Multiphysics Core Dynamics Simulation Using the Improved Quasi-Static Method

Zachary M. Prince and Jean C. Ragusa

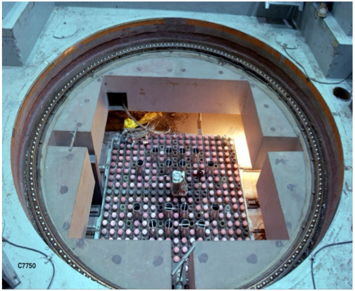
Department of Nuclear Engineering, Texas A&M University, College Station, TX

April 18, 2017

email: zachmprince@tamu.edu

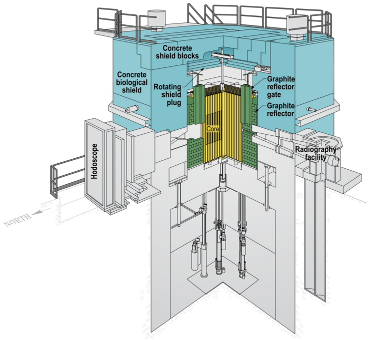
Outline

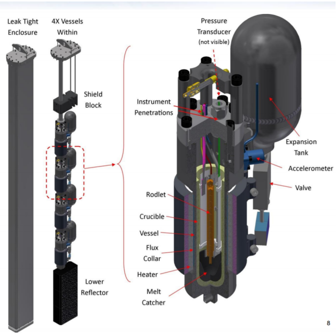
Transient Reactor Testing Facility (TREAT)

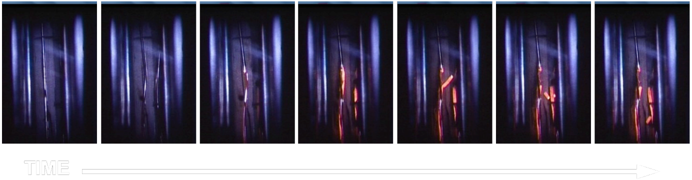


- Operation started in 1959, stand-by status in 1994, expected restart by 2020
- Designed to induce accident-like scenarios to fuel and other reactor components
- Air-cooled, graphite moderated, 100 kW steady-state, up to 19 GW peak transients

TREAT Experiments

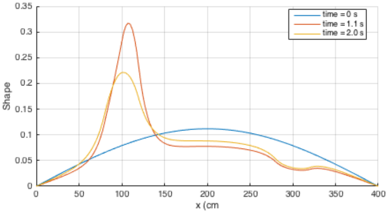






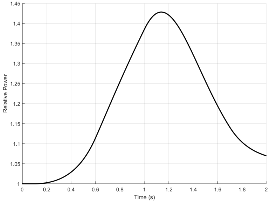
IQS mitigates neutronics expense

- IQS involves factorizing flux into space-time-energy-dependent space and time-dependent amplitude
- Shape maintains the difficulty of flux to evaluate, but amplitude is much easier
- The impetus of IQS is that shape is weakly dependent on time
- Shape and amplitude can be evaluated on different time scales to maximize efficiency



Shape$(\vec{r}, E, t)$

$\times$



Amplitude(t)

Time-dependent Multigroup Diffusion

Group Fluxes ϕ^g $(1 \le g \le G)$ with Precursors C_i $(1 \le i \le I)$

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

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

- Direct time discretization of these equations is termed "implicit discretization"
- These equations are particularly stiff due to the large value of v
- Implicit schemes are necessary with many time steps

Improved Quasi-Static Method (IQS)

IQS Factorization

Decomposition of the multigroup flux into the product of a time-dependent amplitude (p) and a space-/time-dependent multigroup shape (φ^g):

$$\phi^g(\vec{r},t) = p(t)\varphi^g(\vec{r},t)$$

- Factorization is **not** an approximation.
- Note that $p(t)$ and $\varphi^g(\vec{r},t)$ are not unique.
- Impetus is that $\varphi^g(\vec{r},t)$ is much slower varying than $\phi^g(\vec{r},t)$ and $p(t)$
- Equations for $\varphi^g(\vec{r},t)$ and $p(t)$ need to be derived

IQS Shape Equations

Shape Equations

Implementing factorization and solving for φ^g:

$$\frac{1}{v^g}\frac{\partial\varphi^g}{\partial t} = \frac{\chi_p^g}{k_{\text{eff}}}\sum_{g'=1}^{G}(1-\beta)\nu^{g'}\Sigma_f^{g'}\varphi^{g'} - \left(-\vec{\nabla}\cdot D^g\vec{\nabla} + \Sigma_r^g + \frac{1}{v^g}\frac{1}{p}\frac{dp}{dt}\right)\varphi^g$$

$$+ \sum_{g'\neq g}^{G}\Sigma_s^{g'\to g}\varphi^{g'} + \frac{1}{p}\sum_{i=1}^{I}\chi_{d,i}^g\lambda_i C_i \ , \quad 1\leq g\leq G$$

$$\frac{dC_i}{dt} = p\sum_{g=1}^{G}\nu_{d,i}\Sigma_f^g\varphi^g - \lambda_i C_i, \quad 1\leq i\leq I$$

Differences with original transport equation

1. An additional removal term based on $\frac{1}{v^g}\frac{1}{p}\frac{dp}{dt}\varphi^g$
2. Delayed neutron source term scaled by $\frac{1}{p}$
3. The delayed fission source in the precursor equation scaled by p

Amplitude equations (PRKE)

Principle

To obtain the amplitude equation, we multiply the shape equations with a weighting function (initial adjoint flux, ϕ^{*g}), then integrate over domain.

Uniqueness of the factorization

In order to impose uniqueness of the factorization, one requires:

$$K_0 = \sum_{g=1}^{G} \left(\phi^{*g}, \frac{1}{v^g} \varphi^g \right) = constant$$

Notation

$$(\phi^{*g}, f) = \int_D \phi^{*g}(\vec{r}) f(\vec{r}) dr^3$$

PRKE for IQS

PRKE

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

$$\frac{d\xi_i}{dt} = \frac{\bar{\beta}_i}{\Lambda} p - \bar{\lambda}_i \xi_i \quad 1 \le i \le I$$

PRKE Coefficients

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

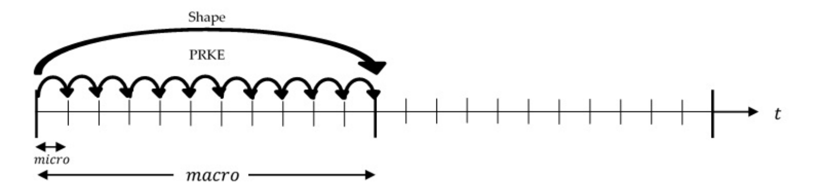
$$\frac{\bar{\beta}}{\Lambda} = \sum_{i=1}^{I} \frac{\bar{\beta}_i}{\Lambda} = \frac{1}{k_{eff}} \frac{\sum_{i=1}^{I} \sum_{g=1}^{G} (\phi^{*g}, \beta_i \nu^g \Sigma_f^g \varphi^g)}{\sum_{g=1}^{G} \left(\phi^{*g}, \frac{1}{v^g} \varphi^g\right)}$$

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

Quasi-Static Process

Time scales and IQS solution process

Because solving for the shape can be expensive, especially in two or three dimensions, it is attractive to make the assumption that the shape is weakly time-dependent so the shape can be computed after a multitude of PRKE calculations:



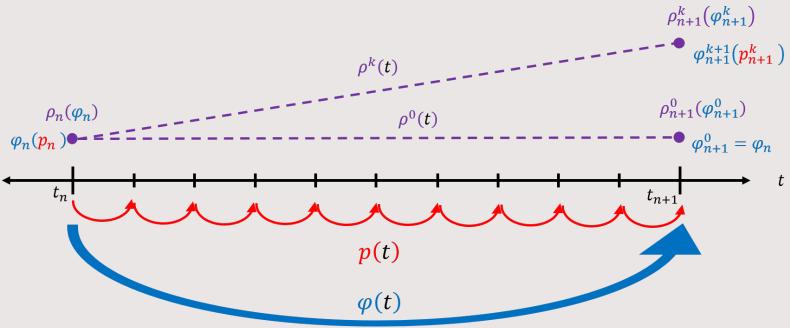
IQS is nonlinear

Nonlinear systems need an iterative solution process

There are two general iteration processes:

1. Fixed-point (Picard): back and forth corrections between amplitude and shape with relevant convergence criteria
2. Newton: residual-Jacobian based approach on shape

Fixed-Point Process



IQS Predictor-Corrector

IQS P-C Linearizes the System

IQS P-C linearizes the system and avoids iterations on the shape:

1. Evaluate multigroup diffusion equation to get predicted flux $\phi_{n+1}^{g,pred}$
2. Scale predicted flux to obtain shape:

$$\varphi_{n+1}^{g} = \phi_{n+1}^{g,pred} \frac{\sum_{g=1}^{G}\left(\phi^{*g}, \frac{1}{v^g}\phi_0^g\right)}{\sum_{g=1}^{G}\left(\phi^{*g}, \frac{1}{v^g}\phi_{n+1}^{g,pred}\right)} = \phi_{n+1}^{g,pred}\frac{K_0}{K_{n+1}}$$

3. Compute PRKE parameters at t_{n+1}
4. Evaluate PRKE along micro step using interpolated parameters to obtain p_{n+1}
5. Scale φ_{n+1}^{g} to obtain corrected flux:

$$\phi_{n+1}^{g,corr} = p_{n+1} \times \varphi_{n+1}^{g}$$

Advantage: No IQS nonlinear iteration is necessary
Disadvantage: Assumes $\sum_{g=1}^{G}\left(\phi^{*g}, \frac{1}{v^g}\varphi_{n+1}^{g}\right)$ is inherently constant

Multiphysics: Adiabatic heat-up with absorption cross-section feedback

Implemented Form

$$\rho c_p \frac{\partial T(\vec{r},t)}{\partial t} = \kappa_f \sum_{g=1}^{G} \Sigma_f^g \varphi^g(\vec{r},t) p(t)$$

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

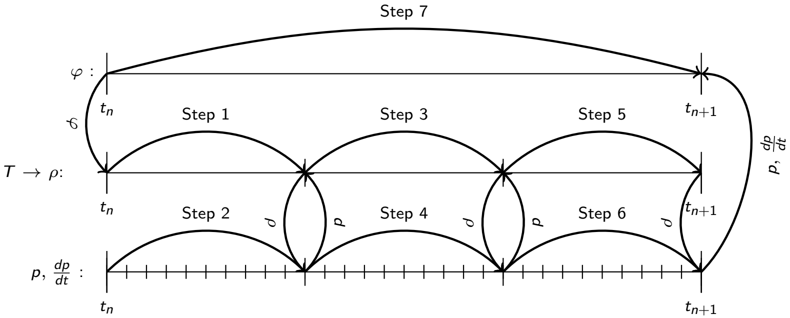
Analytical temperature integration with IQS

$$T^{n+1} = T^n + \frac{\kappa_f}{\rho c_p}\sum_{g=1}^{G}\left(a_2(\Sigma_f^g\varphi^g)^{n+1} + a_1(\Sigma_f^g\varphi^g)^n\right)$$

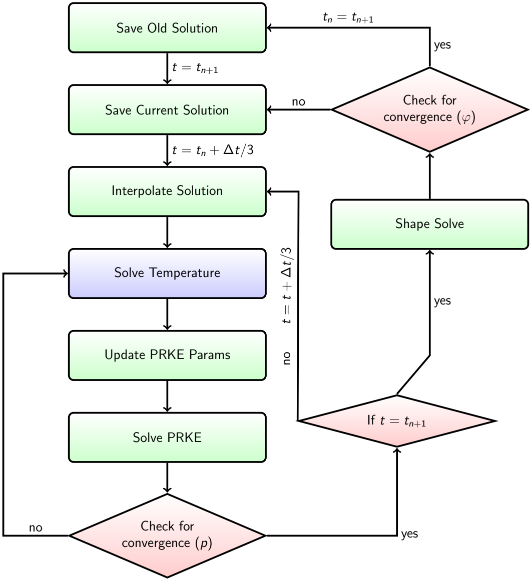
$$a_1 = \int_{t_n}^{t_{n+1}} \left(\frac{t_{n+1}-t'}{\Delta t}\right) p(t')dt'$$

$$a_2 = \int_{t_n}^{t_{n+1}} \left(\frac{t'-t_n}{\Delta t}\right) p(t')dt'$$

Intermediate time scale for temperature



Time scale programming logic



Time Scale Analysis

Dynamical Time Scale

- The time variance of each physics (θ) can be quantified by defining a dynamical time scale (τ):

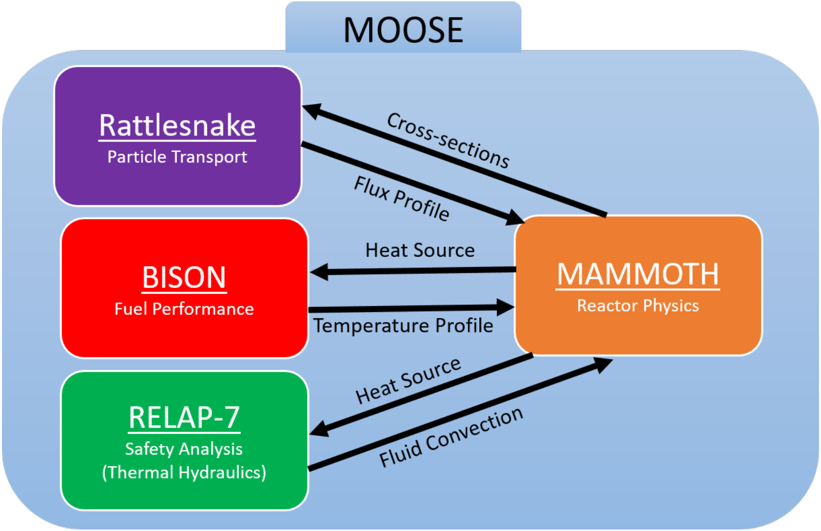
$$\tau = \frac{1}{\left|\frac{1}{\theta}\frac{d\theta}{dt}\right|}$$

- Finite difference approximation for $d\theta/dt$ and average for $1/\theta$
- Only temporal behavior is of interest, so the L^2 norm will be taken of each quantity, resulting in:

$$\tilde{\tau}_{n+1} = \frac{\|\theta_{n+1} + \theta_n\|_{L^2}}{2} \frac{\Delta t}{\|\theta_{n+1} - \theta_n\|_{L^2}}$$

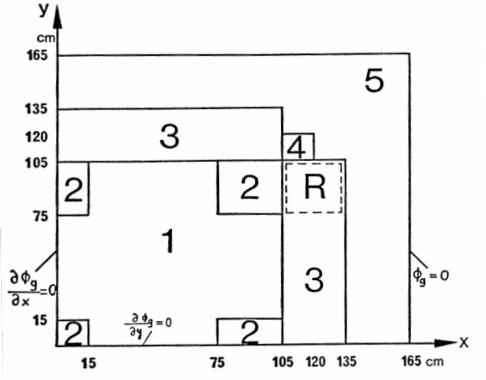
- According to the a priori hypothesis, τ is large for shape, somewhat smaller for temperature, and much smaller for amplitude and flux

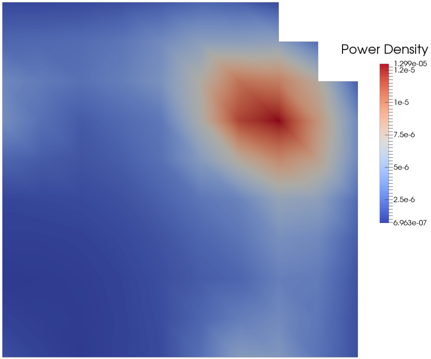
Reactor Simulation in MOOSE



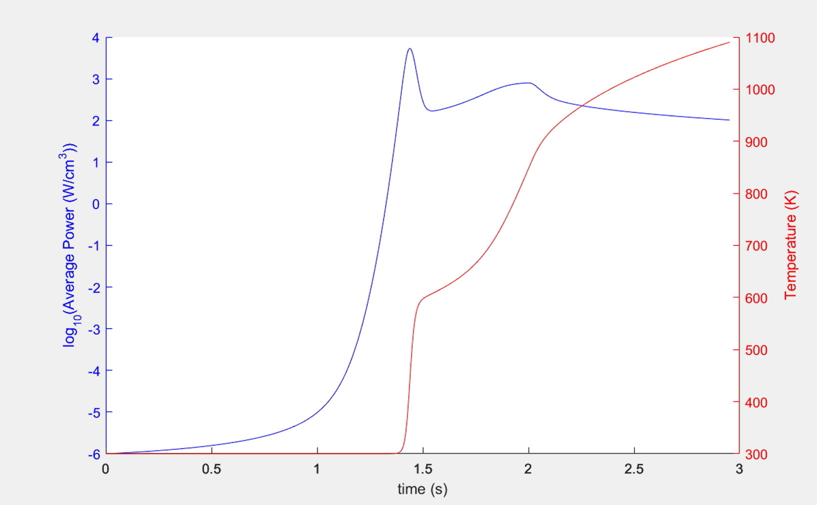
Results

LRA Benchmark





LRA Power and Temperature Profile



LRA Time Step Error Convergence

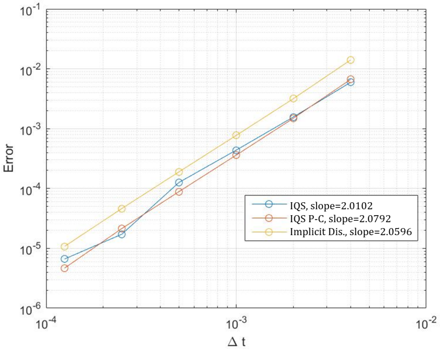


Figure: Only one temperature update per macro step

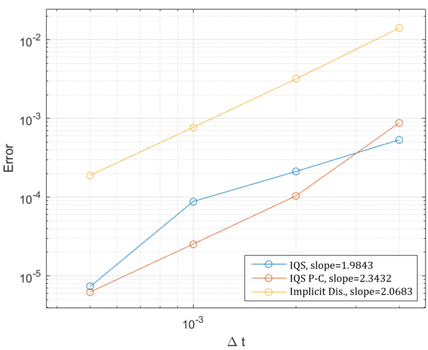
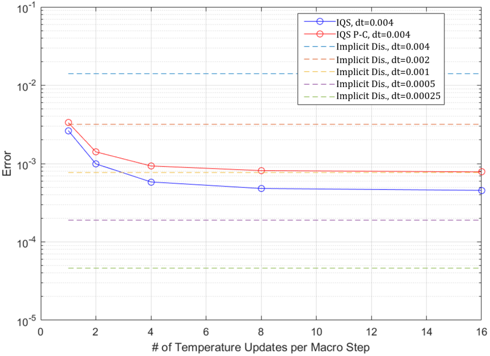


Figure: Five temperature updates per macro step

Analysis on Temperature Updates



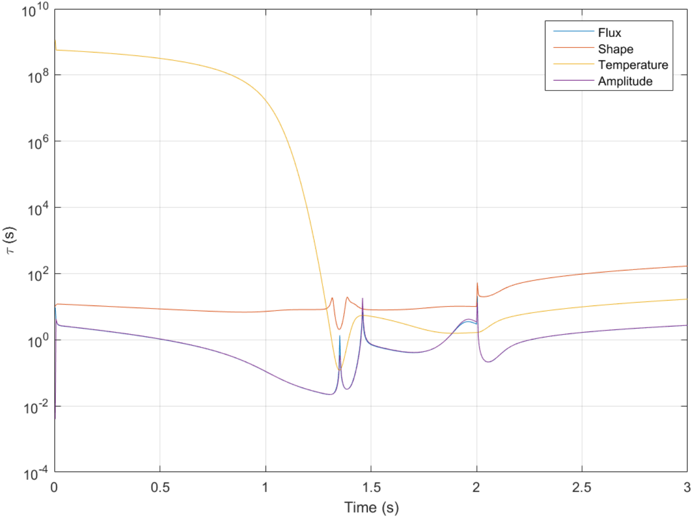
Implicit Discretization				
Run	Δt	Error	Runtime (hr)	Linear Iter.
1	4.0e-3	1.407e-2	4.11	7.13e4
2	2.0e-3	3.174e-3	6.01	9.49e4
3	1.0e-3	7.690e-4	10.38	1.45e5
4	5.0e-4	1.892e-4	21.91	2.08e5
5	2.5e-4	4.590e-5	25.23	3.16e5

IQS				
Run	Temperature Updates	Error	Runtime (hr)	% Increase in Runtime*
1	1	2.612e-3	3.96	-3.18%
2	2	9.893e-4	6.02	47.1%
3	4	5.796e-4	7.87	92.3%
4	8	4.772e-4	12.61	207.9%
5	16	4.516e-4	22.14	440.7%

IQS P-C				
Run	Temperature Updates	Error	Runtime (hr)	% Increase in Runtime*
1	1	3.488e-3	2.91	-28.9%
2	2	1.349e-3	3.73	-9.00%
3	4	9.161e-4	3.97	-3.04%
4	8	8.052e-4	5.39	31.7%
5	16	7.905e-4	8.19	100%

* runtime difference from $\Delta t = 0.004$ implicit dis.

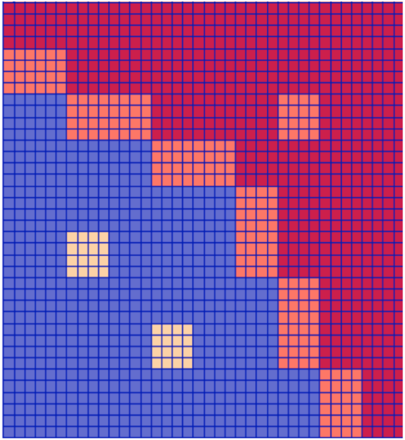
LRA Dynamical Time Scale Analysis



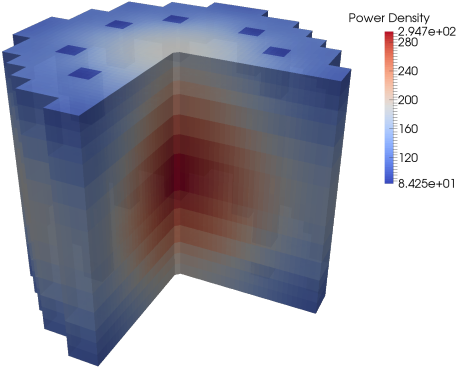
TREAT: Transient-15 Example

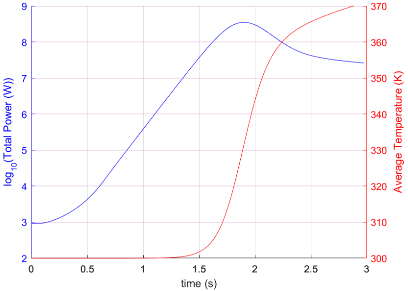
	A	B	C	D	E	F	G	H	J	K	L	M	N	O	P	R	S	T	U
1	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
2	A	A	A	A	A	A	A	A	Z	Z	Z	A	A	A	A	A	A	A	A
3	A	A	A	A	A	A	Z	Z	F	F	F	Z	Z	A	A	A	Z	A	A
4	A	A	A	A	A	Z	F	F	F	F	F	F	F	Z	Z	A	A	A	A
5	A	A	A	A	Z	F	F	F	F	F	F	F	F	F	F	Z	A	A	A
6	A	A	A	Z	F	F	F	C	F	F	F	C	F	F	F	Z	A	A	A
7	A	A	Z	F	F	F	F	F	F	F	F	F	F	F	F	F	Z	A	A
8	A	A	Z	F	F	C	F	F	F	F	F	F	F	C	F	F	Z	A	A
9	A	Z	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	Z	A
10	A	Z	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	Z	A
11	A	Z	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	Z	A
12	A	A	Z	F	F	C	F	F	F	F	F	F	F	C	F	F	Z	A	A
13	A	A	Z	F	F	F	F	F	F	F	F	F	F	F	F	F	Z	A	A
14	A	A	A	Z	F	F	F	C	F	F	F	C	F	F	F	Z	A	A	A
15	A	A	A	Z	F	F	F	F	F	F	F	F	F	F	Z	A	A	A	A
16	A	A	A	A	Z	Z	F	F	F	F	F	F	F	Z	A	A	A	A	A
17	A	A	A	A	A	A	Z	Z	F	F	F	Z	Z	A	A	A	A	A	A
18	A	A	A	A	A	A	A	A	Z	Z	Z	A	A	A	A	A	A	A	A
19	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A

- A Al-Clad Dummy Assembly
- C Control Rod Fuel Assembly (Short Poison Section)
- F Standard Fuel Assembly
- Z Zr-Clad Dummy Assembly



Transient-15 Power Profile





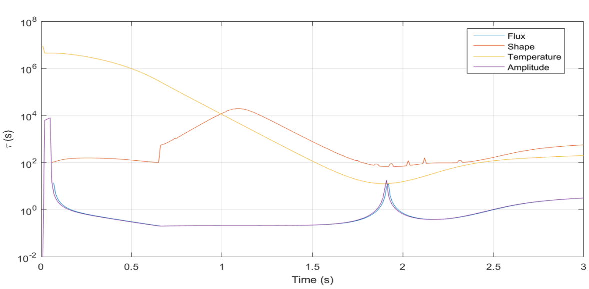
Transient-15 Results

Accuracy and Runtime

Method	No. of Steps	% Increase Runtime*	Max Power Error
Implicit Dis.	300	—	7.875e-4
IQS	300	-11.9%	8.385e-5
IQS (5 updates)	300	49.7%	3.687e-5
IQS P-C	300	-2.1%	7.527e-4
IQS P-C (5 updates)	300	26.5%	1.227e-4

* difference in runtime from implicit discretization

Dynamical Time Scale



Conclusions

Summary

- Derivation of IQS and implementation of quasi-static time stepping
- Semi-analytic evaluation of temperature
- Implementation of adiabatic heat up in quasi-static process
- Testing with LRA benchmark and TREAT Transient-15 model

Conclusions

- Temperature-amplitude iteration decreased temperature-shape iteration
- Semi-analytic temperature evaluation improved IQS CPU performance
- Intermediate time scale significantly improved IQS time step performance.
- Optimal CPU performance with 1 update for IQS and 4 updates for IQS P-C
- Time constant analysis showed adaptation could further improve performance

Future Work

- Implement intermediate time scale to more advanced multiphysics
- Develop adaptation technique for number of multipysics updates

Questions?

Acknowledgments

- This material is based upon work supported under an Integrated University Program Graduate Fellowship

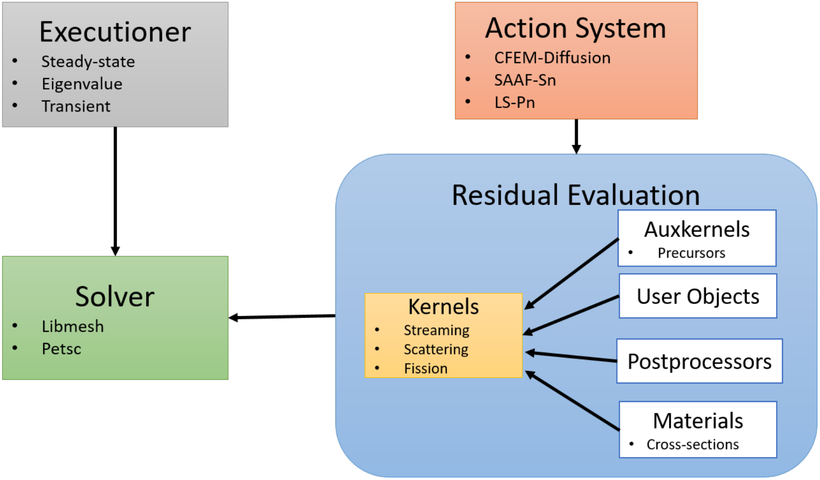
IUP Integrated University Program
U.S. Department of Energy

- Any opinions, findings, conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the Department of Energy Office of Nuclear Energy

NUCLEAR ENGINEERING
TEXAS A&M UNIVERSITY

Idaho National Laboratory

Rattlesnake Structure



IQS Implementation in Rattlesnake

IQS Components in Rattlesnake

- IQS Executioner
  - Convergence criteria for Picard iteration:

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

  - Evaluates PRKE using implicit Euler, Crank-Nicolson, or SDIRK33 with step doubling adaptation for $\frac{1}{p}\frac{dp}{dt}$ term
- PRKE Parameter Postprocessors
  - Performs integrations for PRKE parameters
  - Residuals from kernels are saved for $\rho - \bar{\beta}$ integration
- PRKE User Object
  - Gathers postprocessor values
- IQS Removal Kernel
  - Removal kernel for $\frac{1}{v^g}\frac{1}{p}\frac{dp}{dt}\varphi^g$ term
- Auxkernels
  - Precursor auxkernel with analytical integration
  - Temperature auxkernel with analytical integration

IQS Implementation in Rattlesnake (cont.)

IQS Kernels

$$\frac{1}{v^g}\frac{\partial \varphi^g}{\partial t} = \underbrace{\frac{\chi_p^g}{k_{eff}}\sum_{g'=1}^{G}(1-\beta)\nu^{g'}\Sigma_f^{g'}\varphi^{g'}}_{FluxKernel} + \underbrace{\sum_{g'\neq g}^{G}\Sigma_s^{g'\to g}\varphi^{g'}}_{FluxKernel} - \underbrace{\left(-\vec{\nabla}\cdot D^g\vec{\nabla}\right)\varphi^g}_{FluxKernel} - \underbrace{\Sigma_r^g\varphi^g}_{FluxKernel}$$

$$\underbrace{-\frac{1}{v^g}\boxed{\overbrace{\frac{1}{p}\frac{dp}{dt}}^{FromExecutioner}}\varphi^g}_{IQSKernel} + \underbrace{\frac{1}{p}\sum_{i=1}^{I}\chi_{d,i}^g\lambda_i C_i}_{ModifiedFluxKernel}$$