible ranges of gap conductance that can be experienced by the fuel during depletion, and thus provides a basis for the following sensitivity study.

1.3.3 Neutronic Behavior as a Function of Gap Conductance

Using the results of the FRAPCON calculation as a reference, a range of gap conductances between the beginning-of-cycle minimum value of roughly 5000 W/m^2-K through the middle-of-cycle maximum value of 200000 W/m^2-K were input into PARCS and used to represent the gap conductance for the entire core throughout the whole 47 month fuel depletion history. Of interest were the following neutronic quantities:

- Core average Doppler temperature
- Reactivity (k_{eff})
- Axial peaking factor F_z
- Radial peaking factor F_{xy}
- Total peaking factor F_q

It is important to keep in mind that, in this sensitivity study, the gap conductance was used as a surrogate, easily controllable quantity to represent a range of fuel-related effects. During an actual reactor depletion, each node in the core will have a distinct gap conductivity (and other fuel-related quantities) based on its individual power and exposure history. The treatment used here is very coarse with the purpose of showing that there is a noticeable impact on the neutronic quantities when there are significant variations in the thermo-mechanical properties, therefore justifying development of more explicit fuel representation in depletion analysis.

With this in mind, the results of the sensitivity study for the Doppler temperature is presented in Figure 1.3.3. The differences in core reactivity and peaking factors are computed relative to the outputs of the Beginning of Cycle (BOC) gap conductance case (using 5000 W/m^2-K) and are presented in Figure 1.3.4(a)-(d).

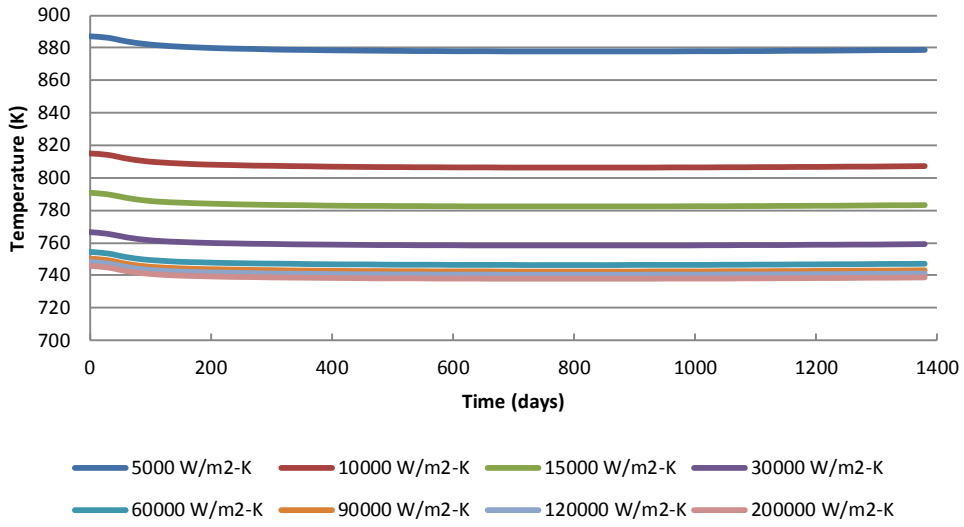


Figure 1.3.3: Doppler temperatures from gap conductance sensitivity study

It is clear from this figure that the gap conductance has a direct, inverse relationship to the Doppler temperature; as expected, the higher the gap conductance, the lower the fuel temperature becomes. Over the range of gap conductances considered, the difference can be as much as 140 K; also evident is an asymptotic convergence to the value at 200,000 W/m^2-K. The interpretation of this is that at a certain point, the gap conductance's effect on (at the very least) core average Doppler temperature becomes fairly small (i.e., the difference in gap temperature drop computed between a calculation run with, for example, 5000 W/m^2-K and 7000 W/m^2-K would be much greater than the difference in a calculation run with 150,000 W/m^2-K and 152,000 W/m^2-K).The reason for this is that when the fuel and cladding are in sufficient contact, the temperature drop across the gap becomes small enough that the impact of changing the value on the fuel surface temperature of the gap conductance becomes vanishingly small, which (in the absence of consideration of fuel thermal property changes with burnup) results in a decreasingly small impact on the overall Doppler temperature. It is important to keep in mind here that the value under comparison is the core average Doppler temperature which is an integral quantity that characterizes the whole core behavior. Local temperatures may show much greater variation depending on the conditions that each individual node experiences.

The impact of the differences in Doppler temperature on the reactor physics parameters are presented in the following Figure:

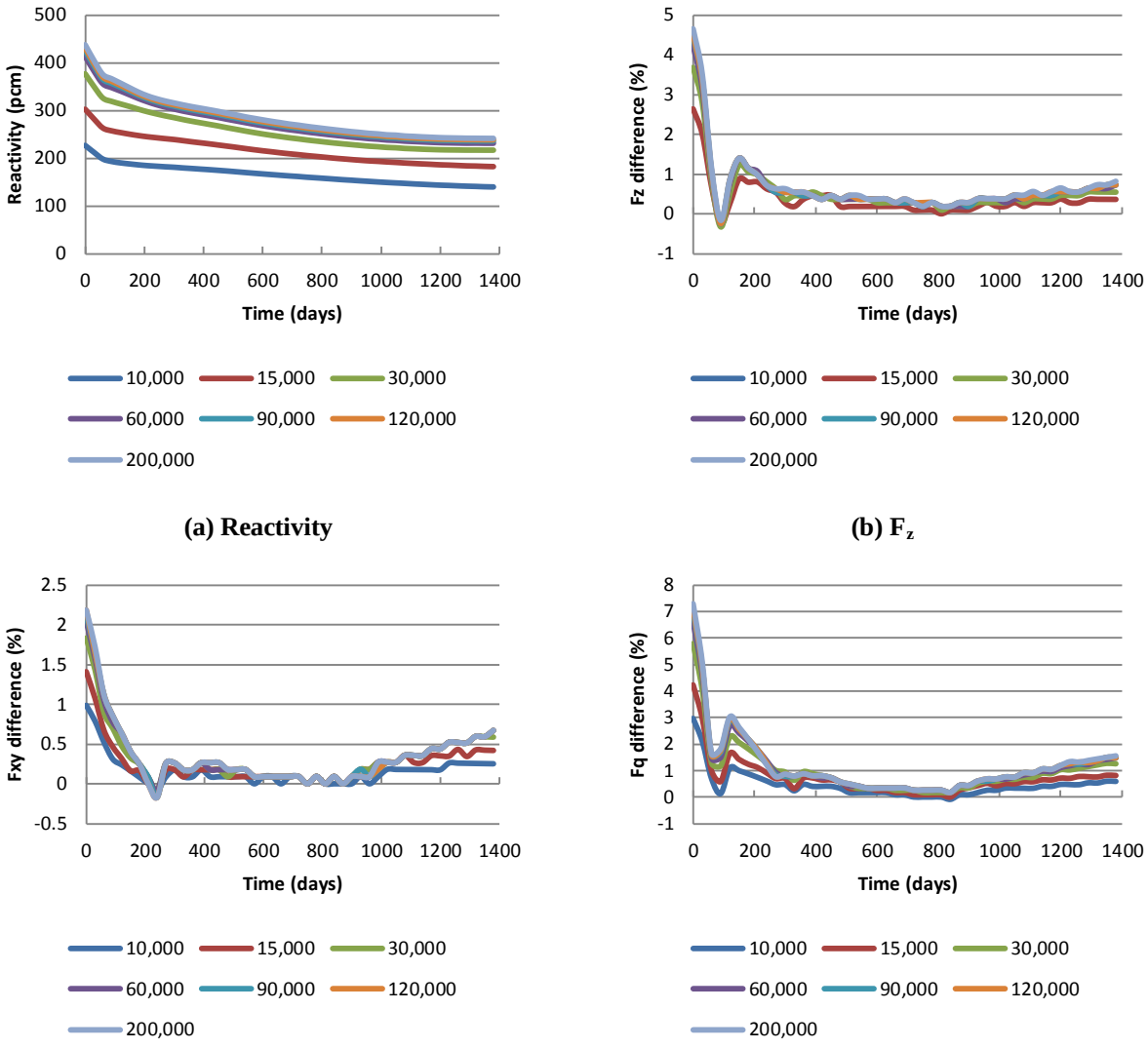


Figure 1.3.4: Neutronics outputs relative to 5000 W/m^2-K

From Figure 1.3.4(a), we see that the difference in core reactivity relative to the BOC gap conductance, presented in 10^{-5} Δk (i.e., pcm), can be significant. Because of the lower Doppler temperatures, the core k-effective predicted by PARCS is several hundred pcm higher using the best estimate FRAPCON gap conductances. The difference in approximately 140 K in Doppler temperature translates to a reactivity difference of approximately 425 pcm at the initial point in cycle, decreasing as the core is burned and plutonium buildup in the hotter cases contributes to decrease the difference. As a point of comparison, the reactivity insertion required to render a pure uranium system prompt supercritical is approximately 640 pcm; therefore, these results

show that the reactivity differences can be quite significant. As expected, at some point fractional changes on gap conductance start to have an asymptotically small impact on core criticality; this is because the fractional changes in core average temperature become very small. Even from this simplified analysis, it is evident that there can be a non-negligible impact (order of hundreds of pcm) on the reactivity resulting directly from the fuel temperature variations that are observed. This is particularly evident for gap conductances that characterize low-burnup fuel.

The effect of the gap conductance on the various peaking factors is shown in Figure 1.3.4(b)-(d). It is evident that early on, the gap conductance changes do cause somewhat significant differences, particularly in the axial direction (of magnitude ~4%), but as the core burns through the initial portion of the cycle the power shape smooths out and the differences become less substantial (< 1%). Given the local nature of peaking factors, and the global nature of how the gap conductances were applied to each of the different depletion calculations, it is difficult to draw a great deal of physics-based insight into the behavior of the system. What is clear, however, is that modifying fuel thermal properties can have a measurable impact on these parameters, and further study with a more detailed model is warranted to fully understand the phenomena.

1.3.4 Sensitivity Study Conclusions

In this section, a preliminary sensitivity study was performed of the various neutronics behavior to the fuel gap conductance. A core-average fuel rod computation using the fuel thermo-mechanical code FRAPCON was performed and used to provide the variation in the gap conductivity to PARCS. A full core PWR model was then depleted in PARCS over a 47 month period, and the resulting changes in the neutronic parameters relative to a beginning of life value of gap conductivity were tabulated. The conclusions from this preliminary study were that the core can show noticeable changes in both integral (core-averaged Doppler and k-effective) and local (peaking factor) parameters depending on how the fuel is represented, but the exact extent of those changes is still unclear because gap conductivity (and other fuel properties) are fundamentally local phenomena, and the changes were applied on a global (core-wide) basis.

That being said, it is clear that the effects can be significant, and further exploration of their natures using a more detailed approach is warranted. This forms the basis for the development of a coupled PARCS/PATHS/FRAPCON tool that can be used to investigate the

importance of representing the fuel properties (and therefore temperatures) more accurately on a local level that can accommodate the heterogeneous burnup distribution in the fuel that evolves during the burnup cycle.

1.4 Outline of Thesis

The purpose of this introductory chapter was simply to provide the overall context for the research performed in this thesis. The remainder of the thesis is organized as follows: In Chapter 2, a literature review is presented to provide background for this work within the framework of current reactor analysis methods. In Chapter 3, a detailed description of FRAPCON and PARCS/PATHS is provided, as well as the algorithm modifications and coupling methodology. The process of integrating the coupled code package FRAPARCS within the DAKOTA uncertainty framework is also discussed. In Chapter 4, the results are presented of a pin cell neutronics model and a mini-core for depletion to demonstrate the effectiveness and impact of the coupled code system relative to executing them in the traditional sequential mode. In Chapter 5, the theoretical basis is presented for the uncertainty and sensitivity analysis methods employed to characterize specific fuel parameters which are important to neutronics depletion calculations. In Chapter 6, the implementation of these methods using the coupled code system and the DAKOTA package is described. Finally, in Chapter 7, overall conclusions and suggestions for further work are provided.

CHAPTER 2

Review of Literature

2.1 Introduction

This chapter reviews the state of the art in coupled fuel mechanical/neutronics/thermal-hydraulics calculations, and provides a context for the research described in this research. First, an overview of the different codes available for fuel thermo-mechanical calculations is presented. Then, a description of the codes and methods available for reactor depletion calculations is given. There are two parts to this description: first, the methods used in "traditional" reactor analysis (i.e., lattice physics transport to generate assembly cross sections, which then provide input to diffusion-based core calculations) are discussed. Second, the more modern, full core transport methods are described. Following the discussion of the codes and methods for individual fields, an overview of multiphysics calculations within nuclear engineering will be provided. Finally, a discussion of methods available for sensitivity and uncertainty analysis is presented, followed by a description of relevant nuclear applications.

2.2 Fuel Performance Analysis Codes

During depletion nuclear fuel undergoes many thermo-mechanical changes as a result of exposure to the temperature, pressure and radiation fields present within the reactor while operating. In order to predict the magnitude of these changes, and to have reasonable assurance that fuel rods will maintain their integrity during operation, a number of computer codes have been developed by various entities over the years. Fuel modeling is a very complex process, and

each code uses a number of approximations and adjustments to capture the physics appropriately. Rashid, et al, provide an overview of the history of fuel performance modeling, as well as the current plans for the immediate future [21]. They present a history of fuel performance codes, a description of the material and mechanical models typically embedded within with codes, and describe the need to develop fully three-dimensional, more mechanistic- (as opposed to empirically-) based constitutive materials. Figure 2.2.1 illustrates the complexity of fuel modeling by demonstrating the inter-connectivity of all the relevant physics.

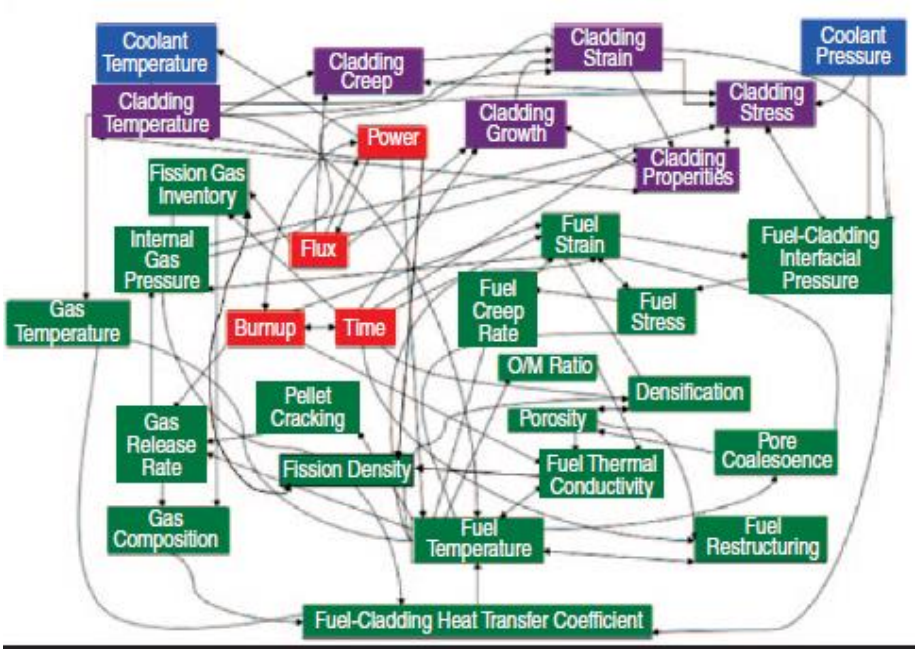


Figure 2.2.1: Complexity of fuel performance modeling (taken from [21])

Any nuclear fuel rod performance code must have models, correlations, and computational structure to account for each of these phenomena in order to accurately predict the integral behavior of the fuel. Below is a discussion of several of the fuel rod performance codes that are currently under development and used within the international community for LWR analysis. For the purposes of this work, the NRC code FRAPCON is used; however, there are many other fuel codes with different methods and degrees of fidelity to the basic physics used in the codes. For the sake of completeness, brief descriptions of the other available tools and some of their applications are included.

2.2.1 FRAPCON

FRAPCON-3.4 is the steady-state fuel rod performance code with development sponsored by the U.S. NRC and which is currently maintained by Pacific Northwest National Laboratory [5]. It is capable of simulating steady-state (depletion) and slow transient (on the order of minutes to hours) fuel behavior. The modeling approach is considered "one and half" dimensional, in that the coolant temperature distribution is computed up the subchannel surrounding the fuel rods. The heat conduction solution is radial, but the stacked nodes are coupled through the coolant temperature distribution and the computed axial elongation of each node. FRAPCON-3.4 models the following phenomena/behavior: conduction of heat generated in the fuel through the fuel, fuel-cladding gap, cladding, and into the coolant; plastic and elastic deformation of the fuel pellet and cladding; fuel-cladding mechanical interaction (FCMI); fission gas generation and release into the gap and fuel rod plenum; and cladding oxidation and hydrogen uptake. Detailed descriptions of these calculations will be provided in the next chapter; here we provide a brief summary.

NRC/PNNL have published numerous reports describing the methods within FRAPCON, comparisons of FRAPCON predictions to experimental data, and applications of FRAPCON to fuel licensing efforts. For example, FRAPCON-3.4 was used to support the design certification of Westinghouse's AP-1000 [22], General Electric's ESBWR [23], and Mitsubishi's US-APWR [24]. It has also been used for power uprates [25] and new fuel designs [26].

There are numerous additional applications and modifications to FRAPCON described in the literature. FRAPCON-3.3 was used by Herranz, et al, in a code-to-code comparison study of power ramp scenarios within the SCIP program (along with FALCON-PSI, ALCYONE-V1.1 and STAV-7.3) [27]. The comparisons included prediction of cladding oxidation, diameter, and elongation for four high-burnup fuel segments that underwent ramp tests in the Studvik R2 reactor. In general, FRAPCON-3.3 did not perform as well as the other codes investigated in predicting the phenomenon investigated; as the authors pointed out, the type of ramps simulated were well outside the user guidance for maximum allowable local power change within a time step. Additionally, there were limitations in the mechanical models within FRAPCON-3.3 that only considered data from the 1970's and 1980's. The version of FRAPCON used within this thesis (FRAPCON-3.4) includes modifications that have addressed some of these concerns [28].

FRAPCON has also been modified for more advanced fuel applications. For example, Karahan, et al, extended FRAPCON into FRAPCON-EP for high temperature and high burnup applications [29]. Feng, et al, then extended and applied FRAPCON-EP to model the behavior of uranium nitride fuel [30]. FRAPCON has also been modified to model thorium oxide based fuel (Loewen, et al [31]; Long, et al [32]). Furthermore, Vega et al modified FRAPCON to utilize atomistic UO_2 thermal conductivity and thermal expansion models; they found that, while radial temperature profiles were well-predicted, the pellet surface displacement was not. The shortcomings were likely due to shortcomings in capturing the fuel inter-atomic potential [33].

A study using FRAPCON/FRAPTRAN that is of particular interest in this work was conducted by Geelhood et al [28], in which the models and correlations in FRAPCON-3.3 and FRAPTRAN-1.3 were investigated for conservative bias and compared with the most recently available data within the open literature. The models that required modification to better capture their intended phenomena were updated in the FRAPCON-3.4 and FRAPTRAN-1.4 code releases (in 2011). In addition, a sensitivity study of various manufacturing and modeling parameters was carried out to determine the models and input data that had the most influence over the outputs of regulatory interest (such as oxide thickness, cladding hydrogen content, maximum cladding strain, peak rod internal pressure and maximum fuel temperature). The results of this study lead to the inclusion of eight additional "uncertainty parameter" inputs in FRAPCON-3.4 which allow the user to modify certain models within the computational scheme (for example, allowing the user to bias the fuel thermal conductivity up or down a given amount). Using these additional inputs, it is possible to perform a statistical evaluation of fuel rod performance and to determine distributions and confidence intervals on possible outputs. This capability will be employed within the work described in the subsequent chapters.

2.2.2 FRAPTRAN

FRAPTRAN-1.4 is the transient companion code to FRAPCON, which is used to predict fuel rod performance under rapid transient conditions (i.e., on the order of seconds to minutes) [34]. It computes the temperature and deformation history of a fuel rod undergoing rapid changes in flow conditions and/or power level. The phenomena modeled by FRAPTRAN includes: heat conduction; heat transfer from cladding to coolant; elastic and plastic fuel and cladding deformation; cladding high temperature oxidation; fission gas release; and fuel rod gas pressure.

FRAPCON is designed to be used as an initialization tool to provide high-burnup initialization for transient FRAPTRAN calculations.

One application of FRAPTRAN in the literature is Geelhood and Beyer using the code to develop a best-estimate failure criterion for enthalpy deposition during a Reactivity Initiated Accident (RIA) [35]. Related to this effort was a study undertaken by Pagani and Apostolakis to generate a probability distribution of fuel failure due to enthalpy deposition for high-burnup fuel using FRAPCON and FRAPTRAN [36]. This example is of particular note because it includes consideration of spallation of the cladding material (i.e., for highly corroded fuel rods a layer of oxide breaks away from the base material) and it provides for an explicit (Monte Carlo-based) statistical treatment for determining the failure threshold. Their conclusions were that the threshold to failure tended to decrease with burnup, and that spallation was an important phenomenon to consider. A final example of not here is the use of FRAPTRAN to model the fuel behavior during power oscillations during a BWR instability event [37]; in this work Hu and Kazimi showed that under the conditions of the particular oscillations considered, both the stress-strain and the fatigue limits of the cladding would be easily met.

FRAPTRAN is not used implemented in the current work, but it is possible that in the future transient simulations using it coupled to the PARCS code for power information, initialized to a specified burnup with the FRAPARCS system, could be developed.

2.2.3 FALCON

The industry counterpart to the NRC's FRAPCON/FRAPTRAN codes is the Electric Power Research Institute code FALCON [38]. FALCON combined the steady-state and transient capabilities of their previous codes ESCORE and FREY, respectively, to give finite-element based fuel performance predictions on a unified framework. It is capable of either 1 or 2-dimensional (R-Z or R-θ) geometry, and uses material properties from MATPRO [39] (which is also the source of many of the material properties in FRAPCON/FRAPTRAN) augmented with additional models from the open literature or from the Nuclear Fuel Research Institute program.

FALCON's primary user community exists within the nuclear utilities, but there are some examples in the scientific literature describing applications. For example, Mai, et al, used FALCON to evaluate the UO_2 fuel creep model currently included in the MATPRO database [40]. The determination was that, based upon comparisons with data, a modified fuel creep

model provided closer agreement with experiments and captures the fuel creep phenomenon more accurately. Other work includes the work of Khvostov, et al, implementing the GRSW-A model for high-burnup fission gas release in the FALCON code, which led to greatly improved prediction of FGR phenomena compared with experimental results [41], and other work by Khvostov et al in conjunction with the Halden Reactor Project examining the importance of axial redistribution of fission gas during LOCA events [42], and the implementation of the two-dimensional capabilities to evaluate cladding failure from pellet-cladding interaction (PCI) due to missing pellet surface [43].

2.2.4 INTERPIN

Another code that is used within the nuclear industry to model fuel pin performance is INTERPIN, which is a subcode within the CASMO/SIMULATE core design suite developed by Studsvik [44]. INTERPIN is used both at the lattice level during cross section generation with CASMO to automatically adjust fuel pin dimensions based on the user-input temperature conditions and at the core simulator level to generate gap conductance vs. burnup and linear power rating tables used by the core simulator SIMULATE to improve prediction of Doppler temperature. The range of fuel pin modeling within INTERPIN is limited, and does not consider many of the fuel phenomena that are important in the temperature calculation (particularly microstructure changes and plastic deformation/creep). However, Grandi and Hagrmann recently reported improvements to INTERPIN that include modifications for high-burnup and mixed-oxide fuels (including updating of fuel thermal conductivity, waterside corrosion, intra-pellet power distribution, and gaseous gap conductance) that allow it to match measurements of core critical power and axial offset during coastdown of cycle 18 of the Ringhalls Unit 4 PWR [45]. This work demonstrated the importance of fuel thermal-mechanical behavior in core simulation.

2.2.5 Advanced Fuel Codes

There are two advanced fuel performance codes currently under development by the national laboratories which will be briefly mentioned here. First, BISON is the fuel rod code developed by the Idaho National Laboratory [46]. Relative to codes like FRAPCON and FALCON, it is very advanced, capable of full three-dimensional finite element simulations of nuclear fuel rod behavior. The code is capable of simulating each fuel pellet within the rod individually. Like other fuel codes, BISON currently employs many empirically-based models to

capture individual fuel phenomena, but the framework is in place to include additional, more first-principles-based (i.e., meso-scale) modeling using the MARMOT code [47]. BISON has been applied to several practical LWR phenomena such as Pellet-Clad Mechanical Interaction (PCMI) [48].

The final code that will be discussed to describe the breadth of fuel modeling tools currently available is the Advanced Multi-Physics (AMP) fuel performance code developed at Oak Ridge National Laboratory (ORNL) [49]. This code is similar in capability to BISON, with the additional ability to employ the ORIGEN-S depletion code to track the isotopics within the fuel pellets as they undergo irradiation. AMP has been compared to the IFA-432 experiment carried out at the Halden test reactor and has demonstrated results within the experimental uncertainty. Currently development is underway to include internal neutronics and thermal hydraulics directly within the code rather than rely on empirical models and external boundary conditions.

The previous discussion provides a brief overview of the different codes and methods currently in either widespread use or under development to model nuclear fuel rod behavior. The following section will review the methods currently in use or under development to perform the neutronics solution for reactor core simulation.

2.3 Neutronics Analysis Codes and Methods

In order to evaluate the depletion of a nuclear reactor, it is important to accurately determine the neutron distribution within the reactor in order to predict reaction rates, which then allow for accurate prediction of local power and fuel depletion. In order to perform this evaluation, the Boltzmann Transport Equation is solved in space, energy, angle, and time. The Bateman equation is then solved to determine the change of the nuclide densities in space and time. In principle, these are relatively simple equations to solve; unfortunately, due to the complex spectral behavior of nuclear cross sections and the high degree of spatial non-uniformity in the reactor design itself, full core solutions of the transport equation is very difficult and in order to make depletion calculations feasible, a number of approximate strategies such as diffusion theory have been developed. In this section, we discuss some of the more popular codes and methods available to perform these computations. Also, with the advent of

massively parallel computer systems and highly efficient software, some of the full core problems that were infeasible over the first several decades of nuclear analysis now may be solved with more direct methods. Although these advanced methodologies are not used in this work, for the sake of completeness some of them are described.

2.3.1 Generation of Cross Section Libraries for Lattice Calculations

As discussed in Chapter 1, neutronics calculations of nuclear reactors are greatly complicated by the material heterogeneity of the structure and the spectral complexity of the cross sections which describe the interaction between neutrons and the host material. Therefore, from the early days of nuclear technology through today, methods have been developed to simplify the geometrical and spectral representation of the reactors while attempting to maintain the detailed physics implicitly embedded within. The general approach has been to collapse in energy (from point-wise cross sections to fine-group to broad group) and to homogenize in space (for example, replace an array of fuel pins with a uniform block that is neutronically equivalent) through a series of steps. Classic texts describing this process were written by Stamm'ler and Abbate [11] and Duderstadt and Hamilton [50]; the following discussion is a brief overview of the content within those books that is applicable to the methods used in the research here.

The initial step of generating fine-group cross sections from point-wise data is undertaken with a library processing code like AMPX [51] or NJOY [52], which often employ an infinite-medium approach with dilution parameters that allow for consideration of self-shielding in subsequent steps. This reduces the neutron spectral dependence from essentially a continuum to perhaps hundreds of discrete bins (referred to as energy groups). Once these fine-group cross sections have been generated, a lattice level code is employed to generate broad-group cross sections for a core simulator. The following two subsections describe first a selection of lattice codes in common use for cross section generation, and then a few core simulators that are also currently used for reactor analysis.

2.3.2 Lattice Codes

Once a fine-group library is generated, it is used by lattice codes to generate broad-group cross section libraries for use in core simulators. The lattice code usually represents the detailed geometry of a fuel assembly and solves for the spatial and energy dependence of the neutron flux within the fine-group structure. Using this detailed spatial- and energy-dependent flux, energy

condensation and spatial homogenization is employed to produce broad (for LWRs usually two) group cross sections for use in a diffusion-based core simulator. The collapsing of cross sections is based on the preservation of reaction rates at the detailed level, so that a neutronics calculation at the broad group level will maintain the characteristics of the lattice calculation. Cross sections are typically evaluated at a number of fixed fuel temperatures, moderator temperatures and densities, control rod states, and/or soluble poison concentrations, out to a specified burnup, in order to envelope all possible states the reactor core might experience during depletion. The key approximation here is that each lattice calculation is usually performed assuming reflective boundary conditions (i.e., the assembly is depleted within an infinite sea of itself). At the core level, of course, this is not the case; assemblies of different types may be adjacent to one another, and reactor cores are not infinite in extent. That being said, the cross section generation process as described here has proven to be very effective in enabling tractable, sufficiently accurate reactor performance analysis and prediction.

The first lattice code discussed is HELIOS [53], a two-dimensional particle transport code based on the current-coupling collision probability method developed and maintained by Studsvik Scandpower. HELIOS allows considerable flexibility in the geometric representation and material characteristics of the particular nuclear system being studied, as well as a large variety of available output options depending on the specific aspects of the study in question. This flexibility has led to HELIOS being used for many applications, including light water reactor analysis ([54], [55]), high temperature gas reactors [56], CANDU lattice design [57], small thorium-fuel core cross section generation [58], and RBMK calculations [59].

In contrast to the generalized nature of HELIOS, CASMO is a lattice physics tool developed by Studsvik that is primarily geared towards LWR assembly analysis [60]. By tailoring the input structure for such lattices and making heavy use of Studsvik-recommended defaults, cross sections can be generated relatively easily and quickly for core calculations. There are many examples of CASMO usage across the industry, including licensing-basis core analysis for operating reactors ([61], [62]), generation of multigroup libraries for MOX lattices [63], and as part of a computational scheme to perform reactor pressure vessel fluence predictions [64].

A third cross section generation tool commonly employed is the TRITON sequence from the SCALE neutronics code package developed by Oak Ridge National Laboratory (ORNL) [65]. SCALE (Standardized Computer Analysis for Licensing Evaluation) is a suite of

computational methods developed by ORNL under Department of Energy (DOE) and NRC funding to perform a number of different neutronics evaluation, most notably spent fuel pool criticality evaluation (using the CSAS modules), shielding analysis (using the MAVRIC sequence) and reactor physics analysis (using the TRITON module). TRITON is actually a sequence of calls to subcodes within SCALE; for resonance processing (i.e., generating fine-group cross sections for the assembly level transport calculation), the code CENTRM is called. Then, the deterministic transport code NEWT is employed to compute the assembly flux distribution. NEWT is based on the Extended Step Characteristics (ESC) method to perform the spatial component of the flux calculation. Unlike HELIOS or CASMO, which explicitly support curvilinear geometries (such as fuel pins), cylindrical components in NEWT are represented by polygons. When TRITON is executed in depletion mode, a predictor-corrector scheme is used with NEWT in conjunction with ORIGEN-S to perform isotopics within each region in the lattice model based on the local flux. TRITON has been used for LWR lattice cross section generation, HTGR cross section, and spent fuel characterization [66].

There are many other lattice-level codes available for cross section generation (such as WIMS [67], PARAGON [68], or TGBLA [69], but the three detailed above give a representative picture of the methods and applications commonly employed. Within the context of this work, HELIOS is used to generate the cross sections employed in the test problems. Next, an overview of core simulators is provided.

2.3.3 Core Neutronics Simulators

The purpose of the core simulator is to use the collapsed cross sections generated at the lattice level to simulate the core at a macroscopic level, evaluating the reactivity, power distribution, and nuclide density distribution under nominal steady-state or transient operation. The general approach to simulating an LWR core is to divide it into a Cartesian mesh (typically, each assembly is represented by 2 nodes in each of the radial (x- and y-) directions and approximately 6-inch long nodes in the axial (z-) direction). The two-group neutron diffusion equation is then solved on this node structure, with the fuel and fluid conditions within each node determined based on the power distribution. Typically this is solved via an iterative process, because the nuclear parameters (and thus power distribution) within each node depend on its temperatures and densities, and the temperatures and densities are determined by the power

distribution. The collapsed cross sections generated by the lattice code are interpolated upon to produce their appropriate values based on the conditions actually computed by the core simulator. Typically, simulators will have simplified thermal-hydraulics and conduction and convection solvers contained within them in order to determine the temperature and density conditions.

There are a number of different codes available to perform these evaluations. The code that is used within the context of this work is PARCS, developed at the University of Michigan for the NRC [2]. PARCS solves the two-group neutron diffusion equation to perform steady-state and transient simulation of nuclear reactor cores. The neutronics methods are based on the coarse-mesh finite difference scheme, with nodal updates to correct for node-to-node leakages. The nodal calculations can be carried out with either the Analytic Nodal Method (ANM) or the Nodal Expansion Method (NEM), depending on the nature of the problem to be solved. As with FRAPCON, further details on the specifics of PARCS's schemes will be provided in Chapter 3.

PARCS has been used throughout the international community for reactor physics analysis, particularly for coupled neutronics/thermal-hydraulics transient and accident evaluations, which will be discussed in subsequent sections. Some examples of applications of stand-alone PARCS include Demaziére, et al, evaluating PARCS predictions against in-core measurements taken at the Ringhals-3 PWR and the Forsmark-2 BWR [70]. The conclusions of this work indicated that in both the PWR and BWR cases, the axial power distribution was predicted quite well by PARCS. However, for the PWR, while the radial power distribution was generally predicted accurately, there was a tendency for PARCS to overpredict power in the center of the core at beginning of life, and underpredict core power in the center at end of life. With respect to the BWR measurements, the radial power distribution was not captured nearly as well as with the PWR, particularly in the regions nearest to control blades. This may have been due to limitations with the representation of multiple composition control rods.

PARCS was also assessed by Hursin et al [71] using a more advanced full-core transport code DeCART for a rod ejection accident. The object of the study was to quantify the differences in local pin power computed by a reactor analysis tool based on the traditional approach against a higher-fidelity model. The comparisons indicate that the solution provided by PARCS was in excellent agreement with the DeCART results, except in the prediction of the control rod worth; PARCS slightly overpredicted worth relative to DeCART, and therefore predicted a slightly

more severe transient. Other assessments of PARCS include work by Diamond, et al, who compared PARCS solutions to a rod ejection with the BARS and CHRONOS codes [72]. BARS employs a Greens function based approach and explicitly models each fuel pin in the radial plane, whereas CRONOS employs a nodal representation of the core similar to PARCS. The conclusion from this study was that all codes were in very close agreement (within 3%) of pinwise enthalpy deposition. There are other examples of PARCS use in the literature, but this listing serves to illustrate some of the applications of the code.

There are other core simulators in common use within academia, including the NESTLE code developed for DOE by North Carolina State University [73]. NESTLE is also a nodal diffusion code which can used in two or four energy groups. The final core simulator mentioned here is the Studsvik code SIMULATE, which (like their lattice code CASMO) has found widespread use in the nuclear industry [3]. A kinetics version is commonly used for reactivity accident analysis [74].

The above discussion highlights methods in use for neutronics analysis in the nuclear community that characterize the traditional, multi-step staged approach used for core simulation. For the sake of completeness, a brief discussion follows of some more advanced, explicitly transport-based methods for core simulation.

2.3.4 Advanced Neutronics Codes

The primary motivation for developing the multi-step reactor analysis procedure described above was the computational cost of directly evaluating the Boltzmann equation in a reactor core with the full level of spatial and spectral detail; early computers simply did not have the processing power to generate transport solutions, and therefore approximate methods were necessary. With the advent of high performance computing and massively parallel computers, it has become feasible to perform full-core transport calculations with pin resolved detail. The following discussion highlights one of the tools currently under development under the auspices of the CASL program [75].

The MPACT [76] code has been under development at the University of Michigan to provide detailed pin resolved neutronics solutions for full core problems. MPACT uses two-dimensional Method of Characteristics (MOC) in the radial (planar) direction, coupled with an lower order axial solver. It is capable of both steady-state and depletion calculations, and several

recent applications of this code system to typical PWR problems has demonstrated a factor of at least a factor of 2 improvement in the accuracy of predicting the pin-wise power distribution and critical condition of the core [77] compared to the current generation of nodal core simulators. This code typically request thousands of processors for the core calculation, and obviously this kind of computational machinery is unavailable for the vast majority of the neutronics community. However, MPACT does provide a demonstration of what may be possible in the future.

In this section, both the traditional, multi-step, and more advanced, transport-based core simulation approaches and tools were described. Within the context of this work, the HELIOS/PARCS sequence will be used to provide neutronic information during depletion calculations. In the next section, we highlight the methods that have been developed to perform multiphysics simulation of reactor processes.

2.4 Multiphysics Calculations

Now that the capabilities available within individual code systems have been described, the next topic relevant to this thesis research is the use of computer codes to represent various physics simultaneously. In the past, the physics represented within individual codes was often limited to a particular class in order to maintain a reasonable level of complexity in the equations and solution methods for the computational resources available at the time of their development. This separation of physics is referred to as "operator splitting" and is very common within codes in use today [10].

As computational resources improved and nuclear plant operators were motivated to remove some of the conservatisms that are required to ensure safety margin when using operator-split methods, research began on how to incorporate a tighter coupling between the different fields present in a nuclear system. The move towards multiphysics evaluations gained international attention in the mid- to late-1990s with the establishment of a PWR Main Steam Line Break (MSLB) benchmark problem with IAEA/NEA [17]. Following that benchmark problem, there were specifications for a PWR Rod Ejection Accident (RIA) benchmark [78], as well as BWR Turbine Trip (TT) benchmark [79]. Most of these efforts involved the coupling of a core simulator with a system-level thermal hydraulics code such as RELAP5 [80] or TRACE

[81], but with the advancement of methods and techniques coupling principles have been applied to a variety of problems beyond these. In the following sections, a survey of some of these multiphysics methods and applications will be discussed. First there will be descriptions of some of the general approaches to coupled code calculations. Then, some applications in coupled neutronics/thermal-hydraulics work are discussed, and finally some description of efforts within neutronics/fuel thermal-mechanics is presented.

2.4.1 General Approaches

As stated above, the first wide-spread multiphysics approaches were developed for application to certain classes of safety analysis in which the interaction between the core power distribution and the fluid and thermal conditions within the reactor system were deemed to be important. Most of these coupling strategies fundamentally consisted of the core simulator and the thermal-hydraulic code executed in parallel, with a data transfer architecture that allowed for information to be shared between the two codes via either file transfer or a message passing interface [82]. Alternatively, as has been done with TRACE and PARCS, one of the codes can be embedded as a library within the other. In any case, in explicit code coupling one code serves as the "master," driving the calculation (usually the code that required the smaller timesteps for convergence) while the other serves as the "slave." Figure 2.2 (Figure 1.7 from [10]) provides an illustration of this process.

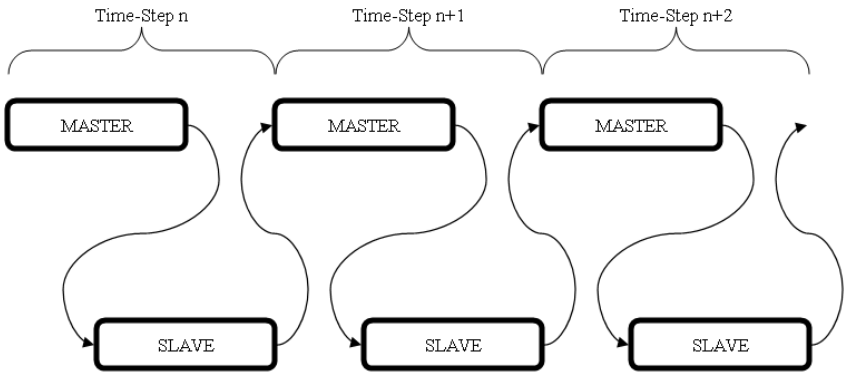


Figure 2.4.1: Illustration of master/slave code coupling arrangement

As outlined by Zerkak et al [10], it was upon this framework that many of the initial code coupling efforts were based. This was primarily due to the relative ease in which coupling could

be achieved, since modifications to the respective source codes (i.e., their intrinsic solution methods) could be minimized, thereby leveraging existing technology. However, there are some limitations to this method, particularly with respect to properly resolving the non-linearity that exist within the two (or more) fields represented by the codes, which leads to fundamental limitations on the accuracy of the methods. To address this, one could employ either a staggered time grid (as illustrated in Figure 2.3 (taken from [10]), or one could employ a Fixed Point Iteration (FPI) scheme which supports iteration within each timestep to resolve the non-linearities explicitly. As will be discussed in Chapter 3, the FPI approach is used to couple FRAPCON and PARCS.

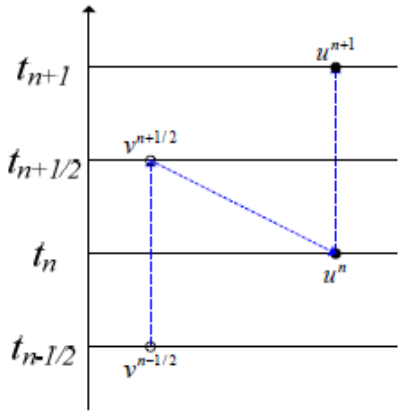


Figure 2.4.2: Staggered time grid method of external code coupling (adapted from [10], *v* and *u* represent fields computed by codes 1 and 2, respectively)

An alternative to performing multiphysics analysis by coupling external codes is to solve all fields simultaneously within the framework of one code [10]. The basic idea is to formulate a large system of non-linear equations considering all the fields of interest. Then, the equations are linearized and Newton's method is employed to find the solution (i.e., forming the Jacobian matrix derived from the system equations and finding the roots). At this point, one could use standard linear solvers to accomplish the Newton iteration; however, the approach most often employed is the Jacobi-Free Newton-Krylov (JFNK) method, which has gained a significant amount of traction over the past few years (consider, for example, [83], [84], and [85]). The basic idea is to project the update vector within the Newton iteration on a Krylov subspace and use a Krylov subspace iterative method (such as GMRES or BiCGSTAB) to generate increasingly accurate approximations to the update vector. The basic advantage to using JFNK is

that it does not actually require the formation of the Jacobian matrix (which can be computationally expensive and memory-intensive); instead, only the action of the Jacobian on the update vector is required. Therefore, the storage requirements can be kept to reasonable levels, even for large systems. The downside is, as with all Krylov subspace methods, that preconditioning is required to obtain convergence within a reasonable number of iterations. However, on the whole it can be said that JFNK methods are powerful tools for solving multiphysics problems, and are very likely to serve as the basis for such computational tools of the future.

To summarize, there are two basic approaches to performing multiphysics computations: coupling external codes which solve fields individually and exchanging data between them, or developing methods which solve all fields simultaneously with advanced numerical methods. There are benefits and drawbacks to each approach, and both will likely be used for the foreseeable future. In order to leverage the existing capabilities of PARCS/PATHS and FRAPCON, we use the former approach in this thesis. In the next section, we will detail some of the efforts that have been undertaken for multiphysics calculations in the nuclear community.

2.4.2 Neutronics/Thermal-Hydraulics

First, we will discuss some of the work that has been accomplished in the coupling of thermal-hydraulics and neutronics codes. To begin, we discuss some of the results obtained from external coupling of thermal-hydraulics codes and neutronics codes. As discussed above, much of the work from the international benchmark specifications led to the application of coupled RELAP5/PARCS or TRACE/PARCS to solving these problems. For example, Kozlowski, et al, published a number of papers describing the application of both coupled code systems to the OECD/NEA MSLB benchmark problem ([86], [87]). Some of the conclusions from this work were that the point kinetics models for MSLB were indeed overly conservative in their prediction of return-to-power scenarios, and that the three-dimensional core power distribution was an important contributor to calculating the correct reactivity feedback. Lee, et al, used the TRAC-M/PARCS code to simulate the Peach Bottom Turbine Trip measurements [88], and found that the coupled code system predicted measured results quite well. The coupled code system was also exercised for more extreme scenarios (i.e., turbine trips coupled with the failure of different mitigation systems) and it was found that direct moderator heating had a strong influence on the

resulting transient. The TRACE/PARCS system has also been used to evaluate BWR instability events [89], Anticipated Transients Without Scram (ATWS) ([90], [91]) and has been benchmarked against the SPERT tests from the 1960's for rod ejection analysis [92]. There are many other examples of explicit code coupling; this listing simply details some applications using the neutronics code employed in this work.

In addition to these explicit coupling examples, there are several approaches under development for fully implicit multiphysics simulation of neutronics/thermal-hydraulics problems, some of which are based on JFNK methods. One of the primary contributing authors in this area is Mousseau, who developed a second-order accurate fully-implicit model for fluid mechanics, heat transfer, and nuclear feedback using the thermal-hydraulics methods from RELAP5 [83]. Within this work, JFNK with physics-based preconditioning was used, and the results for a rod ejection test problem were compared with the traditional operator-splitting method. Based on his work, the JFNK method showed superior convergence in time, allowing for the same accuracy to be gained with larger timesteps relative to the traditional methods. Using this information, Watson adapted a fully-implicit neutron kinetics-thermal hydraulics solution scheme in the TRACE/PARCS framework ([93], [94]), albeit with direct Newton iteration rather than using the JFNK method to solve the fully-coupled nonlinear system. Due to the problem size, one-dimensional test problems representative of BWR and PWR transient scenarios demonstrated the scheme's feasibility, although it was acknowledged that more advanced numerical methods would need to be implemented in order to achieve full-scale reactor simulations. Likewise, a fully-implicit scheme for modeling high-temperature gas reactors was developed by Ward [95]; in this case, both direct Newton and JFNK methods of various types were tested for effectiveness of solution. The conclusions were that the solution of the coupled fields showed improvement in convergence relative to the operator split case, and that for the thermo-fluids equations, the JNFK methods did not perform substantially better than the direct Newton method. The final example discussed here of advanced multiphysics frameworks is the Idaho National Laboratory's MOOSE framework [96], which is designed as a generic tool for solving coupled non-linear equations on an adaptive finite-elements basis using the JFNK method. The specific physics solvers are incorporated as modules that can be chosen depending on the application in question; problems as diverse as pin-level modeling of a nuclear fuel rod,

neutronics/thermal solution of a sodium-cooled fast reactor, or predictions of pebble bed gas reactors can all be achieved through one framework.

This section has discussed a sampling of methods and applications of coupled neutronics/thermal-hydraulics; the next section will highlight some of the efforts that have been employed for multiphysics solution of neutronics/fuel thermal-mechanical problems.

2.4.3 Neutronics/Fuel Thermal-Mechanics

Compared with coupled neutronic/thermal-hydraulic calculations, there are relatively few efforts in the neutronics/fuel thermal-mechanical area. The most prominent example of current methods employed in this area is the INTERPIN code used within Studsvik's core design suite (referenced above); however, within the context of CASMO/SIMULATE, INTERPIN is used a priori to generate gap conductivities parameterized against burnup and local linear heat generation rate, rather than a true simultaneous solution. An example of one-way coupling using more advanced methods was performed by Hursin et al [97] with DeCART and FALCON (both discussed above) to model a rod ejection transient in a PWR. A pin-by-pin rod ejection transient was simulated with DeCART, and the resulting powers and coolant temperatures were exported to FALCON as boundary conditions. FALCON was then used to determine the cladding response in terms of Strain Energy Density (SED). The results of the high-fidelity neutronics solution were compared with those from a PARCS solution using the traditional, two-step approach to generating two-group cross sections. The results indicated that the differences in the energy deposition and SED computed by FALCON were largely due to the differences in reactivity worth of the rods computed by the neutronics codes. Further, there was a significant difference in the mechanical behavior of the fuel rods when azimuthal variation in power distribution within the rod was computed with DeCART as opposed to the azimuthally-averaged treatment that results from pin power reconstruction methods available in PARCS (up to 20% difference in predicted SED, which roughly corresponds with the uncertainty in stored energy computed by FALCON). The authors conclude that this phenomenon should be taken into account in REA analysis, particularly for high-burnup fuel.

In addition to the one-way coupling described above, there are also some examples of two-way external code coupling in the literature. For example, the BISON code is currently being coupled with DeCART to give three-dimensional fuel performance predictions at the fuel

pellet level [98]. Another example is the coupling of the Swedish fuel performance code ENIGMA with the Monte Carlo-based neutronics evaluator SERPENT [99]. The authors performed a single-pin evaluation of coupled fuel behavior out to 52 GWd/MTU burnup in order to study the effect of fixing a priori the fuel temperature and geometry (representative of current depletion methods) as opposed to allowing them to vary with irradiation (which is more realistic). The result indicate that both geometry and temperature changes with burnup are important; relative to the static case, the differences in k-effective were on the order of hundreds of pcm; differences in one-group collapsed fission and capture cross sections were roughly 2-3%. The conclusion that the authors made based on this information was that the effects were, while relatively minor, certainly non-negligible, and worthy of further study. Another notable factor was that the pin-level computation (especially with a Monte Carlo based method) took several days to accomplish.

In this section, the basics approaches for multiphysics computations were discussed, and some applications within the nuclear community were described. It should be clear from this discussion that multiphysics calculations have reached a degree of maturity, particularly with respect to external code coupling of neutronics and thermal-hydraulics codes. Less work has been performed to date on coupling neutronics and fuel thermal-mechanical behavior, and the methods employed for this purpose are generally of an "advanced" nature. Therefore, one of the unique aspects of this work is achieving coupled simulation with current-generation tools. In the next section, methods to quantify the uncertainty and sensitivity of engineering calculations will be highlighted.

2.5 Sensitivity and Uncertainty Methods

This section will review the research in the final topic relevant to the work performed in this thesis, which is sensitivity and uncertainty analysis methods (S/UA). These are two related but distinct ideas; in sensitivity analysis, the objective is to characterize the influence that the input parameters have on code output, whereas in uncertainty analysis, the goal is to characterize the range of possible output behavior due to uncertainty in the input data. In order to minimize the uncertainty in the output, it is necessary to reduce the uncertainty in the input; sensitivity analysis allows for the identification of those parameters which will have the greatest influence

on the computed results, and guide the gatherers of data to where additional knowledge (for example, new measurements) will be most beneficial towards increasing confidence in the simulations.

As discussed in Chapter 1, in the early days of nuclear power, uncertainties were dealt with via conservatism; the input parameters were biased in the "worst" direction depending on the output of interest. This approach, while generally justifiable from a safety standpoint, does not lend itself to economical operation of the power plants, nor does it allow for a precise quantification of the margins that exist to the safety criteria. In order to enable more realistic evaluation of the margins that exist in nuclear reactors (particularly for LOCA evaluation), the Department of Energy and the Nuclear Regulatory Commission embarked on a large data collection effort throughout the late 1970's into the 1980's, sponsoring a series of tests all designed to improve the state of knowledge regarding some of the phenomena that were critical in reactor simulation [100]. One result of this effort was the eventual publication of the Code Scaling, Applicability, and Uncertainty Evaluation (CSAU) approach to performing LOCA analysis [13]; within the context of CSAU, a response surface approach was employed, in which the input parameters were ranged in such a way as to create an algebraic structure (the response surface) that represented how the simulation output (in this case, Peak Clad Temperature (PCT)) behaved from variations in the input. Then, Monte Carlo sampling of that response surface enabled construction of the upper bound on PCT at the desired probability and confidence level. Nuclear fuel vendors adopted this, and other, approaches to perform licensing-basis LOCA analyses, which directly contributed to several power uprates over the last two decades [101]. This fact demonstrates just how important the insights gained from uncertainty analysis can be at the applied level; an excellent overview of the applications of this class of methods within the reactor licensing arena can be found in [102].

In this section, some of the general frameworks for S/UA in common use within the nuclear community will be discussed, followed by a sampling of the applications of said methods present in the literature. Cacuci and Ionescu-Bujor provided a general overview of deterministic- and statistics-based S/UA methods in use in the nuclear community ([103] and [104], respectively); here a summary of the most common methods is provided.

2.5.1 S/UA Frameworks

The classic framework for S/UA within the neutronics field is adjoint-based perturbation theory. The basic ideas from this subject area stemmed from the need to perform sensitivity analysis (i.e., estimate the effect of a small system configuration change) without entirely recalculating the system state. The classic example is the estimate of static control rod worth. Suppose it is desired to determine the reactivity effect of removing a control rod some amount from a reactor core. Rather than performing a full neutronics calculation with the control rod inserted and then removed and directly comparing the difference in reactivities to determine the rod worth, the adjoint function of the unperturbed state can be employed, in conjunction with the change in the migration operator resulting from the configuration change (which is known a priori), to estimate to the first order what the rod worth is. A full discussion of this particular problem is contained within [105].

The key aspect of perturbation theory is the use of the adjoint function. The solution of the base neutronics problem from the particle balance equations (whether diffusion or transport) is referred to as the "forward" solution, and the operators in the balance equation and the source term are referred to as "forward" operators and the "forward" source. The adjoint solution and operator on the other hand, are chosen in such a way as to satisfy the following inner product:

$$\langle f^* A f \rangle = \langle f A^* f^* \rangle \tag{2.1}$$

where f, A, f^*, and A^* are the forward solution, forward operator, adjoint solution, and adjoint operator, respectively. By utilizing this definition and a linear Taylor series expansion, the neutron balance equations can be manipulated to provide a first-order accurate estimate of the reactivity change between a base and a perturbed system that only depends on the change in the balance operator and the unperturbed system forward and adjoint solutions. Therefore, explicit evaluation of the perturbed system balance equation (which can be computationally expensive) is not needed. As demonstrated exhaustively by Lewins [106], the adjoint can be interpreted as an "importance" function whose magnitude is related to how significant a region in phase space (space, angle, or energy for transport problems; space or energy for diffusion problems) is to the output of interest.

An excellent survey of perturbation theory as applied to nuclear reactor analysis was given by Williams [107]. Within this work, different approaches for arriving at the adjoint equations are given, and the standard problem to be analyzed (reactivity changes due to small system changes) is discussed thoroughly. From this work, however, the perturbation theory approach is extended to beyond simply reactivity differences; it can be employed to estimate the changes in any arbitrary system output (for example, reaction rates or peaking factors) due to small changes in any input parameter, depending on the particular formulation of the problem in question. Furthermore, perturbation theory is not limited to simply neutronics problems; applications exist within, for example, thermal-hydraulics [108], or any arbitrary framework [109]. This generalization of perturbation theory is referred to as Generalized Perturbation Theory (GPT). Cacuci developed the Forward Sensitivity Analysis Procedure (FSAP) to generalize the forward approach to sensitivity analysis (i.e., recalculation of the system states) and the Adjoint Sensitivity Analysis Procedure (ASAP) to generalize the use of the adjoint in sensitivity calculations [110].

While the adjoint approach can be very powerful, and can give precise values for the sensitivity of output parameters to inputs, it has two main limitations: first, in its most usual form, GPT is formulated as first-order accurate. This means that if the system is highly non-linear, or if the perturbations become large, that the answers it produces can be substantially in error. To deal with this, higher-order approaches have been implemented which keep terms beyond the linear [107]; unfortunately, this approach often requires the solution of additional adjoint equations, which can become computationally expensive. The other limitation to perturbation theory is particularly present within multiphysics analysis; that is, the formulation of the required adjoint equations requires manipulation of at times extremely complex equations. Depending on the specific phenomena involved, it may prove to be intractably difficult (if not impossible) to properly formulate the problem.

An alternative to adjoint-based methods with the benefit of not being conceptually complex are random sampling methods (also referred to as Monte Carlo methods). A general, excellent introduction to the concepts of the sampling approach is contained within [111]. The general idea with these methods is that every input parameter (whether it be model, correlation, or initial condition) has an associated distribution that characterizes the range of its possible values, rather than a fixed value. Each of these parameters has some impact on how the outputs

will behave. In basic Random Sampling (RS), each of these inputs is sampled according to its assigned distribution, and a calculation run. The resulting outputs are tabulated. This process is repeated for a number of realizations, depending on the confidence with which the analyst wishes to characterize the output distributions. After constructing an adequate number of possible outputs, statistical analysis is performed to both assign an uncertainty distribution to the output and to determine the sensitivity of the output to each of the studied input variables.

The benefit to RS is that it is simple to understand and explain, and relatively easy to implement. The drawbacks, on the other hand, are what motivated the development of adjoint-based methods in the first place; that is, depending on the problem, it can be extremely computationally expensive. Since all values of input variables are chosen randomly, a large number of realizations must be performed to ensure adequate coverage of the possible parameter space; the number of required samples for adequate confidence in the results can become intractably large. In order to combat these issues, two modifications to RS have been proposed and developed ([111], [112]): importance sampling and Latin Hypercube Sampling (LHS). The first, importance sampling, is based on the division of the sample space into "strata" from which the samples are taken. These strata can be defined so as to emphasize particular areas within the sample space which are considered to be important to the outcomes of interest. The benefit to this approach is that adequate coverage of the sample space can be ensured; the downsides are that, first, it is often not clear a priori which regions of the sample space should be emphasized, and second, the raw estimates obtained of the output distributions are not unbiased; the way in which the strata were defined must be accounted for in the analysis of the output. A more straightforward (and popular) modification to random sampling is LHS; within this approach, the range of each random input variable is divided into a predetermined number of equal-probability bins. Then, from each bin one value is sampled at random. The resulting collection of sampled variables are randomly combined to create sets of random variables for each realization that (a) adequately cover the sample space and (b) produces unbiased estimated for means and distribution functions of the output. More detailed discussion of RS and LHS is given in Chapter 5.

Another uncertainty analysis technique is the Response Surface Method (RSM), which was the basis for the best-estimate LOCA methodology demonstration presented by Boyack, et al [13]. The classic text summarizing the basis and applications of RSM was composed by Myers in

[113], although here the applications described are primarily for optimization problems based on analysis of experimental data; regardless, the basic ideas have been demonstrated to be very applicable to the types of analysis of interest in the simulation community.

The general philosophy of RSM is to approximate the complex non-linear response of a system by a fitted surface, which can then be used as a surrogate for the "true" response in subsequent analysis, whether it be optimization (selection of operating point which maximizes the desired output) or determination of the likely limits of system behavior. As described in [114], the procedure of RSM can be summarized by the following steps:

1. Select a number of model parameters thought to be important in the analysis. Determine the ranges within which these model parameters may vary.
2. Select a "design point" (usually the nominal values of the parameters) about which the response of the system is of interest. Compute the response at this design point.
3. Recalculate the response of the system over the ranges of the model parameters identified in (1).
4. Using the recalculated results, construct a response surface that approximates the true functional form of the response about the design point. Usually, some sort of regression fit is employed.
5. Use the response surface in place of the original model in subsequent statistical studies to evaluate sensitivities and uncertainties of the response due to the input parameters.

The difference between RSM and sampling-based methods is that there is an intermediate step between the actual physics evaluations (which can be expensive) and the subsequent statistical analysis which gives estimates for the sensitivity and uncertainties in the response. Presumably, the construction of the response surface requires significantly fewer evaluations than a more brute-force sampling strategy for the same level of confidence in the results of interest. As with any method, there are limitations of RSM; first, it is better suited to problems in which there are a limited number of important parameters under consideration. Second, it requires a priori identification of those parameters which are most important in the system response, and since (depending on the specific application) this may not be known in advance, the potential exists to miss important effects. Finally, as with sampling-based methods, the uncertainties and sensitivities of the output are statistical estimates, but not computed exactly (which is an additional advantage to more deterministic methods).

Related methods to RSM are stochastic expansion methods, of which Polynomial Chaos Expansion (PCE) and Stochastic Collocation (SC) are the primary approaches available ([115]

and [116], respectively). In both cases, similar to RSM, a functional expression of the response in terms of the uncertain input variables is sought. The difference is that, rather than use a regression fit, in these approaches the response is expressed as an expansion either in orthogonal polynomials related to the input probability density functions (pdfs), in the case of PCE, or on Lagrange interpolants, in the case of SC. The benefit to using these methods, as will be discussed in detail in Chapter 5, is that sensitivity information can be directly accessed from the coefficients of the expansion (for PCE) or from easily-computed integrals of the response expansion using quadrature rules (for SC). Further discussion of these methods is presented in Chapter 5.

In this section, four general approaches to S/UA were presented: the perturbation theory approach, which employs the adjoint solution; sampling based methods, which evaluate the forward solution a number of times with randomly sampled input parameters; response surface methods, which construct a surrogate model upon which statistical methods can be applied; and stochastic expansion methods, which express the response as a linear combination of expansion polynomials. The DAKOTA package [7] allows users to easily implement any of these methods. Within the past few decades other approaches have been developed, but most of them have their roots in one (or more) of these three methods ([117], [118], [119]). In the next section, some applications of S/UA in the literature are described.

2.5.2 Applications

As discussed previously, some of the most prominent applications of S/UA methods are in the thermal-hydraulics area, particularly within LOCA analysis. The nuclear fuel vendors Westinghouse and AREVA have both developed best-estimate methods for Large Break LOCA predictions. Westinghouse was the first with the development of the response-surface-based CQD methodology [120]; CQD was applied to the AP600 nuclear reactor design and demonstrated the effectiveness of the advanced passive core cooling features that are the unique features of that design [121]. They then developed a LHS-based method (referred to as ASTRUM) [122] that has been applied to a number of operating PWRs seeking power uprates ([123], [124]). AREVA's methodology is also based on the random sampling method [125], and has been applied to operating PWRs [126] as well as the proposed EPR [127]. In addition to the methods developed by the vendors, there are applications of best-estimate methods elsewhere.

For example, Trevidi et al compared Wilkes' [128], RSM, and random sampling with LHS methods as applied to the pressurized heavy water reactor for LBLOCA analysis [129]; they found that the LHS method was superior due to the manageable number of computations required, and the unbiased estimates of cladding peak clad temperature (PCT) that it returned. Finally, some more advanced methodologies for best-estimate LOCA analysis have been explored; for example, Fynen et al explored the use of Unscented Transform (UT) within the context of a sampling-based method for PCT evaluation [130]. The goal of UT is to improve estimates of the mean and variance of the output variables using fewer computer runs. They compared the UT method with a response surface technique and determined that UT can comparable results in a fraction of the necessary calculations required to construct the response surface.

Another subject area where S/UA methods have been commonly applied is within the neutronics domain. As discussed above, many adjoint-based sensitivity analysis methods were developed specifically for neutronics evaluation. For example, Williams developed a depletion perturbation theory that coupled the solution of the transport equation with the Bateman equation (which tracks nuclide densities) for depletion calculations ([131], [132]). By developing the adjoint equations for the coupled field applied quasistatically (i.e., during each depletion timestep the neutron flux was assumed to be constant), he was able to produce a method that is capable of estimating the response of a system which has undergone burnup to the initial conditions without requiring the depletion to be explicitly recalculated. This approach was extended by Yang and Downar with the addition of using an improved flux estimate within the depletion steps [133]. They applied their method to the test problem employed in [132] and found it gave significantly different results; the method was also applied to a full-scale Liquid Metal Reactor problem and the sensitivities they computed to breeding ratio agreed well with exact calculations.

An additional area of S/UA within the nuclear realm is the propagation of nuclear data uncertainties through the reactor design process. The fundamental cross section data that is the starting point for all nuclear analysis, being derived from measurements, have uncertainties associated with them; moreover, the uncertainties in each of the specific types of measured cross sections may be correlated to one another. An international benchmark specification promulgated by OECD/NEA (Uncertainty in Analysis and Modeling – LWR (UAM-LWR)) [18] is an effort

by the international community to systematically address these issues and better quantify the margins that exist in LWRs. Part of this benchmark effort is to address the uncertainties that exist in two group cross sections as the result of the uncertainty in the cross section libraries used at the lattice level to generate them. In order to evaluate this impact, Yankov et al ([134], [135]) investigated two approaches using the SCALE code package: the first approach was using the XSUSA method [136] which employs a sampling-based approach to calculate cross section libraries; for every perturbed cross section set that is used in the core analysis, a corresponding transport calculation at the lattice level must be performed. The second was a "two-step" approach which employs GPT at the lattice level and uses the resulting sensitivities to generate perturbed two-group cross section sets without needing to perform numerous transport calculations. Both a lattice cell and the Three Mile Island core specified in the benchmark problem were investigated for the methods' relative performance. The results indicate that both methods produce similar results, and even though the XSUSA method requires far more transport calculations, with the use of parallel computing the amount of time required to perform the analysis can be equivalent. The two-step method performed well for the problems considered, but some concern about the applicability of GPT-generated data for more heterogeneous problems (such as MOX cores specified in the benchmark) may make sampling based methods like XSUSA more robust.

The final area where some applications of S/UA methods are relevant to this work is in fuel performance evaluations. RSM was employed by Kerrigan and Coleman to evaluate the uncertainty in stored energy calculations for PWR fuels using FRAPCON's predecessor code FRAP-S3 [137]; nineteen input parameters were varied to create a response surface based on 79 FRAP-S3 runs for sensitivity and uncertainty calculations. Stored energy for beginning, middle and end of cycle cores were all considered in the analysis. The conclusions of the study found that stored energy was most influenced by the fuel thermal conductivity, with rod average power and fuel density also being important. The methods described in the paper gave a basis for judging the conservatism in fuel vendor stored energy calculations performed at the time.

Another application of S/UA was presented by Kim [138]; in this work, a hybrid LHS-RSM method was employed to evaluate the 95% confidence upper bound on Rod Internal Pressure (RIP) calculations, and compared with a more simplified statistical methodology based on the system moment method. The hybrid approach involved generating the response surface

with LHS sampling rather than parametric calculations, and then Monte Carlo sampling the response surface. In the simplified methodology, limiting rod power histories were identified a priori and used to estimate mean RIP values and the sensitivity of RIP to the input parameters. The results of the study indicated good agreement in the expected value of RIP between the sophisticated and simplified method; however, the output RIP distribution from the simplified method was broader (indicating more variance in the output) and therefore the estimated 95% upper bound was higher.

A further example of S/UA in fuel modeling was published by Bouloré et al [139]. In this work, LHS was used to propagate uncertainties in fuel thermal conductivity, linear heat rate, fuel-clad heat transfer, pellet geometry, and pellet radial power distribution through fuel centerline temperature calculations using the fuel performance code METEOR V2 within the URANIE framework (an uncertainty framework comparable to DAKOTA), and the sensitivities were estimated using Sobol' indices of first order. In order to provide a physical basis, two experiments were used as the basis for the evaluations: the GRIMOX2 experiment was used to evaluate temperature behavior for UO2 fuel at low burnup, and the REMORA2 experiment was used to evaluate high burnup. The results of the comparisons indicate that the uncertainty in fuel centerline temperature is much less in the low burnup case (15 K vs. 80 K), which is to be expected, and that the majority of the uncertainty came from uncertainty in the linear heat rate; the other prime contributor was the fuel thermal conductivity.

A final example that will be detailed here is the application of a non-parametric order statistics-based Monte Carlo method to provide a 95/95 confidence limit on fuel rod performance parameters for licensing analysis [140]. This method is similar to the methods described above by nuclear fuel vendors for best-estimate LOCA calculations; within this work, a demonstration using a simplified transfer function to stand in for an actual fuel rod code is presented. In addition to steady-state evaluation, the effect of potential Anticipated Operational Occurrences (AOOs) (i.e., postulated transients that significantly increase the rod linear power) was also considered within the methodology. The conclusions of the paper were that the methodology presented a well-defined method for allowing more realistic fuel performance evaluations while maintaining to a high degree of confidence the safety margins to fuel failure that are necessary for licensing analysis.

2.6 Conclusions

In this Chapter, basic approaches to fuel thermal-mechanical and neutronic modeling were introduced, and methods for performing multiphysics calculations using some of these methods were discussed. Several techniques for S/UA were also described, and some of the applications within the context of this thesis were presented. From the literature search presented here, it appears that the specific coupling fuel performance and core simulator codes has not yet been published, and that an uncertainty analysis of that coupled framework as it relates to depletion would be a significant and original contribution to the field. In the following chapters, the methodology for coupling FRAPCON and PARCS/PATHS will be described, and the use of that coupling to generate information about the relationship between neutronic and fuel thermal-mechanical performance will be investigated.

CHAPTER 3

FRAPCON/PARCS/PATHS Coupling Methodology

3.1 Introduction

In order to assess the sensitivity and uncertainty in core depletion calculations due to fuel performance parameters, it is first necessary to develop a tool that couples the requisite fields. As stated previously, the tools that will be used to effect this coupling are the NRC fuel performance code FRAPCON and the core simulator package PARCS/PATHS. The coupled tool is referred to as FRAPARCS. The linkage of the coupled codes to the DAKOTA uncertainty analysis package will also be discussed.

This chapter provides the overview of the methods utilized by FRAPCON and PARCS/PATHS to solve the physics they represent. First, the current algorithms used within the respective codes are discussed, and then, the overall flow of the code coupling is reviewed. The modifications which were made to each code are then described which includes a detailed description of the coupling algorithms and their impact on overall convergence. Finally, the process by which the coupled code system is integrated within the DAKOTA framework is explained.

3.2 Fuel Performance Modeling with FRAPCON

FRAPCON-3.4 models the behavior of oxide fuel rods with zirconium-based cladding alloys under steady-state and slow (order of minutes) transient conditions. Each FRAPCON calculation models a single fuel rod. As discussed in the previous chapter, the process of modeling nuclear fuel performance requires the solution to a system of complex, non-linear

equations. There are many approaches to solving such systems; FRAPCON-3.4 decomposes the system by the class of physics and utilizes an iterative approach to converge all the dependent variables in the analysis. An illustration of the geometry of a FRAPCON calculation is presented in Figure 3.2.1.

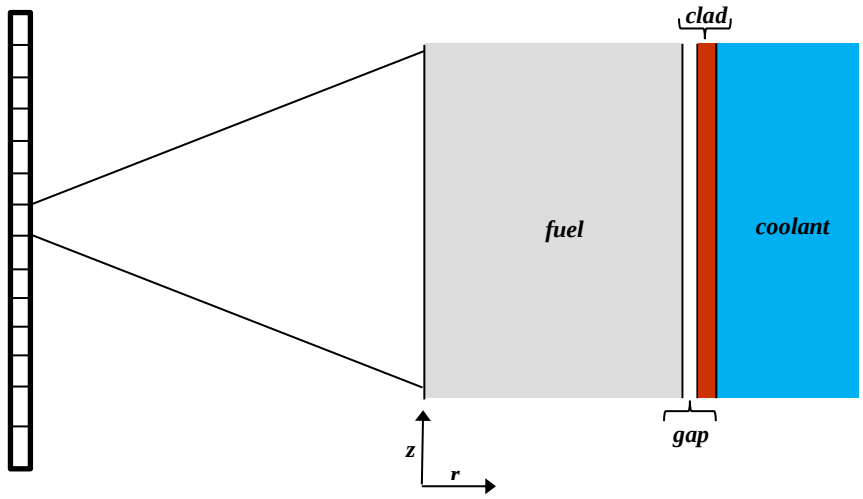


Figure 3.2.1: Geometry of a FRAPCON calculation – fuel pin axial and radial regions

This section first provides details of the FRAPCON-3.4 computational scheme as described in [5]. The discussion here sets the framework for the discussion in Section 3.4.2, which describes the algorithm modifications that were performed in order to couple FRAPCON with PARCS. Also in this section, an uncertainty analysis package developed at PNNL for FRAPCON-3.4 (known as ARM-FRAPCON) is discussed. As part of this discussion, the eight model parameters identified as most important for fuel rod uncertainty calculations [28] are presented.

3.2.1 FRAPCON Computational Scheme

The two major classes of physics solved by FRAPCON are the fuel thermal response and the mechanical response. Other physics represented in the code are primarily captured by means of semi-empirical or empirical relationships which are strongly dependent upon the temperatures and stresses/strains computed by the heat and mechanical response modules. In order to aid the following discussion, a FRAPCON-3.4 simplified flowchart (taken from [5]) is presented in Figure 3.2.2.

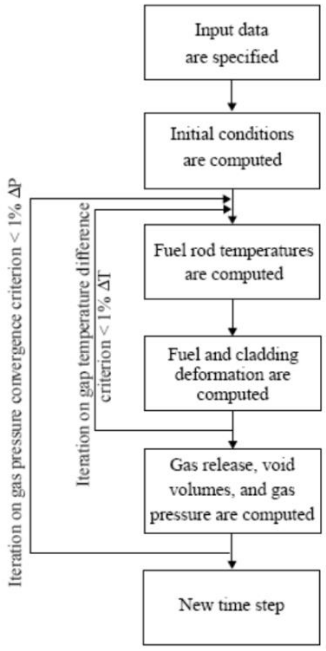


Figure 3.2.2: Simplified FRAPCON-3.4 flowchart (Figure 2.1 from [5])

In the iterative method used by FRAPCON first, the input data, including fuel rod design information (dimensions and materials), time history of the power level and shape, and thermal-hydraulic conditions, are processed by the code. The calculation then proceeds in time, with two major levels of iteration. First, the temperatures of the fuel and cladding are computed using the input power shapes and thermal-hydraulic conditions. Then, the fuel rod mechanical response (i.e., stresses and strains in the fuel and cladding) are computed. Iteration between the thermal response and the mechanical response continues until the temperature drop across the fuel-clad gap is converged to within 1% from the previous iteration. Using the temperatures and strains, the fission gas release, void volume, and gas pressure is then computed. A second level of iteration is used to compute the internal gas pressure; the updated pressure is compared with the previous iterate, and if the difference is greater than the convergence tolerance (again, 1% difference from the previous iteration is used), then the temperature/mechanical iteration is repeated using the new pressure. Once the gap temperature drop and the gas pressure are sufficiently converged, the calculation moves on to the next time step.

Some additional detail on each component of the fuel performance simulation is provided in the following subsections. This discussion is not meant to be comprehensive, but rather to highlight specific topics which are important within the coupling methodology. The primary

feedback between the fuel performance calculation and the neutronic depletion calculation is through the fuel temperatures used to evaluate nodal cross sections (i.e., the Doppler temperature); therefore, the models and methods specifically relating to this calculation are discussed. More detail on all the methods in FRAPCON can be sought from the theory manual [5].

3.2.1.1 Thermal Response

In order to compute the temperature distribution within the fuel rod, FRAPCON-3.4 implements an outside-in approach that begins with the computation of the rod surface temperature. At the beginning of the timestep, the fuel rod axial power distribution is known. If FRAPCON's internal heat transfer modules are used, a closed-channel energy balance is solved to compute the coolant axial temperature profile. Using this information, the appropriate heat transfer coefficient is computed (Dittus-Boelter for forced convection, Jens-Lottes for nucleate boiling), which is then used to compute the cladding surface temperature. Alternatively, the user can specify a cladding outer surface temperature axial distribution. The latter approach is used in this work, in order to leverage the more sophisticated thermal-hydraulics methods in PATHS.

Once the rod outer temperature is computed, the temperature drop across crud, oxide, and cladding materials are computed using the analytic solution to the one-dimensional radial heat conduction equation using the thermal conductivity at the region-averaged temperature. The gap temperature drop (and therefore gap conductance) is computed considering radiation from fuel to clad, conductance through the gap gas, and direct conductance through points of contact from fuel to cladding. The fuel surface temperature is thus obtained by adding the gap temperature drop to the previously computed clad inner surface temperature.

Once the fuel surface temperature is obtained, the solution to the heat conduction equation is obtained for the nodal radial fuel temperature profile. Axial conductivity is considered negligible compared with radial conduction; therefore, within each axial node the one-dimensional heat conduction equation in radial cylindrical geometry is solved. The radial burnup distribution and resulting power profile within each node is computed using the TUBRNP subroutine developed in [141]. This routine tracks the radial concentrations of uranium and plutonium isotopes based on the burnup history of each axial node, and uses a one-group diffusion based analytical solution to compute the resulting radial power profile. The coefficients used within the routine have been adjusted to match plutonium and uranium measurements with

burnup in order to capture the "rim effect" (i.e., power peaking on the periphery of the pellet due to the buildup of plutonium isotopes there). Once the nodal radial power and burnup profiles are known, the finite-difference approach is used to solve the conduction equation for the resulting temperature distribution within the pellet considering the temperature and burnup dependence of the fuel thermal conductivity.

The important elements of the thermal solution considered in this work for uncertainty analysis are the fuel thermal conductivity and cladding corrosion. Also important for thermal evaluation is the heat transfer coefficient, the uncertainty in which is applied through PATHS, since the fixed outer temperature option in FRAPCON is implemented here.

3.2.1.2 Mechanical Response

In order to compute the mechanical response of the fuel rod, FRAPCON-3.4 has two options: the default model is called FRACAS-I, and solves the deformation equations assuming a small displacement and a rigid pellet model. The mechanical simulation of the fuel rod includes separate consideration of the stresses, strains and displacements of the fuel and the cladding. The rod is assumed to maintain its cylindrical geometry throughout the calculation. The general theory of elastic-plastic deformation is implemented, using the method of successive substitutions, to converge upon the computed strains. In summary, this process involves iteration on strain through the successive solution of mechanical equilibrium, stress-strain, compatibility, plasticity, and Prandtl-Reuss flow relations.

In order to simplify the analysis, the fuel pellet is assumed to be rigid, in that it does not deform in response to the cladding. Fuel pellet deformation is restricted to be the result of thermal changes, swelling due to fission product accumulation, and densification. The cladding initially deforms as a thin shell with induced strains and stresses from the internal gas pressure and the external system pressure, until the fuel sufficiently swells to make contact with the cladding (i.e., prior to fuel-cladding contact, the gap is assumed open). Once the fuel-clad gap closes, the clad responds to the fuel's continued swelling by taking the fuel-clad contact pressure into account. In addition, the distinction is drawn between "soft contact," which is the result of the fuel tending to crack and fall against the cladding due to thermal stresses from coming up to operating temperatures, and "hard contact," when the fuel is actually exerting pressure against the cladding. Both the fuel and cladding deforms in both the axial and radial dimension. Under hard contact, it is assumed that there is no axial slip between the fuel surface and the cladding,

and therefore the clad follows the axial expansion/swelling of the fuel. This is an appropriate assumption because the clad (as a ductile metal) is much softer than the ceramic cladding.

In addition to the instantaneous elastic and plastic deformations of the fuel rod materials, there is consideration of thermal- and irradiation-induced creep. In summary, creep is constant-volume deformation as a result of loading over time. It is a stress-driven process that requires additional, empirically-based relations for thermal (due to temperature) and irradiation (due to accumulated neutron fluence) creep. The same elastic-plastic algorithms for mechanical calculations used in FRACAS-I are adapted to perform the creep calculations.

Some of the mechanical models have been found to be important contributors to uncertainty in fuel performance calculations, and have thus have been included in the uncertainty evaluations presented below. These models include fuel thermal expansion, fuel swelling, and cladding creep. Specific implementation of the uncertainties in these models will be provided in succeeding sections.

3.2.1.3 Pressure Calculation

The fuel rod plenum pressure is considered to be a figure of merit in many mechanical calculations, because it provides a driving outward pressure on the fuel rod clad. This could potentially result in cladding rupture, which would provide a release path for fission products into the coolant. Therefore, FRAPCON-3.4 is provided with models that allow for accurate calculation of such a force. Fission gas release is a related phenomenon; gaseous fission products are generated in the fission process but are largely retained within the fuel material matrix. However, under certain conditions they may be released into the free volume of the fuel rod which would provide additional gas inventory and thus potentially increase the plenum pressure. The release rate of these fission products is itself a function of the pressure within the rod. Therefore, in order to accurately calculate the overall fuel rod pressure, the release of fission gases must also be accurately predicted (and vice versa).

The pressure calculation in FRAPCON-3.4 is relatively simple in concept: fundamentally, it is assumed that the ideal gas law holds for fission gases. The free volume of the fuel rod is tracked and used in the calculation, which includes the plenum volume (taking into account the spring which holds the pellets in place), pellet porosity, fuel-clad gaps, fuel pellet cracks, dishes, and surface roughness, and their associated temperatures. Also the gas inventory of the free volume in the fuel rod is required and it is assumed that the fission gas released from

the pellets uniformly mixes with the existing gas (for example, helium fill gas). Once the volumes and region temperatures are obtained, the pressure is a simple calculation.

One of the obvious keys to the accuracy of the pressure calculation is the fission gas release (FGR) model. In its essence, the goal is to predict the amount of generated fission gas that is released from the fuel matrix into the free space within the fuel rod. In order to accomplish this, it is necessary to track the fission gas source term (which is given by the burnup), and then to evaluate the diffusion process of the gas through the fuel grains to the granular boundary, where gas bubbles collect and can either be released or be reabsorbed. There are three models available in FRAPCON-3.4 to predict this phenomenon: the ANS-5.4 standard, a modified Massih-Forsberg model, and the FRAPFGR model. Per PNNL recommendation for standard depletion calculations, the Massih-Forsberg model will be used in this work. Uncertainty in the Massih-Forsberg model as it relates to neutronics performance is given consideration in Chapter 4.

This concludes the overview of FRAPCON-3.4's computational flow and modeled phenomena. To summarize, FRAPCON implements an iterative strategy to solve the fuel rod thermal and mechanical behavior based upon input power distributions and coolant inlet conditions. Additional empirically-based models are implemented to account for material changes (such as cladding corrosion) due to operation at temperature. As background for the research performed here, the existing uncertainty capabilities embedded within the FRAPCON code and some current capabilities to perform stand-alone (i.e., uncoupled) uncertainty and sensitivity analysis are discussed.

3.2.2 Uncertainty Calculations with FRAPCON

As we have discussed in Chapter 2, the industry has moved towards more best-estimate approaches to evaluate the performance of nuclear reactor systems in order to more precisely quantify the margins they have available and to maximize component performance while maintaining safe operation. In order to allow the NRC to respond to such a change in philosophy and to independently verify the results of industry analysis, new model inputs were included in FRAPCON-3.4 that allow for evaluation of the impact of the uncertainty of several key model parameters on the results of interest. From a fuel performance standpoint, these are the outputs that relate to ensuring fuel rod integrity: peak rod pressure, maximum nodal burnup, fission gas

release fraction, maximum cladding strain, and cladding corrosion thickness and hydrogen concentration. Eight key models were chosen from a comprehensive study of the sensitivity of fuel performance to various parameters [28]. Some of these models also relate to the nuclear performance of the reactor system through their contribution to the Doppler temperature; in Chapter 4, we provide evaluations which establish the important factors from this standpoint. For the sake of completeness in the present discussion, however, we will briefly describe all eight parameters available for manipulation. The functional forms and implementation of all of these models are given in Appendix A.

3.2.2.1 Uncertainty Models

Four of the uncertainty models relate to the properties of the uranium oxide fuel material. First, we consider fuel thermal conductivity. This is a basic material property which describes a material's ability to conduct heat. Within a non-metallic material (such as uranium oxide), energy is primarily transmitted by means of phonons generated by the vibrations of atoms within their lattice. The conductivity of UO_2 is known to degrade with burnup for two reasons: first, solid fission products become embedded within the fuel matrix, which tends to interrupt phonon transport within the lattice structure; and second, because the fuel develops cracks and solid-to-solid contact is diminished. At higher temperatures, energy transport due to free electrons also becomes important. The FRAPCON model takes into account these phenomena semi-empirically, as is shown in the model presented in Appendix A. As will be seen in subsequent analyses, fuel thermal conductivity is an important contributor to fuel temperature evaluation.

The second model is fuel thermal expansion. This is related to the differential change in volume due to temperature change; as temperature is increased, the average energy of the atoms within the material also increases, which leads to greater average atom-to-atom separation. Within FRAPCON, the fuel is subdivided into radial rings within each axial node, and the dimension change of each of these rings is calculated based on its temperature. This property is important in the evaluation of the fuel-cladding gap width (and thus gap conductance), at least up to the point where the fuel and cladding come into thermal contact.

The third model is fuel swelling, which is caused by the accumulation of solid fission products within the fuel lattice as a result of the fission process. At first, solid fission products tend to fill whatever residual porosity of the fuel may be left as a result of the manufacturing process, but at a certain burnup (in FRAPCON the value is taken to be 6 GWd/MTU) they begin

accumulating in interstitial locations. This causes the fuel lattice as a whole to swell in size, which, like thermal expansion, can impact the fuel-clad gap.

The fourth model related to the fuel material is fission gas release. Some of the fission products (such as Xenon and Krypton) are released in gaseous form. These gas atoms tend to diffuse to the surface of fuel grains, where they collect as bubbles. These intergranular bubbles collect until a certain saturation concentration is reached, at which point the gases may be released from the fuel matrix into the free space available within the fuel rod. As discussed above, additional gaseous inventory within the rod free space can contribute to an increase in internal pressure, which can influence the fuel rod's mechanical response. The diffusion of gas within the grains is largely a temperature dependent phenomenon. Some athermal release occurs, particularly for fission events which occur near the boundary of a grain, where the initial velocity of a gaseous fission product is sufficient to liberate it directly into the free space. But for most fission events occurring within the fuel matrix, the movement of gas is strongly temperature dependent; the higher the temperature, the greater the gaseous diffusion. Depending on the gas release model in question, the cutoff temperatures for thermally-driven diffusion and release are specified. See Appendix A for details on the Massih-Forsberg model used in this work.

The remaining four models relate to the behavior of the cladding material. Of these, we first discuss cladding creep. Creep is a constant-volume, stress-driven deformation process that occurs over time at temperature. Although there are many creep mechanisms, the most important for cladding materials is dislocation-driven creep, caused at temperature by the climb of dislocations across obstacles and to the grain boundary. This causes a macroscopic elongation of the material over time, which is important in fuel performance calculations. In addition to this thermal creep, irradiation creep must be taken into account; the effect of bombarding the cladding material under stress at temperature with neutrons is to exacerbate the creep which is occurring due to temperature alone, due to additional atoms being dislocated. Both thermally-driven and irradiation creep are considered in this work.

Axial growth is also a cladding property given consideration. Unlike creep, growth occurs due only to irradiation; it can be a product of the manufacturing process. If, in the manufacturing process, narrow metal grains are formed whose long direction is parallel to the direction of the cladding tube, then we can see preferential movement under irradiation of dislocated atoms towards the tips of the grains and their corresponding vacancies towards the

sides. This causes the tube to elongate along its axis. Depending on the cladding material, this effect may or may not be significant; but it is considered here.

The final two cladding models important for uncertainty consideration are cladding corrosion and hydrogen pickup. These are discussed together since hydrogen pickup is related to the corrosion rate. In the presence of oxygen (such as is present in water reactor coolant), zirconium metals corrode. This produces zirconium oxide, which tends to accumulate as a layer on the surface of the fuel rod, and free hydrogen. The corrosion layer has two effects: first, since the thermal conductivity of zirconium oxide is significantly less than that of the base cladding metal, it contributes to temperature increases within the fuel rod. Second, zirconium oxide behaves as a ceramic, rather than a metal, it contributes to embrittlement of the cladding tube. This means that, should the corrosion layer become too thick, the cladding may crack rather than deform as the result of thermal changes, releasing fission products into the coolant. Thus, it is important to calculate the thickness of the oxidation layer within the fuel cladding.

Most of the free hydrogen released in the oxidation process is swept away by the coolant. However, depending on the specific cladding material, some is absorbed into the base metal. In sufficient quantities, the hydrogen within the cladding can combine with the zirconium base metal to create zirconium hydride precipitates. These can also serve to embrittle the cladding, and therefore hydrogen concentrations must be evaluated. In FRAPCON, the amount of hydrogen absorbed is modeled as a cladding-dependent fraction of hydrogen released in corrosion.

This concludes the presentation of the eight models considered to be key contributors to uncertainty in fuel rod performance calculations. Of course, there are other factors (for example, uncertainties due to manufacturing tolerances) that are also frequently considered, but these are the primary models of importance with respect to the physics that a fuel code like FRAPCON represents. Next, we describe a statistical approach developed at PNNL for the NRC to incorporate uncertainties in the fuel rod design process and develop realistic upper bounds on performance parameters and margin quantification.

3.2.2.2 FRAPCON Statistical Wrapper Package ARM

In order to perform uncertainty evaluation of fuel rod performance taking into account the uncertainties in the models described above, the ARM package was developed [142]. ARM is a tool that takes as input a nominal FRAPCON-3.4 input deck and an input file that contains a

description of how each parameter is to be varied. In addition to providing distributions on the eight models described above, the user can specify distributions to use on the manufacturing parameters (for example, fuel rod dimensions) and the power used for the calculation. With this input, the user provides the number of calculations to be run. ARM then spawns this number of unique FRAPCON-3.4 input decks that adjust the input according to the input distributions on each of the parameters. Once each deck has been run, the statistics software package R is used to generate correlation plots and quantify the output distributions based on the values of the input parameters.

The ARM-FRAPCON package has been used by NRC staff in licensing reviews [143]. It is certainly an improvement over standard nominal single FRAPCON calculations in that both the mean and the standard deviation of the output parameters due to the uncertainty in the input parameters are now estimated. However, since the method is not well-documented in the open literature, and because the package is limited at present to using simple Random Sampling, an alternative was sought.

A more general uncertainty evaluation package in common use today within the nuclear community is DAKOTA, developed by Sandia National Laboratories [7]. This package allows the user to implement a number of distinct uncertainty quantification techniques, all the while requiring very little modification to the driver analysis code. The specific implementation of DAKOTA within the context of this work will be discussed in Section 3.5.

Within this section, the methods behind FRAPCON-3.4 were discussed, along with the specific models in the code regarded as important for fuel sensitivity and uncertainty calculations. Additionally, the statistical wrapper package ARM-FRAPCON was described, which has been used by NRC staff to perform estimates of the code uncertainty for licensing evaluations. In the next section, the capabilities and methods of the core simulator package PARCS/PATHS are described.

3.3 Reactor Core Simulation with PARCS/PATHS

Now that the methods and relevant models in the fuel performance code have been described, the methods will be described which will be used to compute the nuclear and thermal-hydraulic behavior of the reactor during depletion. As described in Chapter 2, there are many

methods available for these calculations; within this work, the PARCS code is used for neutronics evaluation, with the PATHS sub-code providing the thermal-hydraulic state of the core during each depletion step. In this section the specific approaches encompassed within these codes are described.

3.3.1 PARCS Methods

The purpose of PARCS is to evaluate core criticality and neutron flux distribution. For LWR analysis, this is accomplished by using a nodal approximation to solve the two-group neutron diffusion equations. PARCS is capable of both steady-state (eigenvalue) and transient neutronics evaluations; for the purposes of this work, the steady-state methods will be used in the fuel depletion analysis. Although depletion is a time-dependent process, the changes in flux and nodal nuclide densities are sufficiently slow that a quasi-static method is employed (i.e., a sequence of steady-state flux calculations).

To facilitate the discussion of the methods in PARCS, a flowchart of the computational logic is presented in Figure 3.3.1. The specific details on the implementation of the methods used in PARCS can be found in the theory manual [2], but here an overview of the process is provided. In order to understand the PARCS flux solution methodology, it is necessary to note that the code performs this flux calculation at two levels: the "global" flux solution is provided by a coarse-mesh finite difference (CMFD) method. However, it is well established that the accuracy of the finite difference method using node sizes typical of core simulators is inadequate; therefore, the CMFD calculation is supplemented periodically with local nodal calculations at each of the nodal interfaces (the default is to use a hybrid Analytic Nodal Method (ANM)/Nodal Expansion Method (NEM) approach) to correct for node-to-node leakages with a much higher order of accuracy than CMFD alone. This method allows for both the efficiency of CMFD and the accuracy of the nodal methods while limiting the drawbacks of both.

The computational flow of PARCS for steady-state calculations is as follows: First, the input is read and the data structures prepared. If PATHS is used as the thermal solver, initial fuel temperature and coolant temperature and density estimates are obtained by assuming axially and radially flat power distribution. Then, the initial linear system is built based on a CMFD discretization of the two-group diffusion equations. Since the resulting linear system can be quite large (for example, a large BWR may have on the order of 1000 assemblies, each divided into 4

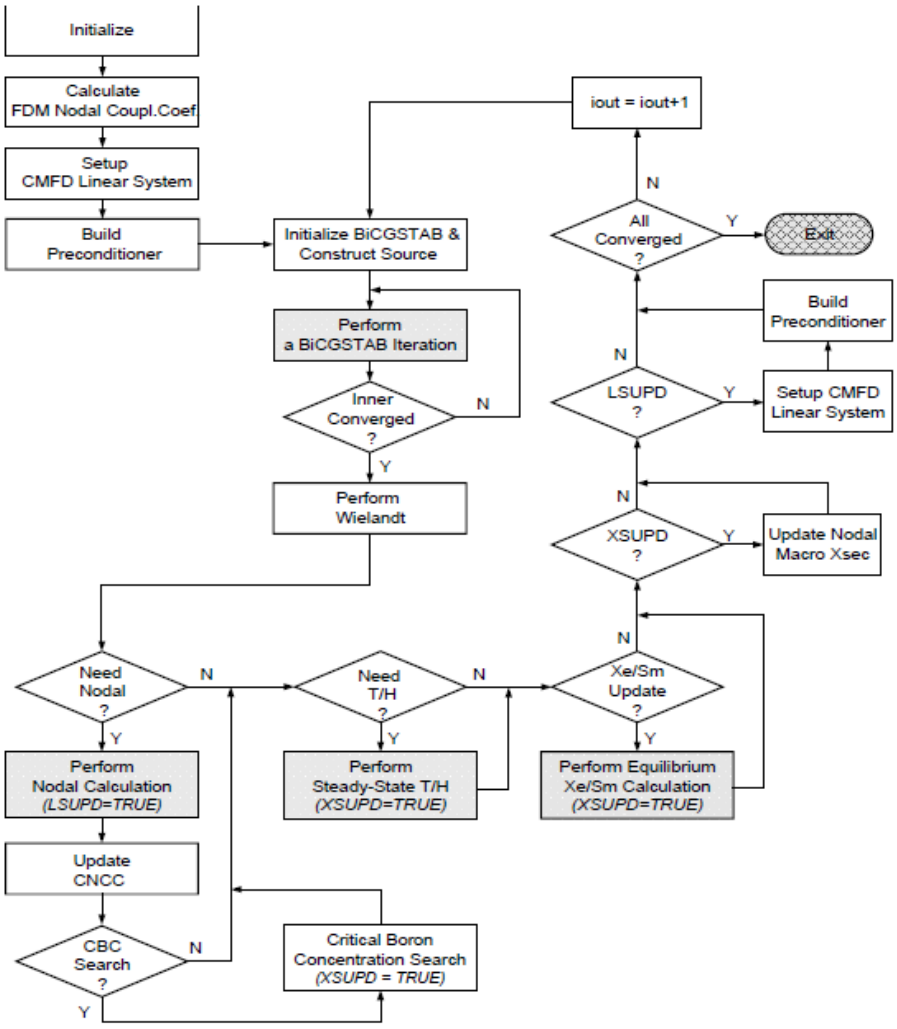


Figure 3.3.1: Steady-state computational flow in PARCS (from [2])

radial nodes and 25 axial nodes, with two energy groups, which results in ~200,000 unknowns), a Krylov method (for PARCS, the default is BiGCStab, with the option to use GMRES) is used to solve for the nodal fast and thermal fluxes. When the iteration has converged, a Weilandt shift of the k-eigenvalue is performed to accelerate convergence of the outer iterations. Then, if needed, local nodal calculations are performed (generating Corrective Nodal Coupling Coefficients (CNCC), which correct the assembly-to-assembly leakage), along with a critical boron search if required. Following the nodal and boron update, the thermal-hydraulics calculation is repeated with updated power distribution to obtain updated fuel temperature and coolant temperature and density distributions. Then, the nodal cross sections are updated, the linear system rebuilt, and if sufficient convergence of the eigenvalue and local and global fission

sources have not been obtained, the process is repeated. For depletion calculations, once the converged power distribution for a state in time has been obtained, the local power density and nodal fuel mass is used in conjunction with the user-specified time step length to compute the core burnup distribution at the beginning of the next time step.

The coupling algorithm described below in Section 3.4 does not impact the computational flow of the neutronics solution described in Figure 3.3.1; rather, the interaction of the depletion and fuel performance calculations occurs within the thermal-hydraulics solution performed by PATHS. In the next section, the methods within PATHS to generate coolant and fuel temperature and density distributions are described.

3.3.2 PATHS Methods

In the initial development of PARCS, a simplified thermal hydraulics treatment based on a closed-channel energy balance in the coolant and a heat conduction solution in the fuel was included to provide the nodal temperature distributions required for cross section evaluation. While this method proved sufficient for PWRs (in which the coolant remains subcooled, single phase liquid), it was not able to provide an axial void distribution which was required for BWR depletion calculations [144]. Therefore, a new steady-state thermal hydraulics solver based on the four equation "drift flux" model was developed to provide an efficient and accurate means to compute nodal flow conditions.

In general, simulating fluid flow and heat transfer involves the solution to mass, momentum, and energy conservation equations. In reactor modeling, two-phase water flow is often represented by a six equation model with separate representations of the fluid and vapor phases which interact through the source and sink terms (ex. TRACE, RELAP5). While this approach is effective and commonly employed in transient simulations, it would be computationally expensive to perform "null transients" at each burnup step to perform core depletion analysis. Thus, to perform depletion calculations, the PATHS code was developed based on Ishii's 4-equation drift flux model [6]. Within this framework, mass continuity, momentum, and energy equations are solved for the liquid-vapor mixture (rather than the phases individually), with the effects of phase separation accounted for with closure models based on the flow conditions. The computational flow of PATHS is shown in Figure 3.3.2:

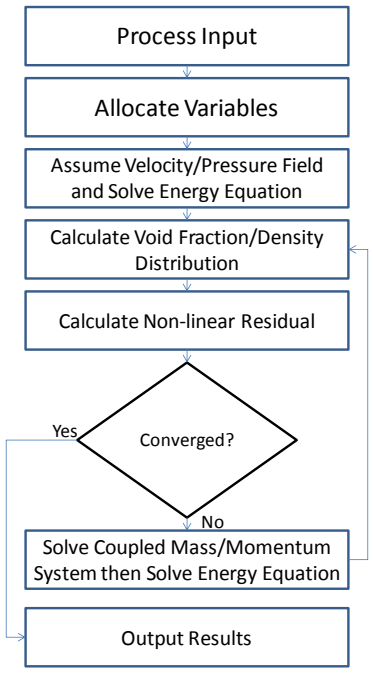


Figure 3.3.2: PATHS computational flow (from [6])

Within PATHS, an iterative process is used within each state point to resolve the fluid profile up each coolant channel. Within a calculation, first, velocity and pressure fields are assumed in the channels. The mixture energy equation is then solved to give an initial estimate of the enthalpy distribution. Then, the mixture mass and momentum equations are solved simultaneously using the Krylov-based Bi-Conjugate Gradient-Stabilized (BiCGStab) algorithm. Using the pressure and velocity fields resulting from this calculation, the energy equation is solved again to update the enthalpy distribution. This process is continued until the pressure, velocity, and density fields have converged. Once the fluid solution has been obtained, the heat transfer coefficient for each node is determined based on the flow conditions, and the temperature drop from the coolant to the cladding surface is determined. Then, the temperature drop across the clad and the fuel-clad gap are calculated, followed by a conduction solve to obtain the temperature distribution within the fuel pellet. This information is used to compute the effective Doppler temperature in the node.

3.3.3 PARCS/PATHS Coupling

Within PARCS, PATHS is called as a subroutine to provide coolant temperature and density and fuel temperature for nodal cross section evaluation. The flow chart which details PATHS's place within this sequence is presented in Figure 3.3.3:

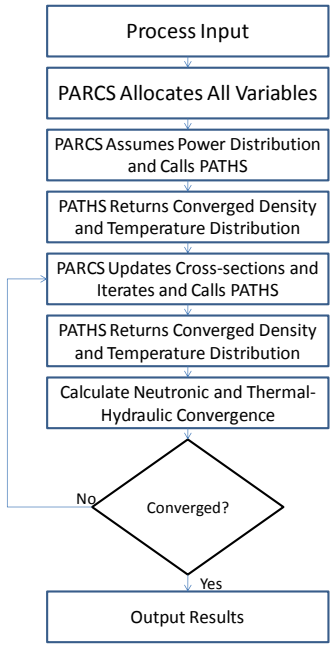


Figure 3.3.3: Coupled PARCS/PATHS program flow (from [6])

Within the context of a depletion calculation, PARCS and PATHS iterate within a depletion step until the temperatures, densities, and neutronic unknowns (power distribution and criticality) are converged. In this section, the methods currently in PARCS/PATHS for depletion calculations were described. In the next section, the scheme to couple PARCS/PATHS with FRAPCON and the algorithm modifications necessary to achieve the coupling will be described.

3.4 Algorithm Modifications for Coupling FRAPCON and PARCS/PATHS

In order to achieve coupling between FRAPCON and PARCS/PATHS, modifications were necessary to the algorithms employed by both codes. The overarching philosophy of this effort was to minimize the invasiveness of the required modifications; the goal was to couple the codes with as few source code modifications as necessary in order to ensure preservation of the physics represented within them. In this section, the implemented coupling scheme is described,

by first describing the overarching coupling philosophy. Then, the specific modifications to FRAPCON and PARCS/PATHS to achieve the coupling are discussed. Finally, the FRAPARCS driver script which implements the modified code versions is described.

3.4.1 Overview of Coupling Scheme

There have been many methods used over the previous two decades to couple different engineering software. For the purposes of this work, primarily due to expediency and simplicity, a simple file transfer technique was implemented. This means that the codes communicate by printing data files which contain information which is read and processed by the other code. The primary information required by PARCS/PATHS for its neutronics calculations are the nodal fuel temperatures used in the cross section evaluation. The primary information required by FRAPCON for its fuel performance calculations are the fuel rod surface temperature and axial power distribution. PARCS and PATHS themselves communicate, with PARCS providing PATHS the power distribution and PATHS returning to PARCS the nodal moderator temperature and densities, again for cross section evaluation. Figure 3.4.1 graphically illustrates the information sharing between the codes.

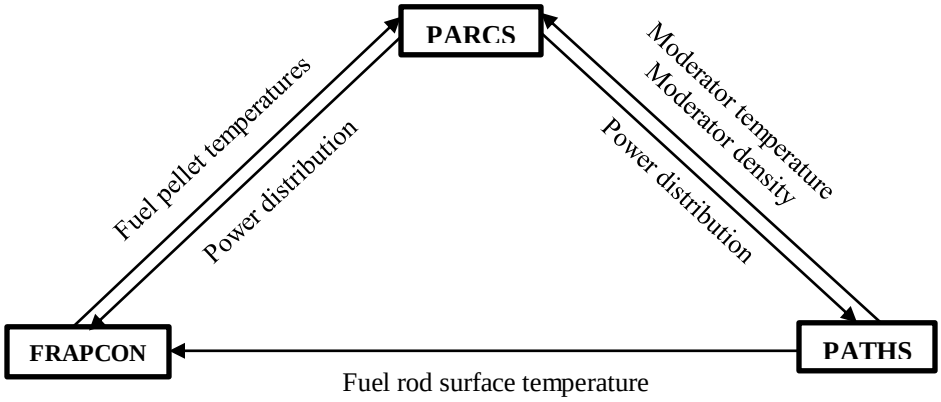


Figure 3.4.1: Information exchange during execution

Generally in coupled calculations, one code is chosen as the "master" to control the execution flow. In this work, PARCS/PATHS is the master, and FRAPCON is the slave. The reason for this is simple: within a calculation, there is one core neutronics calculation, whereas there may be many separate FRAPCON executions, one for each fuel rod explicitly simulated. Therefore, it is logical to allow PARCS/PATHS to read the output from each of the individual

FRAPCON runs, process them, and decide whether convergence is sufficient to move on to the next depletion step. This flow is illustrated in Figure 3.4.2.

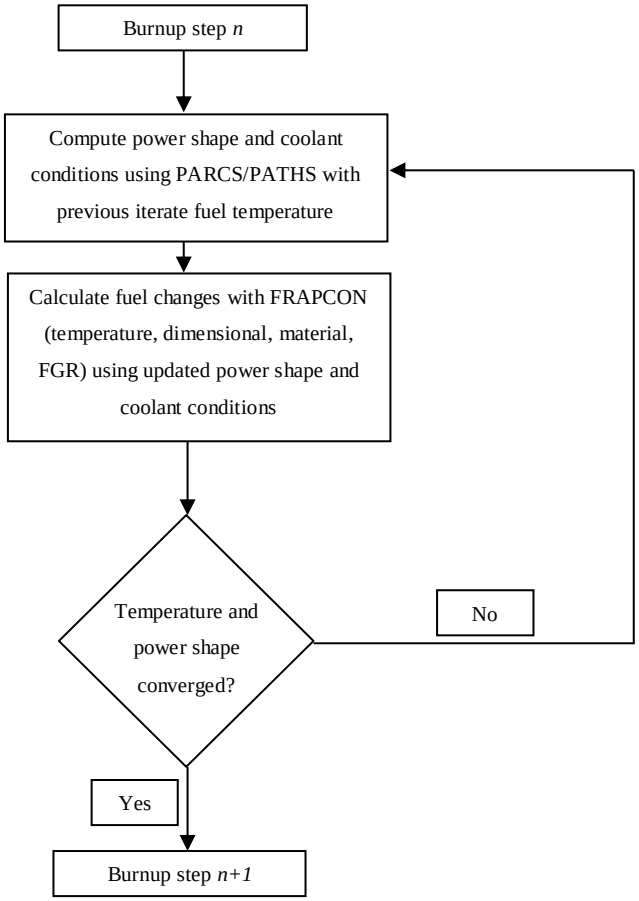


Figure 3.4.2: Computational flow of coupled FRAPCON/PARCS/PATHS

The overall coupling strategy is summarized in the following:

1. First, for a given problem there will be only one PARCS calculation, but potentially several FRAPCON calculations. Thus, we define PARCS to be the "master" code in the coupling strategy (i.e., it will be the code driving the progression through time), and FRAPCON to be the "slave." In order to control execution and coordinate data transfer, a Python script called FRAPARCS is implemented.
2. Within each timestep, PARCS will accept the Doppler temperature for each node computed by FRAPCON by a weighted average of the pellet centerline and surface temperature. Using these temperatures, it will execute in concert with PATHS, and generate a file containing the axial power distribution and the fuel cladding surface temperature for each fuel rod modeled.

3. Each FRAPCON run will be assigned a unique file identifier (an 8 digit number), which will be associated with each of the files to be transferred back between PARCS and FRAPCON. The format of this identifier comes from the following:
    - The first two digits refer to cycle number (of use in multicycle depletions)
    - The second three digits refer to an assembly (in this context, PATHS channel) identifier
    - The final three digits refer to the position of the fuel rod within the assembly. The rod that provides the nodal Doppler temperature used in the cross section evaluation is provided from PARCS the assembly-averaged linear heat generation rate and axial power distribution and is denoted with the identifier '999.'
4. Within a timestep, PARCS and each FRAPCON run will iterate until convergence between successive computations of power distribution and temperatures is achieved. The testing of this convergence will be carried out within PARCS; when it occurs, a flag is generated for the script driving the calculation to advance FRAPCON to the next timestep.

It should be noted at this point that there is a difference between the spatial levels at which FRAPCON and PARCS perform calculations. As stated above, FRAPCON models the evolution under given power and thermal-hydraulic conditions of a single fuel rod. However, PARCS/PATHS calculates the neutron fluxes at the nodal level, which means that the individual fuel rods have been homogenized into blocks which form the mesh for solution of the nodal diffusion equations in PARCS.. In order to provide the necessary single Doppler temperature for each node, each assembly has an associated FRAPCON calculation representing an average rod (i.e., radially averaged). If detailed individual pin histories are required, then pin power reconstruction methods are available in PARCS/PATHS which could allow individual fuel rod powers to be computed and sent to FRAPCON. These individual pin histories do not influence the neutronics calculation. Further details regarding how this process works are described in subsequent sections.

Within the PARCS/PATHS calculation, each assembly (and the fuel rods modeled within the assembly) is assigned an eight-digit integer identifier. All the data that is passed between the codes for that specific fuel rod is contained within data files that have that identifier. This identifier is also used as a new input to the FRAPCON calculation for that specific fuel rod. This strategy ensures that each individual FRAPCON simulation is provided with the power shapes and cladding surface temperatures that characterize the region in the PARCS/PATHS core model

that they represent. The following sections describe the modifications to the individual codes that were required to implement the coupling mechanism described in this section.

3.4.2 FRAPCON Modifications

In order to achieve coupling, modifications were necessary to the FRAPCON source code. Before discussing the specific changes that were implemented, it should be noted that while the current FRAPCON release is version 3.4, the version used as the base platform for this work is an advance version of FRAPCON-3.5 which was obtained from the NRC FRAPCON code manager [145]. FRAPCON-3.5 is structurally and algorithmically equivalent to FRAPCON-3.4; the primary differences are:

- Fixed sized arrays were increased in size to accommodate larger problems. For example, the maximum number of time steps was increased from 400 to 1000, and the maximum number of axial nodes available was increased from 18 to 150.
- Variably-sized axial nodalization was included.
- Axially zoned fuel enrichment capability was included to model fuel rods with blanket regions on the upper and lower ends.
- The cladding creep model was updated to better reflect current publically-available data.

As stated above, in order to preserve the integrity of the methods in FRAPCON, the modifications made in order to enable coupling with PARCS were designed to be as minimally invasive as possible. Therefore, the modifications were in the processing of input and output data, rather than the methods used to obtain the solution itself. As stated above, FRAPCON requires axial power distribution and cladding surface temperature from PARCS/PATHS, and returns the nodal Doppler temperature to PATHS/PARCS. The modifications to the algorithm enable this data transfer.

The input modifications were as follows:

- Added a flag to FRAPCON indicating that a coupled calculation was taking place
- With this flag on, add option to allow FRAPCON to directly read nodal relative powers. In the base FRAPCON, the axial power distribution at each time step is read as a table requiring values specified at the rod bottom (z = 0) and rod top (z = L) with an arbitrary number of points in between; the code then performs internal manipulations to convert the input shape into a histogram. This modification allows for FRAPCON to directly read the histogram shape calculated by PARCS.

The output modification was simply to create a file at the end of the FRAPCON calculation which contains nodal Doppler temperatures to be read by PARCS and used in the cross section evaluation subroutine. Each PATHS channel is modeled by a separate FRAPCON calculation, so within a depletion step, if there are, for example, 25 PATHS channels, then there will be 25 FRAPCON outputs containing the axial Doppler temperature distribution that must be read and processed by PARCS/PATHS. Further details on the functionality of this approach will be provided in a subsequent section.

3.4.3 PARCS/PATHS Modifications

PARCS/PATHS also required modification to enable coupling. As stated in a previous section, the PARCS physics subroutines were unaffected by the coupling; most of the coupling logic is contained within the PATHS subroutine. The only modification to PARCS was the modification of the input reading routines that turned the fuel thermo-mechanical flag on and specified a time limit for each FRAPCON/PARCS/PATHS iteration. The method in which the PATHS solution is performed within a depletion step was presented in Figure 3.3.3; in summary, the steps are as follows:

1. Axial power distributions for all PATHS channels updated from PARCS
2. PATHS computes fluids solution to obtain temperature and density distribution within the channels
3. PATHS uses the fluids solution to compute heat transfer coefficients in each node to obtain nodal fuel clad surface temperatures
4. PATHS uses the surface temperatures in a conduction solve to compute nodal Doppler temperatures
5. Doppler temperatures and coolant temperature and density are passed back to PARCS for cross section evaluation

In order to accommodate the coupling strategy, the control logic was modified as follows:

1. Axial power distributions for all PATHS channels updated from PARCS
2. PATHS computes fluids solution to obtain temperature and density distributions within the channels.
3. PATHS uses the fluids solution to compute heat transfer coefficients and clad outer surface temperatures

4. PATHS writes a file for each PATHS channel containing the channel axially-averaged linear heat generation rate, normalized nodal power density, and rod surface temperature distribution. Once all of these files have been printed, PATHS writes a flag to the working directory which tells the FRAPARCS script that all PARCS-to-FRAPCON data files have been written.
5. PATHS waits for a flag from the script informing it that FRAPCON has evaluated current iterate Doppler temperatures
6. Once a flag is found, loop through all PATHS channels reading in nodal Doppler distributions written by FRAPCON
7. Doppler temperatures and coolant temperature and density are passed back to PARCS for cross section evaluation.

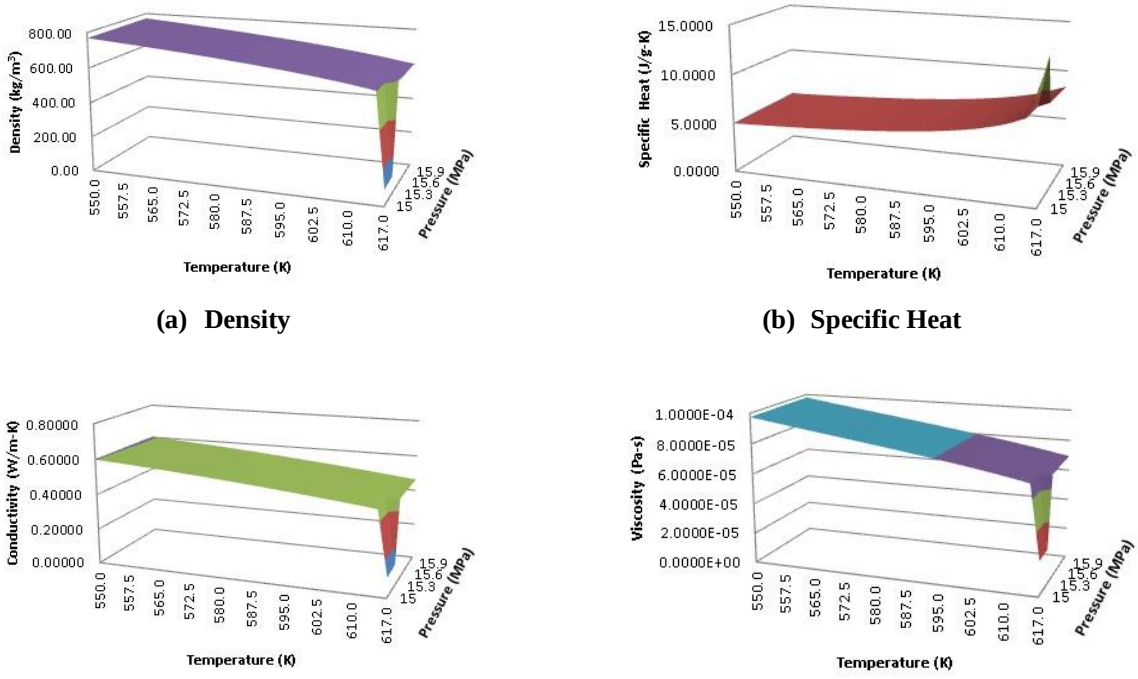
From the PARCS/PATHS point of view, the FRAPCON calculation essentially replaces the conduction solve which provides the Doppler temperature distribution. This allows the coupled calculation to take into account the changing state of the fuel and its effect on the neutronic performance of the core. Within a depletion step, PARCS remains the decision-maker regarding neutronic and thermal-hydraulic (including Doppler temperature) convergence. The convergence criteria that PARCS uses are fractional changes from the previous iteration in k_{eff}, local fission source, global fission source, Doppler temperature, and PATHS fluids solution residual. Once all these parameters are within the tolerance specified by the user, the depletion step is considered converged and the calculation moves to the next step. If one or more of the parameters are not within user-specified tolerance, then PARCS/PATHS writes a flag file to the script indicating that the depletion step will be repeated with updated power shape and coolant temperature distributions for FRAPCON.

In addition to modifications required to enable coupling, subcooled water properties were added to PATHS. As stated above, PATHS was originally developed to provide coolant conditions for BWR depletion calculations and therefore some modifications were made to PARCS v0.42 specifically for the PWR depletion analysis performed as part of this work. BWRs primarily operate as boiling systems, and therefore the fluid properties included in the base version are at saturation. In order to model PWR conditions, subcooled water properties at a range of applicable temperatures and pressures were included as functions in PATHS. The properties required were:

- Density (kg/m^3)
- Viscosity (Pa-s)

- Specific heat at constant pressure (J/g-K)
- Thermal conductivity (W/m-K)

In order to implement these properties, data was obtained from the National Institute of Standards and Technology's (NIST's) chemistry webbook [146] in the form of tables at pressures from 15.0 MPa to 16.2 MPa at 0.2 MPa increments, and temperatures from 550 K to 617 K at 2.5 K increments. A graphical presentation of the included properties is displayed in Figure 3.4.3.



(a) Density **(b) Specific Heat**

(c) Thermal Conductivity **(d) Viscosity**

Figure 3.4.3: Subcooled properties included in PATHS

Functions for each of the properties were constructed and embedded in PATHS, along with a bilinear interpolation function. The calling routine requests the value of the property at a given temperature and pressure, which is returned by the property function. As is clear from these figures, there are some pressure/temperature combinations in the range chosen which correspond to superheated vapor; if such conditions are calculated in PATHS, the program will stop execution. This is to ensure that only subcooled flow conditions are calculated. Within PATHS, where appropriate, the sections of source code that called for the original, saturated

properties were supplemented with an "IF" statement which calls the subcooled properties instead if the local pressure is indicative of PWR-like conditions (i.e., greater than 15 MPa). This included:

- Density update after the momentum/continuity solve and before the energy solve
- Reynolds number calculation for friction coefficient
- Reynolds and Prandtl number calculation for heat transfer coefficient

The calculation of void fraction was unaffected because, for the test problems considered, the power/core inlet temperature combination was selected such that boiling would not happen in the core, and therefore the two-phase models would not be activated. Likewise, the user option to set the two-phase friction multiplier to unity was implemented so that non-physical pressure drop was not calculated.

Finally, in order to support investigation into the potential impact of single-phase heat transfer coefficient uncertainty on the depletion calculation, an input to provide a multiplier on heat transfer coefficient was added to PARCS/PATHS. Additional detail on the basis for this choice will be described in Chapter 4.

With that, the description of the algorithm modifications required to couple PARCS/PATHS and FRAPCON is complete. In the next section, the driver script FRAPARCS which controls the computational sequence is described.

3.4.4 FRAPARCS Driver Script

Each depletion calculation requires coordination between PARCS/PATHS and potentially very many FRAPCON calculations depending on the size of the core. To ensure that the data is transferred in an effective manner and that each process progresses through the calculation in synch, the Python driver script FRAPARCS was developed. FRAPARCS implements the basic coupling algorithm described in Section 3.4.1. A flowchart detailing the algorithm and examples of the FRAPARCS script for a pin cell depletion problem and a multi-assembly multi-cycle model are included in Appendix B.

A coupled calculation begins with FRAPARCS first interpreting the input data. This includes the FRAPCON case identifiers (i.e., the 8-digit numbers described in section 3.4.1), axial nodalization of the FRAPCON cases, the name of the FRAPCON input template file which is used to spawn the cases which are actually executed in the calculation, and the enrichments

used. FRAPARCS is built assuming a user-defined number of depletion cycles, and will shuffle the fuel between each cycle in accordance with a shuffling map provided in the input. Once the initial FRAPCON cases have been spawned, FRAPARCS begins the depletion sequence.

- PARCS/PATHS begins execution for the current cycle as defined in a PARCS input deck
- The script begins the within-step iteration sequence:
    - If PARCS/PATHS has determined that the step is converged, it writes a flag file to the directory telling the script to advance to the next depletion step. The FRAPCON inputs are updated by saving the current step's power level, shape and cladding outer temperature distribution.
    - If PARCS/PATHS has determined that an additional iteration step is required, it writes a flag file telling the script to repeat the current time step. The data from PARCS/PATHS required by FRAPCON has been written to data files containing each PATHS channel's LHGR, normalized nodal power shape, and outer cladding temperature distribution. Using Python's text manipulation capabilities, for each PARCS channel the corresponding FRAPCON input deck has this information inserted into it, and the case is run. For each channel, FRAPCON writes a file containing nodal Doppler temperatures which are read and used by PARCS/PATHS for cross section evaluation
    - If neither flag is present, PARCS/PATHS is still undergoing its neutronic/thermal-hydraulic solution process, and the script will wait before checking for a flag file.
- Once all the depletion steps of the current cycle have been calculated, the script determines if another cycle is requested.
    - If no further cycle is required, the calculation is terminated
    - If another cycle is required, then the input files for fuel to be discharged are moved into a discharge directory. The burned fuel which is remaining in the core is shuffled via file name change to its new location based on a shuffling map provided in the script input section. Then, new FRAPCON inputs for the fresh fuel locations are spawned. Finally, a shuffler program written in FORTRAN is used to generate a PARCS/PATHS restart file based on the previous cycle's final core state. The fuel assemblies' end of cycle burnup, fuel temperature histories, and either coolant temperature or void histories are shuffled to their new location at the beginning of the next cycle, with the fresh assembles represented by the default values for these data. Then, the next cycle's calculation commences.

As stated above, the FRAPCON side of the coupling process is input-driven; this means that, rather than exchanging information with PARCS/PATHS while FRAPCON is executing concurrently, information from PARCS/PATHS is inserted into FRAPCON input files during the

depletion calculation, and the calculations are re-run from the beginning up to the current time step. FRAPCON input files used by FRAPARCS have placeholders for each time step and each piece of required information preceded by an exclamation mark (which is interpreted as a comment by FRAPCON and ignored). For example, the LHGR for step 6 in the FRAPCON file would be represented as `!pw006`; when the depletion cycle reaches step 6, the script reads the LHGR computed by PARCS/PATHS from the PARCS-to-FRAPCON transfer file and inserts it into the FRAPCON input deck in the place of the placeholder. The FRAPCON calculation for that channel is run, and a FRAPCON-to-PARCS transfer file containing the Doppler temperatures for that channel is written. As the calculation progresses in time, the converged LHGRs, power shapes and cladding outer temperature distributions are accumulated in each FRAPCON input file.

Clearly this approach is not optimized; substantial computational drag is introduced by requiring each FRAPCON calculation to restart from the beginning during every iteration. There are several reasons why this method was implemented:

1. The architecture and programming of FRAPCON are not conducive to modification which would allow FRAPCON to run strictly concurrently with PARCS/PATHS. This approach was investigated, but due to difficulties in comprehensively resetting all internal variables back to the previous time step values when a time step was repeated, a consistently repeatable calculation could not be achieved.
2. As stated above, there are individual FRAPCON input decks for each PARCS/PATHS assembly/channel begin modeled. Depending on the problem size, this could create an issue for the computer upon which the calculation was run; for example, a 4-loop PWR contains 193 fuel assemblies. If FRAPCON and PARCS/PATHS were to run as simultaneous processes for the duration of the calculation, this would create 194 processes that would need to be managed by the operating system. While certainly not impossible on large-scale computing frameworks, it would be unwieldy in a desktop environment.
3. The chosen method, while not trivial in its logic or construction, was relatively straightforward to implement within the timeframe necessary for this dissertation. A more efficient implementation may be to embed FRAPCON as a library within PARCS/PATHS, but attempting this carries the same issues as described in (1), in addition to the overhead required to develop an internal PARCS/PATHS-to-FRAPCON interface.
4. Fundamentally, the reason the implemented method is feasible is because, relatively speaking, PARCS/PATHS and FRAPCON calculations are inexpensive. Because of this, even an inefficient coupling strategy can be effectively applied to problems of interest.

In this section, the modifications to PARCS/PATHS and FRAPCON to enable data communication were described, and a comprehensive description of the FRAPARCS algorithm to achieve coupled neutronics/thermal-hydraulics/fuel performance depletion calculations was presented. In the next section, an introduction to the DAKOTA framework for sensitivity and uncertainty analysis, and the methods by which FRAPARCS has been integrated into the DAKOTA environment.

3.5 Sensitivity and Uncertainty Evaluation with DAKOTA

As stated previously, DAKOTA is a framework developed by Sandia National Laboratories for design evaluation, optimization, sensitivity evaluation and uncertainty analysis [7]. The general idea behind DAKOTA is to package commonly used analysis techniques of the preceding types into a convenient repository from which they can be easily implemented by engineers over a broad range of applications. Engineers and software developers can utilize DAKOTA in a variety of ways, up to and including integration of their codes into the tool kit at a source code level; however, for many applications (such as this one), this is not necessary, since the S/UA treatments are contained within program inputs. Therefore, the method of incorporation chosen for this work is the "black box" approach, where the engineering software is treated as a stand-alone model to be called by DAKOTA to generate data for uncertainty analysis. A flowchart detailing the interaction between DAKOTA and engineering software is presented in Figure 3.5.1:

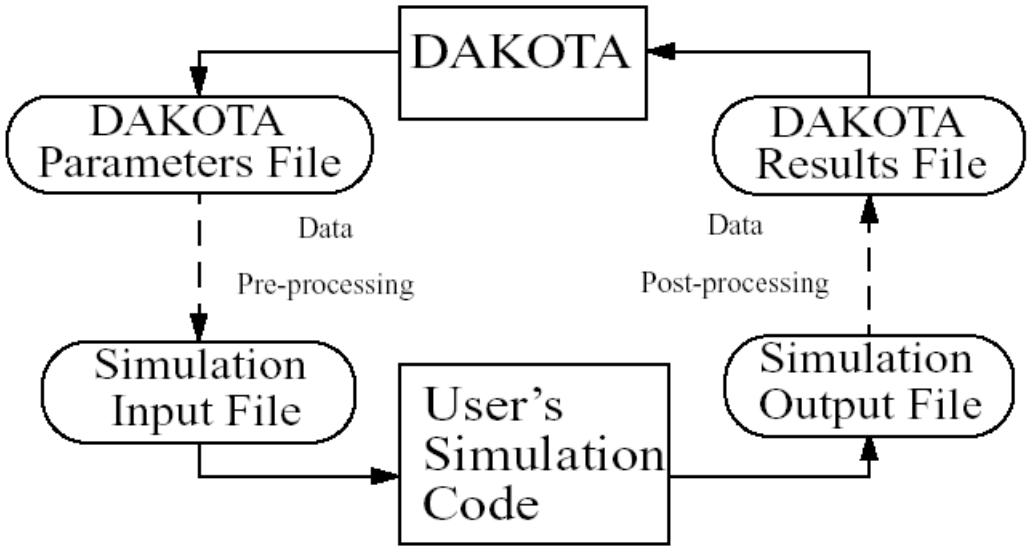


Figure 3.5.1: DAKOTA program flow for "black box" implementation (from [7])

The user specifies in a DAKOTA input file the type of analysis desired, the name and location of the "analysis driver" (usually a script which executes the physics simulator in question) and the parameters of the analysis and how they are to be varied. Within the context of this work, the physics simulator is the FRAPARCS script, and the parameters are the appropriate FRAPCON uncertainty variables. The analysis driver script utilizes the DAKOTA 'deprepro' utility to insert realizations of the uncertainty variables into a FRAPCON template input file based on distributions specified in the DAKOTA input. The simulation is then run, and the resulting responses of interest (for example, criticality or peaking factors) are retrieved from the output and passed back to DAKOTA. Depending on the type of analysis requested, DAKOTA then calculates statistics relating the uncertainties in the input variables to the uncertainties in the responses, and if desired, sensitivity coefficients. Within this work, we employ two varieties of sampling methods (Random Sampling and Latin Hypercube Sampling) and two stochastic expansion methods (Polynomial Chaos Expansion and Stochastic Collocation) to perform uncertainty and sensitivity analysis of a coupled depletion problem. The full description of the S/UA methods which are used in this work will be deferred to Chapter 5.

3.6 Conclusions

In this Chapter, the methodology used in the fuel thermo-mechanical code FRAPCON was presented, as well as the eight uncertainty variables considered important for fuel-related sensitivity and uncertainty analysis. Then, the methods and algorithms for core depletion analysis with PARCS and thermal-hydraulic analysis with PATHS were described. The FRAPARCS coupling methodology was presented, along with the algorithm modifications to each of the individual subcodes necessary to make the coupling of the three codes functional. The basic coupling scheme involves fixed-point iteration within each depletion step to fully converge the fuel temperature distribution and neutronics behavior used by each code prior to moving to the next step. Finally, a brief introduction was provided on the DAKOTA uncertainty package, as well as the implementation of FRAPARCS within the coupled code framework. In the next Chapter, applications of the FRAPARCS script to coupled depletion problems, and comparisons between coupled and uncoupled calculations will be presented.

CHAPTER 4

Application of FRAPARCS to Depletion

4.1 Introduction

The principle purpose of the FRAPARCS code system is to evaluate core criticality, power distribution, temperature distribution, and fuel thermo-mechanical behavior as the reactor undergoes normal operation at power. In order to demonstrate the impact of explicitly modeling the fuel evolution in concert with the neutronics behavior during fuel depletion, two test problems are investigated: a PWR pin cell, representative of the typical geometry modeled by FRAPCON in stand-alone mode, and a 5x5 PWR mini-core, representative of loading patterns commonly used in the industry. We do not include a full-core depletion, as the code coupling is too inefficient to accommodate it in a reasonable time frame. As we will see, however, the two problems we do present capture sufficient physics such that we can draw conclusions about the importance of fuel performance within neutronic depletions for practical cases.

Within the discussion of the pin cell problem, a sensitivity study of the neutronics to the uncertain fuel parameters discussed in Chapter 3, as well as to the heat transfer coefficient computed by PATHS, is presented. These calculations will form the basis for the uncertain variables considered within the uncertainty and sensitivity analysis presented in Chapter 6.

4.2 Pin Cell Model Description

The first test problem analyzed was a pin cell with geometry characteristic of a fuel rod within a 15x15 PWR fuel assembly. This problem was chosen to provide a simple means to evaluate the coupling scheme and ensure that the data transfer between FRAPCON and

PARCS/PATHS was occurring as intended, as well as to provide a quick-running test case that would provide fundamental insights into the physical processes that are being modeled. As will be seen from the comparisons with the mini-core depletion, even the relatively simplistic pin cell depletion contains sufficient physical insights into the importance of fuel performance in depletion to warrant its use within Chapter 6 as the model of choice for uncertainty analysis. In this section, the neutronic model used to generate broad group cross sections with HELIOS is specified, and the FRAPARCS model is described.

4.2.1 Cross Section Generation Model

In order to generate the cross sections for the pin cell, a two-dimensional HELIOS model of a pin cell with reflective boundary conditions was created. The geometry, material data, and power density are based on the Turkey Point Unit 3 three-loop Westinghouse PWR [147]. Figure 4.2.1 illustrates the HELIOS model and Table 4.2.1 presents the design data.

As described in Chapter 2, the traditional process of core evaluation involves generating cross sections at a number of state points, which are then interpolated upon to determine the effective cross section for a specific nodal condition computed by the core simulator. The collection of state points upon which the cross sections are evaluated is referred to as the branch structure. A detailed description of the methods by which PARCS evaluates the cross sections for nodal calculations is presented in the GenPMAX theory manual [148]. In PARCS, distinction is drawn between "history" variables (which characterize the conditions under which depletion takes place) and "branch" variables (which are instantaneous departures from a history state used by PARCS for transient conditions). For example, a given history may specify that the depletion occur at a fuel temperature of 1000 K and a moderator temperature of 570 K for an exposure length of 5000 MWd/MTU with 1000 MWd/MTU depletion steps. The fluxes used to evaluate the nuclide density evolution from step to step use these history temperatures. However, at each of the intermediate points between the beginning and end of the depletion, branch calculations which modify the temperatures and/or densities of each of these regions can be specified which use the nuclide densities resulting from the depletion at the history temperatures, but use cross section data evaluated at the branch temperatures. In this example, at the end of each 1000 MWd/MTU step, there may be branches that change the fuel temperature to 550 K or 590 K, or change the fuel temperature to 800 K or 1200 K. By performing branch calculations, it becomes

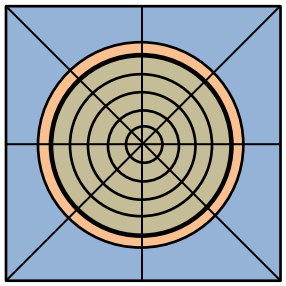


Figure 4.2.1: HELIOS pin cell diagram (blue: coolant, orange: clad, white: gap, grey: fuel)

Table 4.2.1: Pin cell geometry and material data

Cell pitch (cm)	1.43
Clad outer diameter (cm)	1.07
Clad inner diameter (cm)	0.9466
Gap thickness (cm)	0.0095
Fuel outer diameter (cm)	0.9276
Moderator material	Water 11.19 wt/o Hydrogen 88.81 wt/o Oxygen
Cladding material	ZIRLO™ 97.915 wt/o Zirconium 0.005 wt/o Tin 0.11 wt/o Iron 0.9 wt/o Silicon 0.98 wt/o Niobium
Cladding density (g/cc)	6.5
Fuel material	Uranium Oxide (UO_2)
Fuel density (g/cc)	10.4215
Fuel fraction of theoretical density (%)	0.95
Fuel enrichment (wt/o)	4.5
Power density (W/gU)	36.37

possible to model short transients with the core simulator in which the core temperature distributions may change, but the nuclide densities do not appreciably change.

The histories used to generate the pin cell cross sections for the problem here are presented in Table 4.2.2. Since these calculations are performed at PWR conditions, densities are evaluated at the specified temperature at 15.5 MPa pressure. The limits for the moderator temperature ranges were chosen based on the inlet temperature of the core modeled with PARCS/PATHS and the saturation temperature of water at the core outlet pressure. To choose the range of fuel temperatures, a FRAPCON calculation was performed using the rod geometry

and materials with a prototypic cosine power distribution with a rod average LHGR of a typical 15x15 assembly-fueled PWR out to 62 GWd/MTU burnup.

Table 4.2.2: Cross section history structure for pin cell lattice calculations

History ID	Coolant Density (g/cc)	Coolant Temperature (K)	Fuel Temperature (K)
c550f0625	0.769699	550	625
c550f0900			900
c550f1250			1250
c550f1600			1600
c572f0625	0.728872	572	625
c572f0900			900
c572f1250			1250
c572f1600			1600
c594f0625	0.677945	594	625
c594f0900			900
c594f1250			1250
c594f1600			1600
c616f0625	0.603652	616	625
c616f0900			900
c616f1250			1250
c616f1600			1600

Within each history case, there is a branch structure that reflects the other history cases being executed, which is supplemented with a branch at Hot Zero Power conditions should zero power operation need to be modeled with PARCS/PATHS. For example, if the pin cell was being depleted at a fuel temperature of 625K and moderator temperature at 550K, there were 15 branches evaluated at the end of each depletion step which instantaneously changed the temperature conditions to match those of the other cases listed in Table 4.2.2, in addition to a HZP case (fuel temperature is 550K, moderator temperature is 550K), for a total of 16 branches. This was necessary to allow the PARCS/PATHS cross section evaluation methodology to effectively interpolate between history states.

In accordance with Table 4.2.2, there were 16 history cases that were depleted at the power density specified in Table 4.2.1 from 0 GWd/MTU to 65 GWd/MTU. There were a total of 47 depletion steps used, with smaller timesteps used in the early part of the calculation to capture the region in which nuclide densities are changing relatively rapidly (standard practice in lattice calculations, see [149]). The depletion step structure is presented in Table 4.2.3:

Table 4.2.3: Depletion step structure of pin cell model

Burnup range (MWd/MTU)	Depletion Step Size (MWd/MTU)
0 – 0.1	0.1
0.1 – 250	249.9
250 – 500	250
500 – 10,000	500
10,000 – 20,000	1,000
20,000 – 65,000	3,000

In order to generate the cross section data used for the pin cell calculation, the sixteen history cases described above were executed using HELIOS, and the data was post-processed with the Studsvik code ZENITH [150] to provide a format readable by GenPMAX [148], developed at the University of Michigan to provide an interface between lattice codes and PARCS. GenPMAX reads the cross section data provided by the lattice code output, calculates the derivatives which PARCS uses in its cross section evaluation routines, and generates a PMAX file which contains all this information in the format which PARCS can read. Using the computational flow described above, the cross sections for the pin cell case were generated and the PMAX file constructed. In the next section, the FRAPARCS model using these cross sections is described.

4.2.2 FRAPARCS Input Specification

Since FRAPARCS requires the execution of FRAPCON and PARCS/PATHS individually, it is necessary to construct input files for each code. The pin cell models used in this case are representative of a fuel rod from a 15x15 fuel assembly with a power rating consistent with the Turkey Point Unit 3 Nuclear Power Plant [147]. Because a primary impact of improving fuel predictions in core simulation was expected to be observed in the Doppler temperature prediction, a PWR fuel design with slightly thicker rods relative to the more common 17x17 fuel lattice used in industry was used in the pin model to accentuate the effect. Consistency was ensured between the lattice, diffusion, thermal-hydraulic, and fuel thermo-mechanical models under investigation. Input data for the PARCS/PATHS and FRAPARCS models are presented in Table 4.2.4 and Table 4.2.5, respectively. Using this input specification, FRAPARCS was executed to provide comparisons of depletion calculations done with fixed gap conductances similar to the calculations first introduced in Chapter 1. In the next section, these comparisons are presented.

Table 4.2.4: PARCS/PATHS input specifications for pin cell model

Input	Value
Number of assemblies	1
Number of fuel rods/assembly	1
Assembly pitch (cm)	1.43
Active core length	366.0 cm
Axial nodalization	28 total nodes: 2 nodes for bottom reflector (19.05 cm/node) 24 nodes for core region (15.25 cm/node) 2 nodes for top reflector (19.05 cm/node)
Boundary conditions	Reflective (East, West, North, South) Zero incoming current (Top, Bottom)
Depletion Length (d)	1600
Depletion Step Length (d)	10
Core power (kW)	76.21
Inlet coolant mass flow rate (kg/sec)	0.3599
Inlet coolant enthalpy (kJ/kg)	1260.0
Core outlet pressure (MPa)	15.512
Channel area (m^2)	1.143×10^{-5}
Hydraulic diameter (m)	1.357×10^{-2}
Surface roughness (m)	6.800×10^{-5}
Pin pitch (m)	1.43×10^{-2}
Cladding outer radius (m)	5.35×10^{-3}
Fuel pellet radius (m)	4.638×10^{-3}
Cladding thickness (m)	6.170×10^{-4}

Table 4.2.5: FRAPCON input specifications for pin cell model

Input	Value
Cladding outer diameter (m)	1.07×10^{-2}
Cladding thickness (m)	6.17×10^{-4}
Gap thickness (m)	9.5×10^{-5}
Plenum length (m)	0.254
Plenum spring outer diameter (m)	9.1×10^{-3}
Plenum spring wire diameter (m)	1.27×10^{-3}
Number of spring turns	33
Fuel pellet length (m)	1.14×10^{-2}
Fuel pellet dish depth (m)	2.4×10^{-4}
Fuel pellet end - dish shoulder width (m)	1.1×10^{-4}
Fuel stack height (m)	3.66
Number of axial nodes	24
Axial node length (m)	0.1525
Fuel pellet Uranium-235 enrichment (wt/o)	4.5
Fuel fraction of theoretical density	0.95
Cladding material	ZIRLO
Fuel rod fill gas	Helium
Fuel rod fill pressure (MPa)	2.41

4.3 Comparisons with Stand-Alone Depletion

In order to gauge the impact of including fuel performance in the depletion calculation, three PARCS/PATHS standalone models with a fixed gap conductance were used with a representative range of values experienced by fuel during depletion as shown in Table 4.3.1. The solutions with these models were compared with a FRAPARCS model which explicitly models fuel evolution with burnup. As specified in Table 4.2.4, depletion was simulated for 1600 days (roughly corresponding to a rod exposed in a reactor core for three 18-month cycles) at 10 day time steps to simulate the entire range of exposure which might be experience by a fuel rod in a reactor core.

Table 4.3.1: Gap conductances used for stand-alone PARCS/PATHS pin cell model

Case Description	Gap Conductance (W/m^2-K)
Beginning-of-life	7,000
During gap closure	30,000
Closed gap	80,000

Similar to the presentations in Chapter 1, the outputs of interest for this comparative study include the core-averaged Doppler temperature, criticality and axial peaking factor (since this is a single rod problem, radial peaking factors are not present). We also include discussion of the reactivity coefficients at various points in core life to determine if coupling the codes results in significantly different responses to perturbations.

Before delving into the results, we first mention the model that PARCS/PATHS uses for Doppler temperature evaluation. Obviously there is a radial temperature gradient within the fuel as it operates at power; however, as discussed above, core simulators use a representative temperature for each node which is used in cross section evaluation. Therefore, the temperature gradient must be appropriately averaged to produce a single value for use in the core simulator. There are many possible ways to calculate this this value (see, for example, [151]); in PARCS, the following formulation is used:

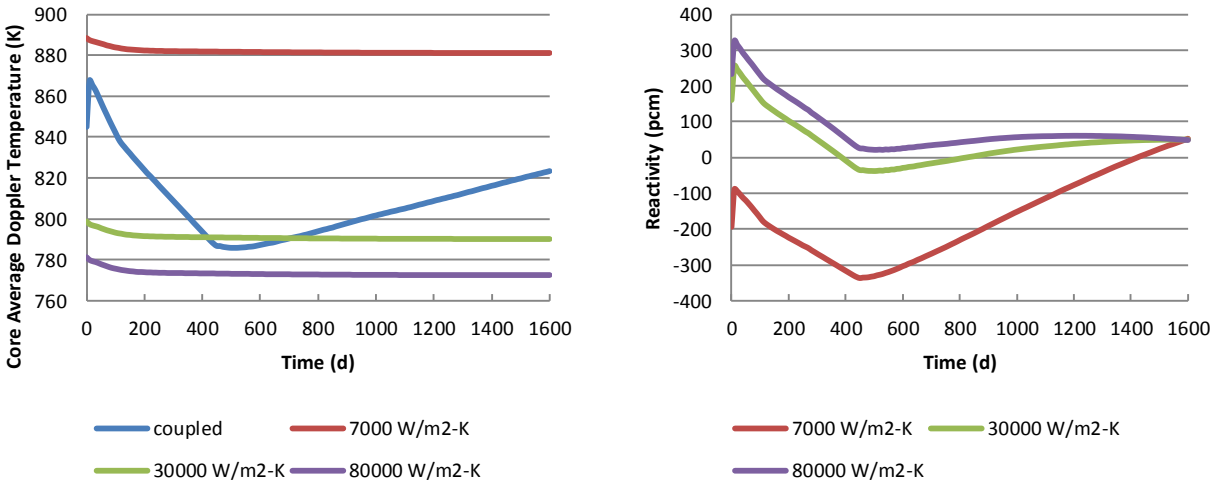
$$T_{dop} = \omega T_{surf} + (1 - \omega) T_{cl} \tag{4.1}$$

where T_{dop} is the Doppler temperature for a given node, T_{surf} is the fuel pellet surface temperature, T_{cl} is the fuel pellet centerline temperature, and ω is a weighting factor, taken as

0.7. In PARCS calculations, the pellet surface temperature is given strong priority over the centerline temperature. We will see the importance of this choice for Doppler temperature in the coupled calculations seen below; it causes a strong interaction between the calculated Doppler temperature and the gap conductance calculated by PARCS.

4.3.1 Doppler Temperature and Core Reactivity

The first results we present are the core-averaged Doppler temperature and the resultant difference in reactivity between the stand-alone and coupled cases (calculated in $10^{-5}\Delta k$ (pcm)). These are presented in Figure 4.4.1:



(a) Doppler Temperature **(b) Reactivity relative to coupled**

Figure 4.3.1: Temperature and reactivity comparisons for standalone vs. coupled case

From Figure 4.3.1(a), it is evident that the core-wide changes in the fuel with depletion cause the Doppler temperature to behave in a way which is not captured by the stand-alone neutronics cases. The initial rise in temperature with the first depletion step is the result of densification, where the fuel pellets become re-sintered and compacted with the rise to operating temperatures. This causes the pellets to slightly separate from the cladding wall, increasing the size of the gap and therefore increasing the temperature drop across it (and by result, the Doppler temperature itself). From this point through the next roughly 500 days of irradiation, the fuel cladding undergoes strain inward as a result of the coolant pressure being greater than the rod internal pressure. At the same time, the fuel is slowly swelling as fission products accumulate within the fuel matrix with irradiation. These two effects slowly close the fuel-clad gap and causes a reduction in Doppler temperature out to about 500 days, when the gap closes. The slow

temperature increase with burnup throughout the remainder of the depletion is due to the degradation in fuel thermal conductivity.

We see from Figure 4.3.1(b) that there can be an approximately 300 pcm reactivity swing between the coupled and uncoupled cases. The reactivity difference basically follows the same trends as the Doppler temperature swing presented in Figure 4.3.1(a). The higher fuel temperatures throughout the cycle computed by the 7000 W/m^2-K case result in a correspondingly low core reactivity relative to the coupled case due to the Doppler effect; the other two standalone cases demonstrate a higher reactivity because their fuel temperatures are lower than the coupled case. As the core depletes and the gap closes, the core average Doppler temperatures of the higher conductance cases approach the temperatures computed by the coupled case, which causes the reactivity difference to be relatively small. As depletion continues and the coupled case's Doppler temperature begins to increase again, there is a corresponding deviation in the system eigenvalue. After gap closure, the reactivity of the 7000 W/m^2-K begins to approach the other cases, despite the higher temperature; this is because the higher-temperature operation results in more resonance absorption. While the instantaneous effect of increased resonance absorption is a negative reactivity effect, over time the creation of plutonium actually causes the reactivity of the system to increase relative to the depletions occurring at cooler temperatures.

In Table 4.3.2, the pre-gap and post-gap closure Doppler temperature swings (i.e., value of the maximum deviation between the coupled and stand-alone cases), and their corresponding reactivity swings, are presented.

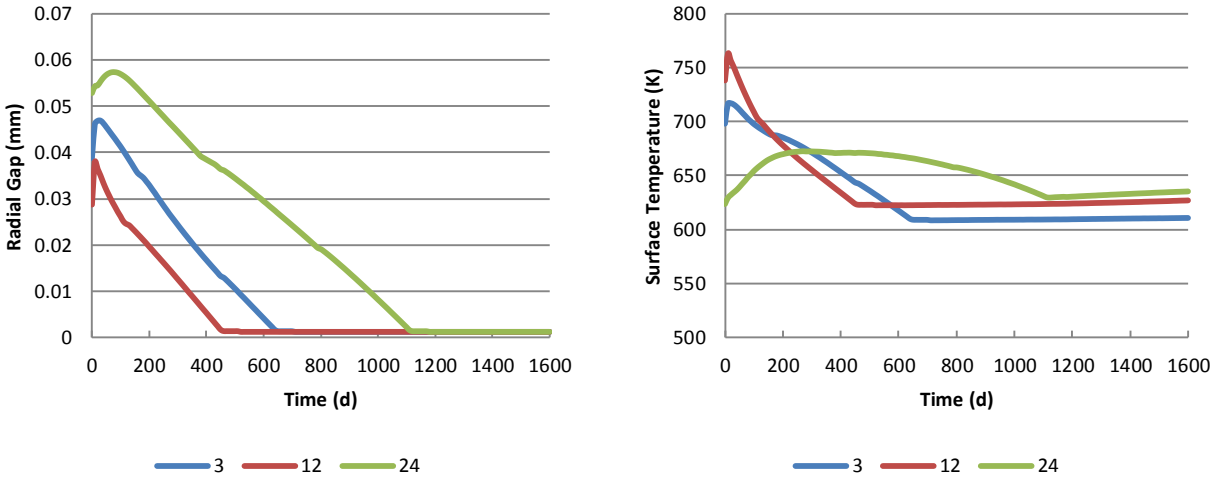
Table 4.3.2: Temperature and reactivity swings for pin cell case

Case		**Doppler temperature swing (K)**	**Reactivity swing (pcm)**
7000 W/m^2-K	Pre-gap closure	74.9	-236.3
	Post-gap closure	-32.9	357.2
30000 W/m^2-K	Pre-gap closure	-74.1	-297.7
	Post-gap closure	33.1	64.9
80000 W/m^2-K	Pre-gap closure	-74.0	-313.0
	Post-gap closure	33.1	21.9

An observation from this data is that there can be a noticeable deviation in system criticality between stand-alone and coupled cases regardless of the fixed gap conductance one choses at the outset of the standalone calculation. This is particularly the case in the range of time wherein the fuel-cladding gap has not yet closed. There is not a simple linear relationship

between the change in core average Doppler temperature and the corresponding change in core reactivity; this is due to history effects (including the buildup of plutonium in resonance absorption) as the fuel depletes.

In order to confirm that it is the closure of the gap and therefore the increased contact between fuel pellet and clad that causes the inflection point in Doppler temperature and reactivity at approximately 500 days, we present in Figure 4.3.2 the structural radial gap and the pellet surface temperature from the FRAPCON calculation for three nodes: one near the bottom (node 3); one at the center (node 12); and one at the top (node 24).



(a) Radial Gap **(b) Pellet Surface Temperature**

Figure 4.3.2: Structural radial gap and pellet surface temperature from FRAPCON calculation

We can see from this figure that once the gap is closed and the fuel and cladding come in thermal contact, the pellet surface temperature stabilizes at essentially a constant value for the remainder of the depletion. For the lower and center nodes, this between 400 and 600 days into the depletion cycle, which corresponds with the inflection points in Doppler temperature and core reactivity evident in Figure 4.3.1. We have thus confirmed that it is gap closure which is the cause of the Doppler and reactivity differences seen early in depletion. Once the gap is closed, the remainder of the differences comes from the thermal conductivity degradation in the fuel, which causes the pellet centerline temperature to increase at constant power. Neither gap closure nor thermal conductivity degradation can be modeled without the use of FRAPCON in the PARCS/PATHS depletion calculation.

4.3.2 Axial Power Distribution

Now that Doppler reactivity has been discussed and found to be an important phenomenon modeled by FRAPARCS, we will present the impact of fuel modeling on other neutronics behavior. Another fundamental output of interest from depletion calculations is the power shape. This is usually characterized with a peaking factor, defined as the ratio of the maximum power in a neutronics node divided by the power of an average node. As we saw in Chapter 1, these factors are computed on a local (nodal), a radial, or an axial basis (denoted F_q, F_{xy}, and F_z, respectively). In this pin cell problem, there is no radial dimension, so $F_{xy} = 1$ and $F_q = F_z$. In Figure 4.3.3, we show the difference in F_z in the standalone cases relative to the coupled case.

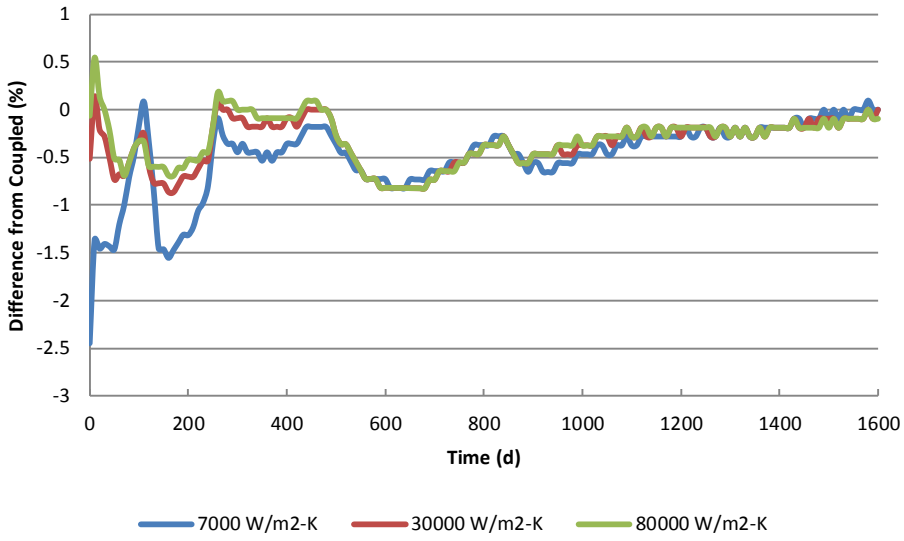


Figure 4.3.3: Axial peaking factor difference relative to coupled case

From this figure, it is evident that the effect that coupling fuel performance with the core depletion on the axial power distribution is relatively limited for the case of a single fuel pin. There are slight differences throughout the depletion which are generally within a few percent. To confirm that the axial power distribution as a whole follows the same trend, the power shapes at Beginning of Life (BOL), Middle of Life (MOL) (i.e., 800 days), and End of Life (EOL), in addition to the time of maximum deviation aside from BOL (in this case, at 160 days) are shown in Figure 4.3.4.

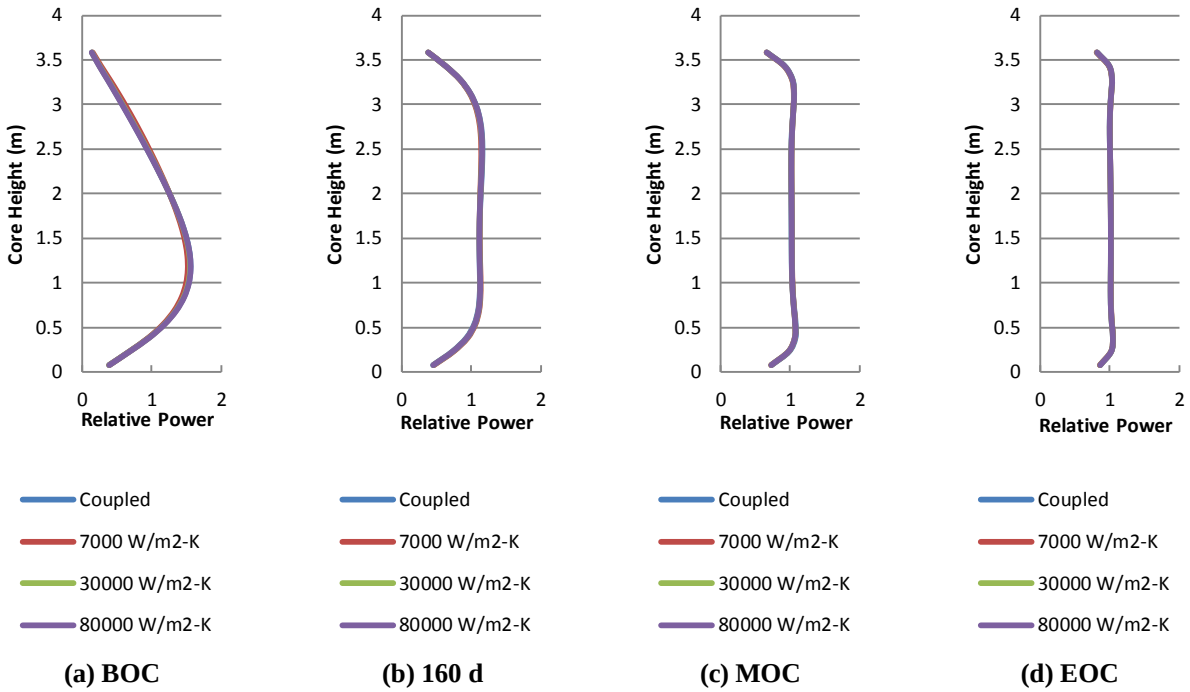


Figure 4.3.4: Power shapes throughout life for pin cell depletion

This figure confirms that, as the pin cell depletes, the power shape between the four cases stays fundamentally the same; at beginning of life, the power shape tends more towards the bottom of the core due to the cooler, denser moderator from the lower plenum. As the fuel in the bottom of the core is depleted, the shape shifts towards the upper portion of the core, and as the fuel in that portion is burned out migrates back down into a slight double-humped shape. These power shapes are characteristic of PWR depletion.

The coupled case shows a very similar power distribution compared to the uncoupled cases primarily because the changes in the fuel temperature resulting from gap closure are reasonably uniform in the channel, even though there are some axial variations in the axial fuel power. In other words, at a given burnup point, while the fuel temperatures within the individual nodes between the different cases might be significantly different, the fuel temperature distributions display fundamentally the same normalized shape, which keeps the resulting axial power distributions very similar to one another. This is confirmed by examining the axial fuel temperature distributions in Figure 4.3.5 which indicate that changing gap conductance tends to shift the temperature distribution left or right as a whole, but not fundamentally change its shape. Therefore, the dominant effect of coupled fuel performance appears to be on criticality, whereas the power distribution is relatively unaffected. Questions

with regard to the radial dimension will be addressed with the mini-core depletions discussed in Section 4.6.

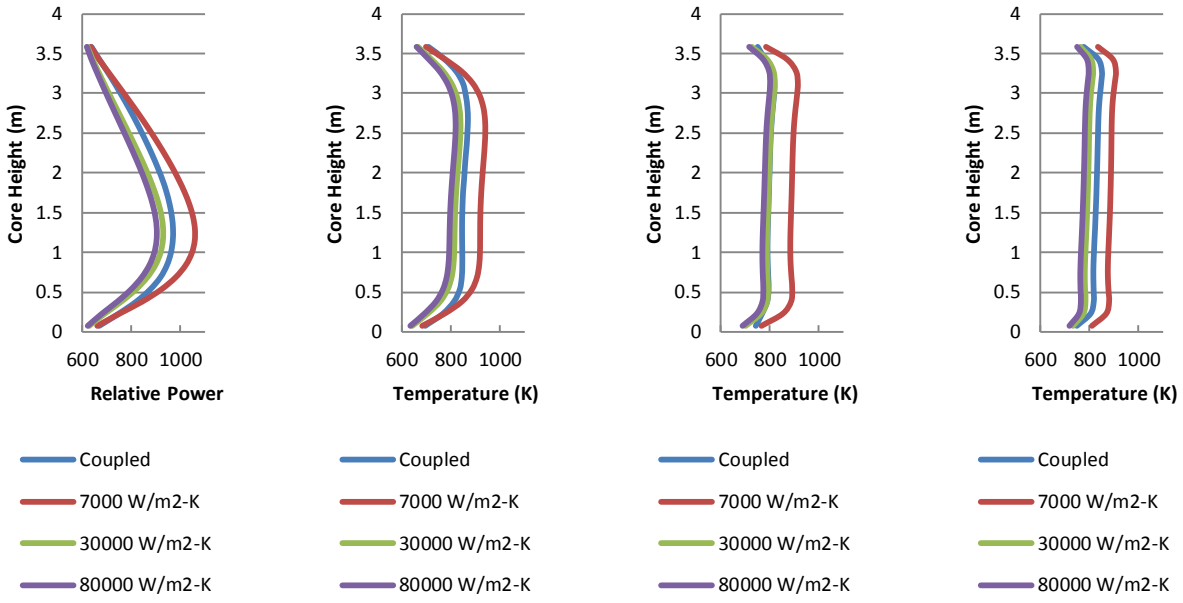


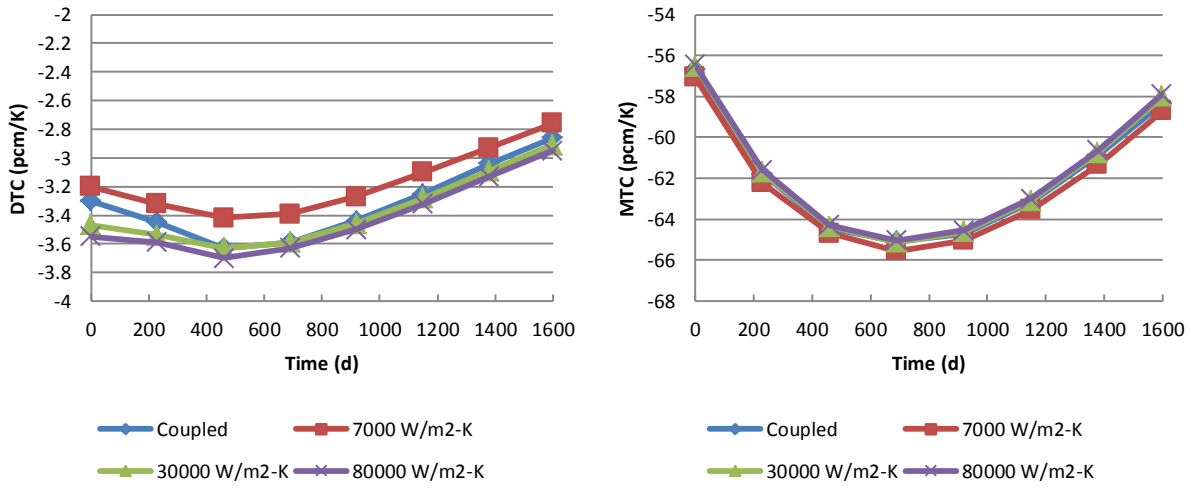
Figure 4.3.5: Doppler temperature distributions throughout life for pin cell depletion

4.3.3 Reactivity Coefficients

The final neutronics quantities of interest to be discussed are the impact of explicit fuel performance on reactivity coefficients. Reactivity coefficients provide a measure for how the criticality of a nuclear system operating at steady-state responds to small perturbations in the system parameters. These coefficients are often used in the design and safety analysis of a reactor core to determine the response of the core to a given perturbation in the operating state. The fuel and moderator temperature coefficients were calculated for the same pin cell problem examined above. In order to compute these coefficients, a system operating point is chosen. Then, using the restart capability of PATHS/PARCS, the temperature distribution at that point is perturbed either up or down, and the system criticality recalculated. The difference in criticality resulting from these up and down perturbations is used to estimate the reactivity coefficient of interest. The following central-differencing formula was used:

$$\alpha_x = \frac{\partial \rho}{\partial x} \approx \frac{k_+ - k_-}{\delta x_+ - \delta x_-} \tag{4.2}$$

where α_x is the reactivity coefficient due to quantity x, x is the quantity to be perturbed (either fuel or moderator temperature), δx is a small perturbation in that quantity (either up (+) or down (-)),and k is the k-eigenvalue computed by PARCS/PATHS. For this valuation, the small perturbation was taken as 5 K up or down applied uniformly to the temperature distributions computed during the depletion. A Python script was written to read the depletion files generated during the depletion calculation, apply the perturbation to the requested variables at specified depletion points, and write restart depletion files for PARCS/PATHS to read to evaluate the resulting system eigenvalue. For the purposes of adequate history coverage, the coefficients were computed at 0, 230, 460, 690, 920, 1150, 1380, and 1600 days. The script read the k-eigenvalues produced by this perturbation and applied the above formula to estimate the resulting reactivity coefficient. The Doppler and Moderator Temperature Coefficients (DTC and MTC, respectively) are presented in Figure 4.3.6:



(a) DTC **(b) MTC**

Figure 4.3.6: Reactivity coefficients throughout depletion for pin cell case

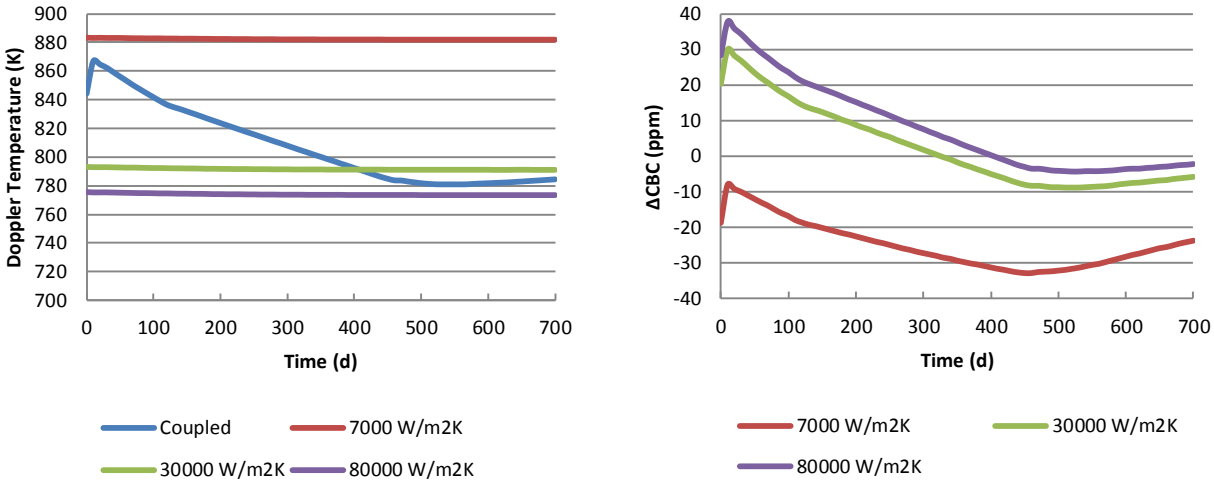
Examination of Figure 4.3.6(a) reveals that, as expected, all DTC are negative because an increase in fuel temperature results in Doppler broadening of the Uranium-238 resonance peaks, which increases parasitic absorption and causes system criticality to decrease. The range of values shown is consistent with the expected values of a PWR (see, for example, [152]). The results do not indicate there is a strong dependence of DTC on gap conductance or fuel modeling; most likely because in the range of Doppler temperatures experienced by the fuel the small changes in fuel temperature do not have a significant impact on the important phenomena

such as resonance absorption. There is a small dependence on gap conductance and therefore fuel temperature; the 7000 W/m^2-K case is operating at a slightly higher temperatures than the other cases, and a small change in temperature about those points does result in systematically lower DTCs. However, the effect is relatively small. The coupled case behaves similar to the 7000 W/m^2-K case early in the depletion, because the fuel temperatures in this region are close, but as the gap closes and the coupled case develops temperatures more akin to the higher-conductance cases the coupled DTC moves towards the uncoupled high-conductance DTCs. With the increase in temperature experienced later in life, the coupled DTC then moves back towards the 7000 W/m^2-K case. In all cases the DTC becomes less negative as the fuel is burned due to the buildup of fissile plutonium isotopes.

From Figure 4.3.6(b), there appears to be an even weaker dependence of MTC on the fuel modeling. This is primarily because, while the fuel temperatures can be significantly different, because this is a steady-state, fixed power problem, the moderator temperatures and axial temperature distributions are almost the same among the different cases run. Therefore, perturbing the moderator temperature has little effect on the physics occurring within the fuel region, where FRAPCON provides its major contribution.

A question could be raised about the reactivity coefficient calculations presented here, as these depletions occurred at zero-boron conditions to allow for examination of fuel mechanical effects out to standard licensing limits. PWRs, however, are operated with soluble boron dissolved in the reactor coolant to maintain the reactor in a critical condition, and it is well-known that the presence of boron has a significant effect on the moderator temperature coefficient. In order to address this, a borated set of pin cell cross sections was developed, with the same geometry and materials as the standard problem described above. The only differences were including a range of boron concentrations from 0 to 3600 ppm (parts per million) in the history and branch structure, and reducing the fuel enrichment from 4.5 wt/o to 2.8 wt/o in order to keep the BOC boron concentration reasonable (i.e., generally representative of real PWR cores). A consequence of choosing a lower enrichment is that the core cannot be depleted for a length of time representative of the full range of burnup experienced by the fuel. The initial enrichment was chosen, however, to allow for a depletion simulation to be carried out through gap closure, which has been established by the initial calculations as the most important part of coupled depletion. This depletion calculation is carried out for 700 days.

In Figure 4.3.7, we present the Doppler temperature and difference in critical boron concentration (CBC) between the FRAPARCS case and the fixed gap conductance cases:



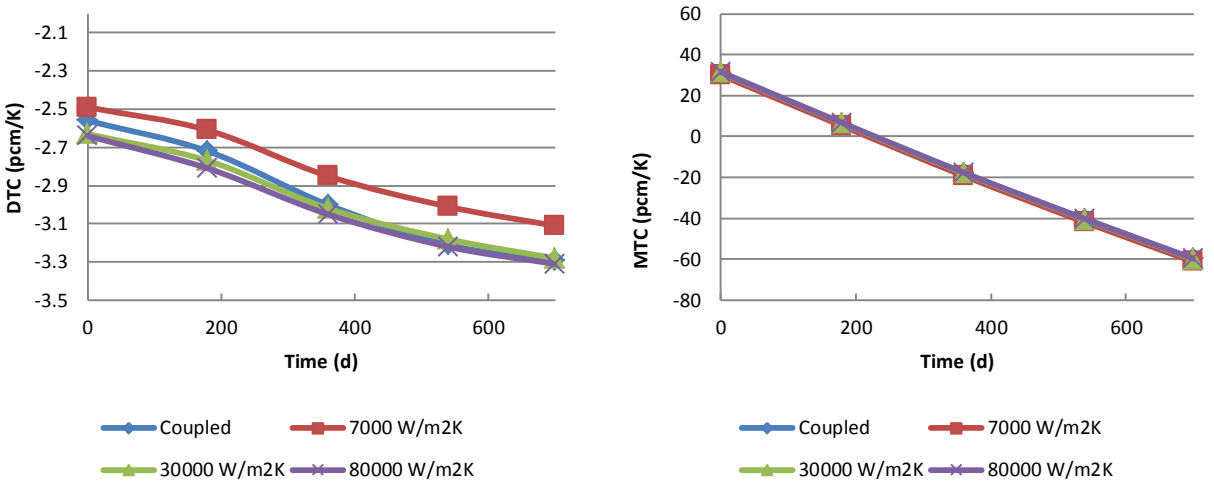
(a) Doppler Temperature **(b) Critical boron concentration relative to coupled**

Figure 4.3.7: Doppler temperature and CBC for borated pin cell depletion

From this figure, we can see very almost identical qualitative behavior between the borated and unborated cases. The Doppler temperature follows the same trend as Figure 4.3.1(a) out to 700 days. As expected, the reactivity curves in Figure 4.3.1(b) follow the CBC curves in Figure 4.3.7(b). The maximum reactivity difference in the unborated cases of approximately 313 pcm translates to a maximum CBC difference of about 37 ppm; this is consistent with typical boron worth.

Using these cases, we can apply the script described above to calculate reactivity coefficients, as presented in Figure 4.3.8. From this figure, we can see that the DTC follows the same trends as the unborated cases; at the beginning of depletion, the coupled DTC is closer to the 7000 $W/m^{2-}K$ case since the Doppler temperature is similar, and as burnup progresses the coupled DTC moves towards the higher conductance cases. For the MTC, we see a quantitative difference in that early in life the MTC is positive; this is because the critical boron concentration is on the order of 3000 ppm. At these boron levels, the effect of lowering the effective density of boron in the core through heating overcomes the negative reactivity effect of reducing the density of the water moderator. However, as burnup occurs and the CBC is reduced, the effect of moderator density decrease becomes dominant. In any case, there is practically no difference in the MTC due to coupling with a fuel performance code; this is again because the important

physics resulting from fuel coupling resides within the fuel region, rather than what is occurring in the moderator.



(a) DTC **(b) MTC**

Figure 4.3.8: Reactivity coefficients throughout depletion for pin cell case

4.3.4 Standalone Comparison Summary

In this section, the results of comparison between stand-alone pin cell problems with different gap conductances and the case where FRAPARCS was used to dynamically model the fuel changes with burnup in the neutronics depletion calculation were analyzed. The results suggest the following:

- There are significant effects on the Doppler temperature, and in a way which fixed gap conductance calculations cannot capture because they lack the appropriate physics.
- The changes in Doppler temperature result in significant time-dependent deviation of system criticality between the fixed gap conductance cases and the coupled case, on the order of approximately 300 pcm. This can be significant from the standpoint of shutdown margin, particularly early in a depletion cycle, where the fuel is changing significantly.
- The impact of coupling fuel physics to the depletion calculation on the power distribution is fairly weak, since the fuel temperature distributions tend to be shifted up or down uniformly as opposed to changing shape significantly.
- The effects on both the Doppler and moderator reactivity coefficients are relatively small, for both borated and unborated cases. Therefore, for follow-up investigations, in order to simulate the entire range of fuel life out to typical licensing limits, we will continue to use the unborated model.

The results of this analysis suggest that dynamically modeling the fuel in a depletion calculation can result in significant changes to some of the outputs of interest relative to stand-alone cases, and therefore further investigation into the specifics of fuel performance is warranted. In the next section, we will examine the impact of the fuel performance uncertainty variables described in the previous chapter, as well as the single-phase heat transfer coefficient, to determine the sensitivity of the neutronics outputs to these parameters.

4.4 Sensitivities to Uncertain Input Variables

Beyond determining the overall effect of coupling fuel performance with neutronic depletion, a further goal of this work is to perform sensitivity and uncertainty analysis to determine which specific fuel models have a significant impact on the calculation. This will assist in broader efforts to determine the uncertainties in nuclear design computations. To facilitate this analysis, we will take advantage of the inherent capabilities of FRAPCON to perform uncertainty evaluations described in Chapter 3. In addition to studying the influence of the fuel parameters, we will also consider the sensitivity of the depletion calculation to the heat transfer coefficient.

FRAPCON allows the user access to eight fuel models which were determined to have a significant impact on fuel performance calculations. As a result of the data evaluations and sensitivity analysis performed in NUREG/CR-7001 [28], FRAPCON's coding was changed to allow the user to bias these models up or down by a fraction of the standard error, calculated by the developers based on model-to-data comparisons. There are two types of biasing available, depending on the specific model in question: absolute biasing, and relative biasing. The method by which biases are applied is described by Equations 4.3 and 4.4, respectively:

$$X = X_{nom} + h\sigma_X \tag{4.3}$$

$$X = \begin{cases} X_{nom}(1 + h\sigma_X) & h > 0 \\ \dfrac{X_{nom}}{(1 - h\sigma_X)} & h \leq 0 \end{cases} \tag{4.4}$$

In these equations, X is the value of the model used in the calculation, X_{nom} is the nominal value returned by the model, σ_X is the standard error computed by the developers and hard-wired into the code, and h is a user-specified bias factor which shifts the model up or down. Usually in

uncertainty calculations performed with FRAPCON, h is drawn from a specified input distribution. In Table 4.4.1, we list the models with available biases, the bias type, and the value of the standard error (from [153]); the fuel models list the values used for uranium dioxide, and the cladding models list the values for the ZIRLO cladding alloy:

Table 4.4.1: Uncertain fuel models and their biases in FRAPCON

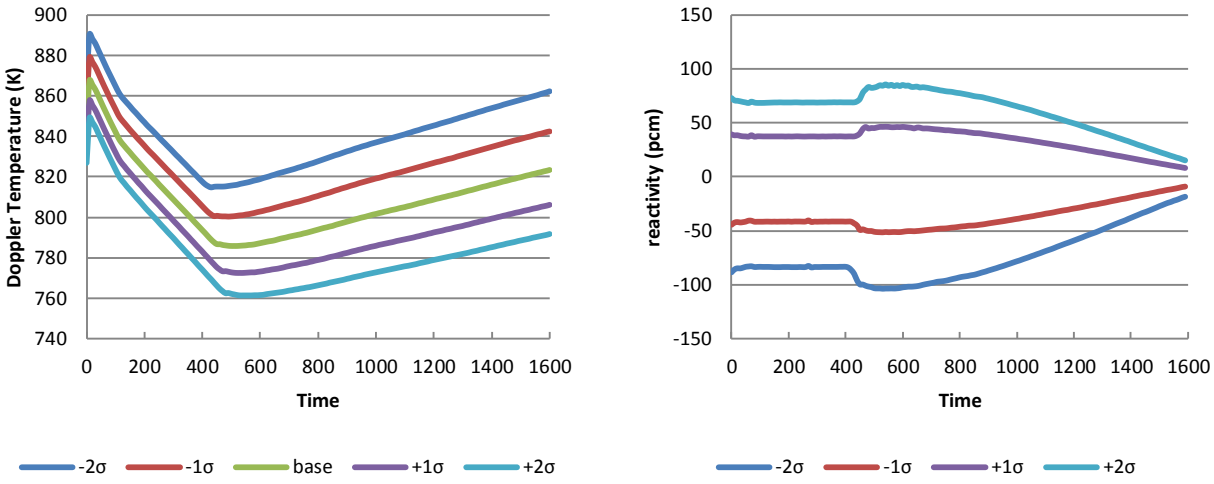
Model	Bias Type	σ_X
Fuel Thermal Conductivity	Relative	8.8%
Fuel Thermal Expansion	Relative	10.3%
Fuel Swelling	Absolute	Below 80 GWd/MTU: 0.08% ΔV/V/10 GWd/MTU Above 80 GWd/MTU: 0.16% ΔV/V/10 GWd/MTU
Fission Gas Release	Relative	100% (applied to gas diffusivity)
Cladding Creep	Relative	14.5 %
Cladding Axial Growth	Relative	22.3%
Cladding Corrosion	Absolute	15 μm
Cladding Hydrogen Pickup	Absolute	45 ppm

In order to assess the influence of each of these parameters on the neutronics calculation, we varied the bias for each input $\pm 2\sigma$ in increments of σ, for a total of four sensitivity calculations per uncertain variable (32 total calculations). Each neutronic output discussed above was then examined to determine the parameters' influences. The reactivity and peaking factors were calculated as the relative difference vs. the unbiased case. This sequence of calculations was performed to form a basis for which inputs proved to be the most relevant for inclusion in further sensitivity and uncertainty analysis.

Of the neutronics outputs, the Doppler temperature and reactivity were found to be the most affected. In most cases, the impact on axial peaking and reactivity coefficients was found to be almost negligible. This can be expected from the standalone analysis presented above; even with rather large changes in fuel temperatures caused by explicit fuel modeling, these three parameters were not strongly affected. Of the eight uncertainty variables, three were found to be particularly important (i.e., causing a reactivity change of > 50 pcm at the $\pm 2\sigma$ level): fuel thermal conductivity, fuel thermal expansion, and cladding creep. Two of the models were found to be moderately important (~20 pcm at $\pm 2\sigma$): fuel swelling and cladding corrosion. Finally, two were found to have minimal influence: fission gas release, cladding axial growth, and hydrogen pickup. The reasons for these will be discussed below.

4.4.1 Strongly Influential Parameters

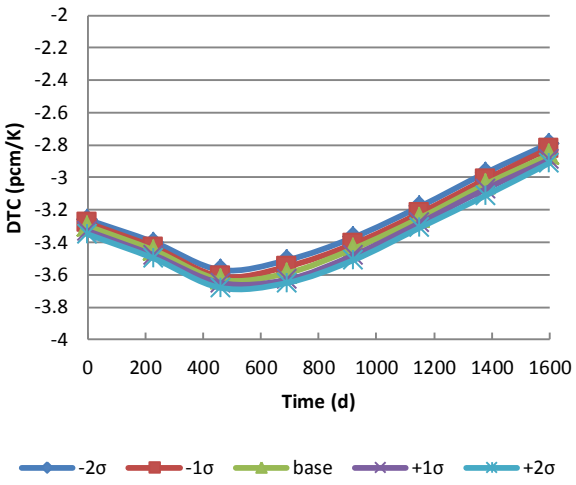
The first parameter we will discuss in the context of coupled depletion is the thermal conductivity of uranium dioxide. As can be expected, this is a very important parameter in thermal analysis, as it is the primary parameter present in the conduction solution within the fuel rod. Biasing this parameter causes a readily-discernible effect on the fuel simulation. In Figure 4.4.1, we show the effect that modifying the thermal conductivity has on Doppler temperature and core reactivity:



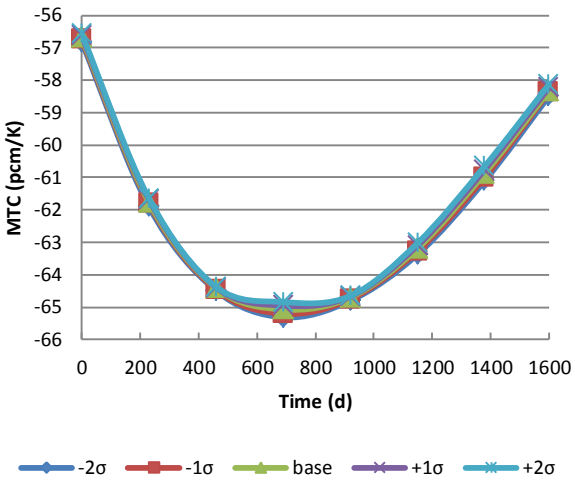
(a) Doppler Temperature **(b) Reactivity Relative to Base**

Figure 4.4.1: Effect of fuel thermal conductivity bias

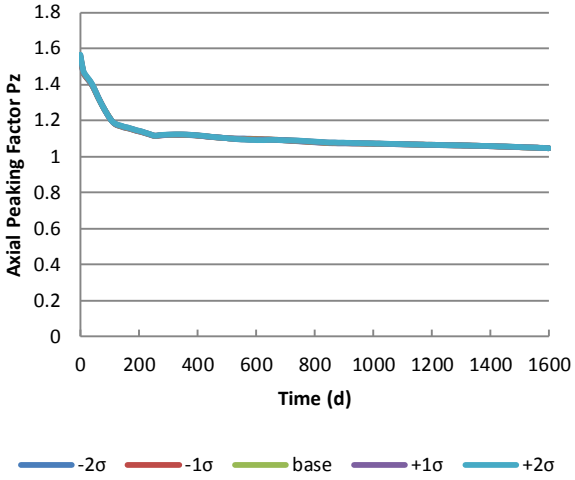
The influence of thermal conductivity on fuel temperatures is obvious from the above figure. By ranging the bias in this parameter from -2σ up to +2σ, the fuel temperature ranges by the end of life by approximately 70 K. As is expected, there is an inverse relationship between the direction of bias and the computed temperatures; biasing the model up causes a lower temperature and vice versa. The reactivity biases in these cases due to the modification of fuel thermal conductivity are very apparent and follow inversely from the direction of the temperature. Prior to gap closure, the reactivity bias is relatively constant. However, once the gap begins to close at around 600 days of depletion there is a slight hump either up or down depending on the direction of bias due to the fuel and cladding coming into contact and a new temperature distribution equilibrating. Then, as the fuel depletes, each of the systems approach the base case; this is due to the buildup of plutonium gradually overcoming the increase in resonance absorption caused by Doppler broadening.



(a) DTC



(b) MTC



(c) Axial Peaking Factor

Figure 4.4.2: Reactivity coefficients and axial peaking factor for thermal conductivity cases

As was the case with the standalone comparisons, the axial peaking factor and reactivity coefficients were minimally sensitive to biases in thermal conductivity. This was also the case with the other variables considered. As shown in Figure 4.4.2(a), DTC is minimally affected; the slight differences that do exist are due to the evaluation of the base Doppler temperature at different points. From Figure 4.4.2(b), it is evident that MTC is almost unaffected, again because perturbing the coolant conditions has very little effect on the fuel. Finally, the differences in axial peaking factor are basically negligible as shown in Figure 4.4.2(c); this is for the same reason the peaking factors were largely unchanged in the standalone comparisons. Although the fuel

temperatures are quite different, the Doppler temperature profile is of the same shape. Therefore, the relative power distribution throughout the rod remains essentially unchanged.

Thermal conductivity is the parameter which has the largest effect on the neutronics evaluations, and its effect is obvious throughout the entire course of depletion. However, there are two other parameters which also have a strong effect on Doppler reactivity in particular because of their influence over the behavior of the fuel-cladding gap. As we have discussed, the gap closes early in life due to a combination of the fuel moving outward due to thermal expansion and swelling, and the cladding moving inward due to creep-down. Biasing the thermal expansion and cladding creep models in particular causes the gap to close at noticeably different rates, resulting in relatively strong deviations in Doppler reactivity. However, as was the case with thermal conductivity, neither of them shows strong influence over peaking factors or reactivity coefficients. The reactivity effects are discussed below.

It is difficult to isolate the contribution of thermal expansion alone from the FRAPCON output. Therefore, we will discuss its contributions in terms of the fuel-clad gap size. To do this, we first present the rod-averaged (i.e., averaged over 24 axial nodes) gap thickness in Figure 4.4.3. Although the averaging process causes us to lose some spatial detail, as we have seen in previous figures the vast majority of the fuel reaches the closed-gap regime within a relatively short span. It is only the low-power regions on the extreme ends of the core that maintain an appreciable gap for most of the depletion.

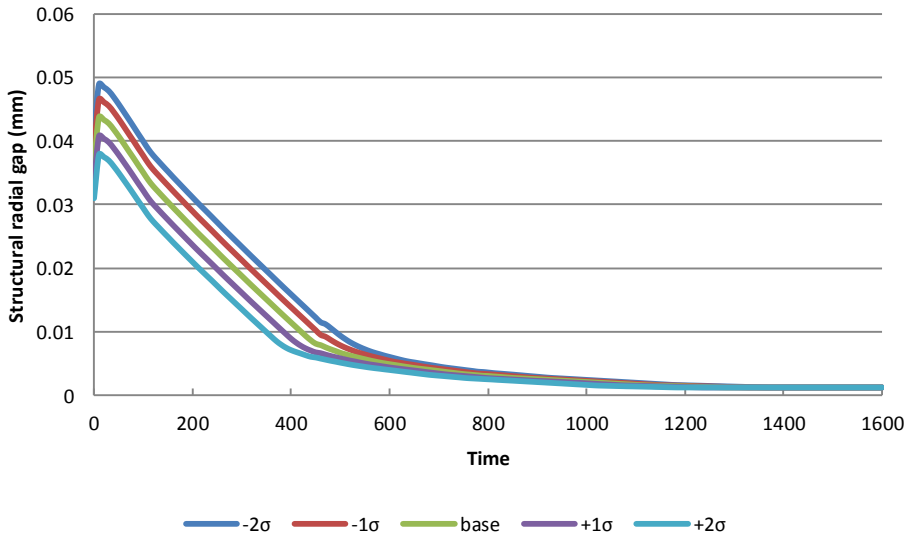
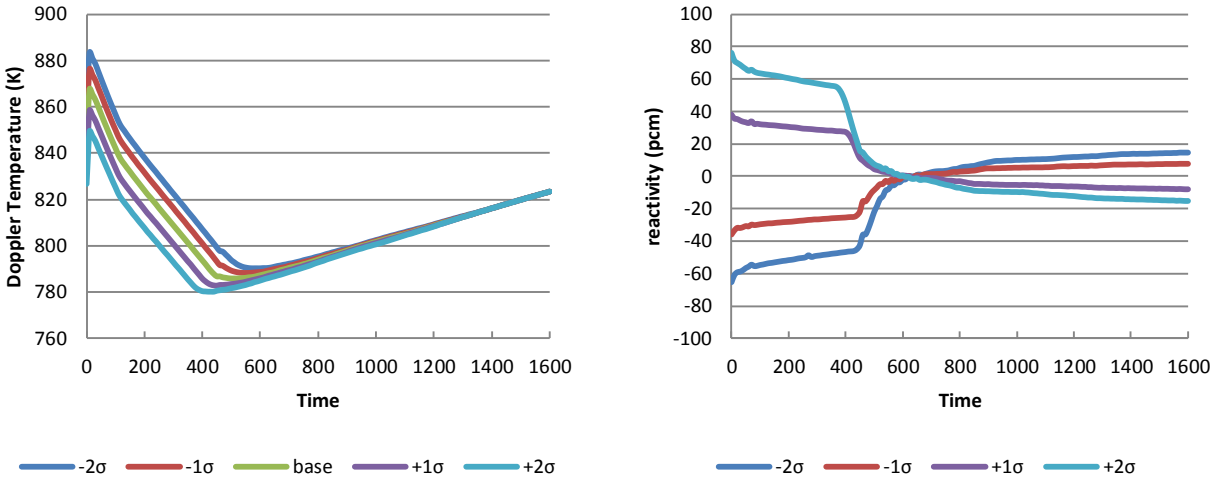


Figure 4.4.3: Rod-averaged gap width for fuel thermal expansion cases

As we can see from this figure, our expectations regarding the size of the fuel-clad gap are met; biasing the model towards higher expansion causes the width to be lower, and biasing the model towards lower expansion causes the gap to persist in time longer. The temperature and reactivity effects of this are shown in Figure 4.4.4.



(a) Doppler Temperature **(b) Reactivity Relative to Base**

Figure 4.4.4: Impact of thermal expansion on Doppler reactivity

The direct influence of the gap size on the computed Doppler temperatures is again apparent in Figure 4.4.4(a); the larger gaps in the downward-biased cases result in higher fuel surface (and thus Doppler) temperatures, whereas the converse is true for the upward-biased models. Once the gap closes, fuel thermal expansion no longer has an influence on the temperature behavior and all models compute approximately the same values. Figure 4.4.4(b) shows some similarities between the fuel thermal conductivity cases prior to gap closure in that the deviation in the keff relative to the unbiased case is approximately constant before the gap begins to close. Once closure is imminent, the differences between the unbiased and biased cases become quite small, until they begin to turn around what appear to be asymptotic values. This behavior is due to the differences in plutonium generated prior to gap closure; for the cases where the fuel is depleted in a hotter environment, the increased resonance absorption causes more plutonium. The effects of this are manifested later in life once the temperature equilibrates across the cases; the cases which had more resonance absorption and initially a lower reactivity now have more fissile material available later in life, which increases reactivity relative to the base case. The opposite is true for the cases depleted in a cooler environment.

We can also investigate the impact of the cladding creep model in a similar way to the fuel thermal conductivity calculations. We again begin the discussion with a presentation of the fuel-clad gap in Figure 4.4.5, because of the unavailability of creep-specific information.

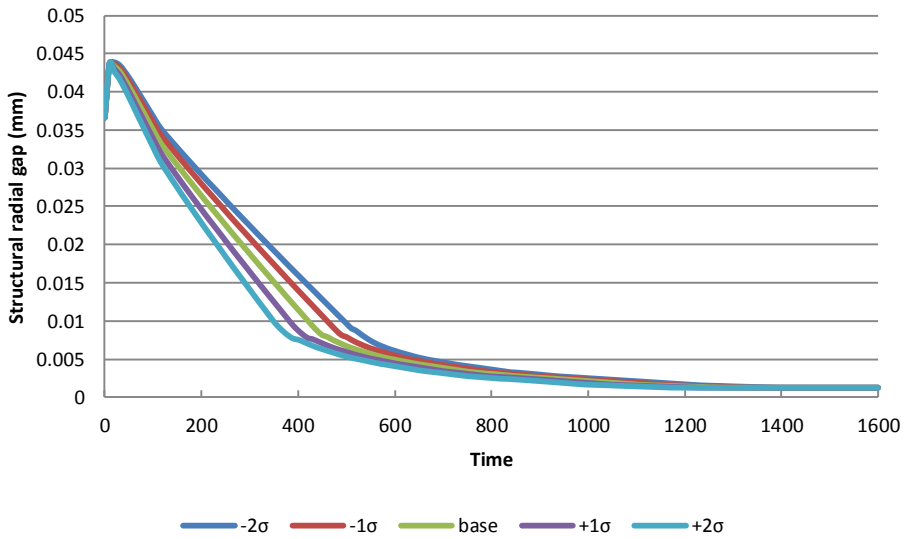
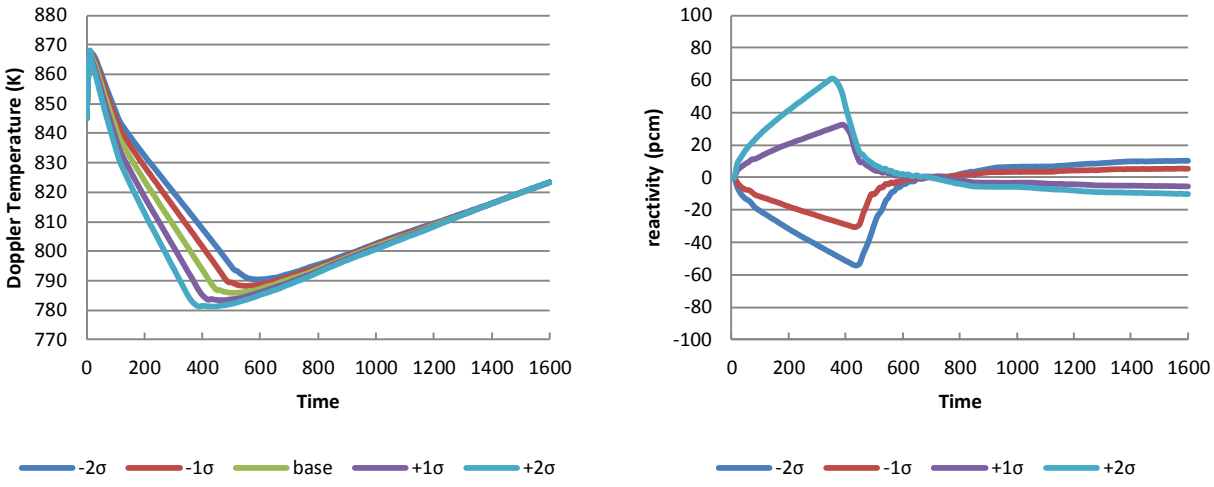


Figure 4.4.5: Gap width in creep sensitivity cases

The influence of the creep bias is very apparent in the above figure. For the models which are biased in the positive direction, the fuel-clad gap is reduced in size at an early point than the base model; in comparison, the models biased in the negative direction have less creep and therefore the gap remains open for longer. However, in all cases the gap is closed in about 500 days. Unlike thermal expansion (which is an instantaneous phenomenon that occurs as the fuel is brought up to temperature), creep is a time-dependent phenomenon which requires some operation to occur before manifesting itself. The influence of creep on Doppler reactivity is presented in Figure 4.4.6:



(a) Doppler Temperature **(b) Reactivity Relative to Base**

Figure 4.4.6: Impact of cladding creep on Doppler reactivity

In Figure 4.4.6(a), we see that the core-averaged Doppler temperature history closely follows the size of the fuel-clad gap displayed in the previous figure. As the gap closes and improved heat transfer out of the fuel through the cladding is achieved, the Doppler temperature hits a local minimum. Afterwards, the gap size does not have a strong influence on the Doppler temperature; rather, it is the degradation of thermal conductivity which causes temperature increase. Figure 4.4.6(b) illustrates the quantitative reactivity effect that occurs because of the differences in gap closure behavior. The cases with positive bias close the gap quicker than the base case, which causes a lower Doppler temperature and thus a positive reactivity effect; in the cases with negative bias, the opposite is true. However, once the gap closes the differences between the biased and unbiased cases become fairly small. Rather than the instantaneous effects we saw in the thermal expansion, the changes in Doppler reactivity are a time-dependent phenomena corresponding with the different creep rates among the cases. As with the expansion cases, once the gap is closed reactivity changes are due to the differences in accumulated plutonium from the exposure conditions early in life.

With this, we conclude our discussion of the parameters considered strongly important on reactivity. The fuel thermal conductivity was found to have the strongest effect (+82 pcm/-101 pcm at $\pm 2\sigma$) and persisted throughout the course of the depletion. On the other hand, the thermal expansion and creep models were lower and primarily important prior to gap closure, but still significant (+76 pcm/-65 pcm and +58 pcm/-51 pcm at $\pm 2\sigma$, respectively). In the next section,

we will discuss some of the models that are of more moderate influence over the depletion calculation.

4.4.2 Moderately Influential Parameters

The two parameters which were found to have less, but still non-negligible influence on the neutronics output, were the fuel swelling and cladding corrosion models. Fuel swelling is phenomenologically similar to cladding creep, in that it is time-dependent. The differences are that there is a threshold value of burnup that must be reached prior to swelling being computed (this is to allow the fission products which cause swelling to populate whatever residual porosity remains in the fuel early in operation, unlike creep, which begins at the outset), and that the swelling causes the gap to decrease from the inside (i.e., the fuel region) rather than from the outside (the cladding). Because of this delayed onset, and because the relative dimensional changes are less than for creep, the reactivity effect is significantly less. Despite this, it is significant enough to include in the more detailed uncertainty analysis presented in Chapter 6.

Cladding corrosion causes extra thermal resistance on the exterior surface of the cladding, which increases Doppler temperature. As we will see, however, the implementation of the bias in the cladding corrosion model in FRAPCON is non-physical. Therefore, because of the relatively small magnitude of neutronics effect and the unphysical way in which it behaves, we will preclude corrosion from the uncertainty calculations later in this thesis.

To begin our detailed discussion, we will draw comparisons between the cladding creep and fuel swelling results by presenting the fuel-clad gap behavior resulting from biasing the swelling model in Figure 4.4.7. From this figure, it is seen that biasing the fuel swelling parameter has a minor effect on the size of the fuel-clad gap. The magnitude of the swelling is purely a burnup-driven effect (in contrast with some of the other fuel physics, which are both temperature- and burnup-dependent). In FRAPCON, there is a burnup threshold of 6000 MWd/MTU prior to which no swelling is predicted to occur. Therefore, the differences in swelling do not begin to become apparent until around 200 days of depletion, when some of the nodes cross the threshold. From that point, there is a slight difference in the size of the gap until the fuel comes in contact with the cladding at about 500 days. From that point, the gap is closed.

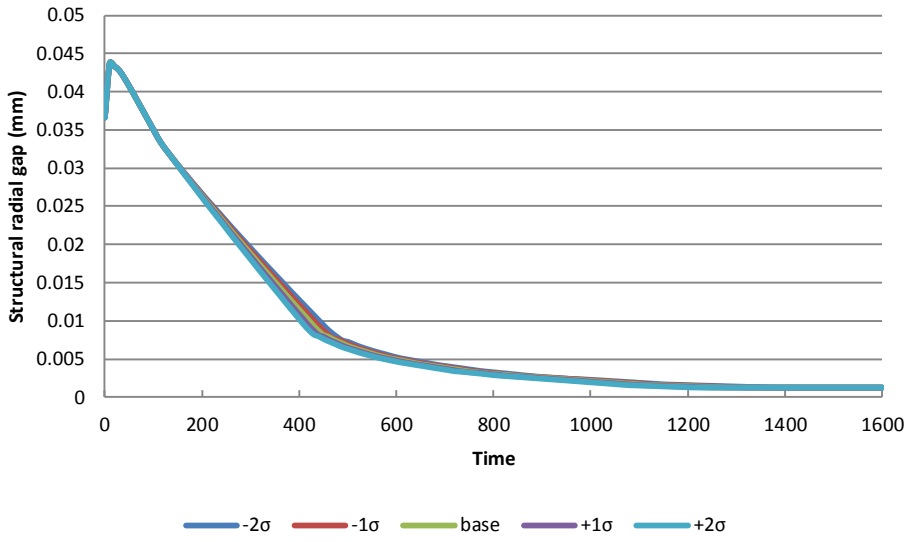
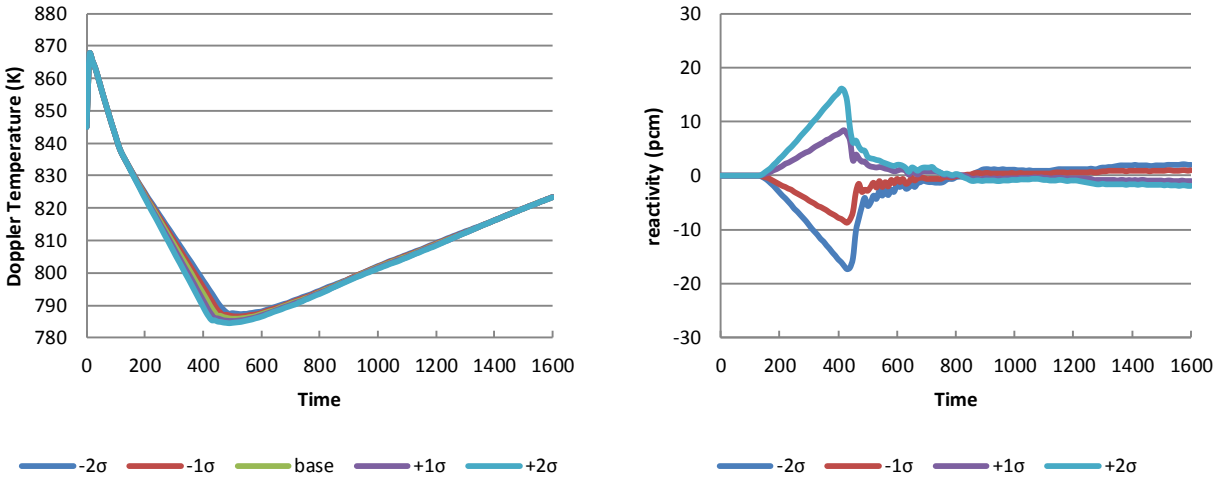


Figure 4.4.7: Gap width in fuel swelling sensitivity cases

Based on the small deviation relative to some of the other sensitivity cases, one can expect a relatively small influence on core Doppler temperature and thus criticality. In Figure 4.4.8, respectively, these impacts are presented.



(a) Doppler Temperature **(b) Reactivity Relative to Base**

Figure 4.4.8: Impact of fuel swelling on Doppler reactivity

As expected, the Doppler and therefore reactivity differences between the biased and unbiased cases are relatively minor. Since all of the nodes are below the 6000 MWd/MTU threshold prior to 140 days of depletion, there is no difference in Doppler temperature or criticality. Once the threshold is reached, the differences in fuel swelling between the different cases begin to manifest themselves in the Doppler evaluation; the models biased up show more

swelling, a smaller gap, lower temperatures and therefore higher reactivity relative to the base case. One the other hand, the converse is true for the models which are biased toward less swelling. Once the gap closes, the fuel approaches the same temperature between all cases, and the residual reactivity effect is due to the history each case experiences early in life; the cases that were more reactive early in life become slightly less reactive as the burnup progress, and vice versa. As expected, the magnitude and shape of the Doppler temperature differences does not significantly influence axial peaking or reactivity coefficients. We see in comparison with the creep results presented in Figure 4.4.6 almost identical qualitative behavior once the burnup threshold is reached.

As we have mentioned previously, the cladding corrosion model does not behave physically. To illustrate this, we present the behavior of the corrosion layer under biased conditions in Figure 4.4.9:

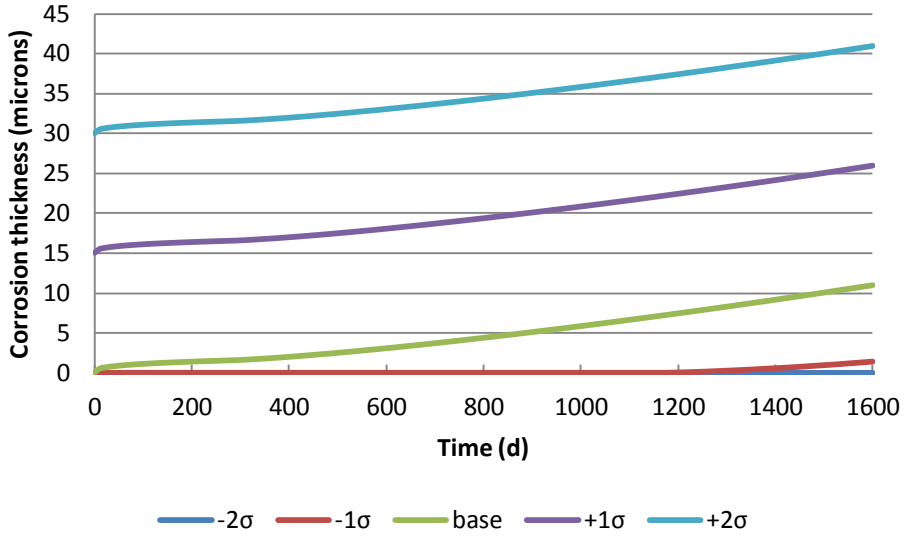


Figure 4.4.9: Gap width in fuel swelling sensitivity cases

In this figure, it is evident that by biasing the corrosion parameter in the negative direction, cladding corrosion layer growth is almost completely suppressed. By the same token, by biasing the model in the positive direction an immediate, thick layer of corrosion is calculated to occur on the cladding surface which then continues to grow as the rod is irradiated. However, this is physically not what we expect by biasing the corrosion model. Effectively what FRAPCON does is add an extra corrosion layer to the initial condition, which then grows at approximately the same rate throughout the depletion. What should happen, on the other hand, is

for the corrosion to start at the same value, and biasing the model should cause the corrosion layer to grow at different rates. Conversations with the FRAPCON developers indicate that they chose this approach based on best fits to higher-burnup fuel data [154]. The data may be matched appropriately, but the way in which the corrosion layer at these burnup levels is reached is not realistic in our opinion. It is outside the scope of this thesis to correct these perceived issues, and since the reactivity effect is on the order of -25 pcm at $+2\sigma$ (i.e., much lower than the "significant" models), we choose to disregard the phenomenon in follow-up uncertainty analysis.

In this section we detailed the two models which were found to have moderate influence on neutronics effects. Like the significant models, no significant differences in peaking factors or reactivity coefficients were found; however, there were influences on the Doppler reactivity. For the fuel swelling model, reactivity differences of +20 pcm/-21 pcm at the $\pm 2\sigma$ level were observed. Cladding corrosion had an influence similar in magnitude, but the implementation of corrosion biasing in FRAPCON was found to be non-physical. Therefore, we will include swelling in the uncertainty analysis presented in Chapter 6, but will not include corrosion studies. In the next section, we will discuss the models which showed no influence over the neutronics calculations, and the reasons why.

4.4.3 Non-Influential Parameters

Three of the parameters were discovered to be practically insignificant with respect to coupled depletion: cladding hydrogen, cladding axial growth, and fission gas release. The following discussion gives the reasons why, for this particular problem.

First, we discuss cladding hydrogen content. This is an output that is computed for informational purposes primarily driven by the need of FRAPCON users to estimate hydrogen concentration in fuel cladding for the purposes of comparisons with Loss of Coolant Accident (LOCA) evaluations. The NRC has recently been engaged in a rulemaking effort to modify the LOCA acceptance criteria found in 10 CFR 50.46(b) to account for the difference in performance of different cladding materials [155]. As discussed in Chapter 3, the presence of hydrogen in cladding alloys has an embrittling effect which may lead to failure under extreme transient conditions. However, as currently implemented the cladding material models in FRAPCON do not account for changes in their properties due to hydrogen. Therefore, while

hydrogen levels are important outputs for strictly fuel performance evaluations, they do not affect the thermo-mechanical calculation, and therefore show no influence on coupled depletion.

At first glance, it appears that cladding axial growth may have some influence over the mechanical calculations. Physically, growth has the effect of increasing the plenum length, which affects the amount of room the free gas within the rod has to expand and contract. This can have the effect of changing the outward pressure on the cladding, therefore changing the mechanical response. The ZIRLO™ material used in these calculations, however, does not exhibit a great deal of growth behavior. Observation of the parameters in the growth model (see Appendix A) show that, for equivalent fluence, the growth of ZIRLO™ is approximately four orders of magnitude less than M5™ or Zircaloy-4. Therefore, the sensitivity cases we executed did not show an appreciable impact.

Finally, we discuss the fission gas release model. As we have noted in Chapter 3, there are two forms of release: athermal release, which occurs no matter the temperature due to fissions that occur near grain boundaries, and thermally driven release, which occurs due to the diffusion of fission gases from within the fuel grains to their boundaries. Similar to the swelling model and burnup, there is a threshold value of temperature which must be reached to activate thermal release. The FGR bias described in Table 4.3.1 is applied to the gas diffusivity which forms a part of the thermal release model. At the conditions experienced within the rod during the pin cell depletion, the temperature does not reach this threshold, and therefore applying the bias does not cause any difference in the coupled calculations considered here. Therefore, we will not consider FGR in subsequent uncertainty calculations in this thesis.

With this, the discussion of the insensitive models is complete. In the next section, we will briefly move on from the fuel performance sensitivities and evaluate the impact of the heat transfer coefficient on PWR depletion.

4.4.4 Heat Transfer Coefficient Sensitivity

Another parameter which has a direct influence over the fuel temperature and therefore Doppler reactivity is the heat transfer coefficient (HTC). This describes the temperature rise from the bulk coolant flowing through the core to the fuel surface, depending on the flow conditions. In Chapter 3 we described the modifications in PATHS to model PWR conditions and to enable the user to manipulate the HTC in a similar manner to how the fuel biases are implemented.

Newton's law of cooling is used to calculate the temperature rise from the bulk coolant to the rod surface. This is given by the following expression:

$$q'' = h(T_w - T_{bulk}) \tag{4.4}$$

In this equation, q'' is the surface heat flux, h is the heat transfer coefficient, T_w is the wall (i.e., rod surface) temperature, and T_{bulk} is the bulk temperature of the coolant flowing up the channel. From the neutronics solution, q'' is known; PATHS solves the coolant mass, momentum, and energy equations to evaluate the profile of T_{bulk} up a channel. Therefore, to calculate the wall temperature, h must be evaluated.

By construction, the PATHS calculation only considers single phase heat transfer; should of the coolant be indicated, the calculation shuts down. To obtain the HTC based on the flow conditions, PATHS makes use of the Dittus-Boelter correlation, widely used to model heat transfer in reactor fuel bundles:

$$\mathrm{Nu} = \mathrm{Re}^{0.23}\mathrm{Pr}^{0.4} = \left(\frac{\rho v L}{\mu}\right)^{0.23}\left(\frac{c_p \mu}{k}\right)^{0.4} \tag{4.5}$$

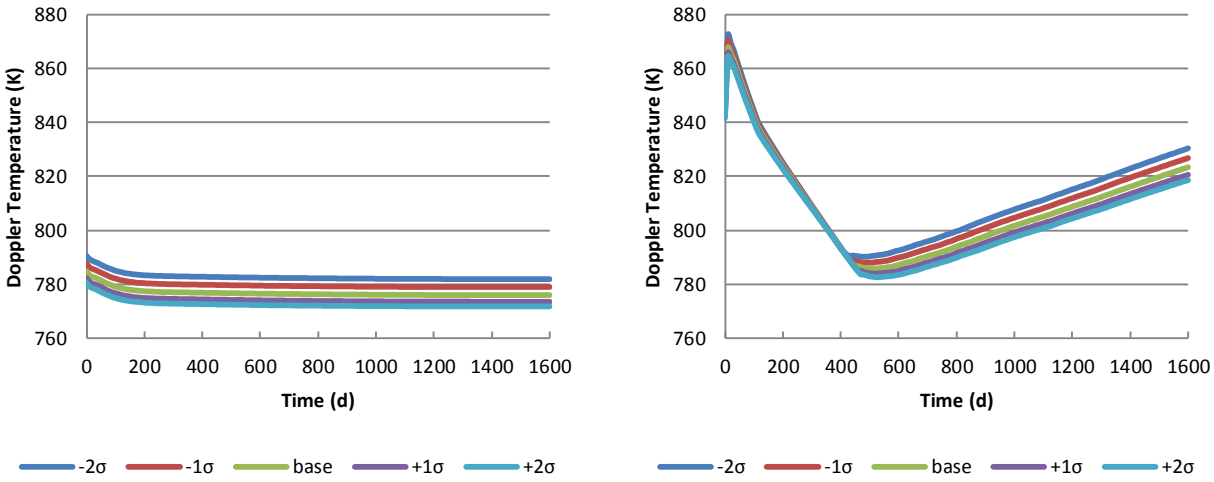
where Nu is the Nusselt number (non-dimensional heat transfer coefficient), Re is the Reynold's number, Pr is the Prandtl number, ρ is the coolant density, v is the coolant velocity, c_p is the coolant specific heat, μ is the coolant viscosity, k is the coolant thermal conductivity, and L is the channel hydraulic diameter. With the coolant solution converged, the Nusselt number can be evaluated. Once this is known, the HTC can be calculated as:

$$h = \frac{\mathrm{Nu}\ k}{L} \tag{4.6}$$

The rod surface temperature is obtained by substituting this expression into Equation 4.4 and solving for T_w.

To perform the sensitivity calculations, PATHS was modified to apply a bias to h on a relative basis in the same manner as Equation 4.2. For the purposes of this exploration, the standard error in h was taken to be σ_h = 20%, and the model biased $\pm 2\sigma_h$ in $1\sigma_h$ increments.

For the purposes of this calculation, the model was executed in both coupled and uncoupled modes to gain understanding of the effect of evaluating sensitivity of depletion to heat transfer in both possible modes. In Figure 4.4.9 we show the impact on Doppler temperature for the uncoupled and coupled cases. Since the power distribution throughout the depletion is unchanged regardless of the method by which the fuel was simulated, the coolant axial temperature distribution (and therefore average moderator temperature) was likewise unaffected. Standalone calculations were performed with a gap conductance of 30000 W/m^2-K.

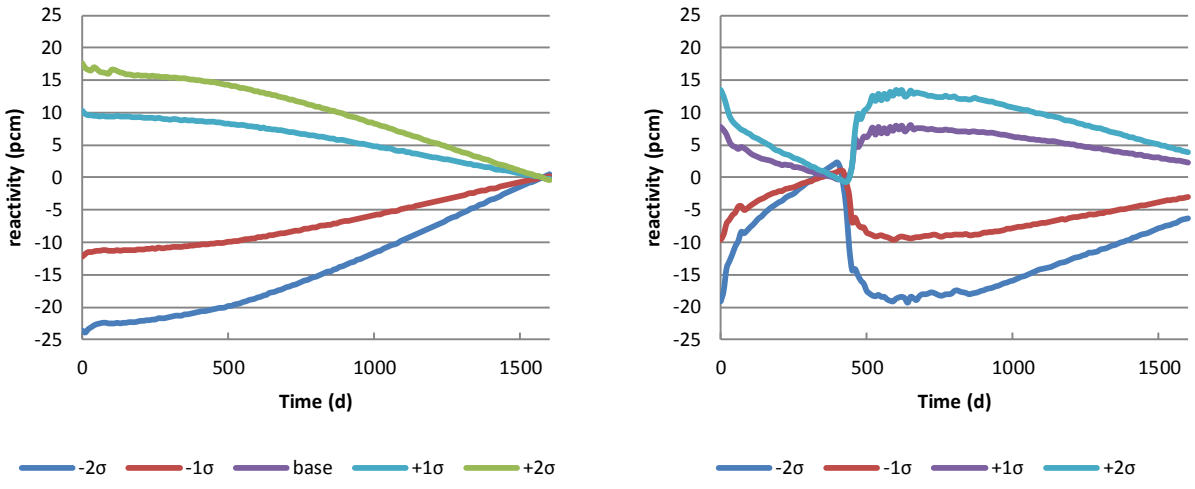


(a) Uncoupled **(b) Coupled**

Figure 4.4.10: Doppler temperatures for HTC sensitivity cases

From this figure, we see similar evolution in Doppler temperature that we saw in the comparisons presented in Section 4.2. It appears that a 20% change in HTC is worth approximately 3 K in Doppler temperature for most of the depletion. For the uncoupled case, the Doppler temperature is essentially constant throughout the calculation; on the other hand, during the gap closure phase of the coupled calculation, the temperatures of all cases become very close. This is because the fuel and cladding are not yet in contact, and therefore the insulating effect of the gap renders the effect of rod surface temperature to be insignificant. Once the gap is closed, however, the coupled case assumes differences between the individual calculations that are very similar to the uncoupled case.

In Figure 4.4.11, we present the respective reactivity effects of gap closure on each set of cases.



(a) Uncoupled **(b) Coupled**

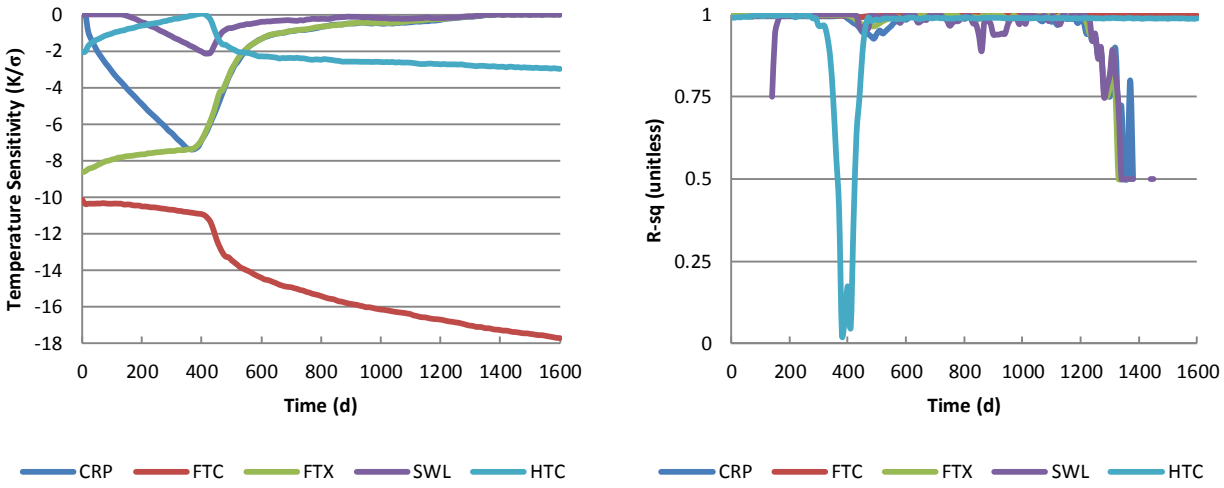
Figure 4.4.11: Reactivity for HTC sensitivity cases

In this figure, we see clear influence of the fuel-clad gap on the calculation. The uncoupled case initially begins with approximately 20 pcm difference at the 2σ level, with the cases biased towards lower HTC being lower in reactivity due to relatively higher temperatures. As the depletion continues, the curves move towards one another since there is excess plutonium building up in the hot cases relative to the cold cases. In the coupled case, we see very little (and decreasing) reactivity effect during the gap closure phase. Once the gap closes, however, the fuel is more directly influenced by the coolant heat transfer conditions, and the shape assumes a similar profile to the uncoupled case.

Examination of other neutronics figures of merit (peaking factors and reactivity coefficients) did not reveal a significant impact from the HTC bias, similar to the other cases. We have found in this study that there is an appreciable, if modest, impact of the heat transfer coefficient on the depletion calculation, both in coupled and uncoupled forms. The influence in the coupled calculation was approximately -19 pcm/+13 pcm at $\pm 2\sigma$ levels, which is comparable to the effect of the fuel swelling model. Therefore, we will include uncertainty in this model in subsequent uncertainty evaluations.

4.4.5 Sensitivity Summary

In this section, we exhaustively examined the effect of each of the fuel uncertainty parameters and the HTC on coupled depletion. We have determined that the depletion calculations are sensitive to fuel thermal conductivity, fuel thermal expansion, cladding creep, fuel swelling, and HTC and therefore warrant inclusion in more detailed sensitivity and uncertainty calculations presented in Chapter 6; the other models were either insignificant, or non-physical. We have also determined that the primary sensitivity lies in the Doppler reactivity relative to the nominal (unbiased) case, with peaking factors and reactivity coefficients being essentially insensitive. To summarize the sensitivity of temperature and reactivity to these parameters as a function of time, a linear regression fit was performed through the five data points computed at each time step. The slope of this fit (in units of K/σ or pcm/σ) gives a measure of the sensitivity, and will provide information on how the relative importance of each parameter changes throughout the depletion. In addition, we calculated the R^2 value at each data point, which is a measure of how appropriate a linear fit is at each time point. R^2 varies from 0 (no fit) to 1 (perfect fit). All this information serves to guide the implementation of more sophisticated sensitivity methods described in Chapter 5.



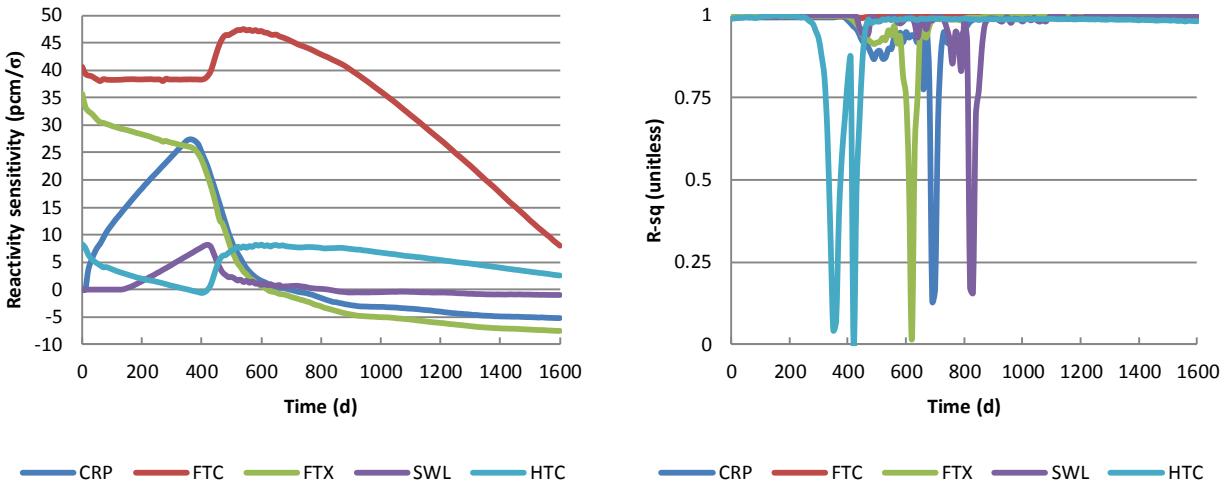
(a) Sensitivity **(b) R^2**

Figure 4.4.12: Sensitivity and R^2 of Doppler temperature

In this figure, we see that the sensitivity to Doppler temperature is for all these models negative; this indicates that biasing the models upwards tends to decrease the temperature. For

the case of the models which affect gap size (creep, swelling, thermal expansion), this is because increasing their value causes the gap to close at a faster rate, which provides additional cooling to the fuel and drives the temperature down. Thermal conductivity controls the temperature of the fuel; by increasing its value, heat transfer through the fuel is enhanced, and therefore the fuel operates at lower temperatures. Increasing HTC causes a lower temperature drop across the thermal boundary layer and therefore decreases fuel temperature. The influence of these various parameters as a function of burnup corresponds with the Doppler temperature plots presented in each subsection above. Temperature dependence on creep and swelling increases as the gap closes, but once thermal contact is established their influence becomes minor. Temperature becomes instantaneously affected by thermal expansion, but its influence decreases once the gap is closed. The heat transfer coefficient is also an important contributor, but primarily after the gap is closed; prior to this, the insulating effect of the gap causes its importance to be minimized as the gap closes. Once it is closed, however, the fuel and coolant are in more direct contact, and therefore it becomes more important.

The information in Figure 4.4.12(b) indicates that, for the most part, the linear fit of each of these parameters to the system response is generally acceptable. The gaps in the creep, swelling, and thermal expansion curves are caused by the locations where sensitivity is computed to be zero; if there is no sensitivity, one cannot apply a line fit. As the gap closes, there are points on the HTC curve which indicate very low correlation, which is due to poor linear fits at times where the HTC is unimportant. This is likewise the case in the creep, swelling, and thermal expansion curves; by these points, the sensitivity to these parameters is essentially (but not identically) zero, which causes the R^2 value to behave erratically. The essential point from this exercise is that, if the Doppler temperature response at most points in time is represented by a linear expansion in each of the models (i.e., using a stochastic expansion method), we expect that the fit will be good. What is not considered in this exercise is potential interactions between the input variables; all of them are varied independently. The sensitivity methods described in Chapter 5 will allow us to evaluate the effect of these interactions in the results we present in Chapter 6.



(a) Sensitivity **(b) R^2**

Figure 4.4.13: Sensitivity and R^2 of reactivity

Figure 4.4.13(a) shows the relative sensitivity of each of the parameters on the core reactivity. As expected, the sensitivity of the thermal conductivity is the strongest. The creep and swelling model become more important as the gap closes. The thermal expansion and heat transfer coefficient show decreasing influence prior to gap closure. After the gap closes, the sensitivity to all the parameters trends downward, as the plutonium buildup occurring earlier in the hotter calculations manifests itself by providing a positive reactivity contribution offsetting the increase in fuel temperature that occurs due to thermal conductivity degradation. As with the temperature plot, Figure 4.4.13(b) shows the R^2 value to be essentially unity for most of the depletion among the parameters; the points at which it deviates reflect the time points where the gap is closing and the relative sensitivities of each of the parameters are not strong. We expect that a linear expansion of the reactivity response will be appropriate for modeling the uncertainty in this value; further exploration of this question and convergence studies will help provide more detailed information in Chapter 6.

With this, we conclude our discussion of the pin cell problem. As we have seen, much of the important physics is represented in this relatively simple problem. The behavior of the fuel-clad gap was found to be very important in modeling the behavior of the neutronics system when taking into account fuel performance. The relative differences in temperature depending on the directions in which the parameters were biased caused competing reactivity effects depending on the instantaneous temperature at which the system was evaluated, and the build-up history of

plutonium that occurred during the depletion. Five models were identified as particularly important and physical; these are the models which will be incorporated in subsequent sensitivity and uncertainty analysis. In the next section, we will discuss the effect of coupling fuel performance on a mini-core depletion model, in order to both determine if the basic processes affecting the pin cell depletion hold true for a larger problem and to gauge the effect of fuel performance on the radial peaking factor.

4.5 Mini-Core Model

While the pin cell model proved to capture much of the important physical effects, it is important to assess the impact of coupling the fuel thermo-mechanics with the neutronics on a better representative of an actual reactor core system. As stated in the introduction to this Chapter, the coupling scheme implemented in FRAPARCS is not yet amenable to full-core calculations due to execution time concerns; however, it is possible to perform evaluations on a multi-assembly model which provides better representation of a full core than a single pin cell.

A 5x5 fuel assembly mini-core model with a loading pattern representative of full PWR cores was developed and tested with FRAPARCS to determine the potential impact of including realistic fuel models in core depletion calculations. A multi-cycle core depletion model was developed to evaluate the effect of having a mixture of fresh and burnt fuel in the core at the same time, with loading and shuffling patterns reflective of those used in industry. As with the pin cell, we compare stand-alone and couple results to determine the importance of fuel performance.

4.5.1 Lattice Model and Cross Section Generation

We use the same cross sections in this evaluation as we used for the preliminary study presented in Chapter 1. The reference plant is the Surry Unit 1 is a 3-loop Westinghouse-designed PWR plant with 15x15 fuel assemblies. Each assembly contains 204 UO2 fuel rods and 21 guide tube thimbles. For the initial Surry core, there were 5 types of assemblies loaded, with the differences primarily being in either enrichment or in burnable poison/power distribution control features. For the purposes of this work, the 5x5 assembly mini-core problem uses the Type 1, 2, and 3 assemblies, which have no burnable poisons and are characterized by differing enrichments and fuel radii. The lattice model input specifications can be found in Appendix C.

Similar to the pin cell problem, history and branch structures were developed to envelop the operating space of the Surry Unit 1 core. Instead of using moderator temperature to parameterize the cross sections, moderator density was used; otherwise, the structure and philosophy are generally similar to the pin cell case. The core conditions and depletion step structure can be found in Appendix C. Using these geometry, materials, and case and depletion step structure, HELIOS was used to perform the lattice calculations, the outputs of which were processed with ZENITH and GenPMAX to generate the cross section files used by PARCS/PATHS. With these data, the mini-core models were assembled and executed.

4.5.2 FRAPARCS Input Specification

First, we describe the loading patterns used in this computation. The initial core loading pattern was developed with a mixture of medium- and highly-enriched assemblies arranged in roughly a checkerboard pattern surrounded by a ring of primarily lower-enriched assemblies, with reflective boundary conditions on the radial periphery. Because an even mixture of the three assembly types was desired, a few highly-enriched assemblies were also placed near the periphery. This is not typically done in operating cores due to neutron economy and vessel fluence concerns; however, the selected loading pattern achieves a power distribution and nodal flux ratios which are roughly equivalent to full-scale PWR systems. The reflective boundary conditions serve to alleviate excessive peaking that would result from such a small core with zero-incoming flux boundary conditions. The initial core loading patter is presented in Figure 4.5.1.

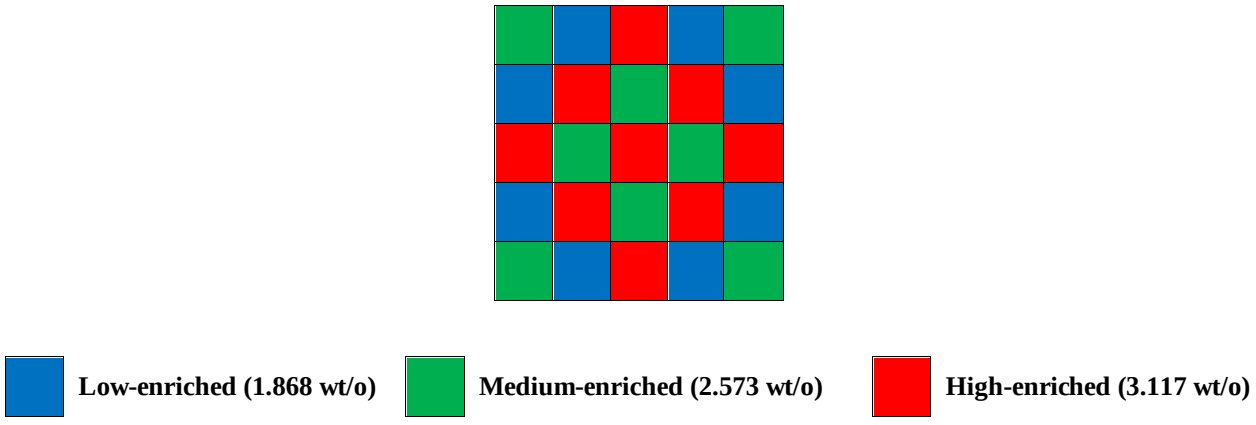
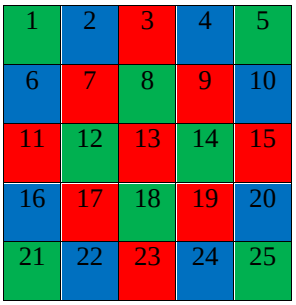
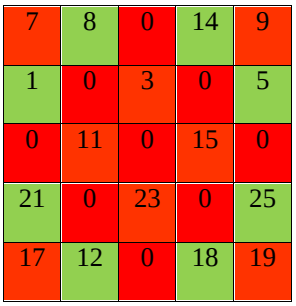


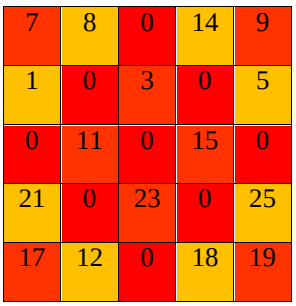
Figure 4.5.1: Initial loading pattern for multi-cycle depletion



(a) First cycle (numbers denote assembly position)



(b) Second cycle (numbers denote position in previous cycle; 0 means fresh)



(c) Third cycle (numbers denote position in previous cycle; 0 means fresh)

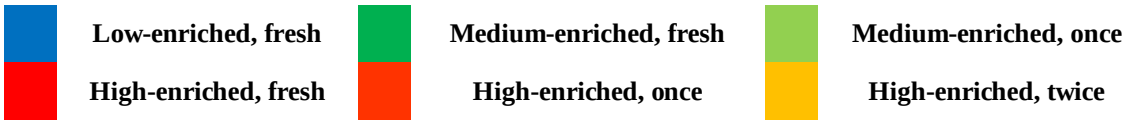


Figure 4.5.2: Shuffling pattern for multicycle depletion calculation

The multicycle depletion calculations began with the loading pattern in Figure 4.5.1. A fuel shuffling strategy was developed to move assemblies around the core in a manner analogous to industry practice. After 540 days of depletion (about 18 months), the reactor core was shuffled; the lower-enriched assemblies and the higher-enriched assembly in the center of the core were removed, the middle-enriched assemblies were shuffled into the place of the lower-enriched assemblies, and the higher-enriched assemblies placed where the middle-enriched assemblies once were. To replace the discharged assemblies, fresh higher-enriched feed

assemblies were inserted. The process was repeated for an additional cycle. The shuffle maps are presented in Figure 4.5.2.

By the beginning of the third cycle, all fuel of Types 1 and 2 (low- and medium-enriched, respectively) have been removed from the core and only Type 3 assemblies at differing levels of initial burnup are present. In order to model this in PARCS/PATHS, a program called SHUFFLER was developed to read the depletion file printed out at the end of the previous cycle by PARCS and move each assembly's axial burnup, fuel history temperature, and moderator history density to their new locations, while filling in the initial values of these quantities used for fresh assemblies. To accommodate shuffling in the FRAPCON calculations, the FRAPARCS script was used to shuffle the fuel pins representing specific assemblies in accordance with the shuffle map used above, and new cases were spawned for fresh assemblies. As a check to ensure that FRAPARCS was moving the fuel appropriately, a tag in each of the FRAPCON input decks was used to denote the fuel rod's first, second, and third cycle position. The PARCS/PATHS and FRAPCON input specifications are given in Appendix C. As with the pin cell problem, care was taken to ensure consistency between the lattice, neutronic, thermal-hydraulic, and fuel models. With the input specified, we next present the results of the calculations.

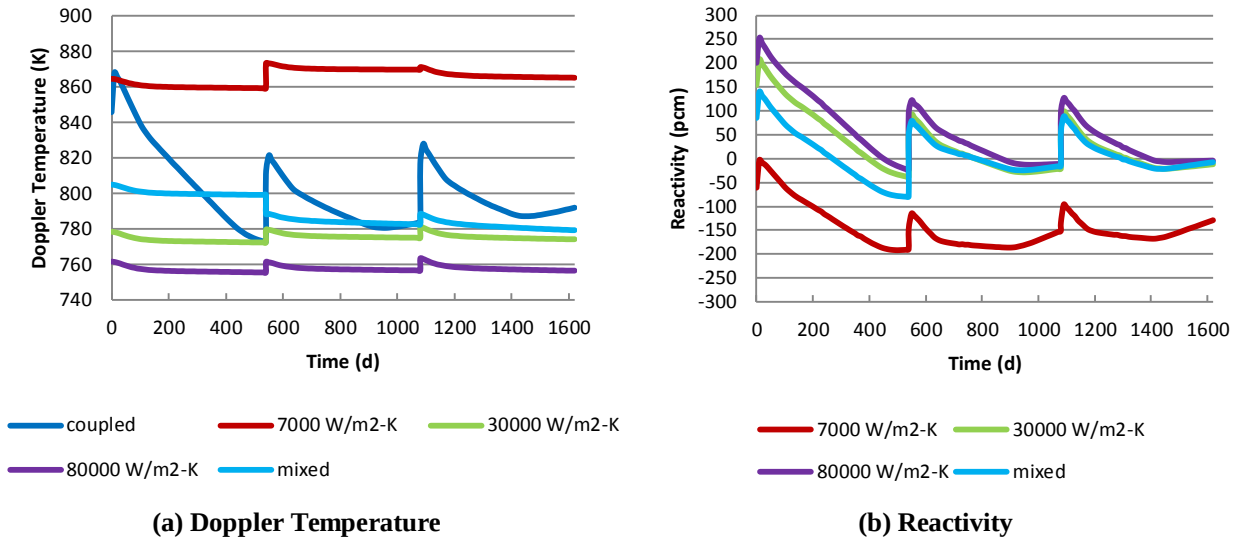
4.5.3 Comparisons with Stand-alone Computations

To provide a basis of comparison between the stand-alone models and the coupled calculation, we employed the same three gap conductances as were used in the pin cell cases. In addition, we took advantage of the break between cycles to attempt to in some way model the gap changing from cycle-to-cycle by incorporating a mixed conductance case where the conductances of the first, second, and third-cycle assemblies were given values representative of their exposure. This data is provided in Table 4.5.1:

Table 4.5.1: Gap conductances employed for mixed case

Fuel condition	Gap conductance (W/m^2-K)
Fresh	15000
Once-burnt	40000
Twice-burnt	100000

We can now present the results from the four stand-alone cases in comparison with the coupled case. In Figure 4.5.3, the Doppler temperature and reactivity are presented.



(a) Doppler Temperature **(b) Reactivity**

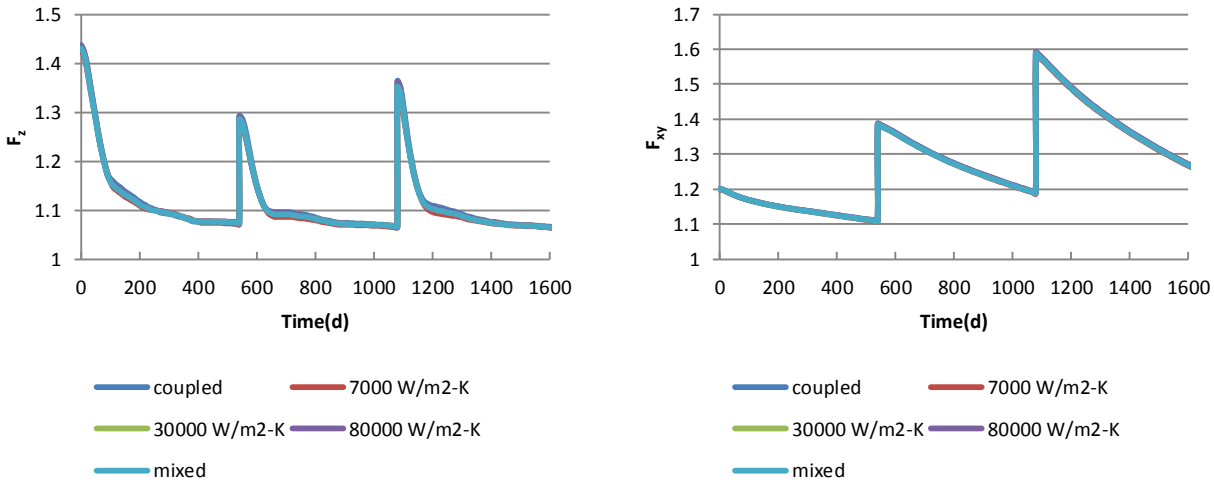
Figure 4.5.3: Doppler temperature and reactivity for multi-cycle depletion

We can see in Figure 4.5.3(a) that even with some clever choices of representative gap conductances in the mixed case, the Doppler temperature cannot be captured accurately without the use of an explicit physics model. The difference in temperature is most pronounced for the first cycle, where the entire core is comprised of fresh fuel. When the cycle ends and the fuel is shuffled, the fresh fuel that is introduced produces highly noticeable "spikes" in the Doppler temperature as the contribution of the fresh fuel with its relatively large gap is felt. The high-burnup effects on fuel thermal conductivity and gap conductance become evident near the end of cycle two and more obviously in the last third of cycle three, as evidenced by the slight upturn in Doppler temperature. The small steps in the 7000 W/m2-K case are due to the introduction of Type 3 fuel, which has a smaller fuel pellet diameter and thus a slightly higher fuel temperature.

Similar reactivity behavior is present in Figure 4.5.3(b) as in the single depletion cases. There is a large discrepancy on the order of 200 pcm initially due to the difference in Doppler temperature which begins to decrease as the fuel-cladding gap closes; the reactivity swing from maximum to minimum difference is somewhere between 200-300 pcm depending on the case in question. The gap closes about the time the first cycle completes; the fresh fuel which is subsequently loaded contributes to "spikes" in reactivity (directly related to the spikes in Doppler temperature); the subsequent reactivity swings are about half the swings seen from the purely fresh core, since only one third of the core is replaced with fresh fuel. This highlights the particular importance of fuel modeling for fresh cores; while the effect is still important for

standard reload cores, it is especially important for the initial cycle. This is a pertinent finding considering the number of nuclear reactor plants currently under construction or consideration across the world at this point in time.

The next comparison which is of interest is in the radial peaking factor. It is possible that, with the introduction of fresh, high-powered fuel next to burned, low-power fuel, that we might see some influence of fuel performance on the radial power distribution. The axial and radial peaking factors are presented in Figure 4.5.4:



(a) Axial **(b) Radial**

Figure 4.5.4: Axial and radial peaking factors for multi-cycle depletion

We see from these results that the concerns over radial power distribution anomalies are unfounded. The impact on radial peaking factor is consistent with the Chapter 1 comparisons, and is even less than that of the axial peaking factor. While the radial peaking factor increases significantly at reload because of the mixture of fresh and burned fuel, there is not a significant difference in the coupled case as opposed to the stand-alone cases. Again, this is because incorporating changing fuel performance is significant to the effective Doppler temperature averaged over the core, locally it only causes the magnitude of the temperature to change and not the basic profile.

In addition to Doppler reactivity and peaking factors, we also ran cases to determine the effect on mini-core reactivity coefficients. We found the same behavior in the mini-core problem that we did in the single pin-cell case; namely, these parameters were not significantly affected.

4.5.4 Mini-Core Conclusions

We have constructed a mini-core problem representative of a PWR in operation. The basic physics between the pin cell calculations and the mini-core depletion were in general the same. The major difference was in the reactivity behavior of the mixed cycles; once there was a mixture of fresh and burned fuel included simultaneously in the core, the reactivity swing throughout the cycle (i.e., maximum difference between stand-alone and coupled depletion) became roughly half as much as a fully-fresh core. The other neutronics parameters remain largely unaffected because it is the magnitude of the temperature rather than its distribution which is affected by including the effects of fuel performance.

4.6 Conclusions

In this Chapter, we have described the application of FRAPARCS to both a pin-cell and a multi-cycle mini-core problem. We found that the major influence of including fuel physics was in the Doppler reactivity computed throughout the depletion cycle. Other neutronics parameters, such as reactivity coefficients and peaking factors, were minimally influenced. Based on these results, and because of practical computational concerns, we can perform sensitivity and uncertainty analysis of the coupled calculation with the relatively fast-running pin cell model, and still make general conclusions about the influence of fuel performance on neutronic depletion. In addition, we evaluated the individual effect of each fuel uncertainty variable available in FRAPCON, as well at the heat transfer coefficient. We found that four of the fuel variables were either unphysical or insignificant for the cases presented here, but that four of them, the fuel thermal conductivity, fuel thermal expansion, cladding creep, and fuel swelling, had significant effects on the nuclear performance of the core. In addition, we determined that the heat transfer coefficient was sufficiently influential to include in further analysis. In the next Chapter, we will discuss the uncertainty methods available in DAKOTA which we have applied to this work. This will set the stage for the discussion in Chapter 6, which presents the robust application of sensitivity and uncertainty analysis methods to the coupled depletion problem.

CHAPTER 5

Uncertainty and Sensitivity Analysis Methods

5.1 Overview

In the previous chapters, the basis for and mechanics of coupling fuel performance and core depletion codes were described, and example calculations demonstrating the impact of this coupling were presented. In this Chapter, we present the theoretical basis for sensitivity and uncertainty methods in general, and then describe specifics regarding the individual methods in DAKOTA applied to this work in particular. In Chapter 6, we will apply the ideas described here to the coupled pin cell problem in order to precisely quantify the sensitivity and uncertainty in neutronics to fuel performance parameters.

As we have discussed previously, the purpose of uncertainty analysis is to quantify the relationship between the uncertainties in the input and problem parameters and the resulting uncertainty distribution of the outputs. There are two general classes of uncertainties [7]:

- *Aleatory* uncertainties are irreducible variabilities inherent in nature
- *Epistemic* uncertainties are reducible uncertainties which arise from a lack of knowledge

The models considered here are considered aleatory, since sufficient data exists to generate probability distributions describing their nature, and therefore probabilistic methods can be employed to evaluate their influence. Epistemic uncertainties, on the other hand, are generally not characterized by probabilistic methods; for example, expert opinion may provide intervals in which an epistemic variable may lay, with no information available about the underlying distribution. Special classes of uncertainty techniques have been developed to perform analysis

where epistemic uncertainty is present (for example, Dempster-Shafer theory of evidence); as our uncertainty variables are aleatory, this topic will not be further explored here.

In order to quantify the response uncertainty, the input uncertainties must be propagated through the numerical model in some way, and the response statistics tabulated or computed. Illustrated graphically:

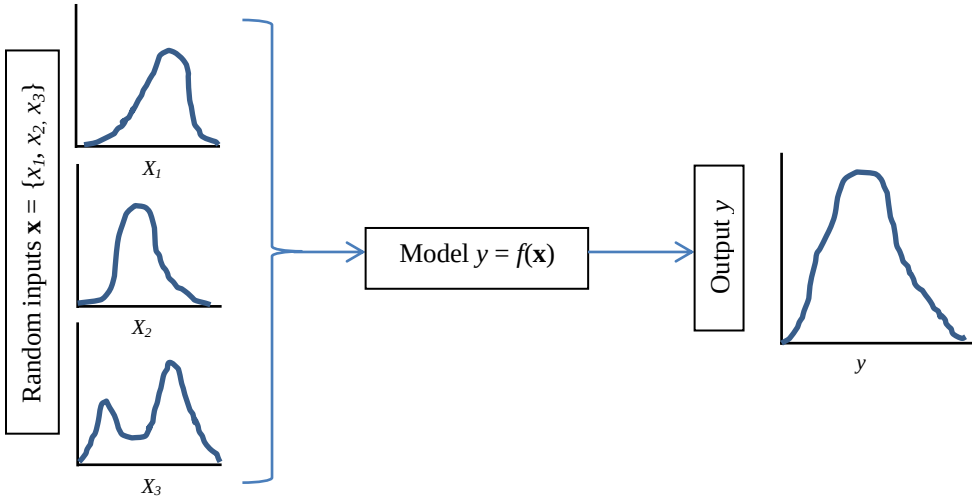


Figure 5.1.1: Uncertainty propagation through a numerical model

In Figure 5.1.1, the basic approach to uncertainty calculations is illustrated. The output in question, y, is a function of three random input parameters $\mathbf{x} = \{x_1, x_2, x_3\}$, each subject to a given probability distribution function (pdf). The goal of the analysis is to in some way propagate those pdf's through the numerical model and estimate the resulting pdf of the output variable subject to those input distributions, from which statistics can be obtained. The output distributions can then be, for example, compared with performance criteria to determine the likelihood that the system will operate within their specified bounds.

There are two basic approaches used to propagate input uncertainties: extrusive (or black box) methods and intrusive methods. In the former, the computational model in question (i.e., computer code) is largely unmodified from its deterministic state. Calculations using the model are repeated a number of times with the uncertain inputs perturbed in accordance with whichever uncertainty analysis technique is to be employed. The resulting collection of outputs is then analyzed to determine statistics of interest, such as means, variances, and confidence and tolerance intervals. Intrusive methods, on the other hand, require that the original model be recast from a deterministic into a stochastic formulation. The advantage of intrusive methods is that they provide deterministic evaluations of the response statistics; the disadvantage is the potential

mathematical complexity and expense involved in developing probabilistic representations of the computational model. For the purposes of this work, non-intrusive methods, including Random Sampling (RS), Latin Hypercube Sampling (LHS), and non-intrusive formulations of Polynomial Chaos Expansion (PCE) and Stochastic Collocation (SC) are employed, given the complexity in particular of the material behavior modeling in FRAPCON.

Once the uncertainty analysis has been performed and the input and response statistics have been generated, sensitivity analysis can be performed to quantify the relative importance of each input variable on the response. In this work, we use Sobol'-expansion based variance decomposition to perform this evaluation, since, as we will describe, it follows relatively easily out of the stochastic expansions.

Before delving into the description of each uncertainty analysis technique, we will present some basic statistical definitions which are used to describe the inputs and responses. First, let us define the *probability density function* (pdf) as a function that describes the relative likelihood that a random variable takes on a value within a particular range. If x is a continuous random variable that is described by pdf $f(x)$, then:

$$P[a < x < b] = \int_a^b f(x)dx \tag{5.1}$$

is the probability that x will take on a value within the interval from a to b. The pdf is defined as a non-negative function such that the integral over the whole support space of x is 1; i.e., if one knows that x must take a value between bounds x_{min} and x_{max}, then the probability that x takes a value within that range is given by:

$$P[x_{min} < x < x_{max}] = \int_{x_{min}}^{x_{max}} f(x)dx = 1 \tag{5.2}$$

Depending on the pdf in question, those upper and lower bounds could extend to $\pm\infty$, respectively. Common pdf's include the uniform distribution, the normal distribution and the log-normal distribution; there are several other examples.

A related function to the pdf is the *cumulative distribution function* (CDF) $F(y)$, which defines the probability that a random variable will take a value less than y:

$$F(y) = \int_{x_{min}}^{y} f(x)dx \tag{5.3}$$

Obviously the CDF is a non-decreasing function of y which takes values between 0 at $y \leq x_{min}$ and 1 at $y \geq x_{max}$. Because of the fundamental theorem of calculus, the pdf is related to the CDF as:

$$f(y) = \left.\frac{dF}{dx}\right|_{x=y} \tag{5.4}$$

These definitions can be generalized for multivariate functions of random variables $\boldsymbol{x} = [x_1, x_2, \dots, x_n]$, in which case the pdf is referred to as the *joint density*:

$$\begin{aligned} &P[a_1 < x_1 < b_1, a_2 < x_2 < b_2, \dots, a_n < x_n < b_n] \\ &= \int_{a_n}^{b_n} \dots \int_{a_2}^{b_2} \int_{a_1}^{b_1} f(\boldsymbol{x}) dx_1 dx_2 \dots dx_n \end{aligned} \tag{5.5}$$

The same normalization condition (i.e., integration over the support space of f yields 1) and non-negativity requirement hold for joint densities. Using the joint density, one can also define the *marginal density* $f_{X_i}(x_i)$ by integrating out the other variables $x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n$ over their respective ranges of support:

$$f_{X_i}(x_i) = \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} f(\boldsymbol{x}) dx_1 \dots dx_{i-1} dx_{i+1} \dots dx_n \tag{5.6}$$

Using this definition for marginal density, one can then define the *conditional density* $f_{-i}(\boldsymbol{x}_{-i}|x_i)$ as:

$$f_{-i}(\boldsymbol{x}_{-i}|x_i) = \frac{f(\boldsymbol{x})}{f_{X_i}(x_i)} \tag{5.7}$$

where $\boldsymbol{x}_{-i} = [x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n]$.

Before describing the calculation of the moments of a pdf, we define the concept of *independence*. That is, if the joint distribution can be decomposed into the product of each variable's marginal densities:

$$f(\boldsymbol{x}) = \prod_{i=1}^{n} f_{X_i}(x_i) \tag{5.8}$$

then the random variables $x_1, x_2, \dots, x_n$ are said to be independent. Generally speaking, independence among the uncertain input variables greatly simplifies the application of uncertainty analysis methods. In the cases where there is some inter-dependence among the inputs, special orthogonalization techniques can be applied to produce an equivalent system with independent transformed inputs (see, for example, [156]).

With these basic statistical definitions in mind, we move on to defining the moments of the output distribution, which are of interest in uncertainty and sensitivity analysis. Suppose g is a function of random variables $\boldsymbol{x} = \{x_1, x_2, \dots, x_n\}$ whose distributions are given by the joint density function $f(\boldsymbol{x})$. Then, the *expected value* (i.e., *mean*) of g is given by:

$$E[g(\boldsymbol{x})] = \mu_g = \int_{\Omega} g(\boldsymbol{x}) f(\boldsymbol{x}) d\Omega \tag{5.9}$$

where Ω is the support space of $\boldsymbol{x}$**.** In simple terms, the mean is simply the average value of g subject to the joint density function *f*. While the expected value is most definitely a useful way to characterize the possible distribution of g, it is by no means sufficient, because it functions as a point quantity that represents only the average value of g. In order to describe other aspects of g's distribution, additional higher moments must be taken. The first of these is the second moment of g subject to $\boldsymbol{x}$, which is referred to as the *variance* of g:

$$V[g(\boldsymbol{x})] = \sigma_g^2 = \int_{\Omega} \left(g(\boldsymbol{x}) - \mu_g\right)^2 f(\boldsymbol{x}) d\Omega \tag{5.10}$$

The variance is related to the spread in the spread in the possible function values; in other words, a high variance is indicative of a function that has a relatively high probability of taking values far from the mean. A low variance, on the other hand, means that there is a high likelihood that any particular sampled value of the function g will lie near the mean. The variance is related to the *standard deviation* $\sigma_g = \sqrt{\sigma_g^2}$, which is of the same units as the function g. This makes for

intuitive comparison of the spread in the data relative to the mean value. A related quantity is the *covariance* between two functions $g_1(\boldsymbol{x})$ and $g_2(\boldsymbol{x})$:

$$cov[g_1(\boldsymbol{x}), g_2(\boldsymbol{x})] = \sigma^2_{g_1,g_2} = \int_{\Omega} \big(g_1(\boldsymbol{x}) - \mu_{g_1}\big)\big(g_2(\boldsymbol{x}) - \mu_{g_2}\big) f(\boldsymbol{x}) d\Omega \tag{5.11}$$

This is a measure of how two functions change with respect to one another; if $cov[g_1(\boldsymbol{x}), g_2(\boldsymbol{x})]$ is positive, the two functions tend to change in the same direction, whereas if $cov[g_1(\boldsymbol{x}), g_2(\boldsymbol{x})]$ is negative, they trend in the opposite direction. This definition becomes important in the definition of correlation coefficients, described below.

In addition to the mean and standard deviation, higher order moments of g can be computed to provide additional information on its distribution. The third moment is referred to as the *skewness*, and is related to the lack of symmetry about the mean value. A positive skewness indicates more of the distribution is shifted above (to the right) of the mean, whereas a negative skewness implies more data shifted below (to the left) of the mean. Finally, the fourth moment is referred to as the *kurtosis*, which is a measure of the peakedness of the distribution. A large kurtosis indicates a highly peaked distribution, while a low kurtosis indicates a flat distribution. Kurtosis is usually judged relative to the standard normal distribution, which has a kurtosis value of about 3.

With these definitions in mind, we re-emphasize that the purpose of uncertainty analysis within the context is to determine and characterize the response uncertainty distribution subject to uncertain input variables. Quantification of the response uncertainty includes estimation of the pdf/cdf, the mean, variance, and higher-order moments. Using this information, the contribution of each input to the resulting variance in the output distributions can be estimated via sensitivity analysis methods. With this clarification, we move to discussion of specific methods from which the response distribution can be estimated.

5.2 Sampling-based Methods

In this section, we describe the two sampling based methods used in this work: Random Sampling (RS) and Latin Hypercube Sampling (LHS).

5.2.1 Random Sampling

The most intuitive and straight-forward approach to uncertainty analysis is Random Sampling. With this method, a value for each of the uncertain variables is sampled at random according to their prescribed PDFs. These values are then propagated through the model (i.e., used in a simulation), and the responses of interest are tabulated. This process is repeated *N* times to generate estimates for the response mean $\hat{\mu}_y$ and standard deviation $\hat{\sigma}_y$:

$$\hat{\mu}_y = \frac{1}{N}\sum_{n=1}^{N} y_i \tag{5.12}$$

$$\hat{\sigma}_y = \left(\frac{1}{N-1}\sum_{n=1}^{N}\left(y_i - \hat{\mu}_y\right)^2\right)^{1/2} \tag{5.13}$$

where y_i is the response calculated for the i^{th} sample. In a likewise fashion higher-order statistics, such as skewness and kurtosis can be estimated. The response cdf can be estimated by using the following function:

$$D(y) = \frac{1}{N}\sum_{i=1}^{N} u(y - y_i) \tag{5.14}$$

where $u(z) = 1$ if $z > 0$ and zero otherwise. In a likewise fashion, one can define an empirical pdf by generating a histogram of the response and counting the number of observations within each bin.

The main advantages to RS are that it is conceptually easy to understand, and that the cost is largely independent of the number of uncertain input parameters. The disadvantage lies primarily in the computational cost that may be required to generate sufficiently resolved statistics, particularly if each individual response evaluation (i.e., simulation) requires more than a few seconds to complete. In particular, since random sampling does not a priori guarantee complete coverage of the sample space, many evaluations are required to ensure that samples are drawn from lower-probability regions so that their impact on the response statistics can be captured. Also, since the estimates of statistical quantities are themselves uncertain variables (i.e., each individual set of RS runs generates a unique estimate for the true mean and variance). It can be shown that, with RS, the estimate of the mean converges to the true mean as $1/\sqrt{N}$,

where *N* is the number of samples [157]. Obviously, depending on the cost of each evaluation and the level of confidence desired in the estimate of the mean, this can result in an excessive computational expense. To attempt to mitigate some of these drawbacks, Latin Hypercube Sampling was developed.

5.2.2 Latin Hypercube Sampling

In order to improve sample space coverage and produce more precise estimates of true values with the same number of response evaluations as RS, the Latin Hypercube approach was developed ([157], [158]). LHS is a constrained sampling method which subdivides the sample space of each input variable into *N* equal-probability bins, where *N* is the total number of samples. Suppose there are a total of *m* random input variables. A sample is drawn randomly from each bin for each variable, to yield *m* sequences of *N* variables, $\{x_1^i, x_2^i, \dots, x_N^i\}, i = 1,2, \dots m$, which are arranged in ascending order. In order to form the Latin Hypercube of samples, each sequence is randomly permuted, and the resulting permuted sequences are combined to form *N* sets of *m* variables which provide a relatively even covering of the sample space.

In order to better illustrate this process, consider a situation where there are two random variables x_1 and x_2, which are both uniform with lower bound equal to 0 and upper bound equal to 1, from which we wish to draw 10 samples for evaluation of the function $g(x_1, x_2)$. In this case, a random sample is drawn from the 10 intervals, as shown in Table 5.2.1:

Table 5.2.1: LHS example, 10 random samples from two uniform random variables

Bin Range				
lower	**upper**	**bin index**	x_1	x_2
0.0	0.1	1	0.025288	0.072574
0.1	0.2	2	0.132135	0.156519
0.2	0.3	3	0.229848	0.230081
0.3	0.4	4	0.37909	0.340184
0.4	0.5	5	0.475412	0.48707
0.5	0.6	6	0.577243	0.519017
0.6	0.7	7	0.667137	0.65478
0.7	0.8	8	0.749587	0.724337
0.8	0.9	9	0.828822	0.869256
0.9	1.0	10	0.997868	0.928379

The next step is to permute each sequence of x_1 and x_2 by their bin indices. In this case, a random permutation of indices might be:

- Permutation set for x_1: $(10, 9, 5, 7, 3, 6, 1, 4, 2, 8)$
- Permutation set for x_2: $(3, 5, 7, 9, 2, 6, 10, 4, 1, 8)$

The final step is to combine the pair the permuted sequences to generate the 10 pairs of x_1 and x_2 which will be used in the evaluations of $g(x_1, x_2)$:

Table 5.2.2: LHS example, sequence of variables used to evaluate $g(x_1, x_2)$

Simulation Index	x_1 bin index	x_2 bin index
1	10	3
2	9	5
3	5	7
4	7	9
5	3	2
6	6	6
7	1	10
8	4	4
9	2	1
10	8	8

With this sequence of pairings defined, Figure 5.2.1 illustrates the difference in the samples drawn using LHS as opposed to simple RS. It is obvious this figure that a simple random sampling can cause clustering of sample points, while leaving some regions of the sample space completely untouched. LHS, on the other hand, achieves better coverage of the entire region. As evidenced by this figure, there are still some regions of the space that neither LHS nor RS manage to access. The obvious solution is to increase the number of samples; whereas clustering is often an issue with RS even with a large number of samples, LHS gives some guarantee of even coverage.

Obviously the brief illustration here can be extended to an arbitrary number of random variables of any distribution type. One potential issue with LHS is the potential for introducing artificial correlations between the inputs, depending on the order in which they are paired. Additionally, the random variables may exhibit a desired correlation structure, and the analyst would like to ensure that it is captured by the sampling procedure. Iman and Conover [159] proposed a method by which the desired correlation structure can be obtained via a refined

permutation procedure which does not pair the sequences of bins randomly, but rather chooses a pairing which closely obtains the desired correlations.

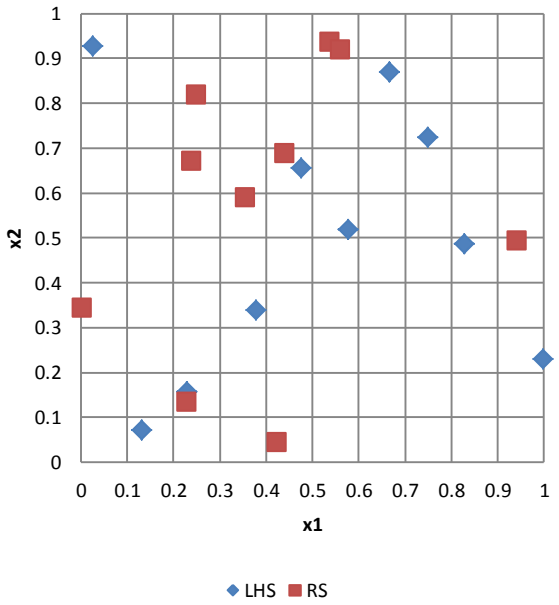


Figure 5.2.1: Contrast between RS and LHS for two uniform variables, *N* = 10

McKay et at showed that the estimates of means and variance produced via LHS are less than that from RS [158]. If we define a class of estimators T as:

$$T(y_1, y_2, \dots, y_N) = \frac{1}{N}\sum_{i=1}^{N} g(y_i) \tag{5.15}$$

where g is an arbitrary function, and the expected value (i.e., true or population mean) of T is τ, then McKay et al proved that LHS yields unbiased estimators of τ. If we take $g(y_i) = y_i$, then T is the sample mean; higher-order moments can be obtained in a similar fashion. Furthermore, they showed that if the response $y = f(\boldsymbol{x})$ is monotonic in each of its arguments x_i (which the sensitivity studies presented in Chapter 4 indicated) and $g(y)$ is monotonic in y, then the variance in the estimator T using LHS is less than or equal to the variance obtained using simple random sampling. In many cases, the variance in estimators from LHS is significantly lower than RS, which is why it is a popular choice for sampling-based uncertainty analysis.

In this section, random sampling methods were described for response distribution characterization. The main advantage is the conceptual simplicity and discipline independence; the main disadvantage is the potential computational expense. In general, LHS provides for lower-variance estimates of response distribution quantities. Even so, in many cases the required accuracy in the statistics is not computationally obtainable with sampling-based methods. Therefore, alternative methods have been explored and implemented. Two of these methods, both based on stochastic expansion, will be explored in the following section.

5.3 Stochastic Expansion Methods

An alternative approach to empirically constructing approximations to response statistics via sampling methods is to develop functional representations of the response with respect to the input variables. So-called response surface approximations (a.k.a. trend models, surrogate models, meta-models) have their roots in the work of Meyers et al ([113], [160]), which sought to develop design of experiments methods. The underlying idea is to represent the response in an analytical form (i.e., as a hypersurface), the coefficients of which are determined by relatively few functional evaluations. The surface can then be used directly in evaluating response statistics, or it can be used in conjunction with a sampling method to inexpensively generate a large number of samples.

A number of different approaches have been explored to represent responses functionally, from simple polynomial regression fits to more advanced approaches such as kriging interpolation or multivariate adaptive regression splines (MARS) [161]. The complexity of the approach is largely dictated by the degree of non-linearity expected in the response; for certain types of physics (such as complex CFD simulations), the response may exhibit a complicated peak and valley structure that simple representations cannot capture accurately.

For this work, stochastic expansion representations of the system response are considered. Both Polynomial Chaos Expansion and Stochastic Collocation seek to construct the functional representation with a linear combination of polynomials. The ideas behind the two methods are distinct; in PCE, the functional expansion is obtained by solving a system of equations which provides the expansion coefficients for known orthogonal basis functions. In contrast, SC requires the determination of grid points and weights for quadrature rules which

allow the functional expansion to be integrated; by construction, the expansion coefficients are simply the values of the response function evaluated at each particular grid point. These ideas will be further explored in the following sections.

5.3.1 Polynomial Chaos Expansion

As stated above, the idea behind PCE is to form a functional expansion of the response in terms of polynomial chaoses. In the original intrusive formulation, polynomials were chosen with certain orthogonality properties. The expansion could then be inserted into the mathematical model, and by a procedure similar to the development of finite-element equations, expansion coefficients can be determined [162]. This work uses related non-intrusive formulations to estimate the expansion coefficients; these will described below.

First, we describe the polynomials with which the PCE is generated. The key feature of PCE is that basis polynomials for each uncertain input are chosen such that they are orthogonal with respect to the pdf of that variable. It is well-established that orthogonal basis polynomials can be defined via a three-term recurrence relationship for arbitrary density functions [163]. In other words, there exists a family of polynomials $p_n(x)$ for any density function $f(x)$ (where n refers to the order of the polynomial) such that:

$$\int_{\Omega} p_n(x)p_m(x)f(x)dx = C_n\delta_{mn} \tag{5.16}$$

where Ω is the support of x, C_n is a multiplicative constant and δ_{mn} is the Kronecker delta (i.e., $\delta_{mn} = 1$ if $m = n$ and zero otherwise). It was shown in [164] that the standard continuous probability density functions have associated hypergeometric orthogonal polynomials (known as the Askey scheme) which satisfy the weighted orthogonality property. For example, Hermite polynomials are orthogonal with respect to the standard normal distribution; Legendre polynomials are orthogonal with respect to the uniform distribution. These polynomials provide an optimal basis for their respective density functions by virtue of the orthogonality property; i.e., as the degree of approximation is increased, the polynomial approximation converges exponentially to the distribution. It is possible to use, for example, Hermite polynomials to represent arbitrary input uncertainties, but convergence with increasing polynomial order is sub-optimal.

Non-intrusive (or black box) PCE has been developed to take advantage of the fundamental ideas of PCE without requiring the extensive investment in code modifications necessary for classical PCE. The idea is to expand the response function R into an infinite series of polynomials in the uncertain input variables $\boldsymbol{\xi} = \{\xi_1, \xi_2, \dots, \xi_m\}$, each with a prescribed density function. Adapting the nomenclature used in [156]:

$$R(\boldsymbol{\xi}) = a_0 B_0 + \sum_{i_1=1}^{m} a_{i_1} B_1(\xi_{i_1}) + \sum_{i_1=1}^{m} \sum_{i_2=1}^{i_1} a_{i_1 i_2} B_2(\xi_{i_1}, \xi_{i_2}) + \sum_{i_1=1}^{m} \sum_{i_2=1}^{i_1} \sum_{i_3=1}^{i_2} a_{i_1 i_2 i_3} B_3(\xi_{i_1}, \xi_{i_2}, \xi_{i_3}) + \cdots \tag{5.17}$$

where $B_n(\xi_{i_1}, \xi_{i_2}, \dots, \xi_{i_n})$ indicates a polynomial of degree n. This particular expansion scheme is referred to as *total order* expansion in that the highest degree of polynomial present is defined by choosing a maximum value of n. This notation can be condensed to a single index-based scheme via a one-to-one correspondence between $a_{i_1 i_2 \dots i_n}$ and α_j and between $B_n(\xi_{i_1}, \xi_{i_2}, \dots, \xi_{i_n})$ and $\Psi_j(\boldsymbol{\xi})$:

$$R(\boldsymbol{\xi}) = \sum_{j=0}^{\infty} \alpha_j \Psi_j(\boldsymbol{\xi}) \tag{5.18}$$

In actual implementation, the expansion is truncated to an upper limit of P .The multivariate polynomials $\Psi_j(\boldsymbol{\xi})$ are constructed as products of the hypergeometric polynomials that correspond with the distribution of each random variable. To illustrate this point, suppose the variability in a mathematical model is represented by two uncertain input variables: one uniform (ξ_1) and one normal (ξ_2). It is desired to represent the response using a total order expansion of degree 2. The first three Legendre and Hermite polynomials (for expansion of uniform and normal variables, respectively) are given in Table 5.3.1:

Table 5.3.1: Legendre and Hermite polynomials up to order 2

Legendre	Hermite
$P_0(\xi) = 1$	$He_0(\xi) = 1$
$P_1(\xi) = \xi$	$He_1(\xi) = \xi$
$P_2(\xi) = \frac{1}{2}(3\xi^2 - 1)$	$He_2(\xi) = \xi^2 - 1$

In this case, the expansion polynomials would be:

$$\begin{aligned}
\Psi_0(\xi_1,\xi_2) &= P_0(\xi_1)He_0(\xi_2) = 1 \\
\Psi_1(\xi_1,\xi_2) &= P_1(\xi_1)He_0(\xi_2) = \xi_1 \\
\Psi_2(\xi_1,\xi_2) &= P_0(\xi_1)He_1(\xi_2) = \xi_2 \\
\Psi_3(\xi_1,\xi_2) &= P_2(\xi_1)He_0(\xi_2) = \frac{1}{2}(3\xi_1^2 - 1) \\
\Psi_4(\xi_1,\xi_2) &= P_1(\xi_1)He_1(\xi_2) = \xi_1\xi_2 \\
\Psi_5(\xi_1,\xi_2) &= P_0(\xi_1)He_2(\xi_2) = \xi_2^2 - 1
\end{aligned} \tag{5.19}$$

These results can be generalized for any combination of uncertain variables and input distributions. For expansion order *p* and number of variables *m*, the number of terms in a total order expansion is given as:

$$N = 1 + P = 1 + \sum_{s=1}^{P} \frac{1}{s!} \prod_{r=1}^{s-1} (m + r) = \frac{(m+p)!}{n!\, p!} \tag{5.20}$$

An alternative to full order expansion is *tensor product* expansion, where instead of fixing the maximum order of any polynomial in the expansion, the maximum order in each individual uncertain variable's expansion is fixed. In the case of the expansion explored above, the expansion polynomials would be:

$$\begin{aligned}
\Psi_0(\xi_1,\xi_2) &= P_0(\xi_1)He_0(\xi_2) = 1 \\
\Psi_1(\xi_1,\xi_2) &= P_1(\xi_1)He_0(\xi_2) = \xi_1 \\
\Psi_2(\xi_1,\xi_2) &= P_2(\xi_1)He_0(\xi_2) = \frac{1}{2}(3\xi_1^2 - 1) \\
\Psi_3(\xi_1,\xi_2) &= P_0(\xi_1)He_1(\xi_2) = \xi_2 \\
\Psi_4(\xi_1,\xi_2) &= P_1(\xi_1)He_1(\xi_2) = \xi_1\xi_2 \\
\Psi_5(\xi_1,\xi_2) &= P_2(\xi_1)He_1(\xi_2) = \xi_1(\xi_2^2 - 1) \\
\Psi_6(\xi_1,\xi_2) &= P_0(\xi_1)He_2(\xi_2) = \xi_2^2 - 1 \\
\Psi_7(\xi_1,\xi_2) &= P_1(\xi_1)He_2(\xi_2) = \xi_1(\xi_2^2 - 1) \\
\Psi_8(\xi_1,\xi_2) &= P_2(\xi_1)He_2(\xi_2) = \frac{1}{2}(3\xi_1^2 - 1)(\xi_2^2 - 1)
\end{aligned} \tag{5.21}$$

The total number of terms in a tensor product expansion is given by:

$$N = 1 + P = \prod_{i=1}^{m} (p_i + 1) \tag{5.22}$$

The advantage to tensor product as opposed to total order expansion is that it readily supports anisotropy in the expansion dimensions. While anisotropy can be captured in total order expansions, it requires careful pruning of polynomials which satisfy the total order bound but violate individual per-dimension bounds. The disadvantage is that tensor product expansion suffers from the "curse of dimensionality" in that the number of terms exponentially increases with the number of uncertain variables. This can lead to a prohibitively high number of evaluations to compute the expansion coefficients, which is the same drawback as sampling based methods described above. Ways to mitigate this drawback (such as sparse grid approaches) are discussed below.

In either case, the basis for the particular expansion employed is primarily driven by the method in which the expansion coefficients are to be evaluated. Random sampling can be used to generate $P + 1$ samples, and the resulting linear system can be solved to give the expansion coefficients. However, this method suffers from the same disadvantage as pure RS methods; the number of responses necessary for accurate evaluation can become intractable [165]. Alternatively, linear regression fits can be employed, but these often require at least $2P$ evaluations. In this work, we employ tensor-product integration to generate the response coefficients, which can be thought of as the most efficient approach.

The response is projected against each polynomial basis function using inner products, and the expansion coefficient is extracted due to the orthogonality properties of the basis polynomial. If the response is represented by the following expansion:

$$R(\boldsymbol{\xi}) = \sum_{j=0}^{P} \alpha_j \Psi_j(\boldsymbol{\xi}) \tag{5.23}$$

then it is evident that, by multiplying both sides by $\Psi_i(\boldsymbol{\xi})$ and taking the inner product with respect to the joint uncertain input density function $\varrho(\boldsymbol{\xi}) = \prod_{i=1}^{m} \varrho_i(\xi_i)$ (assuming independence of the input variables), the expansion coefficients can be extracted thusly:

$$\alpha_j = \frac{\langle R, \Psi_j \rangle}{\langle \Psi_j^2 \rangle} = \frac{1}{\langle \Psi_j^2 \rangle} \int_{\Omega} R(\boldsymbol{\xi}) \Psi_j(\boldsymbol{\xi}) \varrho(\boldsymbol{\xi}) d\boldsymbol{\xi} \tag{5.24}$$

where $\Omega = \Omega_1 \otimes ... \otimes \Omega_m$ is the support range of the density function, and is the tensor product of the supports of the individual random variables. The denominator can be computed relatively easily since it is comprised of known polynomials; the primary computational effort resides in the computation of the numerator.

We can use tensor-product rules to calculate the numerator in Equation 5.24. Quadrature formulas are a way to evaluate definite integrals numerically using function evaluations computed at pre-defined points (abscissas) $x_i,\ i = 1, \dots, N$, where N is the total number of points, and weights ω_i associated with each of those points. In the univariate case, for Gaussian quadrature, if a function $f(x)$ can be expressed as the product of a polynomial function $g(x)$ and a weighting function $\omega(x)$, then the quadrature rule can be expressed as:

$$\int_a^b f(x)dx = \int_a^b g(x)\omega(x)dx \approx \sum_{i=1}^{N} \omega_i g(x_i) \tag{5.25}$$

In this quadrature formula, the abscissas are simply the zeros of the N^{th} order orthogonal polynomial associated with the weighting function $\omega(x)$, and the weights ω_i can be calculated using a formula dependent upon integrals of the orthogonal polynomials. Each standard class of orthogonal polynomials and weighting functions has a Gaussian quadrature rule associated with it; they can also be calculated for arbitrary density functions as discussed above. The approximation formula using N points will exactly integrate polynomials up to order $2N - 1$. There are alternative quadrature approaches, such as Clenshaw-Curtis, which offer the benefit of nested rules (i.e., the points of successively higher quadrature orders include points from lower orders); the downside is that such rules may only integrate polynomials of degree up to the number of points included.

In order to describe tensor product quadrature, first, univariate quadrature rules must be defined. Following the discussion in [156], first, define an index $i \in \mathbb{N}_+, i \geq 1$. This index refers to the i^{th} uncertain variable. Then, for each value of i, there is a sequence of Gauss abscissas $\{\xi_1^i, \xi_2^i, \dots \xi_{m_i}^i\} \in \Omega_i$, where m_i defines the quadrature order (i.e., number of abscissas) in

direction i. For any continuous, not necessarily differentiable function f, there are a sequence of quadrature operators

$$\mathcal{U}^i(f)(\boldsymbol{\xi}) = \sum_{j=1}^{m_i} f\left(\xi_j^i\right)\omega_j \tag{5.26}$$

which, for Gaussian quadrature, integrates polynomials up to order $2m_i - 1$ exactly. In application to PCE, if expansion order p is chosen, then the integral in Eq. 5.26 would require evaluation of integrals up to order $2p$, and thus a quadrature rule of order at least p+1 is necessary. To generalize to multi-dimensional quadrature rules, one simply takes the tensor product of univariate quadrature rules in each dimension. Define a multi-index $\mathbf{i} = (i_1, i_2, \dots, i_n) \in \mathbb{N}_+^n$. Then, the multivariate quadrature operator $\mathcal{Q}_\mathbf{i}^n$ is defined as:

$$\begin{aligned}\mathcal{Q}_\mathbf{i}^n(f)(\boldsymbol{\xi}) &= \left(\mathcal{U}^{i_1} \otimes \dots \otimes \mathcal{U}^{i_n}\right)(f)(\boldsymbol{\xi}) \\ &= \sum_{j_1=1}^{m_1} \dots \sum_{j_n=1}^{m_n} f\left(\xi_{j_1}^{i_1}, \dots, \xi_{j_n}^{i_n}\right)\left(\omega_{j_1}^{i_1} \otimes \dots \otimes \omega_{j_n}^{i_n}\right)\end{aligned} \tag{5.27}$$

To use this quadrature rule, first, the response is evaluated at the points $\xi_{j_1}^{i_1}, \dots, \xi_{j_n}^{i_n}$ corresponding with the abscissas in the desired quadrature scheme. The number of points which are evaluated in each direction depend upon the desired order the polynomial chaos expansion. Once the response values at these points are determined, the quadrature rule can be applied to compute the integral in Eq. 5.24, producing the expansion coefficients. The key idea is that the weights and quadrature points are determined based on the density function of each uncertain variable, which allows the coefficients to be evaluated accurately.

As stated above, a disadvantage to tensor-product quadrature is the sheer number of evaluations that can be required, particularly for problems with a moderate-to-large (i.e., > 5) random inputs. As is obvious from Eq. 5.27, each quadrature evaluation requires $\prod_{j=1}^n m_{i_j}$ function values; if the same order is chosen in all directions ($m_{i_j} = m$), then m^n total function values are necessary. This is referred to as the "curse of dimensionality," and can quickly scale beyond the computationally feasible. In order to reduce the number of function evaluations required, while maintaining accuracy in the expansion coefficients, sparse grid approaches have been implemented. Sparse grids were originally proposed by Smolyak [166], and further

explored by other authors (for example, [167]) as a way to dramatically reduce the number of function evaluations required to construct the expansion. The general approach is to balance the desired accuracy of approximation with the number of points required to form it (see, for example, [168]). The way this is accomplished is by paring down the full tensor-product grid in a specific way and only using tensor products which have a relatively small number of points. Again following the discussion in [156], we describe the Smolyak isotropic formulas $\mathcal{A}(\mathrm{w}, n)$ where w is referred to as the *level* of the sparse grid and *n* is the number of directions. Let $\mathcal{U}^0 = 0$ and for $i \geq 1$ define:

$$\Delta^i = \mathcal{U}^i - \mathcal{U}^{i-1}. \tag{5.28}$$

This is a differencing operator between univariate quadrature rules with i and $i-1$ abscissas, as defined in Eq. 5.26. If we set $|\mathbf{i}| = i_1 + i_2 + \cdots + i_n$, then the isotropic Smolyak quadrature formula is:

$$\mathcal{A}(\mathrm{w}, n) = \sum_{|\mathbf{i}| \leq \mathrm{w}+n} \left(\Delta^{i_1} \otimes \ldots \otimes \Delta^{i_n}\right) \tag{5.29}$$

This can be also written in terms of the original univariate quadrature formulas:

$$\mathcal{A}(\mathrm{w}, n) = \sum_{\mathrm{w}+1 \leq |\mathbf{i}| \leq \mathrm{w}+n} (-1)^{\mathrm{w}+n-|\mathbf{i}|} \binom{n-1}{\mathrm{w}+n-|\mathbf{i}|} \left(\mathcal{U}^{i_1} \otimes \ldots \otimes \mathcal{U}^{i_n}\right) \tag{5.30}$$

The integral in Eq. 5.24 can be computed using this rule, with a greatly reduced number of function evaluations. Instead of m^n evaluations (as was the case in an isotropic tensor grid), only $m^{\log n}$ points are required, with only slightly degraded accuracy (see, for example, [169]). A comparison between a full tensor grid and sparse grids using Gauss and Clenshaw-Curtis abscissas in two dimensions is shown in Figure 5.3.1. It is obvious that the sparse grids contain significantly fewer evaluation points, even in two dimensions; the effect is even greater for multi-dimensional problems.

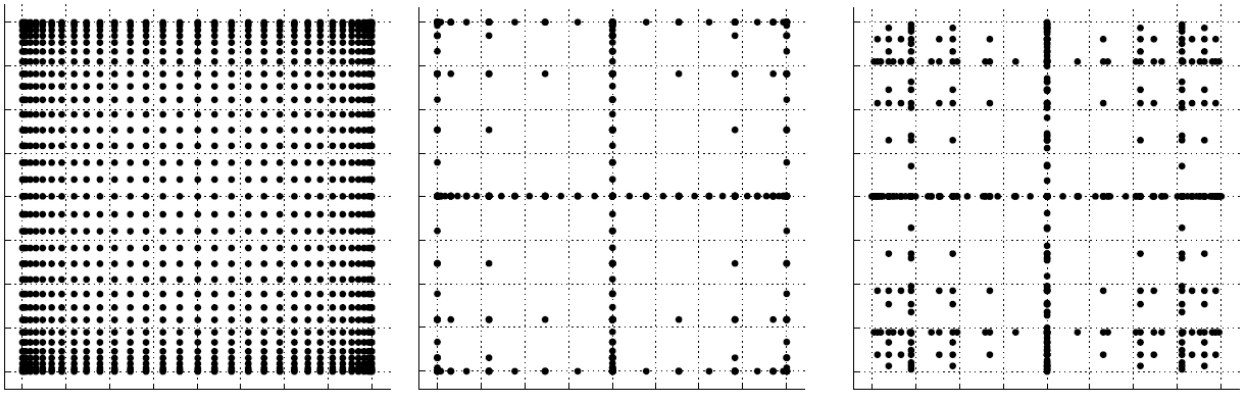


Figure 5.3.1: Tensor product and sparse grid using Clenshaw-Curtis abscissas (left, center) and sparse grid using Gaussian abscissas (right), for dimension = 2 and maximum level w = 5 (taken from [156])

Once the expansion is formed and the expansion coefficients are calculated via the quadrature methods described above, the statistical moments of interest can be easily calculated. The mean and variance, for example, are given by the following expressions:

$$\mu_R = \langle R \rangle = \int_{\Omega} R(\boldsymbol{\xi}) \varrho(\boldsymbol{\xi}) d\Omega \cong \int_{\Omega} \sum_{j=0}^{P} \alpha_j \Psi_j(\boldsymbol{\xi}) \varrho(\boldsymbol{\xi}) d\Omega = \sum_{j=0}^{P} \alpha_j \langle \Psi_j(\boldsymbol{\xi}) \rangle = \alpha_0 \tag{5.31}$$

$$\begin{aligned} \sigma_R^2 = \langle (R - \mu_R)^2 \rangle &\cong \langle \left(\sum_{j=1}^{P} \alpha_j \Psi_j(\boldsymbol{\xi}) \right)^2 \rangle = \sum_{i=1}^{P} \sum_{j=1}^{P} \alpha_i \alpha_j \langle \Psi_i(\boldsymbol{\xi}) \Psi_j(\boldsymbol{\xi}) \rangle \\ &= \sum_{j=1}^{P} \alpha_j^2 \langle \Psi_j(\boldsymbol{\xi})^2 \rangle \end{aligned} \tag{5.32}$$

Higher-order moments of the distribution can also be computed, and analytic moment-fitting methods could be employed to approximate the response pdf. Alternatively, the expansion can be very cheaply sampled via an RS or LHS technique to numerically approximate these quantities.

With this, the tensor-product based methods which are available in DAKOTA to form PCE have been described. In the next section, we will describe an alternative (but related) method to expand the response: stochastic collocation.

5.3.2 Stochastic Collocation

As presented in the previous section, PCE is based upon an infinite expansion of polynomials, which are then truncated to form an approximation of the functional behavior of the

response with respect to the uncertain inputs. Because of this, there are a number of different choices available for the estimation of the expansion coefficients. Stochastic collocation, on the other hand, forms its expansion based on structured Lagrange interpolation on the uncertain variable space, with one polynomial per expansion point, and the expansion coefficients are simply the response values at the interpolation points. The choices of abscissas, therefore, are the crux of the method. It turns out that the optimal strategy is to perform collocation using the Gauss points and weights corresponding to the orthogonal polynomials used in PCE.

As was the case with PCE, we largely follow [156] to discuss the basis of SC. To begin the discussion, recall the definition of Lagrange polynomials on a grid of points $\xi_1, \xi_2, \dots, \xi_m$ in one dimension:

$$L_j(\xi) = \prod_{\substack{i=1 \\ i \neq j}}^{m} \frac{\xi - \xi_i}{\xi_j - \xi_i} \tag{5.33}$$

This polynomial returns 1 when $\xi = \xi_j$ and 0 when $\xi = \xi_i, i \neq j$. It is obvious that it is a polynomial of degree $m - 1$. We can form an $m - 1^{\text{th}}$ degree interpolant of a function $R(\xi)$ over m points as:

$$R(\xi) \cong \sum_{j=1}^{m} r(\xi_j) L_j(\xi) \tag{5.34}$$

This expansion smoothly interpolates between the responses $r(\xi_j)$ evaluated at points ξ_j. To form a multidimensional interpolant, one simply takes the tensor product of the one-dimensional Lagrange interpolants:

$$R(\boldsymbol{\xi}) \cong \sum_{j_1=1}^{m_{i_1}} \dots \sum_{j_n=1}^{m_{i_n}} r\left(\xi_{j_1}^{i_1}, \dots, \xi_{j_n}^{i_n}\right)\left(L_{j_1}^{i_1} \otimes \dots \otimes L_{j_n}^{i_n}\right) = \sum_{j=1}^{N_p} r_j(\boldsymbol{\xi}) L_j(\boldsymbol{\xi}) \tag{5.35}$$

where $\xi_{j_l}^{i_k}$ is the j_l^{th} point in the k^{th} direction and N_p is the total number of expansion points. Equation 5.35 is the Stochastic Collocation expansion of the response.

As mentioned above, the key to maximizing the SC approach is to choose the interpolation points to be the Gaussian abscissas corresponding to the orthogonal polynomials chosen for each direction (which, in turn, depend upon the weighting function (pdf) describing

the variability of the input in that dimension). Both the tensor-product and the sparse grid approaches described above can be employed for this purpose. Noting that the Gauss points of orthogonal polynomials are its roots, one can factor such polynomials as:

$$\psi_j = c_j \prod_{k=1}^{p} (\xi - \xi_k) \tag{5.36}$$

where ξ_k are the roots and ψ_j is a p^{th}-order orthogonal polynomial. The parallels between this factorization and the form of the one-dimensional Lagrange interpolant are obvious. However, it should be noted that in order to form a p^{th}-order Lagrange interpolant using Gauss points, it is necessary to use the roots of the orthogonal polynomial of order $p+1$ and then exclude the Gauss point at which the response is being interpolated. For example, let ξ be a uniform random variable on the interval $[-1,1]$. Legendre polynomials correspond with this distribution. If we wished to form a second-order expansion, we would need to use the Gauss points corresponding with the third-order Legendre polynomial, and then exclude the point at which we were interpolating. In this specific case, the third-order Gauss points are $\left\{-\sqrt{3/5}, 0, +\sqrt{3/5}\right\}$, and the interpolant at $\xi_2 = 0$ is:

$$L_2(\xi) = -\frac{5}{3}\left(\xi + \sqrt{3/5}\right)\left(\xi - \sqrt{3/5}\right) \tag{5.37}$$

Therefore, to form an expansion of order p, one uses the polynomial roots of order $p+1$. As discussed above, integration to determine expansion coefficients in PCE requires a similar approach; for expansion orders up to p, Gaussian quadrature with at least $p+1$ points is required. This means that the collocation points for PCE and SC of equivalent order are the same. However, there is a slight difference in the expansion formulation in that, for PCE, the expansion involves all p roots up to order p, whereas for SC the expansion involves a p root subset of order $p+1$. Therefore, the actual polynomial expansions using the two methods are related, but not equivalent.

In summary, the stochastic collocation method is based upon forming an expansion based on Lagrange interpolants, with the expansion coefficients being the response values at the collocation points. The optimal choices for the collocation points are the Gauss points of the orthogonal polynomials with respect to each uncertain variable's distribution. Once the

expansion is formed and the responses calculated at each point, the mean and variance of the response R can be computed:

$$\mu_R = \langle R \rangle = \int_\Omega R(\boldsymbol{\xi})\varrho(\boldsymbol{\xi})d\Omega \cong \int_\Omega \sum_{j=1}^{N_p} r_j \, \mathrm{L}_j(\boldsymbol{\xi})\varrho(\boldsymbol{\xi})d\Omega = \sum_{j=1}^{N_p} r_j \, \langle \mathrm{L}_j(\boldsymbol{\xi}) \rangle = \sum_{j=1}^{N_p} r_j \, \omega_j \tag{5.38}$$

$$\sigma_R^2 = \langle (R - \mu_R)^2 \rangle = \langle R^2 \rangle - \mu_R^2 \cong \sum_{i=1}^{N_p}\sum_{j=1}^{N_p} r_i r_j \langle L_i(\boldsymbol{\xi}) L_j(\boldsymbol{\xi}) \rangle - \mu_R^2 = \sum_{j=1}^{N_p} r_j^2 \omega_j - \mu_R^2 \tag{5.39}$$

Therefore, the mean value is simply a weighted average of the response over all points in the domain. The above integrals simplify considerably with the application of the appropriate quadrature rule. As with a PCE, additional moments can be computed, or the SC expansion can be cheaply employed in a sampling based method to estimate the response distribution.

With this, the four basic approaches to uncertainty analysis (random sampling, Latin Hypercube sampling, polynomial chaos expansion, and stochastic collocation) employed in this work have been introduced. Before applying these methods to the coupled depletion problem, we provide a description of sensitivity analysis methods used here.

5.4 Sensitivity Analysis

Sensitivity analysis is the process of quantifying the relative importance of input variables to the system response. This usually means determining the fraction of variance in the response due to the variance in a particular input variable. There are a number of different approaches to this problem, varying in mathematical complexity and required computational effort, depending upon the type of information the analyst is attempting to generate. The most intuitive approach to quantifying the sensitivity S_i of response Y to input variable x_i is to use a differential [170]:

$$S_i \equiv \frac{\partial Y}{\partial x_i} \tag{5.40}$$

The analysis presented in Chapter 4 was rudimentary sensitivity analysis of this type; the input variables were ranged over their 95% intervals (i.e., $\pm 2\sigma$), and the resulting change in the output response was observed. The derivative can be thought of as evaluated about the nominal (i.e.,

zero-bias) point. The differential approach can be effective, depending on the required implementation effort; however, in the presence of uncertain input variables and depending on the degree of non-linearity of the response, it may be inappropriate or insufficient to truly characterize the sensitivity. In order to account for the distributional effect in an uncertainty analysis context, it is common to non-dimensionalize the sensitivity coefficient S_i with the standard deviations of Y and x_i:

$$S_i^{\sigma} \equiv \frac{\sigma_i}{\sigma_Y}\frac{\partial Y}{\partial x_i} \tag{5.41}$$

While it does include some influence of the uncertain nature of input and output, this definition still has limited applicability; due to the local nature of the derivative, it is only valid at or near the point at which S_i is evaluated. If Y is strongly non-linear, it may require many evaluations over the span of the input space to properly characterize the sensitivity. This is what is meant by *local* sensitivity analysis.

5.4.1 Pearson and Spearman Correlation Coeffients

Due to the limited nature of local approaches, it is common to pursue *global* sensitivity approaches, in which the sensitivity of a response to a particular input variable is integrated over the range of the input and distilled into a single index. For example, the Pearson correlation coefficient is defined as ([171]):

$$R_i \equiv \frac{\mathrm{cov}(x_i, Y)}{\sigma_i \sigma_Y} = \frac{\sigma_i}{\sigma_Y}\hat{b} \tag{5.42}$$

In this equation, $\hat{b}$ is the slope calculated via linear regression of Y against x_i. This definition follows naturally from a Monte Carlo approach to computing the distribution of Y. In that sense, the effect of x_i over its entire range and distribution is accounted for. It can be shown that the square of the Pearson coefficient R_i^2 is proportional to the variance in Y explained by the variance in x_i assuming that Y is a linear function of x_i [172]. We computed R^2 for the uncertainty parameters in Chapter 4.

The trouble with the Pearson coefficient is the linearity assumption; it only provides adequate explanation of the constituent variances' contribution to the response variance for linear, additive functions. A generalization of the Pearson coefficient to nonlinear, monotonic

functions is provided by the Spearman coefficient [171]. Instead of performing the correlation calculation on the raw data itself, the calculations are performed on the ranks of the data. That is, the realizations of x_i and Y are arranged in ascending order and assigned values from 1 to N. The Spearman coefficient for ranks r_{x_i} and r_Y is thus given by:

$$\rho_i \equiv \frac{\mathrm{cov}(r_{x_i}, r_Y)}{\sigma_{r_{x_i}} \sigma_{r_Y}} \tag{5.43}$$

As with the Pearson coefficient, ρ_i^2 gives an estimate of the i^{th} variable's contribution to the variance of Y. It often performs considerably better than the Pearson coefficient because of the removal of the linear assumption; however, it still may not work particularly well in the case of non-monotonic or non-additive functions. Additionally, both the Pearson and the Spearman coefficient only consider the effect of one input variable at a time, and may not include higher-order input interaction effects which may be important. In order to quantify these possible interactions and give a more robust treatment of sensitivity for general responses, Variance-Based Decomposition (VBD), first proposed by Sobol' [173] is available.

5.4.2 Variance-Based Decomposition

The main idea of VBD is that the total variance of an output Y can be decomposed into a finite sum involving the variances of both the input variables *and* terms that describe the interaction of the inputs. Following the discussion presented in [170], we consider a function f which is square-integrable over the k^{th}-dimensional unit hypercube $\Omega^k = \{X | 0 \leq x_i \leq 1, i = 1, .., k\}$. This function can be expanded into terms of increasing dimension:

$$f = f_0 + \sum_{i=1}^{k} f_i + \sum_{i=1}^{k} \sum_{j=i+1}^{k} f_{ij} + \cdots + f_{12\ldots k} \tag{5.44}$$

where f_0 is a constant, $f_i = f_i(X_i)$, $f_{ij} = f_{ij}(X_i, X_j)$, etc. Sobol' showed that, if each non-constant term in this expansion has a zero mean (i.e., $\int f_i(x_i) d\mu_i = 0$, where $d\mu_i = p_i(x_i) dx_i$, and $p_i(x_i)$ is the pdf associated with x_i), then the terms in the expansion are pair-wise orthogonal (i.e., $\iint f_i(x_i) f_j(x_j) d\mu_i d\mu_j = 0, i \neq j$). Therefore, if the response $Y = f(X_1, \ldots, X_k)$, the terms can be calculated using conditional expectations of Y:

$$\begin{aligned} f_0 &= E[Y] \\ f_i &= E[Y|X_i] - E[Y] \\ f_{ij} &= E\left[Y\middle|X_i, X_j\right] - E[Y|X_i] - E\left[Y\middle|X_j\right] - E[Y] \end{aligned} \tag{5.45}$$

and so on. In this context, the conditional expectations are defined as:

$$E[Y|X_i] = \int_{\Omega^{k-1}} f(x_1, \dots, x_{i-1}, x_{i+1}, \dots x_k) d\mu_1 \dots d\mu_{i-1} d\mu_{i+1} \dots d\mu_k \tag{5.46}$$

From this definition, it is clear that it is the expectation value where all other dependencies except for the conditional variable have been integrated out. What is of interest, of course, is the relationship between the variance of Y and the variance of each of these individual terms in the expansion. If we apply the variance operator (Equation 5.10) to both sides of Eq. 5.44 and take advantage of the term-by-term pairwise orthogonality, one arrives at the following decomposition (see [170] or [174]):

$$V(Y) = \sum_{i=1}^{k} V_i + \sum_{i=1}^{k} \sum_{j=i+1}^{k} V_{ij} + \cdots + V_{12\dots k} \tag{5.47}$$

where $V_i = V(f_i) = V(E[Y|X_i])$, $V_{ij} = V\left(f_{ij}\right) = V\left(E\left[Y\middle|X_i, X_j\right]\right) - V(E[Y|X_i]) - V\left(E\left[Y\middle|X_j\right]\right)$, etc. These terms $V_{i_1 i_2 \dots i_s}$ are referred to as *partial variances*. The expression in the above equation is referred to as the analysis of variance (ANOVA) decomposition [174]; the total variance in the response is a summation of the variances due to each effect alone, the pairwise interaction of each term, the interaction of each triplet of terms, etc. If the above equation is normalized by $V(Y)$, then we arrive at the following:

$$\sum_{i=1}^{k} S_i + \sum_{i=1}^{k} \sum_{j=i+1}^{k} S_{ij} + \cdots + S_{12\dots k} = 1 \tag{5.48}$$

where we have defined sensitivity indices:

$$S_{i_1 i_2 \dots i_m} \equiv \frac{V_{i_1 i_2 \dots i_m}}{V(Y)} \tag{5.49}$$

The advantage of this decomposition is that it allows us to determine what fraction of variance in Y is due to each individual contributor. The first order terms are referred to as the *main effect* indices:

$$S_i = \frac{V(E[Y|X_i])}{V(Y)} \tag{5.50}$$

and characterize each variable's direct contribution to the variance. In the case of a linear model, $S_i = R_i^2$, and all the variance is explained by the main effect terms. However, in many physics simulations, the main effects are not sufficient to capture the complete picture of what contributes to response variability. As is clear from the decomposition presented in Eq. 5.48, it is possible to use VBD to isolate and estimate the effects of each input variable's interaction with other inputs, therefore giving a much more comprehensive view of which effects are important to variance. An excellent application of Sobol' decomposition to fuel thermo-mechanical calculations using FRAPCON was performed by Ikonen and Tulkki in [171]. They investigated a number of phenomena important for fuel performance, and showed the benefit to using VBD over standard measures to estimate contributions to variance. For example, using only Spearman or first-order sensitivity coefficients, less than two-thirds of the variance in gap conductance at a specified burnup was explained. By including second-order interaction terms, the fraction of variance described jumped to 97%; nearly all the variance was explained, and the missing 3% could be found by investigating the higher-order terms. The improvement over linear analysis is obvious.

The main drawback to Sobol' decomposition is that the integrals embedded within it can be quite expensive to calculate, depending on the uncertainty method applied (this will be made clear below). The number of terms in the expansion alone, given by $2^k - 1$, can quickly become overwhelming (for example, an expansion with 10 random inputs would have 1023 terms). In order to provide a means by which the potential interaction effects of each variable can be evaluated without explicitly calculating every term in the expansion, the *total effect* index is available.

The total effect T_i for variable X_i is defined as the summation of all sensitivity indices which involve that variable [171]:

$$T_i = \sum_{j=1}^{k} \delta_{ij} S_j + \sum_{j=1}^{k} \sum_{l=j+1}^{k} (\delta_{ij}+\delta_{il}) S_{jl} + \cdots + S_{12\ldots i\ldots k} \tag{5.51}$$

For example, if there are three uncertain inputs, then the total effect index for X_1 is $T_1 = S_1 + S_{12} + S_{13} + S_{123}$. The difference $T_i - S_i$ can be interpreted as a measure of the "departure from additivity" of X_i; the higher this value relative to the total effect, the more important the interaction contributions to variance are for that variable. It is obvious that if a full Sobol' decomposition of the variance is available, the total effect is trivial to compute. As it stands, however, this is not always a trivial calculation, particularly if a sampling method is used to evaluate the response statistics. As we will see in following discussion, the total effect can be computed using sampling without the need to compute every term in the Sobol' expansion. Therefore, the importance of input interaction can be somewhat (and oftentimes sufficiently) quantified relatively cheaply. On the other hand, as we will also see, one of the chief benefits of using a stochastic expansion method is that the Sobol' decomposition can be extracted from the expansion coefficients (in the case of PCE) or the products of the quadrature weights and response values at the abscissas (in the case of SC) without a great deal of additional calculation.

5.4.3 VBD using Sampling Methods

We now discuss the means by which Sobol' indices can be computed using random sampling. To begin, we describe the method developed by Saltelli using a sampling-based approach and implemented in DAKOTA ([7], [170]). The goal of these calculations is to produce the main effect and total effect indices with a relatively few number of calculation.

First, two $N \times k$ matrices A and B are generated by sampling the random input distributions, where N is the number of sampled outputs and k is the number of input variables:

$$A = \begin{pmatrix} x_1^{(1)} & x_2^{(1)} & \ldots & x_i^{(1)} & \ldots & x_k^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \ldots & x_i^{(2)} & \ldots & x_k^{(2)} \\ \ldots & \ldots & \ldots & \ldots & & \\ x_1^{(N-1)} & x_2^{(N-1)} & \ldots & x_i^{(N-1)} & \ldots & x_k^{(N-1)} \\ x_1^{(N)} & x_2^{(N)} & \ldots & x_i^{(N)} & \ldots & x_k^{(N)} \end{pmatrix}$$

$$B = \begin{pmatrix} x_{k+1}^{(1)} & x_{k+2}^{(1)} & \dots & x_{k+i}^{(1)} & \dots & x_{2k}^{(1)} \\ x_{k+1}^{(2)} & x_{k+2}^{(2)} & \dots & x_{k+i}^{(2)} & \dots & x_{2k}^{(2)} \\ \dots & \dots & \dots & \dots & & \\ x_{k+1}^{(N-1)} & x_{k+2}^{(N-1)} & \dots & x_{k+i}^{(N-1)} & \dots & x_{2k}^{(N-1)} \\ x_{k+1}^{(N)} & x_{k+2}^{(N)} & \dots & x_{k+i}^{(N)} & \dots & x_{2k}^{(N)} \end{pmatrix}$$

Then, k matrices $C_i, i = 1, \dots, k$ are formed by replacing the i^{th} column of B with the i^{th} column of A:

$$C_i = \begin{pmatrix} x_{k+1}^{(1)} & x_{k+2}^{(1)} & \dots & x_{i}^{(1)} & \dots & x_{2k}^{(1)} \\ x_{k+1}^{(2)} & x_{k+2}^{(2)} & \dots & x_{i}^{(2)} & \dots & x_{2k}^{(2)} \\ \dots & \dots & \dots & \dots & & \\ x_{k+1}^{(N-1)} & x_{k+2}^{(N-1)} & \dots & x_{i}^{(N-1)} & \dots & x_{2k}^{(N-1)} \\ x_{k+1}^{(N)} & x_{k+2}^{(N)} & \dots & x_{i}^{(N)} & \dots & x_{2k}^{(N)} \end{pmatrix}$$

The model output using these collections of input variables is then generated and collected in vectors y_A, y_B, and y_{C_i}:

$$y_A = f(A) \quad y_B = f(B) \quad y_{C_i} = f(C_i)$$

where the j^{th} element of $y_X$is the response obtained using the j^{th} row of matrix X. This requires a total of $N(k+2)$ model evaluations; significantly more expensive than simple sampling-based uncertainty analysis, but still not as expensive as alternative methods to estimate the Sobol' indices, which can scale as N^2 [170]. Saltelli showed [175] that the main effect sensitivity index can be estimated as:

$$S_i = \frac{V(E[Y|X_i])}{V(Y)} = \frac{y_A \cdot y_{C_i} - f_0^2}{y_A \cdot y_A - f_0^2} = \frac{(1/N)\sum_{j=1}^{N} y_A^{(j)} y_{C_i}^{(j)} - (1/N)\left(\sum_{j=1}^{N} y_A^{(j)}\right)^2}{(1/N)\sum_{j=1}^{N}\left(y_A^{(j)}\right)^2 - (1/N)\left(\sum_{j=1}^{N} y_A^{(j)}\right)^2} \qquad (5.52)$$

In this formula, y_{C_i} is generated by fixing the value of X_i and resampling all the other variables. If X_i is non-influential, then high and low values in the vectors y_A and y_{C_i} will be randomly associated, and we can expect that the difference in the numerator will not be large. If, on the other hand, X_i is important, then large values of y_A will be associated with large values of y_{C_i}, and the first term in the numerator will be accordingly amplified.

To describe the method used to compute the total effect index, we first recall that the total variance in an output variable can be expressed as the sum of the variance of the mean conditioned on a particular input X_i and the expectation of the variance conditioned on that variable (see [170]):

$$V(Y) = V(E[Y|X_i]) + E[V(Y|X_i)] \tag{5.53}$$

The first term in this sum is the main effect, and the second term is referred to as the residual. Alternatively, the variance can be expressed as the sum of the variance of the expectation of Y conditioned on *all variables except* X_i and the expectation of the variance of Y conditioned on the same:

$$V(Y) = V(E[Y|\boldsymbol{X}_{-i}]) + E[V(Y|\boldsymbol{X}_{-i})] \tag{5.54}$$

The quantity $V(Y) - V(E[Y|\boldsymbol{X}_{-i}]) = E[V(Y|\boldsymbol{X}_{-i})]$ is "the remaining variance of Y that would be left, on average, if we could determine the true values for $\boldsymbol{X}_{-i}$. The average is calculated over all possible combinations of $\boldsymbol{X}_{-i}$, since $\boldsymbol{X}_{-i}$ are uncertain factors and their 'true values' are unknown." [170]. The total effect index for random variable X_i thus found by dividing by $V(Y)$:

$$T_i = \frac{E[V(Y|\boldsymbol{X}_{-i})]}{V(Y)} = 1 - \frac{V(E[Y|\boldsymbol{X}_{-i}])}{V(Y)} \tag{5.55}$$

Similarly to the evaluation of the main effect index, Saltelli showed that $V(E[Y|\boldsymbol{X}_{-i}])$ can be estimated using dot products of y_B, and y_{C_i}:

$$\begin{aligned} T_i &= 1 - \frac{V(E[Y|\boldsymbol{X}_{-i}])}{V(Y)} \\ &= 1 - \frac{y_B \cdot y_{C_i} - f_0^2}{y_A \cdot y_A - f_0^2} \\ &= 1 - \frac{(1/N)\sum_{j=1}^{N} y_B^{(j)} y_{C_i}^{(j)} - (1/N)\left(\sum_{j=1}^{N} y_A^{(j)}\right)^2}{(1/N)\sum_{j=1}^{N}\left(y_A^{(j)}\right)^2 - (1/N)\left(\sum_{j=1}^{N} y_A^{(j)}\right)^2} \end{aligned} \tag{5.56}$$

In this formula, we obtain $y_B \cdot y_{C_i}$ by resampling X_i and keeping all the other values of the uncertain variables fixed. Similar to Equation 5.52, we see that if the other variables are

unimportant, then high and low values of y_B and y_{C_i} will be randomly associated; otherwise, high values of y_B will be preferentially multiplied by high values of y_{C_i}, amplifying the dot product in the numerator. A higher value of this dot product indicates that the other variables (i.e., all except X_i) are more important, which would reduce the value of X_i's total effect.

The methods applied above are ways in which the main and total effect sensitivity indices can be computed via sampling-based methods. As described above, these calculations require $N(k+2)$ model evaluations; as is the case with sampling methods in general, this may become quickly intractable if the cost of each evaluation is excessive. Also, the integrals evaluated via dot product in the above formulas provide statistical estimates of the true quantities; Saltelli et al recommend N to be at least a hundred, and preferably many hundreds or thousands, to ensure that the indices are estimated accurately [170]. Unfortunately, this is not often a tenable strategy, given the costs of executing a model; for example, the pin cell problem described in the Chapter 4 costs approximately one hour per execution; with five random input variables and $N = 1000$, this would require about 7000 hours of execution time. Even on multi-core processor computers, this is an arduous task. We next explore how the stochastic expansion methods provide access to the partial variances without the need for excessive additional calculation.

5.4.4 VBD with Stochastic Expansions

Fortunately, the stochastic expansion methods described above provide a means by which all the terms in the Sobol' variance decomposition can be evaluated analytically, simply by using the expansion coefficients. This means that once the expansion is constructed, it is virtually free to generate sensitivity data and explicitly compute all the interaction terms. In the following discussion, the way in which this is achieved is explained.

First, we will discuss Sobol' index recovery using PCE. Much of the following discussion is taken from Sudret [165], who was the first to relate PCE with Sobol' decomposition. As we have described above, the PCE of a function $Y = f(\boldsymbol{\xi})$ can be expressed as:

$$Y = \sum_{j=0}^{P-1} \alpha_j \Psi_j(\boldsymbol{\xi}) \tag{5.55}$$

where P is the total number of terms in the expansion, α_j are the expansion coefficients, and $\Psi_j(\boldsymbol{\xi})$ are the expansion polynomials. We shall see that, if these terms are simply re-ordered and grouped appropriately, the form of the Sobol' decomposition (Equation 5.44) can be recovered.

To show this, we first represent each expansion polynomial using a tuple $\boldsymbol{\theta} = (\theta_1, \dots, \theta_k)$**,** depending on which uncertain variables are contained within a given polynomial:

$$\Psi_j \equiv \Psi_{\boldsymbol{\theta}} : \Psi_j(\boldsymbol{\xi}) = \prod_{i=1}^{k} P_{\theta_i}(\xi_i) \tag{5.56}$$

This is simply expressing each multivariate expansion polynomial $\Psi_j(\boldsymbol{\xi})$ as a product of the univariate orthogonal polynomials $P_{\theta_i}(\xi_i)$ of which it consists (see the examples given in Equations 5.19 and 5.21). $P_{\theta_i}(\xi_i)$ is interpreted as the orthogonal polynomial of order θ corresponding with dimension i. Next, denote by $\mathcal{I}_{i_1,\dots,i_s}$ the set of $\boldsymbol{\theta}$ tuples such that only the indices $(i_1, \dots, i_s)$ are non-zero (note that P_{0_n} corresponds with a zero$^{\text{th}}$-order polynomial (i.e., a constant) in direction n, which is therefore not a function of any variable):

$$\mathcal{I}_{i_1,\dots,i_s} = \left\{ \boldsymbol{\theta} : \begin{array}{ll} \theta_n > 0 & \forall n = 1, \dots, k, \quad n \in (i_1, \dots, i_s) \\ \theta_n = 0 & \forall n = 1, \dots, k, \quad n \notin (i_1, \dots, i_s) \end{array} \right\} \tag{5.57}$$

We can use this definition to re-order the terms of the PCE in Eq. 5.55 in accordance with the dependent variables:

$$\begin{aligned} Y = \alpha_0 &+ \sum_{i=1}^{k} \sum_{\boldsymbol{\theta} \in \mathcal{I}_i} \alpha_{\boldsymbol{\theta}} \Psi_{\boldsymbol{\theta}}(\xi_i) + \sum_{i_1=1}^{k} \sum_{i_2=i_1+1}^{k} \sum_{\boldsymbol{\theta} \in \mathcal{I}_{i_1,i_2}} \alpha_{\boldsymbol{\theta}} \Psi_{\boldsymbol{\theta}}(\xi_{i_1}, \xi_{i_2}) + \cdots \\ &+ \sum_{i_1=1}^{k} \dots \sum_{i_s=i_{s-1}+1}^{k} \sum_{\boldsymbol{\theta} \in \mathcal{I}_{i_1,\dots,i_s}} \alpha_{\boldsymbol{\theta}} \Psi_{\boldsymbol{\theta}}(\xi_{i_1}, \dots, \xi_{i_s}) + \cdots \\ &+ \sum_{\boldsymbol{\theta} \in \mathcal{I}_{1,2,\dots,k}} \alpha_{\boldsymbol{\theta}} \Psi_{\boldsymbol{\theta}}(\xi_1, \dots, \xi_k) \end{aligned} \tag{5.58}$$

As shown above, the constant term α_0 is the mean of the response, the second term is a summation over all polynomials that only depend on one uncertain variable, the second term includes all terms that involve only two variables, etc, until the last term, which contains the polynomials including all variables. If this expansion is compared with the definition of the Sobol' expansion, Equation 5.44, and it is recognized that due to the orthogonality of the

constituent polynomials, this expansion shares the orthogonality properties required to make the Sobol' expansion unique, we can conclude that this expression of the polynomial chaos expansion is *identical* to the Sobol' expansion. This is particularly clear when the following connection is made between the terms in the Sobol' expansion $f_{i_1 i_2 \dots i_s}$ and the terms in the above re-ordered PCE expansion:

$$f_{i_1 i_2 \dots i_s}(\xi_{i_1}, \xi_{i_2}, \dots, \xi_{i_s}) = \sum_{\boldsymbol{\theta} \in \mathcal{J}_{i_1,\dots,i_s}} \alpha_{\boldsymbol{\theta}} \Psi_{\boldsymbol{\theta}}(\xi_{i_1}, \dots, \xi_{i_s}) \tag{5.59}$$

With the terms in the Sobol' expansion thus available, it is trivial to compute the sensitivity indices for any order of interaction thusly:

$$S_{i_1 i_2 \dots i_s} = \frac{\sum_{\boldsymbol{\theta} \in \mathcal{J}_{i_1,\dots,i_s}} \alpha_{\boldsymbol{\theta}}^2 E\left[\Psi_{\boldsymbol{\theta}}^2(\xi_{i_1}, \dots, \xi_{i_s})\right]}{\sigma_Y^2} \tag{5.60}$$

Total effect indices can then be computed using the definition provided in Eq. 5.51. A further benefit to using the PCE approach is that we can choose to investigate the sensitivity indices of particular pieces of the above summation (i.e., isolating the linear, quadratic, cubic, etc. terms) to determine the contribution of a particular polynomial degree to the total variance.

With the knowledge that PCE can be used to analytically evaluate sensitivity indices, it is natural to question whether an equivalent calculation could be done using the SC expansion. As it happens, Tang et al investigated the means by which sensitivity indices can be computed using SC [176]. Instead of a reordering of terms as with PCE, the SC calculation involves substituting the expansion into the definitions of partial variance and integrating them using the Gauss quadrature rules. We summarize their approach below; additional details can be found in their paper.

Recall that the SC expansion of response $Y = f(\xi_1, \dots, \xi_k) = f(\boldsymbol{\xi})$ can be expressed as:

$$f(\xi_1, \dots, \xi_k) \approx \sum_{j_1=1}^{m_{i_1}} \dots \sum_{j_k=1}^{m_{i_k}} f\left(\xi_{j_1}^{i_1}, \dots, \xi_{j_k}^{i_k}\right)\left(l_{j_1}^{i_1} \otimes \dots \otimes l_{j_k}^{i_k}\right) \tag{5.61}$$

i.e., as a tensor product of univariate Lagrange interpolants l_j^i with expansion coefficients being the function values evaluated at particular points of the input space. Let us denote by u a multi-

index such that $u \in \mathcal{U} = \{1,2,\dots,k\}$. Also, define $\xi_u = \{\xi_i | \forall i \in u\}$ as the set of random variables whose index is contained in u. Also, denote by u' the complement of u, i.e., the set of indices which are not contained in u. For example, for a three-dimensional input space, if $u = \{1\}$, then $u' = \{2,3\}$. We can express the Sobol' basis functions f_u in the following way:

$$f_u = \begin{cases} \int f(\boldsymbol{\xi}) d\mu_{u'} - \sum_{w \subset u} \int f_w(\xi_w) d\mu_u & u \neq \emptyset \\ \int f(\boldsymbol{\xi}) d\mu & u = \emptyset \end{cases} \tag{5.62}$$

where $d\mu_u = \prod_{i \in u} p(\xi_i) d\xi_i$. According to [177] we can express the partial variances V_u in the following way:

$$\begin{aligned} V_u &= \int \left(\int f(\boldsymbol{\xi}) d\mu_{u'} \right)^2 d\mu_u - \sum_{w \subset u} \int \left(f_w(\xi_w) \right)^2 d\mu_u \\ &= \int \left(\int f(\boldsymbol{\xi}) d\mu_{u'} \right)^2 d\mu_u - \sum_{w \subset u} V_w \end{aligned} \tag{5.63}$$

The basic approach to calculating the partial variance will be to compute the Sobol' basis functions recursively (i.e., using Eq. 5.63) and substituting both the SC function expansion and the relevant basis functions into Eq. 5.62. The integrals are evaluated using the quadrature rules defined to evaluate the locations and weights of the SC expansion. We follow the process in [176] to explain how this is done. First, substitute the SC expansion into the inner integral in Eq. 5.63:

$$\int f(\boldsymbol{\xi}) d\mu_{u'} = \int \sum_{j_1=1}^{m_{i_1}} \dots \sum_{j_k=1}^{m_{i_k}} f\left(\xi_{j_1}^{i_1}, \dots, \xi_{j_k}^{i_k} \right) \left(l_{j_1}^{i_1} \otimes \dots \otimes l_{j_k}^{i_k} \right) d\mu_{u'} \tag{5.64}$$

Next, we want to separate the interpolants that are dependent on ξ_u from the interpolants that are dependent on $\xi_{u'}$ and use the Gauss rule to integrate. Suppose there are d elements in a given u:

$$\int f(\boldsymbol{\xi}) d\mu_{u'} = \sum_{j_1=1}^{m_{i_1}} \dots \sum_{j_k=1}^{m_{i_k}} f\left(\xi_{j_1}^{i_1}, \dots, \xi_{j_k}^{i_k} \right) \int \left(\mathbf{l}^{u'} \otimes \mathbf{l}^{u} \right) d\mu_{u'} \tag{5.65}$$

$$= \sum_{j_1=1}^{m_{i_1}} \dots \sum_{j_k=1}^{m_{i_k}} f\left(\xi_{j_1}^{i_1}, \dots, \xi_{j_k}^{i_k}\right)\left(\mathbf{w}^{u'} \otimes \mathbf{l}^{u}\right)$$

where, for notational convenience, we have defined

$$u = \{u_1, \dots, u_d\}$$
$$\mathbf{w}^{u} = w_{j_{u_1}}^{i_{u_1}} \otimes \dots \otimes w_{j_{u_d}}^{i_{u_d}}$$
$$\mathbf{l}^{u} = l_{j_{u_1}}^{i_{u_1}} \otimes \dots \otimes l_{j_{u_d}}^{i_{u_d}}$$

This equation can be simplified by combining the tensor product of the weights $\mathbf{w}^{u'}$ with the response values $f\left(\xi_{j_1}^{i_1}, \dots, \xi_{j_k}^{i_k}\right)$ to form a new set of weighted coefficients $h\left(\xi_{j_{u_1}}^{i_{u_1}}, \dots, \xi_{j_{u_d}}^{i_{u_d}}\right)$:

$$\begin{aligned}
&\sum_{j_1=1}^{m_{i_1}} \dots \sum_{j_k=1}^{m_{i_k}} f\left(\xi_{j_1}^{i_1}, \dots, \xi_{j_k}^{i_k}\right)\left(\mathbf{w}^{u'} \otimes \mathbf{l}^{u}\right) \\
&\qquad = \sum_{j_{u_1}=1}^{m_{u_1}} \dots \sum_{j_{u_d}=1}^{m_{u_d}} \left(\sum_{j_{u'_1}=1}^{m_{u'_1}} \dots \sum_{j_{u'_{k-d}}=1}^{m_{u'_{k-d}}} f\left(\xi_{j_1}^{i_1}, \dots, \xi_{j_k}^{i_k}\right)\mathbf{w}^{u'} \right)(\mathbf{l}^{u}) \qquad (5.66) \\
&\qquad = \sum_{j_{u_1}=1}^{m_{u_1}} \dots \sum_{j_{u_d}=1}^{m_{u_d}} h\left(\xi_{j_{u_1}}^{i_{u_1}}, \dots, \xi_{j_{u_d}}^{i_{u_d}}\right)(\mathbf{l}^{u})
\end{aligned}$$

We are now prepared to evaluate the outer integral present in Eq. 5.63. In order to handle the square in the integral, recall that by definition Lagrange interpolants have the following property:

$$l_r^q\left(\xi_p^q\right) \cdot l_s^q\left(\xi_p^q\right) = \begin{cases} 1 & p = s = r \\ 0 & o/w \end{cases}$$

By virtue of this property, and by applying the quadrature rule to evaluate the integral, we have:

$$\int \left(\int f(\boldsymbol{\xi}) d\mu_{u'} \right)^2 d\mu_u = \int \left(\sum_{j_{u_1}=1}^{m_{u_1}} \dots \sum_{j_{u_d}=1}^{m_{u_d}} h\left(\xi_{j_{u_1}}^{i_{u_1}}, \dots, \xi_{j_{u_d}}^{i_{u_d}}\right)(\mathbf{l}^{u}) \right)^2 d\mu_u \qquad (5.67)$$

$$= \sum_{j_{u_1}=1}^{m_{u_1}} \cdots \sum_{j_{u_d}=1}^{m_{u_d}} h^2\left(\xi_{j_{u_1}}^{i_{u_1}}, \ldots, \xi_{j_{u_d}}^{i_{u_d}}\right) \otimes \mathbf{w}^u$$

Therefore, the fractional variance V_u is found to be:

$$V_u = \sum_{j_{u_1}=1}^{m_{u_1}} \cdots \sum_{j_{u_d}=1}^{m_{u_d}} h^2\left(\xi_{j_{u_1}}^{i_{u_1}}, \ldots, \xi_{j_{u_d}}^{i_{u_d}}\right) \otimes \mathbf{w}^u - \sum_{w \subset u} V_w \tag{5.67}$$

Once the fractional variance is known, the sensitivity index for that particular term can be found simply by dividing by the total variance. A recursive calculation can be used to compute these indices, starting with the main effects and progressing through two-way, three-way, up to however many potential interaction terms exist. The total effect indices can also be computed by simply summing up the interaction terms appropriately. This concludes our discussion of how VBD follows easily and inexpensively from stochastic expansions of the response.

5.5 Conclusions

In this Chapter, we have discussed the theoretical basis for the sensitivity and uncertainty methods which we employ to evaluate the influence of fuel performance on nuclear reactor depletion calculations. In the first section, we discussed the general approach to sensitivity and uncertainty analysis. We then discussed two variants of the random sampling approach, followed by a discussion of stochastic expansion methods. Finally, we discussed the methods by which both random sampling and stochastic expansion methods can be used to generate sensitivity information. In the next Chapter, we will apply these methods to the pin cell depletion problem in order to gain physical insights into which fuel performance variables have the greatest effect on depletion calculations, and evaluate the possible importance of uncertain variable interaction.

CHAPTER 6

Uncertainty and Sensitivity Analysis of Pin Cell Depletion

6.1 Introduction

In this Chapter, we apply the uncertainty and sensitivity methods described in the previous Chapter to the pin cell problem in order to quantify the effect that the important fuel-related models have on the neutronics calculation. The goal is to assign uncertainties (i.e., error bars) in the core reactivity, Doppler temperature, axial peaking factor, and fuel and moderator temperature coefficients due to the explicit consideration of uncertainties in fuel thermo-mechanical models. We expect based on the results shown in Chapter 4 that the peaking factors and reactivity coefficients will display very tight error bands, since none of the models appeared to exert strong influence on these quantities. Doppler temperature and reactivity, on the other hand, may be significantly affected. This analysis will quantify the degree of that influence, and help determine the appropriate level of effort to incorporate these effects in a production-level simulation tool. Once the uncertainties are determined, VBD using Sobol' decomposition is applied to determine which inputs and input interactions have the largest influence on each response of interest.

As we saw in Chapter 4, several of the fuel models have a negligible first-order impact. The possibility exists that there may be a non-trivial second order influence on the variance through the interaction terms, but it can be shown through physical arguments that, for this particular case, these interactions do not exist. The justifications for the treatment of each model will be discussed in the next section.

For all these calculations, DAKOTA Version 6.0 was used on a Windows 7 platform with 16 cores available for processing. As each FRAPARCS calculation required approximately 1 hour, which allowed us to explore a number of different techniques. However, even on this kind of machine there were limitations to what could be achieved in the timeframe required, as will be discussed further below.

6.2 Definition of Input Distributions

As is illustrated in Figure 5.1, the first step in performing uncertainty calculations is to define the distributions used to describe the input variables. In many cases, these variables can be defined using standard distributions (ex. normal, log-normal), depending upon the amount of data available to generate the model. In the case of the uncertain variables investigated in this work, however, we will see that the differences between the models and the data cannot be supported with standard distributions. Therefore, we employ DAKOTA's continuous histogram input specification to define the probability distributions, and base the resulting analysis off of these. This is in contrast to many fuel uncertainty calculations performed to date (see, for example, [137], [143] and [171]), where standard distributions are used.

As noted earlier, eight models are used in FRAPCON for uncertainty calculations:

- Fuel thermal conductivity
- Fuel thermal expansion
- Fuel swelling
- Fuel fission gas release
- Cladding creep
- Cladding growth
- Cladding corrosion
- Cladding hydrogen uptake

From the analysis in Chapter 4, we saw that the three most important factors (i.e., displaying a potential impact of >50 pcm on core reactivity for $\pm 2\sigma$) were thermal conductivity, thermal expansion, and creep, with thermal conductivity being dominant. Additionally, swelling and corrosion affected reactivity by approximately 20 pcm at the point of greatest departure from the nominal calculation. For the pin cell problem in which the system operates at nominal power for the duration of irradiation (out to approximately 60 GWd/MTU maximum nodal burnup), fission gas release, growth and hydrogen uptake had negligible difference from nominal

conditions. These results form the basis for the choices in the variables we include in the following analyses.

Of these eight variables, we chose to eliminate four in the subsequent S/UA for the following reasons:

- Fission gas release: the application of FGR uncertainty in FRAPCON is through a bias applied to the gas diffusion coefficient. At the temperatures experienced by the fuel in the pin cell problem, even in cases where other uncertain variables affect temperature, thermal gas diffusion is not activated through this mechanism. It would likely become a contributor in cases where local powers and therefore temperatures become elevated, particularly at higher burnups; however, for the pin cell depletion used in this work, this is not the case. Therefore, it is not considered in the uncertainty analysis.
- Cladding axial growth: axial growth is caused by irradiation. Growth is not a significant mechanism which affects fuel rod performance for the cladding material considered in this work (ZIRLO™); the cladding simply does not grow sufficiently to affect the mechanical response, for any combination of other inputs. Therefore, it is not considered in uncertainty analysis.
- Cladding hydrogen: hydrogen pickup is a phenomenon that is modeled for informational purposes to predict accumulation against set limits. Although hydrogen accumulation contributes to cladding embrittlement under irradiation, at this time FRAPCON does not use this information to modify cladding material properties. Therefore, it is not considered in the uncertainty analysis.
- Cladding corrosion: while the results in Chapter 4 demonstrate a minor but noticeable influence of corrosion thickness on fuel temperatures, further inspection of the model behavior indicated that the use of this model caused corrosion to grow in an unphysical way (see Fig. 4.4.9). It is expected that a more appropriate treatment would be to influence the rate of corrosion growth, rather than the initial amount of corrosion present at the beginning of irradiation. Since we wish to keep the physics as representative of reality as possible, we choose to disregard this particular phenomenon in the uncertainty analysis.

In addition to the fuel models, the effect of the heat transfer coefficient on core depletion was evaluated in Chapter 4, for both coupled and uncoupled models. PATHS was modified to accept a multiplier on the Dittus-Boelter heat transfer coefficient, in a similar way to how relative uncertainties are applied within FRAPCON (see Equation 4.4). With this approach, we found the core reactivity swing from nominal to be approximately on the order of the secondary fuel effects (i.e., approximately ± 20 pcm swing at $\pm 2\sigma$, taking the standard relative error in the HTC to be 20%); as the HTC is increased, the outer cladding temperature decreases, resulting in lower fuel temperatures and a slightly more reactive system. This result held true for both

coupled and stand-alone cases. While not expected to be a dominant factor, we included consideration of HTC uncertainty in the following results.

With the justification of the five uncertainty variables considered within the S/UA, we now define the distributions used to characterize these variables. First, we discuss the HTC. After a literature review, we have decided to use a uniform distribution on the single phase HTC, because data to model comparisons similar to those available for the fuel models discussed below were unavailable. This is applied via a multiplier with lower bound of 0.79 and upper bound of 1.85, in accordance with the NRC CSAU LOCA calculations [13]. Instead to applying the HTC bias as Equation 4.3, PATHS was modified to apply the bias as simply a multiplier:

$$HTC = HTC_{nom} \times h \tag{6.1}$$

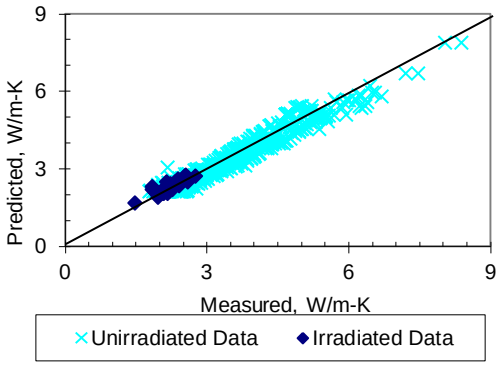
where *h* is the multiplier drawn from the uniform distribution $[0.79, 1.85]$.

The next step is to determine the distributions to use for the fuel models. There are significant data available from work performed to do FRAPCON model assessment [28]. Using these comparisons, PNNL developed standard errors in the models which were hard-wired into FRAPCON to support uncertainty evaluations (see Table 4.4.1). These data used in model assessment was obtained via EXCEL spreadsheets from PNNL [177], in order to determine if standard probability distributions could be used to model the uncertainty in these parameters. We first show the predicted-vs.-measured comparisons in Figure 6.2.1.

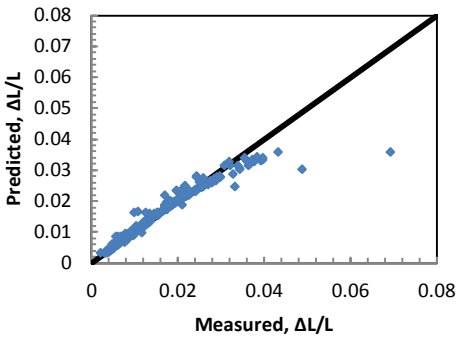
Using this data, we can generate histograms of the deviation of the models' predictions from the measured values, since these are the quantities that we wish to apply in the manner of Equations 4.3 and 4.4. We first tabulated the difference between the model predictions and the data; for conductivity, expansion, and creep, we tabulated the relative difference, whereas for swelling we computed the absolute difference, as shown in Equations 6.2 and 6.3, respectively, consistent with the approach used by PNNL:

$$\Delta_{rel} = \frac{X_{pred} - X_{meas}}{X_{meas}} \tag{6.2}$$

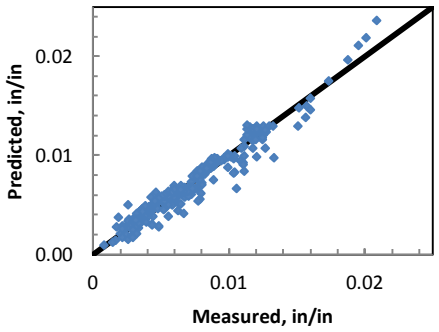
$$\Delta_{abs} = X_{pred} - X_{meas} \tag{6.3}$$



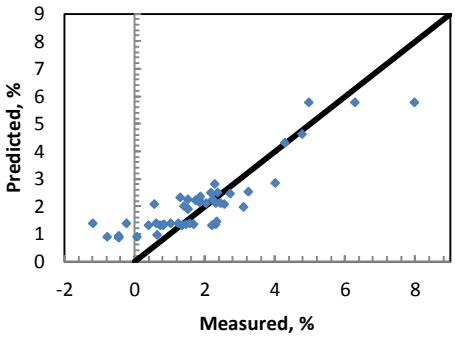
(a) Fuel Thermal Conductivity



(b) Fuel Thermal Expansion



(c) Cladding Creep



(d) Fuel Swelling

Figure 6.2.1: Measured vs predicted values for fuel properties used in S/UA

Note that the fuel swelling uncertainty is applied via the swelling rate. Recall that, for time step i, the swelling that occurs within the step is computed as:

$$\left(\frac{\Delta V}{V}\right)_i = (BU_i - BU_{i-1})(FSR_{nom} + h\sigma_{FSR}) \tag{6.4}$$

where $(\Delta V/V)_i$ is the incremental swelling within timestep i, $(BU_i - BU_{i-1})$ is the burnup accumulated within timestep i, FSR_{nom} is the nominal swelling rate in fractional change per unit burnup, and σ_{FSR} is the standard error in the swelling rate model. In order to use the information presented in Figure 6.2.1(d) to generate an appropriate histogram, the absolute difference in swelling (i.e., $\Delta V/V$) was divided by the burnup at which it was measured to convert from absolute swelling to a rate. Once the appropriate data manipulation was performed, the differences between the model (nominal) values and the measured points across the datasets

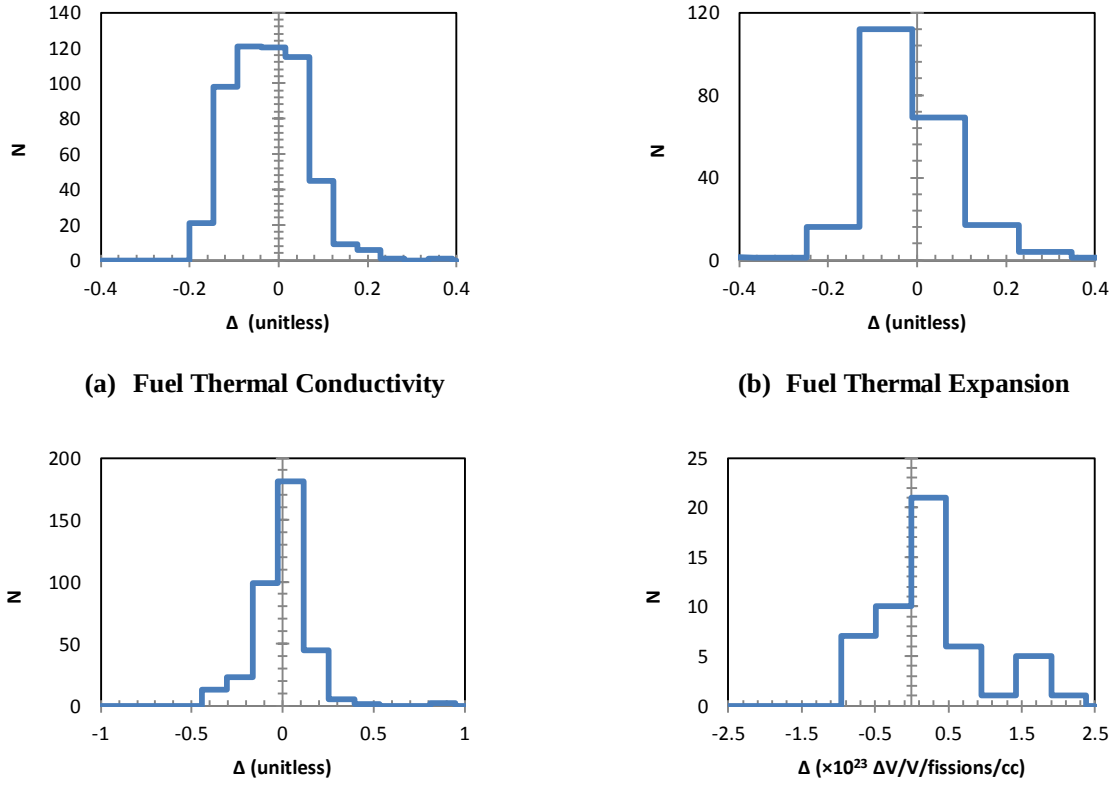
were binned into a histogram. The number of histogram bins was chosen in accordance with the suggestion in [178]:

$$N_{bins} = 1 + 3.3\log(N) \tag{6.5}$$

where N is the number of data points. We choose to use equal bin widths, calculated as:

$$W_{bin} = \frac{\Delta_{max} - \Delta_{min}}{N_{bins}} \tag{6.6}$$

where Δ_{min} and Δ_{max} are the minimum and maximum deviations, respectively, between the model and the fuel data. In Fig. 6.2.2, we present the histograms resulting from this data analysis:



(a) Fuel Thermal Conductivity **(b) Fuel Thermal Expansion**

(c) Cladding Creep **(d) Fuel Swelling Rate**

Figure 6.2.2: Histogram distributions used for uncertain fuel models

Note that several of these distributions are qualitatively similar to the normal distribution, in that the peak is around $\Delta=0$ and tails off as the distance is increased from the origin. There is an obvious benefit to applying standard distributions for stochastic expansion methods, in that the related orthogonal polynomials are well-defined, and these distributions can often be captured with relatively few expansion terms. Therefore, we performed the χ^2 test for normality at 95% confidence as described in [178] for the conductivity, expansion, and creep differences (the swelling rate difference obviously does not follow a normal distribution); the results of these calculations show that the normal distribution is not appropriate for any model. Therefore, these histograms are input directly into DAKOTA as continuous distributions of the `histogram_bin` type (which can be thought of as piecewise uniform distributions) and used in the calculations. A range of expansion orders are employed in conjunction with stochastic expansion methods to study how the statistics and sensitivity coefficients respond with increasing expansion order. This ensures that a expansion is chosen to appropriately capture the underlying input distributions.

Before delving into the uncertainty analysis, we will address the concept of covariance among the input models. In some applications when defining input distributions, different inputs cannot be treated as independent. Therefore, they must be sampled accordingly, using a structure referred to as a *covariance matrix*. The classic example of this in the nuclear engineering field is cross sections (ex. [121], [122]). Methods have been developed to handle these situations, such as ORNL's TSUNAMI tool available in SCALE. The question of whether the fuel models in this work would require the use of a covariance matrix was considered; however, a literature review of the major nuclear materials-related journals did not reveal any consideration of such an issue. Unlike in cross section measurements, where several different subtypes of cross sections are measured simultaneously, the fuel properties are derived from experiments designed to measure one quantity at a time; this makes it quite difficult to manufacture covariance information across data sets where the experimental conditions are not consistent. Therefore, in the absence of such information, we choose to follow the path of other fuel thermo-mechanical uncertainty analysis (see, for example, [137], [143], and [171]) and treat the fuel uncertainties as independent variables. With this, we now turn to the results of the uncertainty analysis.

6.3 Uncertainty Analysis

One goal of the uncertainty analysis, in addition to simply computing the uncertainty associated with the neutronics parameters, was to determine which methods available in DAKOTA provide sufficiently detailed statistics such that they might be used in follow-up analysis with a production-level coupled tool and a larger core. Obviously it is beneficial for any analysis to require as few calculations as possible that meet whatever accuracy requirements exist. Therefore, we will explore a range of different methods with varying degrees of required function evaluations and determine to how the computed statistics change with increasing numbers of samples. Using these results, we can estimate what number of sample sites is sufficient to capture the output distributions.

Before delving into the specifics of each application, we must define what our responses are. Many uncertainty analyses consist of determining the uncertainty in relatively few numbers of responses; for example, in a LOCA analysis we wish to determine the uncertainty and 95% upper bound on three main quantities within the transient: peak cladding temperature, maximum local cladding oxidation, and cladding core-wide oxidation. In this analysis, on the other hand, we are interested in the time history of the neutronics parameters under depletion, as the fuel behavior changes. It would be impractical (and extremely burdensome) to compute the mean, standard deviation, and sensitivity coefficients in each of our outputs of interest at every time step, considering that there are 161 time steps computed in the FRAPARCS calculation (for a total depletion length of 1600 days). Therefore, we will examine the time history of the depletion by sampling at 100 day intervals and computing relevant statistics at these points. This gives sufficient coverage of the time histories in each of the parameters that we can gain insights as to how the fuel performance parameters influence the neutronics calculation throughout the length of the depletion.

We define the following responses, all computed at 100 day intervals:

- Doppler temperature
- Reactivity deviation relative to nominal (unbiased) case
- Axial peaking factor
- Doppler Temperature Coefficient
- Moderator Temperature Coefficient

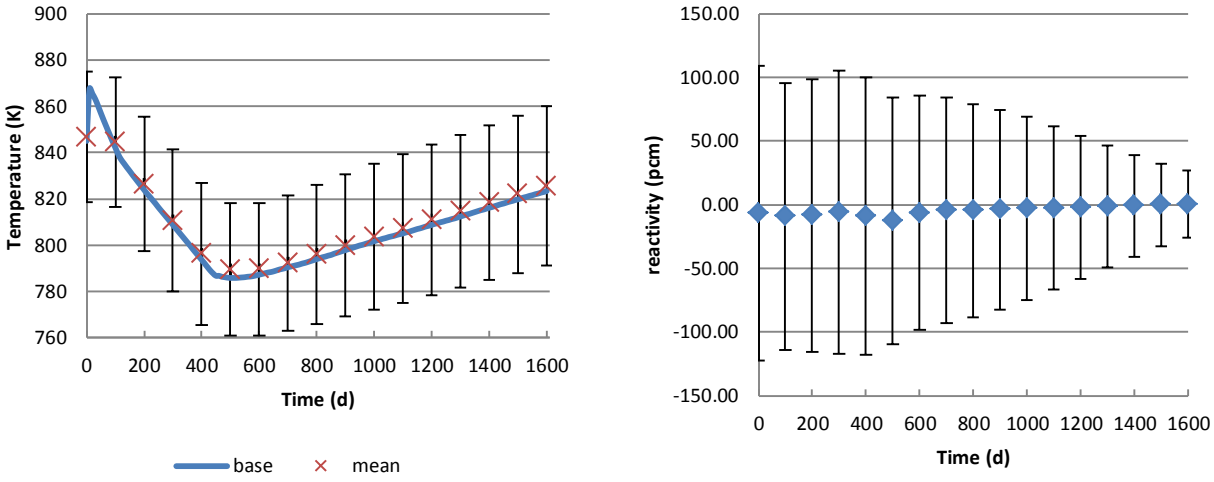
As we have seen from the Chapter 4 comparisons, we do not expect large uncertainties to emerge in the latter three responses of interest. However, for the sake of completeness we will

investigate the potential effect. With this discussion complete, we describe the application of RS and LHS to the coupled pin cell depletion.

6.3.1 Sampling Based Analysis

In order to quantify the effect of increasing the number of samples and to determine the difference between using simple RS and LHS, we performed 5 sets of calculations for each using 10, 50, and 200 samples respectively. Each calculation within a fixed number of samples was seeded with a different number in order to gauge the potential range of possible outputs; as we saw in Chapter 5, we expect the LHS results to have a lower set-to-set variance in the computed statistics. This gives us a total of 30 sets of DAKOTA runs that we performed. We also combined the RS runs to generate a super-set of 1300 calculations which can be used as a reference solution; we cannot do the same with the LHS calculations due to their specific sampling structure. Once we have introduced the reference solution, we will begin delving into the smaller sub-sets to see how many samples were required within each piece to approach the reference.

We begin by presenting the reference results for Doppler temperature and deviation in reactivity from nominal condition in Figure 6.3.1. All plots to follow are provided with 2σ error bars.



(a) Doppler Temperature **(b) Reactivity**

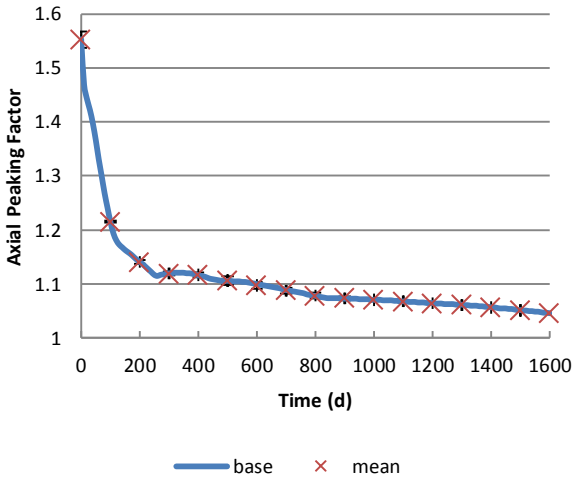
Figure 6.3.1: Doppler temperature and reactivity deviation in RS reference

In this figure, we can see that the mean deviation for both the Doppler temperature and the reactivity are very small; this is to be expected, as the distributions applied to the input variables are approximately (if not perfectly) symmetric. Therefore, we do not calculate a large bias in either parameter based on the probability characteristics of the inputs. We can also see that, while the standard deviation in the Doppler temperature is approximately constant throughout depletion (value ranges from ~14 K at BOC to ~17 K at EOL), the standard deviation in the reactivity decreases from 57 pcm at BOC to 13 at EOC. This is because of the damping effect that plutonium buildup has on reactivity differences later in life; cases which experience higher temperatures and therefore greater negative reactivity relative to the reference solution will develop more plutonium, which later in life yields a positive reactivity effect. This behavior in standard deviation is consistent with what was presented for each of the individual fuel models in Chapter 4. The decrease in reactivity uncertainty particularly begins manifesting itself as the gap closes from 400-600 days, where the main contributor to fuel uncertainty becomes the thermal conductivity. In Table 6.3.1, we present our reference values for Doppler temperature and reactivity:

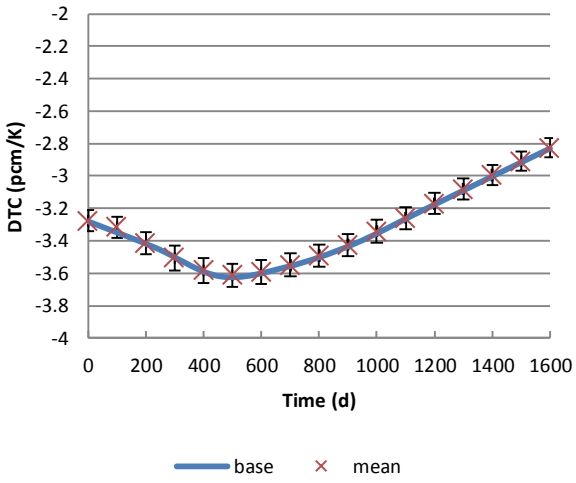
Table 6.3.1: Reference values of Doppler temperature and reactivity

Time (d)	E[T] (K)	σ_T (K)	E[ρ] (pcm)	σ_ρ (pcm)
0	846.66	14.13	-6.82	57.76
100	844.55	14.01	-9.32	52.27
200	826.27	14.50	-8.60	53.48
300	810.59	15.34	-6.11	55.52
400	796.31	15.38	-9.36	54.45
500	789.48	14.29	-12.58	48.50
600	789.63	14.25	-6.40	46.04
700	792.39	14.61	-4.73	44.26
800	795.98	15.01	-4.81	41.96
900	799.84	15.38	-3.98	39.31
1000	803.59	15.69	-2.99	35.86
1100	807.22	15.97	-2.61	32.03
1200	810.97	16.24	-2.11	28.05
1300	814.73	16.51	-1.63	23.95
1400	818.44	16.75	-0.86	19.92
1500	822.06	16.97	-0.12	16.20
1600	825.58	17.19	0.37	13.28

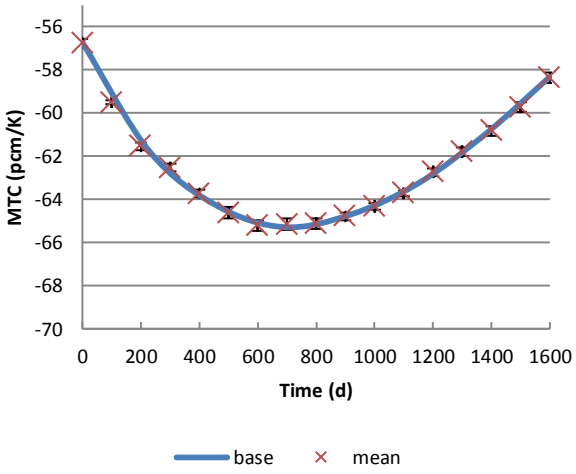
The values we calculate from the subsets and the other methods can be compared with the values in Table 6.3.1 to determine if a sufficiently large sample size (or alternatively, a sufficiently high order expansion) has been used.



(a) Axial peaking



(b) DTC



(c) MTC

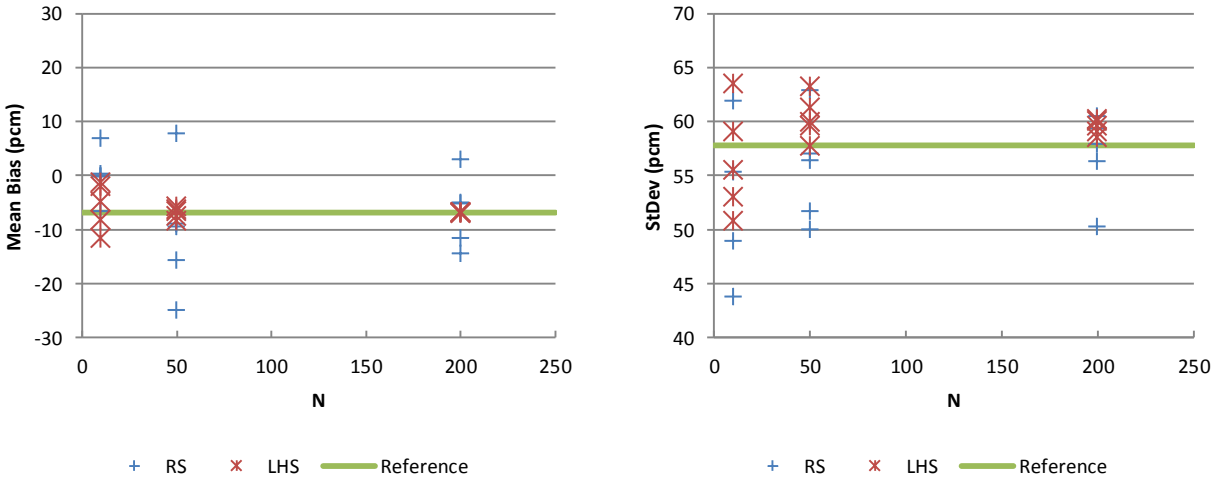
Figure 6.3.2: Axial peaking, DTC and MTC RS reference

In Figure 6.3.2, we can see the effects of varying the fuel parameters on the calculated values for axial peaking factor, DTC, and MTC. As we expected from our results in Chapter 4, no significant uncertainty in these parameters is introduced by uncertainty in the inputs. The axial peaking factor is unaffected because the power shape is not distorted by virtue of including fuel performance physics; rather, the magnitude of the temperature distribution is what changes. The uncertainty in DTC is approximately 1% relative to the nominal value throughout life, likely because the base temperature at which the DTC is evaluated can change significantly enough from case to case to show up in the results. While quantifiable, it is not considered to be a major

concern. The difference in MTC is much smaller (~0.2 %); this is even less important in coupled calculations.

Now that we have presented and explained the reference solution, generated with 1300 random samples, we will examine both the RS subsets and the LHS sets of data. The goal here is to first, determine if the variance in data from set-to-set in the LHS cases is less than the RS data sets (which is what we expect), and second, to determine how many samples we need to generate mean and standard deviation estimates which are comparable to the reference. In order to make efficient presentation, we will choose time points at which either the mean or the standard deviation in reactivity deviation is maximal, and will show the results computed by the subsets at these data points as a function of the number of FRAPARCS evaluations.

First, we will consider the behavior of the BOC reactivity mean and standard deviation across the different data sets. This data is presented in Figure 6.3.3, with the RS reference solution included.

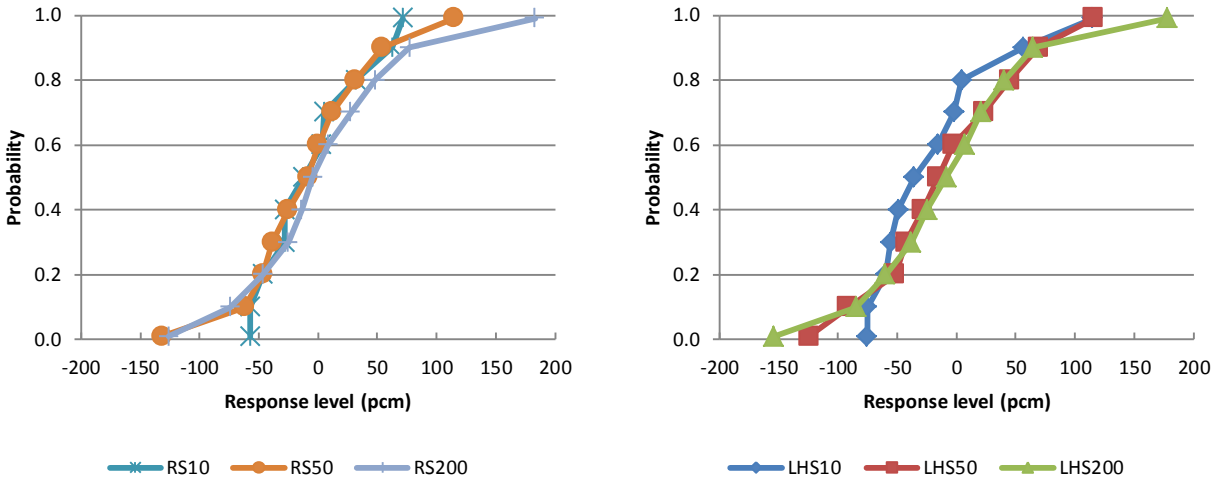


(a) Mean **(b) Standard Deviation**

Figure 6.3.3: BOC mean and standard deviation of reactivity with increasing sample size

As expected, the set-to-set variance of the sample statistics is reduced for LHS relative to RS. From this investigation, the mean has converged by 200 samples in the LHS strategy, but there is still significant variance in the RS mean. A similar behavior us observed in the standard deviation (Figure 6.3.3(b)), although the convergence is not quite as strong; it appears as though the LHS samples estimate a slightly larger standard deviation than the reference solution. This is most likely due to statistical noise and small sample size.

In addition to mean and standard deviation, we can also examine the behavior of the distribution with increasing sample size. For example, let us examine the cumulative distribution function of BOC reactivity. For clarity's sake, we will examine one case from each sample type/number combination to provide an illustration of the effect of increasing the number of samples on resolving the distribution:

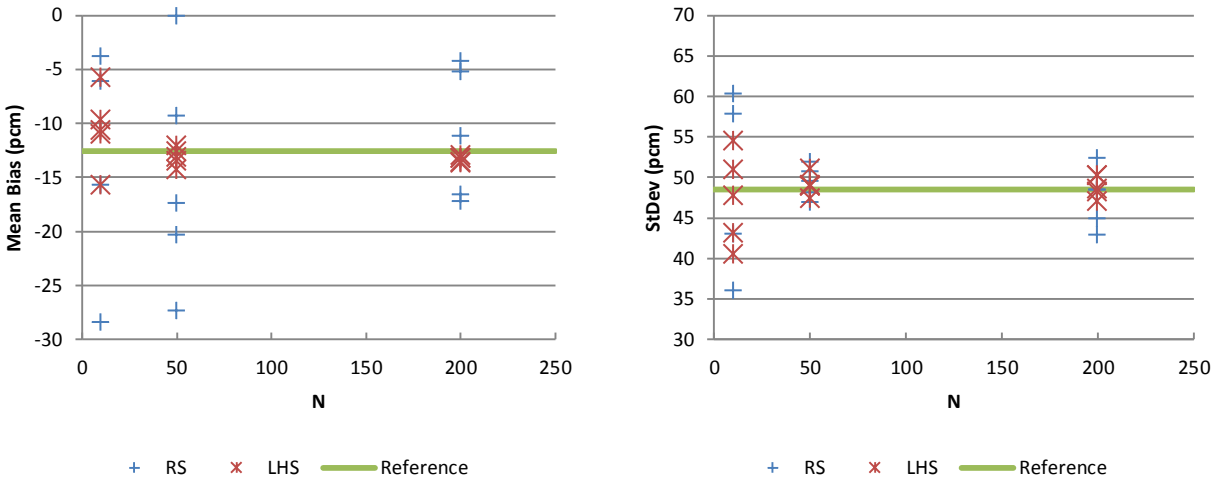


(a) RS **(b) LHS**

Figure 6.3.4: CDF of BOC reactivity with increasing sample size

We can clearly see the effect of increasing the number of samples here; the CDF tends to smooth out, and the tail regions become better resolved as more data is available for the calculation. LHS shows a slightly broader coverage of the sampled regions since the sampling strategy ensures coverage of the lower-probability regions of the input distributions.

We will next present similar results for the time at which the deviation in system reactivity from the nominal value is the greatest; this occurs at 500 days, which is during the time at which the gap is closing throughout most of the fuel rod. We will examine the behavior of both the reactivity and Doppler temperature at this time, to determine if there are physical effects that cause the uncertainty to behave in an interesting way.

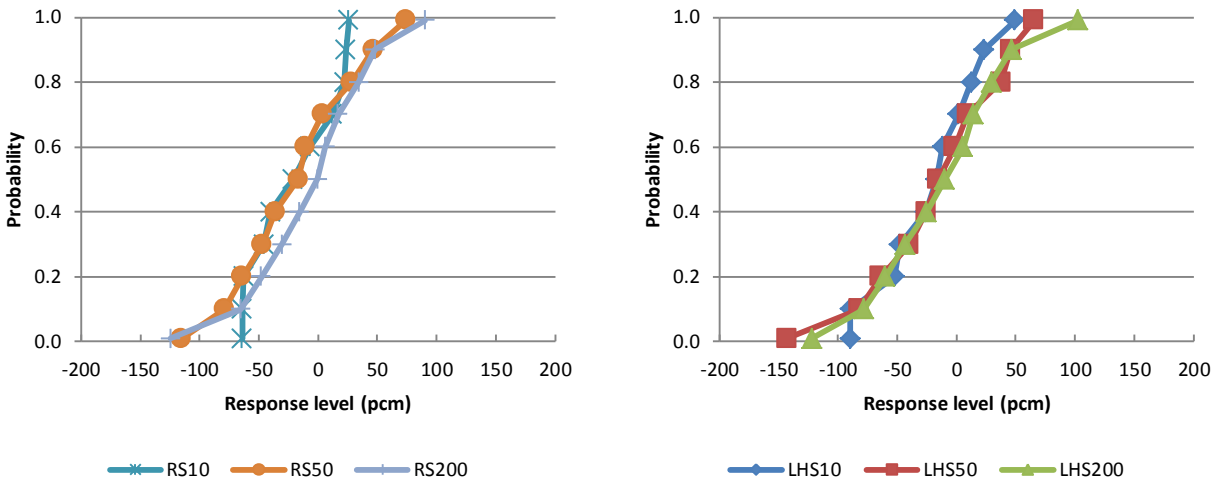


(a) Mean **(b) Standard Deviation**

Figure 6.3.5: 500 day mean and standard deviation of reactivity with increasing sample size

We see similar behavior among the different sampling subsets here as we did in the BOC case; namely, that the LHS samples showed less variance among the different calculations and that, by 200 samples, the mean value in LHS was basically converged. In this case, however, it there appears to be significant agreement between the LHS cases of a slightly lower reactivity bias than the reference. The standard deviation, on the other hand, showed the same clustering behavior, but there is still appreciable spread between the different cases. However, the magnitude of the spread is not what would be considered in most cases significant; in addition to means and standard deviations DAKOTA also computes confidence intervals, which give an interval within which we can be 95% confident that the true mean and standard deviations exist. For the LHS-computed standard deviations, the confidence intervals range about the estimate by approximately ± 5 pcm. Therefore, we can say with confidence that the values we compute here are accurate.

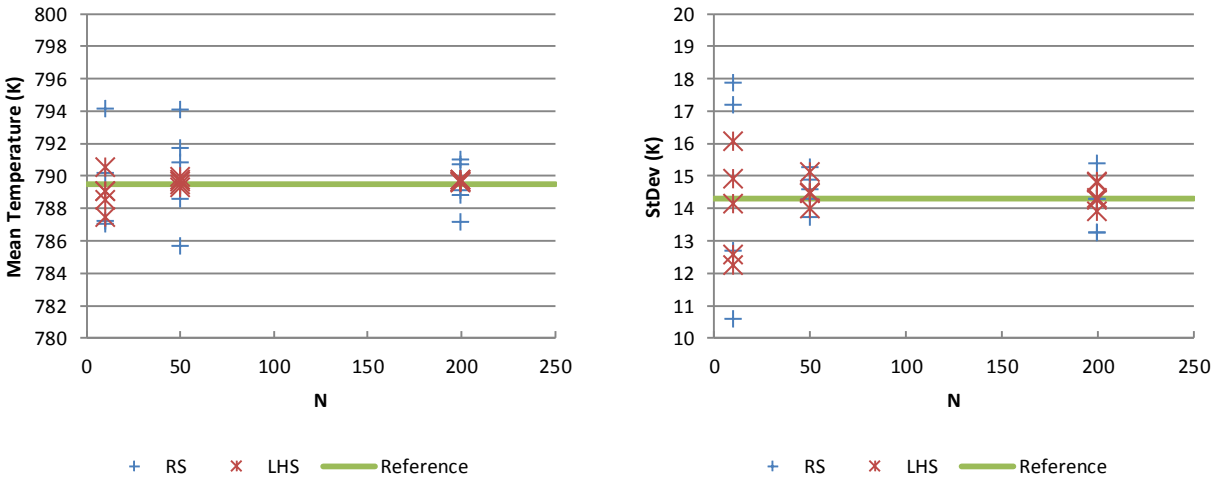
The CDFs for reactivity at 500 days also show similar behavior to those at BOC:



(a) RS **(b) LHS**

Figure 6.3.6: CDF of 500 day reactivity with increasing sample size

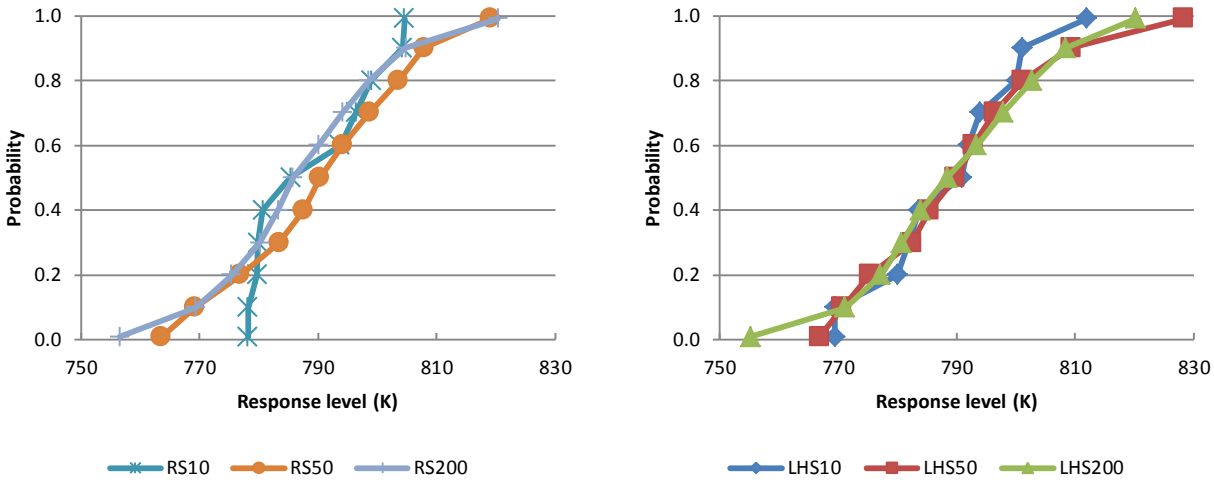
These CDFs are very similar, and show a similar trend in smoothening as the sample size is increased. The range of potential values is narrower in this case than the BOC case; this is consistent with the lower value of standard deviation we compute in this quantity relative to BOC.



(a) Mean **(b) Standard Deviation**

Figure 6.3.7: 500 day mean and standard deviation of temperature with increasing sample size

The Doppler temperature at this time step shows similar clustering behavior that the reactivity cases showed. The temperature mean is approximately the same between the 200 LHS sample cases; there is greater variance in RS vs. LHS, because of the differences in sampling philosophy. We see that the standard deviation shows greater variance than the mean between the different cases, but as with the reactivity case the 95% confidence interval in Doppler temperature is very narrow; we can say with 95% confidence that the standard deviation is within ~1.5 K of the computed value.



(a) RS **(b) LHS**

Figure 6.3.8: CDF of 500 day Doppler temperature with increasing sample size

In Figure 6.3.8, we see that these CDFs also exhibit smoothing and better tail-region coverage with increasing sample size. We can also see the benefit of using the LHS method over the RS method; if we compare the 50 sample vs. 200 sample distributions for the different methods, we see convergence to the higher sample CDF with fewer calculations for LHS as opposed to RS.

Summarizing, we first presented reference values obtained by coalescing the RS results for a total of 1300 samples. We compared the results we obtained from each RS and LHS sample set with the reference values, and showed that within each sample set, there was greater variance in the mean and standard deviations computed with RS as with LHS. It appears from this analysis that sufficient estimates of mean and standard deviation in reactivity and Doppler temperature can be obtained by using 200 LHS samples. In the next section we will explore the

use of stochastic expansion methods to compute these quantities, and determine if similar accuracy can be obtained with fewer function evaluations.

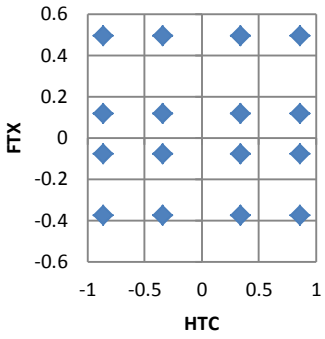
6.3.2 Stochastic Expansions

In this section, we apply the tensor-product quadrature approach available in DAKOTA to explore the potential for representing the coupled depletion uncertainty with Polynomial Chaos and Stochastic Collocation expansions. From the results we saw in Chapter 4, we expect that a low-order expansion (perhaps even linear) will be sufficient to capture the uncertainty behavior; we explore a range of expansions and see how the calculated values change as the number of function evaluations increases. Because tensor-product expansion suffers the curse of dimensionality, we also considered some sparse grids; however, we will see that the sparse grids behave poorly for much of the depletion calculation. The reasons for this are explored below.

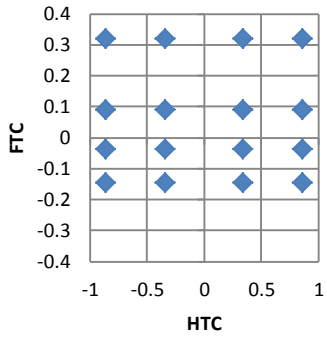
For both PCE and SCE, we explored first, second, and third order tensor expansions respectively. As we explained in Chapter 5, although the expansion polynomials are different, these methods are evaluated at the same tensor grid points. In Figure 6.3.9, we show the projections of each of the fuel variable grid points against the HTC grid points for the third-order grid.

As we have discussed exhaustively in Chapter 5, the grid points are chosen to correspond with Gauss quadrature rules associated with the specified PDFs of the input variables. We can see from this figure that the grid points used for HTC biasing are symmetric and have been transformed from the [0.79, 1.85] interval to the standard uniform interval [-1, 1]. As this is a uniform case, the associated quadrature rule is Gauss-Legendre. DAKOTA numerically generates quadrature rules for the histogram distributions we specified for the fuel variables; we can see from comparing this figure with the pdfs specified in Figure 6.2.2 that the denser collection of points in the quadrature rule correspond with higher-probability regions in the pdf.

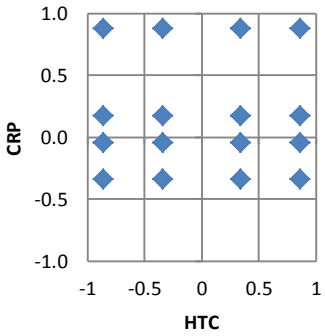
As stated, we have considered isotropic grids in expansion orders 1, 2, and 3, which correspond with 32, 243, and 1024 evaluation points, respectively, in accordance with Equation 5.X. Obviously, since each function call takes approximately one hour to complete, the highest-order cases are quite expensive. The purpose of this analysis is to determine if the statistics can be accurately computed with the lower-order expansions.



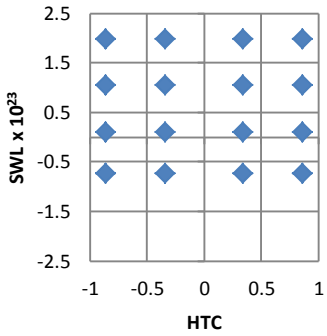
(a) Thermal Expansion vs HTC



(b) Thermal conductivity vs HTC



(c) Cladding creep vs HTC



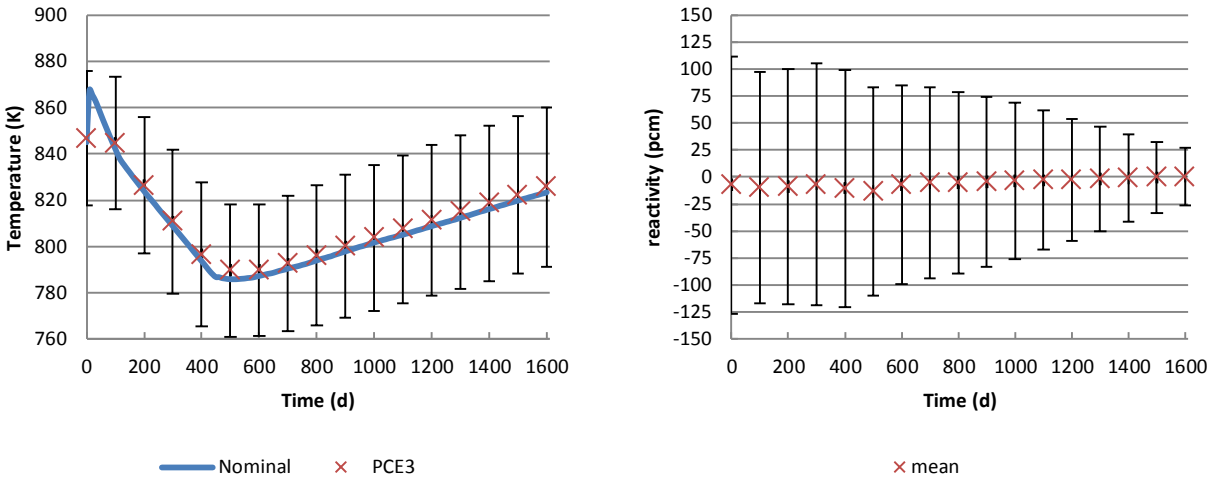
(d) Fuel swelling vs HTC

Figure 6.3.9: Third-order grid for tensor-product expansions

In addition to computing statistics analytically with the expansions, we also applied LHS to the constructed surface in order to generate estimates of the means and standard deviations as well as the CDFs. Ten thousand samples were used to ensure fully adequate coverage. In all cases, the sample-based and analytically-based statistical quantities were the same, which indicates proper convergence of the surface.

In the course of evaluating the results, it became clear that the statistics generated from PCE and SC were identical. This can be the case for tensor-product expansions on synchronized grids, as was discussed in [156]. However, it was not *a priori* obvious that this would be the case for the specific problem in question. As it happens, the expansions end up producing the same polynomials in different settings, leading to equivalence in the calculation of the statistical quantities. In the following presentation of results, we will explore the behavior of the figures of merit as the expansion order is increased.

We will first examine the behavior of Doppler temperature and reactivity of the third-order expansion against the nominal calculation. As in the previous analyses, 2σ error bars are shown:

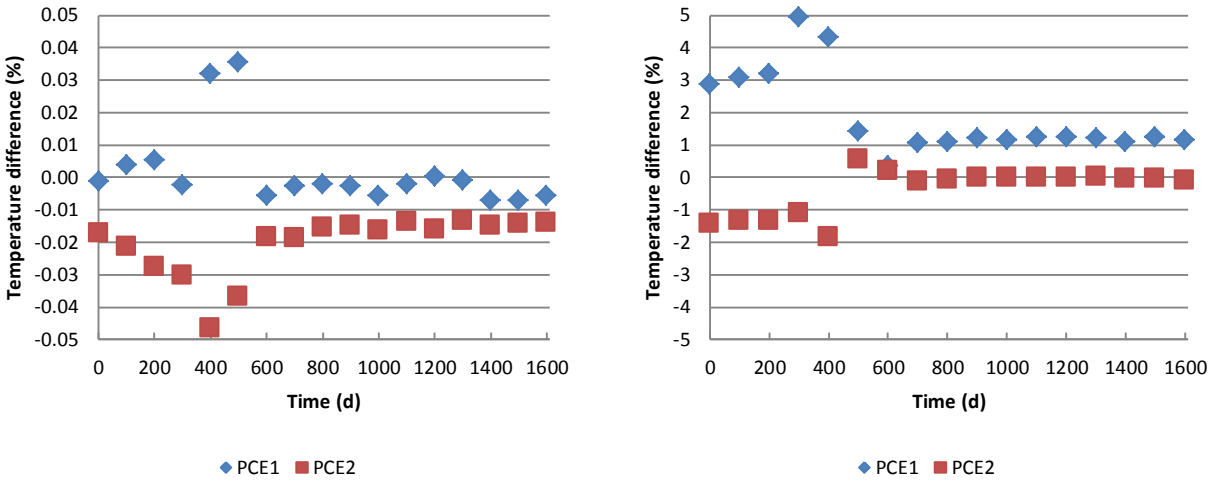


(a) Doppler Temperature **(b) Reactivity**

Figure 6.3.10: Doppler temperature and reactivity evolution relative to nominal

In Figure 6.3.10, the same behavior of the Doppler reactivity can be observed as already seen for the reference case with random sampling. The values computed for the different parameters and the standard deviation in each case are also equivalent. Therefore, it can be concluded that, for the order of calculation considered here, the Random Sampling reference solution can be reconstructed with stochastic expansion methods. The uncertainty in the fuel temperature slightly increases from ~14 K initially to ~17 K at the end of the depletion, mostly due to thermal conductivity uncertainty. The uncertainty in reactivity, on the other hand, is highest initially, where there are a number of models contributing. But after gap closure, the uncertainty decreases with the relative buildup of plutonium in the core damping out reactivity differences. This is expected behavior, as we saw in figures in Chapter 4. As with the random sampling cases, power distribution and reactivity coefficients showed very tight error bands.

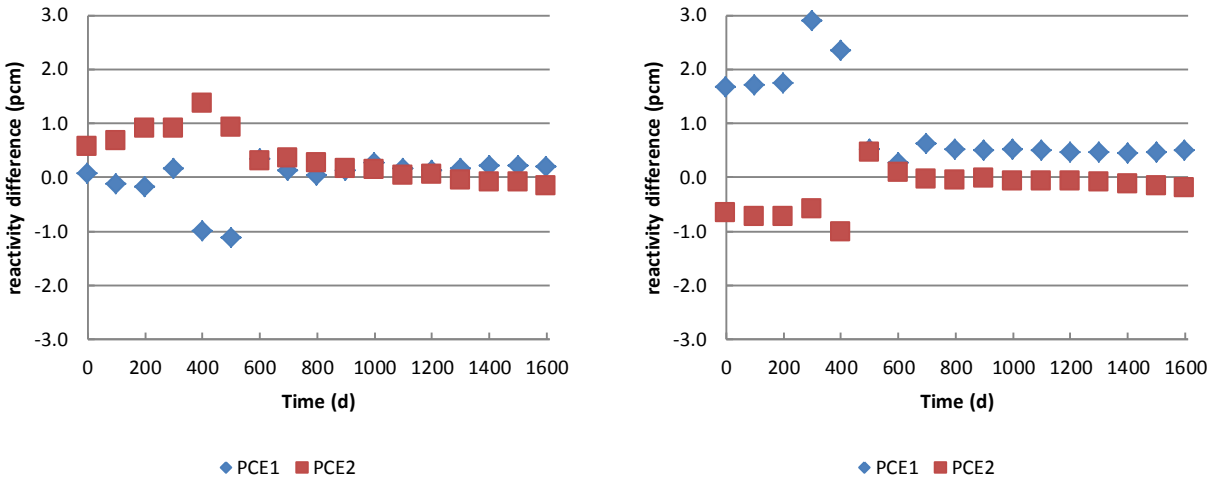
In order to determine the importance of converging the sample space, the differences in Doppler temperature and reactivity as a result of the first- and second-order expansions, respectively, are analyzed relative to the third-order expansion in Figure 6.3.11.



(a) Mean **(b) Standard Deviation**

Figure 6.3.11: Temperature difference relative to third-order expansion

In Figure 6.3.11, it can be observed that the relative difference in Doppler temperature mean is practically insignificant. Differences are primarily manifest as time history effects; there is a tendency for the second order expansion to predict a lower Doppler temperature during the gap closure phase, while the mean of the first order method is slightly higher. This may be due to the placement of the evaluation points in uncertainty space; as we can see from Figure 6.3.9, the third-order expansion (and the first order as well) are evaluated at points in the quadrants of the input ranges, whereas the second-order expansion includes points evaluated in the center. This may have some influence over the differences in mean and standard deviation we see here. On the other hand, the relative differences in standard deviation are somewhat more significant, although for most of the depletion they end up being on the order of less than a percent. Again, the standard deviation is largest during the gap closure phase and the uncertainty contributions from all of the models and their differing relative effects are taken into account. Once the gap is closed, however, the values stabilize. The second-order expansion is clearly sufficient to characterize the spread in the temperature calculations throughout this phase. In all cases, however, the absolute difference in temperature as the expansion order is increased is very small; for computational efficiency purposes, the linear (i.e., first-order) expansion should be sufficient to capture the evolution in temperature uncertainty throughout the depletion.



(a) Mean **(b) Standard Deviation**

Figure 6.3.12: Reactivity difference relative to third-order expansion

In Figure 6.3.12 we see that the differences between the third-order expansion (which required 1024 calculations) and the first- and second-order expansions are very small. In all cases, the differences in the means of the reactivity deviation are less than 2 pcm, which is negligible. Additionally, we see that the computed difference in standard deviation is very small. As we would expect, the second-order approximation is in general closer to the third order reference, particularly in standard deviation during gap closure. Additionally, there is an obvious time history effect; the differences chiefly occur early in life, when the fuel-cladding gap is closing and the effects of the mechanical models important to capturing gap closure are most apparent. If we compare the results in Figure 6.3.10 with Figure 6.3.11, we see that the deviations in reactivity mean and standard deviation occur simultaneously with the deviations we see in the temperature statistics. The reasons for this are clear: deviations in computed temperatures are inversely correlated to the differences in temperature, and the places at which the temperature spread is largest relate to the places at which the reactivity spread is also largest. A higher-order expansion is necessary in order to capture these effects; however, given their level of significance, it is likely that in most applications lower-order expansions would be sufficient.

In addition to these tensor-product expansions, we attempted to implement the sparse grid methods available in DAKOTA to determine if the equivalent of even higher-order expansions could be achieved. However, the DAKOTA machinery for these sparse grid methods was not

able to converge on an expansion for any sparse grid levels that we attempted. One potential reason for this is challenges that sparse grid methods may have in representing the histogram-type input uncertainty distributions; since the distributions are not smooth, there may be issues in attempting to essentially interpolate within the regions of probability space where no points exist. The resulting expansions were therefore not stable, and in many cases integration over the probability space resulted in negative variance. This is obviously unphysical, and should be explored further to better understand the relevant phenomena, since sparse grid methods hold great promise in making PCE and SC expansions applicable to practical engineering analysis problems.

With this, we conclude our presentation of the stochastic expansions we used in this work. Summarizing, we exercised the PCE and SC capabilities of DAKOTA for the pin cell depletion problem and determined that the expansions are essentially the same. The means and variances in the responses for the highest-order expansion were very comparable to the reference random sampling calculation. We also saw that the lower-order expansions were able to capture the mean and standard deviation in Doppler reactivity relative to the third-order expansion to within a few pcm of reactivity and generally a fraction of a percent of Doppler temperature. The differences in the expansions mostly manifested themselves during the gap closure phase, where the locations and numbers of calculations performed can have an impact on capturing the uncertainty behavior. However, in our judgment the linear expansion (requiring only 32 calculations) would be sufficient for most practical analysis.

6.3.3 Conclusions

Before moving to the sensitivity analysis, we will provide a comparison of the temperatures and reactivities computed using random sampling and PCE/SC. In Table 6.3.2, we present the temperatures, and in Table 6.3.3 we present reactivity for the random sampling reference case and the three polynomial chaos expansions.

It should be obvious from these tables that there are no significant differences in the values we estimate for Doppler reactivity across the different methods. Indeed, even with only 32 simulations used with the linear stochastic expansion, we were able to compute very closely the statistics of the system responses in question. Therefore, especially given that these calculations

can be quite expensive, it would be advantageous to explore the use of PCE for the depletion uncertainty analysis of more practical problems.

Table 6.3.2: Doppler temperatures across methods

	RS Reference		PCE1		PCE2		PCE3	
Time (d)	**E[T] (K)**	**σ_T (K)**	**E[T] (K)**	**σ_T (K)**	**E[T] (K)**	**σ_T (K)**	**E[T] (K)**	**σ_T (K)**
0	846.66	14.13	846.74	14.96	846.60	14.34	846.75	14.55
100	844.55	14.01	844.67	14.77	844.46	14.14	844.64	14.33
200	826.27	14.50	826.44	15.22	826.16	14.55	826.39	14.74
300	810.59	15.34	810.75	16.26	810.53	15.33	810.77	15.50
400	796.31	15.38	796.83	16.16	796.21	15.21	796.58	15.49
500	789.48	14.29	789.99	14.46	789.43	14.34	789.71	14.26
600	789.63	14.25	789.76	14.28	789.66	14.25	789.80	14.23
700	792.39	14.61	792.57	14.79	792.45	14.62	792.59	14.64
800	795.98	15.01	796.18	15.22	796.08	15.04	796.20	15.05
900	799.84	15.38	800.03	15.59	799.93	15.41	800.05	15.41
1000	803.59	15.69	803.77	15.91	803.69	15.72	803.82	15.73
1100	807.22	15.97	807.42	16.22	807.32	16.02	807.44	16.02
1200	810.97	16.24	811.19	16.49	811.06	16.29	811.19	16.29
1300	814.73	16.51	814.94	16.75	814.84	16.55	814.94	16.55
1400	818.44	16.75	818.61	16.98	818.54	16.79	818.67	16.80
1500	822.06	16.97	822.22	17.22	822.17	17.00	822.28	17.01
1600	825.58	17.19	825.76	17.41	825.69	17.20	825.80	17.21

Table 6.3.3: Reactivities across different methods

	RS Reference		PCE1		PCE2		PCE3	
Time (d)	**E[ρ] (pcm)**	**σ_ρ (pcm)**	**E[ρ] (pcm)**	**σ_ρ (pcm)**	**E[ρ] (pcm)**	**σ_ρ (pcm)**	**E[ρ] (pcm)**	**σ_ρ (pcm)**
0	-6.82	57.76	-7.11	61.20	-6.62	58.86	-7.18	59.52
100	-9.32	52.27	-9.73	55.16	-8.93	52.74	-9.59	53.46
200	-8.60	53.48	-9.24	56.14	-8.16	53.67	-9.07	54.40
300	-6.11	55.52	-6.63	58.95	-5.88	55.47	-6.79	56.05
400	-9.36	54.45	-11.35	57.15	-8.99	53.81	-10.36	54.82
500	-12.58	48.50	-14.51	48.79	-12.46	48.75	-13.39	48.28
600	-6.40	46.04	-6.59	46.22	-6.64	46.05	-6.93	45.95
700	-4.73	44.26	-5.22	45.00	-4.99	44.36	-5.35	44.39
800	-4.81	41.96	-5.35	42.65	-5.13	42.09	-5.39	42.14
900	-3.98	39.31	-4.34	39.94	-4.30	39.42	-4.46	39.45
1000	-2.99	35.86	-3.19	36.53	-3.32	35.95	-3.47	36.02
1100	-2.61	32.03	-2.85	32.67	-2.98	32.11	-3.01	32.18
1200	-2.11	28.05	-2.35	28.65	-2.42	28.12	-2.48	28.20
1300	-1.63	23.95	-1.74	24.57	-1.94	24.01	-1.89	24.10
1400	-0.86	19.92	-0.85	20.54	-1.14	19.96	-1.06	20.10
1500	-0.12	16.20	-0.07	16.84	-0.36	16.23	-0.28	16.38
1600	0.37	13.28	0.51	13.95	0.16	13.26	0.32	13.46

The uncertainty analysis using a spectrum of methods available in DAKOTA has now been presented, and we found that we were able to largely replicate the 1300 sample case with only 32 function evaluations with a polynomial chaos expansion. This would indicate that the response is quite linear in nature; to fully explore this question, we turn to sensitivity analysis, whose goal it is to determine precisely which input variables have the largest effect on the uncertainty in the depletion calculation.

6.4 Sensitivity Analysis

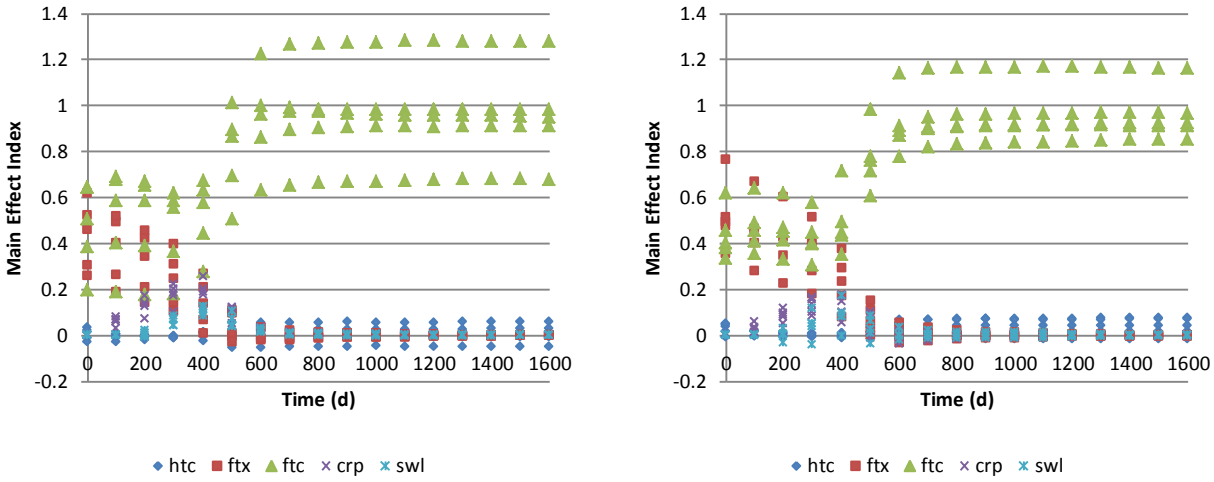
In this section, we discuss the application of Sobol' decomposition to the uncertainty analysis presented in the previous chapter. For the random sampling cases in question, we employed the method described in Chapter 5 to compute the main and total effect indices for each of the five RS and LHS calculations with 50 samples. As discussed in Section 5.4, this means for each estimation of these indices, 350 function evaluations were necessary. While we acknowledge that Saltelli et al. recommend at least 100 base samples to evaluate these coefficients, since the calculations are so expensive and because the response statistics do not show an inordinate sensitivity to the number of calculations, we decided to apply VBD for the 50 sample cases. Additionally, we can examine the simple (Pearson) and rank (Spearman) correlation indices, given in Equations 5.42 and 5.43 respectively.

We also applied VBD to the PCE/SC cases; as we have stated above, one of the chief benefits of stochastic expansions is that the sensitivity indices are computed analytically and do not require additional function evaluations beyond that required to form the expansion. In this section, we will show how the main and total effect indices evolve throughout the depletion, showing the relative importance of each uncertainty variable to the overall system variability. We will also see that the importance of input interactions is in fact almost negligible, and that the variability is mostly dominated by the linear terms.

6.4.1 Random Sampling

We will first investigate the relative contributions to uncertainty with VBD in the random sampling methods. DAKOTA uses Equation 5.52 and 5.55 to evaluate the Main Effect (ME) and Total Effect (TE) indices, respectively. We first present the computed MEs throughout depletion

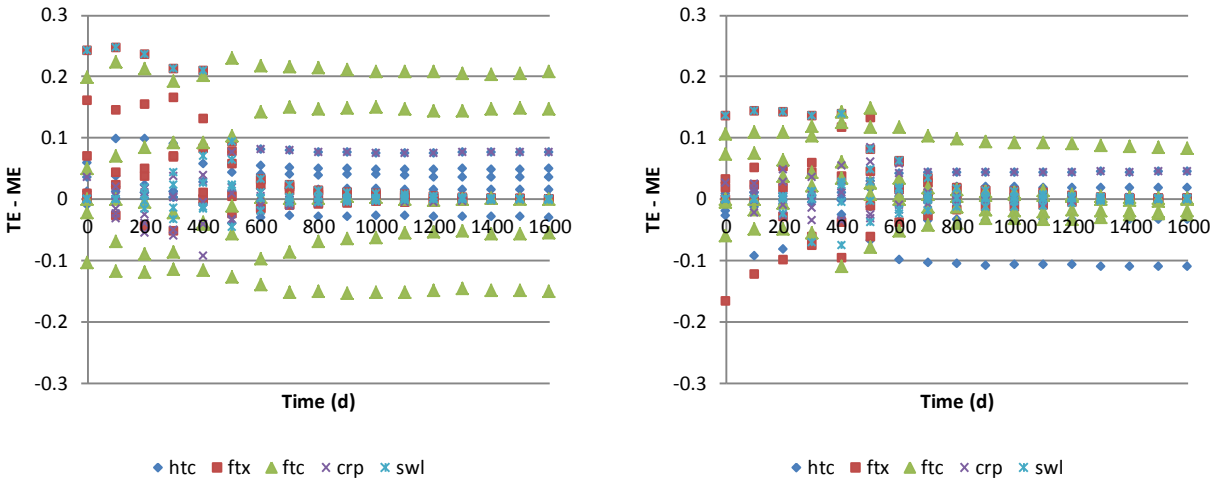
on the variance in Doppler temperature across all five respective cases. Then, we evaluate the difference in TE and ME to estimate the potential importance in model-to-model interaction.



(a) RS **(b) LHS**

Figure 6.4.1: Main Effect Index for Doppler Temperature

In Figure 6.4.1 it can be observed that there is significant variability between the cases with regards to the ME indices estimated. Additionally, we see that in some cases, the ME computed via the random sampling method is outside the expected interval [0, 1]; this is obviously not expected, given the definition of the ME index. However, since the random sampling approach to calculating ME is essentially based on estimating the variance integrals using a Monte Carlo technique, and the sample size we used in this particular exploration was not large, we can conclude that the calculation of this parameter outside its bounds is likely due to relatively poor resolution of the sample space. Even so, from comparing the general trends across these data sets, we find that early in life, prior to gap closure, there is generally competing effects (initially between the fuel thermal conductivity and thermal expansion models, and then creep and swelling make a contribution) in ME. This is because, as we have stated before, this is the time interval in which the fuel is changing the most. Once the gap has closed, however, the Doppler temperature variance is dominated by the variance in fuel thermal conductivity. There does not appear to be a significant difference in the relative spread of the different indices computed by LHS vs. RS.



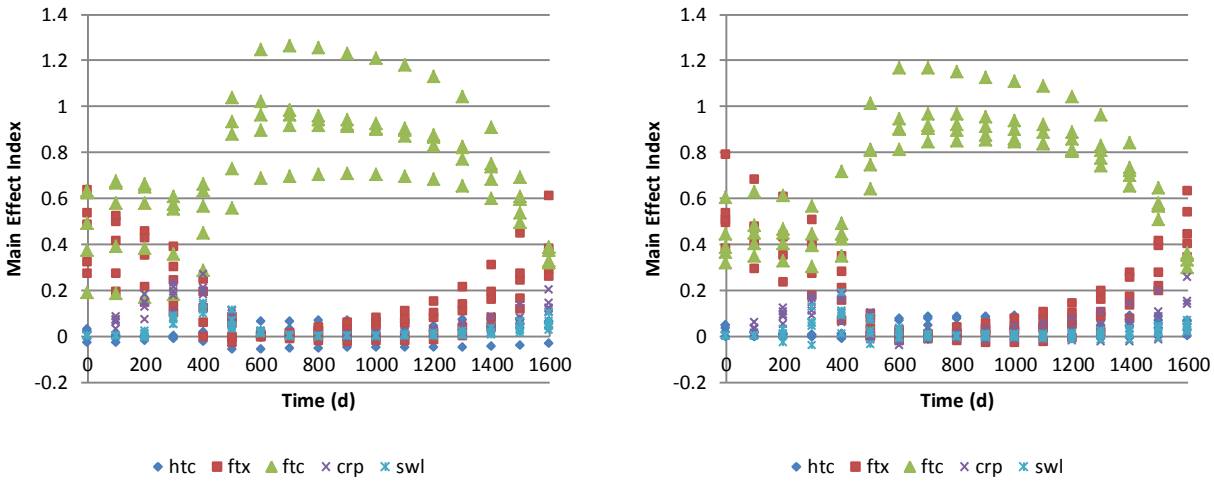
(a) RS **(b) LHS**

Figure 6.4.2: Model-to-model variance effects for Doppler Temperature

In Figure 6.4.2, we see some of the same statistical noise effects in this calculation that we saw in the ME calculation. By definition, the TE is greater than the ME since it is a sum of all contributions to variance involving a particular uncertain variable; the negative values that we see here are likely due to poor estimates of the integrals via Equation 5.56. Nevertheless, we can draw some conclusions from this figure; it appears as though the difference between TE and ME is approximately near zero when averaged across all cases. This would indicate that the variance is primarily dominated by ME rather than interaction terms; however, the resolution is so poor with this small sample size that we cannot draw many additional conclusions. The differences in TE and ME are generally narrower for LHS, which is a product of the structured sampling scheme; the non-linear contributions are better resolved with this strategy.

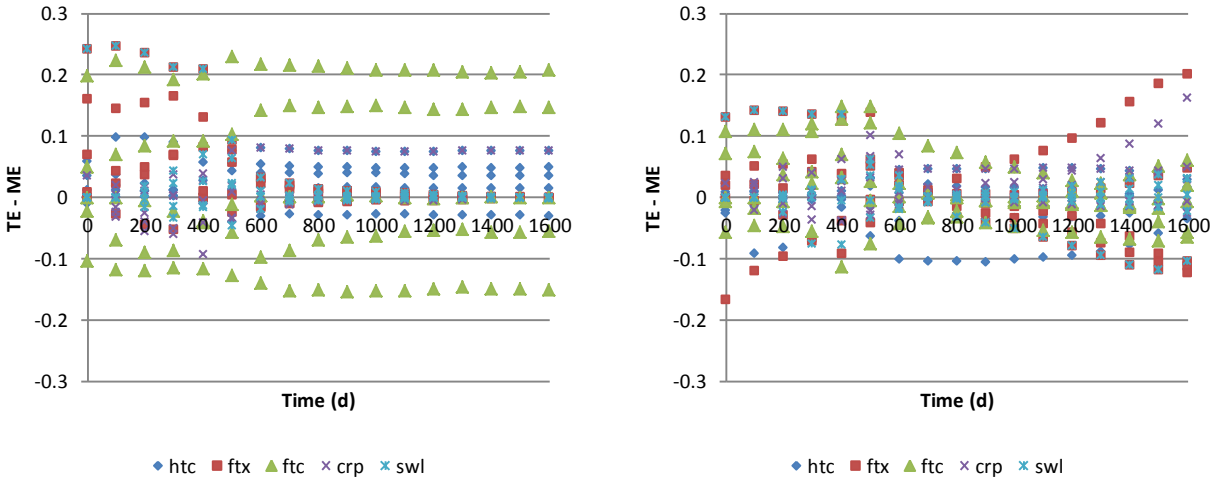
Next, we will present the sampling results for reactivity ME n Figure 6.4.3. We can again see in this figure the poor resolution of the variance integrals due to insufficient sampling. However, despite this we do see evidence of some physical effects that we might not otherwise have seen. The variance in reactivity early in life is due to a combination of the physical effects as the gap is closing, and then Doppler temperature dominates for most of the rest of the depletion. However, we can see here that near the end of depletion, other effects (most importantly the thermal expansion) again manifest themselves as significant contributors to the uncertainty in the reactivity. This is due not so much to the fuel thermal expansion being more important, as it is due to fuel thermal conductivity becoming less important. As we have seen,

the history effect of plutonium buildup depending on prior operating conditions causes the overall uncertainty in the reactivity calculation to decrease later in life. This history effect causes the temperature (dominated by thermal conductivity) to likewise become a less important contributor to uncertainty in reactivity. In its place, fuel thermal expansion becomes relatively more important.



(a) RS **(b) LHS**

Figure 6.4.3: Main effect index for reactivity



(a) RS **(b) LHS**

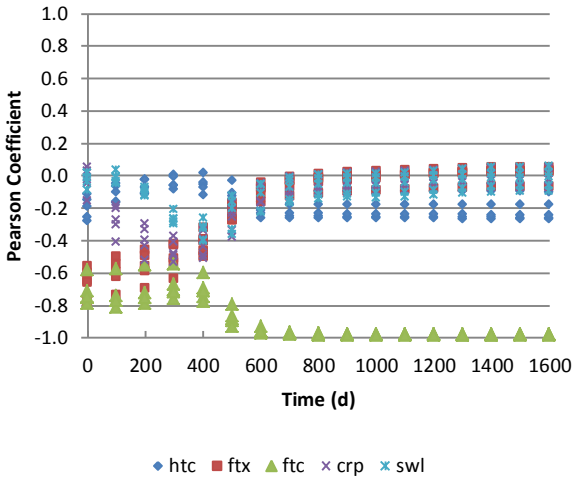
Figure 6.4.4: Model-to-model variance effects for reactivity

In Figure 6.4.4, we present the difference in TE and ME for reactivity throughout the depletion. We see similar behavior here as we saw in Figure 6.4.2; namely, poor resolution in the integral estimates has caused the difference in TE and ME to be outside their defined ranges. Regardless, it is clear that the centroid of the values within each data point is approximately about the zero line, and therefore we can expect that the models are primarily linear. Additionally, there appears to be one LHS sample that predicts the interaction effects of the thermal expansion model to be more important as the depletion continues; this is likely an artificial product of that particular calculation's sampling scheme, and therefore is not likely to be an actual physical effect. We will find no evidence in the analytic expansions of something like this.

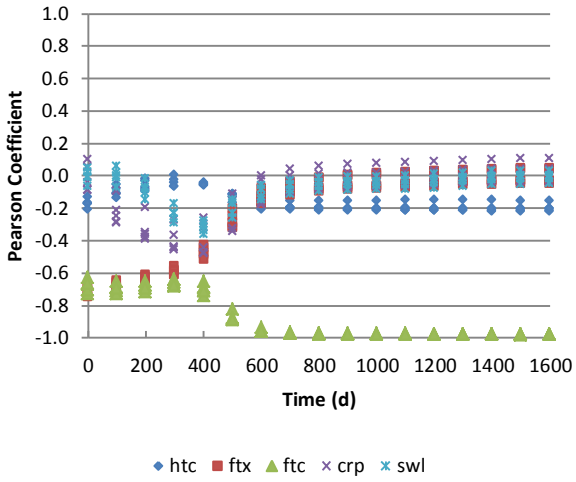
Before discussing the VBD results with stochastic expansions, we will touch on the Pearson and Spearman coefficients for these cases. As we discussed in Chapter 5, these coefficients are defined based on the assumption that there is no interaction between the input variables. Pearson is defined based on a linear regression of the outputs on the inputs, whereas Spearman is derived based on the ranks of the data rather than the values themselves. In the event that a model is linear and non-interacting, we can expect that these coefficients will adequately describe the contributions to variance in the data. DAKOTA calculates these coefficients as part of its standard output. We will next present these results and discuss their values relative to the ME and TE indices; for linear non-interacting systems, the ME index is equivalent to the Pearson coefficient.

We can see from Figure 6.4.5 that there is not an appreciable difference between the Pearson and Spearman correlation coefficients, which is indicative of a system in which linear regression of the outputs against the inputs is appropriate. We see in general negative correlations between the input values we use and the outputs; this is physically reasonable, since for all these models biasing the model in the positive direction produces a negative reactivity effect. Within the cases, there is general agreement in the predicted value of the correlation coefficients. We do not see an appreciable difference in the spread in these quantities between RS and LHS. The magnitude of the ME coefficients presented in Figure 6.4.1 generally agree in value and trend with those present in the above Figure; the fuel thermal expansion and conductivity dominate early in the depletion, with creep and swelling increasing prior to gap

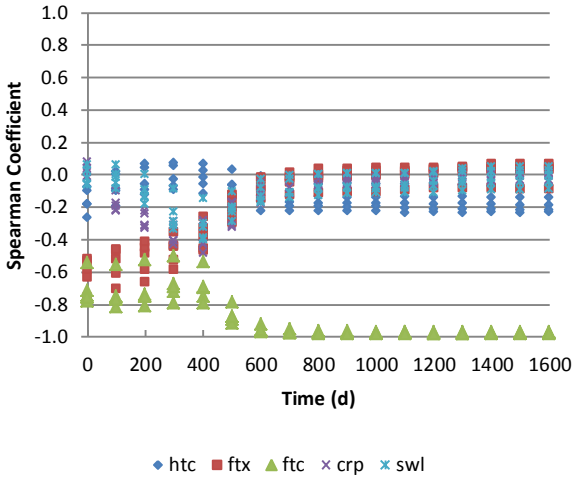
closure. Once the gap is closed, however, only thermal conductivity is closely correlated to temperature, which is what we expect.



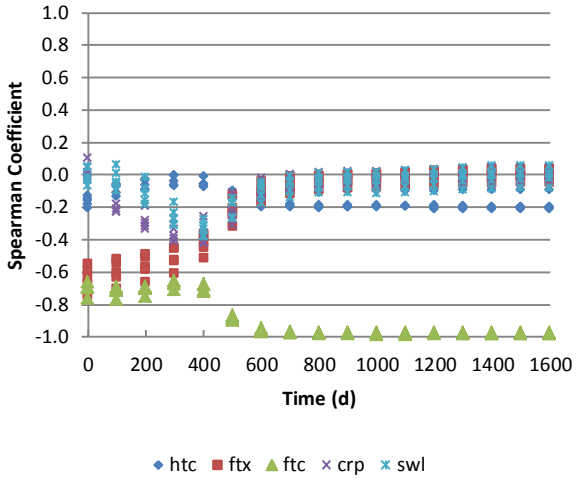
(a) Pearson RS



(b) Pearson LHS



(c) Spearman RS



(d) Spearman LHS

Figure 6.4.5: Pearson and Spearman Coefficient for Doppler Temperature

In Figure 6.4.6 we again see general agreement between the Pearson and Spearman coefficient, indicating that reactivity is also largely a linear function of the uncertain inputs. The fuel thermal conductivity model is positively correlated throughout the depletion, although the correlation decreases later in life as the effects of plutonium buildup on core reactivity are felt. Thermal expansion, cladding creep, and swelling are generally positively correlated early in life, and then become negatively correlated, again because of the plutonium history effects. The heat transfer coefficient is weakly correlated with reactivity, indicating that it is a minor effect.

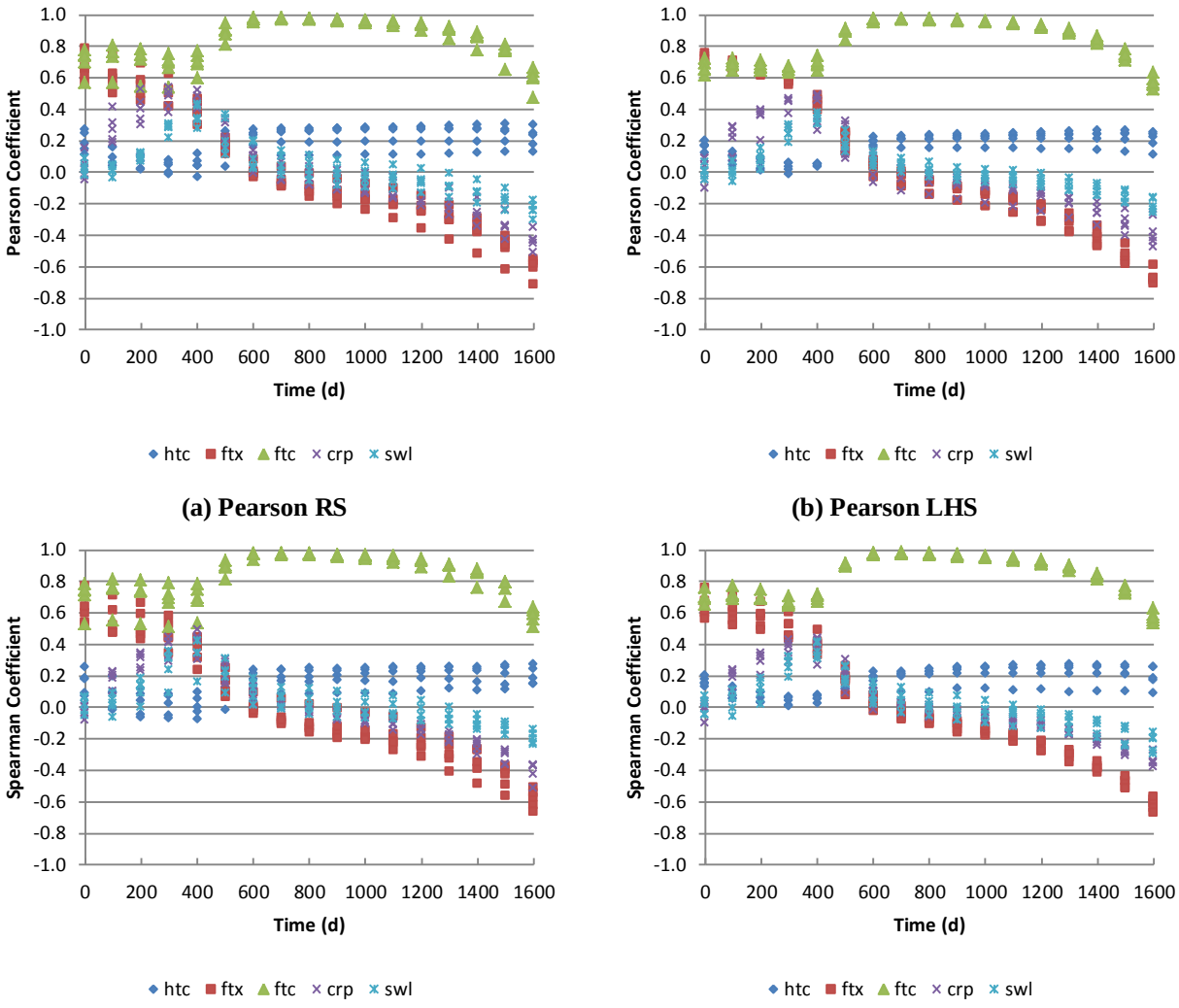


Figure 6.4.6: Pearson and Spearman Coefficient for reactivity

6.4.2 Stochastic Expansions

Now that VBD and correlation methods have been applied to the pin cell problem, we will investigate the analytic approach to calculating the ME and TE indices with stochastic expansions. As was the case with the uncertainty analysis, we found that the PCE and SC expansions of equivalent orders returned almost the same sensitivity indices (on the order of 10^{-12} difference at maximum). Therefore, the results we apply below are representative of both methods. As before, we first discuss the ME index for Doppler temperature:

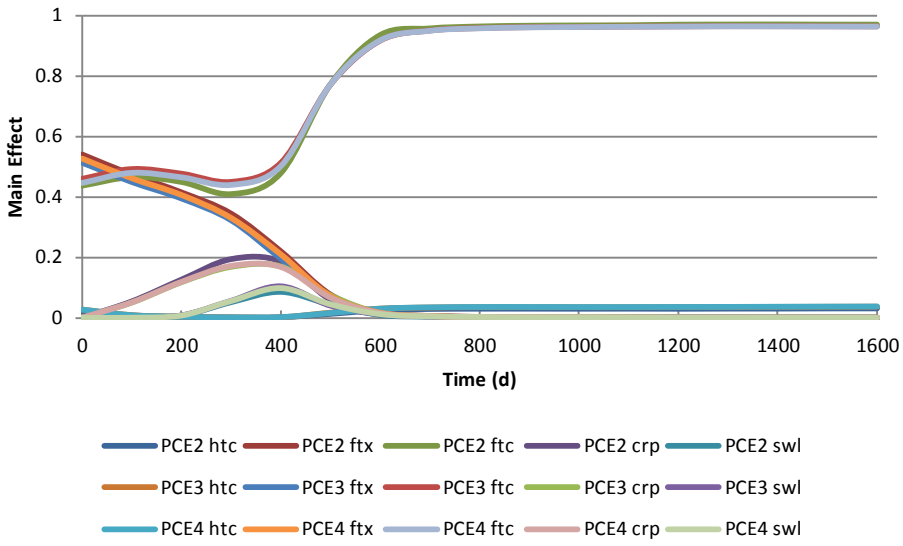


Figure 6.4.7: Main effect index for Doppler temperature

We can see from this figure general agreement in trends between the random sampling-based VBD analysis, albeit with much less noise. Also, as was the case with the uncertainty analysis provided above, we see that the three expansion orders are in very close agreement on the values of ME. This indicates that a linear expansion is sufficient to capture the sensitivity behavior. At the beginning of the depletion, the most important contributor to variance in Doppler temperature comes from the fuel thermal expansion model; physically, this is because this model dictates the degree in which the gap closes as the fuel comes to temperature. The other dominant effect initially is in the thermal conductivity, which is inversely proportional to temperature; obviously variance in this model would play a key role in the overall variance of the output. As the fuel depletes through the gap-closure regime, the instantaneous effect of the thermal expansion model decreases in relative significance as the other models take a more prominent role. The role of creep variance increases from zero initially to reach its maximum value at around the time the gap begins to close throughout the model because this is the time in which its influence over the cladding-gap temperature drop is largest. We see a similar effect in the swelling model, although it is delayed due to the minimum threshold before the model is activated. After the gap closes, the most important contributor to temperature variance becomes the thermal conductivity, with heat transfer coefficient playing a minor role.

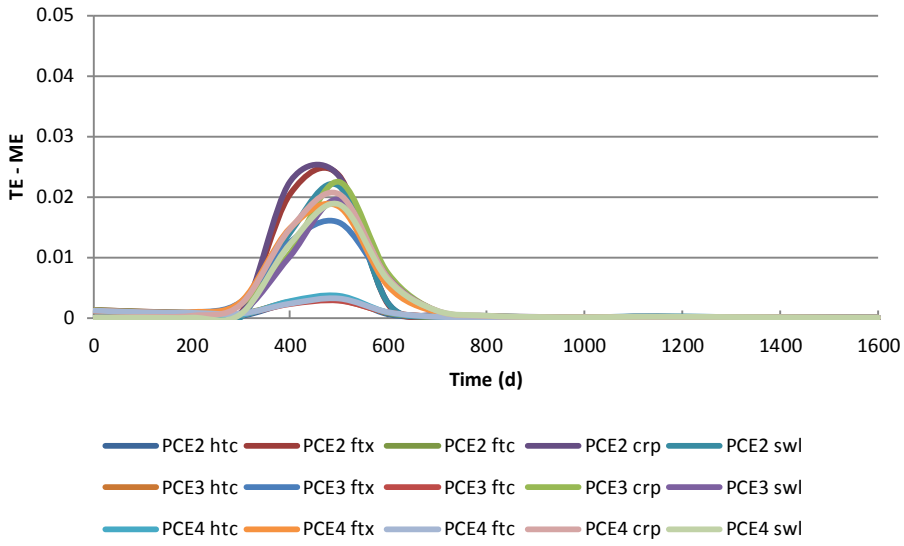


Figure 6.4.8: Input interaction metric for Doppler temperature

In Figure 6.4.8, we investigate the difference in ME and TE to determine whether non-linear input interactions are important for this problem. As we can clearly see in this figure, some interaction effects exist; however, their magnitude is so small (i.e., less than 0.03) it is hard to say that they are important contributors to the overall uncertainty in the calculation. The difference peaks at 500 days; in the following table, we present the significant (i.e., $> 10^{-3}$) interaction sensitivity coefficients for the PCE4 case.

Table 6.4.1: Important interaction coefficients for Doppler temperature

Interaction variables	Sensitivity index
FTX, CRP	8.324E-03
FTC, CRP	1.166E-03
FTX, SWL	6.684E-03
CRP, SWL	8.129E-03
FTX, CRP, SWL	1.216E-03

We can see from this table that the most important interactions are between the thermal expansion, cladding creep, and fuel swelling models. This is an unsurprising result, since these are the three models considered that contribute directly to the manner in which the fuel-cladding gap closes. Simultaneous variability in these models contributes to the variance in the temperature calculation because they all directly affect the fuel-clad temperature drop, directly influencing the Doppler temperature. The other interaction effects have small contributions. In

any event, the magnitude of these effects is quite small, so while interaction effects to occur and contribute to output variance in a manner in which linear coefficients cannot account for, they are not generally significant.

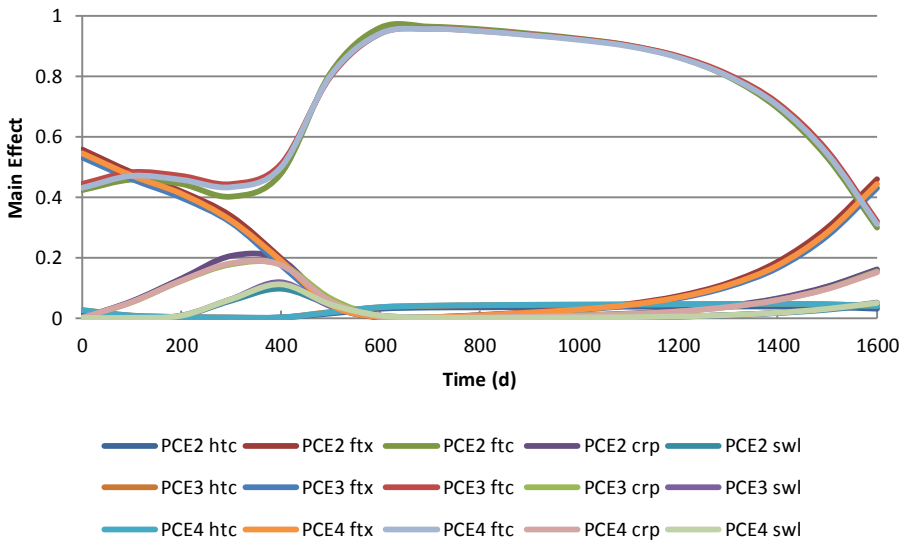


Figure 6.4.9: Main effect index for reactivity

In Figure 6.4.9, we show the ME index for reactivity. Again, the expansion orders return very similar values as the core depletes, which indicates that a linear model is sufficient. We see clearly here effects that the sampling-based VBD analysis showed in a non-converged way; namely, that the relative contributions to the reactivity variance across the models are very similar to the Doppler temperature throughout most of the depletion. However, as the depletion nears the end, the history effects of plutonium buildup in the core contribute to decreasing the relative importance of the thermal conductivity, and increasing the influence of models which have a strong reactivity influence early in life. The thermal expansion model is the most significant contributor here, with cladding creep playing a secondary role. The other models are generally insignificant. In this way, we can see the propagation of the uncertainty in irradiation conditions early in life to the end-of-life reactivity uncertainty. However, as we noted in Section 6.3, the reactivity standard deviation in this region ranges from ~15-30 pcm; from a core operation standpoint, this is not significant.

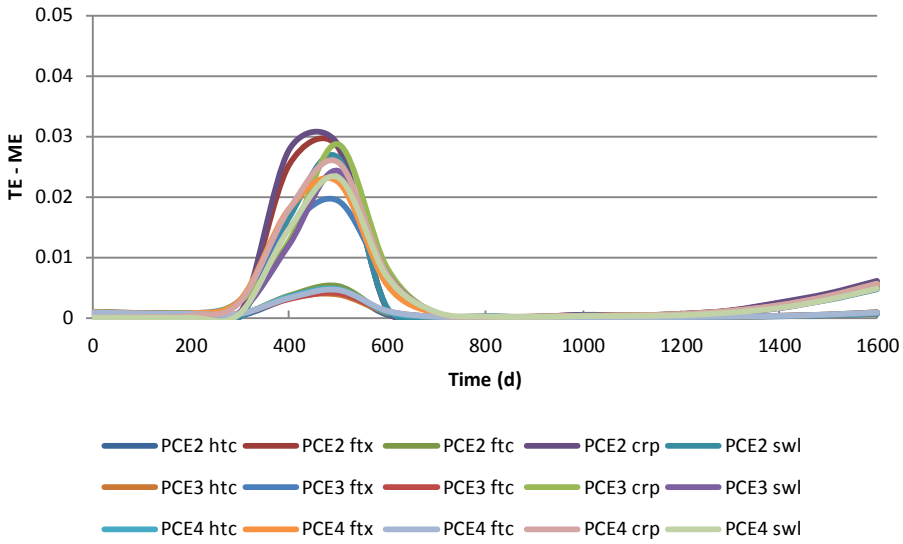


Figure 6.4.10: Input interaction metric for reactivity

In taking the difference between TE and ME indices, we see a similar peak in the importance of input interactions to variance in the region where the gap is closing. Although again the effect is minor, we present in the following table the specific input interactions that are of note.

Table 6.4.2: Important interaction coefficients for reactivity

Interaction variables	Sensitivity index
HTC, FTX	1.17E-03
HTC, CRP	1.08E-03
FTX, CRP	9.95E-03
FTC, CRP	1.60E-03
HTC, SWL	1.25E-03
FTX, SWL	7.74E-03
FTC, SWL	1.20E-03
CRP, SWL	1.00E-02
FTX, CRP, SWL	1.75E-03

As with the Doppler temperature, there are some variable interactions which contribute relatively significantly to the reactivity variance. The most important effects are the interactions between the creep and swelling models and creep and fuel thermal expansion models. Again, these are the models that contribute the most to the behavior of the fuel-cladding gap, affecting fuel surface temperature, which is at this point in the depletion the most important influence over the nuclear behavior of the system.

6.4.3 Comparisons between methods

Before concluding this sensitivity analysis, we make some general remarks on the findings, and in particular the comparisons between the different methods we employed. First, we saw that sampling-based VBD analysis can be employed to evaluate the sensitivity of the key system outputs for this problem, and that comparisons between the VBD indices, correlation coefficients, and VBD indices calculated by expansion methods showed general agreement. However, the effect of small sample size was clearly obvious in these evaluations; the calculated VBDs showed great variability across sample sets, and there were some instances where values outside the defined interval of ME and TE were computed. We expect that using larger sample sizes would help mitigate this issue, but the drawback is that they are extremely expensive for this problem due to the time required to execute the FRAPARCS depletion. On the other hand, we saw that the VBD analysis based on stochastic expansions required relatively few evaluations to achieve consistency across different orders; while the sampling-based calculations required 350 calculations each (and we performed 5 sets of samples each for RS and LHS), a stochastic expansion can be formed and sufficient data extracted using only 32 (for the linear expansion) or 243 (for quadratic) calculations. The benefit to practical analysis is obvious for this type of problem.

6.5 Conclusions

In this Chapter, we have applied the methods described in Chapter 5 to uncertainty and sensitivity analysis of the pin cell depletion problem using FRAPARCS. First, we examined the data which was used by the FRAPCON developers to qualify the models and generate standard errors for use in uncertainty calculations. Finding that these data did not fit standard distributions, we generated histogram distributions which we used in DAKOTA.

We then applied sampling- and stochastic expansion-based uncertainty methods to the pin cell depletion. The results of the uncertainty analysis indicate that the mean and standard deviation closely approximate a 1300 sample reference solution with the application of 200 LHS calculations. These results can also be obtained using stochastic expansion methods. For this problem, PCE and SC expansions generated equivalent polynomials, and therefore the statistical outputs generated by them were the same. We discovered that comparable statistics to the

reference solution can be generated with as few as 32 function evaluations using stochastic expansion. We also attempted to apply sparse grid methods to the stochastic expansions, and found that they were not able to converge; this is likely due to the discontinuous nature of the input distributions and the fact that the sparse grids did not sufficiently cover the sample space. We found that the most significant uncertainties (as could be guessed from the Chapter 4 analysis) were in the Doppler temperature and reactivity; power shape and reactivity coefficients were minimally affected (i.e., the variance and departure from nominal values was very small).

In the follow-up sensitivity analysis, we showed that only using 50 samples to evaluate the main- and total-order effect indices was not sufficient to properly converge them; however, the general trend in these parameters though the depletion was consistent with analysis based on correlation coefficients. We also saw the benefit of using an analytical method to generate the expansion coefficients; a linear expansion appears to be sufficiently accurate to resolve the sensitivity information for the problem in question. From a physics standpoint, the 5 uncertain models competed for importance early in depletion while the fuel-clad gap was closing; the most important model was the fuel thermal conductivity, followed by thermal expansion, cladding creep, and fuel swelling. Under the conditions of this analysis, the heat transfer coefficient was not a major contributor to the variance of the Doppler reactivity. We also saw that input interactions were not a major contributor to the variance in the output values, although they did manifest in a minor way during gap closure.

With this, we conclude our analysis. In the next chapter, we will draw some general conclusions from the results of this work, and make recommendations as to which paths should be followed in the future related to this topic.

CHAPTER 7

Conclusions and Future Work

7.1 Summary

Within this thesis, we have first demonstrated the need to include the thermo-mechanical effects of reactor fuel in a coupled neutronics and thermal-hydraulics depletion calculation. We showed through a sensitivity study that the method used in PARCS to account for fuel-clad gap closure and resulting temperature drop can have significant influence over the reactivity predicted by the code throughout core life.

We then discussed prominent approaches to model nuclear reactor fuel and core components, as well as applications of these methods. The various methods of multi-physics calculations and their importance to the fuel depletion problem were also discussed, including prominent examples of applications in coupled neutronics/thermal-hydraulics and neutronics/fuel thermo-mechanics. We found that, while advanced methods are currently being developed for detailed, transport-level multiphysics behavior reactor components, there was a lack of application of the current-generation reactor analysis tools to the coupled depletion problem. In most cases, current methods rely on the pre-generation of gap conductivity tables in core simulators; while these methods have proven effective in industry, a true coupled calculation did not exist. We also discussed several representative uncertainty and sensitivity analysis methods and applications.

Next, the methods used in the codes for this work were described. FRAPCON and PARCS/PATHS were used to model the fuel performance and neutronics/thermal-hydraulics, respectively. We discussed the algorithms and models included in these codes, and the algorithm

modifications required to apply them to a coupled depletion problem. The code coupling is achieved through a Python script called FRAPARCS. The coupling method was through file transfer where PARCS/PATHS is the master and FRAPCON is the slave. To ensure convergence in the power distribution and fuel temperature, a fixed-point iteration scheme was chosen.

Then, we discussed the application of FRAPARCS to two depletion problems: a pin cell and a mini-core with fuel reloads. Comparisons with PARCS/PATHS standalone computations showed that the fuel temperature and resultant reactivity deviations relative to the coupled calculation. The magnitude of these deviations could be as much as 50 K and 300 pcm in Doppler temperature and reactivity, respectively. Other neutronics figures of merit (such as peaking factors and reactivity coefficients) were minimally affected. We showed in these comparisons that PARCS/PATHS is not capable of representing these physical effects in stand-alone mode. The primary contributor to the deviations was the closing of the fuel-clad gap early in the depletion, and the degradation of the fuel thermal conductivity later in life. The gap effects were most prominent in fresh fuel situations, while in the reload cores, the magnitude in reactivity deviation was roughly half as large. We then explored eight FRAPCON models accessible to the user for biasing, and their relative effects on the core depletion as they were changed. This sensitivity analysis revealed four models that were particularly influential on the Doppler reactivity. In addition, we showed that the single-phase heat transfer coefficient can have a minor but significant effect on the depletion.

With the results of these studies, we applied uncertainty and sensitivity analysis methods to the coupled pin cell depletion problem. We first described sampling-based and stochastic expansion-based methods to determine response distributions and statistics. We also examined methods which can be used to evaluate sensitivity information; in particular, we focused on Variance-Based Decomposition methods, in order to gauge the possible importance of input interactions on the neutronics outputs. Then, we applied these methods to the pin cell depletion. First, input distributions for the uncertainty variables were generated based on data. We explored the use of random and Latin Hypercube sampling to the problem, and found that the mean and standard deviation in the reactivity relative to the nominal case and the Doppler temperature could be well-represented relative to a reference solution by 200 LHS samples. We also applied Polynomial Chaos Expansion and Stochastic Collocation on first, second, and third-order tensor grids. We showed that the expansions between these methods were equivalent, and that the first-

order model predicted nearly the same values for output statistics as the higher-order models. The expansion results also compared well with the random sampling-based results. The mean values in temperature and reactivity deviation in general agreed well with the reference solution. The standard deviations in Doppler temperature were found to range from 14 K to 17 K at beginning and end of life, respectively, and the standard deviation in reactivity was found to vary from ~60 pcm to 14 pcm at beginning and end of life, respectively. As expected based on the Chapter 4 results, the uncertainties in the peaking factors and reactivity coefficients were found to be almost negligible.

Following the estimates of the output statistics, we employed VBD methods to evaluate the relative importance of each uncertain input on the response variance. We found that applying VBD via a sampling-based approach with only 50 samples was insufficient; although general trends were captured, there was far too much noise in the data to draw definitive conclusions. Then, we applied VBD with the stochastic expansions. This method proved to work well, and was able even with the first-order model to capture most of the sensitivity in the data. We found that the fuel models competed for prominence in relative performance during the period of gap closure, but that once thermal contact was established between the fuel and the cladding material the thermal conductivity contributed the most to the variance in temperature and reactivity. The reactivity sensitivity study also showed that, near the very end in life, the conductivity contribution decreased significantly; this is likely due to the influence of plutonium dampening out the effects.

7.2 Suggestions for further development

While this work investigates some important and unique questions, there are still several areas that should be addressed in follow-up study. First of all, as we expressed several times within this text, the coupling method is not very efficient. The method here leads to long execution times even for small test problems, and restricts our ability to perform evaluations on larger, more realistic nuclear systems. We have shown that there is a real benefit to including fuel physics in diffusion-based depletion calculations, even outside of uncertainty considerations. The first suggestion is to further develop the coupling method to reduce the execution time. This could be done by embedding FRAPCON within PARCS/PATHS as a subroutine; one of the

challenges to accomplishing this is FRAPCON's coding and architecture. With appropriate support, however, it should be possible.

The second recommendation is to expand the class of problems that has been considered. Within this work, we used a 15x15 PWR assembly design as the reference; the effect of other geometries (and more representative cores) should be evaluated. Additionally, the potential for using pin power reconstruction methods to evaluate fuel performance on the pin level should be further explored. The primary issue with this is the possibility of evaluating cladding outer surface temperatures at somewhat inappropriate temperatures (PATHS models each assembly as a single channel, as opposed to subchannel-based methods that model the space between each fuel rod explicitly); however, based on the relative importance of heat transfer coefficient, this may not be a limiting factor. Further exploration is warranted.

Within the context of uncertainty analysis, we limited the number of uncertain variables to five, all of which were directly associated with different phenomenological models. However, there are a number of potential uncertainties that have not yet been evaluated, such as the rod geometry. Additional uncertainty in the neutronics results may be possible, depending on how they affect the behavior of the fuel-clad gap. Also, work should be done to integrate the uncertainty approaches described here within the context of neutronic uncertainty in general. As we discussed in the Literature Review, much of the core-level uncertainty work comes though the collapsing of detailed cross sections into two-group cross sections appropriate for diffusion-based simulators. The effect of fuel mechanics in the cross section generation process, and the relative importance of the models we considered here to the uncertainties introduced by including cross section covariance data, should be further studied.

7.3 Conclusions

The work performed here to develop the FRAPARCS code and apply it to the uncertainty and sensitivity analysis of a PWR fuel pin depletion represents a step forward in multi-physics depletion analysis. However, much work needs to be done to integrate these concepts into production-level tools, but the demonstrations provided here show that the effects of fuel performance on reactor core depletion are important, and merit further study.

APPENDIX A

FRAPCON Uncertainty Models

A.1: Fuel Thermal Conductivity

The model for fuel thermal conductivity in FRAPCON is derived from the Nuclear Fuel Industries (NFI) model with modifications. This model is a function of temperature and burnup. For 95% theoretical density (TD) UO_2, it is given by:

$$k_{95} = \frac{1}{A + a \cdot gad + BT + f(Bu) + (1 - 0.9\exp(-0.04Bu))g(Bu)h(T)} + \frac{E}{T^2}\exp\left(-\frac{F}{T}\right) \qquad \text{(A.1)}$$

where:

k_{95}	=	thermal conductivity of 95% TD fuel (W/m-K)
T	=	temperature (K)
Bu	=	burnup (GWd/MTU)
$f(Bu)$	=	effect of fission products in crystal matrix (solution)
	=	$0.00187 \cdot Bu$
$g(Bu)$	=	effect of irradiation effects
	=	$0.038 \cdot Bu^{0.28}$
$h(T)$	=	temperature dependence of annealing on irradiation defects
	=	$\frac{1}{1 + 396\exp(-\frac{Q}{T})}$
Q	=	temperature dependence parameter
	=	6380 K
A	=	0.0452 (m-K/W)
a	=	1.1599
gad	=	weight fraction of gadolinia
B	=	2.46E-4 (m-K/W/K)
E	=	3.5E9 (W-K/m)
F	=	16361 (K)

In FRAPCON-3.4, the thermal conductivity is calculated using the above model. For fuel that is not 95% TD, the following adjustment is applied:

$$k_d = 1.0789 k_{95} \left[\frac{d}{1.0 + 0.5(1 - d)} \right] \quad \text{(A.2)}$$

where k_d is the thermal conductivity at fraction of TD d, and k_{95} is the thermal conductivity predicted by the model above.

A.2: Fuel Thermal Expansion

The equation for temperature-related strain in UO_2 fuel given in NUREG/CR-7024 is as follows:

$$\frac{\Delta L}{L_0} = K_1 T - K_2 + K_3 exp\left(-\frac{E_D}{kT}\right) \quad \text{(A.3)}$$

where

$\frac{\Delta L}{L_0}$	=	linear strain caused by thermal expansion (equal to zero at 300K)
T	=	temperature (K)
E_D	=	energy of formation of a defect
	=	1.32 x 10^{-19} J
k	=	Boltzmann's constant
	=	1.38 x 10^{-23} J/K
K_1	=	9.8 x 10^{-6} K^{-1}
K_2	=	2.61 x 10^{-3} (unitless)
K_3	=	0.316 (unitless)

A.3: Fuel Thermal Swelling

The fuel swelling model in FRAPCON-3.4 is given by the following equations. For burnup less than 80 GWd/MTU:

$$soldsw = 2.315 \times 10^{-23} bus(1 + 0.08 sigsw) \quad \text{(A.4)}$$

For burnup greater than 80 GWd/MTU:

$$soldsw = 3.211 \times 10^{-23} bus(1 + 0.16 sigsw) \quad \text{(A.5)}$$

with

$soldsw$	=	fractional volume change due to solid fission products (m^3/ m^3)
$sigsw$	=	user-defined parameter that incrementally changes error (this is one of the input uncertainty parameters described above)
bus	=	fuel burnup during timestep
	=	$2.974 \times 10^{10} fdens(bu - bul)$
$fdens$	=	initial pellet density (kg/m^3)
bu	=	burnup at end of timestep (MWs/kgU)
bul	=	burnup at end of previous timestep (MWs/kgU)

A.4: Fission Gas Release

As discussed in Chapter 3, the Forsberg-Massih model is chosen as the FGR model for this work, due to its superior agreement with steady-state measurements. However, both this and the FRAPFGR models have similar theoretical structure and form. The basic model for fission gas release in the Forseberg-Massih and the FRAPFGR models is as follows:

$$\frac{dC}{dt} = D(t)\Delta_r C(r,t) + \beta(t) \tag{A.6}$$

where

$C(r,t)$	=	gas concentration
$\beta(t)$	=	gas production (assumed uniform within grain)
Δ_r	=	Laplacian operator in spherical geometry
	=	$\frac{d^2}{dr^2} + \frac{2}{r}\frac{d}{dr}$
$D(t)$	=	diffusion coefficient
t	=	time

This equation describes the diffusion of gas within a spherical fuel grain. The boundary conditions and the correlation for the diffusion coefficient depend on the specific model chosen. It is worth noting that no matter which model is chosen, the diffusion coefficient is a strong function of temperature. The same process for calculating fission gas release is used by all FGR models in FRAPCON:

1. Using local power history, determine source term (i.e., $\beta(t)$)
2. Using local temperatures and burnup, compute *D(t)*

3. Compute the approximate change in concentration within the fuel grains and on the fuel grain boundaries using a 4-term approximation to the integration kernel of the above equation
4. The fraction of gas released is determined based on the local gas saturation levels on the grain boundaries; any fission gas accumulated on the boundaries beyond the saturation limit is assumed to be released into the fuel-cladding gap

In order to compute fission gas release, one must know the source term. Currently, the model FRAPCON-3.4 uses to compute fission gas production during a given timestep at an axial location *z* is:

$$GPT(z) = \frac{BU(z)VF(z)}{100A_v}(PR_{krypton} + PR_{helium} + PR_{xenon}) \tag{A.7}$$

where

$GPT(z)$	=	gas produced at axial location *z* (mole)
$BU(z)$	=	accumulated burnup at *z* (fissions/cm^3)
$VF(z)$	=	fuel volume about *z* (cm^3)
A_v	=	Avogadro's number
PR	=	fission gas production rate (atoms/100 fissions) for Krypton, Helium and Xenon

Within FRAPCON-3.4, this equation is manipulated to be consistent with $\beta(t)$ appearing in the above diffusion equation Eq. (3.4).

Fission gas release is a strongly temperature-dependent formula. However, as implemented in FRAPCON-3.4, there is also an enhancement multiplier that increases the diffusion coefficient as burnup increases. The key point relevant to this work is that thermal diffusion only occurs with temperatures greater than 1381 K; in the depletion calculations considered in this work, the fuel did not reach this threshold. Therefore, only athermal release occurred. As discussed in Chapter 3, the FGR bias is applied to the enhancement factor in *D* of Equation A.6, which is only activated in high-temperature situations.

A.5: Cladding Creep

The models for thermally- and irradiation-induced creep of the cladding material are given by the following, respectively:

$$\dot{\varepsilon}_{th} = A\frac{E}{T}\left(\sinh\frac{a_i\sigma_{eff}}{E}\right)^n \exp\left(\frac{-Q}{RT}\right) \quad \text{(A.8)}$$

$$\dot{\varepsilon}_{irr} = C_0\phi^{C_1}\sigma_{eff}^{C_2} f(T) \quad \text{(A.9)}$$

where

$\dot{\varepsilon}_{irr}$	=	irradiation creep strain rate (in/in/hr)
$\dot{\varepsilon}_{th}$	=	thermal creep strain rate (in/in/hr)
ϕ	=	fast neutron flux (>1 MeV) (neutrons/cm^2-s)
σ_{eff}	=	effective stress (MPa)
$A, E, n, Q, C_0, C_1, C_2$	=	empirical coefficients depending on temperature, flux, and cladding type (stress-relieved annealed (SRA) or fully recrystallized annealed (RXA)
$f(T)$	=	empirical temperature-dependent function depending on cladding type (SRA or RXA)
T	=	temperature (K)
R	=	universal gas constant (0.008314 kJ/mol-K)

At the end of a timestep, the thermal and irradiation creep rates based on the conditions within the node are computed, and then combined to determine the total creep strain.

A.6: Cladding Axial Growth

The model for irradiation-induced cladding axial growth is based on the EPRI model published in 1982 which accounts for high neutron fluences. It is of the form:

$$ax = C\Phi^n \quad \text{(A.10)}$$

where

ax	=	Axial growth increment (m/m)
Φ	=	Fast neutron fluence (n/cm^2) (E > 1.0 MeV)
C, n	=	Material-dependent fitting parameters

Table A.1: Fitting parameters for axial growth model

Parameter	Zircaloy-4	ZIRLO	M5
C	2.18x10^{-21}	9.7893x10^{-25}	7.013x10^{-21}
n	0.845	0.98239	0.81787

A.7: Cladding Corrosion

The corrosion behavior modeled by FRAPCON-3.4 takes into account the cladding material and environment. For PWRs, up to a specific material-dependent (i.e., Zircaloy-4, ZIRLO, or M5) transition thickness (typically on the order of 5 microns), a cubic rate equation is employed. When integrated, the corrosion thickness is given by:

$$s_{i+1} = \left(3A \exp\left\{-\frac{Q_1}{RT_1}\right\}(t_{i+1} - t_i) + s_i^3\right)^{(1/3)} \tag{A.11}$$

where

$i, i+1$	=	Time step level
s	=	oxide thickness (m)
A	=	empirical constant dependent on material (m^3/day)
Q_1	=	empirical constant dependent on material (cal/mol)
R	=	constant (1.98 cal/mol-K)
T_1	=	oxide-metal interface temperature (K)
t	=	time (days)

Once the transition oxide thickness is reached, the growth is assumed to obey a flux dependent linear rate law, with the rate constant given by an Arrhenius function of oxide-metal interface temperature. In the implementation of this equation in FRAPCON-3.4, the nodal oxide weight gain is computed, rather than the oxide thickness itself. This weight gain is then easily converted into thickness. The equation for oxide weight gain is:

$$\Delta w_{i+1} = \Delta w_i + \frac{RT_0^2\lambda}{\gamma Q_2 q''}\ln\left[1 - \frac{\gamma Q_2 q''}{RT_0^2\lambda}k_0 \exp\left\{-\frac{Q_2}{RT_0}\right\}\exp\left\{\frac{\gamma Q_2 q'' \Delta w_i}{RT_0^2\lambda}\right\}(t_{i+1} - t_i)\right]^{-1} \tag{A.12}$$

which is then converted to thickness using:

$$s = \frac{\Delta w \gamma}{100}$$

In these equations,

$i, i+1$	=	Time step level
s	=	oxide thickness (m)
Δw	=	weight gain (g/cm^2)
R	=	constant (1.98 cal/mol-K)
T_0	=	oxide-water interface temperature (K)

λ	=	oxide thermal conductivity (W/cm/K)
γ	=	oxide specific volume (0.6789 cm^3/g)
Q_2	=	empirical constant dependent on material (cal/mol)
q''	=	heat flux (W/cm^2)
k_0	=	$11863+3.5\times10^4(1.91\times10^{-15}\Phi)^{0.24}$ (g/cm^2-day)
Φ	=	fast neutron flux (E>1 MeV) (n/cm^2-s)
t	=	time (days)

The specific values for Q_1, Q_2, and transition thickness vary depending upon the specific material in question.

A.8: Cladding Hydrogen Pickup

Cladding corrosion is an oxidation process described by the following reaction:

$$Zr + 2H_2O \rightarrow ZrO_2 + 2H_2$$

In addition to zirconium oxide formed on the cladding surface, some amount of free hydrogen is produced. Much of this hydrogen is swept away into the coolant, but some amount is absorbed by the cladding material. This is a potential issue because hydrides within fuel cladding can cause embrittlement, making the material more susceptible to cracking during transient operation.

As with corrosion modeling, the specifics of cladding hydrogen absorption depend upon the material in question. For PWR materials, a fixed, experimentally determined hydrogen pickup fraction is used; based on the corrosion of the material, a certain percentage is assumed to be absorbed into the cladding metal. This, in effect, ties the cladding corrosion and hydrogen content tightly together. Table A.2 gives the hydrogen pickup fractions for the PWR cladding materials modeled in FRAPCON-3.4:

Table A.2: Hydrogen pickup fractions for PWR cladding materials

Material	Pickup Fraction
Zircaloy-4	0.15
ZIRLO	0.125
M5	0.10

APPENDIX B

FRAPARCS Example Scripts and Flowchart

B.1: FRAPARCS for pin cell depletion

```
#!/usr/bin/python
#This is the FRAPCON/PARCS driver script. It will be called from DAKOTA
#to perform sensitivity and uncertainty estimations.
#Author: Andrew Bielen
#Date:   November 20 2013
from sys import exit
from subprocess import call
from time import sleep
import fp_util
import numpy as np
import scipy
import os

#Make changes here
opsys = 'windows'     #choose 'windows' or 'linux'
parcfi = 'pwr_pin.inp'
fraptmp = 'TPbase.in.template'
frapout = 'TPbase.out'
f2pfile = 'f2p.dat'
f2pflag = 'f2pready.flg'
frapplot = 'TPbase.plot'
parcpath = 'C:\\PARCS\\v32m10frap_v3\\run\\parcs_ivfr.exe'
frappath =
'C:\\frapcon3_5m\\frapcon3_5m\\frapcon3_5m\\Release\\frapcon3_5m.exe'
frapid = ['00001999']
tmst = [0.01, 10., 20, 30., 40., 50., 60., 70., 80., 90., 100.,
        110., 120., 130., 140., 150., 160., 170., 180., 190., 200.,
        210., 220., 230., 240., 250., 260., 270., 280., 290., 300.,
        310., 320., 330., 340., 350., 360., 370., 380., 390., 400.,
        410., 420., 430., 440., 450., 460., 470., 480., 490., 500.,
        510., 520., 530., 540., 550., 560., 570., 580., 590., 600.,
        610., 620., 630., 640., 650., 660., 670., 680., 690., 700.,
        710., 720., 730., 740., 750., 760., 770., 780., 790., 800.,
        810., 820., 830., 840., 850., 860., 870., 880., 890., 900.,
```

```
       910., 920., 930., 940., 950., 960., 970., 980., 990., 1000.,
        1010., 1020, 1030., 1040., 1050., 1060., 1070., 1080., 1090., 1100.,
        1110., 1120., 1130., 1140., 1150., 1160., 1170., 1180., 1190., 1200.,
        1210., 1220., 1230., 1240., 1250., 1260., 1270., 1280., 1290., 1300.,
        1310., 1320., 1330., 1340., 1350., 1360., 1370., 1380., 1390., 1400.,
        1410., 1420., 1430., 1440., 1450., 1460., 1470., 1480., 1490., 1500.,
        1510., 1520., 1530., 1540., 1550., 1560., 1570., 1580., 1590., 1600.]
x  = [0.0000000000, 0.159130435, 0.31826087, 0.477391304, 0.636521739, 0.795652174, 0.954782609,
1.113913043,
      1.273043478, 1.432173913, 1.591304348, 1.750434783, 1.909565217, 2.068695652, 2.227826087,
      2.386956522, 2.546086957, 2.705217391, 2.864347826, 3.023478261, 3.182608696, 3.34173913,
3.500869565,3.66]
xt = [0.0, 0.1525, 0.305, 0.4575, 0.61, 0.7625, 0.915, 1.0675, 1.22, 1.3725, 1.525, 1.6775,
      1.83, 1.9825, 2.135, 2.2875, 2.44, 2.5925, 2.745, 2.8975, 3.05, 3.2025, 3.355, 3.5075, 3.66]
na = 24

#Leave below unchanged
if opsys == 'windows':
       copycmd = 'copy'
       delcmd = 'del'
       movecmd = 'move'
       parcscmd = ['start','%COMSPEC%','/k',parcpath, parcfi]
elif opsys == 'linux':
       copycmd = 'cp'
       delcmd = 'rm'
       movecmd = 'mv'
       parcscmd = [parcspath,parcfi,'$']
else:
       exit('bad OS specification')

#Interpret input data from above
nstp = len(tmst)
iterhist = np.zeros(nstp)
nfrp = len(frapid)
for i in range(len(x)):
       x[i] = str(x[i])
for i in range(len(xt)):
       xt[i] = str(xt[i])
#Spawn input files for each frapcon case
for id in frapid:
       frapfi_old = 'TPbase'+id+'_old.in'
```

```
        frapfi_new = 'TPbase'+id+'.in'
        call([copycmd,fraptmp,frapfi_old],shell=True)
        call([copycmd,fraptmp,frapfi_new],shell=True)

call([delcmd,'nextstep.txt'],shell=True)
call([delcmd,'repeat.txt'],shell=True)
call([delcmd,'*.flg'],shell=True)

#Get PARCS rolling first

call(parcscmd,shell=True)

#March through time with FRAPCON

i = 1
iter = 1
move = False
stay = False
ready = False
for tm in tmst:
        print 'executing timestep ', i, ' (', tm, ' days)'
        while not move:
                move = os.path.exists('nextstep.txt')
                stay = os.path.exists('repeat.txt')
                if move:
                        call([delcmd,'nextstep.txt'],shell=True)
                        stay = False
                        move = False
                        break
                elif stay:
                        print 'executing f/p iteration ', iter
                        for id in frapid:
#                               flg = 'p2f'+id+'.flg'
                                dat = 'p2f'+id+'.dat'
                                inp = 'TPbase'+id+'.in'
                                old = 'TPbase'+id+'_old.in'
                                f2p = 'f2p'+id+'.dat'
#                               call([delcmd,flg],shell=True)
                                call([delcmd,inp],shell=True)
                                call([copycmd,old,inp],shell=True)
                                #call the substitution routine
```

```
                                fp_util.frap_sub(dat,inp,i,na,x,xt,tmst[i-1])
                                #call frapcon
                                call([frappath,inp],shell=True)
                                call([delcmd,dat],shell=True)
                                call([movecmd,f2pfile,f2p],shell=True)
                        call([delcmd,'repeat.txt'],shell=True)
                        print 'frapcon is done, writing f2p flag file'
                        f2pflg = open(f2pflag,'w')
                        f2pflg.close()
                        stay = False
                        iter = iter + 1
                else:
                        print 'at timestep ', i, ' waiting for parcs to make up its mind'
                        sleep(1)
        print 'PARCS and FRAPCON have finished timestep ', i, 'with ', iter, ' iterations'
        for id in frapid:
                old = 'TPbase'+id+'_old.in'
                new = 'TPbase'+id+'.in'
                call([delcmd,old],shell=True)
                call([movecmd,new,old],shell=True)
        iterhist[i-1] = iter
        i = i + 1
        iter = 1
        sleep(1)
#rerun last time step after turning full output on
for id in frapid:
        inp = 'TPbase'+id+'_old.in'
        fp_util.frap_printsub(inp)
        print 'running frapcon id, ', id, ' for printing'
        call([frappath,inp],shell=True)
        call([movecmd,frapout,'TPbase'+id+'.out'],shell=True)
        call([movecmd,frapplot,'TPbase'+id+'.plot'],shell=True)
#call([delcmd,'*.dat'],shell=True)
#print iteration history
itfile = open('iterhist.txt','w')
for iter in iterhist:
        print>>itfile, str(iter)
itfile.close()
```

B.2: FRAPARCS for Multicycle Depletion

```
#!/usr/bin/python
#This is the FRAPCON/PARCS driver script. It will be called from DAKOTA
#to perform sensitivity and uncertainty estimations.
#This is the multicycle adaptation.
#Author: Andrew Bielen
#Date:   April 3 2014
#
#Core positions
#       1   2   3   4   5
#       6   7   8   9  10
#      11  12  13  14  15
#      16  17  18  19  20
#      21  22  23  24  25
#
#Cycle 1 design (8 Hi, 5 M, 2 Lo)
#       5   2   8   2   5
#       2   8   5   8   2
#       8   5   8   5   8
#       2   8   5   8   2
#       5   2   8   2   5
#
#Cycle 2 design (8 Hi, 5 M)
#       8   5   8   5   8
#       5   8   8   8   5
#       8   8   8   8   8
#       5   8   8   8   5
#       8   5   8   5   8
#
#Cycle 3 design (8 Hi)
#       8   8   8   8   8
#       8   8   8   8   8
#       8   8   8   8   8
#       8   8   8   8   8
#       8   8   8   8   8
#
#Shuffle map - 0 means fresh, position in prev cycle o/w
#   Applied both at EOC1 and EOC2
#       7   8   0  14   9
#       1   0   3   0   5
```

```
#       0  11   0  15   0
#      21   0  23   0  25
#      17  12   0  18  19
#

from sys import exit
from subprocess import call
from time import sleep
import fp_util
import numpy as np
import scipy
import os

#Make changes here
opsys = 'windows'     #choose 'windows' or 'linux'
parcfi = ['pwr_5x5CPL_C1.inp','pwr_5x5CPL_C2.inp','pwr_5x5CPL_C3.inp']
#parcfi = ['pwr_5x5CPL_C2.inp','pwr_5x5CPL_C3.inp']
#parcfi = ['pwr_5x5CPL_C3.inp']
##cycle index
icyc = 1
#icyc = 2
#icyc = 3
shuffi = ['pwr_5x5CPL_1to2.inp','pwr_5x5CPL_2to3.inp']
cycid = ['CYC1','CYC2','CYC3']
frpre = 'sur'
tmpsuf = '.in.template'
insuf = '.in'
outsuf = '.out'
pltsuf = '.plot'
f2pfile = 'f2p.dat'
f2pflag = 'f2pready.flg'
parcpath = 'C:\\PARCS\\v32m10frap_v3\\run\\parcs_ivfr.exe'
frappath = 'C:\\frapcon3_5m\\frapcon3_5m\\frapcon3_5m\\Release\\frapcon3_5m.exe'
shufpath = 'C:\\fraparcs\\shuffler\\shuffler\\Release\\shuffler.exe'
#Fuel rod groupings for initial cycle
frapid1 = ['00003999','00007999','00009999','00011999','00013999',
           '00015999','00017999','00019999','00023999']
frapid2 = ['00001999','00005999','00008999','00012999',
           '00014999','00018999','00021999','00025999']
frapid3 = ['00002999','00004999','00006999','00010999',
           '00016999','00020999','00022999','00024999']
```

```
enr = ['3.117','2.573','1.868']
ftd = ['91.9','92.9','93.5']
gtk = ['0.000108','0.00009525','0.00009525']
shufmap = ['007','008','000','014','009','001','000','003','000',
           '005','000','011','000','015','000','021','000','023',
           '000','025','017','012','000','018','019']
order = ['001','002','003','004','005','006','007','008','009','010',
         '011','012','013','014','015','016','017','018','019','020',
                '021','022','023','024','025']
discard = ['00002999','00004999','00006999','00010999','00013999',
           '00016999','00020999','00022999','00024999']
frapgrp = [frapid1,frapid2,frapid3]
frapid = frapid1 + frapid2 + frapid3
#print frapid
tmst = [0.01, 10., 20, 30., 40., 50., 60., 70., 80., 90., 100.,
        110., 120., 130., 140., 150., 160., 170., 180., 190., 200.,
        210., 220., 230., 240., 250., 260., 270., 280., 290., 300.,
        310., 320., 330., 340., 350., 360., 370., 380., 390., 400.,
        410., 420., 430., 440., 450., 460., 470., 480., 490., 500.,
        510., 520., 530., 540.]
#tmst = [0.01,10]
x  = [0.0000000000, 0.159130435, 0.31826087, 0.477391304, 0.636521739, 0.795652174, 0.954782609,
1.113913043,
      1.273043478, 1.432173913, 1.591304348, 1.750434783, 1.909565217, 2.068695652, 2.227826087,
      2.386956522, 2.546086957, 2.705217391, 2.864347826, 3.023478261, 3.182608696, 3.34173913,
3.500869565,3.66]
xt = [0.0, 0.1525, 0.305, 0.4575, 0.61, 0.7625, 0.915, 1.0675, 1.22, 1.3725, 1.525, 1.6775,
      1.83, 1.9825, 2.135, 2.2875, 2.44, 2.5925, 2.745, 2.8975, 3.05, 3.2025, 3.355, 3.5075, 3.66]
na = 24

#Leave below unchanged
if opsys == 'windows':
       copycmd = 'copy'
       delcmd = 'del'
       movecmd = 'move'
elif opsys == 'linux':
       copycmd = 'cp'
       delcmd = 'rm'
       movecmd = 'mv'
       parcscmd = [parcspath,parcfi,'$']
else:
```

```
        exit('bad OS specification')

#Interpret input data from above
template = frpre+tmpsuf
nstp = len(tmst)
nfrp = len(frapid)
nenr = len(enr)
ncyc = len(parcfi)
cyclen = tmst[nstp-1]
totstp = nstp * ncyc
iterhist = np.zeros(totstp)
for i in range(len(x)):
        x[i] = str(x[i])
for i in range(len(xt)):
        xt[i] = str(xt[i])

#Clean up the directory
call([delcmd,'nextstep.txt'],shell=True)
call([delcmd,'repeat.txt'],shell=True)
call([delcmd,'*.flg'],shell=True)
call([delcmd,'*.dat'],shell=True)
call([delcmd,'*'+insuf],shell=True)

if icyc == 1:
#Spawn input files for each frapcon case
#Consider all individual enrichments - replace wtp flag with actual value
#Also considers differences in theoretical density and gap thickness
        j = 0
        for grp in frapgrp:
                jind = str(j)
                templateenr = template+'.'+jind
                call([copycmd,template,template+'.'+jind],shell=True)
                fileold = open(templateenr).read()
                fileold = fileold.replace('enrch=wtp','enrch='+enr[j])
                fileold = fileold.replace('thkgap=gtk','thkgap='+gtk[j])
                fileold = fileold.replace('den=ftd','den='+ftd[j])
                filenew = open(templateenr,'w')
                filenew.write(fileold)
                filenew.close()
                for id in grp:
                        fiold = frpre+id+'_old'+insuf
```

```
                        finew = frpre+id+insuf
                        call([copycmd,templateenr,fiold],shell=True)
                        fileold = open(fiold).read()
                        fileold = fileold.replace(cycid[0],id[2:5])
                        filenew = open(fiold,'w')
                        filenew.write(fileold)
                        filenew.close()
                        call([copycmd,fiold,finew],shell=True)
                j = j + 1
#write BOC1 directory
        bocdir = 'BOC'+str(icyc)
        call(['mkdir',bocdir],shell=True)
        for id in frapid:
                fiold = frpre+id+'_old'+insuf
                finew = frpre+id+insuf
                call([copycmd,fiold,bocdir+'\\'+fiold],shell=True)
                call([copycmd,finew,bocdir+'\\'+finew],shell=True)
else:
#Copy the frapcon input files from the BOC directory
        bocdir = 'BOC'+str(icyc)
        for id in frapid:
                fiold = frpre+id+'_old'+insuf
                finew = frpre+id+insuf
                call([copycmd,bocdir+'\\'+fiold,fiold],shell=True)
                call([copycmd,bocdir+'\\'+finew,finew],shell=True)
#
#start the multicycle train
#
for pfi in parcfi:

        #Get PARCS rolling first

        parcscmd = ['start','%COMSPEC%','/k',parcpath, pfi]
        call(parcscmd,shell=True)

        #March through time with FRAPCON

        i = 1
        iter = 1
        move = False
        stay = False
```

```
        ready = False
        for tm in tmst:
                print 'executing cycle ', icyc, ' timestep ', i, ' (', tm, ' days)'
                while not move:
                        move = os.path.exists('nextstep.txt')
                        stay = os.path.exists('repeat.txt')
                        if move:
                                call([delcmd,'nextstep.txt'],shell=True)
                                stay = False
                                move = False
                                break
                        elif stay:
                                print 'cycle ', icyc, ' time step ', i, ' f/p iteration ', iter
                                sleep(1)  #added sleep command will hopefully add some stability
                                for id in frapid:
                                        dat = 'p2f'+id+'.dat'
                                        inp = frpre+id+insuf
                                        old = frpre+id+'_old'+insuf
                                        f2p = 'f2p'+id+'.dat'
                                        call([delcmd,inp],shell=True)
                                        call([copycmd,old,inp],shell=True)
# Add logic to include the appropriate timestep index and time value depending on core residence time
                                        if icyc == 1:
                                        # all fresh
                                                iin = i
                                                fraptime = tmst[i-1]
                                        elif icyc == 2:
                                                if id in frapid1:
                                                #fresh
                                                        iin = i
                                                        fraptime = tmst[i-1]
                                                else:
                                                #once-burnt
                                                        iin = nstp + i
                                                        fraptime = cyclen + tmst[i-1]
                                        elif icyc == 3:
                                                if id in frapid1:
                                                #fresh
                                                        iin = i
                                                        fraptime = tmst[i-1]
                                                elif id in frapid2:
```

```
                                        #once-burnt
                                                iin = nstp + i
                                                fraptime = cyclen + tmst[i-1]
                                        elif id in frapid3:
                                        #twice-burnt
                                                iin = 2*nstp + i
                                                fraptime = 2*cyclen + tmst[i-1]
                                #call the substitution routine
                                fp_util.frap_sub(dat,inp,iin,na,x,xt,fraptime)
                                #call frapcon
                                call([frappath,inp],shell=True)
                                #delete the parcs to frapcon data file
                                call([delcmd,dat],shell=True)
                                #rename the frapcon to parcs data file to have the id number
                                call([movecmd,f2pfile,f2p],shell=True)
                        call([delcmd,'repeat.txt'],shell=True)
                        #look for stragglers...rerun case if need be
                        strag = False
                        for id in frapid:
                                dat = 'p2f'+id+'.dat'
                                inp = frpre+id+insuf
                                f2p = 'f2p'+id+'.dat'
                                strag = os.path.exists(dat)
                                if strag:
                                        print 'p2f',id,' is still here, rerunning case'
                                        call([frappath,inp],shell=True)
                                        call([delcmd,dat],shell=True)
                                        call([movecmd,f2pfile,f2p],shell=True)
                                        strag = False
                        print 'frapcon is done, writing f2p flag file'
                        f2pflg = open(f2pflag,'w')
                        f2pflg.close()
                        stay = False
                        iter = iter + 1
                else:
                        print 'at cycle ', icyc, ' timestep ', i, ' iteration ', iter, ' waiting for
parcs to make up its mind'
                        sleep(1)
        print 'PARCS and FRAPCON have finished cycle ', icyc, ' timestep ', i, 'with ', iter, '
iterations'
        iterhist[nstp*(icyc-1)+(i-1)] = iter
```

```
                for id in frapid:
                        old = frpre+id+'_old'+insuf
                        new = frpre+id+insuf
                        call([delcmd,old],shell=True)
                        call([movecmd,new,old],shell=True)
                i = i + 1
                iter = 1
                sleep(1)
#Copy frapcon output files to a storage location
        eocdir = 'EOC'+str(icyc)
        call(['mkdir',eocdir],shell=True)
        for id in frapid:
                fiold = frpre+id+'_old'+insuf
                finew = frpre+id+insuf
                call([copycmd,fiold,eocdir+'\\'+fiold],shell=True)
                call([copycmd,finew,eocdir+'\\'+finew],shell=True)
#perform fuel shuffling between cyc 1 & 2 and between cyc 2 & 3
        if icyc < ncyc:
        #shuffle the fuel - first, mark and discard the rods which will be removed
        #for fuel that is not removed, duplicate file for shuffling
                disdir = 'discharge'+str(icyc)
                call(['mkdir',disdir],shell=True)
                for id in frapid:
                        frpfi = frpre+id+'_old'+insuf
                        if id in discard:
                                disfi = frpre+id+'_discard_cyc'+str(icyc)+insuf
                                call([movecmd,frpfi,disdir+'\\'+disfi],shell=True)
                        else:
                                fiprev = frpre+id+'_prev'+insuf
                                call([copycmd,frpfi,fiprev],shell=True)
                #then, perform the frapcon shuffling in accordance with the shuffle map
                for iassy in range(1,nfrp+1):
                        if iassy < 10:
                                istr = '00'+str(iassy)
                        elif iassy < 100:
                                istr = '0'+str(iassy)
                        else:
                                istr = str(iassy)
                        iprev = shufmap[iassy-1]
                        id = '00'+istr+'999'
                        idprev = '00'+iprev+'999'
```

```
                if iprev != '000':
                #this denotes a burned assembly, need to change the name
                        print 'moving rod ', iprev, ' to position ', istr
                        fiprev = frpre+idprev+'_prev'+insuf
                        fiold = frpre+id+'_old'+insuf
                        finew = frpre+id+insuf
                        call([movecmd,fiprev,fiold],shell=True)
                        fileold = open(fiold).read()
                        fileold = fileold.replace(cycid[icyc],istr)
                        filenew = open(fiold,'w')
                        filenew.write(fileold)
                        filenew.close()
                        call([copycmd,fiold,finew],shell=True)
        call([delcmd,'*_prev'+insuf],shell=True)
        #now, spawn the new cases for fresh rods
        for iassy in range(1,nfrp+1):
                if iassy < 10:
                        istr = '00'+str(iassy)
                elif iassy < 100:
                        istr = '0'+str(iassy)
                else:
                        istr = str(iassy)
                iprev = shufmap[iassy-1]
                id = '00'+istr+'999'
                if iprev == '000':
                #this denotes a fresh assembly, need to spawn a new case - assume type 3 feeds
                        print 'spawning new case for location ', istr
                        templateenr = template+'.0'
                        fiold = frpre+id+'_old'+insuf
                        finew = frpre+id+insuf
                        call([copycmd,templateenr,fiold],shell=True)
                        fileold = open(fiold).read()
                        fileold = fileold.replace(cycid[icyc],istr)
                        filenew = open(fiold,'w')
                        filenew.write(fileold)
                        filenew.close()
                        call([copycmd,fiold,finew],shell=True)
        #for some weird reason, filenew is being generated right but not fileold always
#               exit()
        exists = False
        for id in frapid:
```

```
                        fiold = frpre+id+'_old'+insuf
                        finew = frpre+id+insuf
                        exists = os.path.exists(fiold)
                        if not exists:
                                print 'old ', id, ' file not found, generating it'
                                call([copycmd,finew,fiold],shell=True)
                #store the BOC frapcon inputs for use if the cycle craps out for some reason
                bocdir = 'BOC'+str(icyc+1)
                call(['mkdir',bocdir],shell=True)
                for id in frapid:
                        fiold = frpre+id+'_old'+insuf
                        finew = frpre+id+insuf
                        call([copycmd,fiold,bocdir+'\\'+fiold],shell=True)
                        call([copycmd,finew,bocdir+'\\'+finew],shell=True)
                # finally, call the PARCS shuffling routine
                call([shufpath,shuffi[icyc-1]],shell=True)
                icyc = icyc + 1

itfile = open('iterhist.txt','w')
for iter in iterhist:
        print>>itfile, str(iter)
itfile.close()

#rerun last time step after turning full output on
#for id in frapid:
#       inp = frpre+id+'_old'+insuf
#       fp_util.frap_printsub(inp)
#       print 'running frapcon id, ', id, ' for printing'
#       call([frappath,inp],shell=True)
#       call([movecmd,frpre+outsuf,frpre+id+outsuf],shell=True)
#       call([movecmd,frpre+pltsuf,frpre+id+pltsuf],shell=True)
#call([delcmd,'*.dat'],shell=True)
```

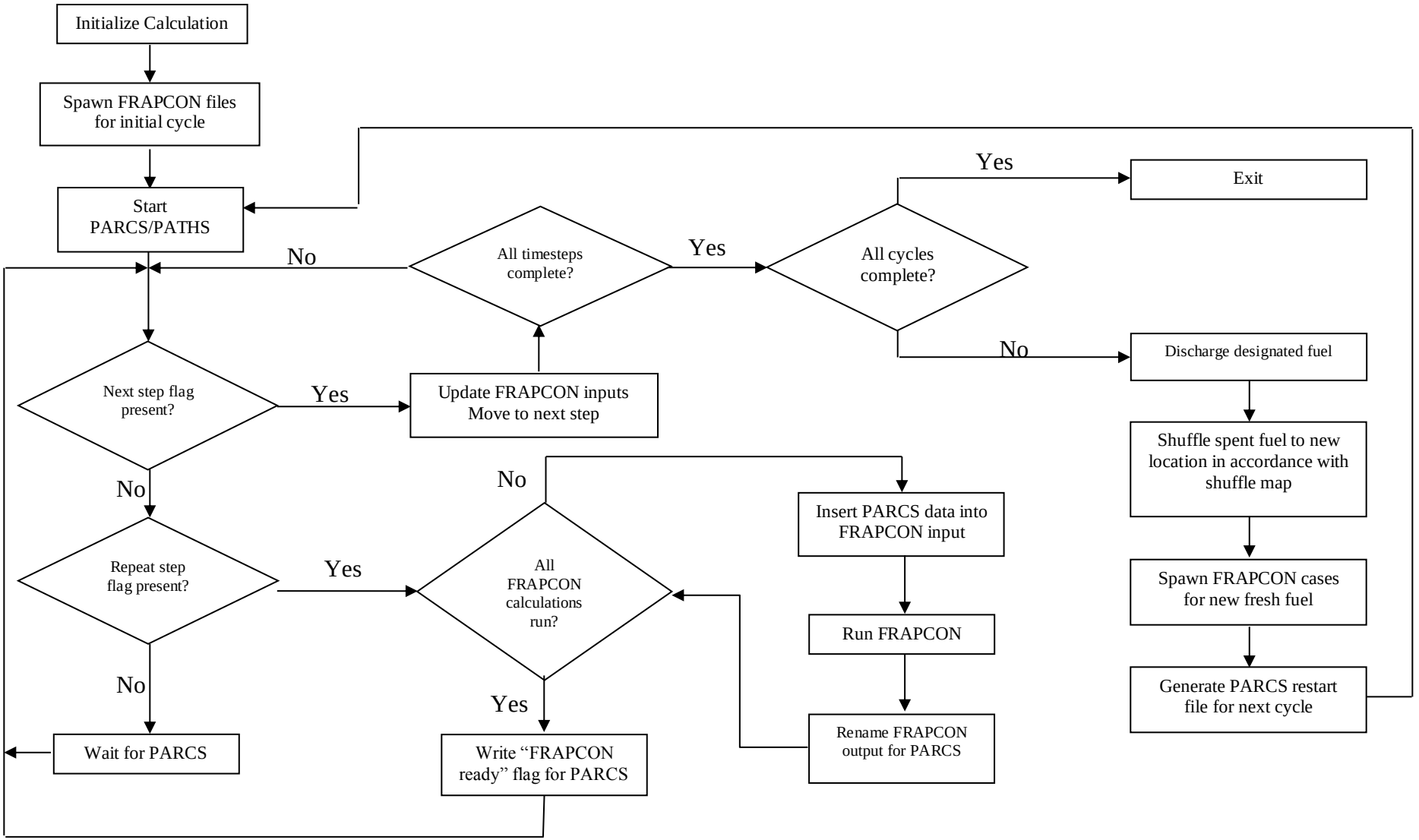


Figure B.1: FRAPARCS flowchart

APPENDIX C

Mini-Core Design Information

C.1: Lattice Model

Table C.1: Design data for 15x15 lattice calculations

Pin Pitch (cm)		1.43
Number of fuel pins		204
Number of control rod guide tubes		20
Number of instrumentation tubes		1
Fuel pin cladding diameter (cm)		1.07188
Fuel pin cladding thickness (cm)		6.1722×10^{-2}
Fuel pin fuel diameter (cm)	Type 1/2	0.929
	Type 3	0.926
Guide/instrumentation tube outer diameter (cm)		1.380
Guide/instrumentation tube inner diameter (cm)		1.228
Guide/instrumentation tube thickness (cm)		7.62×10^{-2}
Guide/instrumentation tube material		Zircaloy-4
Assembly pitch (cm)		21.504
Coolant material		Borated water
Fuel cladding material		Zircaloy-4
Fuel material		Uranium dioxide
Fuel density (g/cc)	Type 1	10.25
	Type 2	10.18
	Type 3	10.07
Fuel fraction of theoretical density (%)	Type 1	93.5
	Type 2	92.9
	Type 3	91.9
Fuel enrichment (wt/o)	Type 1	1.868
	Type 2	2.573
	Type 3	3.117
Power density (W/gU)		44.98

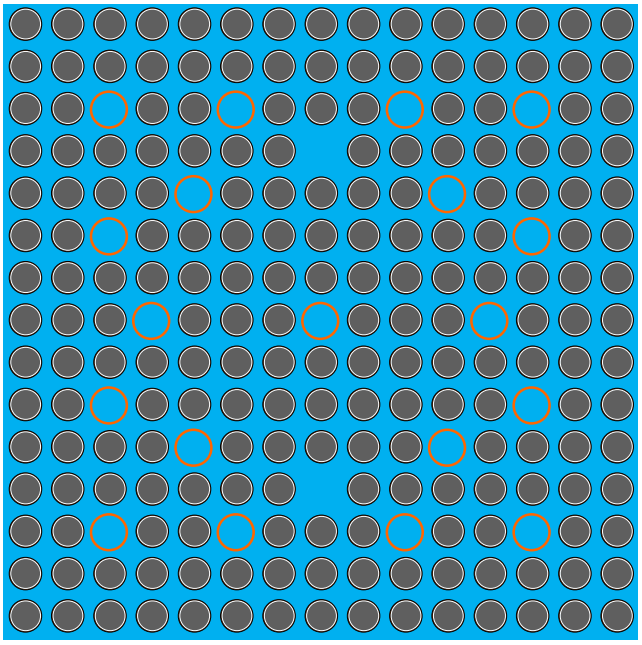


Figure C.1: 15x15 assembly design (blue: coolant; orange: clad; grey: fuel; white: gap)

Table C.2: Core conditions spanned by Surry Unit 1 benchmark cross sections

Parameter	Range
Fuel temperature (K)	293 – 1500
Moderator Density (g/cc)	0.11888 – 1.00519
Boron concentration (ppm)	0.1 – 2400

Table C.3: Depletion step structure for assembly cross section generation

Range (MWd/MTU)	Step Size (MWd/MTU)
0.0-0.1	0.1
0.1-0.5	0.4
0.5-1.0	0.5
1.0-11.0	1.0
11.0-12.5	1.5
12.5-60.0	2.5

C.2: PARCS/PATHS and FRAPCON Input Specification

Table C.4: PARCS/PATHS input for multi-cycle depletion

Input		Value
Number of assemblies		25
Number of fuel rods/assembly		204
Assembly pitch (cm)		21.504
Active core length		366.0 cm
Axial nodalization		28 total nodes: 2 nodes for bottom reflector (19.05 cm/node) 24 nodes for core region (15.25 cm/node) 2 nodes for top reflector (19.05 cm/node)
Boundary conditions		Reflective (East, West, North, South) Zero incoming current (Top, Bottom)
Number of cycles		3
Cycle depletion length		540
Depletion Step Length (d)		10
Core power (kW)		388670.0
Inlet coolant mass flow rate (kg/sec)		2024.4
Inlet coolant enthalpy (kJ/kg)		1268.0
Core outlet pressure (MPa)		15.512
Channel area (m^2)		2.469×10^{-2}
Hydraulic diameter (m)		3.17372×10^{-3}
Surface roughness (m)		6.800×10^{-5}
Pin pitch (m)		1.43×10^{-2}
Cladding outer radius (m)		5.3594×10^{-3}
Fuel pellet radius (m)	Type 1&2	4.64693×10^{-3}
	Type 3	4.63423×10^{-3}
Cladding thickness (m)		6.1722×10^{-4}
Number of water rods/assembly		21
Outer radius of water rod (m)		6.901×10^{-3}

Table C.5: FRAPCON input for multi-cycle depletion

Input		Value
Cladding outer diameter (m)		1.07188×10^{-2}
Cladding thickness (m)		6.1722×10^{-4}
Gap thickness (m)	Type 1&2	9.525×10^{-5}
	Type 3	1.080×10^{-4}
Plenum length (m)		0.254
Spring outer diameter (m)		9.1×10^{-3}
Spring wire diameter (m)		1.27×10^{-3}
Number of spring turns		33
Fuel pellet length (m)		1.14×10^{-2}
Fuel pellet dish depth (m)		2.4×10^{-4}
Fuel pellet end - dish shoulder width (m)		1.1×10^{-4}
Fuel stack height (m)		3.66
Number of axial nodes		24
Axial node length (m)		0.1525
Fuel pellet Uranium-235 enrichment (wt/o)	Type 1	1.868
	Type 2	2.573
	Type 3	3.117
Fuel fraction of theoretical density	Type 1	93.5
	Type 2	92.9
	Type 3	91.9
Cladding material		Zircaloy-4
Fuel rod fill gas		Helium
Fuel rod fill pressure (MPa)		2.41

BIBLIOGRAPHY

[1] Gehin, J., *CASL: The Consortium for Advanced Simulation of Light Water Reactors,* NRC Headquarters briefing, October 28, 2010, Rockville, MD, 2010.

[2] Downar, T., Xu, Y., Sekar, V., and Hudson, N. "PARCS v3.0, U.S. NRC Core Neutronics Simulator: Theory Manual," U.S. Nuclear Regulatory Commission, ADAMS Accession No. ML12215A526, January 2012.

[3] K.S. Smith et al, "SIMULATE-3 Advanced Three-Dimensional Two-Group Reactor Analysis Code," Studsvik Report STUDSVIK/SOA-95/15, 1995.

[4] Liu, Y.S. et. Al., "ANC: A Westinghouse Advanced Nodal Computer Code," Westinghouse Electric Company, WCAP-10966-A, 1986.

[5] Geelhood, K.J., Luscher, W.G. and Beyer, C.E., "FRAPCON-3.4: A Computer Code for the Calculation of Steady-State Thermal-Mechanical Behavior of Oxide Fuel Rods for High Burnup," US Nuclear Regulatory Commission, NUREG/CR-7022 Vol. 1, March 2011.

[6] Collins, B., Downar, T., Manera, A., Xu, Y. and A. Wysocki, "PATHS: PARCS Advanced Thermal Hydraulic Solver: Theory Manual," University of Michigan, Ann Arbor MI, May 2012.

[7] Adams, B.M. et al, "Dakota, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis," Sandia National Laboratories, Report No. SAND2011-9106, 2013.

[8] Python scripting language, version 2.7.3, downloaded as part of Anaconda installation package from <https://store.continuum.io/cshop/anaconda/>, November 2013.

[9] Zerkak, O., Manera, A., Gajev, I. and Kozlowski, T., "Review of Multi-Physics Coupling Techniques and Suggestions of Improvements in the Context of NURISP," Nuclear Reactor Integrated Simulation Project Document Number D-3.2.1.1, 2010.

[10] Stamm'ler, R.J.J. and Abbate, M.J., *Methods of Steady-State Reactor Physics in Nuclear Design*, Academic Press, Inc., New York, 1983.

[11] Szilard, R., Kothe, D. and Turinsky, P., "The Consortium for Advanced Simulation of Light Water Reactors," Enlarged Halden Programme Group Meeting, Sandefjord, Norway, 2011.

[12] Code of Federal Regulations, Title 10, Part 50, Appendix K: ECCS Evaluation Models

[13] Boyak et. al., "Quantifying Reactor Safety Margins: Application of Code Scaling, Applicability, and Uncertainty Evaluation Methodology to a Large-Break, Loss-of-Coolant Accident," US Nuclear Regulatory Commission, NUREG/CR-5249, December 1989.

[14] US NRC, Regulatory Guide 1.157, "Best-Estimate Calculations of Emergency Core Cooling System Performance," 1989.

[15] Bajorek, S. et. al., "Code Qualification Document for Best Estimate LOCA Analysis," Westinghouse Electric Company, WCAP-12945-A, 1998.

[16] "Realistic Large Break LOCA Methodology for Pressurized Water Reactors," AREVA NP Inc., EMF-2103(NP) Rev. 2, 2010.

[17] Ivanov, K.N. et. al., "Pressurized Water Reactor Main Steam Line Break (MSLB) Benchmark Volume I: Final Specifications," OECD Nuclear Energy Agency, NEA/NSC/DOC(99)8 1999.

[18] Ivanov, K.N. et. al., "Benchmark for Uncertainty Analysis in Modeling (UAM) for Design, Operation, and Safety Analysis of LWRs, Volume 1: Specification and Support Data for the Neutronics Cases (Phase I)," OECD Nuclear Energy Agency, NEA/NSC/DOC(2007)23, 2007.

[19] Carlson, R.W., "Reactor Core Physics Design and Operating Data for Cycles 1, 2, and 3 of Surry Unit 1 PWR Power Plant," Electric Power Research Institute, EPRI NP-79-2-LD, 1979.

[20] Personal communication from Dr. Nathanial Hudson, U.S. Nuclear Regulatory Commission, December 2013.

[21] J.Y.R. Rashid, S.K. Yagnik, R.O. Montgomery, "Light Water Reactor Fuel Performance Modeling and Multi-Dimensional Simulation," Journal of Materials, Vol. 63 No. 8, pp. 81-88, August 2011.

[22] U.S. Nuclear Regulatory Commission, "Final Safety Analysis Report Related to the Certification of the AP1000 Standard Plant Design," NUREG-1793, U.S. NRC, Washington, DC, September 2011.

[23] U.S. Nuclear Regulatory Commission, "Final Safety Analysis Report Related to the Certification of the ESBWR Standard Plant Design," NUREG-1966, U.S. NRC, Washington, DC, April 2014.

[24] US Nuclear Regulatory Commission, "Final Safety Evaluation by the Office of New Reactors: Topical Report MUAP-07008, Revision 2, 'Fuel System Design Criteria and Methodology,'" June 2013, ADAMS Accession No. ML13155A219.

[25] US Nuclear Regulatory Commission, "Turkey Point Units 3 and 4: Issuance of Amendments Regarding Extended Power Uprate," June 2012, ADAMS Accession No. ML11293A365.

[26] U.S. Nuclear Regulatory Commission, "Safety Evaluation of Westinghouse Electric Corporation Topical Report WCAP-12610, 'VANTAGE+ Fuel Assembly Reference Core Report'," U.S. NRC, Washington, DC, July 1991.

[27] L.E. Herranz, I. Vallejo, G. Khvostov, J. Sercombe and G. Zhou, "Assessment of fuel rod performance codes under ramp scenarios investigated within the SCIP project," Nuclear Engineering and Design, Vol. 241, pp. 815-825, 2011.

[28] K.J. Geelhood, W.G. Luscher, C.E. Beyer, D.J. Senor, M.E. Cunningham, D.D. Lanning and H.E. Adkins, NUREG/CR-7001, "Predictive Bias and Sensitivity in NRC Fuel Performance Codes," US NRC, October 2009.

[29] A. Karahan, A. Lerch, and M.S. Kazimi, "Development of FRAPCON-EP for high burnup and high temperature applications," Proceedings of 2010 LWR Fuel Performance/TopFuel/WRFPM, Orlando, FL, 2010.

[30] B. Feng, A. Karahan, M.S. Kazimi, "Steady-state fuel behavior modeling of nitride fuels in FRAPCON-EP," Journal of Nuclear Materials, Vol. 427, pp. 30-38, August 2012.

[31] E.P. Loewen, K.D. Weaver, J.K. Hohorst, "Reactivity, Isotopic and Thermal Steady-State Analysis of Homogeneous Thoria-Urania Fuel," Nuclear Technology, Vol. 137 No. 2, pp. 97-110, February 2002.

[32] Y. Long, L.J. Siefken, P. Hejzlar, E.P. Loewen, J.K. Hohorsrt, P.E. MacDonald, and M.S. Kazimi, "The Behavior of ThO_2-Based Fuel Rods During Normal Operation and Transient Events in LWRs," Nuclear Technology, Vol. 147 No. 1, pp. 120-139, July 2004.

[33] D.A. Vega, T. Watanabe, S.B. Sinnott, S.R. Phillpot, and J.S. Tulenko, "Toward an Atomistically Informed Fuel Performance Code: Thermal Properties using FRAPCON and Molecular Dynamics Simulation," Nuclear Technology, Vol. 165 No. 3, pp. 308-312, March 2009.

[34] K.J. Geelhood, W.G. Luscher, C.E. Beyer and J.M. Cuta, NUREG/CR-7023, "FRAPTRAN 1.4: A Computer Code for the Transient Analysis of Oxide Fuel Rods," US NRC, March 2011.

[35] K.J. Geelhood and C.E. Beyer, "FRAPTRAN Predictions of Cladding Failure in Accident Conditions," Transactions of the American Nuclear Society, Vol. 93 No. 1, pp. 757-758, November 2005.

[36] L.P. Pagani and G.E. Apostolakis, "A Methodology for Developing a Probability Distribution for the Failure Enthalpy of High-Burnup Fuels Via Simulation," Nuclear Technology Vol. 153 No. 1, pp. 9-17, January 2006.

[37] R. Hu and M.S. Kazimi, "Cladding Performance Under Power Oscillations in BWRs," Transactions of the American Nuclear Society, Vol. 103 No. 1, pp. 543-545, November 2010.

[38] *Fuel Analysis and Licensing Code: FALCON MOD01: Volume 1: Theoretical and Numerical Bases*, EPRI, Palo Alto, CA, 2004.

[39] L.J. Siefken et al, NUREG/CR-6150, Vol. 4, Rev. 2, "MATPRO – A Library of Materials Properties for Light-Water-Reactor Accident Analysis," US NRC, January 2001.

[40] A.T. Mai et al, "An Evaluation of the MATPRO Fuel Creep Model Using the FALCON Fuel Analysis Code," Transactions of the American Nuclear Society, Vol. 102 No. 1, pp. 888-889, June 2010.

[41] G. Khvostov et al, "A model for fission gas release and gaseous swelling of the uranium dioxide fuel coupled with the FALCON code," Nuclear Engineering and Design, Vol. 241 No. 8, pp. 2983-3007, August 2011.

[42] G. Khvostov et al, "Some insights into the role of axial gas flow in fuel rod behaviour during the LCOA based on Halden tests and calculations with the FALCON-PSI code," Nuclear Engineering and Design, Vol. 241 No. 5, pp. 1500-1507, May 2011.

[43] G. Khvostov et al, "Application of the FALCON code to PCI induced cladding failure and the effects of missing pellet surface," Annals of Nuclear Energy Vol. 62, pp. 398-412, December 2013.

[44] D. Hagrman, "INTERPIN-3 User's Manual," Studsvik Scandpower, Inc. Report SSP-01/400, 2001.

[45] G.M. Grandi and D. Hagrman, "Improvements to the INTERPIN Code for High Burnup and MOX Fuel," Transactions of the American Nuclear Society, Vol. 97. No. 1 pp. 614-615, November 2007.

[46] R.L. Williamson, et al, "Multidimensional multiphysics simulation of nuclear fuel behavior," Journal of Nuclear Materials, Vol. 423 No. 1, pp. 149-163, April 2012.

[47] M. Tonks, et al, "Fuels Performance Modeling Based on Microstructure Rather Than Burnup," Transactions of the American Nuclear Society, Vol. 110 No. 1, pp. 741-742, June 2014.

[48] S.R. Novascone et al, "Assessment of PCMI Simulation Using Multidimensional Multiphysics BISON Fuel Performance Code," Proceedings of Top Fuel 2012, Manchester, UK, June 2-6 2012.

[49] K.T. Clarno et al, "The AMP (Advanced MultiPhysics) Nuclear Fuel Performance code," Nuclear Engineering and Design, Vol. 252, pp. 108-120, November 2012.

[50] J.J. Duderstadt and L.J. Hamilton, *Nuclear Reactor Analysis*, John Wiley and Sons, Hoboken, NJ, 1976.

[51] M.E. Dunn and N.M. Greene, "AMPX-2000: A Cross-Section Processing System for Generating Nuclear Data for Criticality Safety Applications," Transactions of the American Nuclear Society, Vol. 86 No. 1, pp. 118-119, November 2002.

[52] R.E. MacFarlane and D.W. Muir, "The NJOY Nuclear Data Processing System, Version 91," Los Alamos National Laboratory Report LA-12740-M, 1994.

[53] "HELIOS Methods (version 1.10)", Studsvik Scandpower, Inc. Waltham MA, 2008.

[54] G. Espinosa-Paredes and J.R. Guzman, "Reactor physics analysis for the design of nuclear fuel lattices with burnable poisons," Nuclear Engineering and Design, Vol. 241 No. 12, pp. 5039-5054, December 2011.

[55] T. Kozlowski and T.J. Downar, "Analysis of Results for the OECD/NEA and U.S. NRC PWR MOX/UO_2 Core Transient Benchmark," Transactions of the American Nuclear Society, Vol. 92 No. 1, pp. 639-641, November 2005.

[56] K. Lee et al, "IAEA GT-MHR benchmark calculations by using the HELIOS/MASTER physics analysis procedure and the MCNP Monte Carlo code," Nuclear Engineering and Design, Vol. 238 No. 10, pp. 2654-2667, October 2008.

[57] G. Roh and H. Choi, "Benchmark Calculations for Standard and DUPIC CANDU Fuel Lattices Compared with the MCNP-4B Code," Nuclear Technology, Vol. 132 No. 1, pp. 128-151, October 2000.

[58] M.M. Kyaw and M. Kim, "Nuclear Design of Small PWR Core with Thorium Fuel," Transactions of the American Nuclear Society, Vol. 99 No. 1, pp. 747-749, November 2009.

[59] F. D'Auria et al, "The three-dimensional neutron kinetics coupled with thermal-hydraulics in RBMK accident analysis," Nuclear Engineering and Design, Vol. 238 No. 4, pp. 1002-1025, April 2008.

[60] M. Edenius et al, "CASMO-4, A Fuel Assembly Burnup Program, User's Manual," STUDSVIK/SOA-95/1, Studsvik of America, Inc., 1995.

[61] M. Pusa, "Incorporating sensitivity and uncertainty analysis to a lattice physics code with application to CASMO-4," Annals of Nuclear Energy, Vol. 40 No. 1, pp. 153-162, February 2012.

[62] H. Lin et al, "Qualification of the Taiwan Power Company's pressurized water reactor physics methods using CASMO-4/SIMULATE-3," Nuclear Engineering and Design, Vol. 253, pp. 71-76, December 2012.

[63] S. Palmtag and J. Rhodes III, "CASMO-4 and Multigroup MCNP Comparisons for MOX Fuel Assemblies," Transactions of the American Nuclear Society, Vol. 92 No. 1, pp. 508-509, June 2008.

[64] A. Vasiliev et al, "Development of a CASMO-4/SIMULATE-3/MCNPX calculation scheme for PWR fast neutron fluence analysis and validation against RPV scraping test data," Annals of Nuclear Energy, Vol. 34 No. 8, pp. 615-627, August 2007.

[65] M.A. Jessee and M.D. DeHart, "TRITON: A Multipurpose Transport, Depletion and Sensitivity and Uncertainty Analysis Module," Oak Ridge National Laboratory Report ORNL/TM-2005/39, June 2011.

[66] M.D. DeHart and S.M. Bowman, "Reactor Physics Methods and Analysis Capabilities in SCALE," Nuclear Technology, Vol. 174 No. 2, pp. 196-213, May 2011.

[67] D.V. Altiparmakov, "New Capabilities of the Lattice Code WIMS-AECL," Proceedings of Physor 2008, Interlaken, Switzerland, September 14-19 2008.

[68] M. Ouisloumen, et al, "PARAGON: The New Westinghouse Assembly Lattice Code," ANS International meeting of Mathematical Methods for Nuclear Applications, Salt Lake City, UT, 2001.

[69] M. Yamamoto, et al, "Development and Validation of TGBLA BWR Lattice Physics Methods," Proceedings of the Topical Meeting on Reactor Physics and Shielding, Chicago, IL, 1984.

[70] C. Demaziére et al, "Comparison of the U.S. NRC PARCS Core Neutronics Simulator Against In-Core Detector Measurements for LWR Applications," U.S. NRC Report NUREG/IA-0414, April 2012.

[71] M. Hursin et al, "Analysis of the Core Power Response During a PWR Rod Ejection Transient Using the PARCS Nodal Code and the DeCART MOC Code," Nuclear Science and Engineering, Vol. 170 No. 2, pp. 151-167, February 2012.

[72] D.J. Diamond et al, "Intercomparison of results for a PWR rod ejection accident," Nuclear Engineering and Design, Vol. 208 No. 2, pp. 181-189, September 2001.

[73] P.J. Turinsky, et al, "Code Abstract – NESTLE: A Few-Group Neutron Diffusion Equation Solver Utilizing the Nodal Expansion Method for Eigenvalue, Adjoint, Fixed-Source, Steady-State and Transient Problems," Nuclear Science and Engineering, Vol. 120, pp. 72, 1995.

[74] "SIMULATE-3K, Models and Methodology," Studsvik Scandpower Report SSP-98/13 Rev. 6, 2009.

[75] R. Szilard et al, "The Consortium for Advanced Simulation of Light Water Reactors," Enlarged Halden Programme Group Meeting, Sandefjord, Norway, 2011.

[76] B. Kochunas, et al, "MPACT: Michigan Parallel Advanced Characteristics Transport," ANS M&C Conference, Sun Valley, Idaho, May, 2013.

[77] B. Collins, et al, "Assessment of the 2D/1D Implementation in MPACT," PHYSOR-14, Kyoto, Japan, September, 2014.

[78] T. Kozlowski and T.J. Downar, "OECD/NEA and US NRC PWR MOX/UO_2 Core Transient Benchmark, Final Specifications, Revision 2," OECD Nuclear Energy Agency, NEA/NSC/DOC(2003)20, December 2003.

[79] J. Solis et al, "Boiling Water Reactor Turbine Trip (TT) Benchmark, Volume I: Final Specifications," OECD Nuclear Energy Agency, NEA/NSC/DOC(2001)1, February 2001.

[80] C.D. Fletcher and R.R. Schulz, "RELAP5/MOD3 Code Manual, Volume I: Theory," U.S. NRC Report NUREG/CR-5535, Washington, DC 1995.

[81] "TRACE V5.0 User's Manual," U.S. NRC Office of Nuclear Regulatory Research, Division of Systems Analysis, Washington, DC 2010.

[82] T.J. Downar, et al, "PARCS v3.0, U.S. NRC Core Neutronics Simulator: User Manual," University of Michigan, Ann Arbor, MI, January 2012.

[83] V.A. Mousseau, "A Fully Implicit, Second Order in Time, Simulation of a Nuclear Reactor Core," Proceedings of ICONE14, Miami, FL, July 17-20 2006.

[84] H. Park et al, "High-Order Spatio-Temporal Coupling of Radiation-Diffusion and Heat Conduction using Jacobian-Free Newton Krylov Discontinuous Galerkin Method,

invited," Transactions of the American Nuclear Society, Vol. 98 No. 1, pp. 597-599, June 2008.

[85] R. Hooper et al, "A Multi-Physics Coupling Approach for Integrated Nuclear Reactor Safety Calculations," Transactions of the American Nuclear Society, Vol. 98 No. 1, pp 609, June 2008.

[86] T. Kozlowski et al, "Analysis of the OECD MSLB Benchmark with the Coupled Neutronics and Thermal-Hydraulics Code RELAP5/PARCS," Proceedings of PHYSOR 2000, Pittsburgh, PA, May 2000.

[87] T. Kozlowski et al, "Consistent Comparison of the Codes RELAP5/PARCS and TRAC-M/PARCS for the OECD MSLB Coupled Code Benchmark," Nuclear Technology, Vol. 146, April 2004.

[88] D. Lee and T.J. Downar, "Analysis of the OECD/NRC BWR Turbine Trip Transient Benchmark with the Coupled Thermal-Hydraulics and Neutronics Code TRAC-M/PARCS," Nuclear Science and Engineering, Vol. 148 No. 2, pp. 291-305, October 2004.

[89] I. Gajev et al, "Space-Time Convergence Study Based on the OECD Ringhals-1 Stability Benchmark," Transactions of the American Nuclear Society, Vol. 106 No. 1, pp. 984-987, June 2012.

[90] L. Cheng et al, "TRACE/PARCS Core Modeling of a BWR/5 for Accident Analysis of ATWS Events," Transactions of the American Nuclear Society, Vol. 109 No. 1, pp. 979-982, November 2013.

[91] L. Cheng et al, "BWR Anticipated Transients Without Scram Leading to Instability," Transactions of the American Nuclear Society, Vol. 109 No. 1, pp. 983-986, November 2013.

[92] R. Wang et al, "Validation of the U.S. NRC Coupled Code System TRITON/TRACE/PARCS Using the Special Power Excursion Reactor Test III," Nuclear Technology, Vol. 183 No. 3, pp. 504-514, September 2013.

[93] J.K. Watson and K.N. Ivanov, "Demonstration of Implicit Coupling of TRACE/PARCS Using Simplified One-Dimensional Problems," Nuclear Technology, Vol. 180 No. 2, pp. 174-190, November 2012.

[94] J.K. Watson, "Implicit Time-Integration Method for Simultaneous Solution of a Coupled Non-Linear System," Ph.D. Dissertation, Pennsylvania State University Department of Mechanical and Nuclear Engineering, May 2010.

[95] A.M. Ward, "A Newton-Krylov Solution to Coupled Neutronics-Porous Medium Equations," Ph.D. Dissertation, University of Michigan Department of Nuclear Engineering and Radiological Sciences, February 2012.

[96] D. Gaston et al, "MOOSE: A parallel computational framework for coupled systems of nonlinear equations," Proceedings of the International Conference on Mathematics, Computational Methods & Reactor Physics 2009, Saratoga Springs, NY, May 3-7 2009.

[97] M. Hursin et al, "Impact of improved neutronic methodology on the cladding response during a PWR reactivity initiated accident," Nuclear Engineering and Design, Vol. 262, pp. 180-188, September 2013.

[98] F. Gleicher et al, "Coupling the Core Analysis Program DeCART to the Fuel Performance Application BISON," Proceedings of the International Conference on Mathematics, Computational Methods & Reactor Physics 2013, Sun Valley, ID, May 5-9 2013.

[99] T. Viitanen and V. Tulkki, "Combining Reactor Physics and Fuel Performance Calculations," Proceedings of TOPFUEL 2012, Manchester, UK, June 2-6 2012.

[100] "Compendium of ECCS Research for Realistic LOCA Analysis," U.S. NRC Report NUREG-1230, Office of Nuclear Regulatory Research, Division of Systems Research, Washington, DC, December 1988.

[101] FirstEnergy Nuclear Operating Company, "Beaver Valley Power Station Extended Power Uprate Licensing Report," FENOC Report 6517-fm.doc-092304, Akron, Ohio, September 2004.

[102] F.D'Auria et al, "The Best Estimate Plus Uncertainty (BEPU) approach in licensing of current nuclear reactors," Nuclear Engineering and Design, Vol. 248, pp. 317-328, July 2012.

[103] M. Ionescu-Bujor and D.G. Cacuci, "A Comparative Review of Sensitivity and Uncertainty Analysis of Large-Scale Systems – I: Deterministic Methods," Nuclear Science and Engineering, Vol. 147 No. 3, pp. 189-203, July 2004.

[104] D.G. Cacuci and M. Ionescu-Bujor, "A Comparative Review of Sensitivity and Uncertainty Analysis of Large-Scale Systems – II: Statistical Methods," Nuclear Science and Engineering, Vol. 147 No. 3, pp. 204-217, July 2004.

[105] K.O. Ott and R.J. Neuhold, *Introductory Nuclear Reactor Dynamics*, American Nuclear Society, La Grange Park, IL, 1985.

[106] J. Lewins, *Importance: The Adjoint Function*, Pergamon Press, Oxford, 1965.

[107] M. Williams, "Perturbation Theory for Nuclear Reactor Analysis," *The CRC Handbook of Nuclear Reactors Calculations, Vol. III,* CRC Press, West Palm Beach, FL, 1986.

[108] E.M. Oblow, "Sensitivity Theory for Reactor Thermal-Hydraulics Problems," Nuclear Science and Engineering, Vol. 68, pp. 322-337, 1978.

[109] D.G. Cacuci et al, "Sensitivity Theory for General Systems of Nonlinear Equations," Nuclear Science and Engineering, Vol. 75, pp. 88-110, 1980.

[110] D.G. Cacuci, *Sensitivity and Uncertainty Analysis*, Chapman & Hall/CRC, Boca Raton, FL, 2003.

[111] A. Achintya and Sankaran Mahadeva, *Probability, Reliability, and Statistical Methods in Engineering Design*, John Wiley & Sons, Hoboken, NJ, 2000.

[112] Yigal Ronen (editor), *Uncertainty Analysis*, CRC Press Inc., West Palm Beach, FL, 1988.

[113] R.H. Myers, *Response Surface Methodology*, Allyn and Bacon, Inc., Boston, 1971.

[114] A. Saltelli et al (editors), *Sensitivity Analysis*, John C. Wiley and Sons, Hoboken, NJ, 2000.

[115] S. Hosder, R. W. Walters and R. Perez, "Towards stochastic fluid mechanics via polynomial chaos," AIAA 2006-891, Proceedings of the 41st AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, January 2006.

[116] L. Mathelin and M. Y. Hussaini, "A Stochastic Collocation Algorithm for Uncertainty Analysis," NASA Report NASA/CR-2003-212153, February 2003.

[117] H.F. Stripling, et al, "A generalized adjoint framework for sensitivity and global error estimation in time-dependent nuclear reactor simulations," Annals of Nuclear Energy, Volume 52, pp. 47-58, February 2013.

[118] H.S. Abdel-Khalik, et al, "Overview of hybrid subspace methods for uncertainty quantification, sensitivity analysis," Annals of Nuclear Energy, Vol. 52, pp. 28-46, February 2013.

[119] H.S. Abdel-Khalik, et al, "Hybrid Uncertainty and Sensitivity Algorithms for High Dimensional Nonlinear Models, Part I: Introduction to the Theory," Transactions of the American Nuclear Society, Vol. 103 No. 1, pp. 375-377, November 2010.

[120] S. Bajorek et al, "Code Qualification Document for Best Estimate LOCA Analysis," Technical Report WCAP-12945-A, Westinghouse Electric Company, Pittsburgh, PA, 1998.

[121] J. Zhang et al, "Application of the WCOBRA/TRAC best-estimate methodology to the AP600 large-break LOCA analysis," Nuclear Engineering and Design, Vol. 186, pp. 279-301, 1998.

[122] M.E. Nissley et al, "Realistic Large-Break LOCA Evaluation Methodology Using the Automated Statistical Treatment of Uncertainty Method (ASTRUM)," Technical Report WCAP-16009-NP, Westinghouse Electric Company, Pittsburgh, PA, 2003.

[123] U.S. Nuclear Regulatory Commission, "Safety Evaluation by the Office of Nuclear Reactor Regulation Related to Amendment No. 174 to Facility Operating License No. NPF-72, Amendment No. 174 to Facility Operating License No. NPF-77, Amendment No. 181 to Facility Operating License No. NPF-37, and Amendment No. 181 to Facility Operating License No. NPF-66, Exelon Generation Company LLC, Braidwood Station Units 1 & 2, Byron Station Units 1 & 2," U.S. NRC, Washington, DC, April 2014.

[124] U.S. Nuclear Regulatory Commission, "Safety Evaluation by the Office of Nuclear Reactor Regulation Related to Amendment No. 249 to Renewed Facility Operating License No. DPR-31, and Amendment No. 245 to Renewed Facility Operating License No. DPR-41, Florida Power and Light Company, Turkey Point Plant Unit Nos. 3 and 4" U.S. NRC, Washington, DC, June 2012.

[125] "Realistic Large Break LOCA Methodology for Pressurized Water Reactors," Technical Report EMF-2103(NP) Rev. 2, AREVA NP Inc., Lynchburg, VA, 2010.

[126] U.S. Nuclear Regulatory Commission, "Safety Evaluation by the Office of Nuclear Reactor Regulation Related to Amendment No. 213 to Facility Operating License No. DPR-67, Florida Power and Light Company, St. Lucie Plant, Unit No. 1," U.S. NRC, Washington, DC, July 2012.

[127] "U.S. EPR Realistic Large Break Loss of Coolant Accident Topical Report," Technical Report ANP-10278NP Rev. 1, AREVA NP Inc., Lynchburg, VA November 2011.

[128] S.S. Wilk, "Determination of sample sizes for setting tolerance limits," The Annals of Mathematical Statistics, Vol. 12, pp. 91-96, 1941.

[129] A.K. Trivedi, et al, "Uncertainty analysis of large break LOCA for pressurized heavy water reactor," Nuclear Engineering and Design, Vol. 245, pp. 180-188, April 2012.

[130] D.A. Fynan, et al, "Unscented Transform for Approximating Mean and Covariance of Distributions in BEPU Methodologies," Transactions of the American Nuclear Society, Vol. 109 No. 1, pp. 2178-2181, November 2013.

[131] E. Greenspan, et al, "Time-Dependent Generalized Perturbation Theory for Coupled Neutron-Nuclide Problems," Nuclear Science and Engineering, Vol. 73, pp. 210-218, 1980.

[132] M.L. Williams, "Development of Depletion Perturbation Theory for Coupled Neutron/Nuclide Fields," Nuclear Science and Engineering, Vol. 70, pp. 20-36, 1979.

[133] W.S. Yang and T.J. Downar, "Generalized Perturbation Theory for Constant Power Core Depletion," Nuclear Science and Engineering, Vol. 99, pp. 353-366, 1988.

[134] A. Yankov et al, "Comparison of XSUSA and 'two-step' approaches for full-core uncertainty quantification," Proceedings of the International Conference on the Physics of Reactors (PHYSOR) 2012, Knoxville, TN, April 2012.

[135] A. Yankov et al, "A Two-Step Approach to Uncertainty Quantification of Core Simulators," Science and Technology of Nuclear Installations, Vol. 2012, 2012.

[136] M.L. Williams et al, "A Statistical Sampling Method for Uncertainty Analysis with SCALE and XSUSA," Nuclear Technology, Vol. 183 No. 3, pp. 515-526, September 2013.

[137] J.D. Kerrigan and D.R. Coleman, "Application of the Response Surface Method of Uncertainty Analysis to Establish Distributions of FRAP-S3 Calculated Stored Energy for PWR-Type Fuels," Nuclear Engineering and Design, Vol. 54, pp. 211-224, 1979.

[138] K. Kim, "The study on a statistical methodology for PWR fuel rod internal pressure evaluation," Nuclear Engineering and Design, Vol. 240, No. 6, pp. 1397-1402, June 2010.

[139] A. Bouloré, "Uncertainty and sensitivity analysis of nuclear fuel thermal behavior," Nuclear Engineering and Design, Vol. 253, pp. 200-210, December 2012.

[140] V.I. Arimescu, "Quantifying Uncertainties and Best-Estimate, Realistic Methods for LWR Fuel Mechanical Analysis during Steady-State and AOO Transients," Proceedings of TOPFUEL 2012, Manchester, UK, June 2-6, 2012.

[141] K. Lassman, C. O'Carrol, J. VanderLaar and C.T. Walker, "The Radial Distribution of Plutonium in High Burnup UO_2 Fuels," Journal of Nuclear Materials, Vol. 28 pp. 223-231, 1994.

[142] D. Engel and A. White, "Graphical User Interface and Stochastic Framework for the FRAPCON Code," Pacific Northwest National Laboratory Letter Report to U.S. NRC, Richland, WA, December 2009.

[143] U.S. Nuclear Regulatory Commission, "St. Lucie Plant Unit 2 Extended Power Uprate Fuel Rod Thermal-Mechanical Performance Summary of Results," U.S. NRC, Washington, DC, April 2012.

[144] B. Collins et al, "PATHS: A Steady-State Two Phase Thermalhydraulics Solver for PARCS Depletion," Proceedings of NURETH-14, Toronto, Canada, September 25-30, 2011.

[145] Personal communication from Dr. Patrick Raynaud, US Nuclear Regulatory Commission, March 2013.

[146] E. W. Lemmon, M. O. McLinden and D. G. Friend, "Thermophysical Properties of Fluid Systems," in NIST Chemistry WebBook, NIST Standard Reference Database No. 69, Eds. P. J. Lindstrom and W. G. Mallard, National Institute of Standards and Technology, Gaithersburg, MD, http://webbook.nist.gov, retrieved April 2014.

[147] "Chapter 3: Reactor Core," *Updated Final Safety Analysis Report for Turkey Point Units 3 and 4*, Florida Power and Light Company, June 2010.

[148] Y. Xu and T. Downar, "GenPMAX-V6: Code for Generating the PARCS Cross Section Interface File PMAX," University of Michigan, Ann Arbor MI, March 2012.

[149] B.J. Ade, "SCALE/TRITON Primer: A Primer for Light Water Reactor Lattice Physics Calculations," U.S. NRC Report NUREG/CR-7041, Washington, DC, 2012.

[150] "User Manual: Zenith," Studsvik Scandpower, Inc., April 2008.

[151] G. Grandi, K. Smith, Z. Xu and J. Rhoades, "Effect of CASMO-5 cross section data and Doppler temperature definitions on LWR reactivity initiated accidents," PHYSOR-2010 – Advances in Reactor Physics to Power the Nuclear Renaissance, Pittsburgh, PA, May 2010.

[152] "Chapter 15: Transient and Accident Analysis," *US-APWR Final Safety Analysis Report*, Mitsubishi Heavy Industries, September 2013.

[153] K. Geelhood, PNNL to J. Voglewede, NRC, "Bounding Design Calculations in FRAPCON-3," NRC letter report, Pacific Northwest National Laboratory, November 2008.

[154] Personal communication from Kenneth Geelhood, Pacific Northwest National Laboratory, July 2, 2014.

[155] NRC-2008-0332, "Advance Notice of Public Rulemaking: Performance-Based Emergency Core Cooling System Acceptance Criteria," US Nuclear Regulatory Commission, Washington, DC (2008).

[156] M. S. Eldred, "Recent advances in Non-Intrusive Polynomial Chaos and Stochastic Collocation Methods for Uncertainty Analysis and Design," AIAA-2009-2274, 50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Palm Springs, California, 2009.

[157] G. D. Wyss and K. H. Jorgensen, "A User's Guide to LHS: Sandia's Latin Hypercube Sampling Software," SAND98-0210, Sandia National Laboratories, New Mexico, February 1998.

[158] M.D. McKay et al, "A comparison of three methods for selecting values of input variables in the analysis of the output of a computer code," Technometrics, Vol. 21, pp. 239-245, 1979.

[159] R. L. Iman and W. J. Conover, "A Distribution-Free Approach to Inducing Rank Correlation Among Input Variables," Communications in Statistics, Vol. 11 No. 3, pp. 311-334, 1982.

[160] R. H. Myers and D. C. Montgomery, *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*, John C. Wiley & Sons, NY, 1995.

[161] A. A. Giunta, J.M. McFarland, L. P. Swiler and M. S. Eldred, "The promise and peril of uncertainty quantification using response surface approximations," Structure and Infrastructure Engineering, Maintenance, Management, Life-Cycle Design and Performance, Vol. 2 No. 3-4, pp. 175-189, 2007.

[162] R. G. Ghanem and P. Spanos, *Stochastic Finite Elements: A Spectral Approach*, Springer-Verlag, New York, 1991.

[163] G. H. Golub and J. H. Welch, "Calculation of Gauss Quadrature Rules," Mathematics of Computation, Vol. 23 No. 106, pp. 221-230, 1969.

[164] D. Xiu and G. E. Karniadakis, "The Weiner-Askey polynomial chaos for stochastic differential equations," SIAM Journal of Scientific Computing, Vol. 24 pp. 619-644, 2002.

[165] B. Sudret, "Global Sensitivity Analysis using polynomial chaos expansion," Reliability Engineering and Systems Safety, Vol. 93, pp. 964-979, 2008.

[166] S. Smolyak, "Quadrature and interpolation formulas for tensor products of certain classes of functions," Dolk. Akad. Nauk SSR, Vol. 4, pp. 240-243, 1963.

[167] F. Nobile, R. Tempone and C. G. Webster, "An anisotropic sparse grid collocation method for partial differential equations with random input data," SIAM Journal of Numerical Analysis, Vol. 46, No. 5, pp. 2411-2445, 2008.

[168] T. Gerstner and M. Griebel, "Sparse Grids," *Encyclopedia of Quantitative Finance*, R. Cont (editor), John Wiley & Sons, NY, 2008.

[169] F. Nobile, R. Tempone and C. G. Webster, "A sparse grid stochastic collocation method for partial differential equations with random input data," SIAM Journal of Numerical Analysis, Vol. 46, No. 5, pp. 2309-2345, 2008.

[170] A. Saltelli et al, *Global Sensitivity Analysis: The Primer*, John C. Wiley & Sons, West Sussex, UK, 2008.

[171] T. Ikonen and V. Tulkki, "The importance of input interactions in the uncertainty and sensitivity analysis of nuclear fuel behavior," Nuclear Engineering and Design, Vol. 275, pp. 229-241, 2014.

[172] N. Draper and H. Smith, *Applied Regression Analysis*, John C. Wiley & Sons, Hoboken, NJ, 1998.

[173] I.M. Sobol', "Sensitivity analysis for non-linear mathematical models," Mathematical Modeling and Computational Experiment, Vol. 1 pp. 407-414, 1993.

[174] M. Holtz, *Sparse Grid Quadrature in High Dimensions with Applications in Finance and Insurance*, Lecture Notes in Computational Science and Engineering 77, Springer-Verlag Berlin Heidelberg, 2011.

[175] A. Saltelli, "Making best use of model evaluations to compute sensitivity indices," Computer Physics Communications, Vol. 145, pp. 280-297, 2002.

[176] G. Tang and G. Iaccarino, "Global Sensitivity Analysis for Stochastic Collocation," AIAA 2010-2922, 51st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Orlando, Florida, April 2010.

[177] Personal communication from Kenneth Geelhood, Pacific Northwest National Laboratory, August – October 2014.

[178] A. Haldar and S. Mahadevan, *Probability, Reliability, and Statistical Methods in Engineering Design*," John Wiley & Sons, NY, 2000.