The Challenges of Developing and Maintaining Simulation Codes for Nuclear Astrophysics

- Our interests include
  - Modeling nucleosynthesis in X-ray bursts
  - Modeling different progenitor systems of Type Ia supernovae
  - Modeling convective phases in stellar evolution
  - These consume reaction rate data and nuclei properties
  - Can inform us about nuclear equation of state
- Simulation codes are complex
  - Many different algorithms incorporating different physical behaviors work together
  - Microphysical inputs (equation of state, reactions) require different backgrounds than PDE solvers
    - This is where we benefit a lot with collaborations outside of the simulator community, e.g. JINA reaclib
  - No longer can we expect a graduate student to implement a simulation code from scratch

The Challenges of Developing and Maintaining Simulation Codes for Nuclear Astrophysics

- Interdisciplinary development teams are a must
  - Funding mechanisms are needed to support this
- Machines are a moving target
  - How much time do we spend on optimizing for the newest machine vs. doing science?
  - Again, interdisciplinary funding mechanisms help a lot here
- What's the right balance between science and coding for students?
  - They're the next generation—they need to learn the software development techniques
    - Projects like *software carpentry* can help here
    - Sharing lecture/teaching material can help students self-learn
    - Summer schools are especially helpful
  - Coders are usually not given the same credit as scientists

The Challenges of Developing and Maintaining Simulation Codes for Nuclear Astrophysics

- Open codes benefit everyone
  - Leverage DOE's investiment to the community
  - Enables reproducibility
  - Allows for comparisons between groups, and allow for meaningful understanding of sensitivity to simulation parameters
- Sustainabilty?
  - What happens to codes when...
    - The lead developers move on?
    - Funding runs out?
  - WSSSP meetings are starting to address this