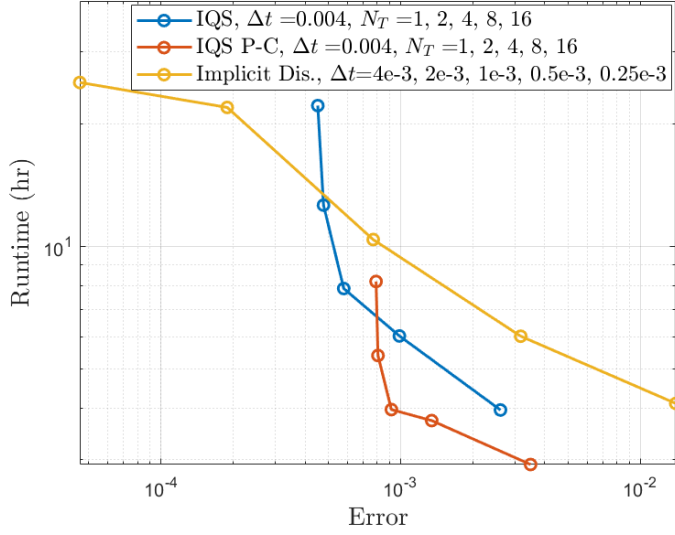


## Major Issues

- ✓ Justify page 8: "we expect temperature to be more rapidly varying than the shape but less so than the amplitude"
  - This idea stems from the fact that in a adiabatic temperature model, the temperature's spatial variation follows the neutron flux shape, while the temperature amplitude follows the flux amplitude. Because of the heat capacity of the fuel, the temperature amplitude is more slowly varying in time than the heat inflow from the flux amplitude, while still considerably faster than the time variance of the flux shape. This logic is empirically defended in the results of Section 5.
- ✓ Justify page 5: "Using IQS, one expects the time dependence of the shape to be weaker than that of the flux itself, this allowing for larger time step sizes in updating the shape".
  - Reference first IQS paper
- Results are often just stated and not discussed, e.g. page 19 "It also shows that traditional IQS performed better with large etol, while IQS-PC was better with smaller etol". The statement simply bloats the manuscripts; discuss the reasons and implications or remove it. Another example is page 15, "However, switching the precursors solve to be performed semi-analytically...". In this case, the authors should take the time to discuss the implications of this finding.
  - I will remove the first referenced quote.
  - I will add that if one expecting a certain tolerance to be reached in the iterative process, the semi-analytical integration of the precursors is necessary. Since, in the majority of cases, this evaluation is negligible compared to the evaluation of the shape, it is fortuitous to apply this technique in all cases.
- For the comparison of LRA and TREAT results, the efficiency [how accurate can you be given a fixed execution time or given an accuracy how much execution time do you need] is a very important quantity; the authors discuss it for LRA and TREAT. However, the authors first present results for the errors and then runtime results. Then they try to synthesize efficiency by comparing different cases that have different runtimes and different errors, e.g. page 23 right before Table 5 "IQS-PC shows and error less than implicit discretization at delta t = 0.002, signifying ...". I suggest the authors create error vs. execution time plots for different cases and based on these curves discuss efficiency. The manner efficiency is discussed is very confusing and not very useful for the reader.
  - Fig. 17 visualizes the dependence of runtime on error. This figure shows that IQS and IQS P-C generally has a lower runtime for certain error than implicit discretization.
  - I need help interpreting this result

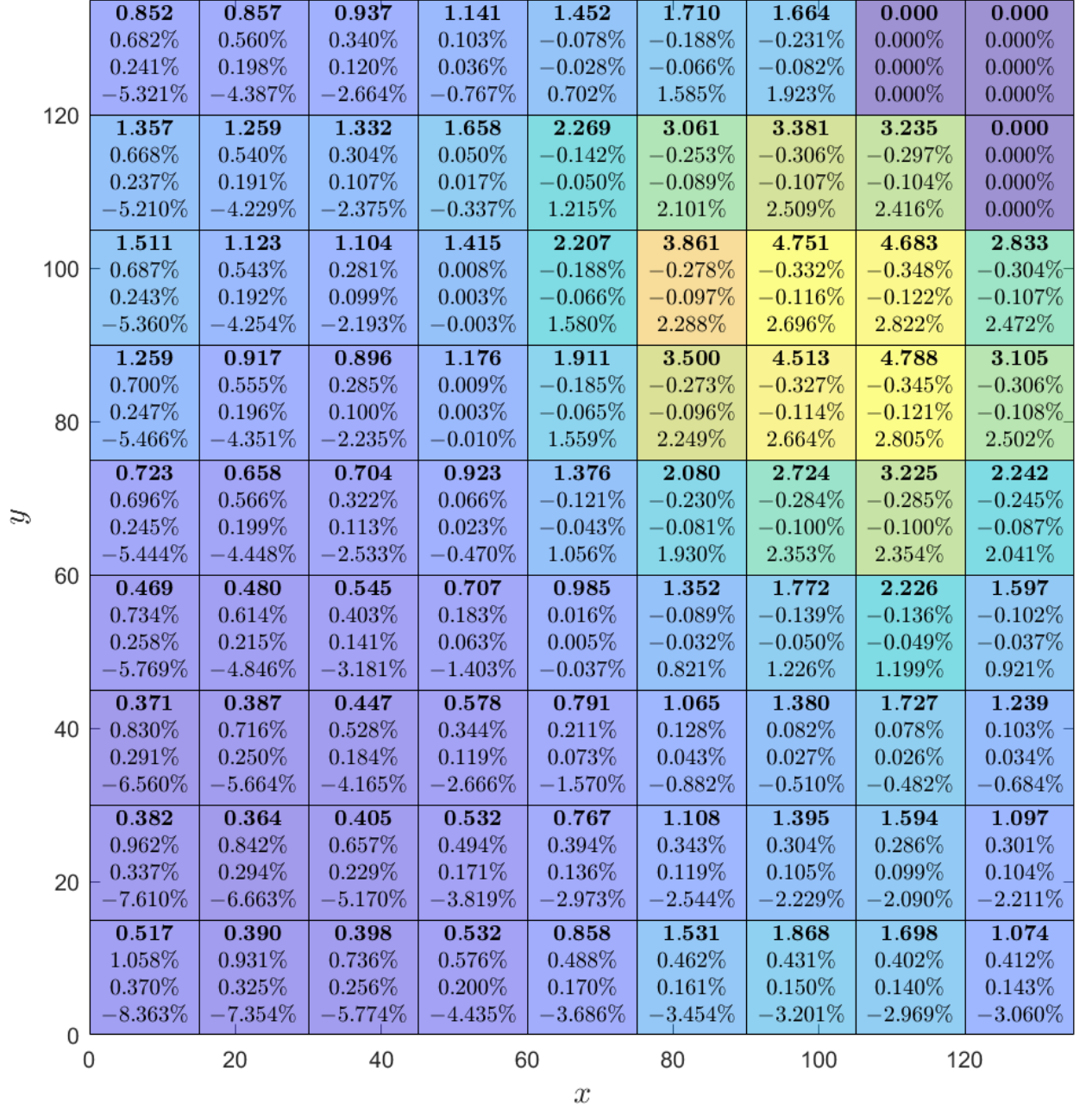


- ✓ In the results section, Error is sometimes used without introducing it, e.g. page 25, "Table 9 shows the error ...". Further "Error" is often used as y-label in plots., e.g. in the discussion of Figure 10 in which "convergence results" are presented; the sentence is not clear in this context because it was not stated if this is iterative convergence or convergence to the right solution as the time step is reduced. In some instances error is defined but it is tedious for the reader to find the corresponding passage in the text, figures, or tables.

- Included an explicit definition of error in each of the plots and tables

- ✓ LRA and TREAT only compare errors computed with the peak powers. Peak powers are an integral quantity and may favor IQS over spatial kinetics. The authors should also compare power distributions or similar distributed quantities.

- All the values for error in this section are errors in the globally integrated flux. However, it is noteworthy to compare the spatial dependence of error. Fig. 18 shows the difference in normalized power at each block of the LRA geometry for each method. The resulting spatially dependent error is in general an order of magnitude lower for IQS and IQS P-C than implicit discretization. This observation corresponds well with the globally integrated flux errors shown in Tables 6-8.



**Normalized Power (Ref.)**  
Difference (IQS) -  $\Delta t = 0.004$ ,  $N_T = 4$   
Difference (IQS P-C) -  $\Delta t = 0.004$ ,  $N_T = 4$   
Difference (Implicit Dis.) -  $\Delta t = 0.004$

$$\text{Difference} = \frac{\text{Power}^{\text{ref}} - \text{Power}}{\text{Power}^{\text{ref}}}$$

- Whenever errors are computed, baseline calculations are performed. How do the authors know that these baseline calculations are accurate enough to measure the error w.r.t. them?
- Can we just say that a proper error convergence means that the baseline is good enough?

## Minor Issues

- ✓ Page 1: "... the amplitude, and a space- and time-dependent component, the shape [1,2,3,4,5]". What about energy?
  - IQS involves factorizing the neutron flux solution into a time-only-dependent component, the amplitude, and a full phase-space component, the shape
- ✓ Is  $\phi^g$  the scalar flux or scalar flux moments in Eq 1a?
  - Note that  $\phi^g$  for neutron transport is the angular neutron flux ( $\phi^g(\vec{r}, \vec{\Omega}, t)$ ), while for neutron diffusion it is scalar flux ( $\phi^g(\vec{r}, t)$ ).
- All quantities in Eq. 1a and 1b need to be defined.
  - Created a table of the operator definitions (Table 1)
  - Do I need to include a definition of all the properties?
- ✓ Eq. 3a:  $S_d^g$  is not defined.
  - See Table 1
- ✓ For the uninitiated reader the term phase-space domain may not be familiar. The authors could introduce the term phase space. IQS may be interesting for people in other fields.
  - In neutron transport, the neutron flux solution lives in a seven dimensional phase-space, dependent on time, space, energy, and direction. The neutron diffusion equation reduces this phase-space by eliminating direction.
- ✓ Algorithm in section 2.1 on page 5. Step 2: "linearly interpolate". It is later stated that interpolation of higher order is done for high order schemes.
  - Removed linearly from step 2
  - It should be noted that Step 2 of the process generally involves a linear interpolation; however, for greater than second order shape discretizations, a higher-order Hermite interpolation is necessary.
- ? For the Linf nor of the shape: why is the maximum difference over the maximum value used and not the maximum relative difference ( $1 - \phi^{k+1}/\phi^k$ ).
  - I included the reference for this error estimation.
- ✓ Eq (11) is restated in Eq (24). Why not just make Eq. 11 more general. Also, in Eq. (24)  $cp(T(t))$  and originally the heat equation reads  $d(\rho * cp(T) * T)$ . Hence, the  $cp(T)$  cannot be pulled out so Eq. (24) is incorrect.
  - For the purposes of this derivation, the adiabatic model, given in Eq. (11), has linear heat capacity. However, a nonlinear heat capacity can easily be implemented with a iterative procedure.
  - Put  $\rho C_p(T)$  inside the derivative
- ✓ Figure 2. uses the  $\Delta t / 3$  step that was claimed to be only for illustration purposes in Fig. 1.
  - I'll change the 3 to  $N_T$  in Figure 2, which is the number of temperature solves within a macro step
- ✓ Statement page 9: "Furthermore, a very accurate representation of  $p(t)$  over the macro step is available from the PRKE solve...". I think it should be precise or else one would have to show that the equations are solved accurately. Second, what does "very accurate" mean, "very accurate" is comparison to what?

- Furthermore, a highly refined representation of  $p(t)$  over the macro step is available from the PRKE solve (micro time scale). Using this fine scale information yields a more precise integration of amplitude than assuming an interpolation of the amplitude between macro step points.
- ✓ Implementation uses code specific jargon: kernels, auxkernels, Transient executioner, user-object. These need to be explained or cast in more general terms.
  - Kernel - Evaluation of the weak-form residual for a particular piece of physics.
  - Executioner - Establishes the type of simulation. The Transient type moves the simulation in time conducting spatial evaluations at each step.
  - Auxiliary variable and kernel - "Optional" variables that live on the same mesh as the solution and are computed using auxiliary kernels.
  - Postprocessor - Computes scalar values over the entire spatial mesh, usually in an integral sense.
  - User-object - A generic type of postprocessor that allows connectivity of relevant quantities between different MOOSE objects.
  - A more detailed and comprehensive description of MOOSE objects can be found in [18]
- ✓ Page 13 top of the page "looping over cells to evaluate", what cells, mesh cells?
  - Spatial mesh cells
- ✓ LRA benchmark: the number of elements makes no sense  $11 \times 11 = 121$  and when using five uniform refinement steps one would end up with 123,904 elements.
  - Fixed it to 123,904 elements, number of nodes was originally correct
- ✓ LRA benchmark description does not state what equations are solved, if that is provided in the ANL benchmark book it should be made clear. Equations could also be stated directly.
  - Seems tedious to provide them, so will reference the book
- ✓ In plots where delta t is plotted on the x-axis, it should be made clear that it is the macroscopic time step.
  - Put a note at the end of the intro of each results section
- ✓ Conclusions, page 27: incomplete sentence "IQS showed expected error convergence up".
  - ...through fourth-order discretization of the shape
- ? Timing estimates for LRA: the author should make clear how they execute the LRA problem, i.e. number of processors and processor model. Given the long runtimes, it appears that a single processor is used. Is this to avoid problems with asynchronous behavior in parallel?
  - Already stated: These run-times are based on total alive time of the execution where the diffusion evaluation is distributed over 24 processors.