AN ADAPTIVE RUNGE-KUTTA-FEHLBERG METHOD FOR TIME-DEPENDENT DISCRETE ORDINATE TRANSPORT

A Dissertation
Presented to
The Academic Faculty

by

Christopher A. Edgar

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in Nuclear Engineering in the
George W. Woodruff School of Mechanical Engineering

Georgia Institute of Technology
August 2015

AN ADAPTIVE RUNGE-KUTTA-FEHLBERG METHOD FOR TIME-DEPENDENT DISCRETE ORDINATE TRANSPORT

Approved by:

Dr. Farzad Rahnema, Chair
George W. Woodruff School of Mechanical Engineering
Georgia Institute of Technology

Dr. Bojan Petrovic
George W. Woodruff School of Mechanical Engineering
Georgia Institute of Technology

Dr. Glenn E. Sjoden
Chief Scientist
Air Force Technical Applications Center

Dr. Ce Yi
George W. Woodruff School of Mechanical Engineering
Georgia Institute of Technology

Dr. Alireza Haghighat
Department of Mechanical Engineering
Virginia Polytechnic Institute and State University

Dr. Mark Costello
School of Aerospace Engineering
Georgia Institute of Technology

Date Approved: July 9, 2015

To my beautiful wife, Ashley. For your unconditional love, faithful encouragement, and resolute support. You have always believed in and inspired the pursuit of my dreams.

ACKNOWLEDGEMENTS

I would first like to express my immense gratitude to my advisor, Dr. Glenn Sjoden. What started as an undergraduate summer research project transformed into my path in graduate school. As both an advisor and a friend, thank you for all the years of valuable instruction, guidance, and support. You have been the best advisor, mentor, and advocate a student could ask for.

My sincere appreciation goes to the members of my dissertation reading committee: Dr. Farzad Rahnema, Dr. Alireza Haghighat, Dr. Bojan Petrovic, Dr. Ce Yi, and Dr. Mark Costello. Thank you all for your instruction, insights, service, and guidance through my years in graduate school.

I am indebted to my colleagues in the former Computational Radiation Transport Laboratory (CRiTCEL) at the Georgia Institute of Technology, especially Dr. Kevin Manalo, who has provided a much needed sounding board during this process. I wish to express my appreciation to Mr. Michael Chin, Dr. Jessica Paul, and Mr. Evan Redd for their kind words of support and friendship these past few years.

I would like to express my tremendous appreciation for my wife, Ashley, for her unwavering love, support and inspiration – completing this dissertation would have been much harder without her. Lastly, I would like to thank my family, especially my parents, Christopher and Kelly, for their ongoing support and encouragement throughout my life.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ANL	Argonne National Laboratory
ARKTRAN-TD	Adaptive Runge-Kutta Time-Dependent Transport Code
BWR	Boiling Water Reactor
DD	Diamond Differencing Scheme
LANL	Los Alamos National Laboratory
ORNL	Oak Ridge National Laboratory
pcm	Per Cent Mil
PENTRAN	Parallel Environment Neutral Particle Transport Code
RKF	Runge-Kutta-Fehlberg
RKF1/2	Runge-Kutta-Fehlberg 1^{st} Order with 2^{nd} Order Truncation Error Control
RKF2/3	Runge-Kutta-Fehlberg 2^{nd} Order with 3^{rd} Order Truncation Error Control
RKF3/4	Runge-Kutta-Fehlberg 3^{rd} Order with 4^{th} Order Truncation Error Control
RKF4/5	Runge-Kutta-Fehlberg 4^{th} Order with 5^{th} Order Truncation Error Control
S_N	Discrete Ordinate

LIST OF SYMBOLS

Ψ	Angular Flux
λ_l	Decay Constants of Delayed Group, , *l=, ndg*
χ_d^l	Delayed Fission Spectra, *l=1,ndg*
β_l	Delayed Neutron Fraction for Delayed Group, *l=, ndg*
$C_l(z,t)$	Delayed Precursor Concentration for Delayed Group¸ *l=, ndg*
E	Energy
q^{ext}	External Source Term
μ	Flight Direction
Δt	Length of Time
ν	Neutron Velocity
ndg	Number of Delayed Neutron Groups
β	Overall Delayed Neutron Fraction
χ_P	Prompt Fission Yield
$\nu\sigma_f$	Prompt Neutron Yield Times Fission Cross Section
ϕ	Scalar Flux
σ_s	Scatter Cross Section
t	Time
σ	Total Cross Section

SUMMARY

This dissertation focuses on the development and implementation of a new method to solve the time-dependent form of the linear Boltzmann transport equation for reactor transients. This new method allows for a stable solution to the fully explicit form of the transport equation with delayed neutrons by employing an error-controlled, adaptive Runge-Kutta-Fehlberg (RKF) method to differentiate the time domain. Allowing for the time step size to vary adaptively and as needed to resolve the time-dependent behavior of the angular flux and neutron precursor concentrations. The RKF expansion of the time domain occurs at each point and is coupled with a Source Iteration to resolve the spatial behavior of the angular flux at the specified point in time. The decoupling of the space and time domains requires the application of a quasi-static iteration between solving the time domain using adaptive RKF with error control and resolving the space domain with a Source Iteration sweep. The research culminated with the development of the 1-D Adaptive Runge-Kutta Time-Dependent Transport code (ARKTRAN-TD), which successfully implemented the new method and applied it to a suite of reactor transient benchmarks.

CHAPTER 1

INTRODUCTION

Solution methods of time-dependent radiation transport problems require a significant increase in computational time (and memory) relative to steady-state (time-independent) problems. For reactor transient problems, the inclusion of delayed neutrons is imperative when modeling realistic transient behavior. However, these delayed neutrons evolve over multiple time scales, which span up to six orders of magnitude. The solution to the fully explicit form of the transport equation, including explicit definitions of the delayed neutrons, can lead to unstable solutions when a fixed time step is applied. This effect is partially due to the large difference in time step based on prompt versus delayed neutron behavior.

1.1 Brief Discussion of Previous Works in Time-Dependent S_N Transport

As mentioned previously, time-dependent discrete ordinates transport codes have either functioned using a fully-implicit or semi-implicit solution methods to date, which allows for the stability of the time-dependent transport equation with fixed time steps. The definition of implicit and explicit form can be found in Eq. 1.1, where *H* represents the entire transport operator, and *n* refers to the time step.

$$\frac{\Psi^{(n+1)} - \Psi^{(n)}}{v\Delta t_n} = \begin{cases} H^{(n)}\Psi^{(n)} & explicit \\ H^{(n+1)}\Psi^{(n+1)} & implicit \end{cases} \tag{1.1}$$

The most recent work by Pautz and Birkhofer has shown success in solving the time-dependent transport equation using a discrete ordinates transport solver in 2-D by

separating the time and space domains in TORT-TD. (Pautz and Birkhofer 2003) In this work, Pautz and Birkhofer use a combination of the implicit and explicit transport equation using the relationship in Eq. 1.2

$$\frac{\Psi^{(n+1)} - \Psi^{(n)}}{v\Delta t_n} = \frac{1}{2}H^{(n)}\Psi^{(n)} + \frac{1}{2}H^{(n+1)}\Psi^{(n+1)} \qquad (1.2)$$

The form of Eq. 1.2 requires both a multiplication and inversion of the transport operator, *H*. Using the implicit discretized form in Eq. 1.1 and applying a significant amount of algebra, one can arrive at a form of the transport equation that is fully implicit. However, the neutron precursor concentrations are still differenced explicitly. Another solution method, employed in the TIMEX code casts the transport equation explicitly (with the exception of upscatter neutrons and precursors). However, the difference in neutron precursors is performed by a fully implicit scheme which results in a scheme that is semi-implicit overall. (Hill and Reed 1976) . Fixed time steps were applied in both of these solution methods, which are unconditionally stable by definition. The numerical accuracy of the time difference is 1st order with 2nd order truncation error for these works.

1.2 Introduction to New Method for Time-Dependent Transport

This research work develops a new method, which allows for a stable solution to the fully explicit form of the transport equation with delayed neutrons by employing an error-controlled, adaptive Runge-Kutta-Fehlberg (RKF) method to difference the time domain. This method allows for the time step size to vary adaptively and as needed to resolve the time-dependent behavior of the angular flux and neutron precursor concentrations. Furthermore, current methods used in time-dependent discrete ordinates codes only differentiate the time domain to 1st order, resulting in truncation error of 2nd

order. The RKF expansion about points in time can be performed for a variety of expansion order pairs, of which the 4^{th} order expansion with 5^{th} order truncation error is most popular.

By limiting the size of the truncation error, one can ensure stability of the fully explicit form of the equation by mandating that the time step selected results in a truncation error (in both angular flux and delayed neutron precursors) falls within a specified tolerance. If this is not met, the time step size can be reduced and the updated truncation error determined. The process then repeats until a time step is found that meets this criteria. In an effort to reduce the number of these repetitions, the RKF method is performed on each point in space. However, this decoupling of the space domain from the time domain requires the implementation of a quasi-static iteration between solving the time domain using adaptive RKF with error control and resolving the space domain with a standard Source Iteration space sweep. The theory behind this new method is presented in Chapter 3.

The following chapter addresses the general theory applied to the discrete ordinates approximation to the linear Boltzmann transport equation and methodology behind the Runge-Kutta-Fehlberg expansion with error control. Chapter 4 provides benchmark results for the newly developed 1-D Adaptive Runge-Kutta Time-Dependent Transport code (ARKTRAN-TD) when solving steady-state solutions (the starting basis for reactor transient calculations). Chapter 5 presents ARKTRAN-TD results to a suite of benchmark problems and provides comparisons to other time-dependent codes. Chapter 6

presents a newly implemented adaptive RKF scheme, which automatically selects the optimum RKF expansion order pair that allows for the largest stable time step at any point in the transient simulation. Chapter 7 concludes the dissertation and discusses the possibilities for future works.

CHAPTER 2

BACKGROUND

Many different methods have been explored in an effort to solve the steady-state form of the linear Boltzmann transport equation, including Discrete Ordinates, Monte Carlo, Collision Probability, and Even-Parity Methods to name a few. (Lewis and Miller 1993) (Bell and Glasstone 1970) Furthermore, there are many mathematical methods available for solving differential equations in the time domain, including the Runge-Kutta-Fehlberg Method, which allows for one to determine the optimum time step rather than simply apply a global limit on the step size. (Fehlberg 1969) This dissertation focuses on utilizing the Discrete Ordinates Method for solving the transport equation, in order to resolve the spatial behavior of the neutron flux, followed by coupling with a Runge-Kutta-Fehlberg numerical method to resolve the behavior of the neutron flux in the time domain using optimized varying time steps. The full application of these two methods to the transport equation, assumptions, and coupling between them are addressed in Chapter 3. However, the subsequent sections in this Chapter are devoted to providing an introduction to these methods in the general sense.

2.1 The Discrete Ordinates (S_N) Method

The Discrete Ordinates (S_N) Method results in a discretized form of the transport equation for the space, energy, and angular domains. This dissertation focuses on the development of a new solution method for the time-dependent form of the transport

equation, in one spatial dimension. Thus, the 1-D form of the time-dependent transport equation is presented in Eq. 2.1, below. (Lewis and Miller 1993)

$$\begin{aligned}\frac{1}{\nu(E)}\frac{\partial}{\partial t}\psi(z,E,\mu,t)+\mu\cdot\vec{\nabla}\psi(z,E,\mu,t)+\ \sigma(z,E)\,\psi(z,E,\mu,t)\\ =\int_0^\infty dE'\int d\mu'\,\sigma_s(z,E'\to E,\mu\cdot\mu')\,\psi(z,E',\mu',t)\\ +\chi_P(z,E)[1-\beta]\int_0^\infty dE'\int d\mu'\,\nu\sigma_f(z,E')\,\psi(z,E',\mu',t)\\ +\sum_{l=1}^{ndg}\chi_d^l(z,E)\lambda_l C_l(z,t)+q^{ext}(z,E,\mu,t)\end{aligned} \tag{2.1}$$

where,

$\psi(z,E,\mu,t)$ = angular flux

$\nu(E)$ = neutron velocity

μ = flight direction

σ, σ_s, and $\nu\sigma_f$ = total, scatter, and prompt neutron yield times fission cross sections

ndg = number of delayed neutron groups

χ_P and χ_d^l = prompt and delayed fission spectra, *l=1*,*ndg*

β, β_l, *and* λ_l = overall fraction, delayed fraction, and decay constants of delayed neutron groups

$q^{ext}(z,E,\mu,t)$ = external source term

The neutron precursor concentrations $C_l(z,t)$, are governed by:

$$\frac{\partial}{\partial t}C_l(z,t)=\ -\lambda_l C_l(\vec{r},t)+\beta_l\int dE'\nu(E')\sigma_f(z,E')\,\phi(z,E',t) \tag{2.2}$$

2.1.1 Discretization of the Energy Variable

The energy domain of the transport equation can be partitioned into a number of discrete energy ranges, commonly referred to as "multi-group." (Bell and Glasstone 1970) Traditionally, one starts with the highest energy particles (g=1) and ends with the lowest energy particles in the highest group index (g=G). Thus, particles in energy group g are defined with energies between E_{g-1} and E_g. Where E_{g-1} denotes the highest energy particle contained in group g, and E_g is the loewst energy particle in group g. Overall, the result of this discretization is an approximation to the linear Boltzmann transport equation, whereby the angular flux in each discrete energy group is defined by Eq. 2.3.

$$\psi_g(z,\mu,t) = \int_g dE(z,E,\mu,t) = \int_{E_g}^{E_{g-1}} dE\,\psi(z,E,\mu,t) \tag{2.3}$$

Furthermore, the energy integrals are partitioned to represent the contribution from each energy group.

$$\int_0^\infty \mathrm{dE}' \approx \sum_{g'=1}^{G} \int_{g'} dE' \tag{2.4}$$

By integrating Eq. 2.1 over energies between E_{g-1} and E_g, one obtains the multigroup form of the transport equation. Whereby, each discrete energy domain is defined by its own transport equation, which is coupled to the other discrete energy ranges via the multigroup cross sections. Thus, the multigroup cross sections representing total, fission, and scattering reactions can be defined by Eqs. 2.5-2.7.

$$\sigma_{t,g}(z) = \frac{\int_{E_g}^{E_{g-1}} dE \int d\mu\,\sigma_t(z,E)\,\psi(z,E,\mu,t)}{\int_{E_g}^{E_{g-1}} dE \int d\mu\,\psi(z,E,\mu,t)} \tag{2.5}$$

$$\sigma_{f,g}(z) = \frac{\int_{E_g}^{E_{g-1}} dE \int d\mu \sigma_f(z,E)\,\psi(z,E,\mu,t)}{\int_{E_g}^{E_{g-1}} dE \int d\mu \psi(z,E,\mu,t)} \tag{2.6}$$

$$\sigma_{s,gg'}(z,\mu\cdot\mu')$$

$$= \frac{\int_{E_{g'}}^{E_{g'-1}} dE' \int_{E_g}^{E_{g-1}} dE \int d\mu' \sigma_s(z,E'\to E,\mu'\cdot\mu)\,\psi(z,E',\mu',t)}{\int_{E_g}^{E_{g-1}} dE' \int d\mu' \psi(z,E',\mu',t)} \tag{2.7}$$

The angular dependency of the neutron scattering cross-section can be expanded using spherical harmonics (Eq. 2.8 and 2.9). The scattering cross-section is assumed to only be dependent on the cosine of the scattering angle ($\mu_0 = \mu\cdot\mu'$); where μ' and μ represent the direction of the particle before and after the scattering event, respectively. This implies the probability of scattering into direction μ' with energy g' is independent of the initial direction of the particle.

$$\sigma_{s,g'\to g}(z,\mu_0) \cong \sum_{l=1}^{L} (2l+1)\,\sigma_{sl,g'\to g}(z) P_l(\mu_0) \tag{2.8}$$

$$\sigma_{sl,g'\to g}(z) = \int_{-1}^{1} \frac{d\mu_0}{2}\,\sigma_{s,g'\to g}(z,\mu_0)\,P_l(\mu_0) \tag{2.9}$$

It is important to note that, as neutron energies become very low (i.e. cold neutrons), this assumption becomes an approximation. Furthermore, in 1-D Cartesian coordinates, the streaming operator (the $\mu\cdot\vec{\nabla}$ in Eq. 2.1) becomes:

$$\mu\cdot\vec{\nabla} = \mu\frac{\partial}{\partial z} \tag{2.10}$$

Substituting Eqs. 2.8 and 2.10 into Eq. 2.1, and using the definition of multigroup cross sections Eqs. 2.5-2.7, one arrives at:

$$\frac{1}{v_g}\frac{\partial}{\partial t}\psi_g(z,\mu,t) + \mu\cdot\frac{\partial}{\partial z}\psi_g(z,\mu,t) + \sigma_t(z)\psi_g(z,\mu,t)$$

$$= \sum_{g'=1}^{G}\sum_{l=0}^{L}(2l+1)\sigma_{sl,g'\to g}(z)P_l(\mu)\phi_{l,g'}(z,t)$$

$$+\chi_{P,g}(z)[1-\beta]\sum_{g'=1}^{G}\nu\sigma_{f,g'}(z)\phi_{0,g'}(z,t) \qquad (2.11)$$

$$+\sum_{l=1}^{ndg}\chi_{d,g}^{l}(z)\lambda_l C_l(z,t) + q_g^{ext}(z,\mu,t)$$

The scalar flux moments in Eq. 2.11 are defined as follows:

$$\phi_{l,g}(z,t) = \int_{-1}^{1}\frac{d\mu}{2}P_l(\mu)\,\psi_g(z,\mu,t) \qquad (2.12)$$

2.1.2 Discretization of the Angular Variable

The angular component of the transport equation, denoted by μ in the one-dimensional form, is discretized by considering a finite number of discrete directions (referred to as ordinates). By requiring the transport equation (Eq. 2.11) to hold for a distinct number of angles μ_n, and then applying an accurate quadrature approximation to the integral terms, one can arrive at the multigroup form of the time-dependent transport equation in 1-D, shown in Eq. 2.13 below. (Lewis and Miller 1993) The angular flux $\psi_g(z,\mu,t)$, is now written as $\psi_{n,g}(z,t)$, where *n* refers to the angle or ordinate number in the quadrature set. Thus, one must choose a quadrature set on the interval of $-1 \le \mu \le 1$, having *N* ordinates with corresponding weights w_n.

$$\frac{1}{v_g}\frac{\partial}{\partial t}\psi_{n,g}(z,t) + \mu_n\frac{\partial}{\partial z}\psi_{n,g}(z,t) + \sigma_g(z,t)\psi_{n,g}(z,t)$$

$$= \sum_{l=0}^{L}(2l+1)P_l(\mu_n)\sum_{g'=1}^{G}\sigma_{l,g'\to g}(z,t)\phi_{l,g'}(z,t)$$

$$+ \chi_{P,g}(z)[1-\beta]\sum_{g'=1}^{G}\nu\sigma_{f,g'}(z,t)\phi_{g'}(z,t) \qquad (2.13)$$

$$+ \sum_{l=1}^{ndg}\chi_{d,g}^{l}(z)\lambda_l C_l(z,t) + q_{n,g}^{ext}(z,t)$$

The scalar flux and scalar flux moments, in Eq. 2.13, are approximated by the quadrature set and can be reconstituted as follows using the associated weights of the quadrature.

$$\phi_g(z,t) = \frac{1}{2}\sum_{n=1}^{N} w_n\psi_{n,g}(z,t) \qquad (2.14)$$

$$\phi_{l,g}(z,t) = \frac{1}{2}\sum_{n=1}^{N} w_n P_l(\mu_n)\psi_{n,g}(z,t) \qquad (2.15)$$

The transport code developed as part of this dissertation utilized Gauss-Legendre quadrature sets from S_2 to S_{20}, Appendix A provides the quadrature angles μ_n and the weights w_n associated with the angle, for S_n orders from 2 to 20 in one-dimension.

2.1.3 Discretization of the Spatial Variable

The time-dependent form of the linear Boltzman equation, as it appears in Eq. 2.13 can be recast in an abbreviated form as follows, where the entire right hand side of the equation has been collapsed into $Q_{n,g}(z,t)$, and the time-derivative term has been moved from the left-hand side to the right-hand side.

$$\left(\mu_n \frac{\partial}{\partial z} + \sigma_{t,g}(z)\right) \psi_{n,g}(z,t) = Q_{n,g}(z,t) \qquad (2.16)$$

The angular dependence in the equation is denoted by *n*, which represents the discrete ordinate direction of the angular flux from $n = 1$, to N, with N being the total number of angles. The energy dependence is denoted by the subscript, *g*, referencing the energy group from $g = 1$, to G, with G being the total number of energy groups. The spatial domain is the partitioned into spatial cells. Within each of these spatial cells, the material cross sections and source terms are assumed to be constant. Integrating Eq. 2.16 over the volume of the cell and then dividing by the cell volume, results in the volume and surface averaged angular flux values.

$$\frac{\mu_n}{\Delta z_i}\left(\psi_{i+\frac{1}{2},n,g}(t) - \psi_{i-\frac{1}{2},n,g}(t)\right) + \sigma_{i,t,g}\,\psi_{i,n,g}(t) = Q_{i,n,g}(t) \qquad (2.17)$$

The index *i* refers to the cell center value, while $i\pm1/2$, represents the cell surface values. Furthermore, these cells can be defined based on a coarse and fine grid. Whereby, the fine grid is contained within a coarse grid, thereby allowing for discontinuous cell size in the spatial dimension. The ability to model a system using a coarse/fine cell grid allows for refinement in areas where there are steep flux gradients or optically thin/thick cells and mitigates one problem area from dictating the cell size for the entire spatial domain. Therefore, this enables an increase in computation speed and decrease in memory storage requirements.

2.1.4 Differencing Scheme, Boundary Conditions, and Iterative Methods

The solution to the discrete ordinate transport equations in a phase space are obtained through a process referred to as a *"transport sweep."* Whereby the solutions are

calculated by marching along each ordinate, in the direction of particle flight. (Lewis and Miller 1993) For each fine cell in the system, the incoming angular flux is known on one boundary. Thus, a relationship can be formed between this incoming angular flux and the cell averaged volumetric source term (assumed to be known from the previous iterations), to the outgoing angular flux for each ordinate. These relationships are known as *"differencing schemes."*

2.1.4.1 Diamond Difference (DD)

While there are several forms of differencing schemes available for use in transport calculations, the Diamond Difference (DD) Scheme was used in the transport code developed for this dissertation; and will be presented in this section for comprehensiveness.

In the Diamond Differencing Scheme, the cell averaged angular flux is represented by the mathematical average of any two opposite cell boundary fluxes, or in equation form:

$$\psi_{i,n,g}(t) = \frac{1}{2}\left(\psi_{i+\frac{1}{2},n,g}(t) + \psi_{i-\frac{1}{2},n,g}(t)\right) \tag{2.18}$$

In order to obtain the cell center angular flux, when traveling in the direction where μ_n is positive, one must eliminate the outgoing angular flux in Eq. 2.17; since it is guaranteed to have the incoming angular flux in the direction of flight, either from boundary conditions or the previous cell calculation. In doing so, one arrives at the following:

$$\psi_{i,n,g}(t) = \frac{\frac{2\mu_n}{\Delta z_i}\left(\psi_{i-\frac{1}{2},n,g}(t)\right) + \ Q_{i,n,g}(t)}{\sigma_{i,t,g} + \frac{2\mu_n}{\Delta z_i}} \tag{2.19}$$

The outgoing angular flux can then be evaluated by recasting Eq. 2.18 in terms of the cell averaged angular flux (just obtained) and the incoming angular flux on each ordinate, taking into account if μ_n is positive or negative. For example, if μ_n is positive, the outgoing angular flux would be given by:

$$\psi_{i+\frac{1}{2},n,g}(t) = 2\,\psi_{i,n,g}(t) - \psi_{i-\frac{1}{2},n,g}(t) \tag{2.20}$$

The diamond differencing scheme is accurate to second order, but may result in negative flux solutions. (Lewis and Miller 1993) To mitigate this nonphysical issue, one can reduce the cell size or set the negative fluxes to zero and recalculate the cell average flux to maintain a balance of particles. Furthermore, it is important to note that non-physical oscillations can be inherent in solutions obtained through diamond differencing schemes. (Petrovic and Haghighat 1995)

2.1.4.2 Boundary Conditions

Three standard boundary conditions: albedo, specular reflective, and vacuum (zero return angular flux), can be expressed by:

$$\psi_g(z, \mu_n, t) = \alpha\,\psi_g(z, \mu'_n, t) \tag{2.21}$$

where $\mu_n = -\mu'_n$ and α is defined based on the type of boundary condition. Due to the nature of the even order symmetric quadrature set, where N is the number of quadrature angles, we have

$$\mu_{N+1-n} = -\mu_n, \quad for\ n = 1, 2, \dots, \frac{N}{2} \tag{2.22}$$

and

$$w_{N+1-n} = w_n. \tag{2.23}$$

With even ordinate sets, one can then define the specular reflective boundary condition, at a z boundary location z = 0 as:

$$\psi_{g,n}(0, t) = \psi_{g,N+1-n}(0, t), \quad for\ n = 1, 2, \dots, \frac{N}{2}. \tag{2.24}$$

One can also define the vacuum, or zero return boundary condition, on the right hand boundary z = a, by:

$$\psi_{g,n}(a, t) = 0, \quad for\ n = 1, 2, \dots, \frac{N}{2}. \tag{2.25}$$

Lastly, one can define an albedo (partially reflective) boundary condition, similarly to the reflective boundary condition in Eq. 2.24, but with the addition of the energy group dependent α_g, which provides the fraction of particles which are reflected back into the system. For example, at the general boundary z_{bnd}, the albedo condition can be defined as:

$$\psi_{g,n}(z_{bnd}, t) = \alpha_g \psi_{g,N+1-n}(z_{bnd}, t), \quad for\ n = 1, 2, \dots, \frac{N}{2}. \tag{2.26}$$

2.1.4.3 Iterative Methods

Based on the integro-differential nature of the transport equation, the solution of the multigroup discrete ordinates equations can be obtained via the use of an iterative process. For fixed-source problems, only the Source Iteration Method is required. For eigenvalue problems (steady-state), both the Source Iteration and Power Iteration

Methods are required to obtain a converged spatial flux profile as well as an accurate eigenvalue.

2.1.4.3.1 Source Iteration Method

The Source Iteration Method, defines the process of guessing a source (generally the in-group scattering source), and then sweeping through the angular, spatial and energy domains of the discretized system representing the problem geometry. (Lewis and Miller 1993) After completion of the initial transport sweep, the scalar flux and flux moment solutions are calculated from the angular fluxes and an updated in-group scattering source is determined. The process is repeated until a convergence criteria (or tolerance) is met for each cell averaged angular flux solution, on all ordinates, for each energy group, and at every spatial location, as given by Eq. 2.27.

$$\frac{\left|\psi^{i}-\psi^{i-1}\right|}{\psi^{i-1}}<\in_{\psi} \tag{2.27}$$

Typical the convergence criteria, $\in_{\psi}$ will vary from problem to problem and may depend on the importance of a particular region's solution to the rest of the problem. Values for $\in_{\psi}$ generally span from 10^{-3} to 10^{-4} for most applications, depending upon the problem.

2.1.4.3.2 Power Iteration Method

Criticality eigenvalue problems can be solved with the use of the Power Iteration Method, whereby the eigenvalue of the system is associated with the nonnegative distribution of fission neutrons. Traditionally, source iterations are performed based on an initial guess of the angular flux (at all spatial locations, for each angle, and every energy group). After the convergence of the source iteration, a power iteration is performed to obtain the updated eigenvalue. This process is then repeated until both the source

iteration convergence criteria (eq. 2.27) and the power iteration tolerance (Eq. 2.29) have been met. The eigenvalue for the ℓ^{th} power iteration can be calculated using Eq. 2.28.

$$k^{\ell} = k^{\ell-1} \frac{\int dE \int dz\, \nu\sigma_f(z,E)\phi^{\ell}(z,E)}{\int dE \int dz\, \nu\sigma_f(z,E)\phi^{\ell-1}(z,E)} \tag{2.28}$$

The typical initial guess for the eigenvalue *k* is 1.00, however if the eigenvalue of the system can be guessed more accurately at the beginning of the problem, faster convergence of the system may be obtained. Convergence criteria for the eigenvalue $\in_k$, is typically taken to be 10^{-5} at a minimum.

$$\frac{\left|k^{\ell} - k^{\ell-1}\right|}{k^{\ell-1}} < \in_k \tag{2.29}$$

It is important to note, in the computational framework for reactor physics solvers, power iterations are often called *"outer iterations,"* while source iterations are referred to as *"inner iterations."* This nomenclature is derived from the location of the source iteration calculation being inside the computational loop that performs the power iteration.

2.2 The Runge-Kutta-Fehlberg (RKF) Method

The Runge-Kutta-Fehlberg (RKF) Method utilizes the traditional Runge-Kutta method of recurrence (Eq. 2.30)

$$y_{i+1} = y_i + a_1k_1 + a_2k_2 + a_3k_3 + \cdots + a_nk_n \tag{2.30}$$

to calculate successive values of the dependent variable *t* of differential equations of the form shown in Eq. 2.31. (James, Smith and Wolford 1993)

$$\frac{dy}{dt} = f(t,y) \tag{2.31}$$

where, k_n can be defined with a step size of h as:

$$
\begin{aligned}
k_1 &= hf(t_i, y_i) \\
k_2 &= hf(t_i + p_1 h, y_i + q_{11} k_1) \\
k_3 &= hf(t_i + p_2 h, y_i + q_{21} k_1 + q_{22} k_2) \\
&\ldots \\
k_n &= hf\big(t_i + p_{n-1} h, y_i + q_{n-1,1} k_1 + q_{n-1,2} k_2 + \cdots \\
&\qquad + q_{n-1,n-1} k_{n-1}\big)
\end{aligned}
\tag{2.32}
$$

Eqs. 2.30 and 2.32 constitute the Runge-Kutta equations, which may be written more compactly with Eqs. 2.33 and 2.34.

$$y_{i+1} = y_i + \sum_{j=1}^{n} a_j k_j \tag{2.33}$$

$$k_j = hf\left(t_i + p_{j-1} h, y_i + \sum_{l=1}^{j-1} q_{j-1,l} k_l\right), \quad j = 1, 2, \ldots, n \tag{2.34}$$

and by definition,

$$p_0 = 0 \; and \sum_{l=1}^{j-1} q_{j-1,l} k_l = 0, \;\; for \; j = 1. \tag{2.35}$$

The determination of the a, p, and q coefficients must be made such that Eq. 2.30 yields successive values of y, and is accomplished by expanding y in a specified order of Taylor series about a point t_i. (Fehlberg 1969)

The inclusion of Fehlberg's work to the traditional Runge-Kutta formulas (presented above) allow for error control of the time step size in the equation, which is

accurate to the truncation error of the Taylor Expansion order of the Runge-Kutta. (Burden and Faires 2010) For example, and RKF4/5 provides a solution to the differential equation which is accurate to 4^{th} order, within a tolerance of the 5^{th} order truncation error control. The ability to control the error of the time step size, and adaptively refine the time step as a function of time makes the Runge-Kutta-Fehlberg Method an ideal candidate for solving the time-dependent form of the Boltzmann transport equation.

The derivation of Fehlberg's formulas to obtain the coefficients of the Runge-Kutta method may be found in the References. (Fehlberg 1969) Solutions for Runge-Kutta-Fehlberg methods of $1^{st}/2^{nd}$, $2^{nd}/3^{rd}$, and $3^{rd}/4^{th}$ can be found in Appendix B, while the $4^{th}/5^{th}$ order method can be found in the upcoming section. Section 2.2.2 address error control of the time-step size when using the RKF Method.

2.2.1 4^{th} Order RKF with 5^{th} Order Truncation Error Control

Given a differential equation of the form of Eq. 2.31, one can apply the Runge-Kutta method with a local truncation error of order five, to obtain

$$\hat{y}_{i+1} = y_i + \frac{16}{135}k_1 + \frac{6656}{12825}k_3 + \frac{28561}{56430}k_4 - \frac{9}{50}k_5 + \frac{2}{55}k_6 \qquad (2.36)$$

which can be used to estimate the local error in the Runge-Kutta method of order four, where the subscript *i* denotes the time step index. The resultant solution at the *i+1* time step, is given by the 4^{th} order extrapolation of derivatives.

$$y_{i+1} = y_i + \frac{25}{216}k_1 + \frac{1408}{2565}k_3 + \frac{2197}{4104}k_4 - \frac{1}{5}k_5 \qquad (2.37)$$

The coefficient terms present in Eqs. 2.36 and 2.37 are given by Eq. 2.38, where *h* is the time step size:

$$
\begin{aligned}
k_1 &= hf(t_i, y_i) \\
k_2 &= hf\left(t_i + \frac{h}{4}, y_i + \frac{1}{4}k_1\right) \\
k_3 &= hf\left(t_i + \frac{3h}{8}, y_i + \frac{3}{32}k_1 + \frac{9}{32}k_2\right) \\
k_4 &= hf\left(t_i + \frac{12h}{13}, y_i + \frac{1932}{2197}k_1 - \frac{7200}{2197}k_2 + \frac{7296}{2197}k_3\right) \\
k_5 &= hf\left(t_i + h, y_i + \frac{439}{216}k_1 - 8k_2 + \frac{3680}{513}k_3 - \frac{845}{4104}k_4\right) \\
k_6 &= hf\left(t_i + \frac{h}{2}, y_i - \frac{8}{27}k_1 + 2k_2 - \frac{3544}{2565}k_3 + \frac{1859}{4104}k_4 - \frac{11}{40}k_5\right)
\end{aligned}
\tag{2.38}
$$

The major advantage of the Fehlberg method is that it only requires six evaluations of the function *f* per step. (Fehlberg 1969) Should one apply a traditional arbitrary Runge-Kutta method of fourth-order and an additional fifth-order evaluation to formulate relations similar to those in Eqs. 2.36 and 2.37, one would have a minimum of ten evaluations of the function *f* per step, four for the fourth-order equation and six for the fifth-order equation. Thus, the Runge-Kutta-Fehlberg method has at least a 40% reduction in the number of function evaluations (when using RKF4/5) over the use of a pair of arbitrary fourth-order and fifth-order methods. (Burden and Faires 2010) The local truncation error of the 4th order method (Eq. 2.37) can now be obtained by subtracting Eq. 2.37 from Eq. 2.36, resulting in a truncation error (TE) in the 4th order method of

$$TE = \frac{1}{360}k_1 - \frac{128}{4275}k_3 - \frac{2197}{75240}k_4 + \frac{1}{50}k_5 + \frac{2}{55}k_6\,. \tag{2.39}$$

2.2.2 Runge-Kutta-Fehlberg Error Control Theory

When applying error control theory, an initial value for the step size *h* at the i^{th} step is used to calculate the first values of $\hat{y}_{i+1}$ and y_{i+1}. Although this procedure leads to additional evaluations of the function *f*, and therefore additional evaluations of all the k_n*'s*, it does allow for the user to control the maximum allowable truncation error, or tolerance on the truncation error, denoted by ε in Eq. 2.41. One must first compute the residual, which is a function of the truncation error as follows:

$$R = \frac{|\hat{y}_{i+1} - y_{i+1}|}{h}. \tag{2.40}$$

If the residual, R, in Eq. 2.40 is greater than the tolerance on the truncation error, then one must evaluate *q*, as it appears in Eq. 2.41 to determine the appropriate change in the step size *h*.

$$q = 0.84\left(\frac{\varepsilon h}{|\hat{y}_{i+1} - y_{i+1}|}\right)^{\frac{1}{4}} \tag{2.41}$$

When *q* is less than 1, reject the initial choice of the time step size *h* and repeat the calculations for the time step using *qh* as the time step. If *q* is greater than or equal to 1, accept the value for the time step and change the step size to *qh* for the *i+1* time step. A pseudo-code for an RKF solver algorithm is provided in Figure 1. Although the application of error control theory in the RKF method can result in an increase in the number of function evaluations, the benefit of knowing the solution is converged within the local truncation error outweighs the additional computational overhead in most cases. Bearing in mind the additional cost of function re-evaluations, *q* tends to be chosen conservatively.

```
RKF_SOLVER (a, b, alpha, tolerance, hmax, hmin)

t = a
y = alpha
h = hmax
FLAG = 1

DO WHILE (FLAG .EQ. 1)
   Solve Equation 2.38 for k1, k2, k3, k4, k5, k6

   Calculate R = (1/h)|(1/360)k1 − (128/4275)k3 − (2197/75240)k4 + (1/50)k5 + (2/55)k6|

      (Note: R = (1/h)|ŷ_{i+1} − y_{i+1}|)

   IF (R .LE. tolerance) THEN
      Accept Approximation

      t = t + h
      Solve Equation 2.37 for y_{i+1}

     OUTPUT(t,y,h)
   ELSE
      Solve Equation 2.41 for q

      Change Step Size using h = constant*q*h
         (Note: Account for low q & high q with if statements)

      IF (h .GT. hmax) h = hmax

      IF (t .GE. b) FLAG = 0          !Reached last time point
      IF (t+h .GT. b) h = t - b
   ENDIF

END DO

OUTPUT(t,y,h)

END RKF_SOLVER
```

Figure 1: RKF solver pseudo-code.

The error controlled RKF can be applied to a coupled system of differential equations, such as a set of equations defining angular fluxes and neutron precursors at the same point in space. The application of the RKF method to coupled equations is identical to if one was solving a single equation, however when a system of equations is solved the

error control is applied until all equations within the set meet the convergence tolerance set on the residual. It is imperative all equations meet the tolerance, as failure of one equation in an interrelated system would cause divergence in the solution of other related equations. Therefore, the choice of step size in a set of coupled equations must be the global minimum step size satisfying the tolerance, resulting in a solution to the set of equations within the specified tolerance criteria.

CHAPTER 3

THEORY

The major focus of this dissertation is a coupled method which solves the discrete ordinates form of the time-dependent Boltzmann transport equation (Eq. 2.13). The Adaptive Runge-Kutta Time-Dependent Transport Code (ARKTRAN-TD) was developed using the methodoloy presented in this chapter. This is accomplished by coupling a standard source iteration, to resolve the spatial profile and thus the spatial derivative of the angular flux, with a RKF solver which takes the time step for both the angular fluxes as well as the delayed neutron precursor concentrations. The starting point for any time-dependent reactor transient will be a converged steady-state solution, which is then scaled to power before launching into the RKF solver to take the initial time step, after which the source iteration is performed, resulting in a converged flux profile at a time of $t+\Delta t$. This quasi-static iteration between the time sweep and space sweep is repeated until the total time is greater than the time cutoff provided by the user. The first major section in this chapter addresses the application of RKF to the transport equation and its implementation in the computer code, while the second major section addresses the application of the source iteration to resolve the spatial flux profile.

3.1 Application of RKF Method to the Time Domain of the S_N Transport Equation

By using the adaptive Runge–Kutta–Fehlberg method for the time domain, one can estimate the time step error using a difference of higher order Runge-Kutta methods. By determining the maximum acceptable time step size for each spatial location in the

global solution space, then choosing the minimum of these to be the global Δt for the time step, one can ensure the stability of the explicit solution form of the transport equation and neutron precursor equations over the time domain; as the time step size is now adaptive and specific to the physics of the problem of interest. The RKF method then uses an extrapolation of derivatives of both the angular flux and neutron precursors to determine the angular flux and precursor concentrations at a time of $t+\Delta t$. When looking at the general time dependent form of the transport equation, one realizes that the equation can be cast in the form:

$$\frac{d\Psi}{dt} = f(t,\Psi), \qquad a \leq t \leq b, \qquad \Psi(a) = \propto. \tag{3.1}$$

Where the time derivative of the angular flux is a function of the angular flux and time. Thus, one can apply the Runge-Kutta Method to resolve the solution to the initial value problem in Eq. 3.1, on the time domain from a to b. The initial condition $\Psi(a)$, is given by the steady-state solution to the transport equation, resolved with a standard coupled Source Iteration and Power Iteration Method. Initial delayed neutron precursors are assumed to be in equilibrium at steady-state, thus their initial condition is defined at $t=a$ as well.

For the sake of discussion of the application methods in this chapter, it is assumed the code will utilize a Runge-Kutta-Fehlberg method of 4^{th} order, with a 5^{th} order local truncation error. Applying the equations presented in Section 2.1.4 to the generalized transport functional, presented in Eq. 3.1, one arrives at the 5^{th} order local truncation given by the following, where i refers to the time step and $i+1$ denotes the next time step.

$$\hat{\psi}_{i+1} = \Psi_i + \frac{16}{135}k_1 + \frac{6656}{12825}k_3 + \frac{28561}{56430}k_4 - \frac{9}{50}k_5 + \frac{2}{55}k_6 \qquad (3.2)$$

This truncation error is used to determine the appropriate time step size to evaluate the angular flux and neutron precursors (for neutron precursors, the Ψ term in all equations in this section simply become C_l where l=*1-6*). Resulting in the angular flux at time $i+1$, given by a 4^{th} order Taylor series expansion of the transport equation and the neutron precursor concentrations.

$$\Psi_{i+1} = \Psi_i + \frac{25}{216}k_1 + \frac{1408}{2565}k_3 + \frac{2197}{4104}k_4 - \frac{1}{5}k_5 \qquad (3.3)$$

The coefficient terms in Eq. 3.2 and 3.3 are the same as those presented in Eq. 2.38 for the generalized form of the RKF4/5, however they are included below as a function of Ψ for completeness.

$$\begin{aligned}
k_1 &= hf(t_i, \Psi_i) \\
k_2 &= hf\left(t_i + \frac{h}{4}, \Psi_i + \frac{1}{4}k_1\right) \\
k_3 &= hf\left(t_i + \frac{3h}{8}, \Psi_i + \frac{3}{32}k_1 + \frac{9}{32}k_2\right) \\
k_4 &= hf\left(t_i + \frac{12h}{13}, \Psi_i + \frac{1932}{2197}k_1 - \frac{7200}{2197}k_2 + \frac{7296}{2197}k_3\right) \\
k_5 &= hf\left(t_i + h, \Psi_i + \frac{439}{216}k_1 - 8k_2 + \frac{3680}{513}k_3 - \frac{845}{4104}k_4\right) \\
k_6 &= hf\left(t_i + \frac{h}{2}, \Psi_i - \frac{8}{27}k_1 + 2k_2 - \frac{3544}{2565}k_3 + \frac{1859}{4104}k_4 - \frac{11}{40}k_5\right)
\end{aligned} \qquad (3.4)$$

The truncation error can be calculated with Eq. 2.38 and error control is applied based on the general method discussed in Section 2.2.2 to determine the appropriate time step size before taking the time step.

After determining the scaled-to-power angular flux profile, based on the steady-state solution, the code will launch into the RKF scheme, using the RKF4/5 method discussed in Eqs 3.2-3.4. Solving the time-dependent form of the discrete ordinate transport equation (Eqn. 2.13) in the following form:

$$\begin{aligned}\frac{\partial}{\partial t}\psi_{n,g}(z,t) = \nu_g \Bigg[& -\mu_n \frac{\partial}{\partial z}\psi_{n,g}(z,t) - \sigma_g(z,t)\psi_{n,g}(z,t) \\ & + \sum_{l=0}^{L}(2l+1)P_l(\mu_n)\sum_{g'=1}^{G}\sigma_{l,g'\to g}(z,t)\phi_{l,g'}(z,t) \\ & + \chi_{P,g}(z)[1-\beta]\sum_{g'=1}^{G}\nu\sigma_{f,g'}(z,t)\phi_{g'}(z,t) \\ & + \sum_{l=1}^{ndg}\chi_{d,g}^{l}(z)\lambda_l C_l(z,t) + q_{n,g}^{ext}(z,t)\Bigg]\end{aligned} \tag{3.5}$$

where neutron precursor concentrations are given by:

$$\frac{\partial}{\partial t}C_l(z,t) = -\lambda_l C_l(z,t) + \beta_l \sum_{g'=1}^{G}\nu\sigma_{f,g'}(z,t)\phi_{g'}(z,t)\,. \tag{3.6}$$

The values for the neutron precursor concentrations at *t = 0* can be found by setting the time derivative in Eqn. 3.6 to zero and solving for each of the C_l from the steady-state flux profile; as this is representative of the assumption that the neutron precursor concentrations are in equilibrium at steady-state. The scalar flux and scalar flux moments of Eqs. 3.5 and 3.6 are given by the following, where w_n is the quadrature weight associated with the angle μ_n for each angular flux ordinate (see Appendix A for values of μ_n and w_n for different S_N orders):

$$\phi_g(z,t) = \frac{1}{2}\sum_{n=1}^{N} w_n \psi_{n,g}(z,t) \tag{3.7}$$

$$\phi_{l,g}(z,t) = \frac{1}{2}\sum_{n=1}^{N} w_n P_l(\mu_n)\,\psi_{n,g}(z,t) \tag{3.8}$$

3.1.1 Assumptions in Applying RKF Method to the Transport Equation

Taking a closer look at the time-dependent transport equation (Eq. 3.5), it quickly becomes apparent, with the presence of the space derivative term $\frac{\partial}{\partial z}\psi_{n,g}(z,t)$, that the entire phase space is coupled together. This would result in a number of equations equal to the following:

$$\begin{aligned} \#Eqs. = (\#Spatial\ Points)[\#Delayed\ Groups \\ + (\#Energy\ Groups \cdot S_N\ Order)]\,. \end{aligned} \tag{3.9}$$

By performing the quasi-static time/space sweep, one effectively decouples the space/time dependence in the RKF time step sweep by assuming that over the time step the space derivative term is constant. Thereby reducing the number of coupled differential equations to a number of coupled differential equations at each cell center spatial location equivalent to:

$$\begin{aligned} \#Eqs.\ per\ Cell = \#Delayed\ Groups \\ +(\#Energy\ Groups \cdot S_N\ Order). \end{aligned} \tag{3.10}$$

For example, a S_4 calculation, with 2 energy groups, and 6 delayed neutron precursor groups would result in 14 coupled equations at each point in space. One would then march through each space point in the phase space to determine the maximum value of Δt for each ordinate, at every point in space, for all energy groups. The global minimum of these maximum local Δt values is then used as the time step size. The values of k_1 through k_6 in Eq. 3.4 are then recalculated at each point in space, for all coupled

equations present and the time step is taken at the point to get the updated values for neutron precursor concentrations, as well as angular flux. This space derivative is recalculated during the space sweep of each time step, and is therefore the most recent representation of the space derivative in the equation.

The total number of coupled equations evaluated globally remains constant when this decoupling is performed. However, the computation time is reduced significantly, as if a time step size does not meet the error control criterial, one is only re-evaluating the coupled functionals at a single point in space, rather than in the entire phase space. When the maximum value of Δt has been determined, it is checked against the global minimum value of Δt. If the global minimum is greater than the local value, the local value becomes the new global minimum. When proceeding to the next spatial point, the most current local minimum Δt, is used as the starting point to see if the time step meets the residual convergence criteria, given by Eq. 2.40. So all in all, the assumption that the space derivative of the angular flux is constant over the time step allows for a much faster computational time; even though by doing this, one is now required to perform a spatial source iteration to resolve the space profile of the angular flux. It is far more computationally intensive to re-evaluate the transport and precursor functionals in the entire phase space once, than it is to perform a source iteration. In most cases, the functionals are re-evaluated at least 10 times before the residual tolerance is within the convergence criteria.

3.1.2 Subtleties to Consider when Applying RKF to the Transport Equation

Observing the time-dependent transport equation (Eq. 3.5) and the neutron precursor concentration equation (Eq. 3.6), one realizes that the scalar flux and/or scalar flux moments are used when resolving the contribution from scatter, fission, and delayed neutrons. Looking at Eqs. 3.7 and 3.8, it becomes apparent that the scalar flux and scalar flux moments are reconstituted from the angular fluxes, as a function of time. Therefore, any term in Eq. 3.5 which includes a scalar value needs to have the updated scalar value at the appropriate time designated in the functionals that represent the expansion of the angular flux (e.g. the k_n values of Eq. 3.4) and using the expanded values of the angular flux to represent the scalar quantities. This is accounted for in the computer code by solving the equations in terms of the angular fluxes only (substituting Eqs 3.7 and 3.8 into Eqs 3.5 and 3.6, where appropriate), as shown below:

$$\begin{aligned}
&\frac{\partial}{\partial t}\psi_{n,g}(z,t) \\
&= \nu_g \Bigg[-\mu_n \frac{\partial}{\partial z}\psi_{n,g}(z,t) - \sigma_g(z,t)\psi_{n,g}(z,t) \\
&+ \sum_{l=0}^{L}(2l+1)P_l(\mu_n)\sum_{g'=1}^{G}\sigma_{l,g'\to g}(z,t)\left(\frac{1}{2}\sum_{n=1}^{N}w_n P_l(\mu_n)\psi_{n,g'}(z,t)\right) \\
&+ \chi_{P,g}(z)[1-\beta]\sum_{g'=1}^{G}\nu\sigma_{f,g'}(z,t)\left(\frac{1}{2}\sum_{n=1}^{N}w_n\psi_{n,g'}(z,t)\right) \\
&+ \sum_{l=1}^{ndg}\chi_{d,g}^{l}(z)\lambda_l C_l(z,t) + q_{n,g}^{ext}(z,t)\Bigg]
\end{aligned} \tag{3.11}$$

and the neutron precursor concentrations are given by,

$$\frac{\partial}{\partial t}C_l(z,t) = -\lambda_l C_l(z,t) + \beta_l \sum_{g'=1}^{G}\nu\sigma_{f,g'}(z,t)\left(\frac{1}{2}\sum_{n=1}^{N}w_n\psi_{n,g'}(z,t)\right). \tag{3.12}$$

As discussed previously, at $t = 0$, the angular flux is considered to be at steady-state and scaled to power before entering into the time sweep and the initial neutron precursor concentrations are solved by setting the time derivative in the precursor equation equal to zero and solving for $C_l(z, 0)$. After the initial time step, the Ψ_i that appears in the RKF expansion and function evaluations (Eqs. 3.2-3.4), is set to equal to the converged angular flux values coming out of the source iteration sweep. While the neutron precursor concentrations $C_{l,i}$ are set to be equal to the neutron precursor concentration from the previous time step, it is then expanded and calculated using the RKF method applied to Eq. 3.12.

Regarding data storage, all memory that is not represented by character, integer, or logical variable is stored as a double precision variables in FORTRAN (equivalent to double in C++). Quadrature weights and angles were hard coded into the program with 15 significant digits of accuracy. Lastly, all file inputs and outputs or messages printed to the screen show time in seconds or values printed with units of seconds or per second. Inside the algorithm, the unit of time used is the *"shake."* One *shake* is the equivalent to 10^{-8} seconds (10 nanoseconds). The unit has a history in nuclear physics and is originally attributed to the Manhattan Project where, given the nature of the project, it was convenient to have a timescale that was not understood outside of the group. This conversion was performed internally to clean up the numerics as the calculation of time step residuals (Eq. 2.40) as well as the k_n values, requires one to either divide or multiply by the time step size. Given the range of *normal* time steps size of

10^{-18} to 10^{-6} seconds for most calculations (though the average value is typically between 10^{-9} and 10^{-7} seconds), having a unit of time which increases the magnitude of the time domain internally helps to significantly reduce numerical discrepancies and ill-conditions which may occur due to multiplication and division by values which are approaching zero.

3.2 Spatial Transport Sweep Coupled to the RKF Time Sweep

After completing the optimized time step for the angular fluxes and neutron precursor concentrations, a source iteration must be performed in order to resolve the effects of radiation transport across the fine cell boundaries. To do this, a standard source iteration is performed, where the time-dependent form of the discrete ordinates transport equation (Eq. 2.13) is recast as follows, by shifting the time derivative term from the left-hand side of the equation to the right-hand side of the equation.

$$\begin{aligned}\mu_n \frac{\partial}{\partial z}\psi_{n,g}(z,t) + \sigma_g(z,t)\,\psi_{n,g}(z,t) &= \sum_{l=0}^{L}(2l+1)P_l(\mu_n)\sum_{g'=1}^{G}\sigma_{l,g'\to g}(z,t)\phi_{l,g'}(z,t) \\ &+ \chi_{P,g}(z)[1-\beta]\sum_{g'=1}^{G}\nu\sigma_{f,g'}(z,t)\phi_{g'}(z,t) \\ &+ \sum_{l=1}^{ndg}\chi_{d,g}^{l}(z)\lambda_l C_l(z,t) + q_{n,g}^{ext}(z,t) \\ &- \frac{1}{\nu_g}\frac{\partial}{\partial t}\psi_{n,g}(z,t)\end{aligned} \tag{3.13}$$

Substituting the collective source terms (right-hand side of Eq. 3.13) for the variable $Q_{n,g}(z,t)$; representing the scatter, prompt fission, delayed fission, external sources, and time rate of change of neutrons, one arrives at a more simplified form of the transport equation.

$$\mu_n \frac{\partial}{\partial z} \psi_{n,g}(z,t) + \sigma_g(z,t)\psi_{n,g}(z,t) = Q_{n,g}(z,t) \quad (3.14)$$

The Diamond Difference Scheme can then be applied to Eq. 3.14, using appropriate boundary conditions, to perform a Source Iteration that converges the angular flux at the time step. The $Q_{n,g}(z,t)$ in Eq. 3.14 is the $Q_{i,n,g}(t)$, which appears in the equation for the DD scheme (Eq. 2.19). The reader can refer to Section 2.1.4 to familiarize themselves with Differencing Schemes, Boundary Conditions, and Iterative Methods. After converging the angular fluxes in the spatial sweep, the resultant angular fluxes are used as the starting point for the angular fluxes in the next RKF time step.

3.2.1 Assumptions in the Spatial Sweep

While performing the Source Iteration, the value of the space derivative is stored for use in the next RKF time step (where it is assumed to be constant for the next time step). The space derivative at the fine mesh center is assumed to be equivalent to the difference in fine mesh edge fluxes divided by the mesh size along the flight path of the radiation. For example, for radiation traveling in a direction where $\mu_n > 0$,

$$\frac{\partial \psi_{n,g}(z,t)}{\partial z} = \frac{\psi_{i+\frac{1}{2},n,g}(t) - \psi_{i-\frac{1}{2},n,g}(t)}{\Delta z} \quad (3.15)$$

where *i* is the fine mesh center, and *i*±½ are the left and right edge fluxes for the mesh. Furthermore, looking at the spatial sweep form of the transport equation (Eq. 3.13) the time derivative of the angular flux in mesh *i*, along ordinate *n*, for energy group *g* is assumed to be constant during the space sweep and is represented by taking the difference in angular fluxes from the current and previous time steps and dividing them by current time step size, as shown below.

$$\frac{\partial \psi_{n,g}(z,t)}{\partial t} = \frac{\psi_{n,g}(z,t) - \psi_{n,g}(z,t-\Delta t)}{\Delta t} \tag{3.15}$$

Lastly, the neutron precursor concentrations are assumed to be constant during the spatial sweep as well. The time derivative and precursor concentrations must be constant in the transport sweep resolving the spatial behavior of the angular flux. By decoupling the space and time domains in the transport equation, and performing the quasi-static iteration between the space and time domains, all time dependent terms must be held constant in the space sweep, and all space dependent terms must be held constant in the time sweep. To do otherwise would result in an over-definition of the precursor concentrations, time rate of change of the angular flux, and/or space derivative of the angular flux; giving rise to non-physical behavior of the neutron flux and all associated quantities (precursors, fission source, scatter source, system power, *et cetera*) as a function of time.

3.2.2 Addressing the Lack of Power Iteration

When considering the general approach to solving S_N transport problems with fission sources, a power iteration is performed as the outer iteration which sets the fission source term. Thereafter, an inner source iteration is performed to converge the scatter source term for that fission source term. This process is repeated until both the fission source term and the angular flux (and scatter source) are converged. However, in the realm of time-dependent transport, at the current point in time the fission source term and delayed neutron precursor concentrations are known (coming out of the RKF time step). Thus, the Power Iteration is not needed to converge on the fission source or precursor concentrations, and only a Source Iteration is required to converge the scatter source and

account for the spatial migration of radiation between the fine mesh boundaries. Should one include a Power Iteration as part of the quasi-static space/time solution method, one is effectively forcing the fission and precursor contributions to assume steady-state values, which is not representative of a time-dependent solution. The resultant angular flux would immediately transition to a flux profile which would be consistent with the "settled out" solution after a transient. Pictorially, one would observe the prompt jump or drop from the transient occurring immediately and the flux profile of the solution would simply increase or decrease in magnitude thereafter.

3.3 Concluding Remarks on Quasi-Static RKF Time/Space Transport Sweep

Overall, the main benefit of the quasi-static RKF time/space transport sweep is the reduction in the number of RKF expansion functionals which are evaluated at every time step (compared to a purely RKF based time-dependent transport code) while obtaining the appropriate Δt. Furthermore, using the RKF with the adaptive time step size yields a solution guaranteed to be within the truncation error tolerance, and is a huge improvement over the guess work involved in employing a user-specified time step, which can yield incorrect solutions if the time step size it chosen to be too large. Figure 2 provides a brief pseudo-code representation of the layout of the quasi-static space/time sweep, as implemented in the code developed.

```
Perform steady-state calculation
Scale flux to desired power level

t=0

Calculate neutron precursor concentrations by setting derivative in
equation 3.6 to zero

DO WHILE (t .LE. tmax)

  Solve Eqn. 3.5 & 6 coupled precursor equations using RKF find the
  required time step at each spatial location, for each energy group,
  and every ordinate

  Determine the minimum global time step

  Calculate k1 through k6 (Eq. 3.4) using min global time step and
  solve for angular flux and precursor concentrations at t+dt, for
  each ordinate, group, and location using Eq. 3.3

  Calculate the new fission source term using the updated angular
  flux values

  Perform the Source Iteration spatial sweep to resolve the space
  behavior of the angular flux and the space derivative (Eqs. 3.14 &
  3.15 with Diamond Difference)

  t=t+dt

ENDDO

Print Results
```

Figure 2: Pseudo-code of the RKF Time/Space Transport Sweep.

CHAPTER 4

VERIFICATION OF THE STEADY STATE TRANSPORT SOLVER

As discussed in Chapter 3: Theory, the starting basis for a reactor transient calculation is a converged steady-state solution, scaled to the system power. Therefore, it is imperative the steady-state transport solver incorporated into ARKTRAN-TD be validated against an additional transport code, using several benchmark problems of interest. To this point, this dissertation has solely been focused on the time-dependent form of the transport equation, therefore the first major section of this chapter addresses the steady-state equation form and solution method. The subsequent sections provide code to code comparisons with several problems of interest between the ARKTRAN-TD solution and the PENTRAN solution. PENTRAN is the Parallel Environment Neutral Particle Transport Code by Sjoden and Haghighat, which uses the S_N method to solve 3-D forward and adjoint transport calculations for fixed source and eigenvalue problems at steady-state. (Sjoden and Haghighat 2008)

4.1 Brief Overview of Steady-State Transport Solution Methods

The steady-state form of the multigroup discrete ordinates transport equation is given by Eq. 4.1. (Lewis and Miller 1993) The energy, angle, and space discretization from the integral form of the transport equation is identical to the methods discussed in Chapter 2. However, with the time domain removed, there is no time derivative, delayed neutron concentrations, or beta in the fission source term. The delayed neutrons are

considered to be in equilibrium, thus all fission neutrons are incorporated into the fission term observed in Eq. 4.1.

$$\mu \cdot \frac{\partial}{\partial z}\psi_g(z,\mu) + \sigma_t(z)\psi_g(z,\mu) = \sum_{g'=1}^{G}\sum_{l=0}^{L}(2l+1)\sigma_{sl,g'\to g}(z)P_l(\mu)\phi_{l,g'}(z) + \frac{\chi_g(z)}{k}\sum_{g'=1}^{G}\nu\sigma_{f,g'}(z)\phi_{0,g'}(z) + q_g^{ext}(z,\mu) \quad (4.1)$$

The *k* present in Eq. 4.1 represents the eigenvalue of the system. An eigenvalue equal to 1 implies the reactor system is critical, an eigenvalue less than 1 results in a subcritical system, and an eigenvalue greater than 1 represents a supercritical system. The right hand side of Eq. 4.1 equals the total source from scatter, fission, and external sources, respectively. One can solve for both the eigenvalue and angular flux using a combination of the Power Iteration and Source Iteration methods coupled with the boundary conditions discussed in Chapter 2. The Diamond Difference scheme was employed when calculating the angular fluxes in the transport sweep. For purely fixed source calculations that do not have a multiplying (e.g. fissionable) medium, only the Source Iteration is used, as the Power Iteration converges the fission source and *k* eigenvalue that would not be present in the system. Figure 3 depicts a pseudo-code of the steady-state solver as coded in ARKTRAN-TD.

```
Perform steady-state calculation
Scale flux to desired power level

Guess initial k, ϕ and Ψ

DO WHILE (k_tol .lt. cutoff)
  Calculate fission source

  DO WHILE (flux_tol .lt. flux_cutoff)

    Perform DD Transport Sweep using Eq. 2.19 and 2.20 in the
    positive and negative directions of μ

    Determine flux_tol using Eq. 2.27

    Set the new values of Ψ to the old value of Ψ

  ENDDO

  Calculate new k using Eq. 2.28
  Determine k_tol using Eq. 2.29

  Set the new values of ϕ to the old value of ϕ

ENDDO

Print Results
```

Figure 3: Pseudo-code depicting the steady-state solver in ARKTRAN-TD.

The inner loop in Figure 3 represents the Source Iteration process that converges the scatter source and angular flux profiles, while the outer loop contains the Power Iteration which determines the eigenvalue and fission source term. In addition to the convergence criteria from Eqs. 2.27 and 2.29, for angular flux and eigenvalue, respectively, one can also specify a maximum number of inner and outer iterations which will also cause the code to proceed to the next iteration. This can help move the solution towards convergence faster, especially if the fission source term is not well defined by the initial guess.

4.2 Alcouffe Slab Problem

The first model used to validate the steady-state transport solver in ARKTRAN-TD, is a test problem originally presented in a paper by Alcouffe, *et. al.* containing a mixture of both streaming and diffuse mediums. (Alcouffe, et al. 1979) This problem is purely a fixed source calculation, without any fissionable material. While the time-dependent code is designed to solve reactor transients, it is important that the fixed source representation in the transport solver is designed correctly, should one need to include a combination of fixed sources and fission sources in a transient model.

4.2.1 Detailed Problem Geometry

The Alcouffe slab problem consists of 4 coarse meshes (or slabs), each containing different materials and numbered sequentially from left to right. Figure 4 depicts the problem geometry. The first zone consists of a half-scattering region, the second zone has a unit density source in a pure absorber, the third zone consists of a typical shielding material, and the fourth zone is a diffuse region with a unit density source. The coarse slab boundaries span the z-axis and are located at 0.0, 3.0, 6.0, 36.0, and 48.0 cm. The number of fine meshes in each coarse slab was specified to be 60, 60, 600, and 240 for slabs 1, 2, 3, and 4, respectively.

Vacuum boundary conditions are applied to the left and right hand z-boundaries. As PENTRAN is a 3-D transport code, which was used as a basis of comparison, reflective boundary conditions were employed in the x- and y-directions to effectively model the 1-D slab geometry. The convergence criteria on the angular flux was set to 10^{-5}

and evaluated using Eq. 2.27. Quadrature specified for evaluation of this problem as S_8, which results in 8 angles in ARKTRAN-TD and 80 angles in PENTRAN (as PENTRAN is using a 3-D quadrature set). One energy group, P_0 cross sections were specified in the problem documentation, and provided in Table 1 for each slab material present.

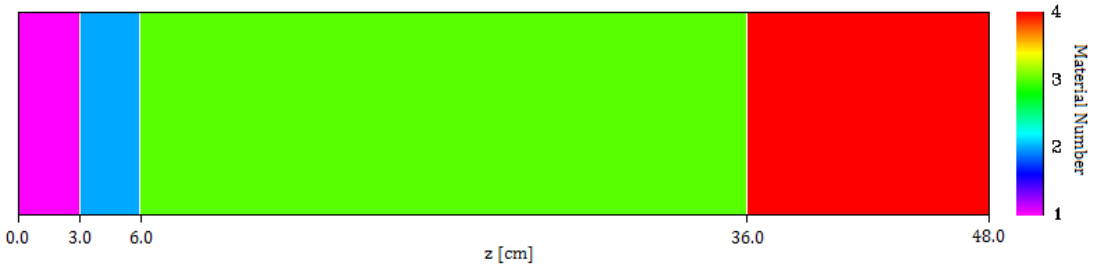

Figure 4: Alcouffe slab test problem geometry.

Table 1: Alcouffe slab test problem cross sections.

Material	σ_a	$\nu\sigma_f$	σ_t	σ_s
1. Half Scatterer	0.50	0.00	1.00	0.50
2. Pure Absorber + Source (1 n/cm^{3}s)	1.00	0.00	1.00	$1\text{x}10^{-6}$
3. Shield	0.95	0.00	1.00	0.05
4. Diffuse Region + Source (1 n/cm^{3}s)	0.05	0.00	1.00	0.95

4.2.2 Results of Alcouffe Test Problem

The resulting scalar flux profiles for the converged solution from PENTRAN and ARKTRAN-TD are shown in Figure 5. The yellow line represents the scalar flux profile obtained with ARKTRAN-TD, while the dotted black line depicts the solution from PENTRAN. When viewing the scalar flux plot, excellent agreement is seen between the

two transport codes, as the lines sit directly on top of each other. An average relative error in scalar flux values of 0.25% was computed according to Eq. 4.2.

$$Relative\ Error = \left| \frac{\phi_{ARKTRAN-TD} - \phi_{PENTRAN}}{\phi_{PENTRAN}} \right| \tag{4.2}$$

The low relative error between the codes, coupled with the overlapping scalar flux profiles, verifies the capability of ARKTRAN-TD to model fixed source transport simulations as well as its ability to resolve the spatial profile of angular (and scalar) fluxes in geometries containing purely absorbing, shielding, and highly scattering materials.

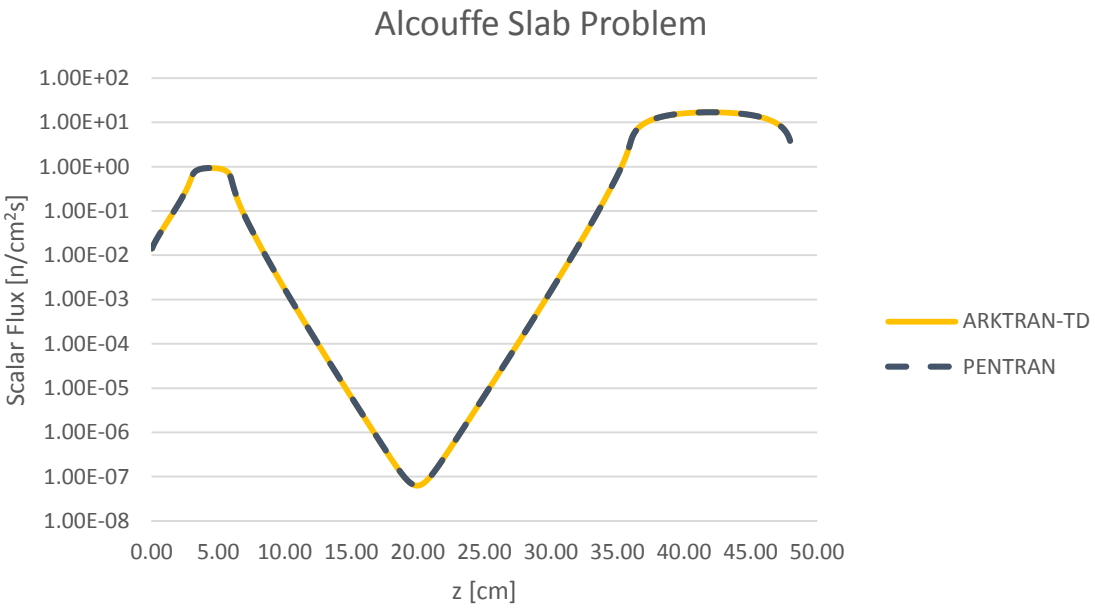

Figure 5: Alcouffe test problem flux profiles from PENTRAN and ARKTRAN-TD.

4.3 Boiling Water Reactor (BWR) Test Problems

In order to verify ARKTRAN-TD's ability to solve multigroup eigenvalue problems representative of reactor calculations, a set of four benchmark problems originally developed by Douglas were chosen for their versatility using a set of thirteen materials to model four different GE9 based BWR fuel bundles. (Douglass 2012) The next section addresses the problem geometries for the fuel, while Section 4.3.2 discusses the results obtained with ARKTRAN-TD and compares these to the PENTRAN calculated solutions.

4.3.1 BWR Bundle Geometries and Core Configurations

The BWR benchmark problems are composed of a set of different pin cells, representing twelve fuel types (10 enrichments and 2 gadded pins) which were used to generate 47 group cross sections, collapsed via a flux weighting and originally generated with the HELIOS code. (Simenov 2003) Furthermore, the cross section definition utilizes upscatter cross sections and P_1 anisotropic scattering moments. Each fuel bundle or assembly is composed of a total of 10 material regions. The central 8 regions contain fuel pins while the outer region on each side contains a moderator material. The fuel regions are 1.6256 cm in width each, and the outer moderator regions are slightly smaller at 1.1176 cm in with. Thus, the total length of a single fuel bundle is 15.24 cm, which is consistent with BWR bundle size.

Table 2 contains the material map for each of the 4 bundle types modeled, were M refers to the moderator material, the numbers reflect the fuel type used in that pin cell.

Fuel types 1 through 10 contain various enrichments, while fuel types 11 and 12 represent the gadded pins. Each fuel bundle was modeled using a single coarse slab for each pin or moderator boundary (e.g. a total of 10 coarse slabs), while moderator regions contained 24 fine meshes each and fuel regions consisted of 12 fine meshes each, for a total of 144 fine meshes per bundle. All fuel bundle simulations were performed with S_8 quadrature, P_1 scattering moments, and reflective boundary conditions on the positive and negative z-axis boundaries. Convergence criteria were set at 10^{-3} for the angular flux and 10^{-5} for the eigenvalue, which represent standard values for these tolerances. The Diamond Differencing scheme was used by both transport codes to resolve the angular flux.

Table 2: BWR benchmark fuel bundle layouts by material type.

Bundle Type	Material Layout
1	M 03 03 08 08 08 08 03 03 M
2	M 03 03 03 03 03 03 03 03 M
3	M 03 03 11 11 11 11 03 03 M
4	M 11 11 11 11 11 11 11 11 M

4.3.2 Fuel Bundle Benchmark Results

The resulting eigenvalues from ARKTRAN-TD and PENTRAN are shown in Table 3, and display agreement between the two solution methods. The difference in the two eigenvalues were calculated and provided in pcm (1 pcm or per cent mil is equivalent to 10^{-5}), using Eq. 4.3.

$$\Delta k\ [pcm] = |k_{ARKTRAN-TD} - k_{PENTRAN}| \cdot 10^5 \qquad (4.3)$$

The largest difference in the calculated eigenvalues was 1.70 pcm in Bundle 2, while Bundle 4 displayed the smallest delta *k* of 1.13 pcm. It is important to note that while these differences are relatively small, some difference is expected given PENTRAN's use of a 3-D quadrature in the 1-D model, while ARKTRAN-TD is using a 1-D quadrature only. Furthermore, comparing computer run time for these models would not be a fair metric, as the 3-D quadrature set is represented by a factor of 10 more angles, thus increasing the number of equations PENTRAN solves per point. Thus, computer runtime will not be compared for these benchmarks.

Table 3: Eigenvalues of BWR fuel bundles

Bundle	ARKTRAN-TD	PENTRAN	Δk [pcm]
1	1.334821	1.334936	1.15
2	1.278310	1.278480	1.70
3	0.694472	0.694590	1.18
4	0.316440	0.316327	1.13

Scalar flux profiles obtained from each bundle type were compared between the PENTRAN solution (dotted black line in plots) and the ARKTRAN-TD solution (solid gold line in plots) in order to verify ARKTRAN-TD's ability to not only resolve the system eigenvalue, but also determine the correct spatial profile of the flux. Two energy groups were chosen to display the flux profiles from both codes, the first Group 5 depicts the flux between energies of 0.82 MeV and 1.35 MeV. The Group 39 fluxes were also plotted, to show the behavior in a more thermal region between 0.018 eV and 0.027 eV. Flux profiles were normalized to the maximum value of the scalar flux (for each code) in

Group 5. This allows for a fair comparison between the flux profiles' shape, as the fluxes were not scaled to a particular power during the simulation.

The scalar flux profiles for Bundle 1 are shown in Figure 6. Observing the fast flux (Group 5) one notices an increase when following the profile through the fuel region. The sight increase between 4 cm and 6 cm depicts the transition from one fuel type to another and suggests that the central fuel pins are more highly enriched than the outer pins. The thermal flux (Group 39) profile verifies this, as a more pronounced absorption is seen in the central pins compared with the edge pins. The average relative error, from Eq. 4.2, was calculated to be 0.20% for Group 5 and 0.08% for Group 39. Maximum relative errors were observed to be 0.55% in Group 5 and 0.15% in Group 39.

Flux profile behavior for Bundle 2 is shown in Figure 7. Compared with the profile for Bundle 1, one observes a much more pronounces plateau region in the Group 5 flux, which is attributed to the inner fuel regions all having the same enrichment. Based on this profile, one would expect the central fuel pins to have a lower enrichment than the outer fuel pins. Again we see a depression in the fast flux going into the moderator regions, as the flux is scattering into more thermal energy groups. The profile of the Group 39 flux, again displays almost an inverse behavior compared with the Group 5 flux. Average relative error in the scalar flux, was calculated to be 0.21% and 0.08%, for the Group 5 and Group 39 fluxes, respectively. While the maximum relative errors were 0.60% in Group 5 and 0.13% in Group 39.

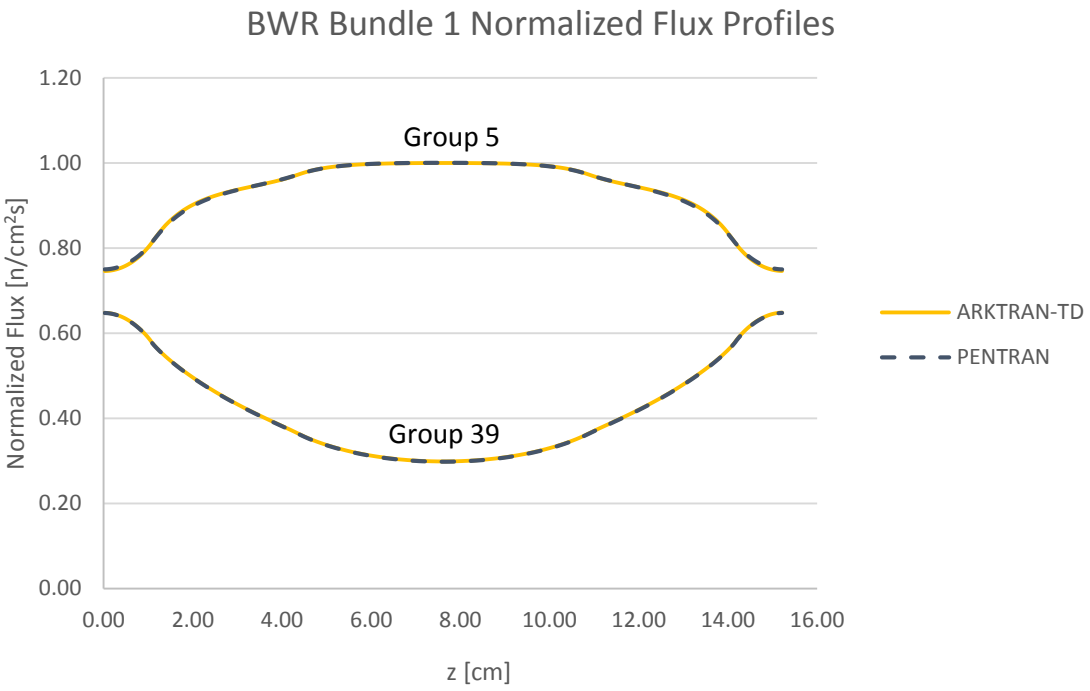

Figure 6: BWR benchmark Bundle 1 scalar flux profiles for select groups.

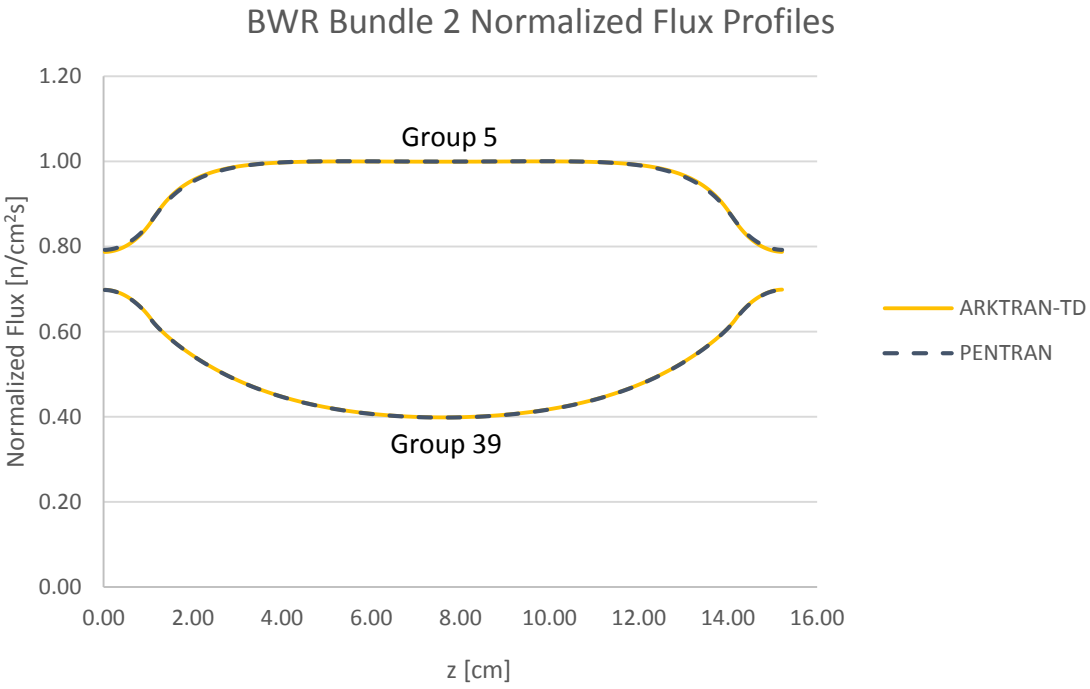

Figure 7: BWR benchmark Bundle 2 scalar flux profiles for select groups.

Bundle 3 provides the flux profiles which exhibit the most variant behavior as a function of position across the bundle. Observing the behavior of the flux, shown in Figure 8, one notes a rise in the fast flux through the traditional fuel pin region before noticing a decrease in the fast flux as neutrons enter the gadded fuel pins. The thermal flux is the highest at the problem boundaries (due to the moderator region) and is absorbed as neutrons travel further and further into the fuel. The gadded pins have a higher absorption cross section for the thermal neutrons, which results in the thermal neutron well in the center of the fuel bundle. Given the more oscillatory shape of the flux profiles in the Bundle 3 model, mean relative errors were slightly higher than the previous two bundle cases; 0.58% in Group 5 and 0.38% in Group 39. The maximum relative error in the scalar flux for Bundle 3 was 1.25% and 0.62%, for Group 5 and Group 39, respectively.

The last of the BWR bundle benchmark cases consisted of the assembly made up entirely of gadded pins. The flux profiles for Group 5 and Group 39 are shown in Figure 9 and depict the anticipated behavior through the moderator region and into the gadded fuel pin region. As the flux profiles are well behaved, the average relative error between the PENTRAN and ARKTRAN-TD solutions are more in line with those of the Bundle 1 case; 0.20% for Group 5 and 0.11% for Group 39. While the maximum relative error in the fast group was 0.58% at the boundary of the problem and 0.41% in the thermal group at the interface between the moderator region and the first gadded fuel pin.

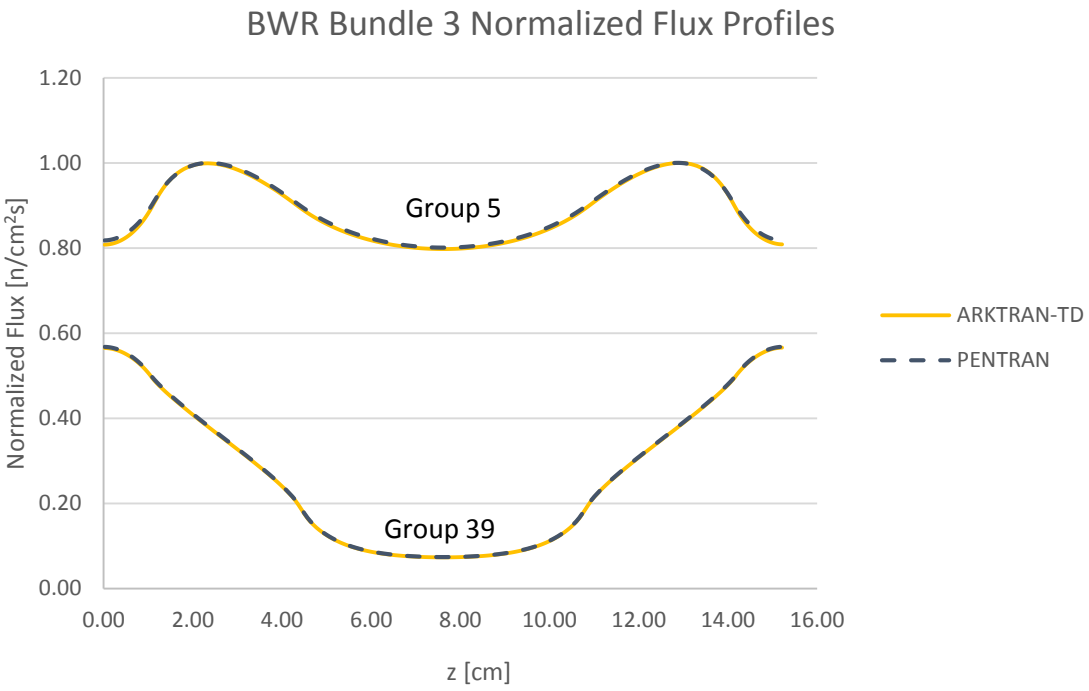

Figure 8: BWR benchmark Bundle 3 scalar flux profiles for select groups.

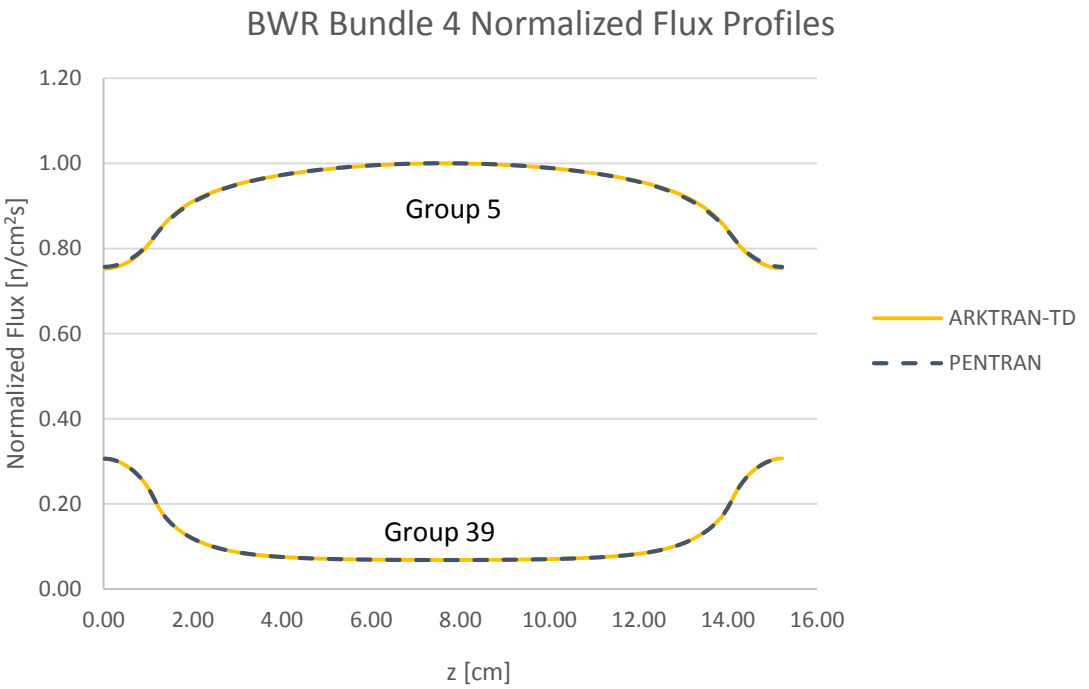

Figure 9: BWR benchmark Bundle 4 scalar flux profiles for select groups.

4.3.3 Brief Comment on Cross Section Format Conversion

Since both PENTRAN and ARKTRAN-TD operate using the standard LANL cross section format, these cross sections had to be converted into the correct format. Some discrepancies in the conversation of the transport cross section (output from HELIOS) and the reconstituted total cross section (used in PENTRAN and ARKTRAN-TD) were observed. While results between PENTRAN and ARKTRAN-TD depict excellent comparison, some of the eigenvalues are slightly different than the values reported by Douglass. This is due purely to discrepancies in the cross sections causing minor differences in the eigenvalue. All flux profiles exhibit the same behavior as the benchmark document.

CHAPTER 5

VALIDATION OF THE TIME-DEPENDENT TRANSPORT SOLVER

The previous chapter demonstrated the steady-state starting basis for time-dependent calculations has been verified to be working correctly, from solving the benchmark problems. Therefore, this chapter goes on to validate that the quasi-static coupled RKF method with source iterations to solve time-dependent reactor transient problems. Results will be presented from a suite of problems in the number 16 benchmark of the *ANL Benchmark Book*. (Argonne National Laboratory 1985) The *Benchmark Book* Problem 16 provided an excellent point of reference for evaluating ARKTRAN-TD. Three separate fast reactor transient problems: a delayed supercritical transient, a prompt supercritical transient, and a rod ejection followed by rod insertion were explored and compared against solutions from TIMEX and TDA provided in the *Benchmark Book*. (Hill and Reed 1976) (Engle 1967) The next section provides a detailed geometry of the benchmark setup. Subsequent sections present and discuss results.

5.1 ANL Time-Dependent Benchmark Problem 16 Geometry

The ANL time-dependent benchmark Problem 16, provides a 2-group fast reactor transient benchmark containing seven homogeneous slabs and fully utilizing the time-dependent theory (including delayed neutron effects) developed in Chapter 3. Three separate transient situations are then formulated from this base model, where the transients are initiated by changing the density of the material in some of the fuel zones

or via ejection/insertion of a control rod. Some of the basic parameters of the benchmark problem include:

- One-dimensional, two-group neutron transport simulation
- Isotropic scattering (scattering moments are P_0)
- Vacuum boundary conditions at the problem boundaries
- Initial condition (t=0) steady-state critical
- Six delayed neutron precursor groups

Figure 10 depicts Problem 16's basic geometry, consisting of a total of 7 coarse slabs and spanning a total distance of 226.748 cm in the z-direction. The lengths for each slab can be found in Figure 10. Slabs 1 and 7 contain a blanket type material; slabs 2, 4, and 6 are the core material; and slabs 3 and 5 contain a mixture of sodium and control rod materials. The benchmark specification called for S_4 quadrature and a fine mesh density per slab equivalent to the values in Table 4. Convergence criteria was set to 10^{-5} for the flux and 10^{-6} for the eigenvalue when resolving the initial condition.

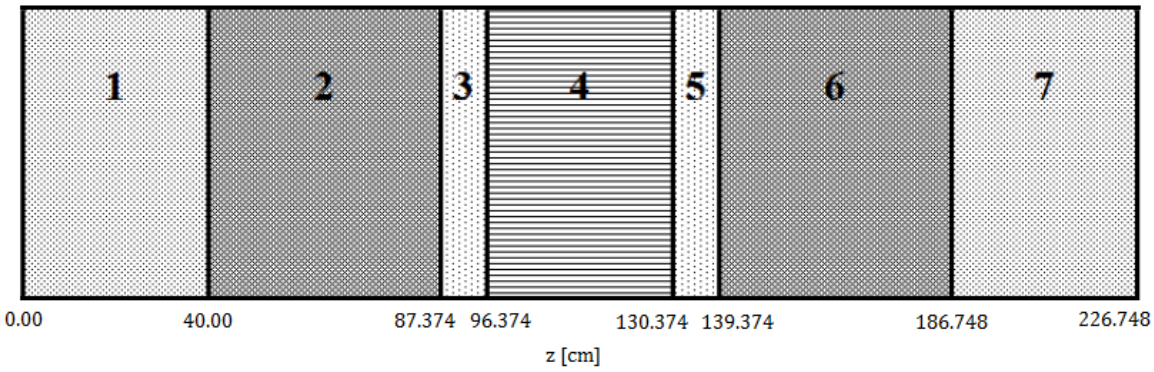

Figure 10: *ANL Benchmark Book* Problem 16 geometry.

Table 4: *ANL Benchmark Book* Problem 16 fine mesh density.

Slab	Number of Fine Meshes
1	20
2	24
3	5
4	16
5	5
6	24
7	20

There are no external sources present in the model (e.g. all neutrons are a product of fission or decay of fission products). Furthermore, all neutrons (both prompt and delayed) are considered to be born in the fast group; meaning that $\chi_1 = 1.0$ and $\chi_2 = 0.0$. The delayed neutron precursor parameters are defined in Table 5 for all six delayed groups, and the velocity of neutrons in energy group 1 was given as 5.40249×10^8 cm/s and 9.19118×10^7 cm/s in energy group 2. Cross sections for each of the seven slab regions are given in Table 6. The benchmark problem geometry is initially made critical by dividing the production cross sections ($\nu\sigma_{f,g}$) by the steady-state criticality eigenvalue, *k*, before beginning the transient calculation.

Table 5: *ANL Benchmark Book* Problem 16 delayed neutron parameters.

Delayed Group	β_l	λ_l [s^{-1}]
1	8.100×10^{-5}	0.0129
2	6.870×10^{-4}	0.0311
3	6.120×10^{-4}	0.1340
4	1.138×10^{-3}	0.3310
5	5.120×10^{-4}	1.2600
6	1.700×10^{-4}	3.2100

Table 6: *ANL Benchmark Book* Problem 16 cross sections [cm^{-1}].

Slab	Group	σ_a	$\nu\sigma_f$	σ_t	$\sigma_s(g,g)$	$\sigma_s(g',g)$
1, 7	1	3.85800E-03	3.84410E-04	2.41100E-01	2.33644E-01	0.00000E-00
	2	1.01960E-02	3.27760E-04	4.17200E-01	4.07004E-01	3.59800E-03
2, 4, 6	1	5.10400E-03	7.45180E-03	1.84900E-01	1.77711E-01	0.00000E-00
	2	1.30790E-02	1.10612E-02	3.66800E-01	3.53721E-01	1.71680E-03
3, 5	1	8.61000E-03	0.00000E-00	9.43200E-02	8.57100E-02	0.00000E-00
	2	1.45930E-02	0.00000E-00	1.87620E-01	1.71310E-01	1.71860E-03

Solutions for the three transients using the Problem 16 geometry were computed using TIMEX and TDA, resulting in plots of power versus time and scalar flux at specific points in time. These results were tabulated and included in the *ANL Benchmark Book*, allowing for comparison of these transient results to the ARKTRAN-TD results. Reviewers then examined the results of both codes, and set a window of time where solutions were considered to be valid for each transient. For completeness, the next two brief sub-sections will discuss the TIMEX and TDA code approaches to solving the time-dependent transport equation. However, both of these simulation methods are fundamentally different than the quasi-static RKF Time method coupled to a Source Iteration Space method developed for this dissertation.

5.1.1 Brief Description of the TIMEX Code

The TIMEX code utilizes the discrete ordinates approximation of the transport code and solves across a fine mesh domain using the Diamond Difference scheme. The transport method is based on ONETRAN, which uses a linear discontinuous finite element representation of for the angular flux. (Hill 1975) The time-domain is differenced explicitly via a method which is considered to be unconditionally stable, to

allow for large time steps to be taken. (Hill and Reed 1976) Furthermore, while the casting of the time-dependent transport equation is explicit, the precursor concentrations are differenced by a fully-implicit scheme. TIMEX utilizes user supplied fixed time step sizes, which can be set over different ranges of time during the course of the transient. Typically, one starts with smaller time steps and then increases the step size as the code progresses further in time. The TIMEX code utilizes single precision data storage, with 14 digits of accuracy on the decimal places.

5.1.2 Brief Description of the TDA Code

The TDA code functions using the discrete ordinates approximation of the Boltzmann transport equation and uses a weighted difference to resolve the space and time domains with automatic coarse mesh rebalance. (Engle and al. 1969) The ANISN code (one-dimensional anisotropic discrete ordinates) is used to solve the Boltzmann equation and TDA is effectively a time-dependent wrapper that uses ANSIN to solve for the space profile. (Engle and al. 1969) There is minimal documentation present in the literature that discusses exactly how the time-domain is cast (explicit vs. implicit). However, the description of the mathematical model in the *ANL Benchmark Book*, leads one to believe the time domain is differenced semi-implicitly using Engle's weighted scheme in a similar manner to the way one differences the spatial domain. Fixed sized time steps are taken by the code, however, the user can specify multiple step sizes a different points in time. ANSIN and TDA use single precision storage with 6 decimal places of accuracy. The low number of significant digits may likely cause discrepancies

in observed code solutions, as only having 6 significant digits to represent the quadrature and weights can cause difficulties is resolving the angular flux solutions.

5.2 Problem 16-A1: Delayed Supercritical Transient

Starting from a stable critical solution, with delayed neutron precursors in equilibrium (e.g. steady-state critical), at time t = 0.00 seconds, the density of the material in slab 2 is increased by 5% and the density of the material in slab 6 is decreased by 5%; resulting in a step insertion of reactivity at t = 0.00 seconds. A 5% change in density results in a 5% change in the number of nuclei per unit volume, which directly results in a 5% change in the cross sections from Table 6. Reactor power is normalized to yield 1.0 neutron/second a t = 0.00 seconds. Plots of total power as a function of time and scalar flux as a function of position at t = 0.00, 0.01, and 1.00 seconds were plotted. Plots are given out to 1.00 seconds for comparison, however the valid time interval of the benchmark is considered to be $0 \leq t \leq 0.1$ seconds. ARKTRAN-TD was run, using a RKF4/5 expansion to solve the time domain, with a residual tolerance of 10^{-3} (see Eq. 2.40 for definition of residual).

5.2.1 Time Step Size

ARKTRAN-TD uses an adaptive time step determined by the RKF4/5 expansion, which represents the optimized time step, to attain a solution at the next time step that is within the desired tolerance. TIMEX and TDA utilized user specified time steps. Table 7 provides the average time step size for ARKTRAN-TD and the fixed time step size for TIMEX and TDA at various times after the initial transient. Over the entire post transient time domain, ARKTRAN-TD takes a time step several orders of magnitude less than the

TDA time step. However, at early times the steps are in line with time steps taken by TIMEX. At times greater than t = 10^{-3} seconds, the ARKTRAN-TD time step remains consistent with the 25 nanosecond to 27 nanosecond steps the code has been taking. TIMEX moves to step sizes almost 100 times larger, while TDA is performing step sizes almost 200 times larger. Though one would expect larger allowable time steps for code methodologies only employing a first-order difference method in the time domain (TIMEX and TDA) and partially implicit solutions, the large discrepancies in time step size could account for the larger difference observed in the power as a function of time at times greater than t = 0.1 seconds between the TIMEX and TDA codes. ARKTRAN-TD was within 2% of the power determined by TIMEX over the entire 1 second past the transient.

Table 7: Problem 16-A1 time step comparison.

Time [s]	$\overline{\Delta t}$ **ARKTRAN-TD [s]**	Δt **TIMEX [s]**	Δt **TDA [s]**
0 to 10^{-6}	$1.00\text{x}10^{-9}$	$2.00\text{x}10^{-8}$	$1.00\text{x}10^{-6}$
10^{-6} to 10^{-5}	$2.20\text{x}10^{-8}$	$2.00\text{x}10^{-8}$	$1.00\text{x}10^{-6}$
10^{-5} to 10^{-4}	$2.30\text{x}10^{-8}$	$2.00\text{x}10^{-8}$	$1.00\text{x}10^{-6}$
10^{-4} to 10^{-3}	$2.60\text{x}10^{-8}$	$2.00\text{x}10^{-8}$	$5.00\text{x}10^{-6}$
10^{-3} to 10^{-1}	$2.70\text{x}10^{-8}$	$2.00\text{x}10^{-6}$	$5.00\text{x}10^{-6}$
10^{-1} to 1	$2.70\text{x}10^{-8}$	$2.00\text{x}10^{-5}$	$5.00\text{x}10^{-5}$

5.2.2 Total Power Behavior Post Transient Onset

The normalized power post transient onset calculated by ARKTRAN-TD, TIMEX, and TDA is shown in Figure 11. Over the entirety of the 1.00 second post transient timeframe, the total power calculated by ARKTRAN-TD is within 2% of the

values from TIMEX. When comparing to the TDA values, ARKTRAN-TD provides solutions within 2% at times less than 10^{-3} seconds, 4% at times less than 0.1 seconds and 6% at times greater than 0.1 seconds. The larger discrepancy between the ARKTRAN-TD power and the TDA power is most likely due to TDA using time steps that are too large for the simulation; as when one observes the choice of step size employed by TIMEX (Table 7) it is quickly apparent that the TIMEX step size is almost the same as the automatically optimized time step obtained by AKCTRAN-TD's RKF4/5 expansion.

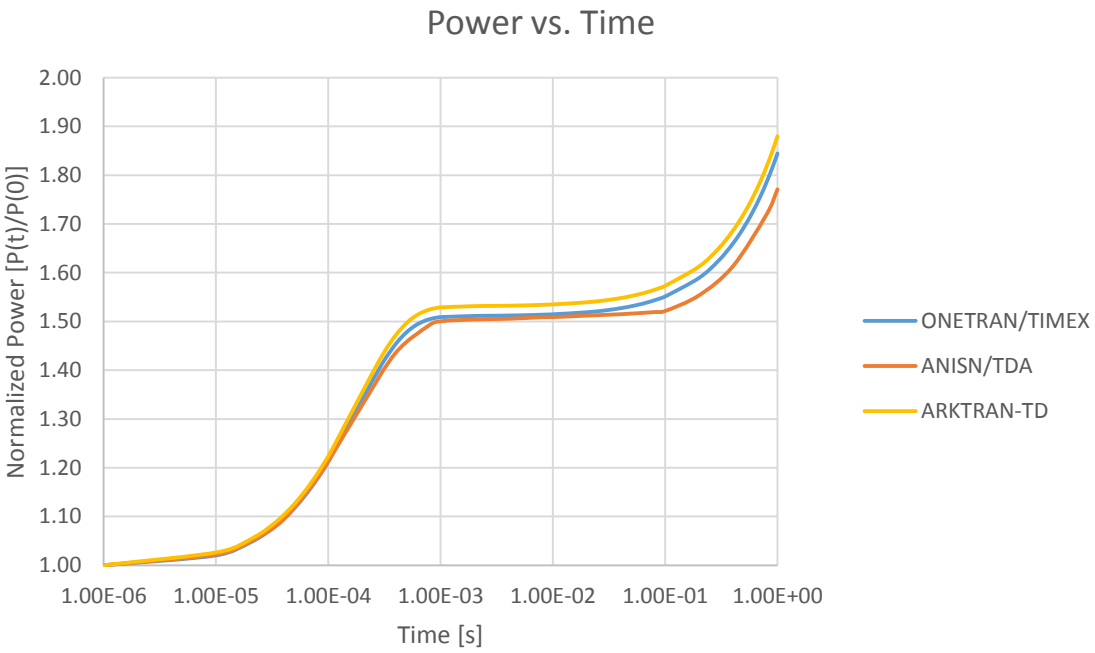

Figure 11: Problem 16-A1 power vs. time post transient.

ARKTRAN-TD's optimized step size seems to show the prompt jump effect starting slightly sooner than the other codes, which accounts for the normalized power being a bit higher than the TIMEX solution. The adaptive step optimization allows for a

more accurate modeling of the delayed neutron effects and initial transient changes at the onset of the problem as time steps 20 times smaller than the other codes were required to attain the residual tolerance. Overall, the ARKTRAN-TD power profile solution is well behaved and comparable to the code to code comparisons provided in the *ANL Benchmark Book.*

5.2.3 Behavior of Neutron Flux as a Function of Time

The Problem 16-A1 benchmark specification called for scalar flux profiles of both energy groups at times of 0.00, 0.01, and 1.00 seconds post transient onset. The first set of flux profiles at t = 0.00 seconds for energy Group 1 and energy Group 2 are shown in Figure 12 and Figure 13, respectively. These two plots essentially show the steady-state solution to the scalar flux for the sodium fast reactor modeled in the Problem 16 benchmark series. Behavior of the initial flux profiles are as expected: the fast flux is highest in the fuel regions, dips in the control regions, and decreases through the blanket as neutrons approach the zero return current boundaries. The Group 1 fluxes obtained from ARKTRAN-TD exhibit a 0.36% mean relative error (Eq. 4.2) from the TIMEX flux profiles and a 0.31% mean relative error from the TDA flux profiles, calculated using an equation of the form of Eq. 4.2. The maximum relative error at t = 0.00 in Group 1 was 2.76% and 1.92% for the TIMEX and TDA solutions, respectively at the boundary. Similar mean relative errors of 0.25% (TIMEX) and 0.21% (TDA) are observed for Group 2 fluxes, with slightly higher maximum relative errors of 2.91% (TIMEX) and 2.22% (TDA). Thermal flux behavior is consistent with the anticipated shape for a fast spectrum reactor, one observes a higher amount of thermal neutrons in the fuel regions

(since they are not being preferentially absorbed), depressions in the control regions, and a decrease through the blanket as neutrons approach the vacuum boundaries.

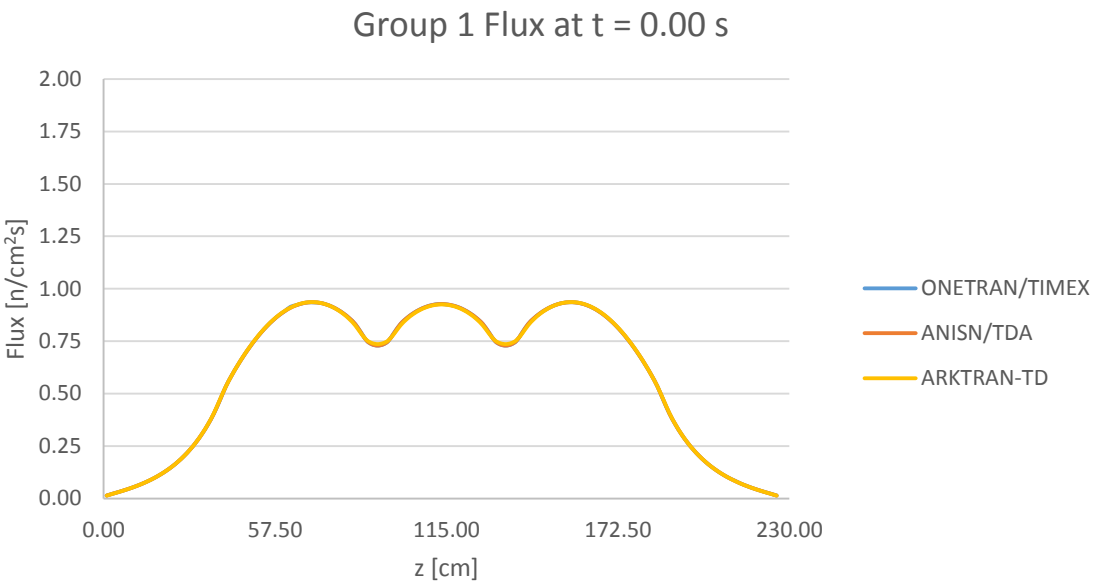

Figure 12: Problem 16 scalar flux in energy group 1 at t = 0.00 seconds.

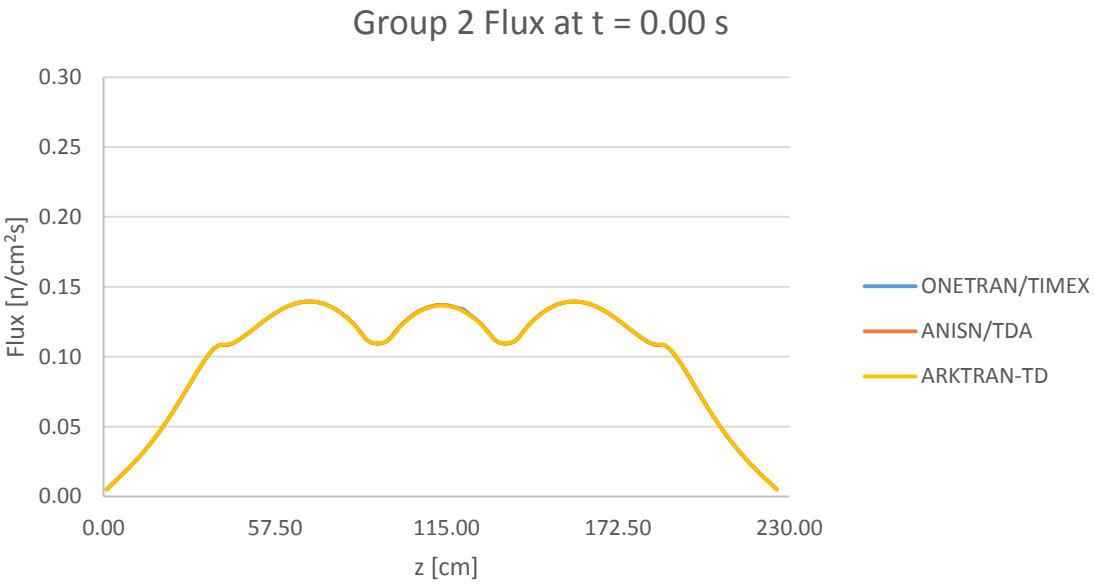

Figure 13: Problem 16 scalar flux in energy group 2 at t = 0.00 seconds.

As the time increases beyond the onset of the transient, since the transient results in a supercritical configuration and increasing power level as a function of time, one would expect the overall magnitude of both the Group 1 and Group 2 fluxes to increase for this reactor. Additionally, the flux profile should shift from its symmetric shape observed at t = 0.00 seconds, to a profile showing a peaked flux in slab 2 and a somewhat depressed flux in slab 6. This results from the increase in density in slab 2 and decrease in slab 6. It is important to note that the flux is proportional to total power. Thus, the difference in total power between codes has a large impact on relative error values in the flux (discussed in the next paragraph).

The Group 1 flux at t = 0.01 seconds is provided in Figure 14 while the Group 2 flux is given in Figure 15. As expected, both the fast and thermal fluxes increased in overall magnitude and an asymmetrical flux profile peaked in slab 2 is observed, while slab 6 yields the smallest fuel zone peak. Mean relative error in the scalar flux for Group 1 is 0.93% for TIMEX values and 1.68% for TDA values; with maximum relative errors of 1.98% (TIMEX) and 3.91% (TDA). Group 2 mean relative errors were calculated to be 0.92% and 1.66% for TIMEX and TDA, respectively; with maximum relative errors of 1.64% (TIMEX) and 2.45% (TDA). At t = 0.01 the relative error in the normalized power was 1.34% (TIMEX) and 1.73% (TDA); accounting for the majority of the relative error in the fluxes. Therefore, the overall shape of the flux profile is of more concern than the relative error values as the total powers diverge between transport code methodologies.

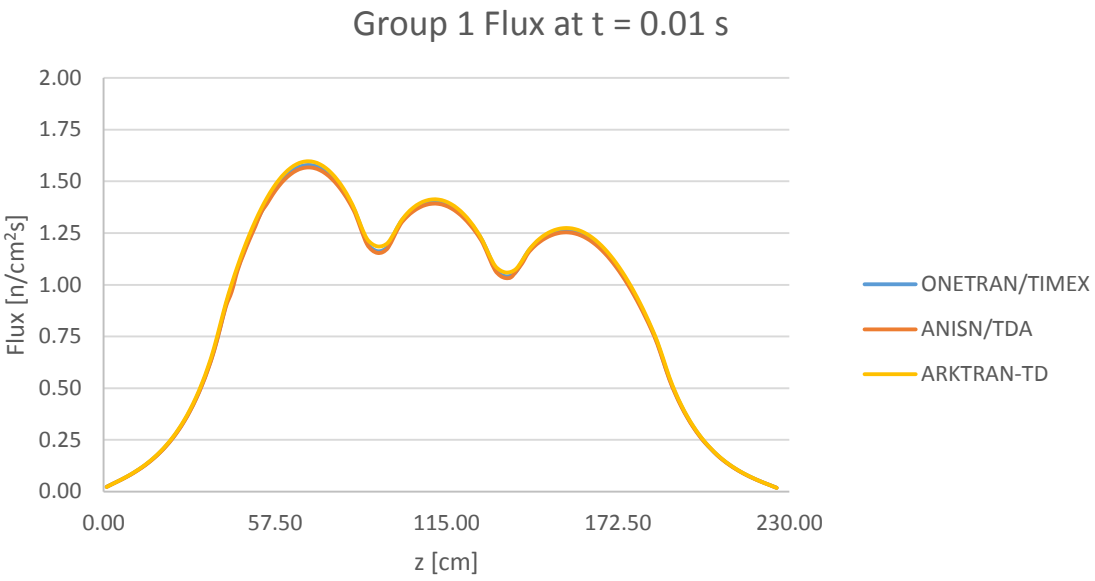

Figure 14: Problem 16-A1 scalar flux in energy group 1 at t = 0.01 seconds.

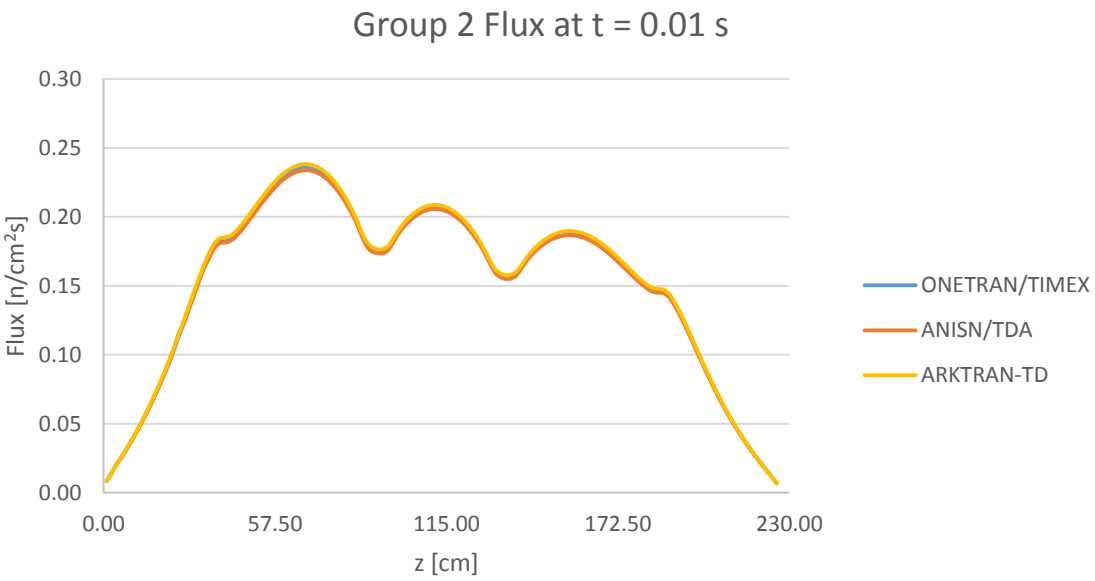

Figure 15: Problem 16-A1 scalar flux in energy group 2 at t = 0.01 seconds.

According to the benchmark documentation, the Problem 16-A1 benchmark was considered valid for code-to-code comparisons up to 0.10 seconds, after which the difference in normalized power between TIMEX and TDA was between 2% and closer to 4% as the simulations approached t = 1.00 second. Though outside the recommended time range for simulation, scalar flux and power data was provided, so it was compared to results obtained from ARKTRAN-TD. Interestingly, ARKTRAN-TD provided both flux and power results within reasonable errors at t = 1.00 second when compared to TIMEX. Relative error in normalized power at 1.00 second was 1.90% for TIMEX, which was within the 2% desired range for code-to-code comparisons expressed by the benchmark reviewers. However, the relative error in normalized power was 6.13% between ARKTRAN-TD and TIMEX (it was 4.16% between TIMEX and TDA). The discrepancy with TDA at longer simulation times is most likely attributed to poor numerical precision (6 digits vs. 14+ digits for the other codes) and time steps which are much larger than TIMEX or ARKTRAN-TD.

Figure 16 gives the Group 1 scalar flux plots and Figure 17 provides the same for Group 2. Mean relative errors in the scalar flux a t = 1.00 second between ARKTRAN-TD and TIMEX were 1.46% and 1.47% for Group 1 and Group 2, respectively. The maximum relative error for the same codes was 2.54% in Group 1 and 2.08% in Group 2. Average relative errors between ARKTRAN-TD and TDA were consistent with the observed relative error in normalized power at t = 1.00 second.

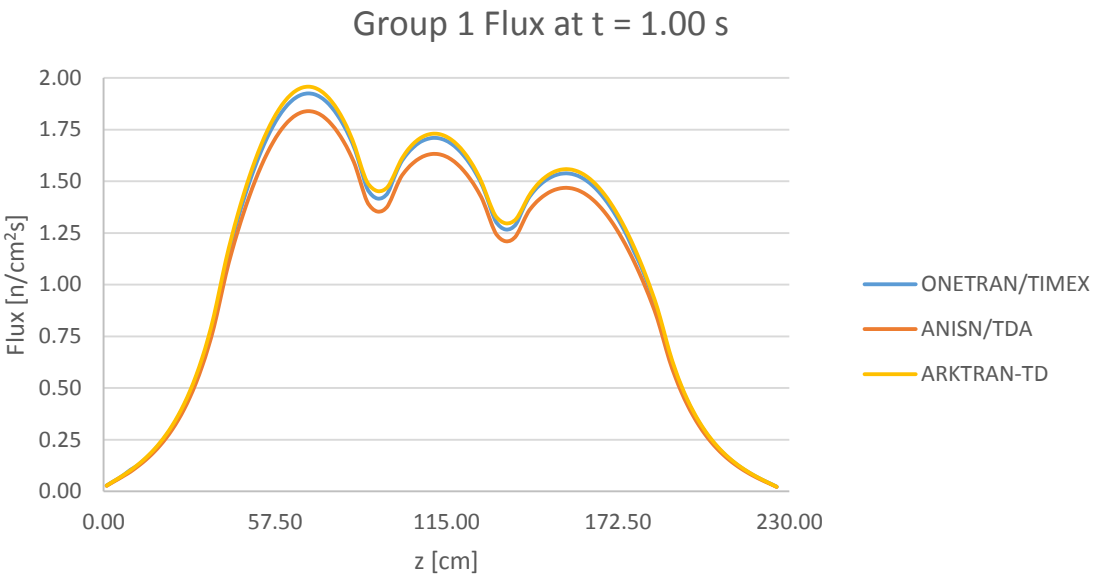

Figure 16: Problem 16-A1 scalar flux in energy group 1 at t = 1.0 seconds.

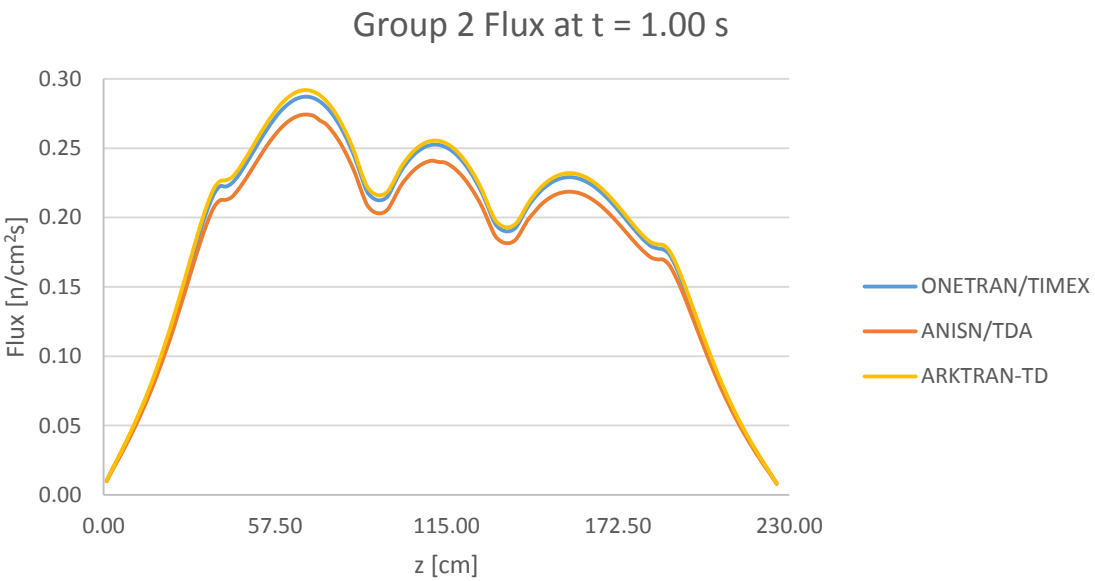

Figure 17: Problem 16-A2 scalar flux in energy group 2 at t = 1.0 seconds.

5.3 Problem 16-A2: Prompt Supercritical Transient

Starting from the stable critical solution, the density of the material in slab 2 is increased by 10% and the density of the material in slab 6 is decreased by 10%; resulting in a step insertion of reactivity at t = 0.00 seconds that yields a core configuration that is supercritical on the prompt neutron contribution alone. Reactor power was normalized to yield 1.0 neutron/second a t = 0.00 seconds. Plots of total power as a function of time and scalar flux as a function of position at 10^{-4} seconds were generated. The plot of normalized power as a function of time is given to 10^{-3} seconds and the valid time interval of the benchmark is considered to be $0 \leq t \leq 10^{-3}$ seconds for code-to-code comparisons. ARKTRAN-TD was run, using a RKF4/5 expansion to solve the time domain, with a residual tolerance of 10^{-3} for the interval $0 \leq t \leq 8.00\text{x}10^{-4}$ seconds and was increased to 10^{-2} for $t > 8.00\text{x}10^{-4}$ seconds (see Eq. 2.40 for definition of residual).

5.3.1 Time Step Size

Table 8 provides the average adaptive time step size for ARKTRAN-TD in the given simulation time range and the fixed time step size for TIMEX and TDA at various times after the initial transient. The adaptive time step determined by the RKF residual in ARKTRAN-TD resulted in average time steps which started around 20 nanoseconds and then decreased, as the prompt supercritical nature of the transient resulted in high flux and power gradients as a function of time. At $8.00\text{x}10^{-4}$ seconds, the residual tolerance was decreased by an order of magnitude to allow for the simulation to run in a reasonable amount to time, as to obtain a residual tolerance of 10^{-3} in that time range would have resulted in time steps on the order of 10^{-12} to 10^{-15} seconds using RKF4/5. The TIMEX

code begins with time steps on the same order of magnitude as ARKTRAN-TD, but then the time step is increased to 100 nanoseconds in the same region where the adaptive RKF is decreasing the step size to resolve the solution. The TDA code begins with a time step which is an order of magnitude higher than ARKTRAN-TD and TIMEX, and then increases the time set size by multiplying each Δt by 1.024798 times the previous Δt.

Table 8: Problem 16-A2 time step comparison.

Time [s]	$\overline{\Delta t}$ **ARKTRAN-TD* [s]**	Δt **TIMEX [s]**	Δt **TDA [s]**
0 to 10^{-5}	$2.00\text{x}10^{-8}$	$1.00\text{x}10^{-8}$	$2.50\text{x}10^{-7}$
10^{-5} to 10^{-4}	$1.70\text{x}10^{-8}$	$1.00\text{x}10^{-8}$	$1.024798\Delta t_{i-1}$
10^{-4} to $4\text{x}10^{-4}$	$1.25\text{x}10^{-8}$	$1.00\text{x}10^{-7}$	$1.024798\Delta t_{i-1}$
$4\text{x}10^{-4}$ to $5\text{x}10^{-4}$	$1.00\text{x}10^{-8}$	$1.00\text{x}10^{-7}$	$1.024798\Delta t_{i-1}$
$5\text{x}10^{-4}$ to $6\text{x}10^{-4}$	$8.00\text{x}10^{-9}$	$1.00\text{x}10^{-7}$	$1.024798\Delta t_{i-1}$
$6\text{x}10^{-4}$ to $8\text{x}10^{-4}$	$7.00\text{x}10^{-9}$	$1.00\text{x}10^{-7}$	$1.024798\Delta t_{i-1}$
$8\text{x}10^{-4}$ to 10^{-3}	$1.00\text{x}10^{-8}$	$1.00\text{x}10^{-7}$	$1.024798\Delta t_{i-1}$

* RKF residual tolerance was increased for $t > 8\text{x}10^{-4}$ to allow for reasonable Δt.

5.3.2 Total Power Behavior Post Transient Onset

The total power as a function of time after the prompt supercritical transient for ARKTRAN-TD, TIMEX, and TDA is plotted in Figure 18 on a log scale. For time less than 10^{-4} seconds, the relative error in the power is 4.64% for TIMEX and 2.56% for TDA when comparing to ARKTRAN-TD. Continuing further along in time, the relative errors increase to 9.40% (TIMEX) and 6.76% (TDA) at $t = 4\text{x}10^{-4}$ seconds to a maximum of 19.78% (TIMEX) and 20.79% (TDA) at $t = 10^{-3}$ seconds. Given the successful benchmarking with Problem 16-A1, the RKF's desire to decrease the time step as the

prompt supercritical transient progresses in time, and the fact that ARKTRAN-TD uses a 4^{th} order expansion of the time derivative with 5^{th} order truncation error, the resulting power profile seems reasonable. It is probable that the first-order difference methods used by TIMEX and TDA to resolve the time derivative, coupled with the use of increasing fixed time steps (when they may need to be decreasing to resolve large flux gradients in time with accuracy) is resulting in an under prediction of the power as a function of time. Thus, contributing to the larger errors observed in the power. Figure 18 shows all three codes yielding results with similar shape functions. It appears that the higher order method used by ARKTRAN-TD, coupled with the application of error control methods on the step size, yield a solution that is more accurate for the difficult to resolve prompt supercritical transient case.

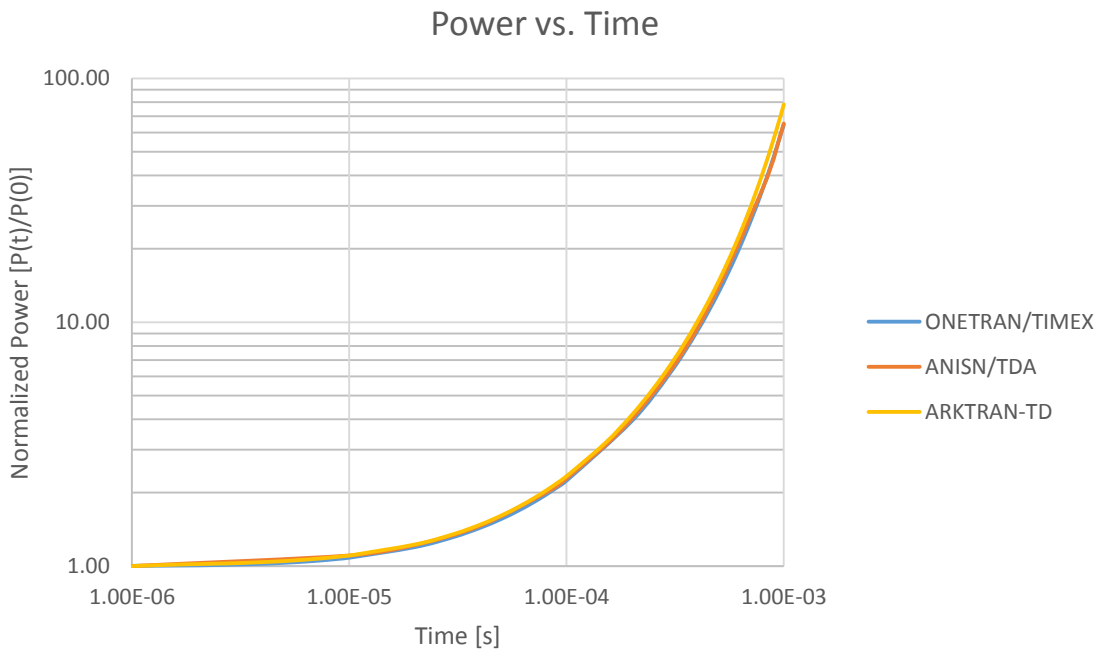

Figure 18: Problem 16-A2 power vs. time post transient.

5.3.3 Behavior of Neutron Flux as a Function of Time

The Problem 16-A2 benchmark specification called for flux profiles to be provided at two time points within the valid time range for code-to-code comparisons. The first plot at t = 0.00 seconds is identical to the plots in Figure 12 and Figure 13, as these represent steady-state starting points just before the transient begins. The second plot was specified a $t = 10^{-4}$ seconds after the transient. Figure 19 provides the scalar flux as a function of position for Group 1, while Figure 20 depicts the same for Group 2 at the specified time. Both the fast and thermal flux plots exhibit a similar change in shape, when compared to the Problem 16-A1 benchmark. However, the effects of the transient are more pronounced (i.e. higher peaking in slab 2 and a larger depression in the relative sense in zone 6) due to the larger change in density for the two regions, and thus the larger transient observed.

The mean relative error in the scalar flux for Group 1 was 2.55% (TIMEX) and 2.14% (TDA); with maximums of 3.77% (TIMEX) and 3.27% (TDA). The Group 2 scalar flux resulted in similar mean relative errors of 2.56% and 2.12% for TIMEX and TDA, respectively; with maximums of 3.30% (TIMEX) and 2.93% (TDA). These are consistent with the relative error in the power at $t = 10^{-4}$ of 4.63% for TIMEX and 2.56% for TDA. The reason the relative errors are slightly larger in the power is due to the power only coming from scalar flux values in regions containing fuel, while the average relative error in the scalar flux is over the entire spatial domain and is lower due to near equal scalar fluxes outside the major fuel zones.

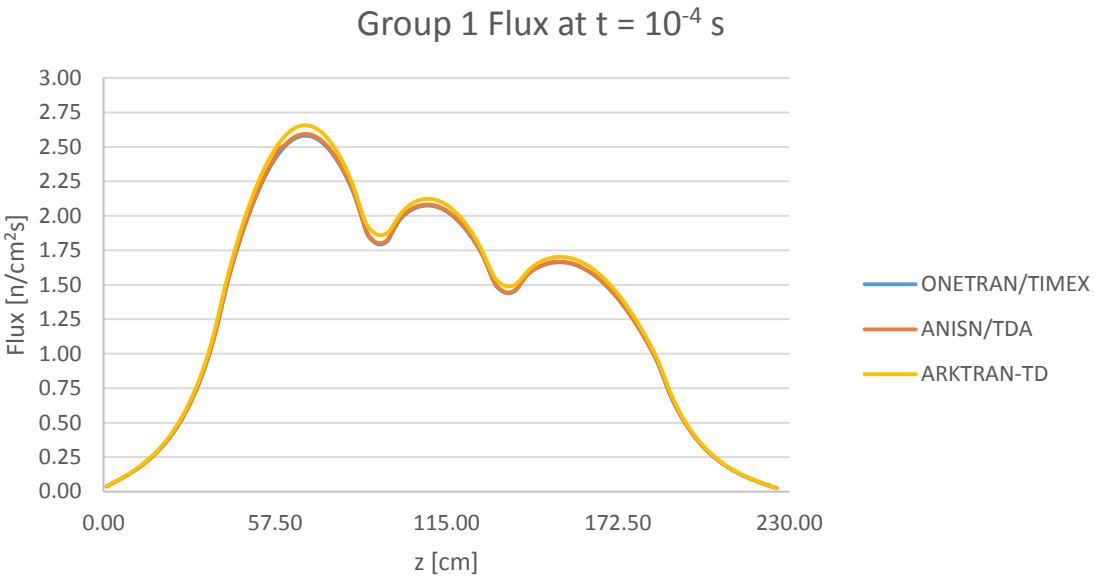

Figure 19: Problem 16-A2 scalar flux in energy group 1 at t = 10^{-4} seconds.

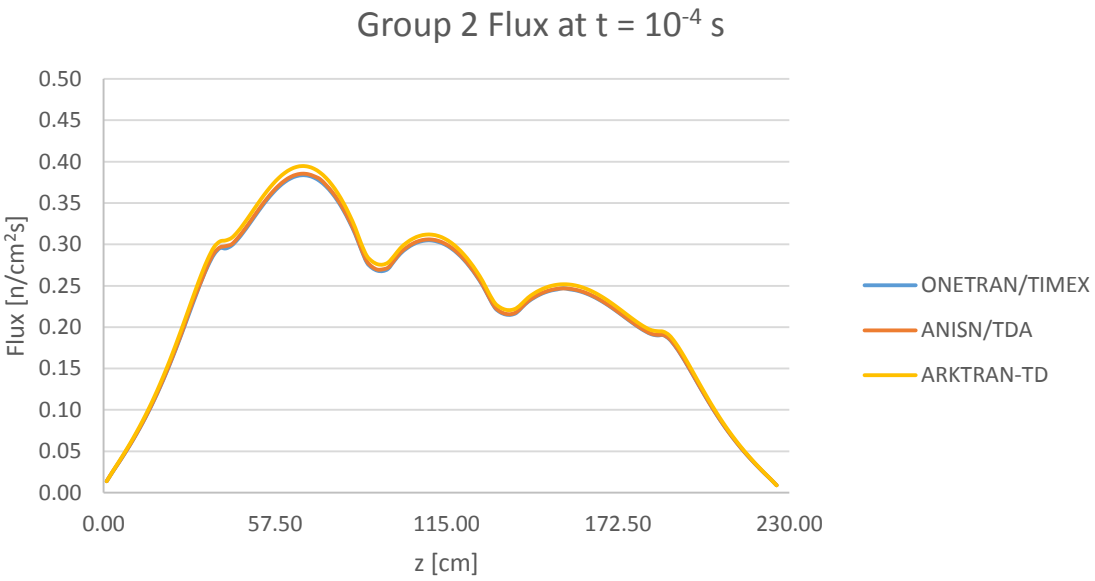

Figure 20: Problem 16-A2 scalar flux in energy group 2 at t = 10^{-4} seconds.

5.4 Problem 16-A3: Rod Ejection and Rod Insertion Transient

Starting from the initial stable-critical solution at t = 0.00 seconds, the material in slab 5 is changed from a mixture of control rod material and sodium to pure sodium; resulting in a control rod bank ejection. When a time of $t = 10^{-4}$ seconds, the material in slab 3 is changed from a mixture of control rod material and sodium to 100% control rod material; resulting in a full insertion of the control rod bank. Thus, this transient models both an initial rod ejection and subsequent rod insertion. Cross sections for the pure sodium and control rod material regions (used in the transient) are provided in Table 9.

Table 9: Problem 16-A3 supplemental cross sections used in transient simulation.

Material	Group	σ_a	$\nu\sigma_f$	σ_t	$\sigma_s(g,g)$	$\sigma_s(g',g)$
Sodium	1	3.71300E-03	0.00000E-00	6.83000E-02	6.32930E-02	0.00000E-00
	2	4.70100E-02	0.00000E-00	1.25800E-02	1.21099E-02	1.29400E-03
Control	1	1.73210E-02	0.00000E-00	1.79500E-01	1.59078E-01	0.00000E-00
Rod	2	5.46390E-02	0.00000E-00	3.90300E-01	3.35661E-01	3.10100E-03

Problem 16-A3 provided an interesting point of analysis for ARKTRAN-TD, when attempting to solve the problem using the adaptive time step size with RKF4/5 and a residual tolerance of 10^{-3} (see Eq. 2.40 for definition of the residual), time step sizes of 10^{-18} to 10^{-15} seconds were required to resolve the time derivative within the specified 5^{th} order truncation error cutoff. This made computation time unfeasible, as it would require a great amount of time to solve the entire simulation time of 0.01 seconds as the step size is incredibly small. The second approach undertaken involved reducing the residual tolerance to 10^{-2}, as implemented at later times in Problem 16-A2. However, this transient

causes the power to increase at an even higher rate than Problem 16-A2, and time step sizes were only reduced to 10^{-16} to 10^{-14} seconds; again unreasonable step sizes for the entire simulation time. It may be noted that one could just fix the time step size and run the simulation like TIMEX and TDA. However, this would effectively cause ARKTRAN-TD to solve the time-domain with a 4^{th} order Runge-Kutta (without error control) coupled to a Source Iteration to resolve the spatial domain. In order to get the time steps into a range that is reasonable for computation time, one needs the majority of the steps to be on the order of 10^{-10} to 10^{-8} seconds. From the initial attempt to run the RKF4/5 it is known that time step sizes in this range are well outside acceptable values for the truncation error, resulting in incorrect solutions should one fix the time step.

In an attempt to mitigate the issues stemming from the small step size required by the high order expansion and understanding that generally one can use slightly larger time steps with lower order expansion methods. An RKF2/3 was added into ARKTRAN-TD, allowing for a 2^{nd} order expansion of the transport equation in the time domain with 3^{rd} order truncation error. Furthermore, this method still allows for adaptive time step sizes with error control of the truncation error within a specified tolerance; guaranteeing the resulting solution is within acceptable limits. With the tolerance of the RKF residual set to 10^{-3}, one observes step sizes from 10^{-15} to 10^{-8} seconds, with the majority of the steps occurring between $5x10^{-11}$ and $2x10^{-8}$ seconds. Thus, the results presented for Problem 16-A3 with ARKTRAN-TD were calculated using a RKF2/3 expansion to solve the time domain, with a residual tolerance of 10^{-3}. This methodology led to the

implementation of adaptive RKF expansion orders as a function of time in ARKTRAN-TD, which are presented and discussed in Chapter 6.

5.4.1 Time Step Size

Table 10 provides a comparison of the average time step taken by ARKTRAN-TD's RKF2/3 expansion of the time-domain with error control to the fixed steps taken by TIMEX and TDA. In the ranges of depicted in Table 10, the average Δt taken by ARKTRAN-TD ranged from $5.0\text{x}10^{-11}$ to $2.3\text{x}10^{-8}$ seconds. During a brief period following the insertion of the control rod banks, time steps on the order of 10^{-14} seconds were required to resolve the rapidly changing transient. Observing the difference in step size between the code, ARCTRAN-TD's time steps are a factor of 10^{-3} to 10^{-2} smaller than the steps taken by TIMEX and a factor of 10^{-4} to 10^{-3} smaller than TDA.

Table 10: Problem 16-A3 time step comparison.

Time [s]	$\overline{\Delta t}$ **ARKTRAN-TD [s]**	Δt **TIMEX [s]**	Δt **TDA [s]**
0 to 10^{-7}	$5.00\text{x}10^{-11}$	$1.00\text{x}10^{-8}$	$5.00\text{x}10^{-7}$
10^{-7} to 10^{-5}	$5.00\text{x}10^{-10}$	$1.00\text{x}10^{-8}$	$5.00\text{x}10^{-7}$
10^{-5} to $5\text{x}10^{-5}$	$2.00\text{x}10^{-10}$	$1.00\text{x}10^{-8}$	$5.00\text{x}10^{-7}$
$5\text{x}10^{-5}$ to 10^{-4}	$1.50\text{x}10^{-11}$	$1.00\text{x}10^{-8}$	$5.00\text{x}10^{-7}$
10^{-4} to $2\text{x}10^{-4}$	$1.00\text{x}10^{-11}$	$1.00\text{x}10^{-8}$	$5.00\text{x}10^{-7}$
$2\text{x}10^{-4}$ to $3\text{x}10^{-4}$	$5.00\text{x}10^{-11}$	$1.00\text{x}10^{-8}$	$1.004996\Delta t_{i-1}$
$3\text{x}10^{-4}$ to $7\text{x}10^{-4}$	$5.00\text{x}10^{-10}$	$1.00\text{x}10^{-7}$	$1.004996\Delta t_{i-1}$
$7\text{x}10^{-4}$ to 10^{-3}	$4.00\text{x}10^{-9}$	$1.00\text{x}10^{-7}$	$1.004996\Delta t_{i-1}$
10^{-3} to 10^{-2}	$2.30\text{x}10^{-8}$	$1.00\text{x}10^{-6}$	$1.004996\Delta t_{i-1}$

The discrepancies in step size shown in Table 10 are noteworthy, since the results comparison for Problem 16-A3 look at error controlled step size with 2^{nd} order time differenced solutions calculated by ARKTRAN-TD (as opposed to 4^{th} order for Problems 16-A1 and 16-A2) and non-error controlled step size 1^{st} order time differenced solutions from TIMEX and TDA.

5.4.2 Total Power Behavior Post Transient Onset

The normalized total power as a function of time for the control rod bank ejection at t = 0.00 seconds and insertion at t = 10^{-4} seconds is shown in Figure 21 on a log scale. Comparing the solutions in the 10^{-6} to 10^{-5} second range, TIMEX and TDA calculate slightly higher powers than ARKTRAN-TD. However, the major jump in power due to the rod ejection occurs during the 10^{-5} to 10^{-4} second range. At t = 10^{-4} seconds ARKTRAN-TD peaks at a relative power of 3268.13 compared to 3038.34 from TIMEX and 3044.50 from TDA. This results in a relative error of the ARKTRAN-TD solution of 7.56% from TIMEX and 7.35% from TDA at the maximum power. The relative error in power increases in the 10^{-4} to 10^{-3} second range as the higher power level before the rod bank insertion results in a longer die away time for ARKTRAN-TD. At 10^{-2} seconds relative errors are 8.14% and 9.39% TIMEX and TDA, respectively.

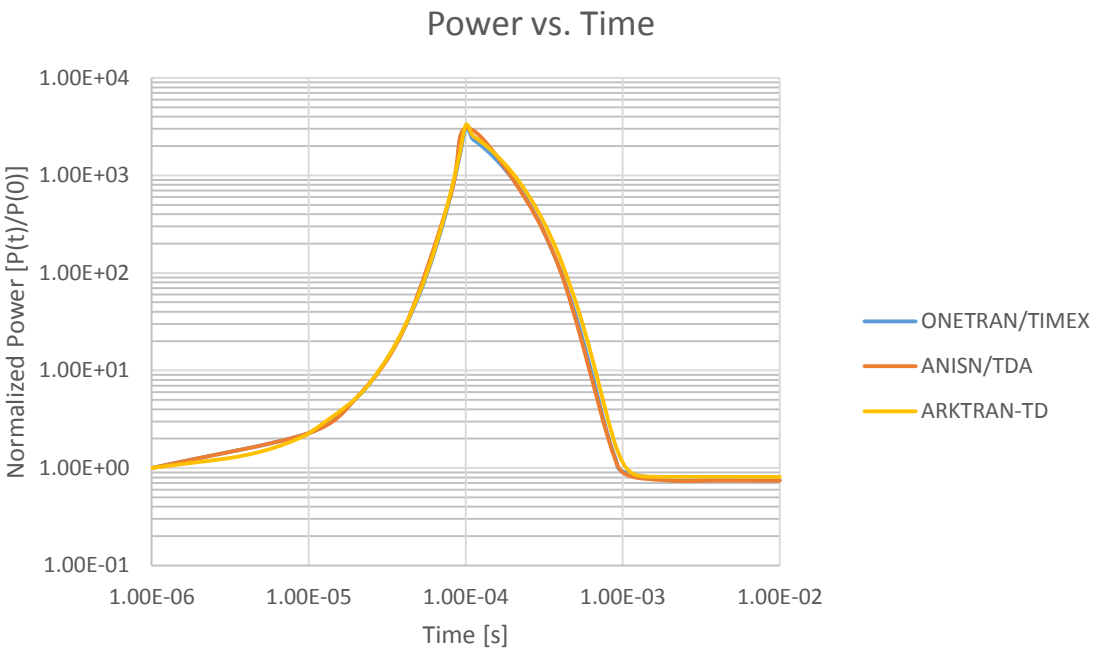

Figure 21: Problem 16-A3 power vs. time post transient.

5.4.3 Behavior of Neutron Flux as a Function of Time

The Problem 16-A3 benchmark specification called for flux profiles to be provided at two time points within the valid time range for code-to-code comparisons. The first plot at t = 0.00 seconds is identical to the plots in Figure 12 and Figure 13, as these represent steady-state starting points just before the transient begins. The second plot was specified a t = 10^{-4} seconds after the transient, where the flux and power are at their maximum values just before the control rod bank in slab 3 is fully inserted. Figure 22 provides the scalar flux as a function of position for Group 1. The scalar flux follows the expected behavior after the control rod is ejected from slab 5, the flux grows more rapidly in the uncontrolled region while being somewhat depressed by the partially

inserted rod bank in zone 3. Figure 23 depicts the scalar flux in energy group 2 as a function of position. The thermal scalar flux also follows the same increasing asymmetrical shape as the fast flux. However, in the region where the rod bank was ejected and increase in the thermal flux profile is observed while a decrease was seen in the fast flux. This is due to the thermalization of the fast flux in the now purely sodium region, which depresses the fast flux and increases the thermal flux at this location. The mean relative error of the ARKTRAN-TD scalar flux profiles for Group 1 were 7.70% (TIMEX) and 7.27% (TDA); with maximums of 8.84% (TIMEX) and 8.40% (TDA). The Group 2 scalar flux resulted in similar mean relative errors of 7.64% and 7.25% for TIMEX and TDA, respectively; with maximums of 8.22% (TIMEX) and 7.70% (TDA). These are consistent with the relative error in the power at $t = 10^{-4}$ of 7.56% for TIMEX and 7.35% for TDA; accounting for a major portion of the relative error in the scalar fluxes between codes.

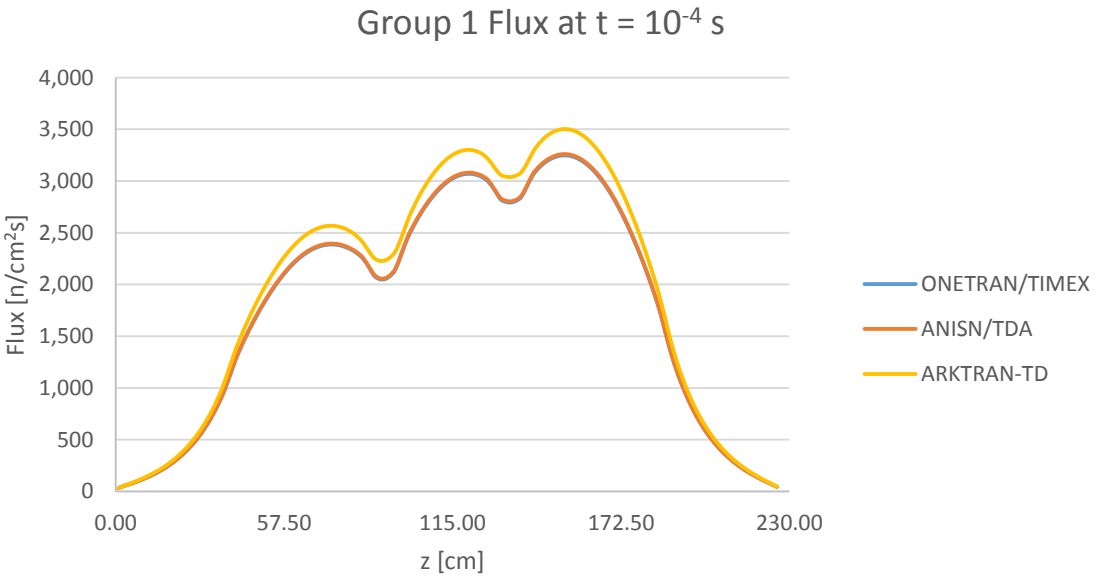

Figure 22: Problem 16-A3 scalar flux in energy group 1 at $t = 10^{-4}$ seconds.

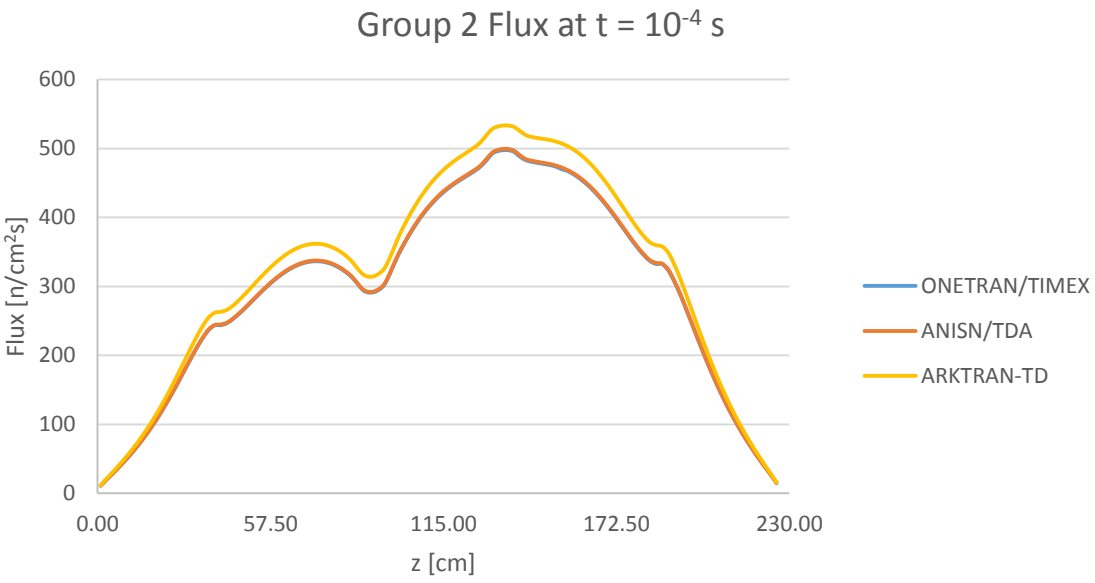

Figure 23: Problem 16-A3 scalar flux in energy group 2 at t = 10^{-4} seconds.

CHAPTER 6

APPLICATION OF ADAPTIVE AND OPTIMIZED RKF EXPANSION ORDERS

Based on the positive results obtained when using a lower order Runge-Kutta-Fehlberg pair to expand the transport equation in the time-domain, an adaptive method which automatically selects the RKF pair providing the largest time step size meeting the specified truncation error tolerance was developed to provide expedited computation times. Since the residual sets a limit on the size of the truncation error term, should a lower order RKF pair provide a time step larger than that of a higher order RKF pair with the same error tolerance, computation time will decrease for two reasons. First, lower order RKF pairs require fewer functional evaluations per step (RKF2/3 requires 40% fewer than RKF4/5) and reducing the number of equations results in faster computation time. Second, a larger time step meeting the same convergence criteria on the truncation error means a faster simulation (due to a smaller number of time steps overall) while still maintaining the desired accuracy of the truncation error. Overall, this gives ARKTRAN-TD the ability to adaptively optimize its choice of expansion in the time-domain (using 1^{st} through 4^{th} order Runge-Kutta, with 2^{nd} to 5^{th} order truncation errors) while still allowing for adaptive time step sizes meeting the desired tolerances. The first section in this Chapter addresses the application of the adaptive RKF scheme in ARKTRAN-TD and the subsequent sections compare results of the adaptive RKF scheme with results

using fixed RKF expansion pairs. The coefficients, truncation error, and function expansions for alternative RKF1/2, RKF2/3, and RKF3/4 can be found in Appendix B.

6.1 Theory and Implementation in ARKTRAN-TD

Observing the behavior of the error controlled time steps while ARKTRAN-TD is executing, it was noted that while the time steps change frequently (generally by less than a couple of percent per step), there is a tendency for the same set of time steps to repeat in a semi-periodic manner over particular time ranges post transient onset. Because of this phenomena, it would be unwise to implement an adaptive RKF determination sequence every set number of evaluated time steps, since this may result in only seeing the same set of step sizes for the RKF expansion pairs. Furthermore, one does not want to check the feasibility of all the expansion pairs (RKF1/2, RKF2/3, RKF3/4, and RKF4/5) implemented in ARKTRAN-TD at each time step, as this would increase the number of equations evaluated for each time step and increase the overall computation time – the exact opposite of the goal for implementing an *adaptive* RKF scheme.

Another method could force the ARKTRAN-TD to evaluate each RKF expansion pair every so many time steps, determine the average Δt required for each expansion order (e.g. every 500 time steps, do 5 steps for each of RKF1/2, RKF2/3, RKF3/4, and RKF4/5), then force the use of the RKF pair with the largest average Δt for the next 500 steps. The issue with this approach is the code will be performing the time step using whatever expansion pair is specified, so if RKF1/2 requires a time step of 10^{-100} seconds to resolve the truncation error, a step of 10^{-100} seconds would be taken and could

introduce numerical instabilities in the solution as the code multiplies and divides by Δt in many places; again resulting in undesirable solution numerics, *et cetera*.

One could impose a limit on the volumetric time rate of change of the neutron population, $\frac{1}{v}\frac{d\Psi}{dt}$, where if the value of this quantity was less than a user imposed limit to upgrade or downgrade to the next RKF expansion pair the code would use the target pair for the next step. For example, if the limit to switch from RKF4/5 to RKF3/4 was $2.5\text{x}10^{8}$ n/cm^{3}s, and the calculated value of $\frac{1}{v}\frac{d\Psi}{dt}$ is $2.0\text{x}10^{8}$ n/cm^{3}s, in the next step ARCTRAN-TD would use RKF3/4. This approach may seem feasible at the onset but there several limitations which make this approach unattractive. First, there is no accounting for the step size used. Therefore it is possible, to downgrade to a lower RKF pair which requires fewer function evaluations but in the process decrease the step size; a more detrimental effect than the additional function evaluations. Second, there is no guarantee that the next lower (or next higher) order RKF pair will provide the optimum time step. It may be more advantageous to go directly from RKF4/5 to RKF2/3 and skip the 3rd order pair entirely. Lastly and most significantly, this would require the user to have a tremendously deep understanding of the physics of the transient (particularly the behavior of $\frac{1}{v}\frac{d\Psi}{dt}$ as a function of time, as well as where in time the step sizes stabilize between RKF expansion pairs) *a priori* to running the simulation.

Taking into account all of the limitations of the methods for creating a code with an adaptive RKF expansion order noted, it is clear that whatever method one uses to

make the automatic determination of the RKF expansion order, there are several major factors which must be accounted for to remove all of the potential issues raised thus far:

- No required *a priori* knowledge of the problem physics
- Ability to determine the RKF pair resulting in the largest Δt
- Must not actually apply the step to avoid numerical instabilities caused by Δts approaching zero
- Accounts for the potential semi-periodic repetition of Δts
- Does not evaluate all RKF pairs at each time step

Taking all of these requirements into consideration, an adaptive method was developed accounting for all of the potential issues just discussed. ARKTRAN-TD's adaptive RKF scheme functions by evaluating the values of Δt that would be used to obtain a solution within the residual tolerance using RKF1/2, RKF2/3, RKF3/4, and RKF4/5 every *random stride* of time steps. The *random stride* is determined by:

$$Random\ Stride = \eta \cdot (\text{Maximum Stride} - 1) \tag{6.1}$$

where, η is a random number and *maximum stride* is a user supplied value representing the maximum number of time steps ARKTRAN-TD can go before evaluating which RKF expansion pair would provide the largest Δt. The time step is not actually taken when ARKTRAN-TD is determining the Δts that would result from each of the RKF pairs, removing the ability to introduce numerical instabilities from suboptimum expansion orders at that point in time. It also removes the issue caused by the semi-periodic repetition of time steps, since the code making its determination of the stride between

"checkups" based on a randomly determined number of time steps. Furthermore, no knowledge of the physics of the transient is required *a priori*. After making the determination of which RKF expansion pair provides the largest Δt, ARKTRAN-TD locks that RKF expansion pair for the next *random stride* of time steps. Figure 24 provides a pseudo-code representation of the process, as implemented in ARKTRAN-TD.

```
t=0
curr_stride = 1
max_stride = 1000

rand_stride = RANDOM*(max_stride – 1)

DO WHILE (t .LE. tmax)

  Perform quasi-static Time/Space Iteration

  if (curr_stride .eq. rand_stride) then

    Determine the dt required for RKF1/2
    Determine the dt required for RKF2/3
    Determine the dt required for RKF3/4
    Determine the dt required for RKF4/5

    RKF Order = RKF with MAX dt from RKF1/2, RKF2/3, RKF3/4 & RKF4/5


  ! Obtain next random stride
    rand_stride = RANDOM*(max_stride – 1)

  ! Reset the current stride counter
    curr_stride = 1

  endif

  curr_stride = curr_stride + 1
  t=t+dt

ENDDO

Print Results
```

Figure 24: Pseudo-code of ARKTRAN-TD's adaptive RKF expansion order scheme.

6.2 Implementing the Adaptive RKF Scheme on Problem 16-A1 Benchmark

With the adaptive RKF expansion order implemented in ARKTRAN-TD, benchmark Problem 16-A1 (Section 5.2) was rerun using the adaptive scheme. Additionally, all four RKF expansion orders were locked to obtain computation times and power profiles as points of comparison. Results from this trial are shown in Table 11. The delayed supercritical transient in Problem 16-A1 is relatively stable after the initial prompt jump, allowing for all four RKF pairs to calculate a solution in a reasonable timeframe (compared to each other) and obtain the same power level at the simulation stop time of t = 1.0 second. Use of the adaptive RKF order provided a speedup of 10.37% over the originally implemented RKF4/5 calculation. The adaptive RKF order also outperformed the RKF1/2 and RKF2/3 methods, yielding speedups of 35.48% faster than RKF1/2 and 7.75% faster than RKF2/3. Interestingly, the RKF3/4 pair saw a 16.91% speedup over the adaptive RKF scheme. While the time steps from RKF3/4 were much smaller at the transient onset (compared with the steps taken by the adaptive RKF), over time the smaller amount of function evaluations compared to the adaptive RKF scheme allowed for a faster computation time with RKF3/4 locked.

Table 11: Results of adaptive RKF orders in ARKTRAN-TD for Problem 16-A1.

Solution Method	Residual Tolerance	Final Power	CPU Time [hr]
RKF1/2	10^{-3}	1.8797	114.55
RKF2/3	10^{-3}	1.8797	91.10
RKF3/4	10^{-3}	1.8797	72.32
RKF4/5	10^{-3}	1.8797	93.32
Adaptive RKF Order	10^{-3}	1.8797	84.55

The results in Table 11 verify the successful implementation of the adaptive RKF scheme and its ability to provide accurate solutions while decreasing computation time. Though locking the RKF3/4 scheme provides a shorter computation time in this case, without any prior user knowledge of the behavior of the system, the adaptive RKF scheme outperformed the other locked RKF pairs while providing a solution that maintained the accuracy across the suite of simulations. Figure 25 depicts the normalized power as a function of time for each locked RKF pair solution and the adaptive RKF solution. All of the power levels as a function of time calculated using the various RKF pairs available in ARKTRAN-TD and the adaptive RKF solution overlap and provide no visible discrepancies. The maximum relative error between the adaptive RKF scheme and any locked RKF pair is 0.28%.

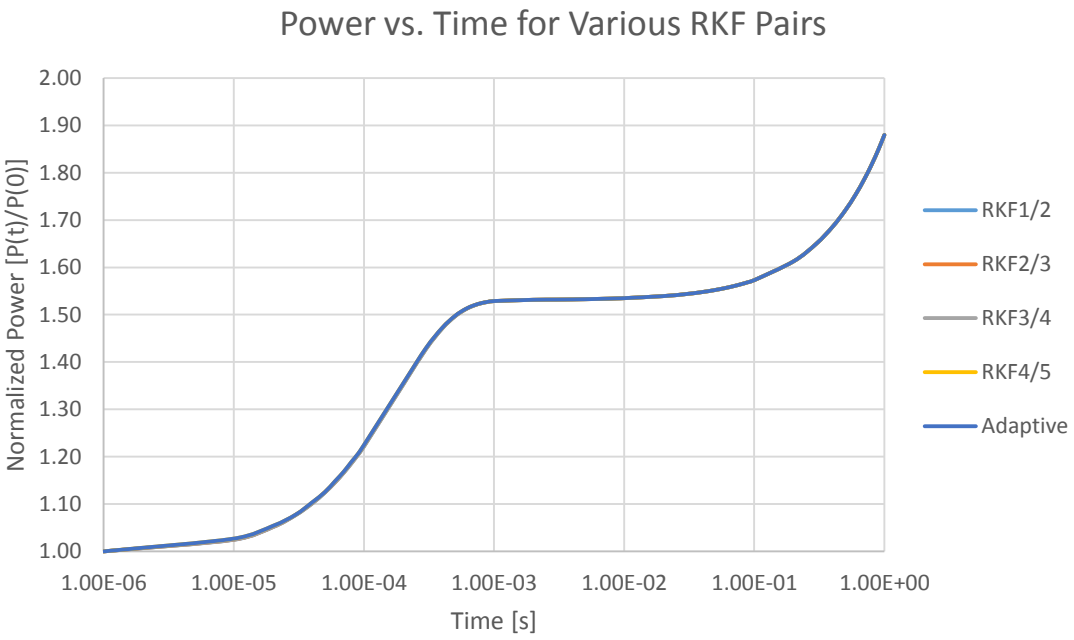

Figure 25: Power profiles for Problem 16-A1 using various RKF pairs.

6.3 Implementing the Adaptive RKF Scheme on Problem 16-A2 Benchmark

The prompt supercritical transient benchmark problem presented in Section 5.3 (Problem 16-A2) was also analyzed using a variety of RKF expansion pairs and residual tolerance values. Results from the suite of solution methods are shown in Table 12, for any locked RKF expansion pair which provided a complete solution to the benchmark in a reasonable amount of computation time. Recalling the discussion of the original results for Problem 16-A2, the residual tolerance was increased from 10^{-3} to 10^{-2} at $t > 8x10^{-4}$ seconds, as the RKF4/5 pair required time steps sizes which were too short for practical computation time. Interestingly, looking at the RKF expansion order selected by the adaptive RKF scheme for the calculation with a residual tolerance of 10^{-3}, the adaptive scheme ran RKF4/5 until $t = 8.7281x10^{-04}$ and then proceeded to run RKF2/3 for the remainder of the problem. The adaptive RKF scheme exhibited a speedup of 3.47 times faster than the locked RKF2/3 scheme when the residual tolerance was set to 10^{-3}.

Table 12: Results of adaptive RKF orders in ARKTRAN-TD for Problem 16-A2.

Solution Method	Residual Tolerance	Final Power	CPU Time [hr]
RKF2/3	10^{-3}	76.5768	2.88
Adaptive RKF Order	10^{-3}	78.0749	0.83
RKF2/3	10^{-2}	77.2303	0.95
RKF4/5	10^{-2}	79.6950	0.17
Adaptive RKF Order	10^{-2}	79.6957	0.08

When the residual tolerance was decreased to 10^{-2}, two of the locked RKF pairs provided complete solutions within a reasonable amount of time. The adaptive RKF scheme displayed a speedup of 2.125 times the locked RKF4/5. Additionally, the

adaptive RKF scheme exhibited a speedup of 11.875 times the locked RKF2/3 solution. Figure 26 depicts the total power as a function of time for each of the solution methods presented in Table 12. All solutions provide results with relative errors of 2% or less from the basis solution (taken to be the adaptive RKF scheme with residual tolerance of 10^{-3}). One would expect the solutions with lower residual tolerances to be more accurate, as the residual tolerance is directly proportional to the truncation error of the Runge-Kutta expansion. Overall, the newly implemented adaptive RKF expansion order scheme coupled with the error controlled time steps showed excellent speedup when compared to the basis locked RKF expansion orders for this case.

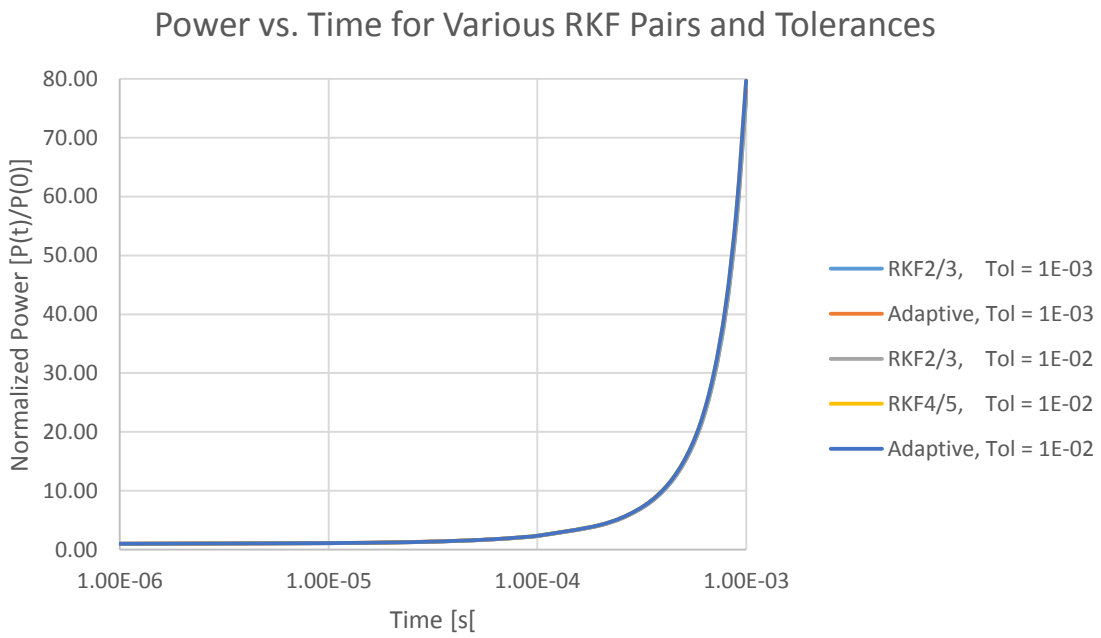

Figure 26: Power profiles for Problem 16-A2 for various RKF pairs and tolerances.

6.4 The Adaptive RKF Scheme Applied to the Problem 16-A3 Benchmark

Originally, the Problem 16-A3 benchmark was evaluated using a locked RKF2/3 expansion in the time-domain with a maximum truncation error on the residual of 10^{-3}. Simulations were run using RKF1/2, RKF2/3, RKF3/4, and RKF4/5 expansion orders, as well as the new adaptive RKF scheme with a residual tolerance of 10^{-3} for all simulations. As expected, the RKF4/5 scheme did not run in a reasonable amount of time. Surprisingly, neither the locked RKF3/4 nor locked RKF1/2 pairs exhibited run times that were reasonable. Thus, the adaptive RKF scheme replicated the simulation using locked RKF2/3, presented in Section 5.4. The total computation time required by ARCTRAN-TD was 27.55 hours to calculate the result of the control rod bank ejection and insertion from t = 0.00 seconds to t = 0.01 seconds.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

In this research work, a new method has been developed to solve the time-dependent form of the linear Boltzmann transport equation which implements a Runge-Kutta-Fehlberg expansion in the time domain with automated truncation error controlled time step selection, coupled with a Source Iteration to resolve the spatial domain of the angular flux. The method was implemented in the 1-D Adaptive Runge-Kutta Time-Dependent Transport code (ARKTRAN-TD), which was developed as part of this dissertation and solves the multigroup form of the time-dependent discrete ordinates transport equation.

ARKTRAN-TD was designed to solve reactor transient calculations in which the reactor system is assumed to be at stable-critical power before the transient commences. Thus, a steady-state solver for the multigroup S_N transport equation was incorporated into ARKTRAN-TD. The steady-state solver was benchmarked against a set of fixed source and eigenvalue problems. Results of these benchmark problems displayed agreement between ARKTRAN-TD and the 3-D S_N code PENTRAN (which is fully benchmarked for steady-state problems in the literature), verifying the ability of ARKTRAN-TD to accurately model the initial conditions for transient calculations.

The newly developed ARKTRAN-TD was validated against a set of fast reactor transient benchmark problems, presented in the *ANL Benchmark Book*. The results (displayed in detail in Chapter 5) depict agreement between ARKTRAN-TD and two other time-dependent transport codes for three transient conditions: delayed supercritical, prompt supercritical, and control rod bank ejection followed by rod bank insertion. ARKTRAN-TD's ability to successfully provide comparable results for these three transient cases verified the use of the fully explicit solution (both in angular flux and neutron precursor concentrations) obtained using the RKF expansion method with error controlled step sizes to expand the time domain, coupled to a quasi-static source iteration which resolved the space derivative.

ARKTRAN-TD was extended further to allow for automatically adaptive Runge-Kutta-Fehlberg expansion pair orders to provide a solution that maintains the accuracy in the truncation error of the expansion while providing solutions in a reduced amount of computation time. For the delayed supercritical transient, the adaptive RKF expansion pair scheme provided a speedup of 1.10 times the originally implemented RKF4/5 solver. The prompt supercritical transient problem displayed an even higher speedup of 2.125 times the RKF4/5 solver when the residual tolerance was 10^{-2}, and 3.47 times the RKF2/3 solver when the residual tolerance was 10^{-3} when the adaptive RKF expansion pair scheme was implemented.

In regards to future work, it would be interesting to explore additional methods to increase computational efficiency even further than the success gained with the adaptive

RKF scheme. Furthermore, the incorporation of additional differencing schemes such as directional theta weighted (DTW) or exponential directional iterative (EDI) would provide for increased accuracy of the spatial sweep compared to Diamond Difference. Inclusion of these schemes could allow for a reduction in computation time, as fewer fine meshes may be required to resolve the solution. Lastly, given the current trend in computational methods for nuclear applications, expanding ARKTRAN-TD to 3-D or including the method in an already established 3-D transport code, and coupling this to an accurate thermal hydraulics code could provide the potential to incorporate a detailed and representative feedback model during the transient.

APPENDIX A

1-D QUADRATURE SETS FROM S_2 TO S_{20}

S_2 Quadrature Set

Angle μ_n	Weight w_n
± 0.577350269189626	1.000000000000000

S_4 Quadrature Set

Angle μ_n	Weight w_n
± 0.339981043584856	0.652145154862546
± 0.861136311594053	0.347854845137454

S_6 Quadrature Set

Angle μ_n	Weight w_n
± 0.238619186083197	0.467913934572691
± 0.661209386466265	0.360761573048139
± 0.932469514203152	0.171324492379170

S_8 Quadrature Set

Angle μ_n	Weight w_n
± 0.183434642495650	0.362683783378362
± 0.525532409916329	0.313706645877887
± 0.796666477413627	0.222381034453374
± 0.960289856497536	0.101228536290376

S_{10} Quadrature Set

Angle μ_n	Weight w_n
± 0.1488743389816312	0.2955242247147529
± 0.4333953941292472	0.2692667193099963
± 0.6794095682990244	0.2190863625159820
± 0.8650633666889845	0.1494513491505806
± 0.9739065285171717	0.0666713443086881

S_{12} Quadrature Set

Angle μ_n	Weight w_n
± 0.1252334085114689	0.2491470458134028
± 0.3678314989981802	0.2334925365383548
± 0.5873179542866175	0.2031674267230659
± 0.7699026741943047	0.1600783285433462
± 0.9041172563704749	0.1069393259953184
± 0.9815606342467192	0.0471753363865118

S_{14} Quadrature Set

Angle μ_n	Weight w_n
± 0.1080549487073437	0.2152638534631578
± 0.3191123689278897	0.2051984637212956
± 0.5152486363581541	0.1855383974779378
± 0.6872929048116855	0.1572031671581935
± 0.8272013150697650	0.1215185706879032
± 0.9284348836635735	0.0801580871597602
± 0.9862838086968123	0.0351194603317519

S_{16} Quadrature Set

Angle μ_n	Weight w_n
± 0.0950125098376374	0.1894506104550685
± 0.2816035507792589	0.1826034150449236
± 0.4580167776572274	0.1691565193950025
± 0.6178762444026438	0.1495959888165767
± 0.7554044083550030	0.1246289712555339
± 0.8656312023878318	0.0951585116824928
± 0.9445750230732326	0.0622535239386479
± 0.9894009349916499	0.0271524594117541

S_{18} Quadrature Set

Angle μ_n	Weight w_n
± 0.0847750130417353	0.1691423829631436
± 0.2518862256915055	0.1642764837458327
± 0.4117511614628426	0.1546846751262652
± 0.5597708310739475	0.1406429146706507
± 0.6916870430603532	0.1225552067114785
± 0.8037049589725231	0.1009420441062872
± 0.8926024664975557	0.0764257302548891
± 0.9558239495713977	0.0497145488949698
± 0.9915651684209309	0.0216160135264833

S_{20} Quadrature Set	
Angle μ_n	**Weight w_n**
± 0.0765265211334973	0.1527533871307258
± 0.2277858511416451	0.1491729864726037
± 0.3737060887154195	0.1420961093183820
± 0.5108670019508271	0.1316886384491766
± 0.6360536807265150	0.1181945319615184
± 0.7463319064601508	0.1019301198172404
± 0.8391169718222188	0.0832767415767048
± 0.9122344282513259	0.0626720483341091
± 0.9639719272779138	0.0406014298003869
± 0.9931285991850949	0.0176140071391521

APPENDIX B

ADDITIONAL RKF EXPANSION ORDER PAIRS

Additional expansion order pairs for the Runge-Kutta-Fehlberg as derived in Fehlberg's original reference and implemented in ARKTRAN-TD's adaptive RKF scheme. (Fehlberg 1969) Refer to Section 2.2 for a complete explanation of the RKF method.

1st Order RKF with 2nd Order Truncation Error

Applying the Runge-Kutta method with local truncation error of order two, one obtains:

$$\widehat{y_{i+1}} = y_i + \frac{1}{512}k_1 + \frac{255}{256}k_2 + \frac{1}{512}k_3 \tag{B.1}$$

which can be used to estimate the local truncation error in the Runge-Kutta method of order one, at the *i+1* time step (Eq. B.2).

$$y_{i+1} = y_i + \frac{1}{256}k_1 + \frac{255}{256}k_2 \tag{B.2}$$

The coefficient terms present in Eqs. B.1 and B.2 for RKF1/2 are given by B.3, where *h* is the time step size:

$$\begin{aligned} k_1 &= hf(t_i, y_i) \\ k_2 &= hf\left(t_i + \frac{h}{2}, y_i + \frac{1}{2}k_1\right) \\ k_3 &= hf\left(t_i + h, y_i + \frac{1}{256}k_1 + \frac{255}{256}k_2\right) \end{aligned} \tag{B.3}$$

The local truncation error (TE) to the 1st order method (Eq. B.2) can be obtained by subtracting Eq. B.2 from Eq. B.1, resulting in the following truncation error in the 1st order method:

$$TE_{RKF1/2} = -\frac{1}{512}k_1 + \frac{1}{512}k_3 \,. \tag{B.4}$$

2rd Order RKF with 3th Order Truncation Error

Applying the Runge-Kutta method with local truncation error of order three, one obtains:

$$\widehat{y_{i+1}} = y_i + \frac{533}{2106}k_1 + \frac{800}{1053}k_3 - \frac{1}{78}k_4 \tag{B.5}$$

which can be used to estimate the local truncation error in the Runge-Kutta method of order two, at the *i+1* time step (Eq. B.6).

$$y_{i+1} = y_i + \frac{214}{891}k_1 + \frac{1}{33}k_2 + \frac{650}{891}k_3 \tag{B.6}$$

The coefficient terms present in Eqs. B.5 and B.6 for RKF2/3 are given by B.7, where *h* is the time step size:

$$\begin{aligned} k_1 &= hf(t_i, y_i) \\ k_2 &= hf\left(t_i + \frac{h}{4}, y_i + \frac{1}{4}k_1\right) \\ k_3 &= hf\left(t_i + \frac{27h}{40}, y_i - \frac{189}{800}k_1 + \frac{729}{800}k_2\right) \\ k_4 &= hf\left(t_i + h, y_i + \frac{214}{891}k_1 + \frac{1}{33}k_2 + \frac{650}{891}k_3\right) \end{aligned} \tag{B.7}$$

The local truncation error (TE) of the 2nd order method (Eq. B.6) can be obtained by subtracting Eq. B.6 from Eq. B.5, resulting in the following truncation error in the 2nd order method:

$$TE_{RKF2/3} = \frac{23}{1782}k_1 - \frac{1}{33}k_2 + \frac{350}{11583}k_3 - \frac{1}{78}k_4 \,. \tag{B.8}$$

3rd Order RKF with 4th Order Truncation Error

Applying the Runge-Kutta method with local truncation error of order four, one obtains:

$$\widehat{y_{i+1}} = y_i + \frac{229}{1470}k_1 + \frac{1125}{1813}k_3 + \frac{13718}{81585}k_4 + \frac{1}{18}k_5 \tag{B.9}$$

which can be used to estimate the local truncation error in the Runge-Kutta method of order three, at the *i+1* time step (Eq. B.10).

$$y_{i+1} = y_i + \frac{79}{490}k_1 + \frac{2175}{3626}k_3 + \frac{2166}{9065}k_4 \tag{B.10}$$

The coefficient terms present in Eqs. B.9 and B.10 for RKF3/4 are given by B.11, where *h* is the time step size:

$$\begin{aligned}
k_1 &= hf(t_i, y_i) \\
k_2 &= hf\left(t_i + \frac{2h}{7}, y_i + \frac{2}{7}k_1\right) \\
k_3 &= hf\left(t_i + \frac{7h}{15}, y_i + \frac{77}{900}k_1 + \frac{343}{900}k_2\right) \\
k_4 &= hf\left(t_i + \frac{35h}{38}, y_i + \frac{805}{1444}k_1 - \frac{77175}{54872}k_2 + \frac{97125}{54872}k_3\right) \\
k_5 &= hf\left(t_i + h, y_i + \frac{79}{490}k_1 + \frac{2175}{3626}k_3 + \frac{2166}{9065}k_4\right)
\end{aligned} \tag{B.11}$$

The local truncation error (TE) to the 3rd order method (Eq. B.10) can be obtained by subtracting Eq. B.10 from Eq. B.9, resulting in the following truncation error in the 3rd order method:

$$TE_{RKF3/4} = -\frac{4}{735}k_1 + \frac{75}{3626}k_3 - \frac{5776}{81585}k_4 + \frac{1}{18}k_5 \,. \tag{B.12}$$

APPENDIX C

ARKTRAN-TD TABULAR RESULTS TO PROBLEM 16-A1

Problem 16-A1 Power versus Time

Time [s]	Normalized Power
1.00000E-06	1.00000E+00
1.07426E-05	1.02757E+00
2.20691E-05	1.05808E+00
3.39367E-05	1.08821E+00
4.58043E-05	1.11642E+00
5.76906E-05	1.14287E+00
7.01127E-05	1.16871E+00
8.19802E-05	1.19167E+00
9.38478E-05	1.21317E+00
1.06191E-04	1.23409E+00
2.11391E-04	1.36482E+00
3.06940E-04	1.43288E+00
4.02488E-04	1.47304E+00
5.11687E-04	1.49922E+00
6.07237E-04	1.51220E+00
7.02785E-04	1.51987E+00
8.11983E-04	1.52490E+00
9.07531E-04	1.52741E+00
1.00308E-03	1.52891E+00
2.01316E-03	1.53145E+00
3.00960E-03	1.53190E+00
4.00603E-03	1.53233E+00
5.00247E-03	1.53277E+00
6.01255E-03	1.53321E+00
7.00899E-03	1.53365E+00
8.00542E-03	1.53409E+00
9.00185E-03	1.53452E+00
1.00119E-02	1.53496E+00
2.00036E-02	1.53931E+00
3.00089E-02	1.54363E+00
4.00005E-02	1.54791E+00
5.00058E-02	1.55217E+00
6.00111E-02	1.55639E+00
7.00027E-02	1.56058E+00
8.00080E-02	1.56474E+00
9.00133E-02	1.56888E+00
1.00005E-01	1.57298E+00
2.00003E-01	1.61260E+00
3.00002E-01	1.65006E+00
4.00000E-01	1.68580E+00
5.00012E-01	1.72019E+00
6.00010E-01	1.75351E+00
7.00008E-01	1.78596E+00
8.00007E-01	1.81772E+00
9.00005E-01	1.84893E+00
1.00000E-00	1.87970E+00

Problem 16-A1 Neutron Flux at t = 0.0 seconds

Z Position	Group 1 Flux	Group 2 Flux	Z Position	Group 1 Flux	Group 2 Flux
1.00000E+00	1.34732E-02	4.93206E-03	1.14437E+02	9.24524E-01	1.36503E-01
3.00000E+00	2.20497E-02	9.14614E-03	1.16562E+02	9.20019E-01	1.35795E-01
5.00000E+00	3.05178E-02	1.31787E-02	1.18687E+02	9.10990E-01	1.34381E-01
7.00000E+00	3.93677E-02	1.73107E-02	1.20812E+02	8.97371E-01	1.32262E-01
9.00000E+00	4.88789E-02	2.16154E-02	1.22937E+02	8.78963E-01	1.29438E-01
1.10000E+01	5.92683E-02	2.61433E-02	1.25062E+02	8.55259E-01	1.25909E-01
1.30000E+01	7.07409E-02	3.09389E-02	1.27187E+02	8.25013E-01	1.21632E-01
1.50000E+01	8.35142E-02	3.60406E-02	1.29311E+02	7.85291E-01	1.16200E-01
1.70000E+01	9.78249E-02	4.14794E-02	1.31274E+02	7.53592E-01	1.11879E-01
1.90000E+01	1.13939E-01	4.72755E-02	1.33074E+02	7.40355E-01	1.10088E-01
2.10000E+01	1.32153E-01	5.34350E-02	1.34874E+02	7.36209E-01	1.09548E-01
2.30000E+01	1.52806E-01	5.99457E-02	1.36674E+02	7.40914E-01	1.10194E-01
2.50000E+01	1.76281E-01	6.67688E-02	1.38474E+02	7.54720E-01	1.12093E-01
2.70000E+01	2.03017E-01	7.38312E-02	1.40361E+02	7.85904E-01	1.16395E-01
2.90000E+01	2.33520E-01	8.10145E-02	1.42335E+02	8.24676E-01	1.21761E-01
3.10000E+01	2.68383E-01	8.81384E-02	1.44309E+02	8.54969E-01	1.26096E-01
3.30000E+01	3.08328E-01	9.49422E-02	1.46283E+02	8.79338E-01	1.29775E-01
3.50000E+01	3.54314E-01	1.01056E-01	1.48257E+02	8.98933E-01	1.32853E-01
3.70000E+01	4.07801E-01	1.05926E-01	1.50231E+02	9.14267E-01	1.35342E-01
3.90000E+01	4.71508E-01	1.08362E-01	1.52204E+02	9.25552E-01	1.37246E-01
4.09870E+01	5.34053E-01	1.08736E-01	1.54178E+02	9.32863E-01	1.38570E-01
4.29609E+01	5.86897E-01	1.09587E-01	1.56152E+02	9.36231E-01	1.39326E-01
4.49348E+01	6.35226E-01	1.11741E-01	1.58126E+02	9.35665E-01	1.39523E-01
4.69087E+01	6.79791E-01	1.14532E-01	1.60100E+02	9.31173E-01	1.39179E-01
4.88826E+01	7.20884E-01	1.17666E-01	1.62074E+02	9.22775E-01	1.38313E-01
5.08565E+01	7.58579E-01	1.20945E-01	1.64048E+02	9.10499E-01	1.36953E-01
5.28304E+01	7.92858E-01	1.24217E-01	1.66022E+02	8.94391E-01	1.35131E-01
5.48044E+01	8.23657E-01	1.27360E-01	1.67996E+02	8.74502E-01	1.32889E-01
5.67783E+01	8.50901E-01	1.30277E-01	1.69970E+02	8.50901E-01	1.30277E-01
5.87522E+01	8.74502E-01	1.32889E-01	1.71944E+02	8.23657E-01	1.27360E-01
6.07261E+01	8.94391E-01	1.35131E-01	1.73918E+02	7.92858E-01	1.24217E-01
6.27000E+01	9.10499E-01	1.36953E-01	1.75892E+02	7.58579E-01	1.20945E-01
6.46739E+01	9.22775E-01	1.38313E-01	1.77865E+02	7.20884E-01	1.17666E-01
6.66479E+01	9.31173E-01	1.39179E-01	1.79839E+02	6.79791E-01	1.14532E-01
6.86218E+01	9.35665E-01	1.39523E-01	1.81813E+02	6.35226E-01	1.11741E-01
7.05957E+01	9.36231E-01	1.39326E-01	1.83787E+02	5.86897E-01	1.09587E-01
7.25696E+01	9.32863E-01	1.38570E-01	1.85761E+02	5.34053E-01	1.08736E-01
7.45435E+01	9.25552E-01	1.37246E-01	1.87748E+02	4.71508E-01	1.08362E-01
7.65174E+01	9.14267E-01	1.35342E-01	1.89748E+02	4.07801E-01	1.05926E-01
7.84913E+01	8.98933E-01	1.32853E-01	1.91748E+02	3.54314E-01	1.01056E-01
8.04653E+01	8.79338E-01	1.29775E-01	1.93748E+02	3.08328E-01	9.49422E-02
8.24392E+01	8.54969E-01	1.26096E-01	1.95748E+02	2.68383E-01	8.81384E-02
8.44131E+01	8.24676E-01	1.21761E-01	1.97748E+02	2.33520E-01	8.10145E-02
8.63870E+01	7.85904E-01	1.16395E-01	1.99748E+02	2.03017E-01	7.38312E-02
8.82740E+01	7.54720E-01	1.12093E-01	2.01748E+02	1.76281E-01	6.67688E-02
9.00740E+01	7.40914E-01	1.10194E-01	2.03748E+02	1.52806E-01	5.99457E-02
9.18740E+01	7.36209E-01	1.09548E-01	2.05748E+02	1.32153E-01	5.34350E-02
9.36740E+01	7.40355E-01	1.10088E-01	2.07748E+02	1.13939E-01	4.72755E-02
9.54740E+01	7.53592E-01	1.11879E-01	2.09748E+02	9.78249E-02	4.14794E-02
9.74365E+01	7.85291E-01	1.16200E-01	2.11748E+02	8.35142E-02	3.60406E-02
9.95615E+01	8.25013E-01	1.21632E-01	2.13748E+02	7.07409E-02	3.09389E-02
1.01687E+02	8.55259E-01	1.25909E-01	2.15748E+02	5.92683E-02	2.61433E-02
1.03812E+02	8.78963E-01	1.29438E-01	2.17748E+02	4.88789E-02	2.16154E-02
1.05937E+02	8.97371E-01	1.32262E-01	2.19748E+02	3.93677E-02	1.73107E-02
1.08062E+02	9.10990E-01	1.34381E-01	2.21748E+02	3.05178E-02	1.31787E-02
1.10187E+02	9.20019E-01	1.35795E-01	2.23748E+02	2.20497E-02	9.14614E-03
1.12312E+02	9.24524E-01	1.36503E-01	2.25748E+02	1.34732E-02	4.93206E-03

Problem 16-A1 Neutron Flux at t = 0.01 seconds

Z Position	Group 1 Flux	Group 2 Flux	Z Position	Group 1 Flux	Group 2 Flux
1.00000E+00	2.29229E-02	8.39194E-03	1.14437E+02	1.40431E+00	2.07412E-01
3.00000E+00	3.75147E-02	1.55624E-02	1.16562E+02	1.38997E+00	2.05239E-01
5.00000E+00	5.19222E-02	2.24239E-02	1.18687E+02	1.36883E+00	2.02004E-01
7.00000E+00	6.69790E-02	2.94548E-02	1.20812E+02	1.34084E+00	1.97713E-01
9.00000E+00	8.31621E-02	3.67795E-02	1.22937E+02	1.30574E+00	1.92377E-01
1.10000E+01	1.00839E-01	4.44843E-02	1.25062E+02	1.26286E+00	1.86000E-01
1.30000E+01	1.20360E-01	5.26449E-02	1.27187E+02	1.21043E+00	1.78527E-01
1.50000E+01	1.42093E-01	6.13263E-02	1.29311E+02	1.14426E+00	1.69373E-01
1.70000E+01	1.66444E-01	7.05816E-02	1.31274E+02	1.09219E+00	1.62194E-01
1.90000E+01	1.93862E-01	8.04449E-02	1.33074E+02	1.06931E+00	1.59048E-01
2.10000E+01	2.24855E-01	9.09275E-02	1.34874E+02	1.05958E+00	1.57712E-01
2.30000E+01	2.59999E-01	1.02007E-01	1.36674E+02	1.06259E+00	1.58085E-01
2.50000E+01	2.99945E-01	1.13620E-01	1.38474E+02	1.07863E+00	1.60253E-01
2.70000E+01	3.45441E-01	1.25640E-01	1.40361E+02	1.11618E+00	1.65382E-01
2.90000E+01	3.97348E-01	1.37866E-01	1.42335E+02	1.16185E+00	1.71632E-01
3.10000E+01	4.56671E-01	1.49993E-01	1.44309E+02	1.19637E+00	1.76522E-01
3.30000E+01	5.24649E-01	1.61576E-01	1.46283E+02	1.22308E+00	1.80560E-01
3.50000E+01	6.02903E-01	1.71985E-01	1.48257E+02	1.24358E+00	1.83840E-01
3.70000E+01	6.93922E-01	1.80280E-01	1.50231E+02	1.25863E+00	1.86384E-01
3.90000E+01	8.02328E-01	1.84435E-01	1.52204E+02	1.26858E+00	1.88203E-01
4.09870E+01	9.11002E-01	1.85120E-01	1.54178E+02	1.27359E+00	1.89309E-01
4.29609E+01	1.00505E+00	1.86768E-01	1.56152E+02	1.27372E+00	1.89720E-01
4.49348E+01	1.09075E+00	1.90774E-01	1.58126E+02	1.26902E+00	1.89455E-01
4.69087E+01	1.16952E+00	1.95897E-01	1.60100E+02	1.25952E+00	1.88538E-01
4.88826E+01	1.24185E+00	2.01595E-01	1.62074E+02	1.24526E+00	1.86998E-01
5.08565E+01	1.30783E+00	2.07503E-01	1.64048E+02	1.22628E+00	1.84873E-01
5.28304E+01	1.36742E+00	2.13336E-01	1.66022E+02	1.20266E+00	1.82207E-01
5.48044E+01	1.42048E+00	2.18872E-01	1.67996E+02	1.17444E+00	1.79056E-01
5.67783E+01	1.46683E+00	2.23930E-01	1.69970E+02	1.14173E+00	1.75485E-01
5.87522E+01	1.50634E+00	2.28371E-01	1.71944E+02	1.10461E+00	1.71579E-01
6.07261E+01	1.53885E+00	2.32078E-01	1.73918E+02	1.06317E+00	1.67437E-01
6.27000E+01	1.56426E+00	2.34963E-01	1.75892E+02	1.01751E+00	1.63182E-01
6.46739E+01	1.58245E+00	2.36954E-01	1.77865E+02	9.67686E-01	1.58965E-01
6.66479E+01	1.59338E+00	2.37996E-01	1.79839E+02	9.13702E-01	1.54974E-01
6.86218E+01	1.59699E+00	2.38045E-01	1.81813E+02	8.55434E-01	1.51454E-01
7.05957E+01	1.59327E+00	2.37069E-01	1.83787E+02	7.92502E-01	1.48775E-01
7.25696E+01	1.58224E+00	2.35046E-01	1.85761E+02	7.23974E-01	1.47759E-01
7.45435E+01	1.56390E+00	2.31959E-01	1.87748E+02	6.40775E-01	1.47276E-01
7.65174E+01	1.53827E+00	2.27797E-01	1.89748E+02	5.54195E-01	1.43962E-01
7.84913E+01	1.50527E+00	2.22556E-01	1.91748E+02	4.81506E-01	1.37341E-01
8.04653E+01	1.46462E+00	2.16234E-01	1.93748E+02	4.19011E-01	1.29031E-01
8.24392E+01	1.41555E+00	2.08825E-01	1.95748E+02	3.64721E-01	1.19783E-01
8.44131E+01	1.35614E+00	2.00253E-01	1.97748E+02	3.17344E-01	1.10100E-01
8.63870E+01	1.28193E+00	1.89880E-01	1.99748E+02	2.75889E-01	1.00337E-01
8.82740E+01	1.22340E+00	1.81733E-01	2.01748E+02	2.39555E-01	9.07381E-02
9.00740E+01	1.19709E+00	1.78070E-01	2.03748E+02	2.07652E-01	8.14653E-02
9.18740E+01	1.18550E+00	1.76434E-01	2.05748E+02	1.79585E-01	7.26171E-02
9.36740E+01	1.18817E+00	1.76708E-01	2.07748E+02	1.54832E-01	6.42459E-02
9.54740E+01	1.20540E+00	1.78987E-01	2.09748E+02	1.32934E-01	5.63690E-02
9.74365E+01	1.24956E+00	1.84934E-01	2.11748E+02	1.13487E-01	4.89776E-02
9.95615E+01	1.30397E+00	1.92277E-01	2.13748E+02	9.61288E-02	4.20442E-02
1.01687E+02	1.34329E+00	1.97783E-01	2.15748E+02	8.05380E-02	3.55273E-02
1.03812E+02	1.37229E+00	2.02115E-01	2.17748E+02	6.64201E-02	2.93739E-02
1.05937E+02	1.39301E+00	2.05347E-01	2.19748E+02	5.34950E-02	2.35241E-02
1.08062E+02	1.40630E+00	2.07488E-01	2.21748E+02	4.14694E-02	1.79089E-02
1.10187E+02	1.41253E+00	2.08542E-01	2.23748E+02	2.99624E-02	1.24289E-02
1.12312E+02	1.41185E+00	2.08516E-01	2.25748E+02	1.83082E-02	6.70228E-03

Problem 16-A1 Neutron Flux at t = 1.0 second

Z Position	Group 1 Flux	Group 2 Flux	Z Position	Group 1 Flux	Group 2 Flux
1.00000E+00	2.81042E-02	1.02888E-02	1.14437E+02	1.71950E+00	2.53966E-01
3.00000E+00	4.59942E-02	1.90798E-02	1.16562E+02	1.70184E+00	2.51291E-01
5.00000E+00	6.36582E-02	2.74923E-02	1.18687E+02	1.67586E+00	2.47315E-01
7.00000E+00	8.21186E-02	3.61125E-02	1.20812E+02	1.64148E+00	2.42049E-01
9.00000E+00	1.01959E-01	4.50926E-02	1.22937E+02	1.59843E+00	2.35502E-01
1.10000E+01	1.23632E-01	5.45388E-02	1.25062E+02	1.54583E+00	2.27680E-01
1.30000E+01	1.47565E-01	6.45436E-02	1.27187E+02	1.48154E+00	2.18517E-01
1.50000E+01	1.74211E-01	7.51877E-02	1.29311E+02	1.40045E+00	2.07298E-01
1.70000E+01	2.04065E-01	8.65348E-02	1.31274E+02	1.33664E+00	1.98500E-01
1.90000E+01	2.37680E-01	9.86275E-02	1.33074E+02	1.30860E+00	1.94644E-01
2.10000E+01	2.75679E-01	1.11479E-01	1.34874E+02	1.29664E+00	1.93002E-01
2.30000E+01	3.18766E-01	1.25063E-01	1.36674E+02	1.30027E+00	1.93450E-01
2.50000E+01	3.67742E-01	1.39300E-01	1.38474E+02	1.31986E+00	1.96096E-01
2.70000E+01	4.23520E-01	1.54037E-01	1.40361E+02	1.36573E+00	2.02363E-01
2.90000E+01	4.87158E-01	1.69027E-01	1.42335E+02	1.42151E+00	2.09997E-01
3.10000E+01	5.59891E-01	1.83894E-01	1.44309E+02	1.46366E+00	2.15968E-01
3.30000E+01	6.43231E-01	1.98094E-01	1.46283E+02	1.49625E+00	2.20897E-01
3.50000E+01	7.39174E-01	2.10855E-01	1.48257E+02	1.52126E+00	2.24898E-01
3.70000E+01	8.50763E-01	2.21024E-01	1.50231E+02	1.53960E+00	2.27999E-01
3.90000E+01	9.83673E-01	2.26117E-01	1.52204E+02	1.55170E+00	2.30215E-01
4.09870E+01	1.11691E+00	2.26956E-01	1.54178E+02	1.55776E+00	2.31560E-01
4.29609E+01	1.23221E+00	2.28975E-01	1.56152E+02	1.55786E+00	2.32054E-01
4.49348E+01	1.33728E+00	2.33886E-01	1.58126E+02	1.55206E+00	2.31722E-01
4.69087E+01	1.43384E+00	2.40164E-01	1.60100E+02	1.54040E+00	2.30593E-01
4.88826E+01	1.52250E+00	2.47148E-01	1.62074E+02	1.52291E+00	2.28703E-01
5.08565E+01	1.60339E+00	2.54388E-01	1.64048E+02	1.49966E+00	2.26098E-01
5.28304E+01	1.67643E+00	2.61536E-01	1.66022E+02	1.47073E+00	2.22832E-01
5.48044E+01	1.74145E+00	2.68319E-01	1.67996E+02	1.43619E+00	2.18973E-01
5.67783E+01	1.79825E+00	2.74517E-01	1.69970E+02	1.39616E+00	2.14603E-01
5.87522E+01	1.84666E+00	2.79955E-01	1.71944E+02	1.35074E+00	2.09822E-01
6.07261E+01	1.88648E+00	2.84496E-01	1.73918E+02	1.30005E+00	2.04753E-01
6.27000E+01	1.91759E+00	2.88027E-01	1.75892E+02	1.24420E+00	1.99546E-01
6.46739E+01	1.93985E+00	2.90461E-01	1.77865E+02	1.18326E+00	1.94387E-01
6.66479E+01	1.95320E+00	2.91731E-01	1.79839E+02	1.11724E+00	1.89504E-01
6.86218E+01	1.95757E+00	2.91784E-01	1.81813E+02	1.04598E+00	1.85198E-01
7.05957E+01	1.95296E+00	2.90580E-01	1.83787E+02	9.69026E-01	1.81921E-01
7.25696E+01	1.93938E+00	2.88092E-01	1.85761E+02	8.85233E-01	1.80677E-01
7.45435E+01	1.91684E+00	2.84299E-01	1.87748E+02	7.83501E-01	1.80085E-01
7.65174E+01	1.88536E+00	2.79188E-01	1.89748E+02	6.77638E-01	1.76032E-01
7.84913E+01	1.84484E+00	2.72755E-01	1.91748E+02	5.88756E-01	1.67935E-01
8.04653E+01	1.79494E+00	2.64995E-01	1.93748E+02	5.12339E-01	1.57774E-01
8.24392E+01	1.73472E+00	2.55903E-01	1.95748E+02	4.45958E-01	1.46465E-01
8.44131E+01	1.66182E+00	2.45387E-01	1.97748E+02	3.88028E-01	1.34625E-01
8.63870E+01	1.57079E+00	2.32661E-01	1.99748E+02	3.37340E-01	1.22687E-01
8.82740E+01	1.49899E+00	2.22669E-01	2.01748E+02	2.92911E-01	1.10950E-01
9.00740E+01	1.46671E+00	2.18174E-01	2.03748E+02	2.53902E-01	9.96114E-02
9.18740E+01	1.45247E+00	2.16163E-01	2.05748E+02	2.19584E-01	8.87920E-02
9.36740E+01	1.45568E+00	2.16491E-01	2.07748E+02	1.89318E-01	7.85563E-02
9.54740E+01	1.47675E+00	2.19277E-01	2.09748E+02	1.62543E-01	6.89245E-02
9.74365E+01	1.53077E+00	2.26551E-01	2.11748E+02	1.38763E-01	5.98869E-02
9.95615E+01	1.59731E+00	2.35531E-01	2.13748E+02	1.17539E-01	5.14092E-02
1.01687E+02	1.64537E+00	2.42260E-01	2.15748E+02	9.84760E-02	4.34405E-02
1.03812E+02	1.68079E+00	2.47550E-01	2.17748E+02	8.12135E-02	3.59166E-02
1.05937E+02	1.70606E+00	2.51494E-01	2.19748E+02	6.54097E-02	2.87637E-02
1.08062E+02	1.72224E+00	2.54101E-01	2.21748E+02	5.07056E-02	2.18978E-02
1.10187E+02	1.72977E+00	2.55379E-01	2.23748E+02	3.66358E-02	1.51973E-02
1.12312E+02	1.72883E+00	2.55332E-01	2.25748E+02	2.23859E-02	8.19511E-03

APPENDIX D

ARKTRAN-TD TABULAR RESULTS TO PROBLEM 16-A2

Problem 16-A2 Power versus Time

Time [s]	Normalized Power
1.00000E-06	1.00000E+00
1.73151E-06	1.01544E+00
2.69150E-06	1.02448E+00
3.52905E-06	1.03284E+00
4.34286E-06	1.04127E+00
5.15667E-06	1.04996E+00
5.97048E-06	1.05885E+00
6.82712E-06	1.06840E+00
7.68376E-06	1.07810E+00
8.54040E-06	1.08792E+00
9.39704E-06	1.09784E+00
1.02537E-05	1.10786E+00
2.03292E-05	1.22893E+00
3.03797E-05	1.35407E+00
4.04303E-05	1.48321E+00
5.04808E-05	1.61645E+00
6.05313E-05	1.75393E+00
7.05819E-05	1.89578E+00
8.06016E-05	2.04167E+00
9.05180E-05	2.19033E+00
1.00461E-04	2.34360E+00
1.50628E-04	3.19311E+00
2.00375E-04	4.17538E+00
2.50466E-04	5.32952E+00
3.00337E-04	6.67062E+00
3.50431E-04	8.24279E+00
4.00465E-04	1.00737E+01
4.50073E-04	1.21819E+01
5.00068E-04	1.46584E+01
5.50157E-04	1.75523E+01
6.00063E-04	2.09149E+01
6.50276E-04	2.48614E+01
7.00077E-04	2.94259E+01
7.50140E-04	3.47760E+01
8.00099E-04	4.10026E+01
8.50059E-04	4.82639E+01
9.00342E-04	5.67875E+01
9.50403E-04	6.66960E+01
1.00026E-03	7.82033E+01

Problem 16-A2 Neutron Flux at t = 10^{-4} seconds

Z Position	Group 1 Flux	Group 2 Flux	Z Position	Group 1 Flux	Group 2 Flux
1.00000E+00	3.78496E-02	1.37582E-02	1.14437E+02	2.08629E+00	3.06823E-01
3.00000E+00	6.19439E-02	2.55141E-02	1.16562E+02	2.05411E+00	3.02021E-01
5.00000E+00	8.57368E-02	3.67648E-02	1.18687E+02	2.01202E+00	2.95676E-01
7.00000E+00	1.10605E-01	4.82956E-02	1.20812E+02	1.96001E+00	2.87807E-01
9.00000E+00	1.37338E-01	6.03118E-02	1.22937E+02	1.89778E+00	2.78435E-01
1.10000E+01	1.66542E-01	7.29557E-02	1.25062E+02	1.82442E+00	2.67578E-01
1.30000E+01	1.98799E-01	8.63526E-02	1.27187E+02	1.73750E+00	2.55162E-01
1.50000E+01	2.34718E-01	1.00611E-01	1.29311E+02	1.63118E+00	2.40377E-01
1.70000E+01	2.74969E-01	1.15818E-01	1.31274E+02	1.54847E+00	2.28916E-01
1.90000E+01	3.20300E-01	1.32032E-01	1.33074E+02	1.51072E+00	2.23678E-01
2.10000E+01	3.71547E-01	1.49272E-01	1.34874E+02	1.49156E+00	2.20991E-01
2.30000E+01	4.29668E-01	1.67504E-01	1.36674E+02	1.49035E+00	2.20707E-01
2.50000E+01	4.95743E-01	1.86622E-01	1.38474E+02	1.50740E+00	2.22931E-01
2.70000E+01	5.71010E-01	2.06423E-01	1.40361E+02	1.55032E+00	2.28677E-01
2.90000E+01	6.56893E-01	2.26576E-01	1.42335E+02	1.60123E+00	2.35497E-01
3.10000E+01	7.55067E-01	2.46578E-01	1.44309E+02	1.63792E+00	2.40587E-01
3.30000E+01	8.67572E-01	2.65699E-01	1.46283E+02	1.66462E+00	2.44609E-01
3.50000E+01	9.97101E-01	2.82901E-01	1.48257E+02	1.68349E+00	2.47711E-01
3.70000E+01	1.14778E+00	2.96632E-01	1.50231E+02	1.69559E+00	2.49931E-01
3.90000E+01	1.32724E+00	3.03549E-01	1.52204E+02	1.70148E+00	2.51290E-01
4.09870E+01	1.51087E+00	3.04820E-01	1.54178E+02	1.70141E+00	2.51811E-01
4.29609E+01	1.67343E+00	3.07942E-01	1.56152E+02	1.69552E+00	2.51520E-01
4.49348E+01	1.82118E+00	3.15192E-01	1.58126E+02	1.68392E+00	2.50444E-01
4.69087E+01	1.95659E+00	3.24337E-01	1.60100E+02	1.66665E+00	2.48618E-01
4.88826E+01	2.08050E+00	3.34415E-01	1.62074E+02	1.64379E+00	2.46083E-01
5.08565E+01	2.19305E+00	3.44772E-01	1.64048E+02	1.61541E+00	2.42889E-01
5.28304E+01	2.29405E+00	3.54898E-01	1.66022E+02	1.58159E+00	2.39094E-01
5.48044E+01	2.38323E+00	3.64399E-01	1.67996E+02	1.54242E+00	2.34769E-01
5.67783E+01	2.46030E+00	3.72963E-01	1.69970E+02	1.49799E+00	2.29998E-01
5.87522E+01	2.52496E+00	3.80336E-01	1.71944E+02	1.44843E+00	2.24883E-01
6.07261E+01	2.57696E+00	3.86331E-01	1.73918E+02	1.39382E+00	2.19547E-01
6.27000E+01	2.61608E+00	3.90790E-01	1.75892E+02	1.33427E+00	2.14140E-01
6.46739E+01	2.64217E+00	3.93599E-01	1.77865E+02	1.26983E+00	2.08843E-01
6.66479E+01	2.65513E+00	3.94667E-01	1.79839E+02	1.20047E+00	2.03883E-01
6.86218E+01	2.65490E+00	3.93930E-01	1.81813E+02	1.12603E+00	1.99552E-01
7.05957E+01	2.64150E+00	3.91333E-01	1.83787E+02	1.04600E+00	1.96304E-01
7.25696E+01	2.61498E+00	3.86853E-01	1.85761E+02	9.59272E-01	1.95122E-01
7.45435E+01	2.57545E+00	3.80467E-01	1.87748E+02	8.51076E-01	1.94500E-01
7.65174E+01	2.52297E+00	3.72166E-01	1.89748E+02	7.36003E-01	1.90097E-01
7.84913E+01	2.45755E+00	3.61956E-01	1.91748E+02	6.39385E-01	1.81317E-01
8.04653E+01	2.37884E+00	3.49843E-01	1.93748E+02	5.56329E-01	1.70306E-01
8.24392E+01	2.28573E+00	3.35831E-01	1.95748E+02	4.84187E-01	1.58061E-01
8.44131E+01	2.17515E+00	3.19828E-01	1.97748E+02	4.21236E-01	1.45248E-01
8.63870E+01	2.03952E+00	3.00788E-01	1.99748E+02	3.66164E-01	1.32335E-01
8.82740E+01	1.93443E+00	2.86106E-01	2.01748E+02	3.17901E-01	1.19645E-01
9.00740E+01	1.88708E+00	2.79470E-01	2.03748E+02	2.75532E-01	1.07392E-01
9.18740E+01	1.86296E+00	2.76024E-01	2.05748E+02	2.38263E-01	9.57060E-02
9.36740E+01	1.86124E+00	2.75581E-01	2.07748E+02	2.05399E-01	8.46546E-02
9.54740E+01	1.88235E+00	2.78267E-01	2.09748E+02	1.76332E-01	7.42604E-02
9.74365E+01	1.94168E+00	2.86096E-01	2.11748E+02	1.50520E-01	6.45112E-02
9.95615E+01	2.01329E+00	2.95553E-01	2.13748E+02	1.27486E-01	5.53695E-02
1.01687E+02	2.06154E+00	3.02183E-01	2.15748E+02	1.06801E-01	4.67800E-02
1.03812E+02	2.09395E+00	3.07028E-01	2.17748E+02	8.80728E-02	3.86731E-02
1.05937E+02	2.11379E+00	3.10218E-01	2.19748E+02	7.09300E-02	3.09685E-02
1.08062E+02	2.12247E+00	3.11776E-01	2.21748E+02	5.49821E-02	2.35747E-02
1.10187E+02	2.12061E+00	3.11718E-01	2.23748E+02	3.97241E-02	1.63605E-02
1.12312E+02	2.10849E+00	3.10061E-01	2.25748E+02	2.42726E-02	8.82231E-03

APPENDIX E

ARKTRAN-TD TABULAR RESULTS TO PROBLEM 16-A3

Problem 16-A3 Power versus Time

Time [s]	Normalized Power	Time [s]	Normalized Power
1.00000E-06	1.00000E+00	2.90000E-04	4.22264E+02
1.98419E-06	1.15118E+00	3.00001E-04	3.81887E+02
3.01129E-06	1.25174E+00	3.10002E-04	3.45376E+02
4.01441E-06	1.36110E+00	3.20001E-04	3.12370E+02
4.99647E-06	1.47903E+00	3.30002E-04	2.82519E+02
5.99488E-06	1.61014E+00	3.40003E-04	2.55530E+02
7.00815E-06	1.75527E+00	3.50000E-04	2.31133E+02
8.00544E-06	1.91074E+00	3.60000E-04	2.09068E+02
9.01666E-06	2.08210E+00	3.70001E-04	1.89116E+02
1.00145E-05	2.26580E+00	3.80001E-04	1.71077E+02
2.00015E-05	5.19385E+00	3.90000E-04	1.54766E+02
2.99993E-05	1.17434E+01	4.00003E-04	1.40013E+02
4.00062E-05	2.64144E+01	5.00006E-04	5.16591E+01
5.00037E-05	5.92123E+01	6.00010E-04	1.93849E+01
6.00009E-05	1.32576E+02	7.00009E-04	7.57144E+00
7.00011E-05	2.96551E+02	8.00018E-04	3.26855E+00
8.00003E-05	6.61378E+02	9.00040E-04	1.70446E+00
9.00003E-05	1.47025E+03	1.00006E-03	1.13611E+00
1.00000E-04	3.26814E+03	1.10004E-03	9.29615E-01
1.10000E-04	2.59346E+03	1.20019E-03	8.54503E-01
1.20000E-04	2.33596E+03	1.30027E-03	8.27297E-01
1.30000E-04	2.11198E+03	1.40011E-03	8.17372E-01
1.40000E-04	1.90972E+03	1.50002E-03	8.13729E-01
1.50001E-04	1.72683E+03	1.60081E-03	8.12375E-01
1.60000E-04	1.56149E+03	1.70039E-03	8.11876E-01
1.70001E-04	1.41196E+03	1.80112E-03	8.11678E-01
1.80000E-04	1.27679E+03	1.90070E-03	8.11592E-01
1.90000E-04	1.15455E+03	2.00029E-03	8.11546E-01
2.00001E-04	1.04401E+03	3.00070E-03	8.11302E-01
2.10001E-04	9.44073E+02	4.00112E-03	8.11072E-01
2.20001E-04	8.53706E+02	5.00040E-03	8.10843E-01
2.30001E-04	7.71996E+02	6.00081E-03	8.10614E-01
2.40001E-04	6.98115E+02	7.00009E-03	8.10386E-01
2.50001E-04	6.31313E+02	8.00051E-03	8.10158E-01
2.60002E-04	5.70910E+02	9.00092E-03	8.09930E-01
2.70001E-04	5.16299E+02	1.00000E-02	8.09703E-01
2.80001E-04	4.66912E+02		

Problem 16-A3 Neutron Flux at t = 10^{-4} seconds

Z Position	Group 1 Flux	Group 2 Flux	Z Position	Group 1 Flux	Group 2 Flux
1.00000E+00	3.39199E+01	1.14326E+01	1.14437E+02	3.24212E+03	4.66631E+02
3.00000E+00	5.55229E+01	2.12028E+01	1.16562E+02	3.27633E+03	4.74908E+02
5.00000E+00	7.68779E+01	3.05636E+01	1.18687E+02	3.29626E+03	4.82137E+02
7.00000E+00	9.92293E+01	4.01778E+01	1.20812E+02	3.30138E+03	4.88608E+02
9.00000E+00	1.23293E+02	5.02251E+01	1.22937E+02	3.29070E+03	4.94712E+02
1.10000E+01	1.49627E+02	6.08339E+01	1.25062E+02	3.26230E+03	5.00995E+02
1.30000E+01	1.78763E+02	7.21181E+01	1.27187E+02	3.21187E+03	5.08347E+02
1.50000E+01	2.11265E+02	8.41791E+01	1.29311E+02	3.12989E+03	5.19341E+02
1.70000E+01	2.47752E+02	9.70998E+01	1.31274E+02	3.06437E+03	5.28383E+02
1.90000E+01	2.88914E+02	1.10940E+02	1.33074E+02	3.04325E+03	5.31722E+02
2.10000E+01	3.35531E+02	1.25727E+02	1.34874E+02	3.03959E+03	5.33278E+02
2.30000E+01	3.88490E+02	1.41442E+02	1.36674E+02	3.05317E+03	5.33135E+02
2.50000E+01	4.48795E+02	1.58006E+02	1.38474E+02	3.08429E+03	5.31194E+02
2.70000E+01	5.17601E+02	1.75253E+02	1.40361E+02	3.16685E+03	5.25449E+02
2.90000E+01	5.96235E+02	1.92906E+02	1.42335E+02	3.27495E+03	5.19394E+02
3.10000E+01	6.86258E+02	2.10534E+02	1.44309E+02	3.35465E+03	5.16713E+02
3.30000E+01	7.89575E+02	2.27503E+02	1.46283E+02	3.41408E+03	5.15139E+02
3.50000E+01	9.08687E+02	2.42896E+02	1.48257E+02	3.45710E+03	5.13842E+02
3.70000E+01	1.04738E+03	2.55332E+02	1.50231E+02	3.48553E+03	5.12343E+02
3.90000E+01	1.21264E+03	2.61797E+02	1.52204E+02	3.50024E+03	5.10293E+02
4.09870E+01	1.37516E+03	2.63163E+02	1.54178E+02	3.50162E+03	5.07453E+02
4.29609E+01	1.51343E+03	2.65807E+02	1.56152E+02	3.48996E+03	5.03638E+02
4.49348E+01	1.64118E+03	2.71834E+02	1.58126E+02	3.46541E+03	4.98733E+02
4.69087E+01	1.76031E+03	2.79568E+02	1.60100E+02	3.42814E+03	4.92660E+02
4.88826E+01	1.87157E+03	2.88275E+02	1.62074E+02	3.37833E+03	4.85390E+02
5.08565E+01	1.97517E+03	2.97466E+02	1.64048E+02	3.31616E+03	4.76935E+02
5.28304E+01	2.07107E+03	3.06767E+02	1.66022E+02	3.24187E+03	4.67350E+02
5.48044E+01	2.15911E+03	3.15882E+02	1.67996E+02	3.15567E+03	4.56718E+02
5.67783E+01	2.23911E+03	3.24574E+02	1.69970E+02	3.05784E+03	4.45174E+02
5.87522E+01	2.31084E+03	3.32647E+02	1.71944E+02	2.94871E+03	4.32902E+02
6.07261E+01	2.37410E+03	3.39949E+02	1.73918E+02	2.82854E+03	4.20136E+02
6.27000E+01	2.42870E+03	3.46350E+02	1.75892E+02	2.69763E+03	4.07186E+02
6.46739E+01	2.47448E+03	3.51749E+02	1.77865E+02	2.55617E+03	3.94442E+02
6.66479E+01	2.51128E+03	3.56054E+02	1.79839E+02	2.40424E+03	3.82401E+02
6.86218E+01	2.53899E+03	3.59203E+02	1.81813E+02	2.24155E+03	3.71722E+02
7.05957E+01	2.55750E+03	3.61134E+02	1.83787E+02	2.06709E+03	3.63399E+02
7.25696E+01	2.56672E+03	3.61793E+02	1.85761E+02	1.87824E+03	3.59719E+02
7.45435E+01	2.56653E+03	3.61147E+02	1.87748E+02	1.65625E+03	3.57798E+02
7.65174E+01	2.55678E+03	3.59150E+02	1.89748E+02	1.43054E+03	3.48921E+02
7.84913E+01	2.53710E+03	3.55770E+02	1.91748E+02	1.24111E+03	3.31897E+02
8.04653E+01	2.50673E+03	3.50977E+02	1.93748E+02	1.07842E+03	3.10844E+02
8.24392E+01	2.46401E+03	3.44719E+02	1.95748E+02	9.37309E+02	2.87644E+02
8.44131E+01	2.40528E+03	3.36802E+02	1.97748E+02	8.14350E+02	2.63547E+02
8.63870E+01	2.32267E+03	3.26092E+02	1.99748E+02	7.06950E+02	2.39421E+02
8.82740E+01	2.25450E+03	3.17260E+02	2.01748E+02	6.12975E+02	2.15853E+02
9.00740E+01	2.22951E+03	3.14142E+02	2.03748E+02	5.30610E+02	1.93220E+02
9.18740E+01	2.23214E+03	3.14729E+02	2.05748E+02	4.58277E+02	1.71749E+02
9.36740E+01	2.26190E+03	3.18866E+02	2.07748E+02	3.94608E+02	1.51547E+02
9.54740E+01	2.31978E+03	3.26789E+02	2.09748E+02	3.38386E+02	1.32639E+02
9.74365E+01	2.44663E+03	3.44135E+02	2.11748E+02	2.88552E+02	1.14987E+02
9.95615E+01	2.61100E+03	3.66755E+02	2.13748E+02	2.44159E+02	9.85116E+01
1.01687E+02	2.74770E+03	3.86200E+02	2.15748E+02	2.04364E+02	8.30968E+01
1.03812E+02	2.86553E+03	4.03740E+02	2.17748E+02	1.68397E+02	6.86052E+01
1.05937E+02	2.96813E+03	4.19536E+02	2.19748E+02	1.35530E+02	5.48806E+01
1.08062E+02	3.05688E+03	4.33649E+02	2.21748E+02	1.05002E+02	4.17480E+01
1.10187E+02	3.13217E+03	4.46142E+02	2.23748E+02	7.58344E+01	2.89616E+01
1.12312E+02	3.19397E+03	4.57096E+02	2.25748E+02	4.63285E+01	1.56161E+01

REFERENCES

Alcouffe, R., E. Larsen, W.J. Miller, and B. Wienke. 1979. "Computational Efficiency of Numerical Methods for the Multigroup Discrete Ordinates Neutron Transport Equations: The Slab Geometry Case." *Nuclear Science and Engineering* 111-127.

Argonne National Laboratory. 1985. *National Energy Software Center: Benchmark Book.* ANL-7416 Supplement 3, Argonne, IL: Mathematics and Computation Division of the American Nuclear Society.

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

Burden, R.L., and J.D. Faires. 2010. *Numerical Analysis, Ninth Edition.* Boston, MA: Brooks/Cole.

Douglass, S.J. 2012. *Consistent Energy Treatment for Radiation Transport Methods.* PhD Dissertation, Atlanta, GA: Georgia Institute of Technology, 93-97.

Engle, W.W. 1967. *A Users Manual for ANISI.* K-1693, Oak Ridge, TN: Oak Ridge National Laboratory.

Engle, W.W., and et. al. 1969. "One Dimensional Time Dependent Discrete Ordinates." *Transactions of the American Nuclear Society* 400.

Fehlberg, E. 1969. *Low-Order Classical Runge-Kutta Formulas with Stepsize Control and their Application to Some Heat Transfer Problems.* Washington, D.C.: National Aeronautics and Space Administration, 1-49.

Hill, T.R. 1975. *ONETRAN: A Discrete Ordinates Finite Element Code for the Solution of One-Dimensional Multigroup Transport Equation.* LA-5990-MS, Los Alamos, NM: Los Alamos Scientific Laboratory.

Hill, T.R., and W.M. Reed. 1976. *TIMEX: A Time-Dependent Explicit Discrete Ordinates Program for the Solution of Multigroup Transport Equations with Delayed Neutrons.* LA-6201-MS, Los Alamos, NM: Los Alamos Scientific Laboratory.

James, M.L., G.M. Smith, and J.C. Wolford. 1993. *Applied Numerical Methods for Digital Computation, Fourth Edition.* New York, NY: HarperCollins.

Lewis, E.E., and W.F. Miller. 1993. *Computational Methods of Neutron Transport.* La Grange Park, IL: American Nuclear Society.

Pautz, A., and A. Birkhofer. 2003. "DORT-TD: A Transient Neutron Code with Fully Implicit Time Integration." *Nuclear Science and Engineering* 1-22.

Petrovic, B., and A. Haghighat. 1995. "Boundary Condition Induced Oscillations in the Sn Solutions of the Neutron Transport Equation." *International Conference on Mathematics and Computation, Reactor Physics, and Environmental Analysis.* Portland, OR: American Nuclear Society. 382-391.

Simenov, T. 2003. *Release Notes - HELIOS System Version 1.8.* SSP-03/221, Studsvik Scandpower.

Sjoden, G.E., and A. Haghighat. 2008. *PENTRAN Code System, Users Guide to Version 9.4.X.1 Series.* HSW Technologies, LLC, 1-300.