Moltres: a MOOSE Application for Simulation of MSRs

Alex Lindsay and Kathryn Huff

University of Illinois

Presentation Overview

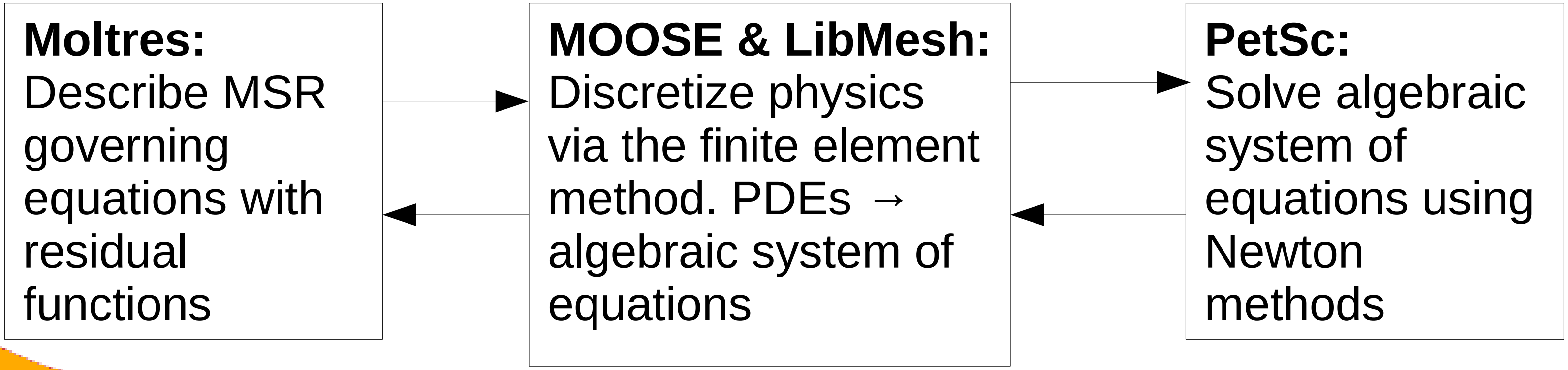
- Intro to Moltres/MOOSE
- Moltres goals
- Results:
  - 3D unstructured: temperature coupling issues
  - 2D axisymmetric: good steady-state profiles
  - 3D structured: steady-state neutron and precursor profiles; incomplete conduction between fuel and moderator
- Future work
  - Transients

Obtaining Moltres

- git clone https://github.com/arfc/moltres
- cd moltres
- git submodule init
- git submodule update

Moltres/MOOSE

- Moltres is built on top of the Multi-physics Object-Oriented Simulation Environment (MOOSE):
- MOOSE interfaces with libMesh to discretize simulation volume into finite elements
- Provides interface for coding residuals that correspond to weak form of governing PDEs; also interface for coding Jacobians → more accurate Jacobians → more efficient convergence
- Residuals and Jacobians handed off to PetSc which handles solution of resulting non-linear system of algebraic equations

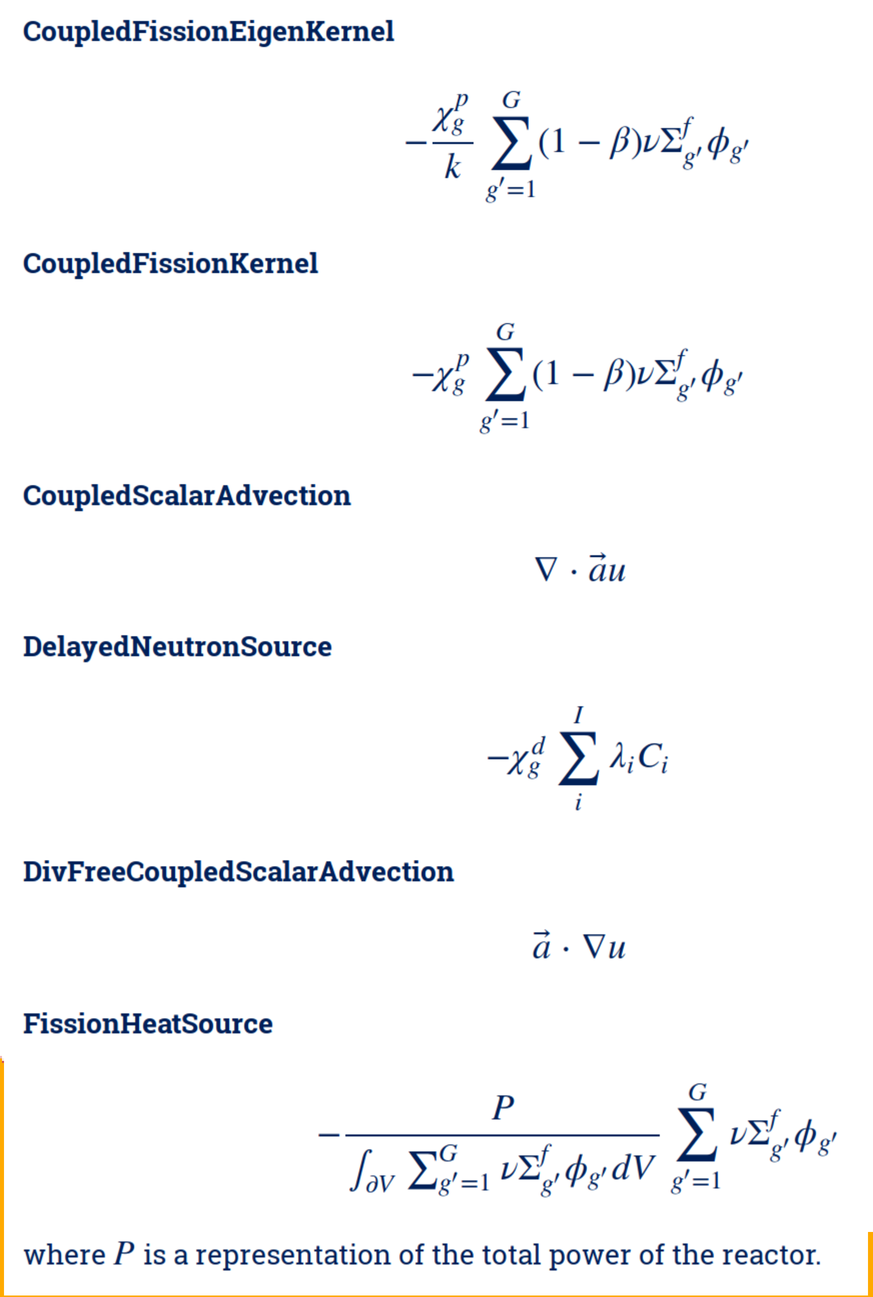


Moltres: Describe MSR governing equations with residual functions

MOOSE & LibMesh: Discretize physics via the finite element method. PDEs → algebraic system of equations

PetSc: Solve algebraic system of equations using Newton methods

Moltres Kernels



CoupledFissionEigenKernel

$$-\frac{\chi_g^p}{k}\sum_{g'=1}^{G}(1-\beta)\nu\Sigma_{g'}^f\phi_{g'}$$

CoupledFissionKernel

$$-\chi_g^p\sum_{g'=1}^{G}(1-\beta)\nu\Sigma_{g'}^f\phi_{g'}$$

CoupledScalarAdvection

$$\nabla\cdot\vec{a}u$$

DelayedNeutronSource

$$-\chi_g^d\sum_{i}^{I}\lambda_iC_i$$

DivFreeCoupledScalarAdvection

$$\vec{a}\cdot\nabla u$$

FissionHeatSource

$$-\frac{P}{\int_{\partial V}\sum_{g'=1}^{G}\nu\Sigma_{g'}^f\phi_{g'}\,dV}\sum_{g'=1}^{G}\nu\Sigma_{g'}^f\phi_{g'}$$

where P is a representation of the total power of the reactor.

- In MOOSE jargon, kernels are individual pieces of governing equations
- Modular in nature; for example, a “Diffusion” kernel could be used equally well to describe conduction or viscous shear

Moltres Kernels

GammaHeatSource

$$-\gamma Q_f$$

where γ is a factor representing heat dissipation by gamma and neutron irradiation in the moderator and Q_f is given by:

$$\sum_{g=1}^{G} \epsilon_{f,g} \Sigma_{f,g} \phi_g$$

with $\epsilon_{f,g}$ the amount of heat given off per fission event.

GroupDiffusion

$$-\nabla \cdot D_g \nabla \phi_g$$

InScatter

$$-\sum_{g \neq g'}^{G} \Sigma^s_{g' \to g} \phi_{g'}$$

NtTimeDerivative

$$\frac{1}{v_g} \frac{\partial \phi_g}{\partial t}$$

PrecursorDecay

$$\lambda_i C_i$$

PrecursorSource

$$-\sum_{g'=1}^{G} \beta_i \nu \Sigma^f_{g'} \phi_{g'}$$

ScalarAdvectionArtDiff

$$\nabla \cdot -\delta \nabla u$$

where δ is an artificial diffusion coefficient determined by:

$$\delta = \frac{|\vec{a}| h_{max}}{2}$$

with $\vec{a}$ the advection velocity and h_{max} the maximum element length dimension.

ScalarTransportTimeDerivative

$$\frac{\partial u}{\partial t}$$

SelfFissionEigenKernel

$$\frac{-\nu_f \Sigma_f \phi}{k}$$

SigmaR

$$\Sigma^r_g \phi_g$$

TransientFissionHeatSource

$$-\sum_{g=1}^{G} \epsilon_{f,g} \Sigma_{f,g} \phi_g$$

Governing Equations

- Piece together the kernels:
  - Multi-group diffusion
  - Precursor balance with drift
  - Heat conduction-convection with fission source in fuel
  - Heat conduction with option for irradiation source in moderator

$$\frac{1}{v_g}\frac{\partial \phi_g}{\partial t} - \nabla \cdot D_g \nabla \phi_g + \Sigma_g^r \phi_g = \sum_{g \neq g'}^{G} \Sigma^s_{g' \to g} \phi_{g'} + \chi_g^p \sum_{g'=1}^{G} (1-\beta)\nu \Sigma^f_{g'} \phi_{g'} + \chi_g^d \sum_{i}^{I} \lambda_i C_i$$

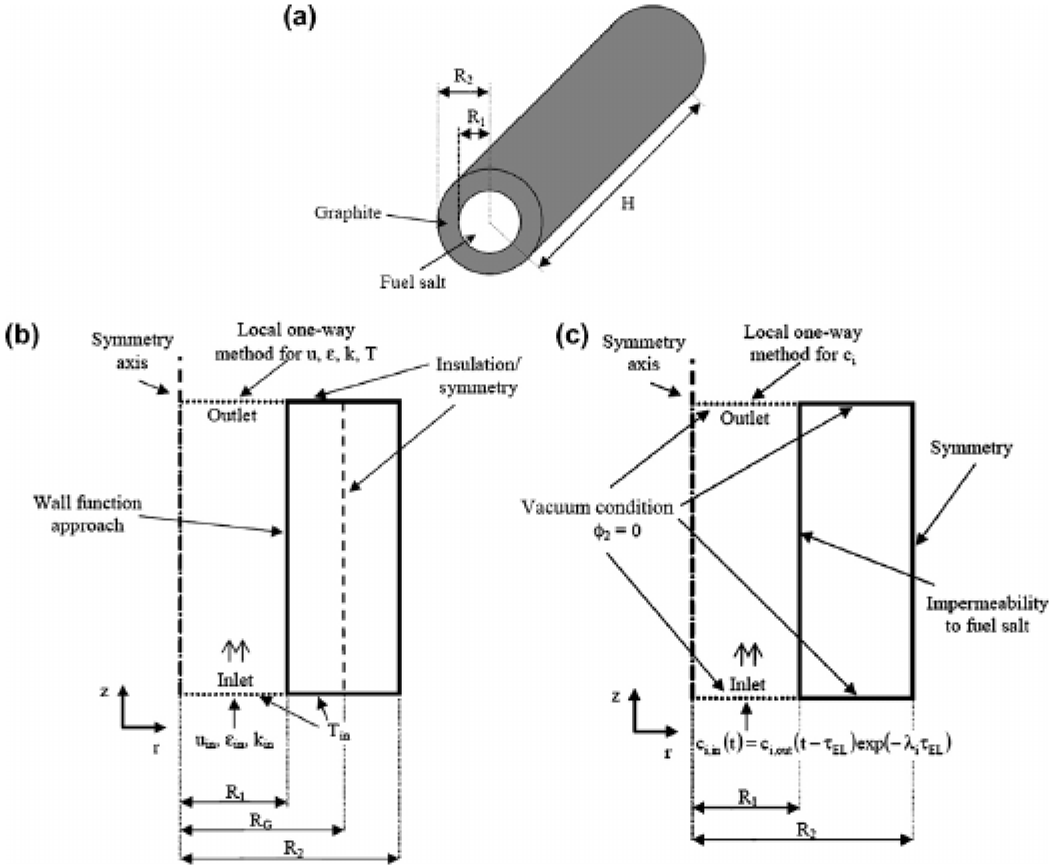
$$\frac{\partial C_i}{\partial t} = \sum_{g'=1}^{G} \beta_i \nu \Sigma^f_{g'} \phi_{g'} - \lambda_i C_i - \frac{\partial}{\partial z} u C_i$$

$$\rho_f c_{p,f} \frac{\partial T_f}{\partial t} + \nabla \cdot (\rho_f c_{p,f} \vec{u} \cdot T_f - k_f \nabla T_f) = Q_f$$

$$\rho_g c_{p,g} \frac{\partial T_g}{\partial t} + \nabla \cdot (-k_g \nabla T_g) = Q_g$$

Example Literature MSR Multi-Physics

- Cammi *Ann. of Nuc. Energy* **38** (2011) 1356
- Single channel
- Reflective/insulating conditions at radial boundary for neutrons/temperature
- Variables: velocity (ur, uz), fluxes (2 group), temperature, precursor conc.



Concurrent efforts in Italy, Switzerland, DOE lab, UCB, and others

Goals

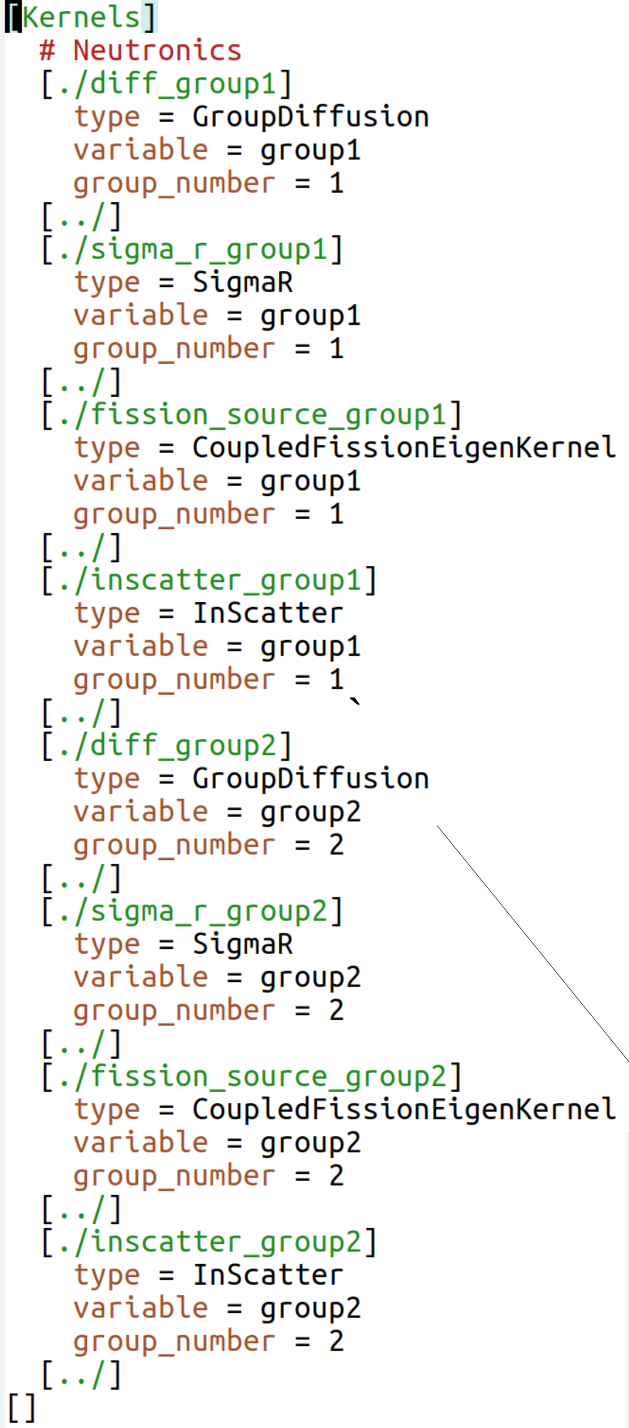
- Current:
  - demonstrate steady-state coupling of neutron fluxes, precursors, and temperature for thermal MSR design
- Future:
  - investigate reactor dynamics under transient accident scenarios to assess safety
  - explore salt processing and chemistry

Problem Set-Up

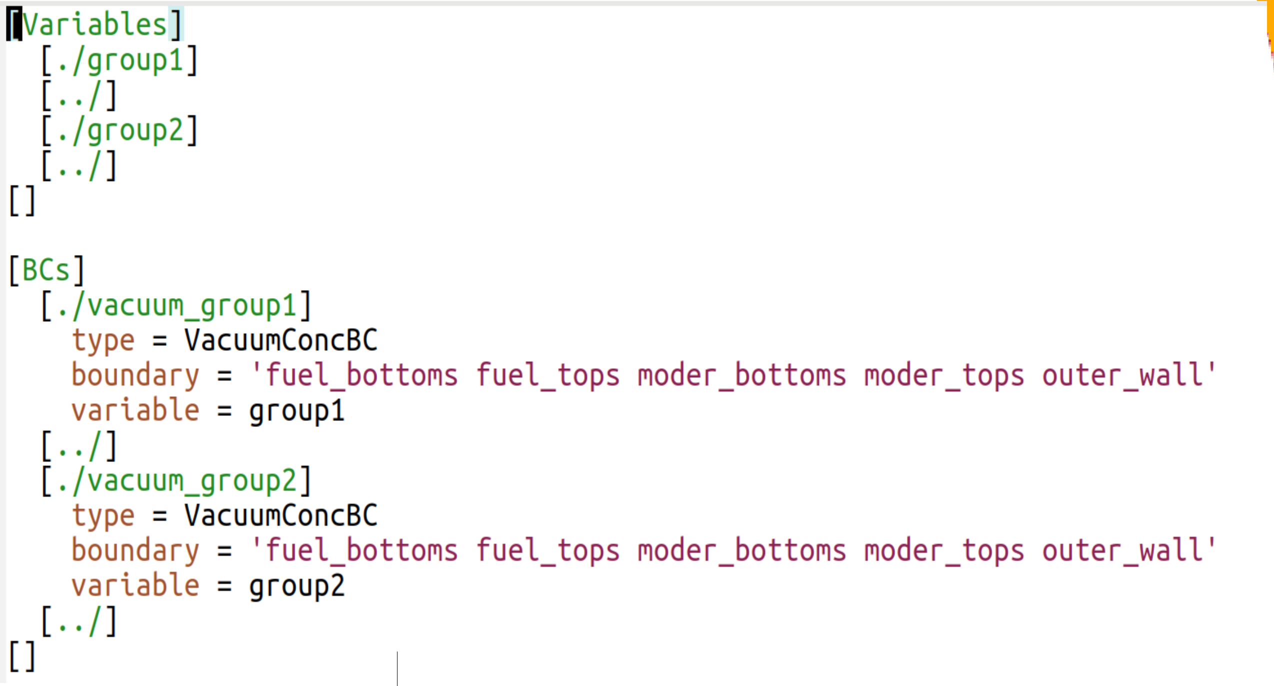
- Base current model on Molten Salt Reactor Experiment
- 22.5% volume fraction fuel
- Remainder graphite
- Fuel salt: Fluoride Lithium Beryllium (FLiBe) with 33% enriched Uranium
- Generate group constants with Serpent or Newt
- Moltres builds neutronics with action so accepts arbitrary number of groups; however, all results presented here will be for two groups

Component	Mass Fraction
Li-7	.1090
Li-6	5E-6
F-19	.6680
Be-9	.0627
U-235	.0167
U-238	.0344

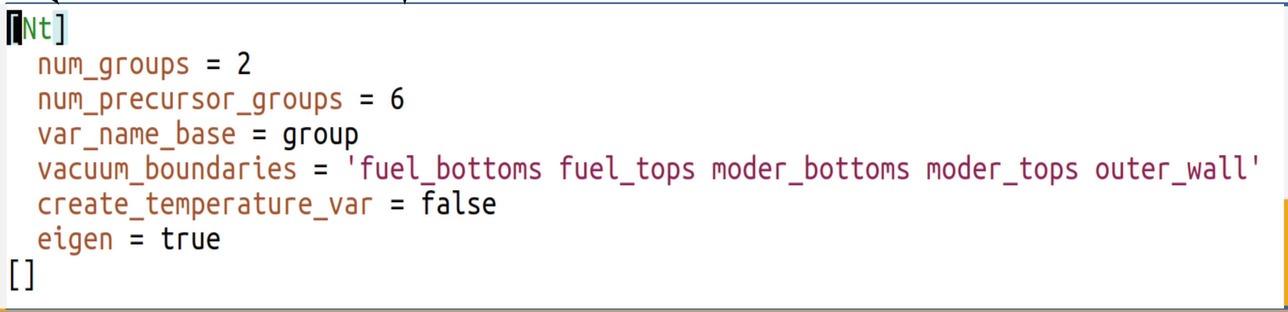
Action System: Simplified Input



```
[Kernels]
  # Neutronics
  [./diff_group1]
    type = GroupDiffusion
    variable = group1
    group_number = 1
  [../]
  [./sigma_r_group1]
    type = SigmaR
    variable = group1
    group_number = 1
  [../]
  [./fission_source_group1]
    type = CoupledFissionEigenKernel
    variable = group1
    group_number = 1
  [../]
  [./inscatter_group1]
    type = InScatter
    variable = group1
    group_number = 1
  [../]
  [./diff_group2]
    type = GroupDiffusion
    variable = group2
    group_number = 2
  [../]
  [./sigma_r_group2]
    type = SigmaR
    variable = group2
    group_number = 2
  [../]
  [./fission_source_group2]
    type = CoupledFissionEigenKernel
    variable = group2
    group_number = 2
  [../]
  [./inscatter_group2]
    type = InScatter
    variable = group2
    group_number = 2
  [../]
[]
```



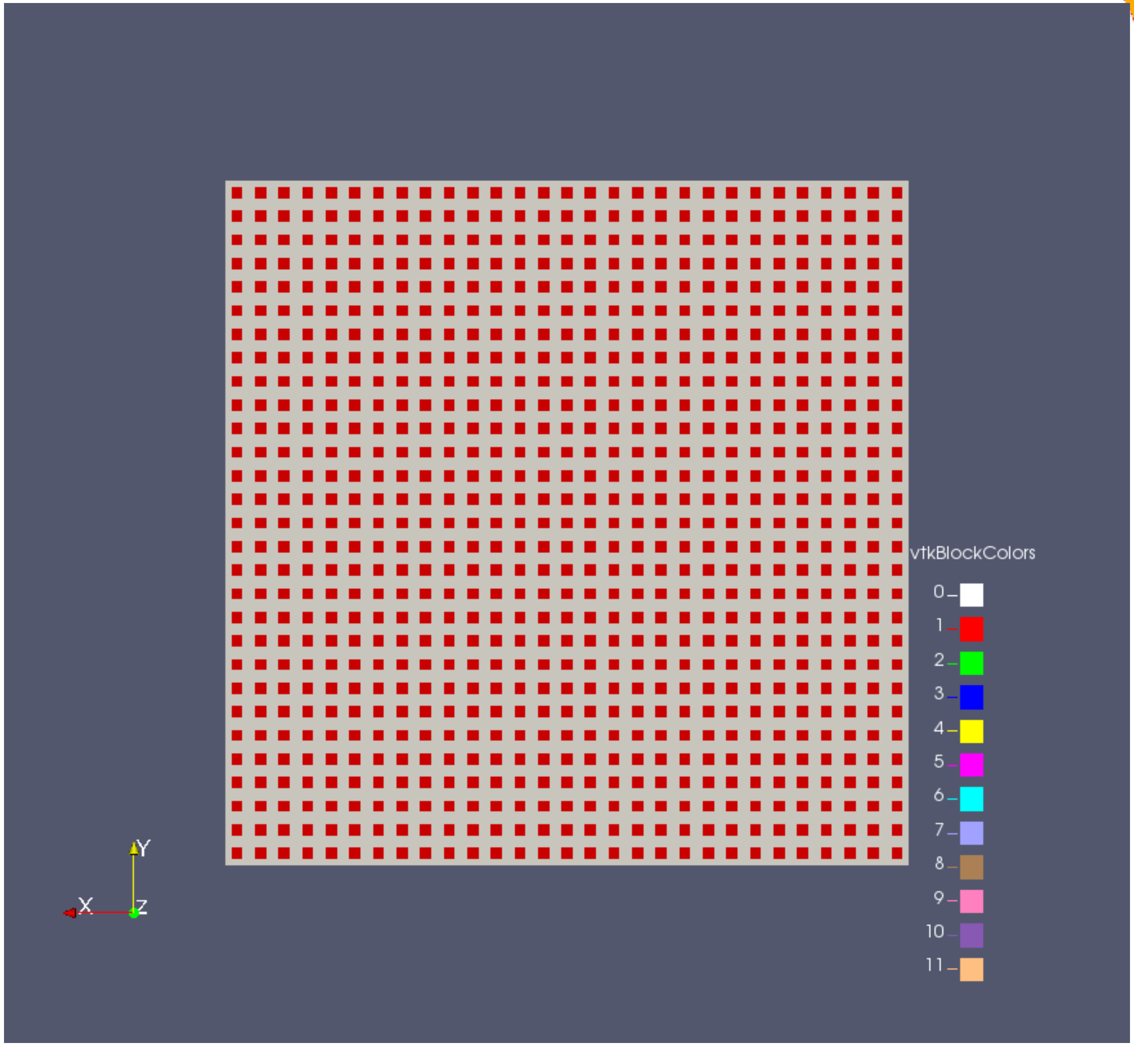
```
[Variables]
  [./group1]
  [../]
  [./group2]
  [../]
[]

[BCs]
  [./vacuum_group1]
    type = VacuumConcBC
    boundary = 'fuel_bottoms fuel_tops moder_bottoms moder_tops outer_wall'
    variable = group1
  [../]
  [./vacuum_group2]
    type = VacuumConcBC
    boundary = 'fuel_bottoms fuel_tops moder_bottoms moder_tops outer_wall'
    variable = group2
  [../]
[]
```



```
[Nt]
  num_groups = 2
  num_precursor_groups = 6
  var_name_base = group
  vacuum_boundaries = 'fuel_bottoms fuel_tops moder_bottoms moder_tops outer_wall'
  create_temperature_var = false
  eigen = true
[]
```

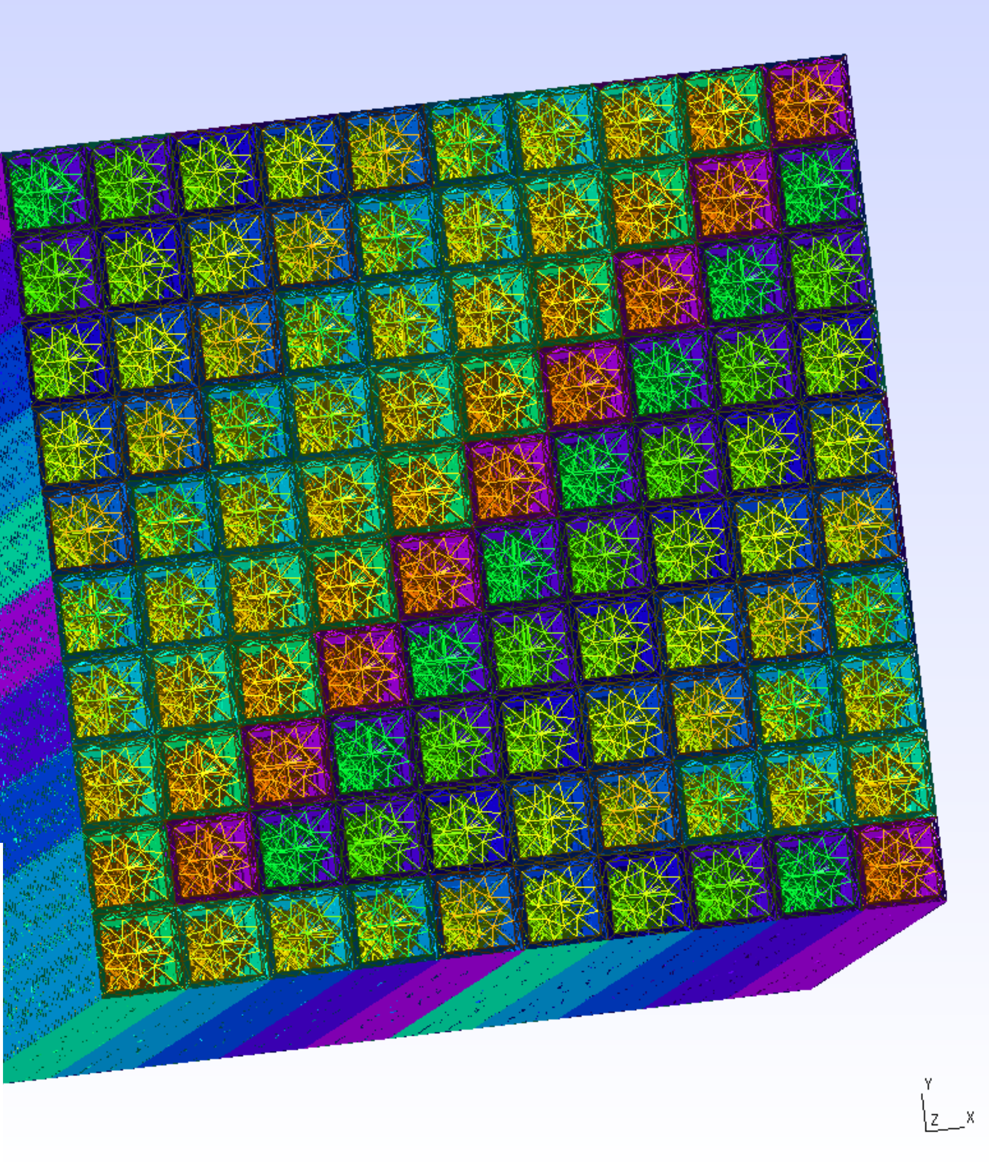
3D Geometry

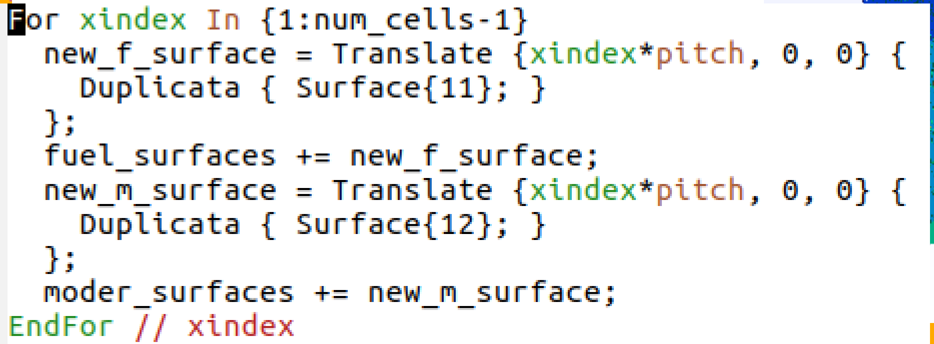
- Start with simple cuboid lattice
- Red regions fuel
- 5 cm pitch



Mesh Construction

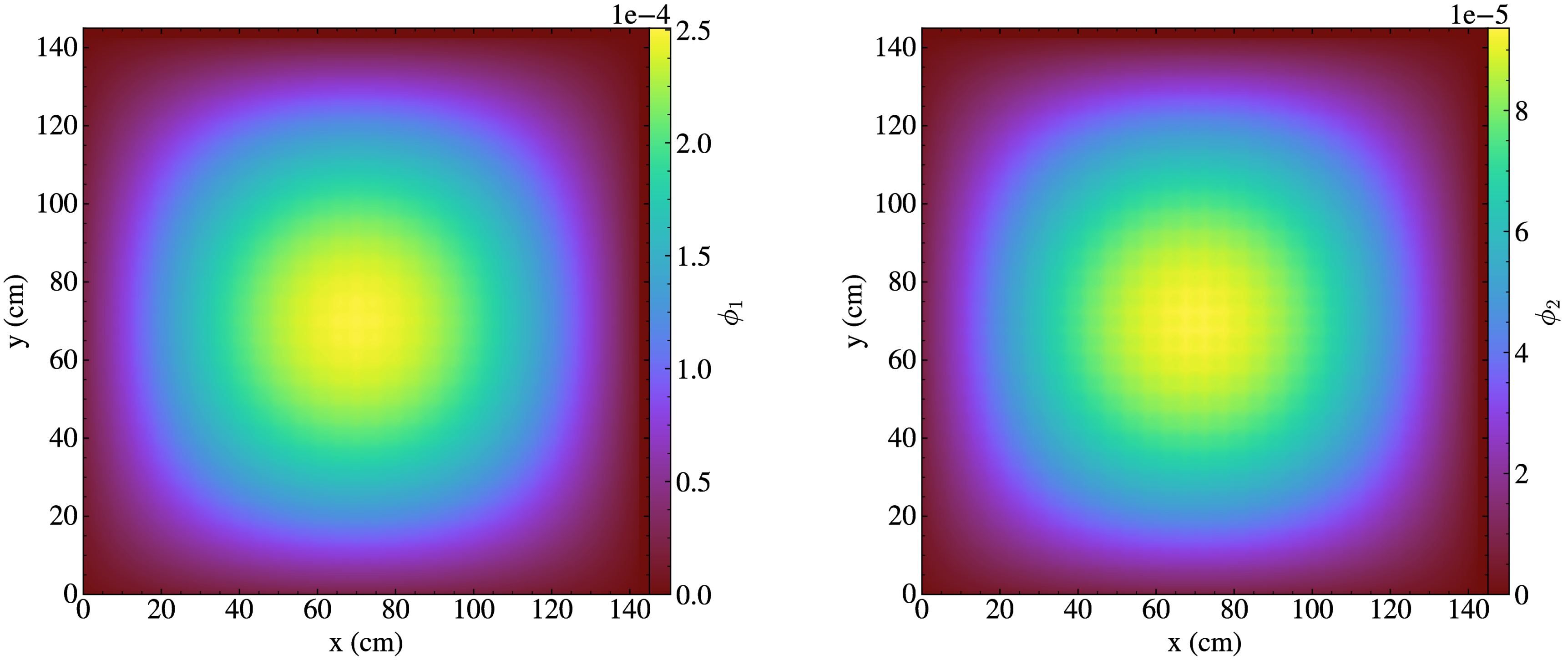
- Repeating structure, block and boundary IDs, generated using gmsh





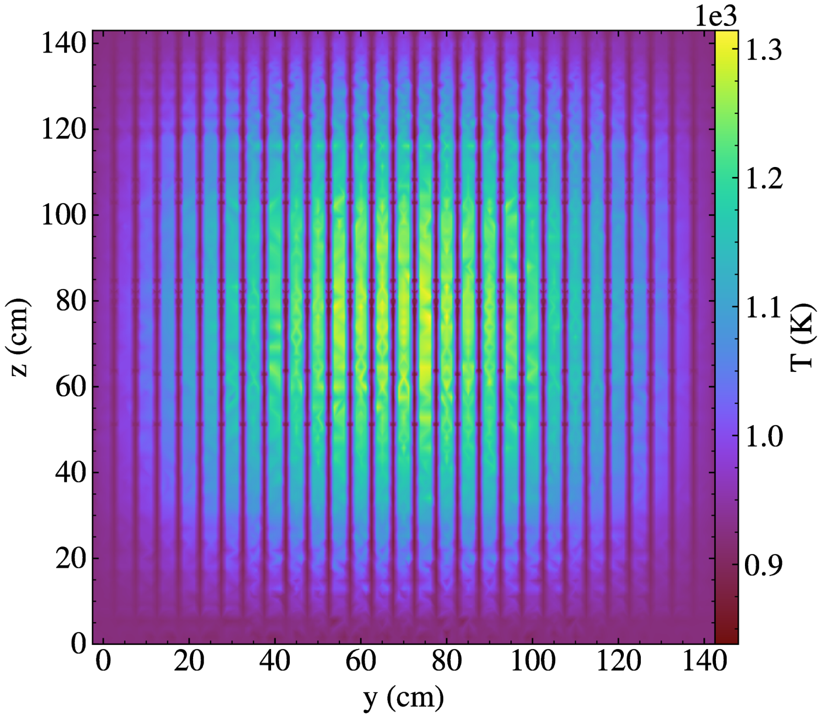
```
For xindex In {1:num_cells-1}
  new_f_surface = Translate {xindex*pitch, 0, 0} {
    Duplicata { Surface{11}; }
  };
  fuel_surfaces += new_f_surface;
  new_m_surface = Translate {xindex*pitch, 0, 0} {
    Duplicata { Surface{12}; }
  };
  moder_surfaces += new_m_surface;
EndFor // xindex
```

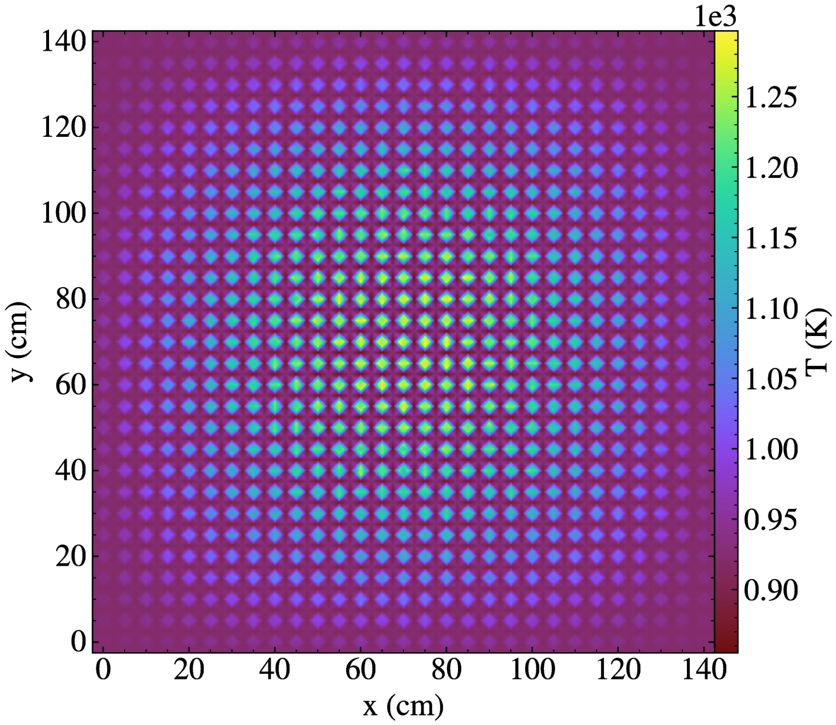
k-eigenvalue simulation; reactor axial mid-plane



T = 922 K; k = 1.002

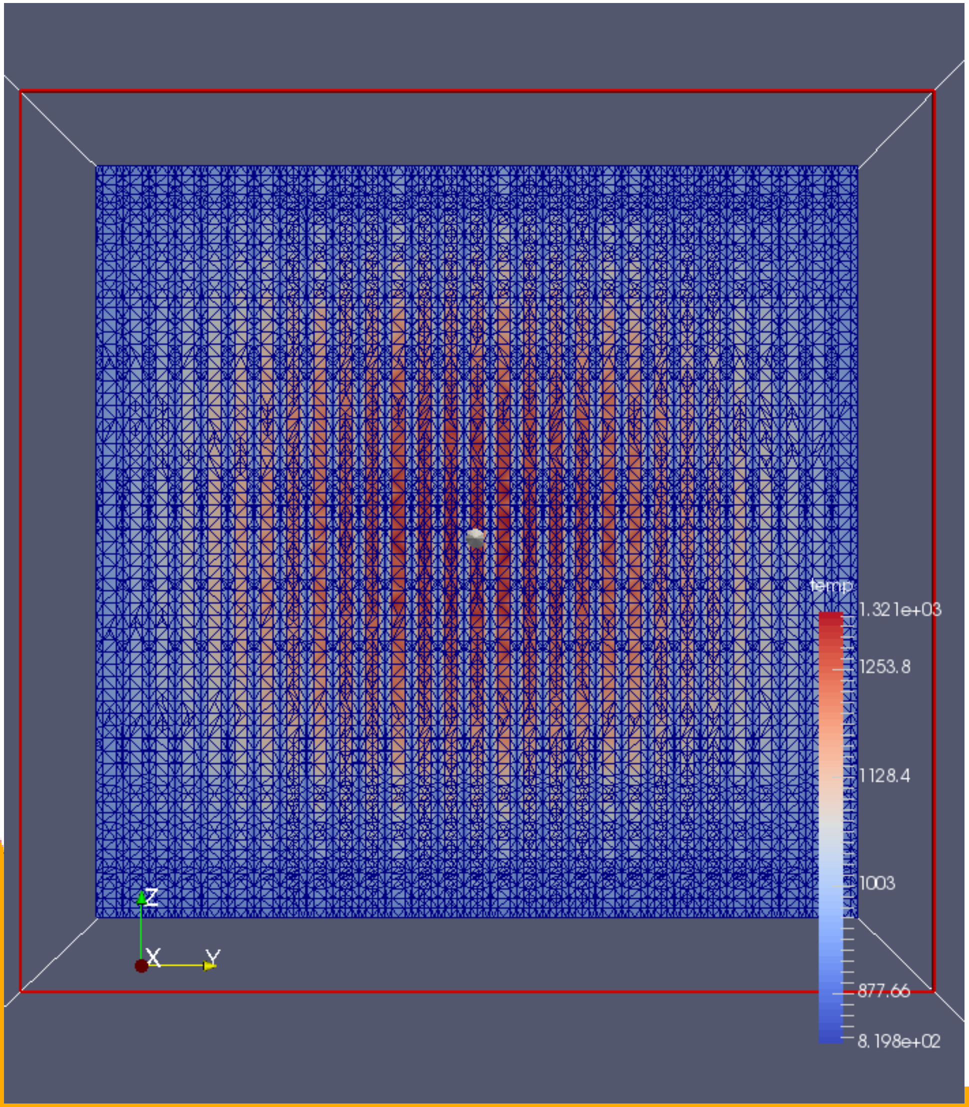
Issues when Coupling in Temperature



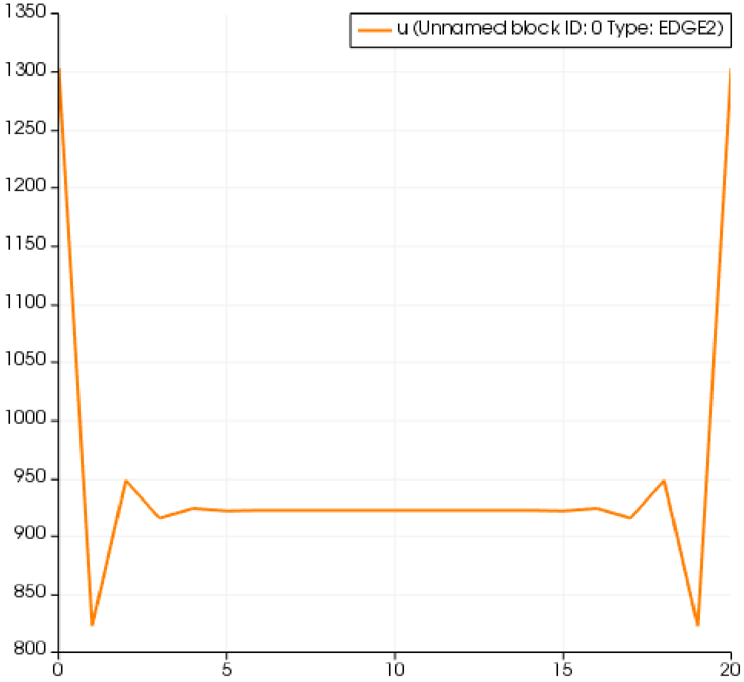


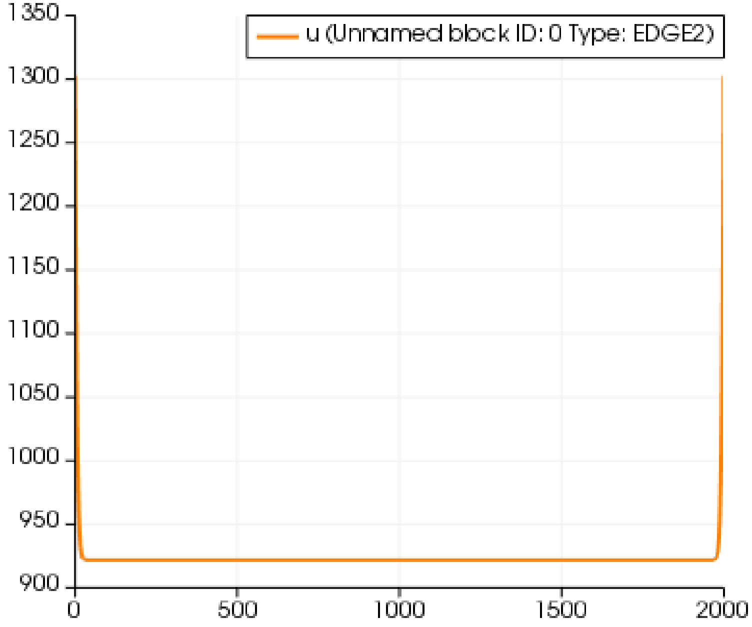
Moderator temperatures next to highest fissioning fuel regions drop below inlet temperature of 922 K

Boundary Shocks



- Over 1 million elements in simulation, but only one internal node spans radial moderator dimension





Boundary Shocks

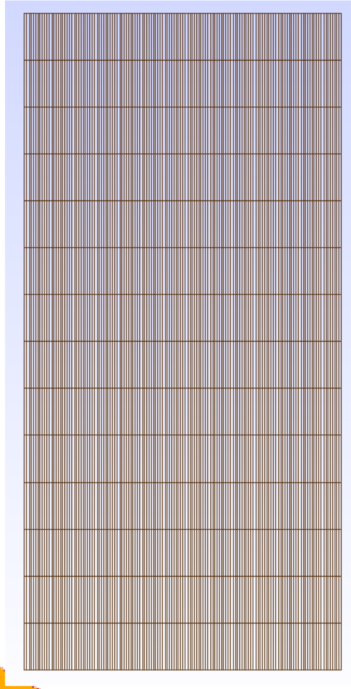
- Can see identical phenomena in simple Transient-Diffusion problem with quickly ramped boundary values

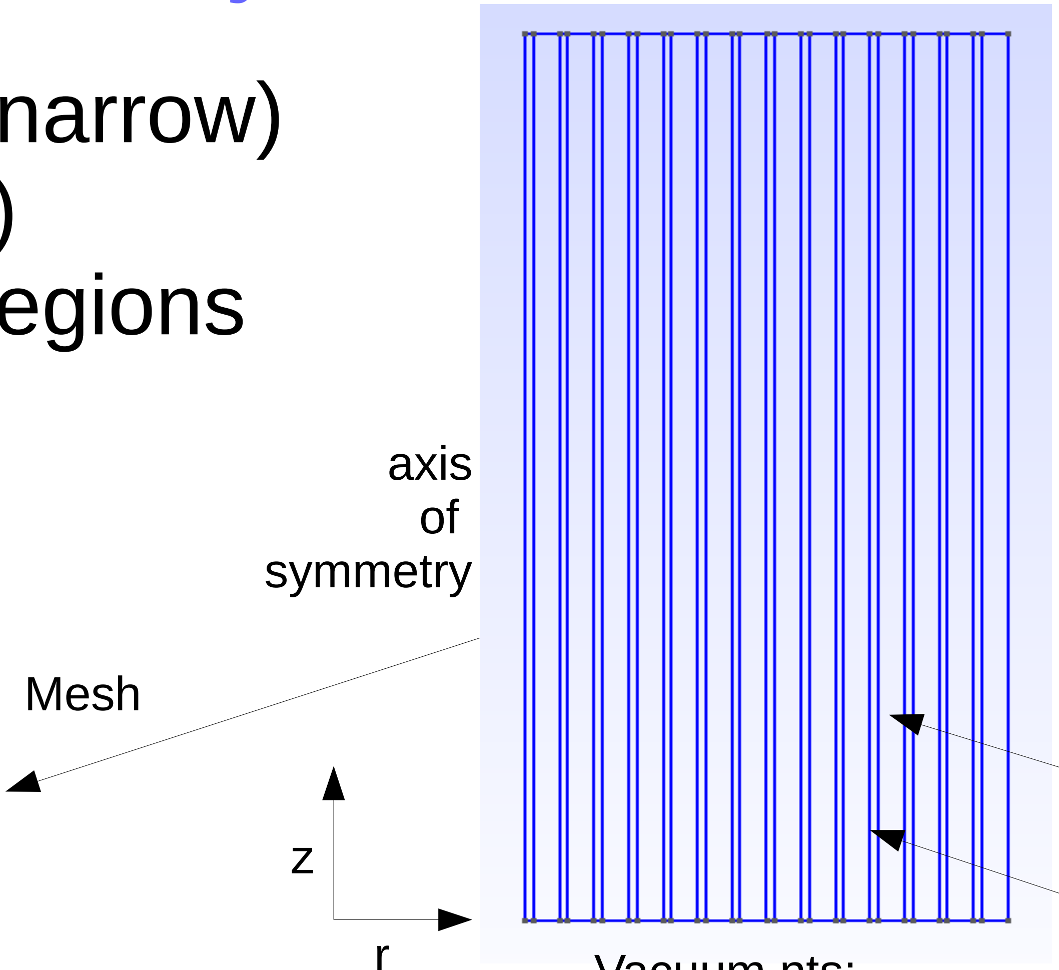
Moving Forward

- Move to simplified 2D-axisymmetric formulation, using same group constants and maintaining correct fuel-moderator volume fractions
- Structured mesh: allows much finer meshing in radial direction where gradients are much larger
- Return later to 3D problem, again using structured mesh

2D-axi geometry

- Repeating (narrow) fuel – (wide) moderator regions





Vacuum nts;
T, Pres:
OutflowBC

axis
of
symmetry

Mesh

z

r

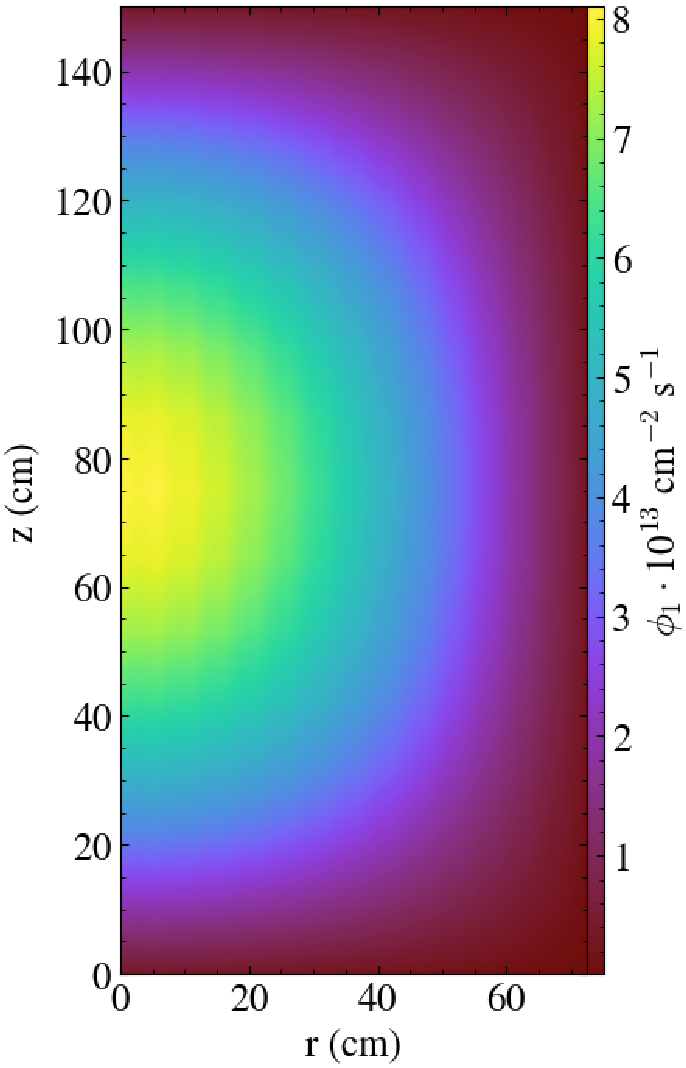
Vacuum nts;
T = 922
(downcomer
cooling)

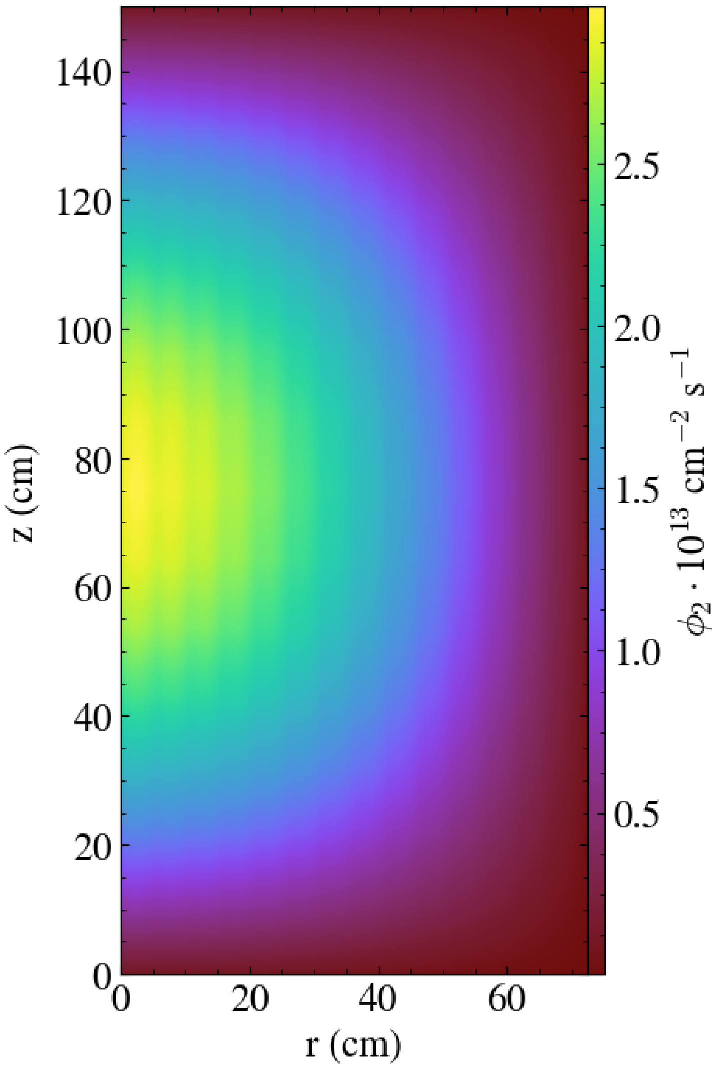
graphite

fuel

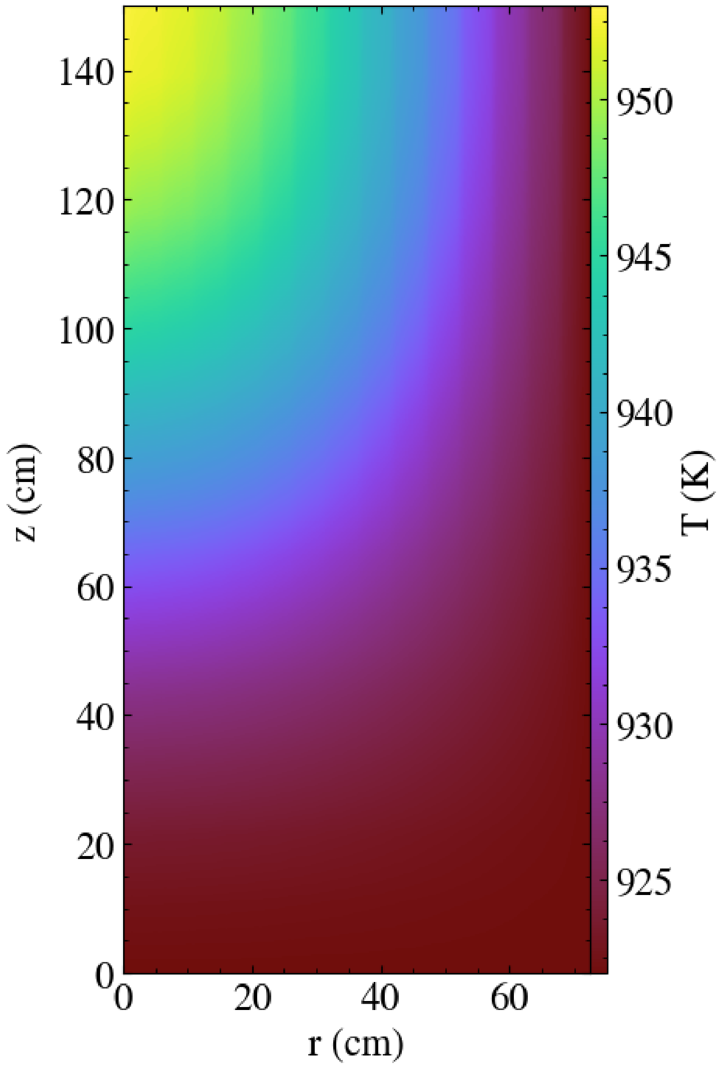
Vacuum nts;
T = 922;
Pre = 0 (inlet)

Results (neutrons)

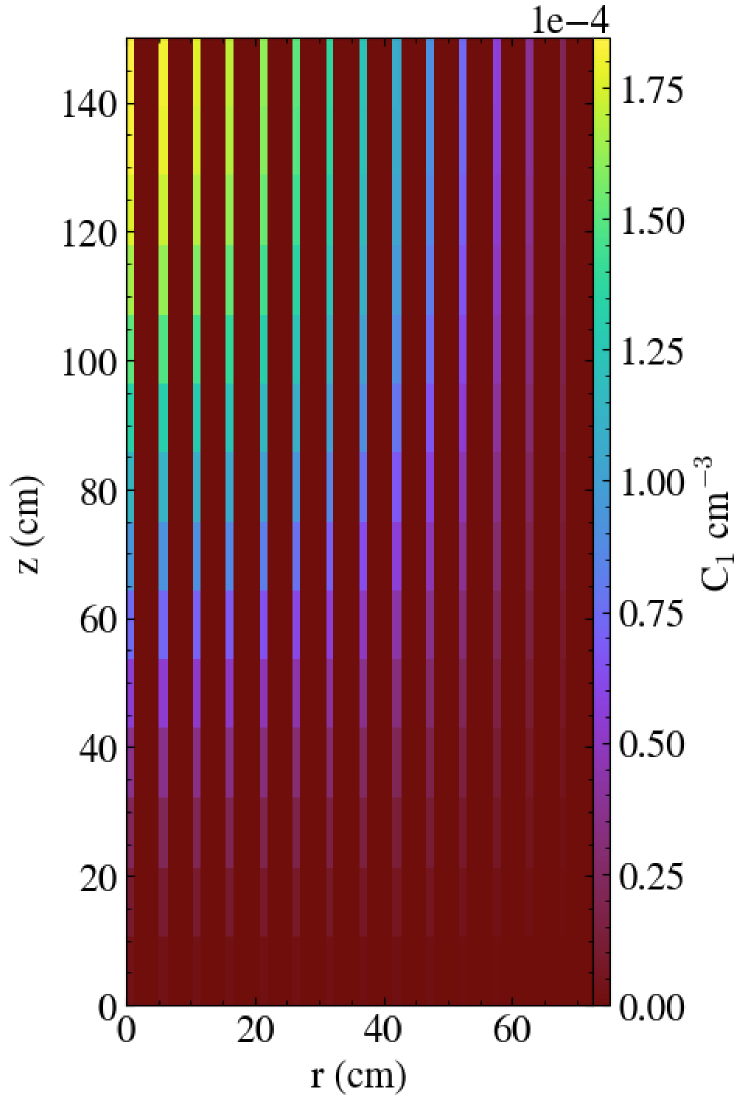


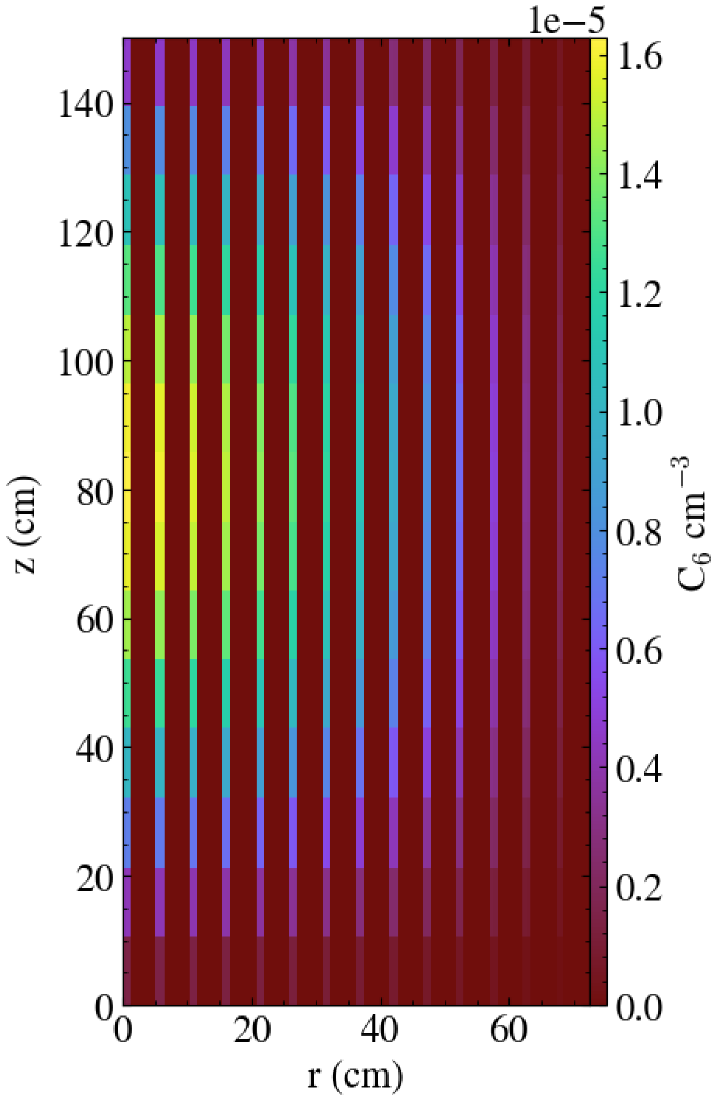


Results (temperature)



Results (precursors)

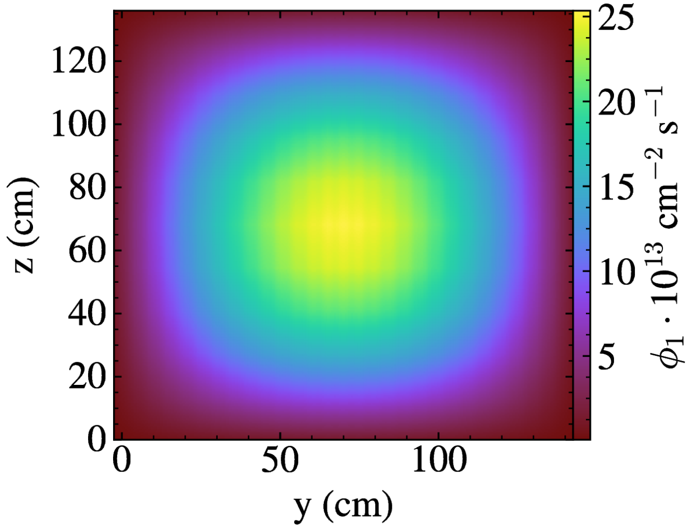




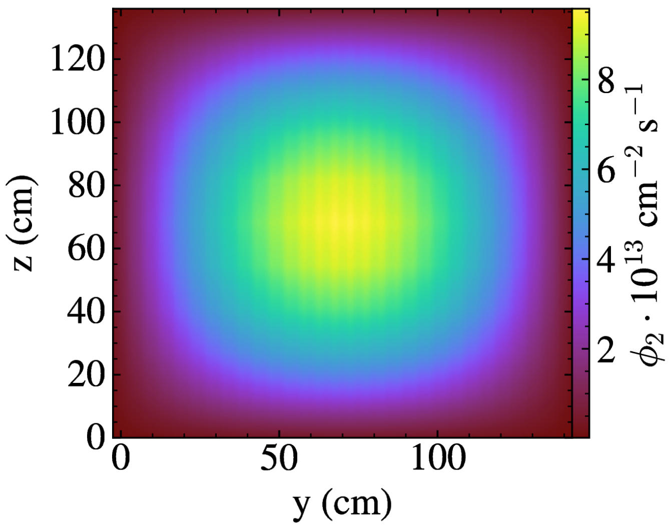
New 3D Results

- Lessons learned from 2D-axisymmetric case: using ***structured*** 3D mesh, can evolve close to steady-state without observing boundary shocks
- Over heat conduction time scales, convergence slow
  - Unsure why
- Simulation run on 160 processors on Illinois's Blue Waters supercomputer

Results (neutrons)

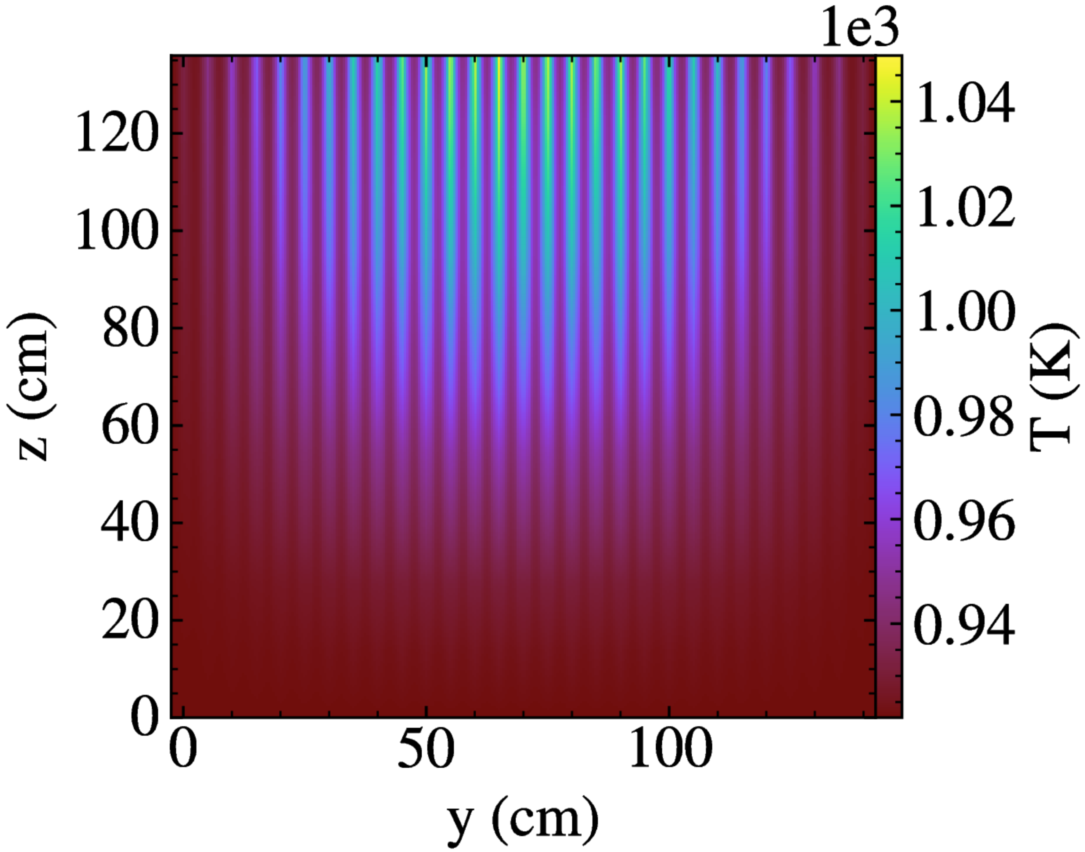


Fast



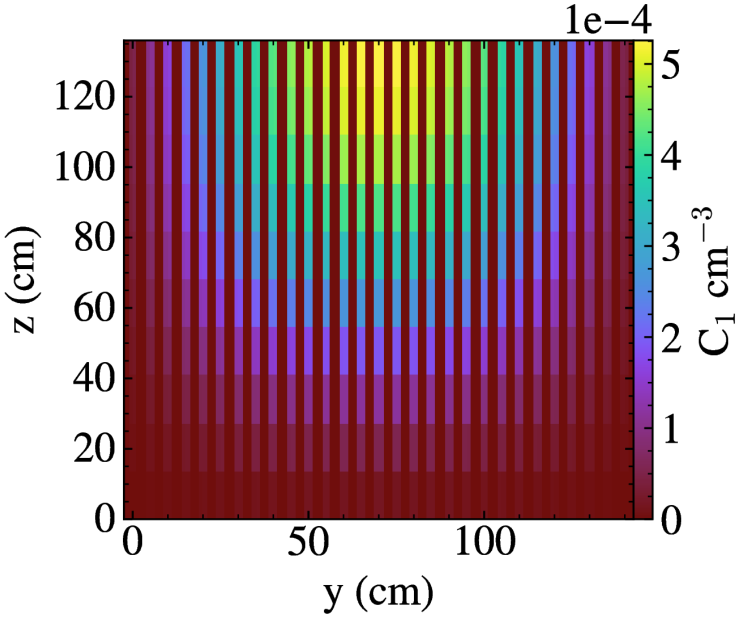
Thermal

Results (temperature)

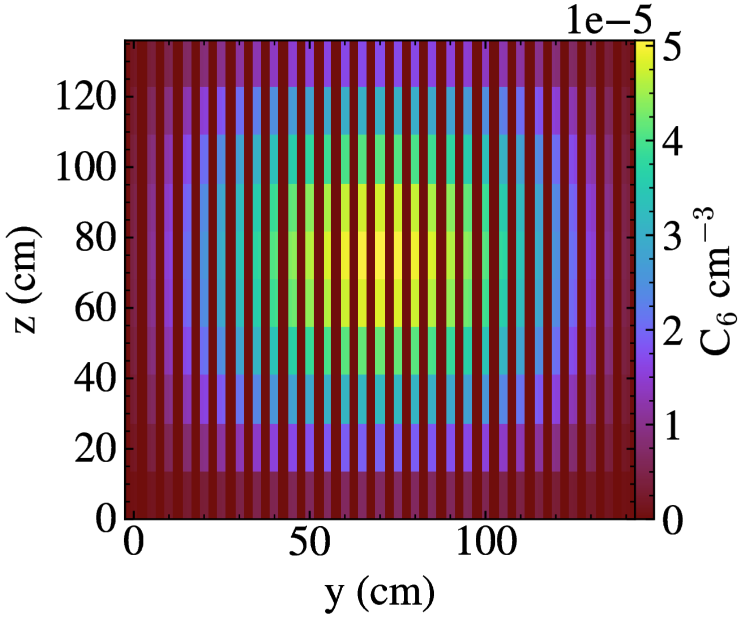


- Not simulated out to long enough times for heat to finish conducting from fuel to moderator
- End time = 17 seconds
- Characteristic diffusion time through moderator = 24 seconds

Results (precursors)



Longest lived precursor



Shortest lived precursor

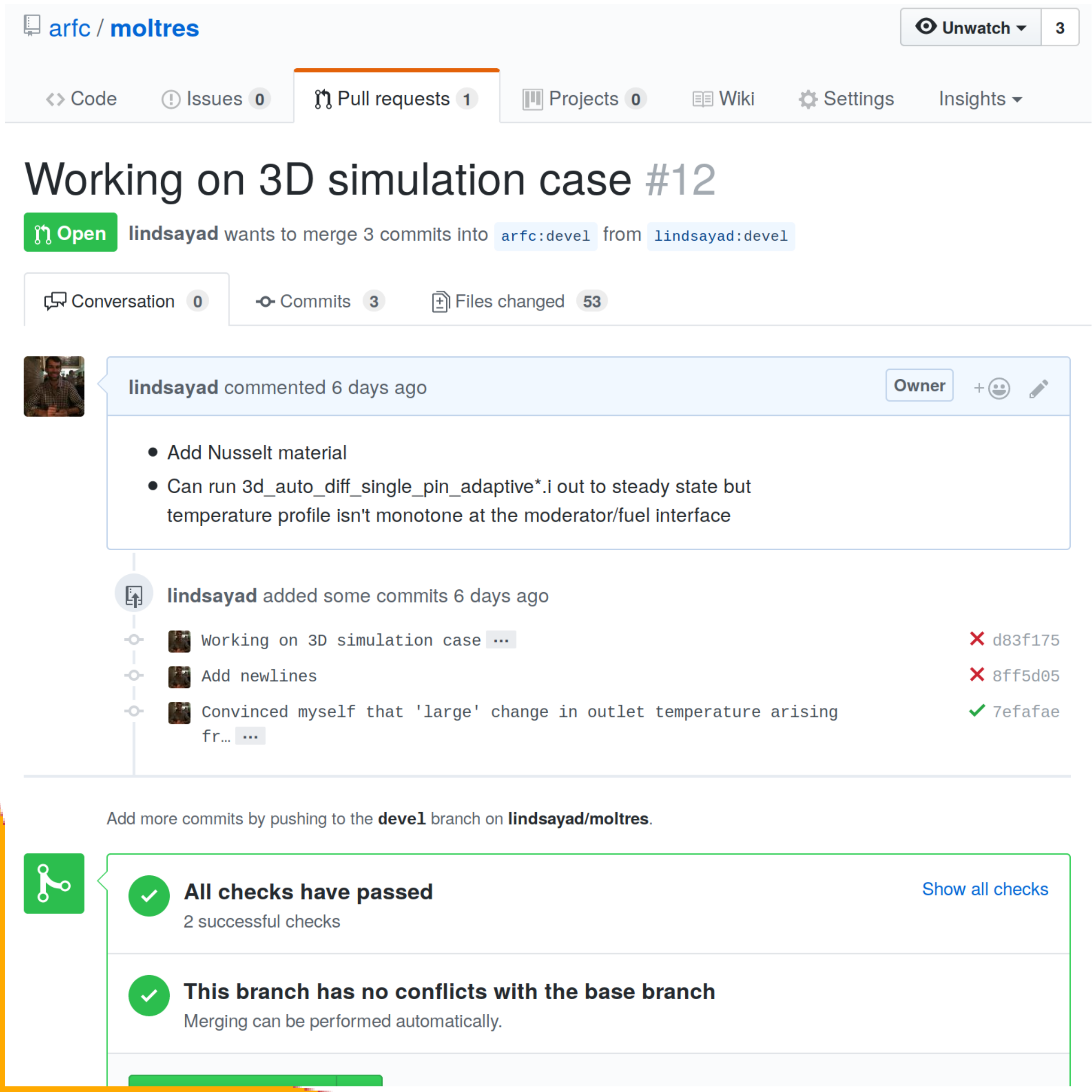
Moltres Summary/Future Work

- Good coupling with fine meshing in 2D
- Some good 3D results; slow convergence at long time scales
- Next:
  - Instead of assuming constant, couple in flow variables
  - Start exploring transients, e.g. explore responses to reactivity insertion
  - Implement cross section dependence on control rods

Moltres Summary/Future Work

- Presented preliminary results
- Still a bit to go to replicate state-of-the-art multi-physics simulation capabilities
- However, we hope use of modern development practices will accelerate growth and assist in review and transparency

GitHub with Continuous Integration



arfc / **moltres**

Unwatch ▾ 3

Code | Issues 0 | Pull requests 1 | Projects 0 | Wiki | Settings | Insights ▾

Working on 3D simulation case #12

Open **lindsayad** wants to merge 3 commits into `arfc:devel` from `lindsayad:devel`

Conversation 0 | Commits 3 | Files changed 53

lindsayad commented 6 days ago Owner

- Add Nusselt material
- Can run 3d_auto_diff_single_pin_adaptive*.i out to steady state but temperature profile isn't monotone at the moderator/fuel interface

lindsayad added some commits 6 days ago

```
Working on 3D simulation case …                                  ✗ d83f175
Add newlines                                                     ✗ 8ff5d05
Convinced myself that 'large' change in outlet temperature arising  ✓ 7efafae
fr… …
```

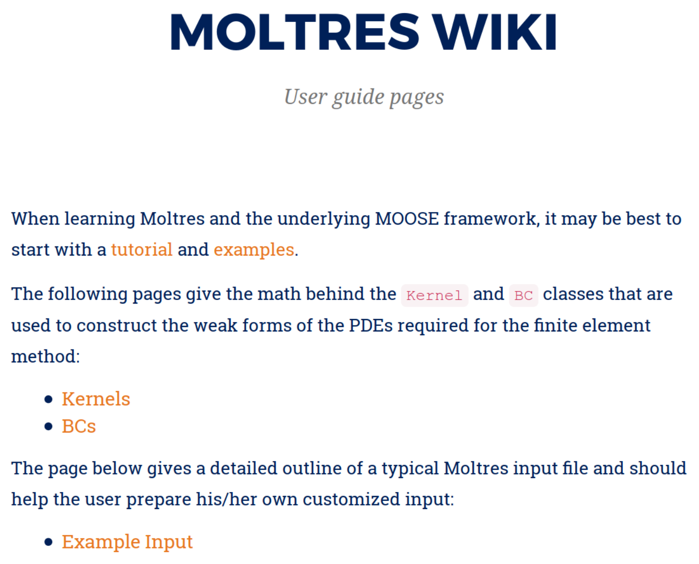
Add more commits by pushing to the **devel** branch on **lindsayad/moltres**.

✓ **All checks have passed** Show all checks

2 successful checks

✓ **This branch has no conflicts with the base branch**

Merging can be performed automatically.



MOLTRES WIKI

User guide pages

When learning Moltres and the underlying MOOSE framework, it may be best to start with a tutorial and examples.

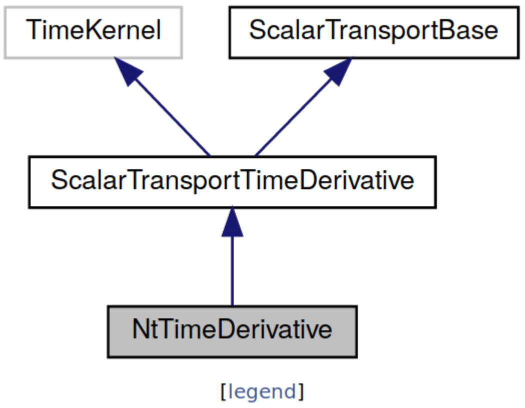
The following pages give the math behind the `Kernel` and `BC` classes that are used to construct the weak forms of the PDEs required for the finite element method:

- Kernels
- BCs

The page below gives a detailed outline of a typical Moltres input file and should help the user prepare his/her own customized input:

- Example Input

Wiki with User Guides



Doxygen pages for helping new developers

Acknowledgements

- MOOSE team
- Andrei Rykhlevskii