3-2021

Nonlinear Solution of the Time Eigenvalue of a Fast Burst Reactor Using the Finite Volume Method

Stephen H. Baxter

Nonlinear Solution of the Time Eigenvalue of a Fast Burst Reactor Using the Finite Volume Method

THESIS

Stephen H. Baxter, 1st Lt, USAF

AFIT-ENP-MS-21-M-100

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENP-MS-21-M-100

NONLINEAR SOLUTION OF THE TIME EIGENVALUE OF A FAST BURST REACTOR USING THE FINITE VOLUME METHOD

THESIS

Presented to the Faculty

Department of Engineering Physics

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Master of Science in Nuclear Engineering

Stephen H. Baxter, B.S.E.

1st Lt, USAF

January 2021

AFIT-ENP-MS-21-M-100

NONLINEAR SOLUTION OF THE TIME EIGENVALUE OF A FAST BURST REACTOR USING THE FINITE VOLUME METHOD

THESIS

Stephen H. Baxter, B.S.E.
1st Lt, USAF

Committee Membership:

LTC Edward L. Hobbs, Ph.D.
Chair

Dr. John W. McClory
Member

Dr. Adib J. Samin
Member

AFIT-ENP-MS-21-M-100

Abstract

Fast burst reactors are used in research environments to study super critical systems and as a source of fast neutron pulses for use in other experiments. These reactors fill an important role in both civilian and defense research. However, due to costs associated with securing the highly enriched uranium fuel used by these reactors, many have been shut down in recent years. Replacing these reactors with less expensive low enriched uranium fuel has been proposed as a solution to this problem. However, before a fast burst reactor using this new fuel type can be constructed, it is desirable to first determine the feasibility of such a design using numerical modeling. Previous attempts have been made to develop these predictive models, but these methods are currently lacking in verification data. Therefore, this work builds off of one such numerical model in order to enhance the verification of this predictive method. This effort models fast burst reactors using the one dimensional, one group neutron diffusion equation to solve for the time eigenvalue. For this problem there exist an analytical solution against which the numerical results can be verified. The existing solution method is enhanced by the addition of a second order accurate finite volume discretization, which is then used to model two separate fast burst reactors. The results of these models are then compared to the results of previous work, the analytical solution, and existing experimental burst width data for each of the two reactors.

Table of Contents

Page

List of Figures

List of Tables

NONLINEAR SOLUTION OF THE TIME EIGENVALUE OF A FAST BURST REACTOR USING THE FINITE VOLUME METHOD

I. Introduction

1.1 Motivation

Fast burst reactors (FBRs) are used in research environments both to study super critical systems themselves, and as a source of fast neutrons pulses for use in other experiments [1]. These reactors operate by rapidly inserting reactivity into the system to induce a super prompt critical state, producing a pulse, or burst, of neutrons before negative temperature-reactivity feedback quenches the system, returning it to a sub-critical state [2].

The original FBRs made use of highly enriched uranium (HEU) fuel in order to achieve this effect [1]. However, in a 2019 report the National Nuclear Security Administration listed HEU fueled research reactors as a security risk and stated a programmatic objective of converting both domestic and international research reactors from HEU to low enriched uranium (LEU) fuel [3]. This proliferation concern posed by the HEU fuel has led to increasing security measures, and consequently to increasing costs associated with operating FBRs, as attested by the office of the USD/AT&L, which in 2005 assessed that these costs had led to the shut down of two out of the three domestic FBRs used for defense applications in the United States [4]. Therefore, the conversion of FBRs to LEU fuel is desirable from both a security and a cost perspective.

However, it must first be determined if it is possible to convert a given reactor

from HEU to LEU fuel while still fulfilling the originally intended purposes of the reactor. It is therefore necessary to model the effects of LEU fuel on FBRs in order to assess the effects the new fuel will have on the system and whether the bursts produced will still satisfy research needs.

It was this problem that the Jacobian Free Newton-Krylov (JFNK) Alpha, and k Eigenvalue Solver (JAKES) was designed to model [5]. This algorithm made use of recent advances in nonlinear solution techniques, using a software package developed by Argonne National Laboratory, and applied them to the FBR problem [5, 6]. These techniques were used to develop a deterministic solution for the time eigenvalue of a FBR modeled using the neutron diffusion equation in one dimension. The algorithm also computed the k eigenvalue of the system as a way of initializing the time eigenvalue calculation. Verification was conducted for both of these solutions, but the verification of the time eigenvalue was more limited in scope. However, it is the time eigenvalue that is of the most interest for the FBR problem.

1.2 Research Problem

The goal of this effort was two-fold: to enhance the time eigenvalue solution accuracy of JAKES by replacing the existing finite difference discretization with a finite volume discretization; and to improve the verification of the JAKES algorithm in order to increase its usefulness as a predictive model for innovative FBR designs. Because of this need for verification, this effort focused only on numerical problems for which an analytical solution for comparison existed. Therefore, this effort only builds off of previous work with the 1D neutron diffusion equation in spherical coordinates, and ignores work with the 2D transport equation. Because of this focus, it was for 1D spherical geometry that the finite volume method was implemented. This particular discretization method was selected because it would enable a more accurate imple-

mentation of heterogeneous models in the future than the finite difference method, while avoiding the complexities introduced by the inclusion of basis functions with the finite element method. This was accomplished, but this effort also enhanced the existing finite difference discretization and applied both methods in cylindrical and Cartesian coordinates as well.

Verification was accomplished by comparing the new finite volume discretization results to the findings for the previous version of JAKES for a model of the Godiva I FBR, as well as to the analytical solution to the eigenvalue problem and the known burst width range measured for Godiva I in past experiments. This task was accomplished during the course of this effort. Additionally, this effort attempted an analogous verification using a model of the White Sands Missile Range (WSMR) FBR in cylindrical coordinates. However, this result is less impactful than the Godiva I comparison, as a 1D model of a cylindrical reactor will be inherently nonphysical. Nevertheless, a calibration against the experimental burst width and a comparison to the analytical solution was conducted. While the results of this WSMR FBR model are not predictive of the real reactor, some conclusions were still drawn regarding the nature and validity of the solution method.

1.3 Prior Work

Over the course of the 20th and 21st centuries, 13 FBRs were built in the United States, only 2 of which remain in use [7]. Research and design of these reactors was a popular topic in the 20th century, but their current rarity has led to reduced academic focus in recent years [1, 4]. The current research in this field tends to focus on the thermal and thermoelastic properties of FBRs [8, 9]. The result of this focus is that the neutronics of FBRs are the area of their study currently in the greatest need of refinement [8].

For typical reactors, these neutronics problems are frequently solved using the k eigenvalue, which reflects the criticality of a steady-state system and can be calculated and used to describe the reactor [10]. This solution method has also been applied to FBRs, with a recent study using the discrete ordinates code PENTRAN to model the WSMR FBR demonstrating that the solution method resulted in an overestimate of the criticality of the system [11]. Furthermore, the extremely transient nature of FBRs means that the time eigenvalue, which reflects the criticality of a time-varying system, is generally of greater use in describing the system, as the fundamental mode of the time eigenvalue translates to the inverse period of an FBR [12].

There are two common methods of solving for the time eigenvalue of a reactor: the first of these is to find the solution using a Monte Carlo code, such as MCNP. One such Monte Carlo method makes use of MCNP by discretizing the model space and taking neutron flux tallies in each cell, a process which itself can be computationally intensive as a large number of cells will necessitate a correspondingly large number of particles to ensure sufficient tally statistics. These tallies are then fed into a transition rate matrix method, producing "very large" matrices which must be stored and processed in order to produce an eigenfunction expansion to approximate the time eigenvalue [13]. This and other Monte Carlo methods of computing the time eigenvalue are therefore potentially quite computationally intensive, and, additionally, such methods do not lend themselves to readily coupling with other physics [13, 14]. For these reasons, a deterministic solution for the time eigenvalue of an FBR is preferable.

The second common solution method is deterministic in nature. It calls for iteratively solving for the k eigenvalue and updating the time eigenvalue based on the result. The most common method is to solve the diffusion equation for k, then "guess" a value of the time eigenvalue and adding a time absorption term, equal to $\frac{\alpha}{v}$, where

α is the time eigenvalue and v is the average neutron velocity, to the equation to bring the effective value of the k eigenvalue to unity. The updated equation is then solved again for k, and the process is repeated until a value of the time eigenvalue is found that produces a result of k = 1 [15, 16]. For models with a large number of cells, this repeated forming and solving of the diffusion equation matrix can result in significant computational and storage requirements.

This lack of an efficient method of solving for the time eigenvalue was the impetus for the Jacobian Free Newton-Krylov (JFNK) Alpha, and k Eigenvalue Solver (JAKES) [5]. This algorithm calculates a deterministic time eigenvalue solution using a nonlinear solution method, that includes employing the neutron diffusion equation for 1D solutions and the even-parity transport equation for 2D solutions. For a given spatial discretization, it initializes the solution with either a k eigenvalue and eigenvector calculation, which is paired with an assumed neutron lifetime to arrive at an initial value for the time eigenvalue, or else it directly calculates the fundamental mode of the time eigenvalue for the initial conditions through an iterative power method calculation. Previous results demonstrated that the k eigenvalue and eigenvector initialization resulted in the most accurate final solution [5]. Therefore this initial k-based value of the eigenvalue and eigenvector is used to initialize the JFNK solution, to ensure that the solution converges on the fundamental mode of alpha, rather than on a different, less relevant eigenvalue [5]. This method is the basis of this research effort.

JAKES was previously used to model theoretical, LEU fueled FBRs: one based on the Godiva I, using the 1D diffusion equation with the finite difference method, and the other based on the WSMR FBR, using the 2D transport equation with the finite element method [5]. Both of these FBRs were modeled with both LEU and HEU fuel. These models, including both fuel types, were then reproduced in a MCNP

Table 1. Previous findings for the Godiva I and WSMR FBR models using JAKES as presented in the original work [5].

	HEU			LEU	
	JAKES	MCNP[1]	Godiva[2]	JAKES	MCNP
Radius [cm]		8.741	8.692 – 8.741	21.562	21.562
Enrichment	93%	93.7%	93 – 94%	19.75%	19.75%
Mass [kg]	51.28	52.43	52 – 52.42	789.878	789.43
Volume [cm^3]	2735.188	2797.5	--	41990.953	41991.0
Cells	1000	--	--	1000	--
Analytical k Numerical k	1.000279 1.000280	-- 0.995	-- --	1.0010736 1.0010742	-- 0.995
Analytical α $[\mu sec]^{-1}$	0.1032	--	--	7.455E-2	--
Numerical α $[\mu sec]^{-1}$	0.1035	--	--	7.458E-2	--
n-lifetime[3] [μ-sec]	2.706E-3	5.729E-3	--	1.440E-2	3.298E-2
FWHM [μ-sec]	34.034	--	35 – 50	47.246	
Max	1.0	--	--	2.76	
Max/ volume	1.0	--	--	0.180	
Max/ mass	1.0	--	--	0.179	
CPU-time [sec]	0.131	111.2	--	0.133	380

	HEU 10%Moly			LEU 10%Moly	LEU 1.5%Moly
	JAKES-EVENT	MolyG	MCNP	EVENT	EVENT
Enrichment	93.2%	93.2%	93.2	19.75%	19.75%
Radius	10.540	10.3	10.3	24.57	21.076
Height	19.752	19.3	19.3	46.044	39.496
Numerical k	1.000882	--	1.09	1.0027996	1.0024298
Numerical α $[\mu sec]^{-1}$	8.4869E-2	--	--	5.937E-2	5.987E-2
n-lifetime [μ-sec]	1.0397E-2	--	8.9108E-3	4.716E-2	4.059E-2
FWHM [μ-sec]	41.5228	35 – 50	--	59.359	58.864
Max	1.0	--	--	2.220	1.943
k calculations (if JAKES not used)	38	--	--	64	64
alpha updates	1	--	--	1	1
Newton Iterations/alpha update	1	--	--	1	1
GMRES iterations/Newton Iteration	9	--	--	12	30

model for comparison and to verify the results. This verification against a stochastic method was limited in scope, however, as the effort only used MCNP to calculate the k eigenvalue of each model, having refrained from employing the computationally intensive method of calculating the time eigenvalue using Monte Carlo method as discussed above [5]. The results of this verification are listed in Table 1, with the finite difference case having been run for N=1,000 nodes, which shows that the JAKES calculation of the k eigenvalue agreed with the MCNP results for the Godiva I case to within 0.5%. Similarly, the WSMR FBR k eigenvalue results agreed to within 8% [5]. While these results show promise, the comparison of k eigenvalue results is not an ideal source of verification for FBR systems, as it is a result of the steady-state diffusion equation, while the FBR is, by definition, a highly transient system. Therefore the time eigenvalue is a preferable metric for the verification of results [12].

Unfortunately, the only verification conducted for the JAKES algorithm's calculation of the time eigenvalue was a comparison of the 1D, spherical finite difference calculation of the Godiva I based model to an analytic solution. The results of this comparison are also shown in Table 1, with an error of 3 x 10^{-4} μs^{-1}, or 0.3% [5]. While this was a promising result, it did not constitute sufficient verification of the JFNK solution for the time eigenvalue [5]. Therefore this effort expanded the verification for the time eigenvalue, applying the 1D diffusion equation solution in cylindrical and Cartesian coordinates, in addition to spherical coordinates, allowing for comparison against analytic solutions and experimental burst width data for both the Godiva I and WSMR FBRs.

II. Theory

2.1 Fast Burst Reactors

The FBRs modeled in this effort are super prompt critical systems assembled by the rapid insertion of excess reactivity into a sub-critical assembly [1]. The reactivity excursions produced by these assemblies typically occur over the course of 10's to 100's of microseconds [2]. The length of these excursions are limited because of the self-quenching property of these reactors: as fissions occur, the energy released heats the fuel of the reactor, this temperature increase results in thermal expansion in the fuel, reducing the density of the fuel and therefore reducing the macroscopic fission cross-section, resulting in fewer fissions and causing a drop off in both neutron and power production [2]. After this self-quenching reduces the power production, a safety block assembly is removed from the FBR in order to scram the reactor [2]. This self-limiting property is what makes FBRs a desirable source of neutron bursts.

The FBRs of interest to this effort are the Godiva I and WSMR FBRs. These two were selected because of the relative simplicity of their designs, which allows models with both analytical and deterministic solutions, enabling an absolute error determination to be made for verification of the solution method. The Godiva I, shown in Figure 1, is a spherical reactor comprised of homogeneous HEU fuel. The figure shows the sphere separated into three, sub-critical pieces on the assembly which was used to assemble the FBR into its super critical state [17]. The WSMR FBR was a slightly later and more advanced design, making use of a fuel of HEU alloyed with molybdenum. It has a cylindrical design, shown in Figure 2, which was based on the design of a successor to the Godiva I FBR, leading to it also being known as the "Moly-G" FBR for "Molybdenum Godiva" [1]. The homogeneity of these reactors is what makes them easier to model, as the model is free of material boundaries, leaving

Figure 1. Image of the Godiva I FBR in its disassembled state [17].

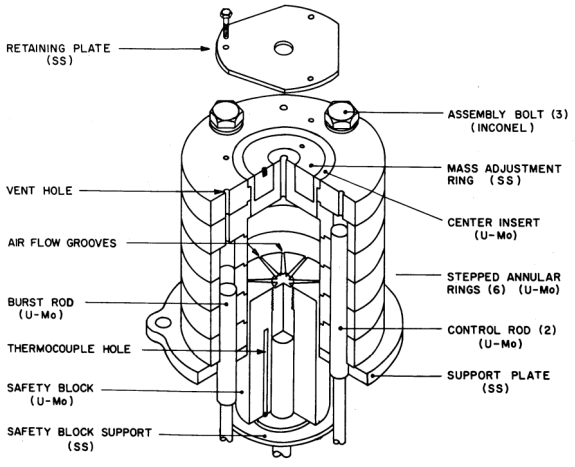

Figure 2. Schematic of the WSMR FBR design [1].

only the boundary conditions at the edges of the fuel assemblies. Additionally, the geometry of Godiva I lends itself well to a 1D model, which is the very reason why it was chosen for use in the original JAKES study [5]. The properties of these reactors are listed in Table 2.

2.2 The Neutron Diffusion Equation

In modeling these systems, this effort sought only to generate a solution to the time eigenvalue for a one dimensional model of a FBR. This effort implemented the ability to model FBRs in 1D in spherical, cylindrical, and Cartesian coordinates into JAKES. Of these three coordinate systems, only spherical geometry produces a result which could fully represent a physical system. This is because if the lone finite dimension is taken to be the radial dimension, then symmetry can be applied across both angular dimensions of the spherical geometry in order to produce a finite, and therefore physical, sphere. The same cannot be said for cylindrical or Cartesian geometries: in cylindrical geometry, angular symmetry can be applied in only one dimension, leaving one dimension infinite; in Cartesian, no angular symmetry can be applied, resulting in two infinite dimensions. A model with one or more infinite dimensions is inherently nonphysical, and therefore cannot be used to represent a real FBR. However, these geometries are still pursued in this effort as a means of further

Table 2. Properties of the Godiva I and WSMR FBRs as used in the original development of JAKES [5].

Reactor	Godiva I	WSMR FBR
Geometry	Spherical	Cylindrical
Radius (cm)	8.6756	10.540
Density (g/cm^3)	18.75	17.90
Mass (kg)	52 - 52.42	97.142
Fuel Composition	100 w/o HEU	90 w/o HEU, 10 w/o Mo
Experimental Burst Width (μs)	35 - 50	31 - 50

verification of the JAKES algorithm, and as a step towards higher dimension models which do not produce non-physicalities.

In addition to these geometric assumptions, this effort also makes simplifying assumptions regarding neutron energies. The systems of interest to this effort are exclusively fast reactors and, as such, all neutrons in the system can be assumed to be born in the fast regime. Both the Godiva I and WSMR FBRs are comprised of homogeneous, metal fuel with no additional moderating material introduced. These pure - or near pure in the case of the WSMR FBR - uranium metal assemblies have minimal cross-sections for moderation [18]. Therefore, it assumed for ease of calculation that all neutrons in the system share a single, average energy. For consistency with previous work, this neutron energy value is chosen to be 1.45 MeV [5]. The isotopic cross-sections for this energy, as used in this effort, are listed in Table 3.

Finally, this effort assumes the applicability of Fick's law, given by [19]:

$$\vec{J} = -D\nabla\phi \tag{1}$$

where $\vec{J}$ is the neutron current, ϕ is the neutron flux, and the diffusion coefficient, D, is given by

$$D = \frac{1}{3(\sigma_t + \frac{\alpha}{v})} \tag{2}$$

where σ_t is the macroscopic transport cross-section.

These assumptions allow the system to be described by the one dimensional, one

Table 3. Isotopic microscopic cross-section values as used in the original development of JAKES [5].

	ν (neutrons)	β	$\tilde{\sigma_f}$ (b)	$\tilde{\sigma_a}$ (b)	$\tilde{\sigma_{tr}}$ (b)
U^{235}	2.60	0.0065	1.40	1.65	6.80
U^{238}	2.60	0.0157	0.095	0.255	6.90

group, time-dependent neutron diffusion equation:

$$\frac{1}{v}\frac{\partial\phi(r,t)}{\partial t} - D\nabla^2\phi(r,t) + \sigma_a\phi(r,t) = \nu\sigma_f\phi(r,t) \tag{3}$$

where σ_a and σ_f are the macroscopic absorption and fission cross-sections, and ν is the average number of neutrons produced per fission. For spherical geometry, the Laplacian, ∇^2, is defined as [20]

$$\nabla^2\phi = \frac{1}{r^2}\frac{\partial}{\partial r}(r^2\frac{\partial\phi}{\partial r}) + \frac{1}{r^2\sin\theta}\frac{\partial}{\partial\theta}(\sin\theta\frac{\partial\phi}{\partial\theta}) + \frac{1}{r^2\sin^2\theta}\frac{\partial^2\phi}{\partial\varphi} \tag{4}$$

which in 1D reduces to

$$\nabla^2\phi = \frac{\partial^2\phi}{\partial r^2} + \frac{2}{r}\frac{\partial\phi}{\partial r}. \tag{5}$$

In cylindrical geometry the Laplacian is defined as [20]

$$\nabla^2\phi = \frac{1}{r}\frac{\partial}{\partial r}(r\frac{\partial\phi}{\partial r}) + \frac{1}{r^2}\frac{\partial^2\phi}{\partial\varphi^2} + \frac{\partial^2\phi}{\partial z^2} \tag{6}$$

which in 1D reduces to

$$\nabla^2\phi = \frac{\partial^2\phi}{\partial r^2} + \frac{1}{r}\frac{\partial\phi}{\partial r}. \tag{7}$$

And in Cartesian coordinates the 1D Laplacian is simply [20]

$$\nabla^2\phi = \frac{\partial^2\phi}{\partial r^2}. \tag{8}$$

This difference in the definition of the Laplacian is the only term in the diffusion equation in which the three coordinate systems used in this effort differ. Therefore,

a coordinate constant, a, is defined, where:

$$a = \begin{cases} 2 & \text{Spherical} \\ 1 & \text{Cylindrical} \\ 0 & \text{Cartesian} \end{cases} \tag{9}$$

Using this constant, a generic Laplacian can be defined and used with any coordinate system:

$$\nabla^2\phi = \frac{\partial^2\phi}{\partial r^2} + \frac{a}{r}\frac{\partial\phi}{\partial r}. \tag{10}$$

Using this generic definition of the Laplacian, the diffusion equation, in the form shown in Equation 3, can then be rewritten as

$$\frac{1}{v}\frac{\partial\phi(r,t)}{\partial t} - D\left(\frac{\partial^2\phi}{\partial r^2} + \frac{a}{r}\frac{\partial\phi}{\partial r}\right) + \sigma_a\phi(r,t) = \nu\sigma_f\phi(r,t). \tag{11}$$

The flux can then be split into its radial component and a time component, from which the time eigenvalue originates [19].

$$\phi(r,t) = \phi(r)e^{\alpha t} \tag{12}$$

Substituting these separated values into Equation 11 gives a result of

$$\begin{aligned} \frac{1}{v}\frac{\partial}{\partial t}\left(\phi(r)e^{\alpha t}\right) - D\left[\frac{\partial^2}{\partial r^2}\left(\phi(r)e^{\alpha t}\right) + \frac{a}{r}\frac{\partial}{\partial r}\left(\phi(r)e^{\alpha t}\right)\right] + \sigma_a\left(\phi(r)e^{\alpha t}\right) \\ = \nu\sigma_f\left(\phi(r)e^{\alpha t}\right) \end{aligned} \tag{13}$$

which reduces to

$$\frac{\alpha}{v}\phi(r)e^{\alpha t} - D\left(\frac{\partial^2\phi}{\partial r^2} + \frac{a}{r}\frac{\partial\phi}{\partial r}\right)e^{\alpha t} + \sigma_a\phi(r)e^{\alpha t} = \nu\sigma_f\phi(r)e^{\alpha t}. \tag{14}$$

The exponential term can then be factored out of the equation, leaving equation as solely a function of r:

$$\frac{\alpha}{v}\phi(r) - D\left(\frac{\partial^2\phi}{\partial r^2} + \frac{a}{r}\frac{\partial\phi}{\partial r}\right) + \sigma_a\phi(r) = \nu\sigma_f\phi(r). \tag{15}$$

This result is useful as it reduces all dependence on time in the equation to the time eigenvalue, α. The equation can then be solved numerically, such as with the JFNK method, or analytically to arrive at a value for this time eigenvalue and a corresponding flux eigenvector. The analytical solution to this form of the diffusion equation is presented for each coordinate system in Appendix A.

An alternative, simplified version of this equation is the steady-state neutron diffusion equation [10].

$$-D\nabla^2\phi(r) + \sigma_a\phi(r) = \frac{1}{k}\nu\sigma_f\phi(r) \tag{16}$$

In this form, there is no time dependence, and the k eigenvalue is used to represent different steady-states at which the system can operate. This simplified form of the neutron diffusion equation can be solved and used in the calculation of the time eigenvalue solution. Using the definition of the Laplacian given in Equation 10, this version of the diffusion equation can be rewritten as

$$-D\left(\frac{\partial^2\phi}{\partial r^2} + \frac{a}{r}\frac{\partial\phi}{\partial r}\right) + \sigma_a\phi(r) = \frac{1}{k}\nu\sigma_f\phi(r). \tag{17}$$

Just as Equation 15 can be solved for the time eigenvalue, so can Equation 17 be solved for the k eigenvalue. The numerical solution method used in this effort, the power iteration, is presented in Section 3.2, and the analytical solution is given in Appendix B.

Both of these versions of the diffusion equation require boundary conditions before they can be solved. These boundary conditions describe the flux of the system at the edges of the model. As the FBRs of interest to this effort, the Godiva I and WSMR FBRs, are both bare assemblies, the outer boundary condition can be taken to be an extrapolated vacuum boundary. At a diffusion vacuum boundary, the flux is equal to zero [10].

$$\phi(R) = 0 \tag{18}$$

However, R here does not occur at the outer edge of the reactor, as there would still be fissions occurring, producing a non-zero flux throughout the FBR. R, therefore, is defined using the extrapolated boundary condition, which is given by [10]:

$$R_{\text{extrapolated}} = R_{\text{reactor}} + 2D \tag{19}$$

In order to simplify the problem, this boundary is defined using the value of D from the steady-state diffusion equation, which assumes $\alpha = 0$. The value of R calculated here is then used throughout both the k and time eigenvalue calculations. This ignores changes in D, which is dependent on α and therefore varies with each iteration of the JFNK solution. This could then be used to calculate a new extrapolated boundary after each update of the value of D. If this update were applied to the extrapolated boundary at each iteration, then the size of the model, and therefore potentially the size of the eigenvector, would change with each iteration. To avoid this complication, the value of R is calculated only once in the JAKES algorithm, neglecting the α term in the definition of D, and then used throughout the calculations of both the k and time eigenvalues.

Conversely, the inner boundary represents the point at r = 0. This boundary, then, represents the center of the FBR. The use of 1D spherical geometry, in particular,

to represent a symmetric assembly lends itself to the use of a reflective boundary condition at the center of the system, reflecting the symmetry that should exist at the center of the sphere. This symmetry in flux can be represented by a zero net rate of change in flux across the center of the model, as shown in Equation 20 [10].

$$\frac{d\phi}{dr} \mid_{r=0} = 0 \tag{20}$$

These boundary conditions, coupled with Equation 15 or 17, represent a complete system that can then be solved analytically or numerically.

2.3 The Finite Difference Method

The finite difference method is a spatial discretization scheme in which the spatial domain is discretized into nodes [20]. Each node is a finite point in the geometry of the system at which the flux can be approximated in relation to its neighboring points. These relations are accomplished by approximating the derivatives of the flux using Taylor expansions [20]. These expansions are used to define the flux and its derivatives at each point in the discretization, defined here as spanning from i = 0 to N, where node 0 is located at position r = 0, and node N is located at position r = R, the outer boundary of the model. Node i is located at r = i $\times$ Δr, where Δr = R/N is the distance between each node. This allows us to form a discretized version of Equation 15, where $\phi(r_i)$ is written as ϕ_i:

$$vD\left(\frac{\partial^2\phi}{\partial r^2}|_i + \frac{a}{r_i}\frac{\partial\phi}{\partial r}|_i\right) + (\nu\sigma_f - \sigma_a)\phi_i = \alpha\phi_i. \tag{21}$$

The relevant Taylor expansions for this equation are:

$$\phi_{i+1} = \phi_i + \Delta r\frac{d\phi}{dr}|_i + \frac{1}{2!}(\Delta r)^2\frac{d^2\phi}{dr^2}|_i + \frac{1}{3!}(\Delta r)^3\frac{d^3\phi}{dr^3}|_i + \frac{1}{4!}(\Delta r)^4\frac{d^4\phi}{dr^4}|_i + O(\Delta r^5)$$

$$\phi_{i-1} = \phi_i - \Delta r \frac{d\phi}{dr}|_i + \frac{1}{2!}(\Delta r)^2 \frac{d^2\phi}{dr^2}|_i - \frac{1}{3!}(\Delta r)^3 \frac{d^3\phi}{dr^3}|_i + \frac{1}{4!}(\Delta r)^4 \frac{d^4\phi}{dr^4}|_i + O(\Delta r^5)$$

Where $O(\Delta r^5)$ refers to the order of error in the equation, as detailed in Appendix D. Subtracting these expansions results in

$$\phi_{i+1} - \phi_{i-1} = 0 + 2\Delta r \frac{d\phi}{dr}|_i + 0 + 2\frac{1}{3!}(\Delta r_i)^3 \frac{d^3\phi}{dr^3}|_i + 0 + O(\Delta r^5)$$

which can be rearranged to form a definition of the first derivative of flux at node i:

$$\frac{d\phi}{dr}|_i = \frac{\phi_{i+1} - \phi_{i-1}}{2\Delta r} + O(\Delta r^3). \tag{22}$$

Similarly, adding the expansions together results in

$$\phi_{i+1} + \phi_{i-1} = 2\phi_i + 0 + \frac{1}{2!}(\Delta r_i)^2 \frac{d^2\phi}{dr^2}|_i + 0 + O(\Delta r^4)$$

which can, in turn, be rearranged to form a definition of the second derivative of flux at node i:

$$\frac{d^2\phi}{dr^2}|_i = \frac{\phi_{i+1} - 2\phi_i + \phi_{i-1}}{\Delta r^2} + O(\Delta r^2). \tag{23}$$

Substituting these values into Equation 21 yields the following second order accurate discretized equation:

$$\frac{\alpha}{v}\phi_i - D\left[\left(\frac{\phi_{i+1} - 2\phi_i + \phi_{i-1}}{\Delta r^2}\right) + \frac{a}{r_i}\left(\frac{\phi_{i+1} - \phi_{i-1}}{2\Delta r}\right)\right] + \sigma_a\phi_i = \nu\sigma_f\phi_i \tag{24}$$

This equation can be rearranged into the following form

$$\begin{aligned}\left(\frac{vD}{\Delta r^2} + \frac{avD}{2r_i\Delta r}\right)\phi_{i+1} + \left(-\frac{2vD}{\Delta r^2} + v(\nu\sigma_f - \sigma_a)\right)\phi_i \\ + \left(\frac{vD}{\Delta r^2} - \frac{avD}{2r_i\Delta r}\right)\phi_{i-1} = \alpha\,\phi_i.\end{aligned} \tag{25}$$

This same method can be applied to the steady-state diffusion equation to produce the following result:

$$\left(\frac{-D}{\nu\sigma_f\Delta r^2} - \frac{aD}{2\nu\sigma_f r_i\Delta r}\right)\phi_{i+1} + \left(\frac{2D}{\nu\sigma_f\Delta r^2} + \frac{\sigma_a}{\nu\sigma_f}\right)\phi_i + \left(\frac{-D}{\nu\sigma_f\Delta r^2} + \frac{aD}{2\nu\sigma_f r_i\Delta r}\right)\phi_{i-1} = \frac{1}{k}\phi_i \tag{26}$$

The derivation of the steady-state diffusion equation discretization which produces Equation 26 is given in Appendix C. Both Equation 25 and Equation 26 are given in forms which can be readily converted into a tridiagonal matrix eigenvalue problem in the form of

$$A\,\vec{\phi} = \alpha\,\vec{\phi} \tag{27}$$

where A is the matrix resulting from the terms on left side of the discretized equations. Equation 27 represents the time eigenvalue problem, but the steady-state diffusion equation version takes the same form, with α being replaced with $\frac{1}{k}$.

These equations define the interior nodes, but those nodes at the boundaries of the model require different equations to account for the boundary conditions. This is where the finite difference discretization implemented in this effort differs from the discretization originally implemented in JAKES. The original JAKES discretization used identical equations to describe the interior nodes, however it described the boundaries using those same equations, rather than applying unique equations for each of the boundaries [5].

The first of these boundary conditions is the outer, vacuum boundary at r = R, given by Equation 18. This boundary is defined at node N, where $\phi_N = 0$. This would introduce an unnecessary non-linearity into the matrix, complicating the numerical solution. Therefore, the finite difference discretization ends at node (N - 1), at which

Equation 21 becomes

$$vD\left(\frac{\partial^2\phi}{\partial r^2}|_{N-1}+\frac{a}{r_{N-1}}\frac{\partial\phi}{\partial r}|_{N-1}\right)+v(\nu\sigma_f-\sigma_a)\phi_{N-1}=\alpha\phi_{N-1} \tag{28}$$

Writing this equation in terms of node fluxes requires defining $\frac{d\phi}{dr}|_{N-1}$ using Equation 22 and the boundary condition that $\phi_N = 0$:

$$\frac{d\phi}{dr}|_{N-1}=\frac{\phi_N-\phi_{N-2}}{2\Delta r}+O(\Delta r^2)=\frac{(0)-\phi_{N-2}}{2\Delta r}+O(\Delta r^2) \tag{29}$$

Likewise, Equation 23, combined with the boundary condition, results in the following equation at this node:

$$\frac{d^2\phi}{dr^2}|_{N-1}=\frac{\phi_N-2\phi_{N-1}+\phi_{N-2}}{\Delta r^2}+O(\Delta r^2)=\frac{(0)-2\phi_{N-1}+\phi_{N-2}}{\Delta r^2}+O(\Delta r^2) \tag{30}$$

Substituting these results into Equation 28 gives

$$vD\left[\left(\frac{-2\phi_{N-1}+\phi_{N-2}}{\Delta r^2}\right)+\frac{a}{r_{N-1}}\left(\frac{-\phi_{N-2}}{2\Delta r}\right)\right]+v(\nu\sigma_f-\sigma_a)\phi_{N-1}=\alpha\phi_{N-1}$$

which simplifies to

$$\left(-\frac{2vD}{\Delta r^2}+v(\nu\sigma_f-\sigma_a)\right)\phi_{N-1}+\left(\frac{vD}{\Delta r_i^2}-\frac{avD}{2r_{N-\frac{1}{2}}\Delta r}\right)\phi_{N-2}=\alpha\,\phi_{N-1}. \tag{31}$$

Similarly, the inner, reflective boundary at r = 0, given by Equation 20, occurs at node 0, at which Equation 21 becomes

$$vD\left(\frac{\partial^2\phi}{\partial r^2}|_0+\frac{a}{r_0}\frac{\partial\phi}{\partial r}|_0\right)+v(\nu\sigma_f-\sigma_a)\phi_0=\alpha\phi_0 \tag{32}$$

Here both r_0 and $\frac{\partial \phi}{\partial r}|_0$ go to zero. Therefore L'Hospital's Rule must be applied [20]:

$$\lim_{x \to 0} \frac{1}{r} \frac{\partial \phi}{\partial r} = \lim_{x \to 0} \frac{\frac{d}{dr}\left(\frac{\partial \phi}{\partial r}\right)}{\frac{d}{dr}(r)} = \frac{\partial^2 \phi}{\partial r^2}|_0$$

Equation 32 then becomes:

$$vD\left[\frac{\partial^2 \phi}{\partial r^2}|_0 + a\left(\frac{\partial^2 \phi}{\partial r^2}|_0\right)\right] + v(\nu\sigma_f - \sigma_a)\phi_0 = \alpha\phi_0 \tag{33}$$

As with the previous case, this boundary also requires a unique term definition, this time for $\frac{d^2\phi}{dr^2}|_0$, which is accomplished by combining Equation 23 with the reflective nature of this boundary. The symmetry inherent to this boundary means that the flux at r_i is the same as the flux at -r_i, therefore ϕ_{-1} can be used as a ghost point with a value equal to ϕ_1, allowing for the following definition:

$$\frac{d^2\phi}{dr^2}|_0 = \frac{\phi_1 - 2\phi_0 + \phi_{-1}}{\Delta r^2} + O(\Delta r^2) = \frac{\phi_1 - 2\phi_0 + (\phi_1)}{\Delta r^2} + O(\Delta r^2) = \frac{2(\phi_1 - \phi_0)}{\Delta r^2} + O(\Delta r^2) \tag{34}$$

Substituting this result into Equation 33 gives

$$vD\left[\left(\frac{2(\phi_1 - \phi_0)}{\Delta r^2}\right) + a\left(\frac{2(\phi_1 - \phi_0)}{\Delta r^2}\right)\right] + v(\nu\sigma_f - \sigma_a)\phi_0 = \alpha\phi_0$$

which simplifies to

$$\left(\frac{2vD(1+a)}{\Delta r^2}\right)\phi_1 + \left(-\frac{2vD(1+a)}{\Delta r^2} + v(\nu\sigma_f - \sigma_a)\right)\phi_0 = \alpha\,\phi_0. \tag{35}$$

Together Equations 25, 31, and 35 form the full finite difference discretization. The corresponding discretization for the steady-state diffusion equation is derived in Appendix C and is also second order accurate.

2.4 The Finite Volume Method

The finite volume method is a spatial discretization scheme in which the spatial domain is discretized into cells [20]. Each cell is integrated over a finite subset of space, in the 1D case over a finite segment of r. For spherical coordinates, when rotational symmetry is applied to the 1D cells, the result is a series of concentric shells, as shown in Figure 3. This integration of the diffusion equation results in a volume averaged value of flux for each cell, with the convenient property that any decrease in flux in one cell must be offset by an increase in an adjoining cell, such that flux is always conserved [20]. All other properties are also cell averaged, but because the FBRs modeled in this effort are both homogeneous, this does not change any of the properties from what they would be in a corresponding finite difference discretization. Were non-homogeneous FBRs modeled here, this cell integration could produce cross-section values which differed from those of a finite difference scheme.

For the sake of consistency, the finite volume discretization presented here uses the same indexing system as the finite difference discretization. Each node in the finite difference scheme is a cell boundary, with cells being defined by their centroid point. So the cell located between nodes i and (i+1) is defined as cell (i+$\frac{1}{2}$). This indexing system is illustrated in Figure 4. Using these definitions, Equation 15 can be integrated across cell (i+$\frac{1}{2}$) in Cartesian coordinates, resulting in the following:

$$\int_{r_i}^{r_{i+1}} \left[vD \left(\frac{d^2\phi}{dr^2} + \frac{a}{r}\frac{d\phi}{dr} \right) + v(\nu\sigma_f - \sigma_a)\phi \right] dr = \int_{r_i}^{r_{i+1}} (\alpha\, \phi) \, dr$$

$$vD \left[\frac{d\phi}{dr}|_{i+1} - \frac{d\phi}{dr}|_i + \frac{a}{r_{i+\frac{1}{2}}}(\phi_{i+1} - \phi_i) \right] + v(\nu\sigma_f - \sigma_a)\phi_{i+\frac{1}{2}}\Delta r = \alpha\, \phi_{i+\frac{1}{2}}\Delta r \quad (36)$$

The terms in this equation can be defined in terms of adjoining cells using Taylor expansions of the flux at each cell, in the same way the finite difference discretization

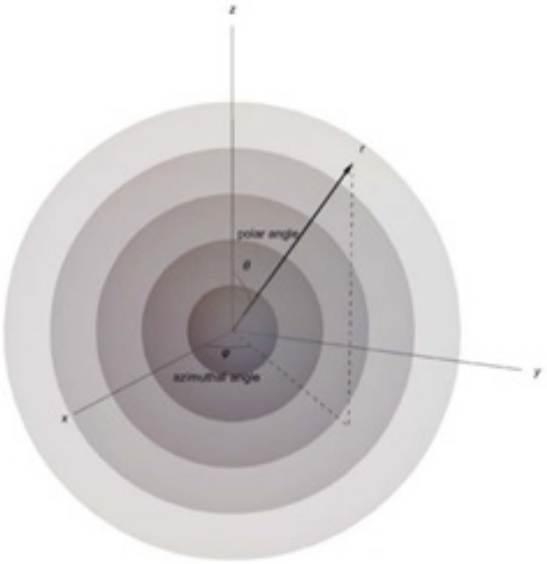

Figure 3. Visualization of the spherical finite volume cells in 3D.

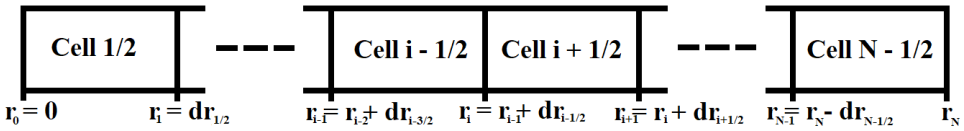

Figure 4. Visual representation of the finite volume indexing scheme used in this discretization of the neutron diffusion equation.

defined derivatives in terms of neighboring nodes. The Taylor expansions relevant to this discretization are:

$$\phi_{i+\frac{1}{2}} = \phi_i + \frac{\Delta r}{2}\frac{d\phi}{dr}|_i + \frac{1}{2!}\left(\frac{\Delta r}{2}\right)^2 \frac{d^2\phi}{dr^2}|_i + \frac{1}{3!}\left(\frac{\Delta r}{2}\right)^3 \frac{d^3\phi}{dr^3}|_i + \frac{1}{4!}\left(\frac{\Delta r}{2}\right)^4 \frac{d^4\phi}{dr^4}|_i + O(\Delta r^5)$$

$$\phi_{i-\frac{1}{2}} = \phi_i - \frac{\Delta r}{2}\frac{d\phi}{dr}|_i + \frac{1}{2!}\left(\frac{\Delta r}{2}\right)^2 \frac{d^2\phi}{dr^2}|_i - \frac{1}{3!}\left(\frac{\Delta r}{2}\right)^3 \frac{d^3\phi}{dr^3}|_i + \frac{1}{4!}\left(\frac{\Delta r}{2}\right)^4 \frac{d^4\phi}{dr^4}|_i + O(\Delta r^5)$$

Subtracting these two equations yields

$$\phi_{i+\frac{1}{2}} - \phi_{i+\frac{1}{2}} = 0 + 2\frac{\Delta r}{2}\frac{d\phi}{dr}|_i + 0 + 2\frac{1}{3!}\left(\frac{\Delta r}{2}\right)^3 \frac{d^3\phi}{dr^3}|_i + 0 + O(\Delta r^5)$$

which can be rearranged to give a definition of the first derivative of flux at r_i, the boundary between cells (1+$\frac{1}{2}$) and (1-$\frac{1}{2}$):

$$\frac{d\phi}{dr}|_i = \frac{\phi_{i+\frac{1}{2}} - \phi_{i-\frac{1}{2}}}{\Delta r} + O(\Delta r^3) \tag{37}$$

Similarly, adding the two Taylor expansion equations yields

$$\phi_{i+\frac{1}{2}} + \phi_{i-\frac{1}{2}} = 2\phi_i + 0 + 2\frac{1}{2!}\left(\frac{\Delta r}{2}\right)^2 \frac{d^2\phi}{dr^2}|_i + 0 + O(\Delta r^4)$$

which results in a definition of the flux at the cell boundary as a function of the two adjoining cells:

$$\phi_i = \frac{\phi_{i+\frac{1}{2}} + \phi_{i-\frac{1}{2}}}{2} + O(\Delta r^2) \tag{38}$$

Substituting these values into the integrated diffusion equation, Equation 36, yields:

$$vD\left[\left(\frac{\phi_{i+\frac{3}{2}} - \phi_{i+\frac{1}{2}}}{\Delta r}\right) - \left(\frac{\phi_{i+\frac{1}{2}} - \phi_{i-\frac{1}{2}}}{\Delta r}\right) + \frac{a}{r_{i+\frac{1}{2}}}\left(\left(\frac{\phi_{i+\frac{3}{2}} + \phi_{i+\frac{1}{2}}}{2}\right) - \left(\frac{\phi_{i+\frac{3}{2}} + \phi_{i+\frac{1}{2}}}{2}\right)\right)\right]$$
$$+ (\nu\sigma_f - \sigma_a)\phi_{i+\frac{1}{2}}\Delta r \;=\; \alpha\,\phi_{i+\frac{1}{2}}\Delta r$$

which simplifies to the following second order accurate equation:

$$\left(\frac{vD}{\Delta r_i^2}+\frac{avD}{2r_{i+\frac{1}{2}}\Delta r}\right)\phi_{i+\frac{3}{2}}+\left(-\frac{2vD}{\Delta r^2}+v(\nu\sigma_f-\sigma_a)\right)\phi_{i+\frac{1}{2}} + \left(\frac{vD}{\Delta r^2}-\frac{avD}{2r_{i+\frac{1}{2}}\Delta r}\right)\phi_{i-\frac{1}{2}}=\alpha\,\phi_{i+\frac{1}{2}} \tag{39}$$

If the indexes in this equation are all subtracted by $\frac{1}{2}$, then this result for Cartesian coordinates becomes identical to the finite difference discretization given in Equation 25. However, because the integration used in this finite volume discretization is specific to Cartesian coordinates, there is no need to use the multi-coordinate definition of the Laplacian. The variable a may therefore be substituted for 0.

$$\left(\frac{vD}{\Delta r_i^2}\right)\phi_{i+\frac{3}{2}}+\left(-\frac{2vD}{\Delta r^2}+v(\nu\sigma_f-\sigma_a)\right)\phi_{i+\frac{1}{2}}+\left(\frac{vD}{\Delta r^2}\right)\phi_{i-\frac{1}{2}}=\alpha\,\phi_{i+\frac{1}{2}} \tag{40}$$

In Cartesian coordinates, for which a=0 in the finite difference discretization as well, the finite difference and finite volume methods are identical for all interior points of the model and its corresponding matrix values, with both having second order accuracy. These two discretizations therefore only differ in their handling of the boundary conditions of the model.

The first boundary condition discretized with the finite volume method in Cartesian coordinates is the outer, vacuum boundary at r = R, given by Equation 18. This boundary occurs at the outer edge of cell (N - $\frac{1}{2}$), at which Equation 36 becomes

$$vD\left(\frac{d\phi}{dr}|_N-\frac{d\phi}{dr}|_{N-1}\right)+v(\nu\sigma_f-\sigma_a)\phi_{N-\frac{1}{2}}\Delta r=\alpha\,\phi_{N-\frac{1}{2}}\Delta r \tag{41}$$

Writing this equation in terms of cell fluxes requires defining $\frac{d\phi}{dr}|_N$, which can be found

from the following Taylor expansions

$$\phi_{N-\frac{1}{2}} = \phi_N + \frac{\Delta r}{2}\frac{d\phi}{dr}|_N + \frac{1}{2!}\left(\frac{\Delta r}{2}\right)^2 \frac{d^2\phi}{dr^2}|_N + \frac{1}{3!}\left(\frac{\Delta r}{2}\right)^3 \frac{d^3\phi}{dr^3}|_N + O(\Delta r^4)$$

$$\phi_{N-\frac{3}{2}} = \phi_N - \frac{3\Delta r}{2}\frac{d\phi}{dr}|_N + \frac{1}{2!}\left(\frac{3\Delta r}{2}\right)^2 \frac{d^2\phi}{dr^2}|_N - \frac{1}{3!}\left(\frac{3\Delta r}{2}\right)^3 \frac{d^3\phi}{dr^3}|_N + O(\Delta r^4)$$

which, noting that $\phi_N = 0$, can be combined as

$$\phi_{N-\frac{3}{2}} - 9\phi_{N-\frac{1}{2}} = (0) + (-3+9)\frac{\Delta r}{2}\frac{d\phi}{dr}|_N + (0) + O(\Delta r^3).$$

This results in a definition of:

$$\frac{d\phi}{dr}|_N = \frac{\phi_{N-\frac{3}{2}} - 9\phi_{N-\frac{1}{2}}}{3\Delta r} + O(\Delta r^2). \tag{42}$$

Substituting this result into Equation 41 gives

$$vD\left[\left(\frac{\phi_{N-\frac{3}{2}} - 9\phi_{N-\frac{1}{2}}}{3\Delta r}\right) - \left(\frac{\phi_{N-\frac{1}{2}} - \phi_{N-\frac{3}{2}}}{\Delta r}\right)\right] + v(\nu\sigma_f - \sigma_a)\phi_{N-\frac{1}{2}}\Delta r = \alpha\,\phi_{N-\frac{1}{2}}\Delta r$$

which simplifies to

$$\left(-\frac{4vD}{\Delta r^2} + v(\nu\sigma_f - \sigma_a)\right)\phi_{N-\frac{1}{2}} + \left(\frac{4vD}{3\Delta r_i^2}\right)\phi_{N-\frac{3}{2}} = \alpha\,\phi_{N-\frac{1}{2}}. \tag{43}$$

Similarly, the inner, reflective boundary at r = 0, given by Equation 20, occurs at the inner edge of cell $\frac{1}{2}$, at which Equation 36 becomes

$$vD\left(\frac{d\phi}{dr}|_1 - \frac{d\phi}{dr}|_0\right) + v(\nu\sigma_f - \sigma_a)\phi_{\frac{1}{2}}\Delta r = \alpha\,\phi_{\frac{1}{2}}\Delta r \tag{44}$$

which, noting that $\frac{d\phi}{dr}|_0 = 0$ due to the reflective boundary, gives

$$vD\left[\left(\frac{\phi_{\frac{3}{2}} - \phi_{\frac{1}{2}}}{\Delta r}\right) - (0)\right] + v(\nu\sigma_f - \sigma_a)\phi_{\frac{1}{2}}\Delta r = \alpha\,\phi_{\frac{1}{2}}\Delta r$$

which simplifies to

$$\left(\frac{vD}{\Delta r^2}\right)\phi_{\frac{3}{2}} + \left(-\frac{vD}{\Delta r^2} + v(\nu\sigma_f - \sigma_a)\right)\phi_{\frac{1}{2}} = \alpha\,\phi_{\frac{1}{2}}. \tag{45}$$

Together Equations 40, 43, and 45 form the full finite volume discretization in Cartesian coordinates. The corresponding discretization of the steady-state diffusion equation is derived in Appendix C. Both of these discretizations are second order accurate.

The 1D integrals for curvilinear coordinates are slightly more complicated than that for Cartesian coordinates, however. The spherical integration of Equation 15 across cell (i+$\frac{1}{2}$) is given by [19]:

$$4\pi\int_{r_i}^{r_{i+1}}\left[vD\left(\frac{d^2\phi}{dr^2} + \frac{2}{r}\frac{d\phi}{dr}\right) + v(\nu\sigma_f - \sigma_a)\phi\right] r^2\,dr = 4\pi\int_{r_i}^{r_{i+1}}(\,\alpha\,\phi\,)\,r^2\,dr$$

which results in:

$$\frac{3vD\left(r_{i+1}^2\frac{d\phi}{dr}|_{i+1} - r_i^2\frac{d\phi}{dr}|_i\right)}{(r_{i+1}^3 - r_i^3)} + v(\nu\sigma_f - \sigma_a)\phi_{i+\frac{1}{2}} = \alpha\,\phi_{i+\frac{1}{2}} \tag{46}$$

Substituting Equation 37 into this result produces the following discretized equation for interior cells in spherical coordinates:

$$\begin{aligned}\left(\frac{3vDr_{i+1}^2}{\Delta r(r_{i+1}^3 - r_i^3)}\right)\phi_{i+\frac{3}{2}} \left(-\frac{3vD(r_{i+1}^2 + r_i^2)}{\Delta r(r_{i+1}^3 - r_i^3)} + v(\nu\sigma_f - \sigma_a)\right)\phi_{i+\frac{1}{2}} \\ + \left(\frac{3vDr_i^2}{\Delta r(r_{i+1}^3 - r_i^3)}\right)\phi_{i-\frac{1}{2}} = \alpha\,\phi_{i+\frac{1}{2}}\end{aligned} \tag{47}$$

Similarly, in cylindrical coordinates the integration of Equation 15 across cell (i+$\frac{1}{2}$)

is given by:

$$2\pi \int_{r_i}^{r_{i+1}} \left[vD \left(\frac{d^2\phi}{dr^2} + \frac{1}{r}\frac{d\phi}{dr} \right) + v(\nu\sigma_f - \sigma_a)\phi \right] \, r \, dr = 2\pi \int_{r_i}^{r_{i+1}} \left(\, \alpha \, \phi \, \right) \, r \, dr$$

which results in:

$$\frac{2vD \left(r_{i+1} \frac{d\phi}{dr}|_{i+1} - r_i \frac{d\phi}{dr}|_i \right)}{(r_{i+1}^2 - r_i^2)} + v(\nu\sigma_f - \sigma_a)\phi_{i+\frac{1}{2}} = \alpha \, \phi_{i+\frac{1}{2}} \tag{48}$$

Substituting Equation 37 into this result produces the following discretized equation for interior cells in spherical coordinates:

$$\begin{aligned} \left(\frac{2vDr_{i+1}}{\Delta r(r_{i+1}^2 - r_i^2)} \right) \phi_{i+\frac{3}{2}} + \left(-\frac{2vD(r_{i+1} + r_i)}{\Delta r(r_{i+1}^2 - r_i^2)} + v(\nu\sigma_f - \sigma_a) \right) \phi_{i+\frac{1}{2}} \\ + \left(\frac{2vDr_i}{\Delta r(r_{i+1}^2 - r_i^2)} \right) \phi_{i-\frac{1}{2}} = \alpha \, \phi_{i+\frac{1}{2}} \end{aligned} \tag{49}$$

Applying these integrations to cell (N-$\frac{1}{2}$), with outer boundary condition defined by Equation 18 and the definition of $\frac{d\phi}{dr}|_N$ defined by Equation 42, results in the following finite volume discretizations:

$$\begin{aligned} \left(-\frac{vD(9r_N^2 + 3r_{N-1}^2)}{\Delta r(r_N^3 - r_{N-1}^3)} + v(\nu\sigma_f - \sigma_a) \right) \phi_{N-\frac{1}{2}} + \left(\frac{vD(r_N^2 + 3r_{N-1}^2)}{\Delta r(r_N^3 - r_{N-1}^3)} \right) \phi_{N-\frac{3}{2}} \\ = \alpha \, \phi_{N-\frac{1}{2}} \end{aligned} \tag{50}$$

$$\begin{aligned} \left(-\frac{2vD(3r_N + r_{N-1})}{\Delta r(r_N^2 - r_{N-1}^2)} + v(\nu\sigma_f - \sigma_a) \right) \phi_{N-\frac{1}{2}} + \left(\frac{2vD(\frac{r_N}{3} + r_{N-1})}{\Delta r(r_N^2 - r_{N-1}^2)} \right) \phi_{N-\frac{3}{2}} \\ = \alpha \, \phi_{N-\frac{1}{2}} \end{aligned} \tag{51}$$

Equation 50 represents the spherical discretization and Equation 51 the cylindrical. Similarly, applying these integrations to cell $\frac{1}{2}$, with the inner boundary condition

defined by Equation 20, results in:

$$\left(\frac{3vD}{r_1\Delta r}\right)\phi_{\frac{3}{2}} + \left(-\frac{3vD}{r_1\Delta r} + v(\nu\sigma_f - \sigma_a)\right)\phi_{\frac{1}{2}} = \alpha\,\phi_{\frac{1}{2}} \tag{52}$$

$$\left(\frac{2vD}{r_1\Delta r}\right)\phi_{\frac{3}{2}} + \left(-\frac{2vD}{r_1\Delta r} + v(\nu\sigma_f - \sigma_a)\right)\phi_{\frac{1}{2}} = \alpha\,\phi_{\frac{1}{2}} \tag{53}$$

for the spherical and cylindrical discretizations respectively. Equations 47, 50, and 52 therefore represent the full 1D, spherical finite volume discretization, while Equations 49, 51, and 53 represent the 1D, cylindrical finite volume discretization. As with the corresponding finite difference and Cartesian finite volume discretizations, both of these results are second order accurate. The corresponding steady-state diffusion equation discretizations for each coordinate system are derived in Appendix C.

Both the finite volume method and finite difference method discretizations presented in this work make use of uniform mesh spacing, wherein dr is a constant value throughout the discretization. However, other mesh spacing methods also exist. This effort also made use of the following two non-uniform mesh spacing schemes: a volumetric based spacing, and a Chebyshev polynomial based spacing. The volumetric spacing used here is specific to spherical coordinates, and defines dr values for each cell such that, when rotational symmetry is applied, all cells will have an equal volume. The second spacing scheme, in turn, uses the Chebyshev polynomial to define the position of each cell boundary, with boundary position i given by [21]:

$$r_i = \left(1 - \cos(\frac{i\pi}{N})\right)\frac{R}{2} \tag{54}$$

2.5 The JAKES Algorithm

JAKES makes use of two different solution methods: a 1D finite difference solution to the neutron diffusion equation, and a 2D even-parity transport equation [5]. It is

the neutron diffusion equation solution method that this effort builds on. In this method, JAKES takes an eigenvalue problem and converts it to a nonlinear equation to be solved using Newton's method [5]. This step is necessary due to the nature of the time eigenvalue. While many eigenvalue problems can be readily solved using a power iteration, this method will only return the dominant mode of the eigenvalue [20]. This is not sufficient for the time eigenvalue problem as the mode of interest is the fundamental mode, being defined as the most positive eigenvalue, as this will be the most physically meaningful for an FBR [10]. The dominant eigenvalue mode of the FBRs modeled in this effort, however, are negative values whose absolute values are orders of magnitude larger than the fundamental mode and which are not of interest as physical solutions [22]. It is for this reason that Newton's method is applied, in order to be able to identify the eigenvalue of interest specifically, and not merely the dominant mode. This solution would require the computing of a matrix of Jacobians, however, JAKES instead implements a block Gaussian elimination method in order to approximate the Jacobian matrix without ever explicitly forming it [5]. This results in a large, Jacobian-free linear system of equations. This system is then solved using the Krylov sub-space method of generalized minimum residual (GMRES). This GMRES method was implemented from the Portable, Extensible Toolkit for Scientific Computation (PETSc) developed by Argonne National Labs [6], and represents the most computationally intensive segment of the code [5]. As a whole, this Jacobian-Free Newton-Krylov (JFNK) method forms the core of the JAKES algorithm.

JAKES was originally built using PETSc version 3.7, but this effort is built using the updated version 3.14 release [5]. As noted previously, it is specifically the GMRES solver from PETSc which JAKES makes use of in its JFNK solution for the time eigenvalue. This numerical solver has seen extensive use in multiple fields of research

[23, 24, 25]. This widespread use allows for high confidence in its accuracy as a numerical solver. It was for this reason that the PETSc toolkit was originally chosen for use in JAKES, and it is for the same reason that it continues to be used in this effort [5].

While it is the time eigenvalue result which is of greatest use for modeling FBRs, the JFNK method must be initialized with an initial approximation of the eigenvalue and eigenvector being solved in order for the method to be effective. It is for this reason that the JAKES algorithm first solves for the k eigenvalue and its associated eigenvector. The k eigenvalue can be used to approximate the time eigenvalue, which provides a value that can then be used to initialize the JFNK solution method. This relation is given by the following equation [10]:

$$\alpha \;=\; \frac{k \;-\; 1}{l} \tag{55}$$

where l is an estimate of the average neutron lifetime. And unlike the time eigenvalue, the fundamental mode of the k eigenvalue is also its dominant mode. This allows for a solution using the power iteration method, which for the systems examined in this effort is computationally trivial [20]. This enables the JFNK method to be initialized with relatively little computational cost.

III. Methodology

3.1 Discretization

The first step in this effort was to derive spatial discretizations of the neutron diffusion equation using the finite difference and finite volume methods. These discretizations were accomplished for both the time eigenvalue equation, as shown in Equation 15, and the steady-state diffusion equation, shown in Equation 17. These derivations are shown in Chapter II for the time eigenvalue equation, and in Appendix C for the steady-state diffusion equation.

Following the derivations, these discretizations were then implemented in the model. Before being incorporated into the full JAKES algorithm, these discretizations were first tested for small scale models, written using Python. Once these models had been tested, the discretizations were them implemented in JAKES itself, written in FORTRAN. The final JAKES implementation made use of uniform mesh spacing, but tests cases were also run using volumetric and Chebyshev polynomial spacing as well. Once the discretizations had been completed, the algorithm could then begin the numerical solution process.

3.2 k Eigenvalue Calculation

Using the discretization of the steady-state diffusion equation, the k eigenvalue could be determined. This discretization results in an equation in the form of

$$A\vec{\phi} = \frac{1}{k}\vec{\phi}.$$

If this equation is inverted, then this becomes a straightforward eigenvalue problem. As discussed in Chapter II, the value of the k eigenvalue which is of interest to this

effort is the dominant mode, which can be readily solved for using the power iteration method. This, therefore, is the numerical solution method used in this effort. The power iteration algorithm can be written as [26]:

$$k\vec{\phi} = \vec{\phi}A^{-1}$$

$$\text{Make initial guess for eigenvector } \vec{\phi}_1$$

$$\text{Iterate though } \vec{\phi}_{k+1} = \frac{A^{-1}\vec{\phi}_k}{|| A^{-1}\vec{\phi}_k ||} \text{ to convergence}$$

$$k = \frac{\vec{\phi}_k^* A^{-1}\vec{\phi}_k}{\vec{\phi}_k^*\vec{\phi}_k}$$

This algorithm returns the dominant eigenvalue of k for the steady-state operation of the FBR. This value can then be related to the time eigenvalue by Equation 55:

$$\alpha = \frac{k-1}{l}$$

In order for this relation to be useful, a neutron lifetime must be estimated. For the operation of an FBR, an average value can be assumed to be 10 ns [12]. This allows us to calculate a value to serve as an initial estimate of the time eigenvalue of the system. This numerical solution was compared to the analytic solution derived in Appendix B, allowing for an absolute error determination to be made to determine the accuracy of the numerical solution for the k eigenvalue. This numerical k eigenvalue and eigenvector result was then used to initialize the JFNK solution method.

3.3 Time Eigenvalue Calculation

With the discretizations and k eigenvalue calculation in place, the final and largest step of the JAKES algorithm to solve for the fundamental mode of the time eigenvalue. The discretization of Equation 15, as derived in Sections 2.3 and 2.4, can be paired

with the initial time eigenvalue estimate resulting from the application of Equation 55 to the result of the power iteration solution for the k eigenvalue. This value, paired with the k eigenvector solution, serves to initialize Newton's method. As discussed in Chapter II, this is necessary because the dominant eigenvalue will most likely be negative, and therefore is not of interest as a physical solution. The initial value therefore is used to ensure that the solution algorithm begins close enough to the fundamental mode that the solution converges on the desired eigenvalue.

These inputs, implemented in FORTRAN, then interface with the JFNK solver as described in Section 2.5. The eigenvalue problem shown in Equation 27 that is fundamentally being solved must be rearranged to be set equal to zero in order to apply Newton's Method. From there, the Jacobian matrix which appears in the solution must be approximated using a block Gaussian elimination method. The resultant system of equations can then be solved using the GMRES solver from the PETSc package, as implemented into FORTRAN using PETSc version 3.14, in order to solve for the time eigenvalue and its corresponding eigenvector.

3.4 Verification

Once the algorithm was fully updated with finite volume discretization, an enhanced finite difference discretization, and an updated PETSc numerical solver, it was then tested as a whole for each coordinate system and discretization method and compared to the known analytical solutions for each case. Prior to this test, however, it was optimized for accuracy and efficiency by adjusting the values of N, the total number of nodes or cells in the discretization, and of tolerance, the criteria for determining when the k eigenvalue power iteration solution has converged. The effects of each of these variables on the k and time eigenvalue results were tested individually, enabling optimal values of each to be selected.

With these tests completed, the updated JAKES algorithm was then ready to model the Godiva I and WSMR FBRs. Both of these cases were modeled using the reactor properties listed in Table 2 and the cross-section values list in Table 3. Both the spherical Godiva I and the cylindrical WSMR FBRs were modeled using both finite difference and finite volume discretization. And the numerical solutions for each of these cases were compared to their corresponding analytical solutions, as derived in Appendices A and B, for both the k and time eigenvalues. The flux eigenvector results for each problem were also compared to their corresponding analytic solutions.

The Godiva I model was the more extensive of these two comparisons. As discussed in Section 2.1, the spherical geometry of the reactor enabled it to be realistically modeled using a 1D coordinate system. Additionally, the data collected previously with JAKES, shown in Table 1, was directly comparable to the results from this new model. The WSMR FBR, however, was more limited. The previous results were produced using the 2D JAKES-EVENT algorithm, and so were not directly comparable to the result of the 1D cylindrical model. Further, due to the non-physical neutron non-leakage introduced by modeling this cylindrical system in 1D, the results of this model could only be compared to the analytical solution for the model, a limitation described further in Section 2.1. Despite this, the FBR was still modeled and its accuracy assessed with the intent of informing future work to expand this model into higher dimensions.

IV. Results and Analysis

4.1 Model Optimization

The first data gathered for this effort was collected as a function of tolerance and N. Tolerance refers to the maximum acceptable residual value used by the power iteration method when solving for the k eigenvalue. N is used to designate the size of the discretization of the problem, indicating the number of nodes used for finite difference or the number of cells used for finite volume. These results were gathered first so that the values chosen for subsequent models could be optimized to achieve maximum solution accuracy and efficiency.

This optimization was conducted using spherical coordinates, with the FBR model having the properties of Godiva I, as shown in Table 2. The results as a function of tolerance held N constant, and likewise the results as a function of N held tolerance as a constant. These data were gathered for both finite difference and finite volume discretizations. It should be noted that the run times measured in this test, as well as all CPU times measured in this effort, were the result of running JAKES on a single 2.40 GHz processor.

The tolerance data are shown in Tables 4 and 5. The trends for finite difference and finite volume are similar, with decreasing tolerance directly resulting in a decrease in the residual of the power iteration calculation of the k eigenvalue. This is to be expected, as reaching a residual lower than the tolerance is, by definition, the exit criteria of the power iteration. To achieve this lower residual, a greater number of iterations are required, which is also clearly shown in the tables. More relevant to the problem to be solved, however, is the effect of the accuracy of eigenvalue calculations for the k and time eigenvalues. The errors presented here are the absolute difference between the eigenvalue found by the numerical calculation and the analytic solution.

At tolerances of 10^{-6} and greater, the k error, and correspondingly the time eigenvalue error, are of the same order of magnitude as the k residual. However, at lower tolerances the magnitudes of the errors remain constant as the residual decreases. This suggests that beyond a k error of approximately 4.38 x 10^{-7} for finite difference and 5.36 x 10^{-7} for finite volume, an accuracy limit is reached that is unrelated to the tolerance of the numerical solution method.

Further, it can be seen in Tables 4 and 5 that as the error in the k eigenvalue decreases, so too does the run time of the JFNK solution method. The run time tracks with the power iteration tolerance and residual as well until a tolerance of 10^{-9}, beyond which the previously mentioned accuracy limit begins to show up. This matches with our expectations that, as the k eigenvalue calculation is being used to initialize the JFNK time eigenvalue calculation, the more accurate the k eigenvalue input from the power iteration, the faster the JFNK calculation can converge on a solution.

Based on these results, a tolerance of 10^{-12} was chosen to be used for subsequent models. This value falls clearly within the less than 1 second JFNK run time regime, and produces errors for both the k eigenvalue and the time eigenvalue, α, which match the lower tolerance results to 6 digits of accuracy for both finite difference and finite volume. Given that the original JAKES results were provided with 7 digits of

Table 4. Finite difference results as a function of tolerance for the spherical, Godiva I model with N = 1,000. 'tol' refers to the tolerance set on the k eigenvalue power iteration solution, and 'Iters' refers to the number of iterations required for the power iteration to converge.

tol	k Error	k Residual	Iters	α Error (μs^{-1})	Run Time (s)
1E-03	2.763711E-04	8.099910E-04	9	1.695520E-04	70.03125
1E-06	7.638021E-07	9.624999E-07	16	1.672183E-04	36.1875
1E-09	4.377989E-07	4.310737E-10	24	1.615828E-04	0.6875
1E-12	4.376530E-07	5.058176E-13	31	1.615041E-04	0.671875
1E-15	4.376529E-07	9.992007E-16	37	1.615039E-04	0.671875
1E-16	4.376529E-07	0.000000E+00	40	1.615039E-04	0.671875

Table 5. Finite volume results as a function of tolerance for the spherical, Godiva I model with N = 1,000. 'tol' refers to the tolerance set on the k eigenvalue power iteration solution, and 'Iters' refers to the number of iterations required for the power iteration to converge.

tol	k Error	k Residual	Iters	α Error (μs^{-1})	Run Time (s)
1E-03	2.768460E-04	8.110052E-04	9	1.609321E-04	49.53125
1E-06	8.623037E-07	9.637198E-07	16	1.992804E-04	37.71875
1E-09	5.358541E-07	4.316222E-10	24	1.974836E-04	0.703125
1E-12	5.357080E-07	5.065948E-13	31	1.974046E-04	0.6875
1E-15	5.357078E-07	9.992007E-16	38	1.974046E-04	0.6875
1E-16	5.357078E-07	0.000000E+00	46	1.974047E-04	0.6875

accuracy for the k eigenvalue and 4 digits of accuracy for the time eigenvalue, this level of precision in our error values should be sufficient for comparison [5].

Using this tolerance, and the same Godiva I parameters that were used to find it, a similar set of data were then gathered as a function of N. These results are shown in Tables 6 and 7 for the finite difference and finite volume discretizations respectively. As expected, the residual from the power iteration calculation of the k eigenvalue had little variation with N, and for all cases presented in these tables the k eigenvalue calculation took 31 iterations to converge, having no variation with N. For the power iteration solution method, k eigenvalue errors decreased by two orders of magnitude for every order of magnitude increase in N for both discretization methods. This behavior was expected, as both the finite difference and finite volume discretization schemes used in this effort are second order accurate. Additionally, these k eigenvalue results illuminate the apparent source of the "accuracy limit" noted in the tolerance results from Tables 4 and 5: the clear second order accuracy progression seen in the k calculation errors relative to N explains why, for a constant N of 1,000, the tolerance results quickly reached a limit on the order of magnitude of accuracy that could be reached. Based on these trends, the selected tolerance of 10^{-12} should be sufficient for values of N up to at least 100,000.

This expected second order accuracy trend does not appear in the time eigenvalue

Table 6. Finite difference results as a function of N for the spherical, Godiva I model with tol = 1E-12.

N	k Error	k Residual	α Error (μs^{-1})	JFNK Run Time (s)
30	2.333711E-04	4.984901E-13	8.607803E-02	0.000000
100	3.736814E-05	5.021539E-13	1.378831E-02	0.031250
300	4.678367E-06	5.054845E-13	1.726365E-03	0.078125
1000	4.376530E-07	5.058176E-13	1.615041E-04	0.640625
3000	4.915331E-08	5.043743E-13	1.811989E-05	5.765625
5000	1.791398E-08	5.081491E-13	6.603304E-06	15.953125
8000	7.105645E-09	5.060397E-13	2.611942E-06	39.546875
10000	4.625060E-09	5.091483E-13	1.362697E-06	64.515625
20000	2.725193E-09	5.201395E-13	5.483590E-07	286.171875
25000	3.421875E-09	5.110357E-13	1.869956E-06	402.437500

Table 7. Finite volume results as a function of N for the spherical, Godiva I model with tol = 1E-12.

N	k Error	k Residual	α Error (μs^{-1})	JFNK Run Time (s)
30	6.245356E-04	5.421219E-13	2.301840E-01	0.015625
100	5.430077E-05	5.100365E-13	2.003537E-02	0.031250
300	5.973305E-06	5.064837E-13	2.203987E-03	0.078125
1000	5.357080E-07	5.065948E-13	1.974046E-04	0.671875
3000	5.946224E-08	5.133671E-13	2.155470E-05	6.062500
5000	2.158015E-08	5.053735E-13	7.544293E-06	16.203125
8000	8.522202E-09	5.044853E-13	2.666628E-06	41.875000
10000	5.525281E-09	5.067058E-13	1.426102E-07	67.781250
20000	2.926622E-09	5.300205E-13	2.659536E-06	482.109375
25000	3.204858E-09	5.050405E-13	3.216231E-07	419.562500

results in the same manner as the k eigenvalue ones, however. Both discretization methods produce time eigenvalue errors that track well with the expected accuracy through N = 8,000. Beyond this point however, the two methods diverge. This change is illustrated in Figure 5, which plots the absolute errors in the time eigenvalue for both discretization methods as a function of N. At N = 30, the finite difference result produces a 2.7 times lower absolute error for the time eigenvalue than the finite volume result. As N increases, the relative difference in error between the two methods decreases until at N = 8,000 there is a less than 2% difference between the two discretizations. This evidences a trend in which the increase in accuracy gained from a increase in the value of N is larger for the finite volume method than it is for the finite difference method. At larger values of N, however, the errors begin to diverge, with the finite volume method being 2.85 times more accurate than the finite difference method at N = 10,000. At N = 20,000, however, a reversal occurs and the finite difference method becomes nearly an order of magnitude more accurate, and at N = 25,000 finite volume switched back to being the more accurate method again. At values of N greater than 25,000, the JFNK method failed to converge, failing to provide output, although the power iteration still produced k eigenvalue output. This was likely due to hardware restraints on the computer system used for this effort.

At large values of N before this maximum limit was reached, a discrepancy can be identified in the k eigenvalue results listed in Tables 6 and 7. The k eigenvalue errors for both discretization methods follow the expected second order accuracy for N $\leq$ 10,000. At N = 20,000 the error decreases by significantly less than the factor of 4 predicted for second order accuracy, and at N = 25,000 the error actually increases. This discontinuity in the expected trend indicates a maximum accuracy limit for the power iteration on the hardware used for this effort. As discussed in Section 2.5, any errors introduced in the power iteration solution will propagate into

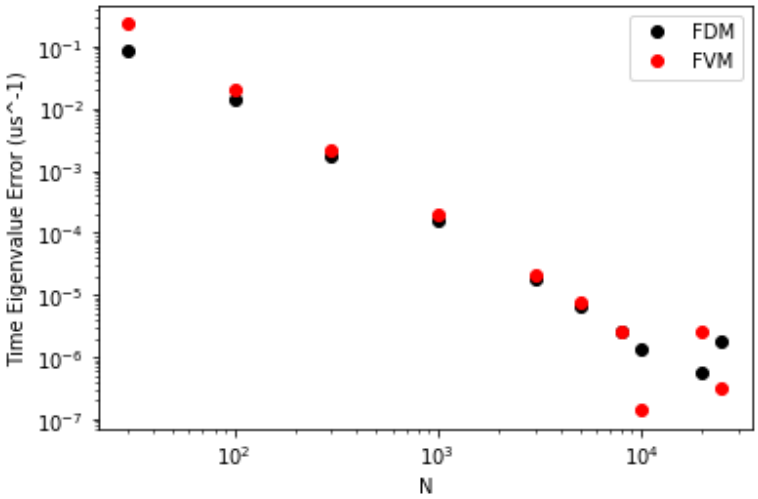

Figure 5. Absolute error in the time eigenvalue as a function of N.

the time eigenvalue solution. However, the first discontinuity in the time eigenvalue results occurs at N = 10,000, at which point the power iteration results still follow the second order accuracy trend. Therefore, this represents a separate error unique to the JFNK solution, albeit compounded by the power iteration error at N $\geq$ 20,000. This instability likely arises during the course of the GMRES solver as it produces a large number of (on the order of N) Krylov vectors. As this method solves the matrix approximated earlier by the JFNK method, a matrix which itself is ill formed due to the lack of any preconditioning, an error likely arises in the Krylov vectors due to hardware restraints which effects the convergence of the solution. Further investigation of these instabilities was left for future research, and a value of N = 1,000 was selected for subsequent models so as to ensure that all results were fully within the regime of second order accuracy for both the k and time eigenvalue solution methods.

While the other results presented in this and subsequent sections made use of uniform mesh spacing, a comparison was also made between this and the volumetric

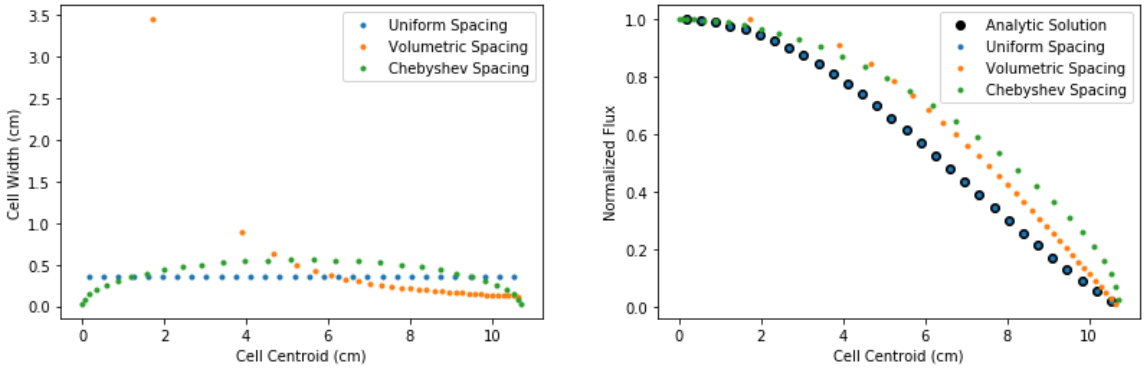

Figure 6. The plot on the left shows the width of each cell as function of its radial position for each mesh spacing scheme for N=30. The plot on the right shows the corresponding flux eigenvector result for each scheme.

and Chebyshev polynomial spacings. These three cases were all implemented and compared using finite volume discretization for a spherical Godiva I case, the results of which are shown in Figure 6. A significant deviation in the flux eigenvector values can be seen for the two non-uniform mesh spacings relative to the analytic and uniform spacing eigenvectors. Additionally, these mesh spacing schemes were used to compute the time eigenvector for a N = 1,000 Godiva I case. The Chebyshev polynomial spacing resulted an eigenvalue absolute error which was larger than the error of the uniform spacing case by a factor of 2.97660 x 10^5. The volumetric spacing had 1.98 times larger error still, differing from the uniform spacing case by a factor of 5.88212 x 10^5. As shown in Figure 6, the effect of these non-uniform mesh spacing schemes is to reduce dr at one or both boundary conditions, for the volumetric and Chebyshev polynomial schemes respectively, at the expense of larger dr values elsewhere in the discretization. From these results, it is clear that any increase in accuracy gained from the lower dr values at the boundary is offset by the decrease in accuracy from the larger dr values elsewhere in the discretization.

4.2 Coordinate System Comparison

Next, a comparison was made between the three coordinate systems implemented in JAKES. The original spherical coordinate system with updated finite difference and finite volume discretizations was compared to the two newly introduced cylindrical and Cartesian systems. All three of these systems were run in 1D, using the properties of the Godiva I FBR shown in Table 2, using the tolerance and N values determined in Section 4.1. The results of this comparison are given in Table 8 for the k eigenvalue power iteration calculation, and in Table 9 for the time eigenvalue JFNK calculation.

The first difference to be noted between these results is differences in the value of the k and time eigenvalues. The time eigenvalue for cylindrical coordinates is two orders of magnitude larger than that for spherical coordinates, and the Cartesian value is larger still. This same pattern of increasing size from spherical as the smallest to Cartesian as the largest can also be seen in the k eigenvalue results. This is the expected result of applying 1D geometry to these coordinate systems. While rotational symmetry can be applied to a 1D spherical system to produce a physical 3D model, its cylindrical and Cartesian counterparts will both be nonphysical. This is because cylindrical coordinates only have one dimension of rotational symmetry, leaving the third to be treated as infinite. The problem is magnified in Cartesian coordinates, which lack any rotational symmetry, resulting in two infinite dimensions. The effect of these infinite dimensions on our FBR models is that they will have no neutron leakage in these dimensions, as a neutron cannot leak through a boundary which does not exist. This raises the criticality of the system in a nonphysical manner, resulting in larger k and time eigenvalues for both coordinate systems, with the Cartesian values being larger due to it having more infinite dimensions.

More interesting are the relative differences between the coordinate systems, and how the accuracies relative to the analytic solutions varied between each combination

Table 8. k eigenvalue results for each coordinate system using the properties of the Godiva I model.

	Spherical		
	Value	Error	Power Iterations
Analytic	1.00027941749		
FDM	1.00027985515	4.37653E-7	31
FVM	1.00027995320	5.35708E-7	31
	Cylindrical		
	Value	Error	Power Iterations
Analytic	1.28875597881		
FDM	1.28875330117	2.67763E-6	29
FVM	1.28875622892	2.50119E-7	29
	Cartesian		
	Value	Error	Power Iterations
Analytic	1.68246885560		
FDM	1.68170317295	7.65683E-4	29
FVM	1.68246893428	7.86779E-8	29

Table 9. Time eigenvalue results for each coordinate system using the properties of the Godiva I model.

	Spherical			
	Value (μs^{-1})	Error (μs^{-1})	FWHM (μs)	Run Time (s)
Analytic	0.10316383654			
FDM	0.10332534059	1.61504E-4	34.105863865579	0.65625
FVM	0.10336124112	1.97405E-4	34.0940178539683	0.6875
	Cylindrical			
	Value (μs^{-1})	Error (μs^{-1})	FWHM (μs)	Run Time (s)
Analytic	70.33893033862			
FDM	70.33849031168	4.40027E-4	0.0501005919288981	0.71875
FVM	70.33897129816	4.09595E-5	0.0501002493349271	0.671875
	Cartesian			
	Value (μs^{-1})	Error (μs^{-1})	FWHM (μs)	Run Time (s)
Analytic	116.02460045641			
FDM	115.96144304745	6.31574E-2	0.0303894114059787	0.65625
FVM	116.02460688199	6.42558E-6	0.030372867400311	0.671875

of coordinate system, discretization method, and eigenvalue solution method. For the spherical coordinate system, the k eigenvalues from finite difference and finite volume differ by 22.4%. Similarly, the spherical time eigenvalue results for both discretization methods differ by 22.2%, with the absolute errors for the time eigenvalues both being 3.7 x 10^2 times larger than their corresponding k eigenvalue errors. The other two coordinate systems share a trend of the time eigenvalue results being approximately two orders of magnitude less accurate than the corresponding k eigenvalue results. This suggests that the JFNK calculation produces a less accurate result than the power iteration does for the Godiva I model. This is in keeping with expectations, because, for the systems of interest to this effort, the k eigenvalue problem is well formed, while the time eigenvalue problem is ill formed. This results in a more efficient, and consequently more accurate, solution for the k over the time eigenvalue when using the same discretization for both. There are, however, other differences in the results seen in Tables 8 and 9 which cannot be attributed solely to differences in the solution method.

The most notable difference in these results is the disparity in eigenvalue solution accuracies, both for k and time, between the three coordinate systems. The previous section demonstrated that the spherical finite difference and finite volume methods, even at their greatest difference, always agreed to within an order of magnitude in error. But for cylindrical coordinates we see the two methods producing errors which differ by a factor of 10.7 for both the k eigenvalue and the time eigenvalue. The Cartesian result is the most extreme, however, with a 9.73 x 10^3 factor of difference for the k eigenvalue and 9.83 x 10^3 for the time eigenvalue. The slightly larger difference between the JFNK results than the power iteration results suggests that the source of the difference is being amplified by the JFNK method. But this effect from the numerical methods is clearly much smaller in scale than the source of the

difference between the two discretization methods.

This variable difference in the accuracies of the two discretization methods between coordinate systems was not expected, but can be explained by examination of the discretized matrix elements (note that the JFNK method does not fully assemble the matrix, but it does compute each of the matrix elements in the course of the calculation, so for this purpose the effect is the same). Particularly curious was that this difference was most extreme in Cartesian coordinates, for which the finite difference and finite volume discretizations have been found to be identical for interior nodes and cells, as shown in Section 2.4. Therefore the difference in discretization must arise from the boundary conditions for Cartesian coordinates, and likely arises from the same source for cylindrical coordinates. The finite difference inner, reflective boundary is described by Equation 32:

$$\left(\frac{2vD}{\Delta r^2}\right)\phi_1 + \left(-\frac{2vD}{\Delta r^2} + v(\nu\sigma_f - \sigma_a)\right)\phi_0 = \alpha\phi_0$$

and the corresponding Cartesian finite volume boundary by Equation 45:

$$\left(\frac{vD}{\Delta r^2}\right)\phi_{3/2} + \left(-\frac{vD}{\Delta r^2} + v(\nu\sigma_f - \sigma_a)\right)\phi_{1/2} = \alpha\phi_{1/2}.$$

At the outer, vacuum boundary, the finite difference result is given by Equation 28:

$$\left(-\frac{2vD}{\Delta r^2} + v(\nu\sigma_f - \sigma_a)\right)\phi_{N-1} + \left(\frac{vD}{\Delta r^2} - \frac{avD}{2r_{N-1}\Delta r}\right)\phi_{N-2} = \alpha\phi_{N-1}$$

and the corresponding Cartesian finite volume boundary by Equation 43:

$$\left(-\frac{4vD}{\Delta r^2} + v(\nu\sigma_f - \sigma_a)\right)\phi_{N-1/2} + \left(\frac{4vD}{3\Delta r^2}\right)\phi_{N-3/2} = \alpha\phi_{N-1/2}.$$

These differences are quantified for an example case in Table 10. This data demon-

Table 10. The value of each finite volume matrix term divided by its corresponding finite difference matrix term at each boundary for a Godiva I model with N = 300.

	Cartesian	Cylindrical	Spherical
A(1,1)	0.49993063	0.49998628	0.49999085
A(1,2)	0.50000000	0.50000000	0.50000000
A(N,N)	2.00013874	2.03231470	2.06593921
A(N,N-1)	1.33333333	1.31401799	1.29558074

strates that at the inner boundary, the ratios between the finite difference and finite volume matrix terms agree between each coordinate system to 4 digits of precision. This indicates that the outer boundary, where there is less agreement between coordinate systems, is the driving cause of the differences in solution accuracies seen in Tables 8 and 9. Further examination of these differences in the handling of the boundary conditions can be seen by comparing the eigenvectors produced by each discretization scheme. The boundary values of the flux eigenvectors resulting from each method are compared to the analytic solution in Table 11. The result of this comparison in cylindrical coordinates is that the finite difference method is less accurate, but with errors of the same order of magnitude, at both boundaries. In Cartesian coordinates, the effect is more extreme, with finite difference being more accurate by a factor of 125 at the inner boundary but less by a factor of 2.0 at the outer boundary. In spherical coordinates, where the finite difference eigenvalue was more accurate, the finite volume eigenvector is more accurate at both boundaries, although with errors still on the same order of magnitude as the finite difference results, just like the cylindrical results. As can be seen in Tables 8 and 9, the result of this is that the spherical finite difference and finite volume discretizations produce eigenvalue results with errors of similar order, while for cylindrical coordinates finite volume becomes an order of magnitude more accurate, and for Cartesian coordinates finite volume becomes nearly 4 orders of magnitude more accurate than the finite difference method.

Table 11. The absolute error in the values of the flux eigenvector at the points closest to the inner and outer boundaries: nodes 1 and N-1 for the finite difference method and cells 1/2 and N-1/2 for the finite volume method for a Godiva I model with N = 1,000.

	Spherical		Cylindrical		Cartesian	
	Inner	Outer	Inner	Outer	Inner	Outer
FDM	3.29535E-9	9.94769E-7	2.88818E-9	1.21432E-6	2.47111E-9	1.57237E-6
FVM	1.64273E-9	5.00812E-7	1.44427E-9	6.24894E-7	3.08425E-7	7.86184E-7

4.3 Verification Cases

The final results of this effort are finite difference and finite volume discretized models of the Godvia I and WSMR FBRs. These models were compared against analytical solutions and existing data to serve as verification for the JAKES algorithm. Both models used the optimized values determined in Section 4.1, with N = 1,000 and a tolerance 10^{-12}. The properties listed in Table 2 were used to populate the models for the two FBRs.

The Godiva I case was used as a benchmark in the original JAKES development effort, and so had previous data to compare against [5]. This original JAKES data was gathered using the same FBR properties, and also used an N of 1,000, so any differences in the new results should be due to changes in the solution method, as all external variables were the same. The results for this FBR are shown in Table 12. For the calculation of the k eigenvalue, the finite difference discretization proved to be 22.4% more accurate than the finite volume method. These updated methods could not be differentiated from the original JAKES results, however, as when concatenated to the same number of significant digits as the original result, all three results yield the same absolute error. Therefore, the only conclusion we can make about the effect of the new discretization methods relative to the original on the k calculation is that they result in errors of the same order of magnitude. The JFNK solutions yielded a more meaningful comparison. Even with concatenation, the errors from

the new discretizations, which both round to 2 x 10^{-4}, still represent an increase in accuracy over the original method. Specifically, the finite difference result represents a minimum reduction in error of 35.4%, and the finite volume result a reduction of 21.0%. Between the updated finite difference and the finite volume discretizations, the finite difference method produced a 22.2% lower absolute error for the time eigenvalue. Additionally, the new methods produced a marginal increase in the FWHM, of approximately 0.07 μs and 0.06 μs for finite difference and finite volume respectively, relative to the original JAKES case. In both the original and new cases, the FWHM falls approximately 1 μs below the range of burst widths experimentally measured for the FBR. While the new methods produced marginally larger values, indicating that the discretization method does have an effect on the value, the primary source of the difference is likely due to the models used by JAKES being inherently for a fully assembled FBR system. In reality, however, the neutron burst can occur before the FBR is fully assembled, resulting in a larger burst width [1]. These variations in assembly states are the reason for the experimental burst width having an associated range rather than a single value. Therefore, the underestimate provided by JAKES may be reflective of the model using an ideal state that was not achieved in real world experiments. Finally, both new discretization methods resulted in increased CPU time for the JFNK solution method, however, it should be noted that while all cases presented in Table 12 were run using N = 1,000, the original case was not run on the same computer as the other cases, making a direct comparison between them difficult. The updated finite difference was, however, 2.33% faster than the finite volume discretization. In addition to this data regarding the k and time eigenvalues, the flux eigenvector was also computed for the new finite difference and finite volume calculations, and was compared against the analytic solution for the eigenvector. These findings are shown in Figure 7. Taking the absolute error between the analytical and

finite difference eigenvectors at each point in the discretization produced a vector of absolute error values. The closer each element in this error vector is to 0, the more accurate the eigenvector solution was. Similarly, the closer the norms of the error vector are to 0, the more accurate the solution. Therefore the norms of these error vectors were compared for both the finite difference and finite volume discretizations, with the finite difference method producing the following norms:

$$||\Delta\phi_{\mathrm{FDM}}||_2 = 0.01448230173$$

$$||\Delta\phi_{\mathrm{FDM}}||_\infty = 0.00066584755.$$

While the absolute error between the finite volume eigenvector and the analytical solution resulted in the following norms:

$$||\Delta\phi_{\mathrm{FVM}}||_2 = 8.50289349977\text{E-}6$$

$$||\Delta\phi_{\mathrm{FVM}}||_\infty = 5.01703511419\text{E-}7.$$

This represents a factor of 1.4 x 10^3 improvement in the two-norm and a factor of 9.5 x 10^2 improvement in the infinity-norm from the finite volume discretization over the finite difference result. These mixed results show advantages and disadvantages to the original JAKES algorithm and the two new discretization methods, but do show a clear improvement in the accuracy of the time eigenvalue calculation for the new methods over the original.

The second verification case examined in this effort was the WSMR FBR. The usefulness of this data is limited, however, as this 1D model of a cylindrical reactor is inherently nonphysical, with all of the associated drawbacks discussed in Section 4.2. This model does, however, serve as a stepping stone towards a physically accurate

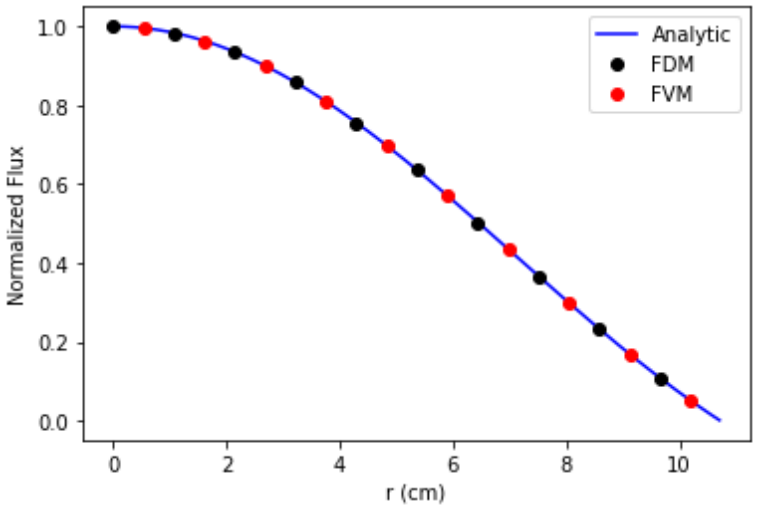

Figure 7. Normalized flux results across the extrapolated boundary for the Godiva I FBR.

representation. Therefore a comparison to the analytic solution does have value in verifying the technique's accuracy, if not its applicability to a real FBR. In order to account for the neutron non-leakage problem caused by the use of 1D geometry, the radius of the model was arbitrarily reduced until it produced a burst width value within the experimentally determined range listed in Table 13. All other properties of the reactor reflect the values of the actual WSMR FBR listed in Table 2, with the JAKES properties N and tolerance set to the values determined in Section 4.1. The results of these calculations are shown in Table 13. For the calculation of the k eigenvalue, the finite volume discretization proved to be more accurate than the finite difference method by a factor of 10.7. The difference in the JFNK solutions was slightly more pronounced, with the finite volume method outperforming finite difference in absolute error by a factor of 10.8. This is in keeping with the results for cylindrical coordinates discussed in Section 4.2. Both discretizations produced FWHM values which fell within the range of the experimental results, but in contrast with the Godiva I results here it was the finite difference method which produced a

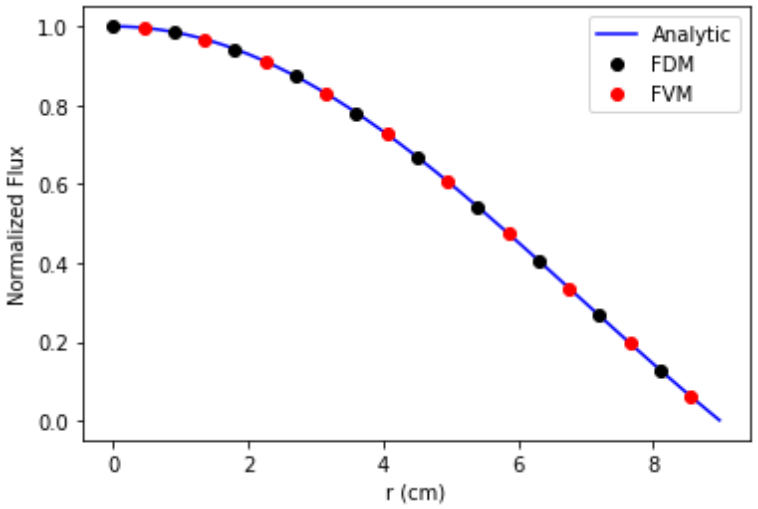

Figure 8. Normalized flux results across the extrapolated boundary for the WSMR FBR.

slightly larger value. The JFNK run times, however, followed the trend set by the Godiva I case, with the finite volume discretization resulting in a 2.38% longer CPU time. In addition to the k and time eigenvalue calculations, the flux eigenvector was also computed for both of the discretization methods, and was compared against the analytic solution for the eigenvector. These findings are shown in Figure 8. The absolute error between the analytical and finite difference eigenvectors resulted in the following norms:

$$||\Delta\phi_{\text{FDM}}||_2 = 0.01321774673$$

$$||\Delta\phi_{\text{FDM}}||_\infty = 0.00059805152.$$

While the absolute error between the finite volume eigenvector and the analytical solution resulted in:

$$||\Delta\phi_{\text{FVM}}||_2 = 9.17918801788\text{E-}6$$

$$||\Delta\phi_{\text{FVM}}||_\infty = 6.24894396276\text{E-}7.$$

This represents a 1.4 x 10^3 improvement in the two-norm and a 9.6 x 10^2 improvement in the infinity-norm from the finite volume discretization over the finite difference result. Though the impact of these results is limited by the use of 1D geometry, they do show a clear improvement in the solution to both the k and time eigenvalue problems for the finite volume discretization over the finite difference discretization.

Table 12. k and time eigenvalue results from the new JAKES discretizations coupled with the original JAKES results and experimental burst width data for the Godiva I FBR.

Case	k	k Error	Iterations	α (μs^{-1})	α Error (μs^{-1})	FWHM (μs)	JFNK Run Time (s)
FVM	1.000279953	5.35708E-7	31	0.103361241	1.97405E-4	34.094017854	0.6875
FDM	1.000279855	4.37653E-7	31	0.103325341	1.61504E-4	34.105863866	0.671875
Analytic	1.000279417			0.103163837			
Original JAKES	1.000280	1E-6		0.1035	3E-4	34.034	0.131
Original Analytic	1.000279			0.1032			
Experimental						35 - 50	

Table 13. k and time eigenvalue results from the new JAKES discretizations coupled with the original JAKES results and experimental burst width data for the WSMR FBR.

Case	k	k Error	Iterations	α (μs^{-1})	α Error (μs^{-1})	FWHM (μs)	JFNK Run Time (s)
FVM	1.000255520	2.57187E-7	25	0.086235117	8.62267E-5	40.865022460	0.671875
FDM	1.000252510	2.75332E-6	25	0.085220126	9.28764E-4	41.351734118	0.65625
Analytic	1.000255263			0.086148890			
Experimental						31 - 50	

V. Conclusions

This effort expanded on the existing JAKES algorithm by implementing a finite volume discretization, as well as an updated finite difference discretization. Both of these methods were implemented in the original spherical coordinates, as well as in cylindrical and Cartesian coordinates. In implementing these changes, JAKES was updated to be compatible with PETSc version 3.14. This updated JAKES algorithm was then used to run a number of FBR models.

The first of these models was intended to optimize the tolerance parameter to maximize the solution accuracy and efficiency. These results demonstrated the dependence of the JFNK time eigenvalue solution accuracy on the accuracy of the power iteration calculation of the k eigenvalue and eigenvector. This matched with expectations, as the power iteration result is used to initialize the JFNK solution method. During these results, a limit in solution accuracy was identified. The next set of data collected illuminated the source of that limit. This data set was gathered to optimize the N parameter, which is the number of nodes used for the finite difference discretization or the number of cells used for finite volume. These results demonstrated that the k eigenvalue calculation followed the trend expected for a second order accurate method for both finite difference and finite volume, with the finite difference method being more accurate at low values of N and the relative difference in the accuracies of the two discretization methods decreasing as N increased. The time eigenvalue solution, however, demonstrated instabilities in accuracy at large values of N. At values of N up to 8,000, the two discretization methods produced time eigenvalue results in agreement in the same manner as the k eigenvalue results. For $N \geq 10{,}000$, however, the accuracies of the discretization methods began to fluctuate. These fluctuations are likely the result of computational hardware restraints, a factor which arises as the ill-formed matrix for the time eigenvalue problem causes instabili-

ties in the JFNK method at large values of N. Following this, a comparison was made between three different mesh spacing schemes: uniform, volumetric, and Chebyshev polynomial spacings. The result of this test was that uniform mesh spacing produced time eigenvalue errors which were orders of magnitude lower than those of the other schemes.

The next data that were collected demonstrated the differences between the original spherical coordinate system and the two new coordinate systems which this effort added to JAKES. These results demonstrated the non-physical nature of using a 1D model for cylindrical and Cartesian coordinates, confirming the expected behavior that cylindrical models were more critical than spherical models of the same parameters, with the effect being amplified for Cartesian models. Beyond this basic result, the data also revealed differences between the finite difference and finite volume discretizations within each coordinate system. While the spherical coordinate system had already been demonstrated to have agreement to within an order of magnitude between the two discretization methods for N = 1,000, the time eigenvalue result for finite volume method was found to be more accurate by a factor of 11 for cylindrical coordinates and more accurate by a factor of 9.8×10^3 for Cartesian coordinates. The reason for this disparity was explored and identified as being a result of differences in the handling of the boundary conditions between the two discretization methods. This difference in accuracies could prove useful in future work to expand this method into 2D and 3D geometries. A Cartesian model in particular, given the significant difference in accuracy, would benefit from use of finite volume discretization rather than using the finite difference method.

The final set of data gathered was for two FBR models for verification purposes. A model of the Godiva I FBR was run which compared this new version of JAKES to the original, as well as compared the eigenvalue and eigenvector solutions to the

analytic solution and compared the burst widths of the models to those measured experimentally. Due to limitations in the existing data, the only conclusion that could be made about the power iteration solution for the k eigenvalue is that the new JAKES methods produce errors on the same order of magnitude as the original method. The JFNK solution for the time eigenvalue and its associated flux eigenvector, however, yielded more meaningful results. The new results represent a minimum reduction in error of 35.1% for the time eigenvalue from the original to the new finite difference result, with the finite volume method producing a 22.2% reduction in error from the original method. The flux eigenvector solutions, however, revealed a larger difference between the two discretization methods, with the absolute error of the finite volume discretization producing a 1.4×10^3 lower two-norm and a 9.5×10^2 lower infinity-norm than the finite difference discretization. These findings indicate that the new JAKES algorithm yields no improvement over the original algorithm for the k eigenvalue solution, but does produce a more accurate time eigenvalue solution for this FBR model. Within the new algorithm, the finite difference and finite volume results only marginally differ in their eigenvalue solution accuracies for the Godiva I case, but the finite volume method does produce a non-negligible increase in accuracy in the eigenvector solution.

The second of these verification models was for the WSMR FBR. The results of this model are less impactful than the Godiva I case, as the 1D model of a cylindrical geometry is inherently non-physical. A comparison was still made to the analytic solution, however, enabling certain conclusions to be drawn. As expected from the earlier comparison of the different coordinate systems, the finite volume discretization produced k and time eigenvalues which were an order of magnitude more accurate than the corresponding finite difference based solutions. Additionally, this model produced eigenvector accuracies similar to those of the Godiva I case. The two-norm

was 1.4 x 10^3 lower and the infinity-norm 9.6 x 10^2 lower for the absolute error of the finite volume result than for that of the finite difference discretization. While these results do not reflect a true, physical FBR, they do indicate a clear advantage in accuracy when using the finite volume discretization for this type of geometry. Future work with this method should therefore focus on finite volume over finite difference discretization for models using cylindrical geometry.

5.1 Future Research

This effort implemented an organic discretization method in JAKES and expanded its applicability to additional geometries, but more work is necessary to improve the validity of the algorithm. JAKES would benefit greatly from expansion into 2D and 3D, especially for the cylindrical and Cartesian coordinate systems. For these two geometries in particular, efforts should be focused on implementing the finite volume over finite difference discretization, given the order of magnitude difference in accuracies seen between the two methods in 1D for these geometries.

Additionally, future researchers would be well served to identify and address the source of the instabilities in time eigenvalue accuracies identified in the JFNK method results at large values of N. Indeed, JAKES would likely benefit from the implementation of an organic numerical solution method that does not rely on PETSc. This could serve to help identify and eliminate both the instability in accuracy and the limitation on N size. Removing such restrictions could enable significantly higher accuracies to be reached, as the instabilities at N $\geq$ 10,000 are the limiting factor in the accuracy of the time eigenvalue solution. Such large N models would require more computational resources, either in terms of run time or processing power. But this need could be at least partially reduced by applying preconditioning to the discretized matrix. In its current state, the matrix created - or effectively created - by JAKES is

ill formed. Applying preconditioning to this matrix could increase the computational efficiency of the method.

Finally, the new discretization methods implemented in JAKES lend themselves as easily to non-homogeneous models as they do to the homogeneous FBRs modeled in this effort. Application of non-homogeneous FBRs, particularly the inclusion of reflective materials, would enhance the applicability of JAKES to modeling real world problems. This addition should be the simplest of the suggestions made here, as it would primarily consist of vectorizing the cell properties, and then modifying the input method to match. The discretization and solution methods themselves, however, would not require any changes to enable this new geometry. These changes would serve to further improve the accuracy and applicability of the JAKES algorithm to modeling new and existing FBRs.

Appendix A. Time Dependent Neutron Diffusion Equation Analytical Solutions

The time eigenvalue form of the neutron diffusion equation, as given in Equation 15:

$$\frac{\alpha}{v}\phi(r) - D\left(\frac{\partial^2\phi}{\partial r^2} + \frac{a}{r}\frac{\partial\phi}{\partial r}\right) + \sigma_a\phi(r) = \nu\sigma_f\phi(r)$$

and, coupled with a reflective inner boundary condition:

$$\frac{d\phi}{dr}|_0 = 0$$

and a vacuum outer boundary condition:

$$\phi(R) = 0$$

can be solved analytically for α and ϕ. The equation can be rearranged in the form of:

$$\frac{d^2\phi}{dr^2} + \frac{a}{r}\frac{d\phi}{dr} + B^2\phi = 0$$

where B^2 is given by:

$$B^2 = \frac{\nu\sigma_f - \sigma_a - \frac{\alpha}{v}}{D}.$$

A general solution to this equation can then be found, and then the system solved using the boundary conditions. However, the form of this general solution varies with the value of a, resulting in a unique solution for each coordinate system.

Spherical Solution

The general solution for spherical coordinates, where $a = 2$, is:

$$\phi = \frac{c_1 e^{-iBr}}{r} - \frac{ic_2 e^{iBr}}{2Br}.$$

Applying the reflective inner boundary gives:

$$\frac{d\phi}{dr} = -\frac{c_1}{r^2}e^{-iBr} + \frac{-iBc_1}{r}e^{-iBr} + \frac{ic_2}{2Br^2}e^{iBr} + \frac{c_2}{2r}e^{iBr}$$

$$0 = \frac{d\phi}{dr}|_0 = -c_1 e^0 + 0 + \frac{ic_2}{2B}e^0 + 0$$

resulting in:

$$c_1 = \frac{ic_2}{2B}.$$

Substituting this value of c_1 into the general equation for ϕ gives:

$$\phi = \frac{1}{r}(\frac{ic_2}{2B})e^{-iBr} - \frac{ic_2}{2Br}e^{iBr}$$

$$\phi = \frac{ic_2}{2Br}(e^{-iBr} - e^{iBr})$$

$$\phi = \frac{ic_2}{2Br}[(\cos(Br) - i\sin(Br)) - (\cos(Br) + i\sin(Br))]$$

$$\phi = \frac{ic_2}{2Br}[0 - 2i\sin(Br)]$$

$$\phi = \frac{c_2}{Br}\sin(Br)$$

This results in a normalized solution for the eigenvector ϕ of:

$$\phi_{norm} = \frac{\sin(Br)}{r}$$

Applying the vacuum outer boundary to this normalized solution gives:

$$0 = \phi(R) = \frac{\sin(BR)}{R}$$

$$BR = \sin^{-1}(0) = n\pi$$

$$B_n = \frac{n\pi}{R}$$

This results in a general, normalized eigenvector solution of:

$$\phi_{norm} = \frac{\sin(B_n r)}{r}$$

and a solution for the time eigenvalue of:

$$\left(\frac{n\pi}{R}\right)^2 = B_n^2 = \frac{\nu\sigma_f - \sigma_a - \frac{\alpha}{v}}{D}$$

$$\alpha = v\left(\nu\sigma_f - \sigma_a - D\left(\frac{n\pi}{R}\right)^2\right)$$

The dominant eigenvalue and eigenvector pair of this solution, which occur at $n = 1$, are given by:

$$\phi_1 = \frac{\sin(\frac{\pi}{R}r)}{r}$$

$$\alpha_1 = v\left(\nu\sigma_f - \sigma_a - \frac{D\pi^2}{R^2}\right)$$

Cylindrical Solution

The general solution for cylindrical coordinates, where $a = 1$, is:

$$\phi = c_1 J_0(Br) + c_2 Y_0(Br)$$

where J_0 and Y_0 are Bessel functions of the first and second kind, respectively. Applying the reflective inner boundary gives:

$$\frac{d\phi}{dr} = -Bc_1 J_1(Br) - Bc_2 Y_1(Br)$$

$$0 = -Bc_1 J_1(B(0)) - Bc_2 Y_1(B(0))$$

resulting in:

$$c_2 = 0.$$

Substituting this value of c_2 into the general equation for ϕ gives:

$$\phi = c_1 J_0(Br) + (0) Y_0(Br)$$

$$\phi = c_1 J_0(Br)$$

Applying the vacuum outer boundary to this equation gives:

$$0 = \phi(R) = c_1 J_0(BR)$$

$$0 = J_0(BR)$$

$$BR = \pm j_{0,n} = \pm 2.4048..., \pm 5.5201..., \pm 8.6537..., \pm 11.7915...$$

Where $j_{0,n}$ is the n^{th} zero of J_0.

$$B_n = \pm \frac{j_{0,n}}{R}$$

This results in a general, normalized eigenvector solution of:

$$\phi_{norm} = J_0 \left(\pm \frac{j_{0,n}}{R} r \right)$$

and a solution for the time eigenvalue of:

$$\left(\pm\frac{j_{0,n}}{R}\right)^2 = B_n^2 = \frac{\nu\sigma_f - \sigma_a - \frac{\alpha}{v}}{D}$$

$$\alpha = v\left(\nu\sigma_f - \sigma_a - D\left(\pm\frac{j_{0,n}}{R}\right)^2\right)$$

The dominant eigenvalue and eigenvector pair of this solution, which occur at $n = 1$, are given by:

$$\phi_1 = J_0\left(\frac{2.4048...}{R}r\right)$$

$$\alpha_1 = v\left(\nu\sigma_f - \sigma_a - \frac{D(2.4048...)^2}{R^2}\right)$$

Cartesian Solution

The general solution for Cartesian coordinates, where $a = 0$, is:

$$\phi = c_1 cos(Br) + c_2 sin(Br).$$

Applying the reflective inner boundary gives:

$$\frac{d\phi}{dr} = -Bc_1 sin(Br) + Bc_2 cos(Br)$$

$$0 = -Bc_1 sin(B(0)) + Bc_2 cos(B(0))$$

resulting in:

$$c_2 = 0.$$

Substituting this value of c_2 into the general equation for ϕ gives:

$$\phi = c_1 cos(Br) + (0) sin(Br)$$

$$\phi = c_1 cos(Br)$$

Applying the vacuum outer boundary to this equation gives:

$$0 = \phi(R) = c_1 cos(BR)$$

$$BR = \cos^{-1}(0) = \frac{n\pi}{2}$$

$$B_n = \frac{n\pi}{2R}$$

This results in a general, normalized eigenvector solution of:

$$\phi_{norm} = cos(\frac{n\pi}{2R}r)$$

and a solution for the time eigenvalue of:

$$\left(\frac{n\pi}{2R}\right)^2 = B_n^2 = \frac{\nu\sigma_f - \sigma_a - \frac{\alpha}{v}}{D}$$

$$\alpha = v\left(\nu\sigma_f - \sigma_a - D\left(\frac{n\pi}{2R}\right)^2\right)$$

The dominant eigenvalue and eigenvector pair of this solution, which occur at $n = 1$, are given by:

$$\phi_1 = cos(\frac{\pi}{2R}r)$$

$$\alpha_1 = v\left(\nu\sigma_f - \sigma_a - \frac{D\pi^2}{4R^2}\right)$$

Appendix B. Steady-State Neutron Diffusion Equation Analytical Solutions

The steady-state neutron diffusion equation, as given in Equation 17:

$$-D\nabla^2\phi + \sigma_a\phi = \frac{1}{k}\nu\sigma_f\phi$$

and, coupled with a reflective inner boundary condition:

$$\frac{d\phi}{dr}|_0 = 0$$

and a vacuum outer boundary condition:

$$\phi(R) = 0$$

can be solved analytically for k and ϕ. The equation can be rearranged in the form of:

$$\frac{d^2\phi}{dr^2} + \frac{a}{r}\frac{d\phi}{dr} + B^2\phi = 0$$

where B^2 is given by:

$$B^2 = \frac{\frac{1}{k}\nu\sigma_f - \sigma_a}{D}.$$

With the exception of the definition of B^2, this problem is identical to that of the time eigenvalue solution derived in Appendix A. The general solutions and resultant general flux equations are, therefore, identical for these two problems. The duplicated steps are therefore skipped in this derivation, and each coordinate systems' unique k eigenvalue is then derived.

Spherical Coordinates

In spherical coordinates, the normalized flux result is:

$$\phi_{norm} = \frac{\sin(Br)}{r}$$

Applying the vacuum outer boundary to this normalized solution gives:

$$0 = \phi(R) = \frac{\sin(BR)}{R}$$

$$BR = \sin^{-1}(0) = n\pi$$

$$B_n = \frac{n\pi}{R}$$

This results in a general, normalized eigenvector solution of:

$$\phi_{norm} = \frac{\sin(\frac{n\pi}{R}r)}{r}$$

and a solution for the k eigenvalue of:

$$(\frac{n\pi}{R})^2 = B_n^2 = \frac{\frac{1}{k}\nu\sigma_f - \sigma_a}{D}$$

$$k = \frac{\nu\sigma_f}{(\frac{n\pi}{R})^2 D + \sigma_a}$$

The dominant eigenvalue and eigenvector pair of this solution, which occur at $n = 1$, are given by:

$$\phi_1 = \frac{\sin(\frac{\pi}{R}r)}{r}$$

$$k_1 = \frac{\nu\sigma_f}{\frac{\pi^2 D}{R^2} + \sigma_a}$$

Cylindrical Coordinates

In cylindrical coordinates, the normalized flux result is:

$$\phi_{norm} = J_0 (Br)$$

Applying the vacuum outer boundary to this normalized solution gives:

$$0 = \phi(R) = J_0 (BR)$$

$$BR = \pm j_{0,n} = \pm 2.4048..., \pm 5.5201..., \pm 8.6537..., \pm 11.7915...$$

$$B_n = \pm \frac{j_{0,n}}{R}$$

This results in a general, normalized eigenvector solution of:

$$\phi_{norm} = J_0 \left(r \frac{j_{0,n}}{R} \right)$$

and a solution for the k eigenvalue of:

$$\left(\frac{j_{0,n}}{R} \right)^2 = B_n^2 = \frac{\frac{1}{k} \nu \sigma_f - \sigma_a}{D}$$

$$k = \frac{\nu \sigma_f}{\left(\frac{j_{0,n}}{R} \right)^2 D + \sigma_a}$$

The dominant eigenvalue and eigenvector pair of this solution, which occur at $n = 1$, are given by:

$$\phi_1 = J_0 \left(\frac{2.4048...}{R} r \right)$$

$$k_1 = \frac{\nu \sigma_f}{\frac{D(2.4048...)^2}{R^2} + \sigma_a}$$

Cartesian Coordinates

In Cartesian coordinates, the normalized flux result is:

$$\phi_{norm} = \cos(Br)$$

Applying the vacuum outer boundary to this normalized solution gives:

$$0 = \phi(R) = \cos(BR)$$

$$BR = \cos^{-1}(0) = \frac{n\pi}{2}$$

$$B_n = \frac{n\pi}{2R}$$

This results in a general, normalized eigenvector solution of:

$$\phi_{norm} = \cos(\frac{n\pi}{2R}r)$$

and a solution for the k eigenvalue of:

$$\left(\frac{n\pi}{2R}\right)^2 = B_n^2 = \frac{\frac{1}{k}\nu\sigma_f - \sigma_a}{D}$$

$$k = \frac{\nu\sigma_f}{\left(\frac{n\pi}{2R}\right)^2 D + \sigma_a}$$

The dominant eigenvalue and eigenvector pair of this solution, which occur at $n = 1$, are given by:

$$\phi_1 = \cos(\frac{\pi}{2R}r)$$

$$k_1 = \frac{\nu\sigma_f}{\frac{\pi^2 D}{4R^2} + \sigma_a}$$

Appendix C. Steady-State Neutron Diffusion Equation Discretizations

The steady-state neutron diffusion equation, as given in Equation 17:

$$-D\left(\frac{d^2\phi}{dr^2}+\frac{a}{r}\frac{d\phi}{dr}\right)+\sigma_a\phi=\frac{1}{k}\nu\sigma_f\phi$$

is discretized below using both the finite difference and finite volume methods. The indexing follows the scheme shown in Figure 4.

Finite Difference

Using finite difference indexing, the generic equation is:

$$-D\left(\frac{d^2\phi}{dr^2}|_i+\frac{a}{r_i}\frac{d\phi}{dr}|_i\right)+\sigma_a\phi_i=\frac{1}{k}\nu\sigma_f\phi_i$$

The following definition, derived in section 3.2 as Equations 22 and 23, can then be substituted into the integrated equation

$$\frac{d\phi}{dr}|_i=\frac{\phi_{i+1}-\phi_{i-1}}{2\Delta r}+O(\Delta r^2).$$

$$\frac{d^2\phi}{dr^2}|_i=\frac{\phi_{i+1}-2\phi_i+\phi_{i-1}}{\Delta r^2}+O(\Delta r^2)$$

which results in the following:

$$-D\left[\left(\frac{\phi_{i+1}-2\phi_i+\phi_{i-1}}{\Delta r^2}\right)+\frac{a}{r_i}\left(\frac{\phi_{i+1}-\phi_{i-1}}{2\Delta r}\right)\right]+\sigma_a\phi_i\Delta r=\frac{1}{k}\nu\sigma_f\phi_i\Delta r$$

Resulting in:

$$\left(\frac{-D}{\nu\sigma_f\Delta r^2} - \frac{aD}{2\nu\sigma_f r_i\Delta r}\right)\phi_{i+1} + \left(\frac{2D}{\nu\sigma_f\Delta r^2} + \frac{\sigma_a}{\nu\sigma_f}\right)\phi_i + \left(\frac{-D}{\nu\sigma_f\Delta r^2} + \frac{aD}{2\nu\sigma_f r_i\Delta r}\right)\phi_{i-1} = \frac{1}{k}\phi_i$$

This equation satisfies the interior cells, but it still must be coupled with a discretization of each of the boundaries. The first of these is the outer vacuum boundary at $r = R$, indexed as $i = N$, is given by:

$$\phi_N = 0$$

Here we make use of Equations 29 and 30:

$$\frac{d\phi}{dr}|_{N-1} = \frac{(0) - \phi_{N-2}}{2\Delta r} + O(\Delta r^2)$$

$$\frac{d^2\phi}{dr^2}|_{N-1} = \frac{(0) - 2\phi_{N-1} + \phi_{N-2}}{\Delta r^2} + O(\Delta r^2)$$

These values, along with the vacuum boundary condition itself, can then be substituted into the governing equation for node (N - 1):

$$-D\left[\frac{d^2\phi}{dr^2}|_{N-1} + \frac{a}{r_{N-1}}\frac{d\phi}{dr}|_{N-1}\right] + \sigma_a\phi_{N-1}\Delta r = \frac{1}{k}\nu\sigma_f\phi_{N-1}\Delta r$$

$$-D\left[\left(\frac{-2\phi_{N-1} + \phi_{N-2}}{\Delta r^2}\right) + \frac{a}{r_{N-1}}\left(\frac{-\phi_{N-2}}{2\Delta r}\right)\right] + \sigma_a\phi_{N-1}\Delta r = \frac{1}{k}\nu\sigma_f\phi_{N-1}\Delta r$$

Resulting in:

$$\left(-\frac{D}{\nu\sigma_f\Delta r^2} + \frac{aD}{2\nu\sigma_f r_{N-1}\Delta r}\right)\phi_{N-2} + \left(\frac{2D}{\nu\sigma_f\Delta r^2} + \frac{\sigma_a}{\nu\sigma_f}\right)\phi_{N-1} = \frac{1}{k}\phi_{N-1}$$

Similarly, the reflective boundary at $r = 0$, given by:

$$\frac{d\phi}{dr}|_0 = 0$$

Using Equation 34:

$$\frac{d^2\phi}{dr^2}|_0 = \frac{2(\phi_1 - \phi_0)}{\Delta r^2} + O(\Delta r^2)$$

This and previous definitions can then be substituted into the integrated diffusion equation for node 0 :

$$-D\left[\frac{d^2\phi}{dr^2}|_0 + \frac{a}{r_0}\frac{d\phi}{dr}|_0\right] + \sigma_a\phi_0\Delta r = \frac{1}{k}\nu\sigma_f\phi_0\Delta r$$

$$-D\left[\left(\frac{2(\phi_1 - \phi_0)}{\Delta r^2}\right) + \frac{a}{r_0}(0)\right] + \sigma_a\phi_0\Delta r = \frac{1}{k}\nu\sigma_f\phi_0\Delta r$$

Resulting in:

$$\left(\frac{-2D}{\nu\sigma_f\Delta r^2}\right)\phi_1 + \left(\frac{2D}{\nu\sigma_f\Delta r^2} + \frac{\sigma_a}{\nu\sigma_f}\right)\phi_0 = \frac{1}{k}\phi_0$$

With these results, the steady-state diffusion equation forms a full finite difference discretization system of equations in 1D coordinates for $0 \leq r \leq R$.

Finite Volume: Cartesian

The first step of the finite volume method is to integrate the equation across each cell. The following notation is for cell $i + \frac{1}{2}$:

$$\int_{r_i}^{r_{i+1}}\left[-D\left(\frac{d^2\phi}{dr^2}\right) + \sigma_a\phi\right]dr = \int_{r_i}^{r_{i+1}}\left(\frac{1}{k}\nu\sigma_f\phi\right)dr$$

$$-D\ \left[\ \frac{d\phi}{dr}|_{i+1} - \frac{d\phi}{dr}|_i\ \right]\ + \sigma_a \phi_{i+\frac{1}{2}} \Delta r = \frac{1}{k} \nu\sigma_f \phi_{i+\frac{1}{2}} \Delta r$$

The following definition, derived in Section 3.3 as Equation 37, can then be substituted into the integrated equation

$$\frac{d\phi}{dr}|_i = \frac{\phi_{i+\frac{1}{2}} - \phi_{i-\frac{1}{2}}}{\Delta r} + O(\Delta r^2)$$

which results in the following:

$$-D\left[\left(\frac{\phi_{i+\frac{3}{2}} - \phi_{i+\frac{1}{2}}}{\Delta r}\right) - \left(\frac{\phi_{i+\frac{1}{2}} - \phi_{i-\frac{1}{2}}}{\Delta r}\right)\right] + \ \sigma_a \phi_{i+\frac{1}{2}} \Delta r \ = \ \frac{1}{k} \nu\sigma_f \phi_{i+\frac{1}{2}} \Delta r$$

Resulting in:

$$\left(\frac{-D}{\nu\sigma_f \Delta r^2}\right) \phi_{i+\frac{3}{2}} + \left(\frac{2D}{\nu\sigma_f \Delta r^2} + \frac{\sigma_a}{\nu\sigma_f}\right) \phi_{i+\frac{1}{2}} + \left(\frac{-D}{\nu\sigma_f \Delta r^2}\right) \phi_{i-\frac{1}{2}} = \frac{1}{k} \phi_{i+\frac{1}{2}}$$

This equation satisfies the interior cells, but it still must be coupled with a discretization of each of the boundaries. The first of these is the outer vacuum boundary at $r = R$, indexed as $i = N$, is given by:

$$\phi_N = 0$$

Here we make use of Equation 42:

$$\frac{d\phi}{dr}|_N = \frac{\phi_{N-\frac{3}{2}} - 9\phi_{N-\frac{1}{2}}}{3\Delta r} + O(\Delta r^2).$$

This value, along with the vacuum boundary condition itself, can then be substituted into the integrated diffusion equation for the outer-most cell, indexed as cell (N - $\frac{1}{2}$) :

$$-D\left[\frac{d\phi}{dr}|_N - \frac{d\phi}{dr}|_{N-1}\right] + \sigma_a\phi_{N-\frac{1}{2}}\Delta r = \frac{1}{k}\nu\sigma_f\phi_{N-\frac{1}{2}}\Delta r$$

$$-D\left[\left(\frac{\phi_{N-\frac{3}{2}} - 9\phi_{N-\frac{1}{2}}}{3\Delta r}\right) - \left(\frac{\phi_{N-\frac{1}{2}} - \phi_{N-\frac{3}{2}}}{\Delta r}\right)\right] + \sigma_a\phi_{N-\frac{1}{2}}\Delta r = \frac{1}{k}\nu\sigma_f\phi_{N-\frac{1}{2}}\Delta r$$

Resulting in:

$$\left(\frac{-4D}{3\nu\sigma_f\Delta r^2}\right)\phi_{N-\frac{3}{2}} + \left(\frac{4D}{\nu\sigma_f\Delta r^2} + \frac{\sigma_a}{\nu\sigma_f}\right)\phi_{N-\frac{1}{2}} = \frac{1}{k}\phi_{N-\frac{1}{2}}$$

Similarly, the reflective boundary at $r = 0$, given by:

$$\frac{d\phi}{dr}|_0 = 0$$

This and previous definitions can then be substituted into the integrated diffusion equation for the inner-most cell, indexed as cell $\frac{1}{2}$:

$$-D\left[\frac{d\phi}{dr}|_1 - \frac{d\phi}{dr}|_0\right] + \sigma_a\phi_{\frac{1}{2}}\Delta r = \frac{1}{k}\nu\sigma_f\phi_{\frac{1}{2}}\Delta r$$

$$-D\left[\left(\frac{\phi_{\frac{3}{2}} - \phi_{\frac{1}{2}}}{\Delta r}\right) - (0)\right] + \sigma_a\phi_{\frac{1}{2}}\Delta r = \frac{1}{k}\nu\sigma_f\phi_{\frac{1}{2}}\Delta r$$

Resulting in:

$$\left(\frac{-D}{\nu\sigma_f\Delta r^2}\right)\phi_{\frac{3}{2}} + \left(\frac{D}{\nu\sigma_f\Delta r^2} + \frac{\sigma_a}{\nu\sigma_f}\right)\phi_{\frac{1}{2}} = \frac{1}{k}\phi_{\frac{1}{2}}$$

With these results, the steady-state diffusion equation forms a full finite volume discretization system of equations in 1D Cartesian coordinates for $0 \leq r \leq R$.

Finite Volume: Spherical

The first step of the finite volume method is to integrate the equation across each cell. The following notation is for cell $i+\frac{1}{2}$:

$$4\pi \int_{r_i}^{r_{i+1}} \left[-D \left(\frac{1}{r^2} \frac{d}{dr} \left(r^2 \frac{d\phi}{dr} \right) \right) + \sigma_a \phi \right] r^2 dr = 4\pi \int_{r_i}^{r_{i+1}} \left(\frac{1}{k} \nu \sigma_f \phi \right) r^2 dr$$

$$\int_{r_i}^{r_{i+1}} -D \, d \left(r^2 \frac{d\phi}{dr} \right) + \int_{r_i}^{r_{i+1}} \sigma_a \phi \, r^2 dr = \int_{r_i}^{r_{i+1}} \frac{1}{k} \nu \sigma_f \phi \, r^2 dr$$

$$-D \left(r^2 \frac{d\phi}{dr} \right) |_{r_i}^{r_{i+1}} + \sigma_a \phi_{i+\frac{1}{2}} \frac{r^3}{3} |_{r_i}^{r_{i+1}} = \frac{1}{k} \nu \sigma_f \phi_{i+\frac{1}{2}} \frac{r^3}{3} |_{r_i}^{r_{i+1}}$$

$$\frac{-3D \left(r_{i+1}^2 \frac{d\phi}{dr}|_{i+1} - r_i^2 \frac{d\phi}{dr}|_i \right)}{(r_{i+1}^3 - r_i^3)} + \sigma_a \phi_{i+\frac{1}{2}} = \frac{1}{k} \nu \sigma_f \phi_{i+\frac{1}{2}}$$

The following definition, derived in Section 3.3 as Equation 37, can then be substituted into the integrated equation

$$\frac{d\phi}{dr}|_i = \frac{\phi_{i+\frac{1}{2}} - \phi_{i-\frac{1}{2}}}{\Delta r} + O(\Delta r^2)$$

which results in the following:

$$\frac{-3D}{(r_{i+1}^3 - r_i^3)} \left[r_{i+1}^2 \left(\frac{\phi_{i+\frac{3}{2}} - \phi_{i+\frac{1}{2}}}{\Delta r} \right) - r_i^2 \left(\frac{\phi_{i+\frac{1}{2}} - \phi_{i-\frac{1}{2}}}{\Delta r} \right) \right] + \sigma_a \phi_{i+\frac{1}{2}} = \frac{1}{k} \nu \sigma_f \phi_{i+\frac{1}{2}}$$

Resulting in:

$$\left(\frac{-3Dr_{i+1}^2}{\Delta r (r_{i+1}^3 - r_i^3) \nu \sigma_f} \right) \phi_{i+\frac{3}{2}} + \left(\frac{3D(r_{i+1}^2 + r_i^2)}{\Delta r (r_{i+1}^3 - r_i^3) \nu \sigma_f} + \frac{\sigma_a}{\nu \sigma_f} \right) \phi_{i+\frac{1}{2}} + \left(\frac{-3Dr_i^2}{\Delta r (r_{i+1}^3 - r_i^3) \nu \sigma_f} \right) \phi_{i-\frac{1}{2}} = \frac{1}{k} \phi_{i+\frac{1}{2}}$$

This equation satisfies the interior cells, but it still must be coupled with a discretiza-

tion of each of the boundaries. The first of these is the outer vacuum boundary at $r = R$, indexed as $i = N$, is given by:

$$\phi_N = 0$$

Here we make use of Equation 42:

$$\frac{d\phi}{dr}|_N = \frac{\phi_{N-\frac{3}{2}} - 9\phi_{N-\frac{1}{2}}}{3\Delta r} + O(\Delta r^2).$$

This value, along with the vacuum boundary condition itself, can then be substituted into the integrated diffusion equation for the outer-most cell, indexed as cell (N - $\frac{1}{2}$) :

$$\frac{-3D\left(r_N^2 \frac{d\phi}{dr}|_N - r_{N-1}^2 \frac{d\phi}{dr}|_{N-1}\right)}{(r_N^3 - r_{N-1}^3)} + \sigma_a \phi_{N-\frac{1}{2}} = \frac{1}{k}\nu\sigma_f \phi_{N-\frac{1}{2}}$$

$$\frac{-3D}{(r_N^3 - r_{N-1}^3)}\left[r_N^2\left(\frac{\phi_{N-\frac{3}{2}} - 9\phi_{N-\frac{1}{2}}}{3\Delta r}\right) - r_{N-1}^2\left(\frac{\phi_{N-\frac{1}{2}} - \phi_{N-\frac{3}{2}}}{\Delta r}\right)\right] + \sigma_a \phi_{N-\frac{1}{2}} = \frac{1}{k}\nu\sigma_f \phi_{N-\frac{1}{2}}$$

Resulting in:

$$\left(\frac{3D(3r_N^2 + r_{N-1}^2)}{\Delta r(r_N^3 - r_{N-1}^3)\nu\sigma_f} + \frac{\sigma_a}{\nu\sigma_f}\right)\phi_{N-\frac{1}{2}} + \left(\frac{-D(r_N^2 + 3r_{N-1}^2)}{\Delta r(r_N^3 - r_{N-1}^3)\nu\sigma_f}\right)\phi_{N-\frac{3}{2}} = \frac{1}{k}\phi_{N-\frac{1}{2}}$$

Similarly, the reflective boundary at $r = 0$, given by:

$$\frac{d\phi}{dr}|_0 = 0$$

This and previous definitions can then be substituted into the integrated diffusion equation for the inner-most cell, indexed as cell $\frac{1}{2}$:

$$\frac{-3D\left(r_1^2\frac{d\phi}{dr}|_1 - r_0^2\frac{d\phi}{dr}|_0\right)}{(r_1^3 - r_0^3)} + \sigma_a\phi_{\frac{1}{2}} = \frac{1}{k}\nu\sigma_f\phi_{\frac{1}{2}}$$

$$\frac{-3D}{(r_1^3 - (0)^3)}\left[r_1^2\left(\frac{\phi_{\frac{3}{2}} - \phi_{\frac{1}{2}}}{\Delta r}\right) - (0)^2\,(0)\right] + \sigma_a\phi_{\frac{1}{2}} = \frac{1}{k}\nu\sigma_f\phi_{\frac{1}{2}}$$

Resulting in:

$$\left(\frac{-3D}{r_1\Delta r\nu\sigma_f}\right)\phi_{\frac{3}{2}} + \left(\frac{3D}{r_1\Delta r\nu\sigma_f} + \frac{\sigma_a}{\nu\sigma_f}\right)\phi_{\frac{1}{2}} = \frac{1}{k}\phi_{\frac{1}{2}}$$

With these results, the steady-state diffusion equation forms a full finite volume discretization system of equations in 1D spherical coordinates for $0 \leq r \leq R$.

Finite Volume: Cylindrical

The first step of the finite volume method is to integrate the equation across each cell. The following notation is for cell $i + \frac{1}{2}$:

$$2\pi\int_{r_i}^{r_{i+1}}\left[-D\left(\frac{1}{r}\frac{d}{dr}\left(r\frac{d\phi}{dr}\right)\right) + \sigma_a\phi\right]\,rdr = 2\pi\int_{r_i}^{r_{i+1}}\left(\frac{1}{k}\nu\sigma_f\phi\right)\,rdr$$

$$\int_{r_i}^{r_{i+1}} -D\,d\left(r\frac{d\phi}{dr}\right) + \int_{r_i}^{r_{i+1}}\sigma_a\phi\,rdr = \int_{r_i}^{r_{i+1}}\frac{1}{k}\nu\sigma_f\phi\,rdr$$

$$-D\left(r\frac{d\phi}{dr}\right)|_{r_i}^{r^{i+1}} + \sigma_a\phi_{i+\frac{1}{2}}\frac{r^2}{2}|_{r_i}^{r^{i+1}} = \frac{1}{k}\nu\sigma_f\phi_{i+\frac{1}{2}}\frac{r^2}{2}|_{r_i}^{r^{i+1}}$$

$$\frac{-2D\left(r_{i+1}\frac{d\phi}{dr}|_{i+1} - r_i\frac{d\phi}{dr}|_i\right)}{(r_{i+1}^2 - r_i^2)} + \sigma_a\phi_{i+\frac{1}{2}} = \frac{1}{k}\nu\sigma_f\phi_{i+\frac{1}{2}}$$

The following definition, derived in Section 3.3 as Equation 37, can then be substituted into the integrated equation

$$\frac{d\phi}{dr}|_i = \frac{\phi_{i+\frac{1}{2}} - \phi_{i-\frac{1}{2}}}{\Delta r} + O(\Delta r^2)$$

which results in the following:

$$\frac{-2D}{(r_{i+1}^2 - r_i^2)}\left[r_{i+1}\left(\frac{\phi_{i+\frac{3}{2}} - \phi_{i+\frac{1}{2}}}{\Delta r}\right) - r_i\left(\frac{\phi_{i+\frac{1}{2}} - \phi_{i-\frac{1}{2}}}{\Delta r}\right)\right] + \sigma_a\phi_{i+\frac{1}{2}} = \frac{1}{k}\nu\sigma_f\phi_{i+\frac{1}{2}}$$

Resulting in:

$$\left(\frac{-2Dr_{i+1}}{\Delta r(r_{i+1}^2 - r_i^2)\nu\sigma_f}\right)\phi_{i+\frac{3}{2}} + \left(\frac{2D(r_{i+1} + r_i)}{\Delta r(r_{i+1}^2 - r_i^2)\nu\sigma_f} + \frac{\sigma_a}{\nu\sigma_f}\right)\phi_{i+\frac{1}{2}} + \left(\frac{-2Dr_i}{\Delta r(r_{i+1}^2 - r_i^2)\nu\sigma_f}\right)\phi_{i-\frac{1}{2}} = \frac{1}{k}\phi_{i+\frac{1}{2}}$$

This equation satisfies the interior cells, but it still must be coupled with a discretization of each of the boundaries. The first of these is the outer vacuum boundary at $r = R$, indexed as $i = N$, is given by:

$$\phi_N = 0$$

Here we make use of Equation 42:

$$\frac{d\phi}{dr}|_N = \frac{\phi_{N-\frac{3}{2}} - 9\phi_{N-\frac{1}{2}}}{3\Delta r} + O(\Delta r^2).$$

This value, along with the vacuum boundary condition itself, can then be substituted into the integrated diffusion equation for the outer-most cell, indexed as cell (N - $\frac{1}{2}$) :

$$\frac{-2D\left(r_N\frac{d\phi}{dr}|_N - r_{N-1}\frac{d\phi}{dr}|_{N-1}\right)}{(r_N^2 - r_{N-1}^2)} + \sigma_a\phi_{N-\frac{1}{2}} = \frac{1}{k}\nu\sigma_f\phi_{N-\frac{1}{2}}$$

$$\frac{-2D}{(r_N^2 - r_{N-1}^2)}\left[r_N\left(\frac{\phi_{N-\frac{3}{2}} - 9\phi_{N-\frac{1}{2}}}{3\Delta r}\right) - r_{N-1}\left(\frac{\phi_{N-\frac{1}{2}} - \phi_{N-\frac{3}{2}}}{\Delta r}\right)\right] + \sigma_a\phi_{N-\frac{1}{2}} = \frac{1}{k}\nu\sigma_f\phi_{N-\frac{1}{2}}$$

Resulting in:

$$\left(\frac{2D(3r_N + r_{N-1})}{\Delta r(r_N^2 - r_{N-1}^2)\nu\sigma_f} + \frac{\sigma_a}{\nu\sigma_f}\right)\phi_{N-\frac{1}{2}} + \left(\frac{-2D(\frac{r_N}{3} + r_{N-1})}{\Delta r(r_N^2 - r_{N-1}^2)\nu\sigma_f}\right)\phi_{N-\frac{3}{2}} = \frac{1}{k}\phi_{N-\frac{1}{2}}$$

Similarly, the reflective boundary at $r = 0$, given by:

$$\frac{d\phi}{dr}|_0 = 0$$

This and previous definitions can then be substituted into the integrated diffusion equation for the inner-most cell, indexed as cell $\frac{1}{2}$:

$$\frac{-2D\left(r_1\frac{d\phi}{dr}|_1 - r_0\frac{d\phi}{dr}|_0\right)}{(r_1^2 - r_0^2)} + \sigma_a\phi_{\frac{1}{2}} = \frac{1}{k}\nu\sigma_f\phi_{\frac{1}{2}}$$

$$\frac{-2D}{(r_1^2 - (0)^2)}\left[r_1\left(\frac{\phi_{\frac{3}{2}} - \phi_{\frac{1}{2}}}{\Delta r}\right) - (0)\,(0)\right] + \sigma_a\phi_{\frac{1}{2}} = \frac{1}{k}\nu\sigma_f\phi_{\frac{1}{2}}$$

Resulting in:

$$\left(\frac{-2D}{r_1\Delta r\nu\sigma_f}\right)\phi_{\frac{3}{2}} + \left(\frac{2D}{r_1\Delta r\nu\sigma_f} + \frac{\sigma_a}{\nu\sigma_f}\right)\phi_{\frac{1}{2}} = \frac{1}{k}\phi_{\frac{1}{2}}$$

With these results, the steady-state diffusion equation forms a full finite volume discretization system of equations in 1D cylindrical coordinates for $0 \leq r \leq R$.

Appendix D. Error Notation

This effort uses "big O notation" to denote the order of magnitude of a mathematical term. Using this notation, a function f(x) equal to O(x) is on the order of x, meaning that there exists some positive constant, c, for which $0 \leq \mathrm{f}(x) \leq cx$ [27]. By this definition, $\mathrm{f}(x) = 2x^2 + x$ can be said to be $\mathrm{f}(x) = \mathrm{O}(x^2)$.

The following Taylor expansion of ϕ_{i+1} about point i:

$$\phi_{i+1} = \phi_i + \Delta r \frac{d\phi}{dr}|_i + \frac{1}{2!}(\Delta r)^2 \frac{d^2\phi}{dr^2}|_i + \frac{1}{3!}(\Delta r)^3 \frac{d^3\phi}{dr^3}|_i + \frac{1}{4!}(\Delta r)^4 \frac{d^4\phi}{dr^4}|_i + \ldots$$

could therefore be rewritten using big O notation as:

$$\phi_{i+1} = \phi_i + \Delta r \frac{d\phi}{dr}|_i + \frac{1}{2!}(\Delta r)^2 \frac{d^2\phi}{dr^2}|_i + \frac{1}{3!}(\Delta r)^3 \frac{d^3\phi}{dr^3}|_i + O(\Delta r^4)$$

or as:

$$\phi_{i+1} = \phi_i + \Delta r \frac{d\phi}{dr}|_i + \frac{1}{2!}(\Delta r)^2 \frac{d^2\phi}{dr^2}|_i + O(\Delta r^3)$$

where the O(Δr^3) term represents a larger error than O(Δr^4), but both of the equations represent the same Taylor expansion [21]. When this Taylor expansion is truncated to the following:

$$\phi_{i+1} = \phi_i + \Delta r \frac{d\phi}{dr}|_i + \frac{1}{2!}(\Delta r)^2 \frac{d^2\phi}{dr^2}|_i + \frac{1}{3!}(\Delta r)^3 \frac{d^3\phi}{dr^3}|_i$$

it can be said to be fourth order accurate, because its error is given by the term O(Δr^4). Therefore when the largest error term in a discretization is O(Δr^2), the discretization is said to be second order accurate [21].

Bibliography

1. T. Wimett, "FAST BURST REACTORS IN THE U.S.A. ," *Los Alamos Scientific Laboratory*, 1965.

2. E. Shabalin, *Fast Pulsed and Burst Reactors*, M. McTaggert, Ed. Pergamon Press, 1979.

3. National Nuclear Security Administration, "Prevent, Counter, and Respond - NNSA's Plan to Reduce Global Nuclear Threats FY2020 - FY2024," 2019.

4. Office of the Under Secretary of Defense For Acquisition, Technology, and Logistics, "Report of the Defense Science Board Task Force on Nuclear Weapon Effects Test, Evaluation, and Simulation," 2005.

5. E. Hobbs, "Asymptotic Neutronic Solutions for Fast Burst Reactor Design," 2017. [Online]. Available: https://digitalrepository.unm.edu/ne_etds/72

6. S. Abhyankar, J. Brown, E. M. Constantinescu, D. Ghosh, B. F. Smith, and H. Zhang, "Petsc/ts: A modern scalable ode/dae solver library," 2018.

7. T. R. Schmidt, *U.S. Fast Burst Reactors: Design and Operational History.* American Nuclear Society, 2017.

8. S. C. Wilson, "Development and Implementation of a Finite Element Solution of the Coupled Neutron Transport and Thermoelastic Equations Governing the Behavior of Small Nuclear Assemblies," 2005. [Online]. Available: http://hdl.handle.net/2152/3706

9. J. M. Wang, Q. L. X. Hui Gao, X. Q. Fan, and D. Z. Qian, "Development of a displacement-reactivity feedback model for dynamic behavior simulation in fast burst reactor," *Nuclear Science and Techniques*, vol. 30, p. 79, 2019.

10. E. E. Lewis and W. F. Miller, *Computational Methods of Neutron Transport*. New York: John Wiley Sons, Inc., 1984.

11. T. R. Schulmeister, "Modeling the White Sands Missile Range Fast Burst Reactor Using a Discrete Ordinates Code, PENTRAN," *Theses and Dissertations*, 2017. [Online]. Available: https://scholar.afit.edu/etd/1625

12. D. L. Hetrick, *Dynamics of Nuclear Reactors*. Chicago: The University of Chicago Press, 1971.

13. B. R. Betzler, *Calculating Alpha-Eigenvalue Spectra with Monte Carlo*. Los Alamos National Laboratory, Los Alamos, 2013.

14. K. D. Parsons, *Alpha-Like Calculations with MCNP*. Los Alamos National Laboratory, Los Alamos, 1997.

15. R. Modak and A. Gupta, "A scheme for the evaluation of dominant time-eigenvalues of a nuclear reactor," *Annals of Nuclear Energy*, vol. 34, pp. 213–221, 2007.

16. K. Singh, S. Degweker, R. Modak, and K. Singh, "Iterative method for obtaining the prompt and delayed alpha-modes of the diffusion equation," *Nuclear Science and Techniques*, vol. 38, pp. 1996–2004, 2011.

17. T. P. McLaughlin, S. P. Monahan, N. L. Pruvost, V. V. Frolov, B. G. Ryazanov, and V. I. Sviridov, *A Review of Criticality Accidents / 2000 Revision / LA-13638*. Los Alamos National Laboratory, 2000.

18. D.A.Brown, M.B.Chadwick, R.Capote, A.C.Kahler, M. A.Trkov, A.A.Sonzogni, Y.Danon, A.D.Carlson, M.Dunn, D.L.Smith, G.M.Hale, G.Arbanas, R.Arcilla, C.R.Bates, B.Beck, B.Becker, F.Brown, R.J.Casperson, J.Conlin,

D.E.Cullen, M.-A.Descalle, R.Firestone, T.Gaines, K.H.Guber, A.I.Hawari, J.Holmes, T.D.Johnson, T.Kawano, B.C.Kiedrowski, A.J.Koning, S.Kopecky, L.Leal, J.P.Lestone, C.Lubitz, J. Damián, C.M.Mattoon, E.A.McCutchan, S.Mughabghab, P.Navratil, D.Neudecker, G.P.A.Nobre, G.Noguere, M.Paris, M.T.Pigni, A.J.Plompen, B.Pritychenko, V.G.Pronyaev, D.Roubtsov, D.Rochman, P.Romano, P.Schillebeeckx, S.Simakov, M.Sin, I.Sirakov, B.Sleaford, V.Sobes, E.S.Soukhovitskii, I.Stetcu, P.Talou, I.Thompson, S. D. Marck, L.Welser-Sherrill, D.Wiarda, M.White, J.L.Wormald, R.Q.Wright, M.Zerkle, G.Žerovnik, and Y.Zhu, "ENDF/B-VIII.0: The 8th major release of the nuclear reaction data library with CIELO-project cross sections, new standards and thermal scattering data," *Nucl. Data Sheets*, vol. 148, pp. 1 – 142, 2018.

19. J. Duderstadt and L. Hamilton, *Nuclear Reactor Analysis*. New York: John Wiley Sons, Inc., 1976.

20. S. Mazumder, *Numerical Methods for Partial Differential Equations: Finite Difference and Finite Volume Methods*. San Diego, CA: Elsevier, 2016.

21. C. F. Gerald and P. O. Wheatley, *Applied Numerical Analysis*, 7th ed. Boston, MA: Pearson Education, Inc., 2004.

22. G. I. Bell and S. Glasstone, *Nuclear Reactor Theory*. New York, NY: Van Nostrand Reinhold Company, 1970.

23. S. Abhyankar and A. J. Flueck, "Real-time power system dynamics simulation using a parallel block-jacobi preconditioned newton-gmres scheme," *2012 SC Companion: High Performance Computing, Networking Storage and Analysis*, 2012.

24. S. Hoshyari and C. F. O. Gooch, "A higher-order unstructured finite volume solver for three-dimensional compressible flows," *2018 AIAA Aerospace Sciences Meeting*, 2018.

25. H. B. Z. Syed, C. Farquharson, and S. MacLachlan, "Block preconditioning techniques for geophysical electromagnetics," *Computational Methods in Science and Engineering*, 2020.

26. R. L. Burden and J. D. Faires, *Numerical Analysis*, 8th ed. Belmont, CA: Thomson Brooks/Cole, 2005.

27. P. E. Black, "big-o notation," *Dictionary of Algorithms and Data Structures*, 2019. [Online]. Available: https://www.nist.gov/dads/HTML/bigOnotation.html

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
21–03–2021	Master's Thesis	Sept 2019 — Mar 2021

4. TITLE AND SUBTITLE

Nonlinear Solution of the Time Eigenvalue of a Fast Burst Reactor Using the Finite Volume Method

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Baxter, Stephen H., 1st Lt, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering an Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENP-MS-21-M-100

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

National Nuclear Security Administration
POC: Mr. Kent Jones
NA-18 Office of Systems Engineering and Integration
Kirtland AFB, NM 87117

10. SPONSOR/MONITOR'S ACRONYM(S)

NNSA

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This effort models fast burst reactors using the one dimensional, one group neutron diffusion equation to solve for the time eigenvalue, a method for which an analytical solution exists against which the numerical results can be verified. An existing solution method is enhanced by the addition of a second order accurate finite volume discretization, which is then used to model two separate fast burst reactors. The results of these models are then compared to the results of previous work, the analytical solution, and existing experimental burst width data for each of the two reactors.

15. SUBJECT TERMS

Fast burst reactors, Modeling and simulation, Finite volume method, Neutron diffusion equation

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			LTC Edward Hobbs, AFIT/ENP
U	U	U	U	94	19b. TELEPHONE NUMBER *(include area code)* (937) 255-6565; edward.hobbs@afit.edu