Improved Quasi-Static Methods for Time-Dependent Neutron Diffusion and Implementation in Rattlesnake

Thesis Proposal

Zachary M. Prince

Department of Nuclear Engineering
Texas A&M University
College Station, TX, 77843-3133
zachmprince@tamu.edu

January 2, 2017

1 Introduction

Transient modeling of nuclear reactors has been a chronically foreboding task due to its computationally expensive nature. However, recent developments in nuclear testing, including the revitalization of the Transient Reactor Testing (TREAT) Facility at Idaho National Laboratory (INL), have brought significant attention to the development of transient reactor modeling. 3-D transient computational methods currently in use at national laboratories, including INL, have proven to be overbearingly computationally expensive for complex, real-world scenarios. Therefore, implementation of more efficient and faster methods for 3-D transient simulations is highly desired. The Department of Energy (DOE), through its NEAMS (Nuclear Energy Advanced Modeling and Simulation) program, has especially sought the development of transient multiphysics capability in INL's MOOSE (Multiphysics Object-Oriented Simulation Environment).

MOOSE is a multiphysics framework being developed at INL that presents the architecture for physics-based applications [5]. Rattlesnake is a large application in MOOSE that involves deterministic radiation transport physics. Currently, Rattlesnake is able to solve steady-state, transient, and k-eigenvalue neutron transport and diffusion problems using finite element methods (FEM) [12].

The difficulty in transient reactor modeling is due to

1. the high-dimensionality of the phase-space for the governing equations that describe the flux of neutrons (6-D+time for multigroup neutron transport and 4-D+time for multigroup neutron diffusion), and

2. the fact that the time discretization has to be implicit, which leads to stiff system of equations.

The transport equation has seven independent variables: space ($\vec{r}$), energy (E), direction ($\vec{\Omega}$), and time (t) [3]. Nuclear reactor simulations often utilize the neutron diffusion approximation, which carefully eliminates the dependence on direction. For the purpose of this research, neutron diffusion will be discussed exclusively.

For computational purposes, the neutron diffusion equation is discretized in space, energy, and time. There are several viable discretization schemes for each of these variables; FEM for space and multigroup in energy are used for the development of this research. Since the purpose of this research involves temporal dependence, discretization in time was kept general. The time dependent neutron diffusion equation with delayed neutron precursors can be seen in Equations 1a and 1b.

$$\begin{aligned}\frac{1}{v^g}\frac{\partial \phi^g}{\partial t} = \frac{\chi_p^g}{k_{\mathit{eff}}}(1-\beta)\sum_{g'=1}^{G}\nu^{g'}\Sigma_f^{g'}\phi^{g'} + \sum_{g'\neq g}^{G}\Sigma_s^{g'\to g}\phi^{g'} \\ + \sum_{i=1}^{I}\chi_{d,i}^g\lambda_i C_i - \left(-\nabla\cdot D^g\nabla + \Sigma_r^g\right)\phi^g \ , \quad 1\leq g\leq G \end{aligned} \tag{1a}$$

$$\frac{dC_i}{dt} = \frac{\beta_i}{k_{\mathit{eff}}}\sum_{g=1}^{G}\nu^g\Sigma_f^g\phi^g - \lambda_i C_i \ , \quad 1\leq i\leq I \tag{1b}$$

where,

ϕ^g	=	Scalar flux in energy group g
C_i	=	Concentration of delayed neutron precursor i
Σ_f^g	=	Fission cross section in energy group g
Σ_r^g	=	Removal cross section in energy group g
$\Sigma_s^{g'\to g}$	=	Scattering cross section from energy group g' to g
v^g	=	Neutron velocity in energy group g
χ_p^g	=	Fission spectrum of prompt neutrons
$\chi_{d,i}^g$	=	Fission spectrum of delayed neutrons from precursor i
ν^g	=	Total number of neutrons per fission
D^g	=	Diffusion coefficient in energy group g
λ_i	=	Decay constant of precursor i
β_i	=	Delayed neutron fraction from precursor i
β	=	Total delayed neutron fraction ($\beta = \sum_{i=1}^{I}\beta_i$)

Most reactor computation frameworks, including Rattlesnake, discretize the time variable directly with these equations using a multitude of schemes (Implicit Euler, Crank-Nicholson, implicit Runge-Kutta, etc.). In this paper, the method of discretizing Equations 1a and 1b is generally referred to as “implicit discretization”. This research intends to improve upon this method by instead implementing the improved quasi-static method (IQS) for neutron kinetics and implement it within a multiphysics setting.

IQS is a spatial kinetics method that involves factorizing the flux solution into space- and time-dependent components [4, 10]. These components are the flux’s amplitude and its shape. Amplitude is only time-dependent, while the shape is both space- and time-dependent. However, the impetus of the method is the assumption that the shape is only weakly dependent on time; therefore, the shape may not require an update at the same frequency as the amplitude function, but only on larger macro-time steps. The proposal for this research contains three sections: Section 2 explains the current theory of IQS and its implementation, Section 3 describes the proposed research in IQS, and Section 4 shows preliminary results of the proposed research.

2 IQS Theory and Status

This section describes the theory of IQS and the current status of its development. The theory and status are broken up into three sections: Section 2.1 analytically derives the method, Section 2.2 and Section 2.3 present two solution techniques and describes their application and development.

2.1 Theory

IQS involves factorizing the flux from Equation 1a into time-dependent amplitude (p) and space- and time- dependent shape (φ). The resulting equation is the shape-diffusion equation with precursors represented by Equations 2a and 2b.

$$\begin{aligned}\frac{1}{v^g}\frac{\partial \varphi^g}{\partial t} =& \frac{\chi_p^g}{k_{\mathit{eff}}}(1-\beta)\sum_{g'=1}^{G}\nu^{g'}\Sigma_f^{g'}\varphi^{g'} + \sum_{g'\neq g}^{G}\Sigma_s^{g'\to g}\varphi^{g'} \\ &- \left(-\nabla\cdot D^g\nabla + \Sigma_r^g + \boxed{\frac{1}{v^g}\frac{1}{p}\frac{dp}{dt}}\right)\varphi^g + \boxed{\frac{1}{p}}\sum_{i=1}^{I}\chi_{d,i}^g\lambda_i C_i, \quad 1 \le g \le G\end{aligned} \tag{2a}$$

$$\frac{dC_i}{dt} = \frac{\beta_i}{k_{\mathit{eff}}}\boxed{p}\sum_{g=1}^{G}\nu^g\Sigma_f^g\varphi^g - \lambda_i C_i\ , \quad 1 \le i \le I \tag{2b}$$

We note that the time-dependent shape equation is similar to the time-dependent flux equation, with the following modifications:

1. The shape equation contains an additional term equivalent to a removal cross section, $\frac{1}{v^g}\frac{1}{p}\frac{dp}{dt}$.
2. The delayed neutron source term is divided by p.
3. The system of equations is now nonlinear due to the factorization.
4. An equation is needed to obtain the amplitude p.

To derive the amplitude equation, the shape/precursors equations are weighted by a space-dependent function and integrated over the phase-space. The weight function is typically the adjoint flux ϕ^{*g}, which can be proven to minimize truncation error [3]. The final expressions are given below:

$$\frac{dp}{dt} = \left[\frac{\rho-\bar{\beta}}{\Lambda}\right]p + \sum_{i=1}^{I}\bar{\lambda}_i\xi_i \tag{3a}$$

$$\frac{d\xi_i}{dt} = \frac{\bar{\beta}_i}{\Lambda} - \bar{\lambda}_i \xi_i \quad 1 \leq i \leq I \tag{3b}$$

This equation is also known as the point reactor kinetics equation (PRKE); where the reactivity, effective delayed-neutron fraction, and delayed-neutron precursor decay constant are defined as follows:

$$\frac{\rho - \bar{\beta}}{\Lambda} = \frac{\sum_{g=1}^{G} \left(\phi^{*g}, \frac{\chi_p^g}{k_{\mathit{eff}}} (1-\beta) \sum_{g'=1}^{G} \nu^{g'} \Sigma_f^{g'} \varphi^{g'} + \sum_{g' \neq g}^{G} \Sigma_s^{g' \to g} \varphi^{g'} - \left(-\nabla \cdot D^g \nabla + \Sigma_r^g \right) \varphi^g \right)}{\sum_{g=1}^{G} \left(\phi^{*g}, \frac{1}{v^g} \varphi^g \right)} \tag{4a}$$

$$\frac{\bar{\beta}}{\Lambda} = \sum_{i=1}^{I} \frac{\bar{\beta}_i}{\Lambda} = \sum_{i=1}^{I} \frac{1}{k_{\mathit{eff}}} \frac{\sum_{g=1}^{G} (\phi^{*g}, \chi_{d,i}^g \beta_i \sum_{g'=1}^{G} \nu^{g'} \Sigma_f^{g'} \varphi^{g'})}{\sum_{g=1}^{G} \left(\phi^{*g}, \frac{1}{v^g} \varphi^g \right)} \tag{4b}$$

$$\bar{\lambda}_i = \frac{\sum_{g=1}^{G} (\phi^{*g}, \chi_{d,i}^g \lambda_i C_i)}{\sum_{g=1}^{G} (\phi^{*g}, \chi_{d,i}^g C_i)} \tag{4c}$$

The following inner product definition has been used: $(\phi^{*g}, f) := \int_D \phi^{*g}(\vec{r}) f(\vec{r}) d^3r$.

Additionally, in order to impose uniqueness on the factorization and to derive the PRKE, the following normalization condition is imposed: $\sum_{g=1}^{G} \left(\phi^{*g}, \frac{1}{v^g} \varphi^g \right) = \mathit{constant}$ [4].

Solving for the shape in Equation 2a can become expensive, especially in two or three dimensions, and even more so when using the transport equations in lieu of the diffusion equations. Using IQS, one expects the time dependence of the shape to be weaker than that of the flux itself, thus allowing for larger time step sizes in updating the shape. The PRKE equations form a small ODE system and can be solved using a much smaller time step size. In transients where the shape varies much less than the flux, IQS can thus be very computationally effective. The two-time scale solution process, a micro scale for the PRKE and a macro scale for the shape, is illustrated in Figure 1.

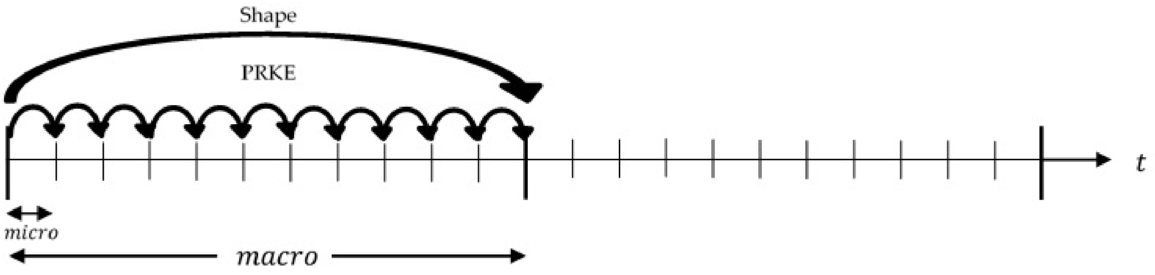

Figure 1: IQS method visualization

2.2 Iterative Solution Techniques

As we noted in Section 2.1, shape-PRKE equations are a nonlinear system and thus may be solved in a iterative manner. Each macro time step can be iterated so the best shape is used to compute power at the micro time steps. Sissaoui et al. from [11], Koclas et al. from [8], Devooght et al. from [2], and Monier from [9] all use iterative techniques for their quasi-static simulations. They all undergo a similar process:

Step 1: Compute the PRKE parameters at the end of the macro step using the last computed shape

Step 2: Linearly interpolate the computed PRKE parameters over the macro step

Step 3: Solve the PRKE on micro steps over the entire macro step

Step 4: Solve the shape equation on the macro step using the computed values of p and dp/dt.

Step 5: Check if the shape solution has converged:

- *No:* Repeat the same macro time step
- *Yes:* Move on to the next macro time step

The major difference between the methods of these authors is the convergence criteria used. Sissaoui and Koclas [8, 11] use fixed point iteration where the criteria is the simply the normalized difference between the last two computed shapes. Monier in [9] also does fixed point iterations with the same criteria, except the solution is scaled by $\frac{\sum_{g=1}^{G}\left(\phi^{*g},\frac{1}{v^g}\varphi^g(t_n)\right)}{\sum_{g=1}^{G}\left(\phi^{*g},\frac{1}{v^g}\varphi^g(t_{n+1})\right)}$ after each iteration. Devooght in [2] does a Newton-SOR iteration where the residual of the shape function evaluation is the convergence criteria and next iteration's solution is computed using Newton-Raphson method.

These techniques are by no means an exhaustive list of the possible iteration techniques for IQS. Dulla et al. in [4] does an in depth analysis of the fixed point iteration technique most similar to Sissaoui and Koclas, involving convergence rates and solution results. However, no comprehensive analysis of iteration techniques exists, comparing both Newton and fixed-point convergence rates.

2.3 Predictor-Corrector version of IQS (IQS P-C)

The Predictor-Corrector (P-C) version of IQS factorizes the flux and derives the PRKE the same way as the standard version, but the solution of the coupled system of equations is different. In the IQS P-C version, the flux equations (not the shape equations) are solved (represented by Equations 1a and 1b) in order to obtain a predicted flux solution. This predicted flux is then converted to a shape by rescaling it as follows:

$$\varphi_{n+1}^{g} = \underbrace{\phi_{n+1}^{g}}_{\text{predicted}} \frac{K_0}{K_{n+1}} \tag{5}$$

where the scaling factors are given by

$$K_{n+1} = \sum_{g=1}^{G}\left(\phi^{*g}, \frac{1}{v^g}\phi_{n+1}^{g}\right) \tag{6}$$

$$K_0 = \sum_{g=1}^{G}\left(\phi^{*g}, \frac{1}{v^g}\varphi_{n+1}^{g}\right) = \sum_{g=1}^{G}\left(\phi^{*g}, \frac{1}{v^g}\phi_{0}^{g}\,.\right) \tag{7}$$

The PRKE parameters are then computed with this shape using Equations 4a - 4c and interpolated over the macro step, then the PRKE ODE system is solved on the micro time scale. With the newly computed amplitude, the shape is rescaled into a flux and the final corrected flux is given by:

$$\underbrace{\phi^g_{n+1}}_{\text{corrected}} = p_{n+1} \times \varphi^g_{n+1} \, . \tag{8}$$

The advantage to the predictor-corrector method is there is no iteration necessary for this method and, in turn, is much simpler and faster than the standard IQS. Ikeda et al. in [7] and Goluoglu et al. in [6] both use IQS P-C for complex, three-dimensional problems. Their results prove IQS P-C to be impressively effective, despite the de-coupling of the system. Dulla et al. in [4] also describes an in depth comparison of IQS P-C with traditional IQS.

3 Proposed Research

The proposed research intends to continue the investigation and development of the improved quasi-static method.

3.1 Investigation of Iteration Convergence

Section 2.2 describes several iterative techniques employed for IQS, with different convergence criteria and solution scaling. A new convergence criteria, not previously attempted, can be applied. This criteria involves checking that the uniqueness criteria described in Section 2.1 is preserved. The error of a certain iteration is described by Equation 9. Where $\varphi^{g,n}$ corresponds to the solution at the most recent time step and $\varphi^{g,0}$ corresponds to the initial solution. This technique is intended to ensure that the solution retains the purity of the analytical derivation.

$$Error_{IQS} = \left| \frac{\sum_{g=1}^{G} \left(\phi^{*g}, \frac{1}{v^g} \varphi^{g,n} \right)}{\sum_{g=1}^{G} \left(\phi^{*g}, \frac{1}{v^g} \varphi^{g,0} \right)} - 1 \right| \tag{9}$$

This research intends to investigate the performance of the previously implemented iterative techniques and compare them to this newly implemented technique.

3.2 Investigation of Various Time Discretization Schemes

A vital part of the verification and validation for IQS is analyzing error convergence. Since any time discretization scheme is capable of being applied to the shape equation of IQS, it is important to investigate IQS's performance to a variety of these schemes. There is lack of literature that applies IQS to schemes other than implicit Euler; higher order schemes are never rigorously tested. This research intends to apply a variety of schemes, including implicit Euler, Crank-Nicholson, backward difference formula (BDF), and diagonally implicit Runge-Kutta (DIRK), to test stability and error convergence.

3.3 Application of Time Adaptation

IQS aims at reducing the time discretization error in the flux solution by splitting the flux into an amplitude (highly resolved at a micro time scale) and a shape (whose time-dependence is weaker than that of the flux itself). Thus, by construction, the IQS approach may employ larger time-step sizes for comparable temporal error. Further enhancements can be gained by using time adaptation (or time step control) in order to increase or reduce the macro time step size for the shape evaluation, depending on error estimates. A step-doubling technique is chosen as the time adaptation technique. The step doubling technique involves estimating the local error for a certain time step by taking the difference between a solution with one full step ($\varphi^g_{\Delta t}$) and a solution with two half steps ($\varphi^g_{\Delta t/2}$). Note: φ is changed to ϕ for implicit discretization and IQS P-C.

The relative error is computed as follows:

$$e_n = \frac{\left\| \sum_{g=1}^{G} \varphi^g_{\Delta t/2} - \sum_{g=1}^{G} \varphi^g_{\Delta t} \right\|_{L^2}}{\max\left(\left\| \sum_{g=1}^{G} \varphi^g_{\Delta t/2} \right\|_{L^2}, \left\| \sum_{g=1}^{G} \varphi^g_{\Delta t} \right\|_{L^2} \right)} \tag{10}$$

If the error is smaller than the user-specified tolerance, e_{tol}, the time step is accepted. In addition, a new time step size is estimated as follows:

$$\Delta t_{new} = S \Delta t \left(\frac{e_{tol}}{e_n} \right)^{\frac{1}{1+q}} \tag{11}$$

Where q is the convergence order of the time integration scheme being used and $S \simeq 0.8$ is a safety factor. If the error is larger than the user-specified tolerance, the time step is rejected. A new time step size is estimated using Equation 10 as well.

To investigate IQS's performance with step-doubling time adaptation, the adaptation will be applied to implicit discretization method, traditional IQS, and IQS P-C. Each of these methods will be applied to several diffusion problems; the number of time steps taken and the resulting error will be used to compare the methods.

3.4 Application of Temperature Feedback

IQS is first and foremost a nuclear reactor simulation method. In nuclear reactors, multiple physics affect the profile of the neutron flux. The most simple example of mulitphysics reactor simulations is adiabatic heat up with Doppler feedback. The principle of Doppler feedback is that fission in a fuel causes the material to increase temperature and induces a change in the neutronics properties. The material heat up is described by Equation 12; where ρ is the material density, c_p is the specific heat, T is temperature, and κ_f is the energy released per fission [1]. The change in temperature of the material mainly affects the thermal macroscopic absorption cross section described by Equation 13 [1].

$$\rho c_p \frac{\partial T(\vec{r},t)}{\partial t} = \kappa_f \sum_{g=1}^{G} \Sigma^g_f \phi^g(\vec{r},t) \tag{12}$$

$$\Sigma_a^{thermal}(\vec{r},t) = \Sigma_a^{thermal}(\vec{r},0)\left[1+\gamma\left(\sqrt{T}-\sqrt{T_0}\right)\right] \tag{13}$$

For IQS, this temperature feedback affects both the shape equation and the reactivity of the PRKE; thus, it is an additional nonlinear component to the already coupled shape-amplitude equations. In foresight to the application of this component, temperature is much more time dependent than the shape, but less so than the amplitude. Therefore, the evaluation of temperature will have its own time scale. A possible solution process for a problem with temperature feedback will have time three time scales portrayed in Figure 2. The first time scale is the shape solve, the second is the temperature evaluation as well as the computation of PRKE parameters, and the third is the PRKE scale.

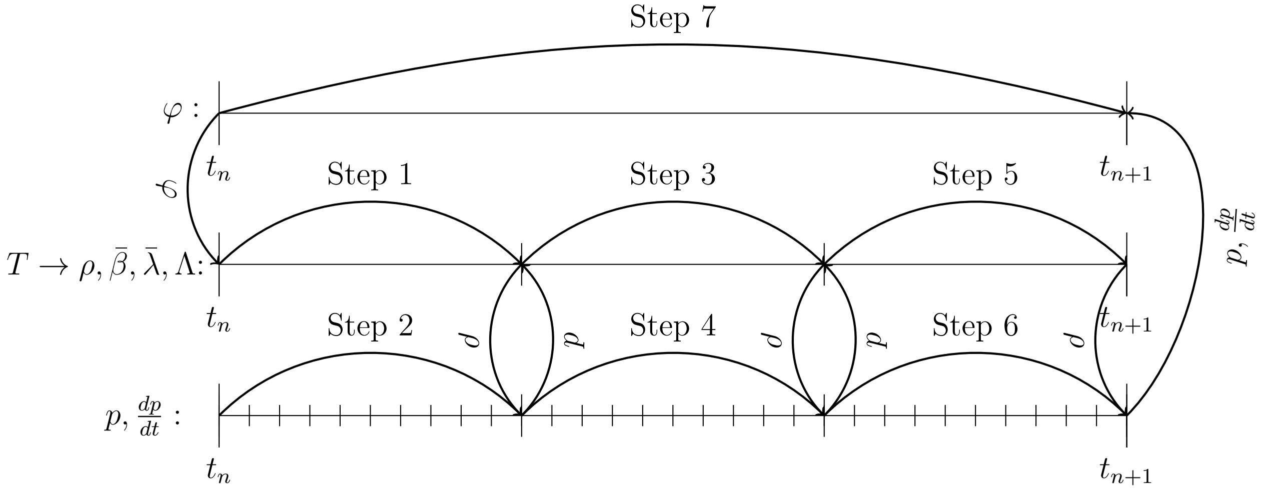

Figure 2: Time scales and process of IQS with temperature feedback

The couplings between temperature, amplitude, and shape are all nonlinear, so iteration processes are needed for each time scale. The amplitude and temperature need to be iterated on the middle time scale until convergence on each temperature step. Then another iterative process needs to occur in the shape time scale on all three variables.

3.5 Implementation into Rattlesnake/MOOSE Framework

A one-dimensional, single group FEM prototype will be created for IQS in order to test validation for simple problems. However, testing complex, real-world problems is most easily performed by implementing IQS in a FEM framework. The Rattlesnake radiation transport framework is a excellent option for performing these problems. Rattlesnake already contains a neutron diffusion method for continuous-FEM equivalent to the implicit discretization method. Additionally, Rattlesnake is well documented and well organized, so implementing IQS may be relatively straightforward. With this implementation, benchmark testing such as 16-A1 [1], TWIGL, LRA, C5G7-2D, etc. can be performed, as well as a full core simulation of the TREAT reactor.

4 Preliminary Results

This section describes preliminary results of an example that tests an IQS implementation and shows its effectiveness on computation speed and accuracy. The example is very simple and computes quickly; it entails a one dimensional, homogeneous 400 cm slab with a heterogenous perturbation in absorption cross section. Figure 3 shows how the regions of the slab are divided and Table 1 shows the initial material properties. Regions 2, 3, and 4 have slope perturbations at different points in time, Table 2 shows the values of the absorption cross-section in each region at the times of interest. The values of Σ_a between these times of interest are linear interpolations between the given values.

1	1	1	1	2	3	1	1	1	1	1	1	1	1	4	4	1	1	1	1

Figure 3: 1-D slab region identification

Table 1: 1-D slab material properties and problem parameters

$D(cm)$	$\Sigma_a(cm^{-1})$	$\nu\Sigma_f(cm^{-1})$	$v(cm/s)$	β	$\lambda(s^{-1})$
1.0	1.1	1.1	1,000	0.006	0.1

Table 2: 1-D slab absorption cross-section at times of interest

Region	Material Property	0.0 s	0.1 s	0.6 s	1.0 s	1.7 s
2	$\Sigma_a(cm^{-1})$	1.1	1.1	1.095	1.095	1.095
3	$\Sigma_a(cm^{-1})$	1.1	1.1	1.09	1.09	1.1
4	$\Sigma_a(cm^{-1})$	1.1	1.1	1.105	1.105	1.105

Figure 4 shows the power at each macro time step as compared to the traditional brute force (full flux implicit discretization) method. The strong correlation between the two curves shows that IQS is consistent with a proven method for a highly transient example. Figure 5 shows the error convergence for implicit discretization and IQS methods. This plot shows that IQS has consistent convergence slope for first and second order time schemes and that IQS has a better error constant for the convergence. Table 3 shows the results for time adaptation. These results show that traditional IQS does not perform well with step doubling time adaptation and IQS P-C performs moderately better than implicit discretization. The IQS performance in this example is not exceptional because the problem is difficult. As seen in Figure 6, the shape undergoes significant change through the transient, which requires many shape evaluations to retain accuracy.

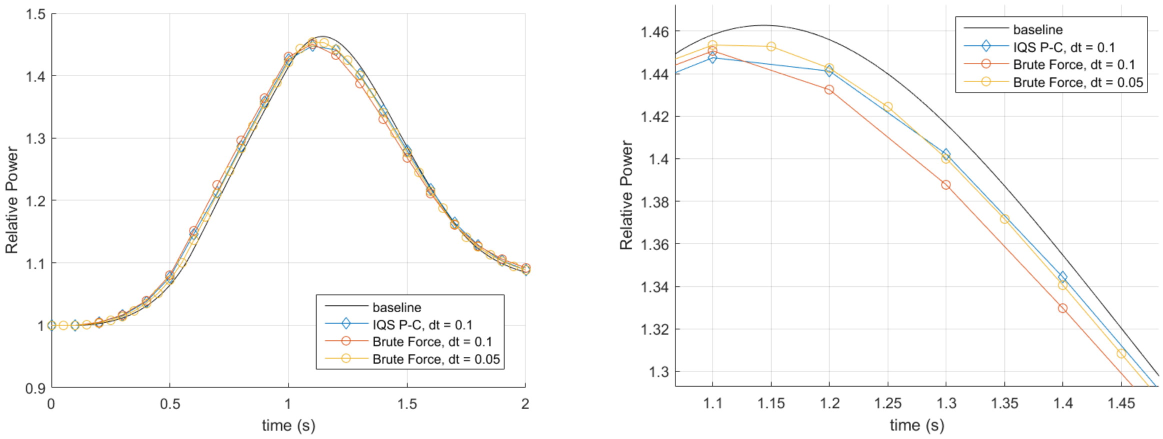

Figure 4: Power level of 1D example

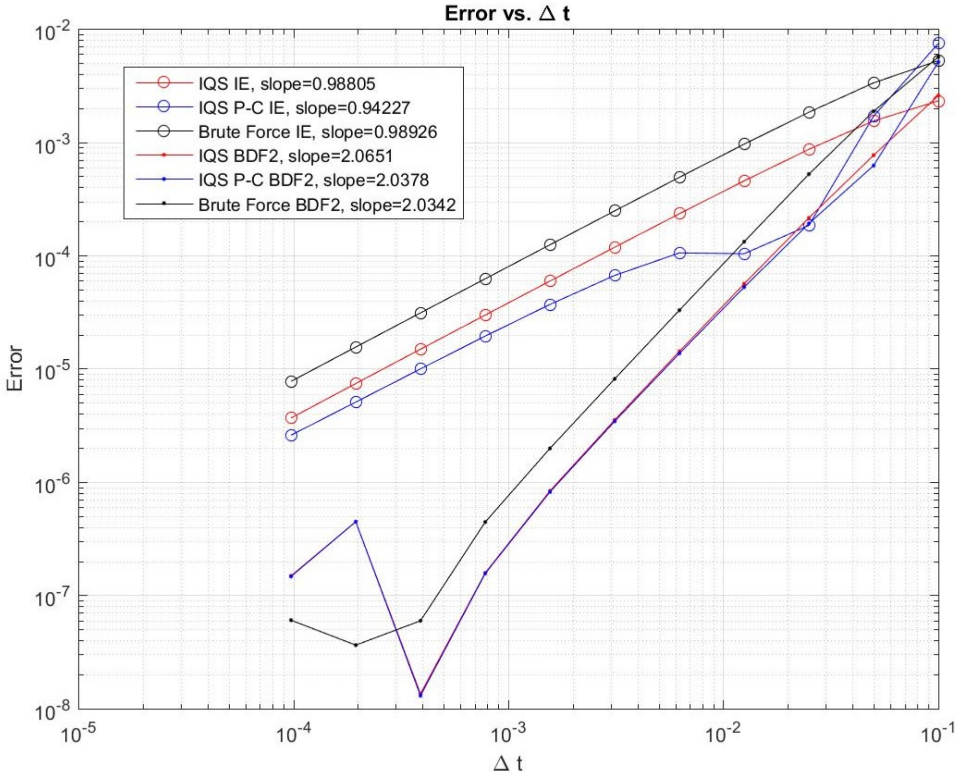

Figure 5: Error convergence comparison of 1D example

Table 3: 1-D slab step doubling results

		Implicit Discretization			IQS			IQS P-C		
Test	e_{tol}	Error	Steps	Solves	Error	Steps	Solves	Error	Steps	Solves
1	0.05	0.021596	10	32	0.10084	12	175	0.028019	10	33
2	0.01	0.032864	21	80	0.08030	41	850	0.044721	20	74
3	0.005	0.053159	27	96	0.01215	115	2220	0.052095	25	96
4	0.001	0.056546	56	188	0.03242	848	12511	0.061274	50	162
5	0.0005	0.062882	77	245	0.03491	1718	25841	0.061240	69	225
6	0.0001	0.060089	177	537	0.03554	8702	129985	0.060824	159	480
7	5.0e-05	0.059513	252	767	0.03622	17282	256463	0.061078	224	680
8	1.0e-05	0.061063	561	1691	0.03142	79988	1104227	0.061901	501	1509

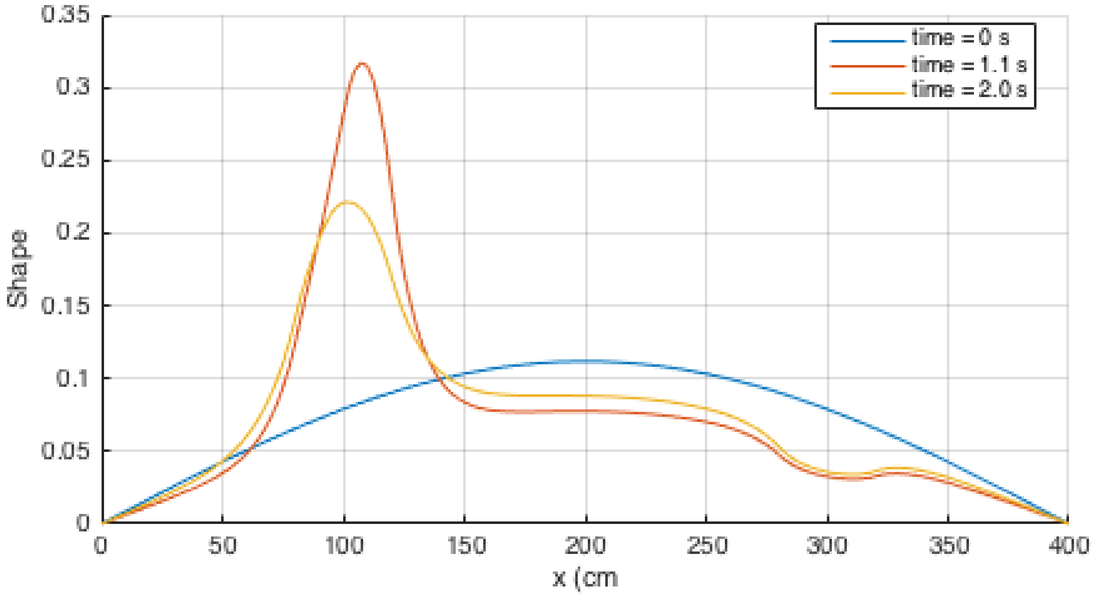

Figure 6: Flux profile of 1D slab at various times

5 Goals of Proposed Research

1. Investigate iteration convergence for previously implemented iterative schemes and newly conceived one described in Section 3.1.
2. Verify IQS method with several low and high order time discretization schemes.
3. Implement step doubling time adaptation with IQS and analyze its performance compared to implicit discretization of the flux equations.
4. Apply temperature feedback with IQS and analyze its performance with multiphysics benchmarks.
5. Implement IQS into the Rattlesnake framework at INL.
6. Perform testing with IQS to benchmarks including TWIGL, LRA, and LMW.

5.1 Possible Extensions

Some extensions, beyond the proposed scope of this research, have been identified as follows:

1. Perform a full core simulation of TREAT with IQS.
2. Investigate other multiphysics coupling with IQS including structural mechanics and thermal hydraulics.
3. Investigate IQS application to self-adjoint angular flux $\mathrm{S_N}$ (SAAF-SN).

References

[1] Argonne Code Center. Benchmark problem book, anl- 7416, suppl. 2. Technical report, Argonne National Laboratory, 1977.

[2] J. Devooght, B. Arien, E. H. Mund, and A. Siebertz. Fast reactor transient analysis using the generalized quasi-static approximation. *Nuclear Science and Engineering*, 88: 191–199, 1984.

[3] J. J. Duderstadt and L. J. Hamilton. *Nuclear reactor analysis*. Wiley, 1976.

[4] S. Dulla, E. H. Mund, and P. Ravetto. The quasi-static method revisited. *Progress in Nuclear Energy*, 50(8):908 – 920, 2008.

[5] D. Gaston, C. Newman, G. Hansen, , and D. Lebrun-Grandie. Moose: A parallel computational framework for coupled systems of nonlinear equations. *Nucl. Engrg. Design*, 239:1768 – 1778, 2009.

[6] S. Goluoglu and H. L. Dodds. A time-dependent, three-dimensional neutron transport methodology. *Nuclear Science and Engineering*, 139:248–261, 2001.

[7] H. IKEDA and T. TAKEDA. Development and verification of an efficient spatial neutron kinetics method for reactivity-initiated event analyses. *Journal of Nuclear Science and Technology*, 38:496–515, 2001. doi: 10.1080/18811248.2001.9715059. URL http://dx.doi.org/10.1080/18811248.2001.9715059.

[8] J. Koclas, M. Sissnoui, and A. Hébert. Solution of the improved and generalized quasistatic methods by kaps and rentrop integration scheme with stepsize control. *Annals of Nuclear Energy*, 23:901–907, 1996.

[9] A. Monier. *Application of the Collocation Technique to the Spatial Discretization of the Generalized Quasistatic Method for Nuclear Reactors*. PhD thesis, Université de Montréal, 1991.

[10] K. Ott. Quasi-static treatment of spatial phenomena in reactor dynamics. *Nuclear Science and Engineering*, 26:563, 1966.

[11] M. Sissnoui, J. Koclas, and A. Hébert. Solution of the improved and generalized quasistatic methods by kaps and rentrop integration scheme with stepsize control. *Annals of Nuclear Energy*, 22:763–774, 1995.

[12] Y. Wang. Nonlinear diffusion acceleration for multigroup transport equation discretized with sn and continuous fem with rattlesnake. In *Proc. International Conference on Mathematics and Computational Methods Applied to Nuclear Science & Engineering, Idaho*, 2013.