INL/MIS-23-73514-Revision-0

Establishing a Methodology for Performing a Multiphysics Run-In Analysis of the GPBR200

July 2020

Changing the World's Energy Future

Ryan Hunter Stewart, Paolo Balestra, Jack M Cavaluzzi

Idaho National Laboratory

INL/MIS-23-73514-Revision-0

Establishing a Methodology for Performing a Multiphysics Run-In Analysis of the GPBR200

Ryan Hunter Stewart, Paolo Balestra, Jack M Cavaluzzi

July 2020

Idaho National Laboratory
Idaho Falls, Idaho 83415

http://www.inl.gov

Prepared for the
U.S. Department of Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

July 25, 2020

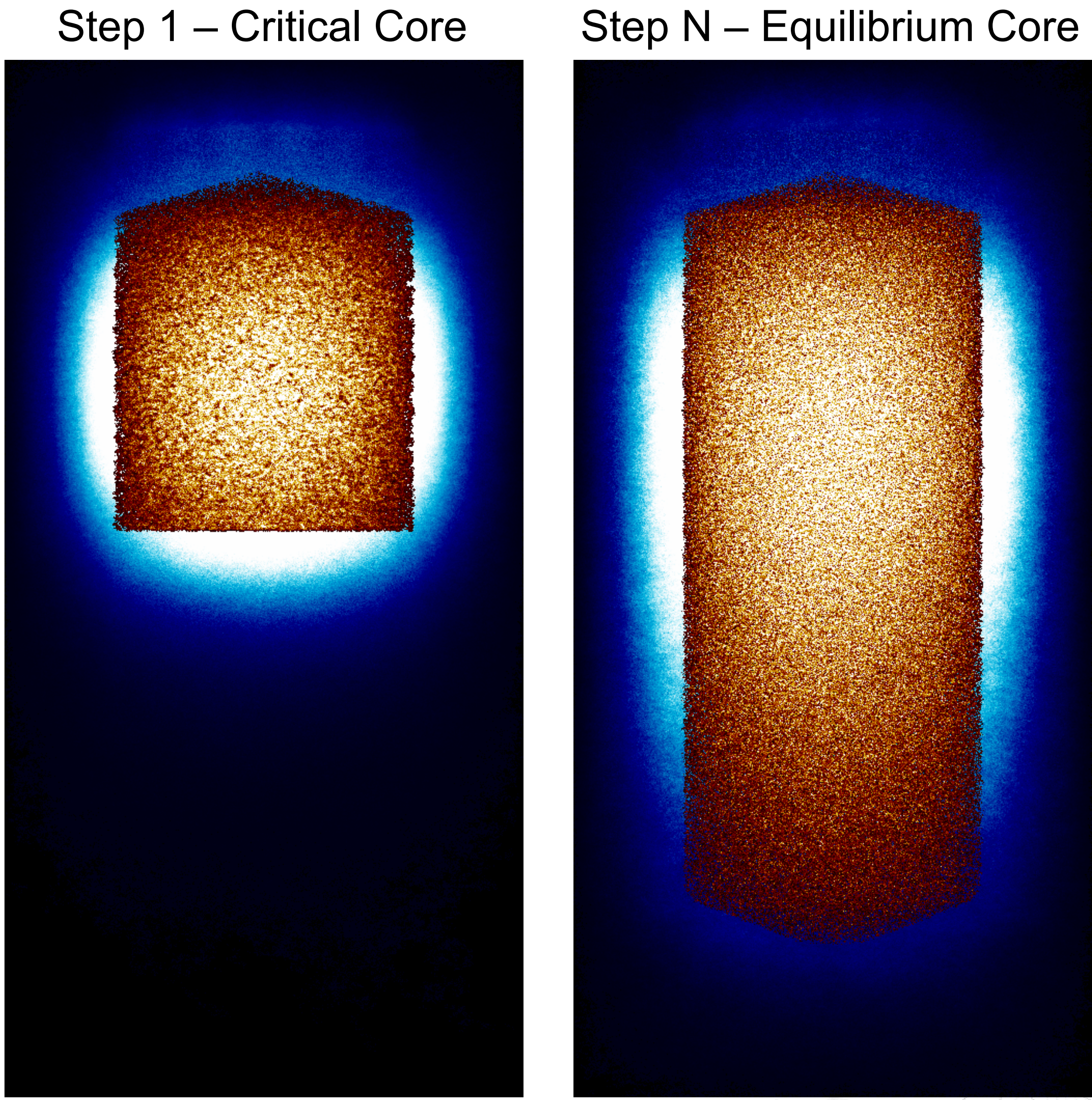
Ryan Stewart, Jack Cavaluzzi, Paolo Balestra

Establishing a Methodology for Performing a Multiphysics Run-In Analysis of the GPBR200

ART
Advanced Reactor Technologies

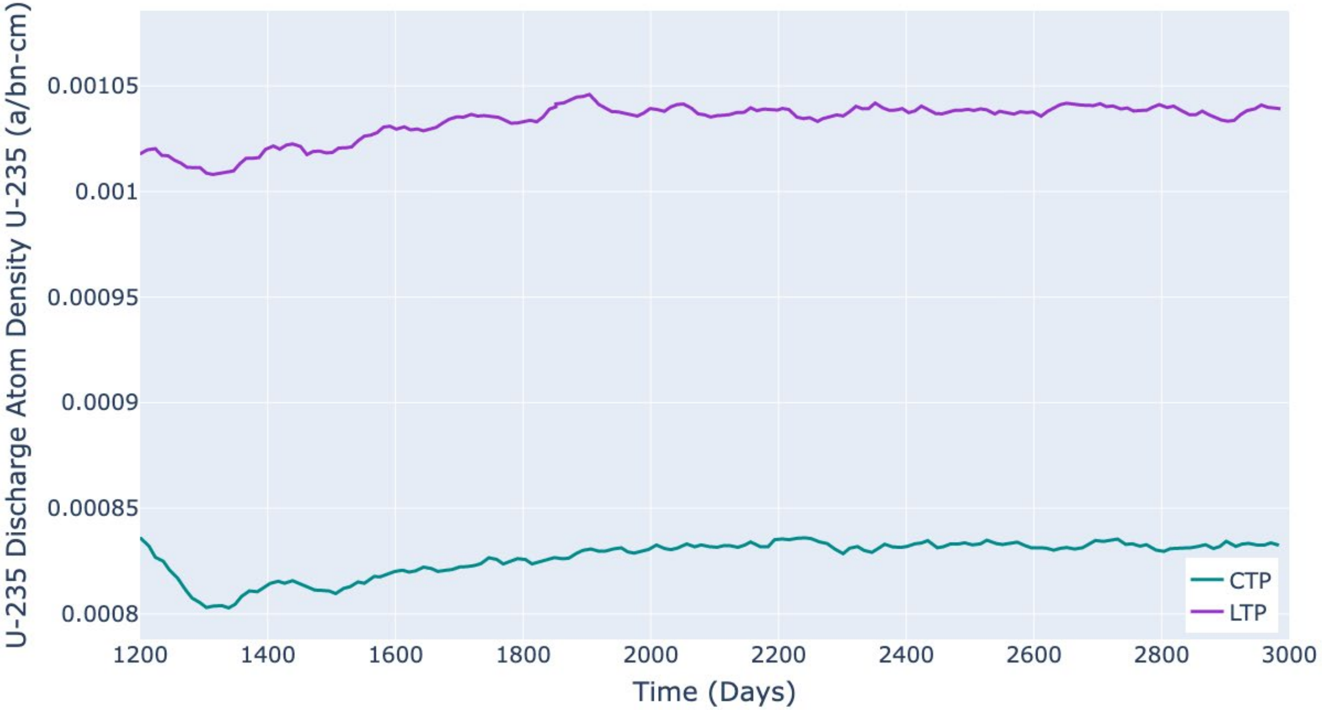
Introduction

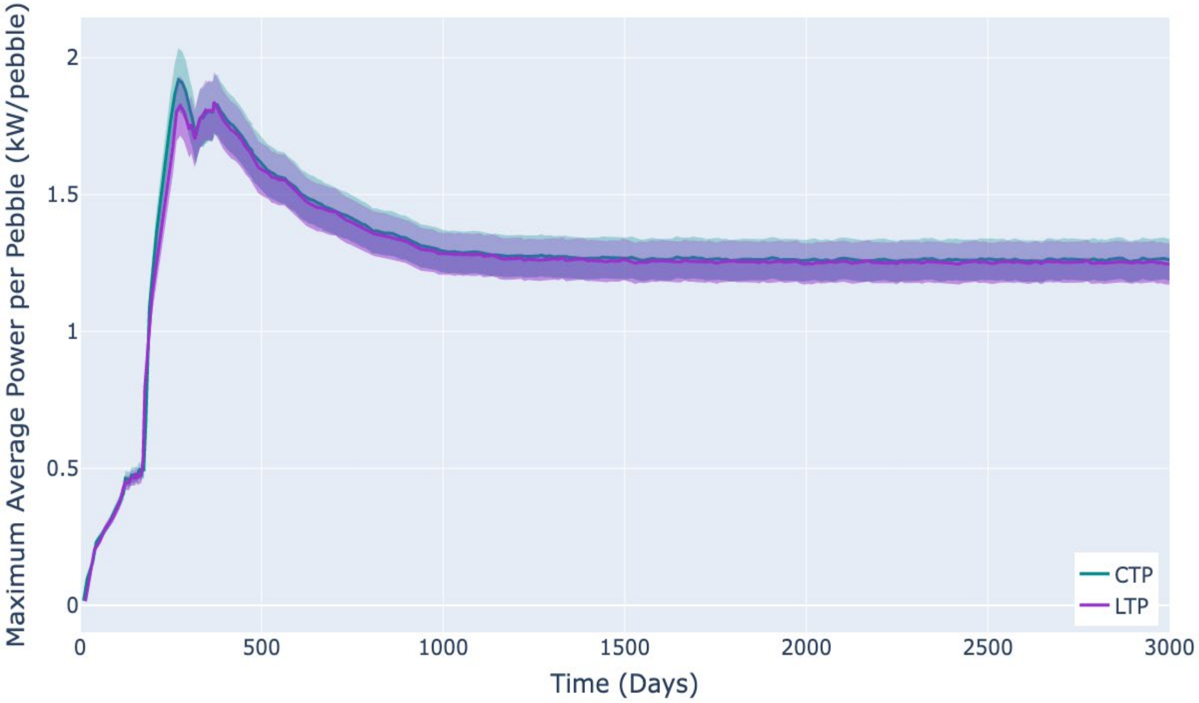
- The running-in phase of a pebble-bed reactor (PBR) is a complex time-dependent problem
  - Involves the use of multiple fuel types, graphite pebbles and a ramp-up of power
- Modeling this problem using high-fidelity simulation tools allows us to examine multiple physical phenomena that is important to PBR operations
  - quantities of interest: discharge burnup, time to full power, pebble power peaking, etc.
- Understanding the temperature distribution will provide insight into key physics missing in isothermal core analysis



Justification of Research

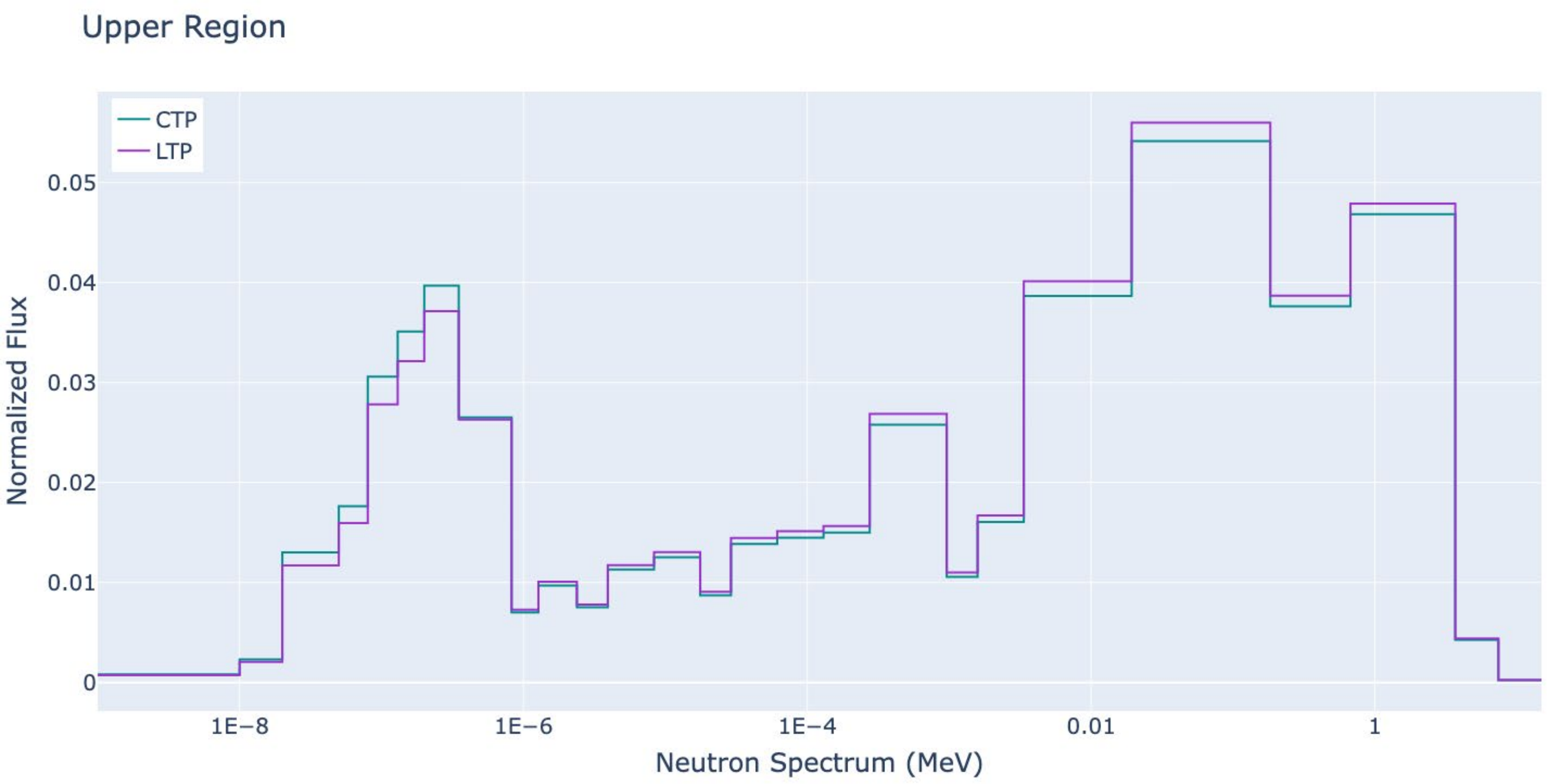
- Previous research showed that temperature can influence core neutronics
  - Isothermal vs linear temperature distribution

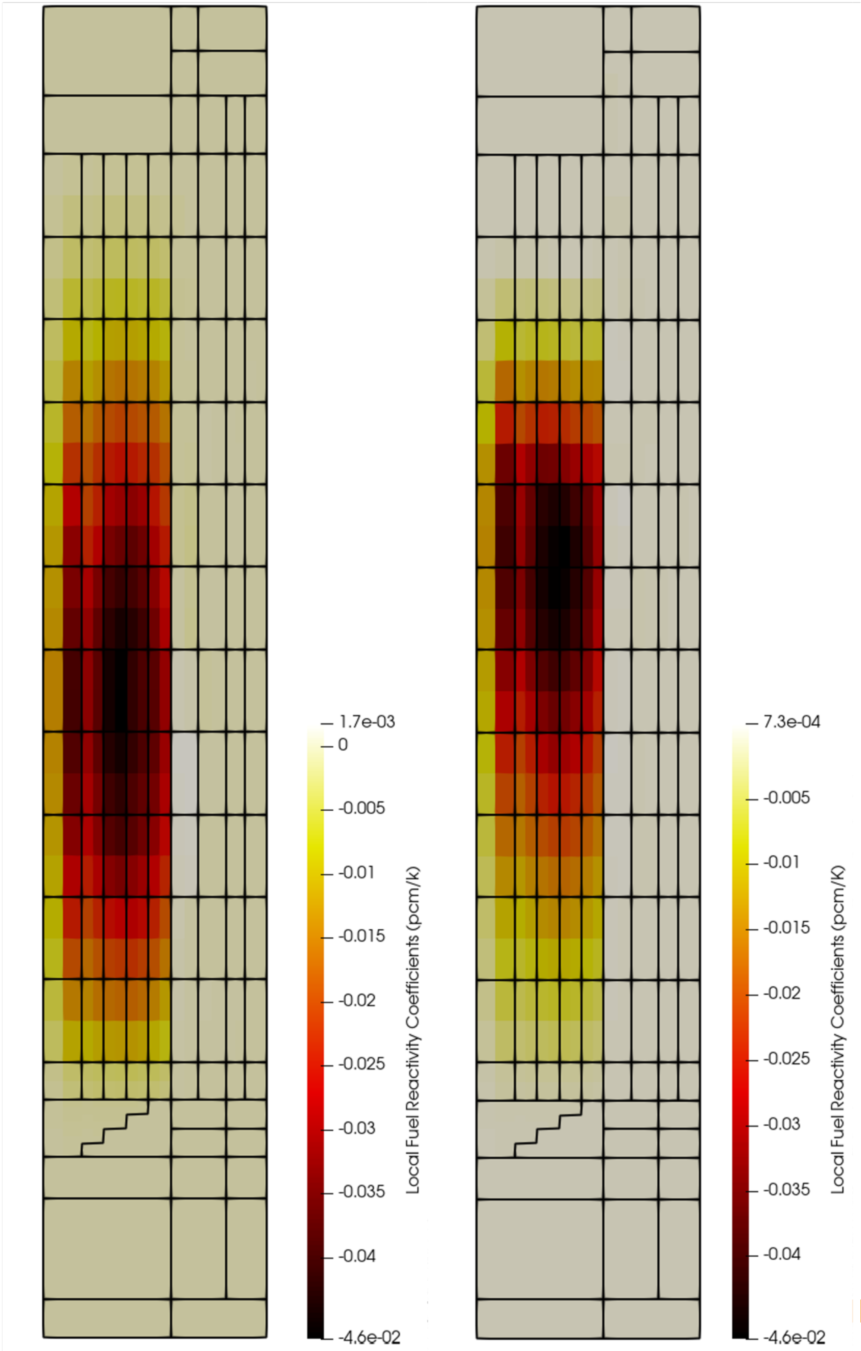




Justification of Research

- Previous research showed that temperature can influence core neutronics
  - Isothermal vs linear temperature distribution
- Temperature reactivity coefficients





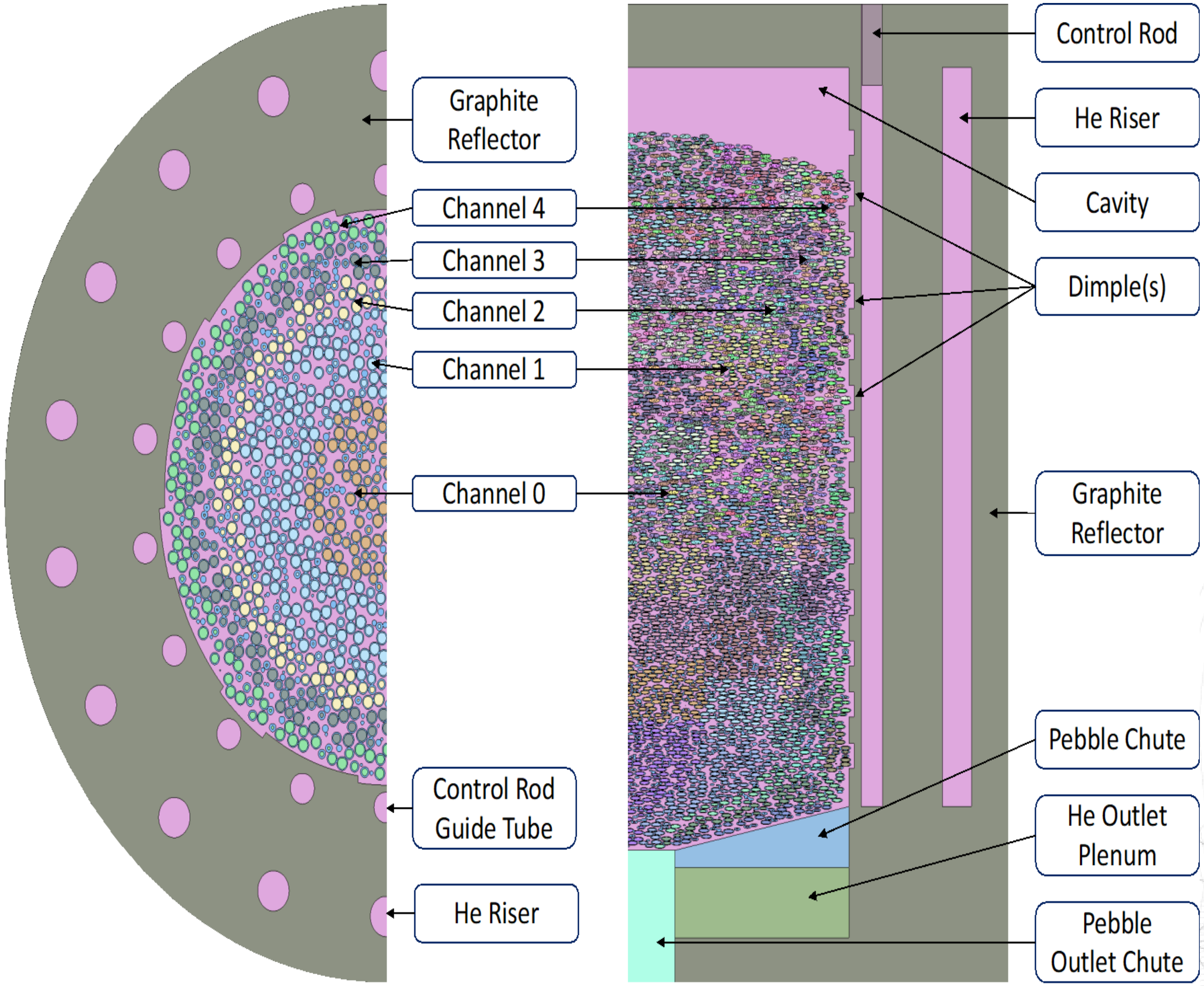
Project Goal

- **Goal**: Develop a methodology for incorporating multi-physics into our high-fidelity run-in simulation.
- **Approach**: Utilize a Python module wrapped around Serpent and Griffin/Pronghorn to simulate pebble movement through the core

Algorithm Outline

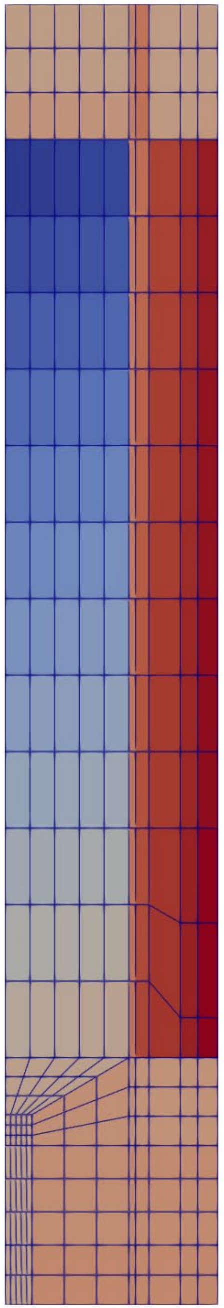
1. Perform burn-up step (Serpent)
2. Determine temperature distribution (Griffin/Pronghorn)
3. Shift pebbles down (Python)
4. Recycle/discharge pebbles (Python)
5. Refuel core (Python)
6. Update power, temperature, pebble type, etc. (Python)

GPBR200 Model: Serpent



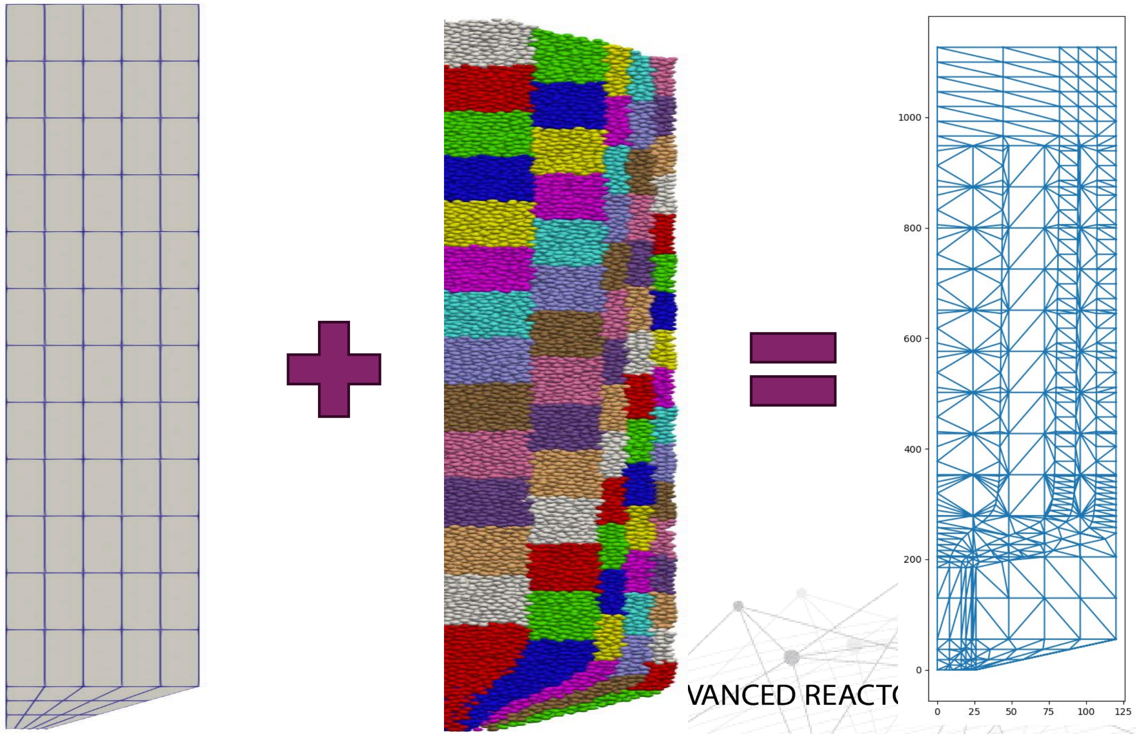
GPBR200 Model: Griffin/Pronghorn

- Captured as much of the geometry into a 2D RZ mesh
  - Varying graphite density to capture lower plenum
  - Control rod and risers implemented
- Core region
  - No streamlines (60 blocks within the core)
  - Conus region is modeled separately
- Note: Griffin blocks **do not** align with the Serpent axial volumes
  - Requires a mapping between the two



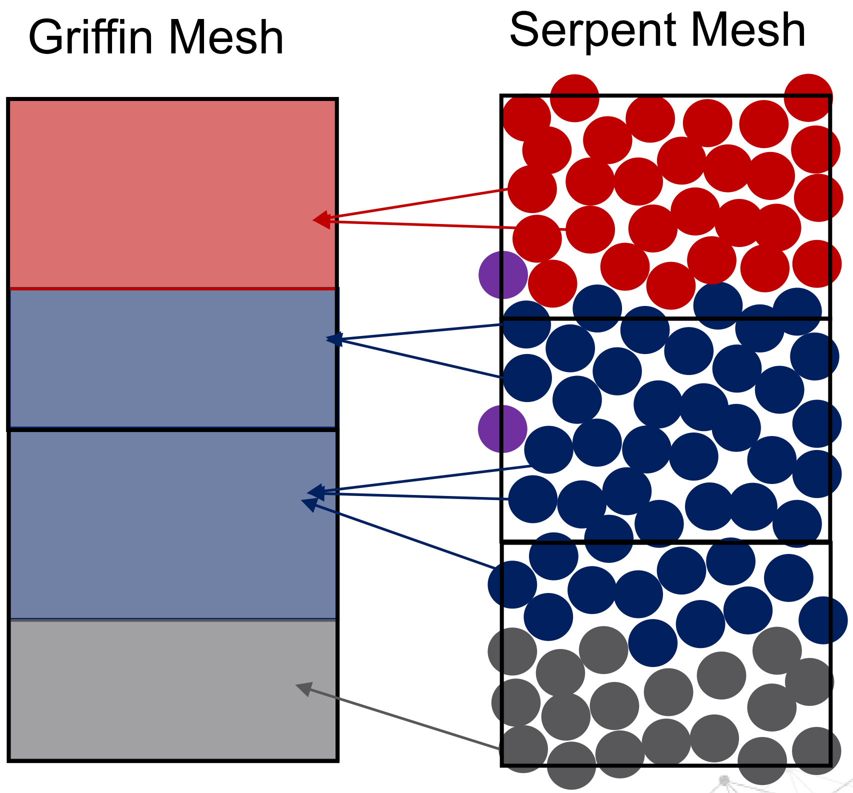
Data Transfer: Initial Attempt

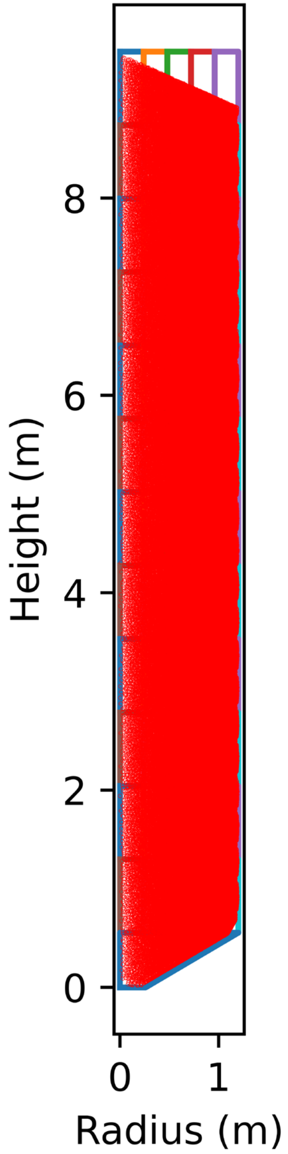
- Create a unifying mesh utilizing triangularization
  - Triangles can't cross mesh in either individual mesh (Griffin or Serpent)
- Map data in each triangle to correspond to a specific Griffin/Serpent mesh
- Volume average number densities to pass information back and forth
- Triangulation mapping became difficult
  - Mainly when vertices and segments approached each other
- Method could be possible for simpler designs



Data Transfer: Finalized Approach

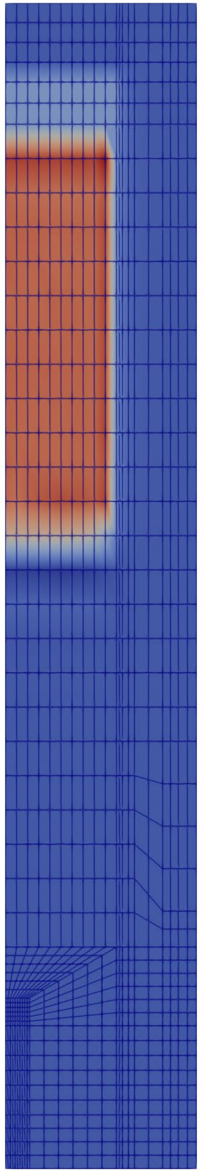
- Serpent provides explicit pebble positions, these can be easily mapped to a Griffin block and vice-versa
  - Material creation in Griffin is based on volume averaging based on the number of pebbles of each pass, fuel type, and each axial volume
- The reverse is performed to pass information in a block to a pebble

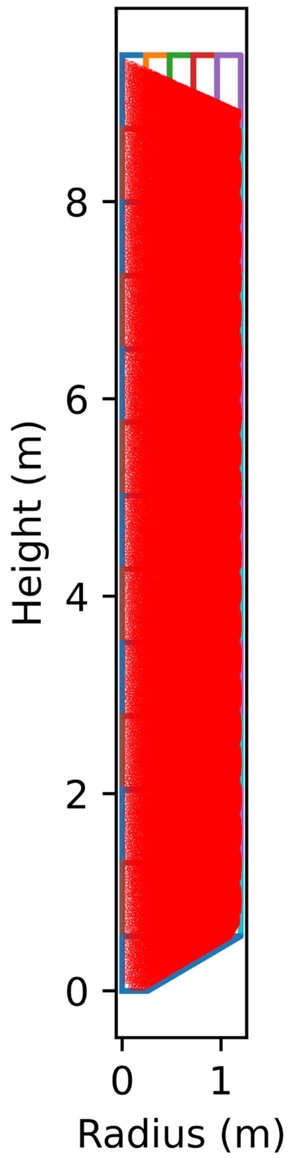




Results: Static Data Transfer

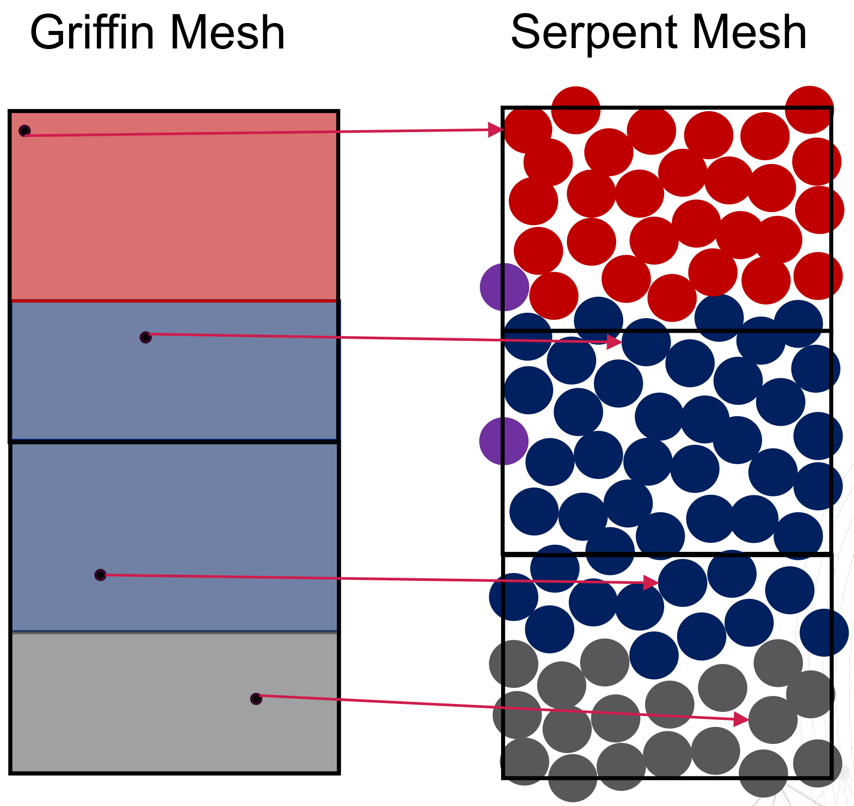
- Microscopic cross-sections have been generated from DRAGON
- Pebble isotopics can be passed from Serpent to Griffin
  - Mapping function performs the previous data transfer
- Griffin-only calculations have been performed
  - Provides confidence in the methodology
- Currently developing SPH factors
  - Help correct our cross-sections an maintain reaction rates





Results: Where are we going?

- Coupled Griffin/Pronghorn
  - Isotopic compositions
  - Generate SPH factors to conserve reaction rates
- Stand-alone Pronghorn
  - Power density
- Regardless of approach, we generate a temperature field
  - Each element will have an associated temperature
  - Pass average temperature to Griffin (50 K increments)



Conclusions & Wrap Up

- We have provided a framework for coupling a high-fidelity Monte Carlo code with a deterministic solver
  - Provides an opportunity to incorporate thermal fluid feedback into our Monte Carlo simulation
- Initial Serpent to Griffin coupling has been performed
  - Working on determining if SPH factors can be used
- Final stage will be to compare isothermal/linear temperature profile simulation with Multiphysics run-in simulation

Questions?