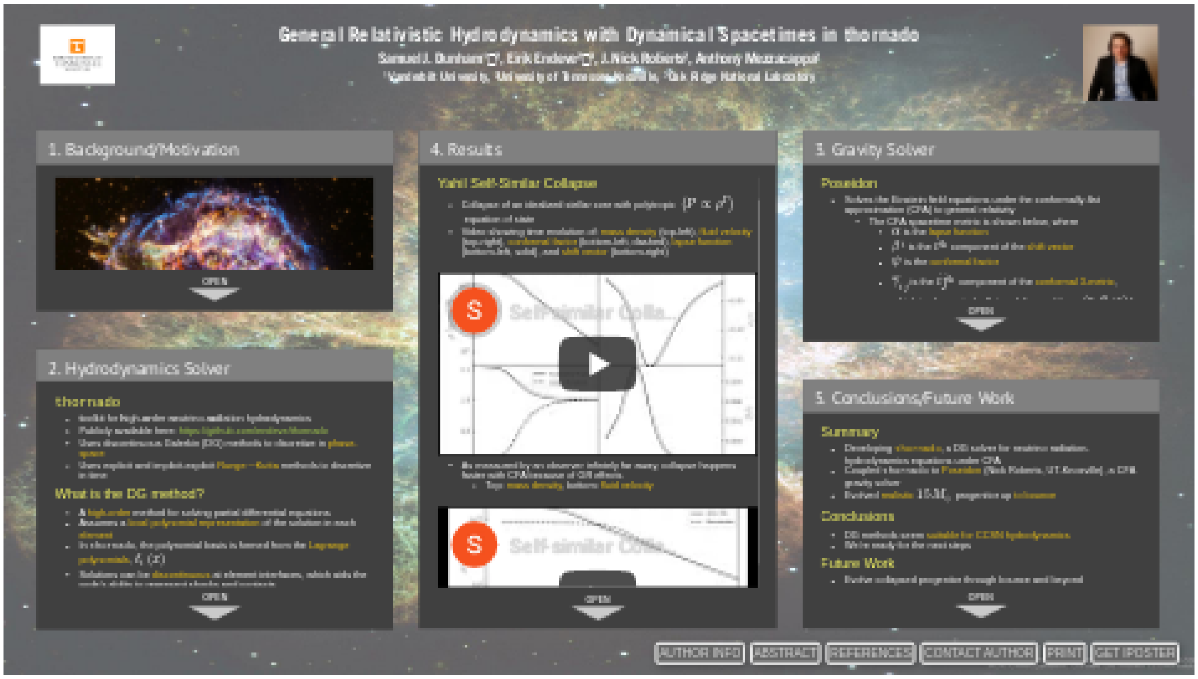
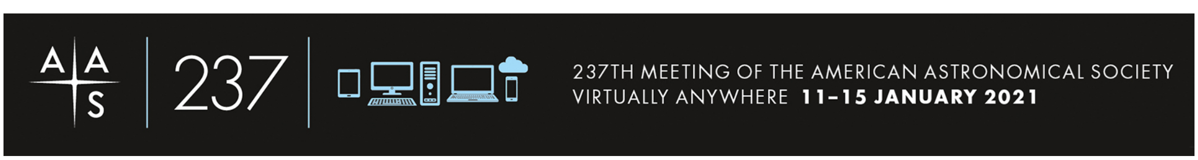
General Relativistic Hydrodynamics with Dynamical Spacetimes in thornado



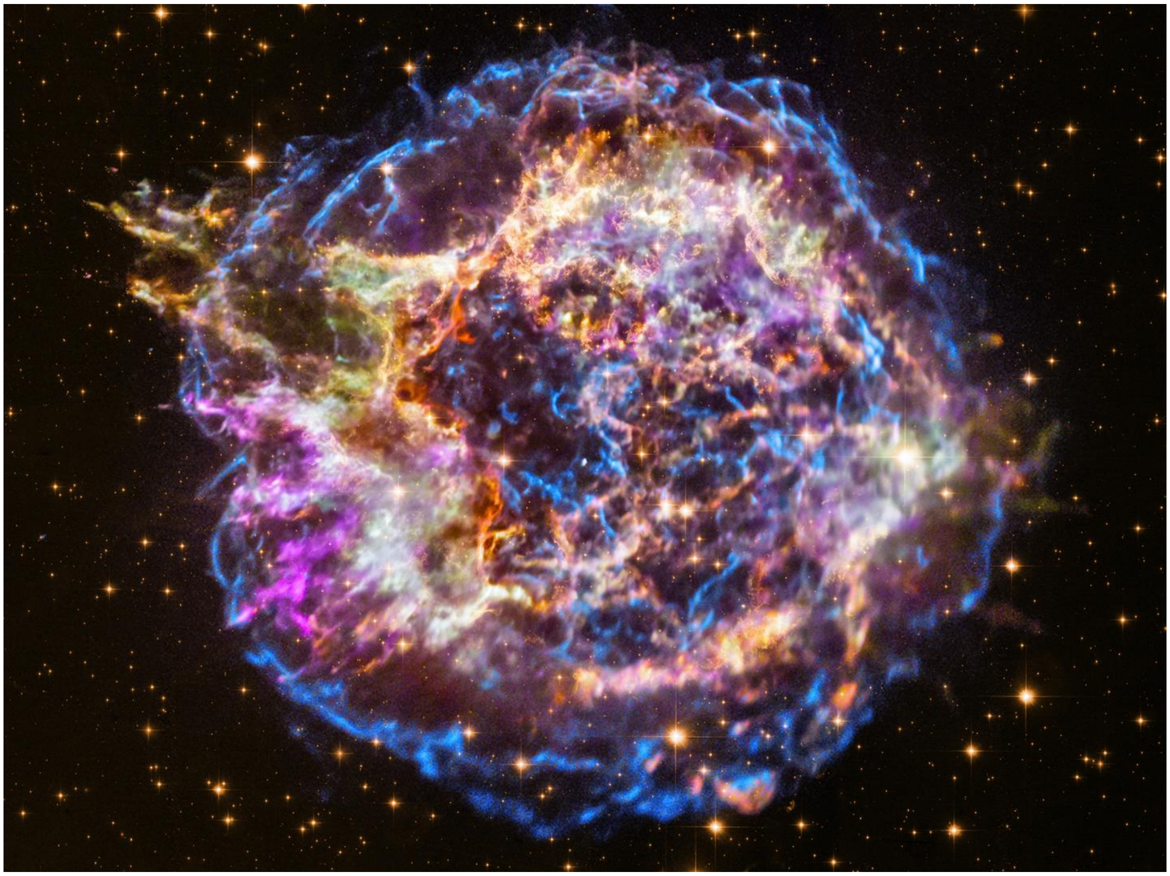
Samuel J. Dunham[1,2], Eirik Endeve[2,3], J. Nick Roberts[2], Anthony Mezzacappa[2]

[1]Vanderbilt University, [2]University of Tennessee-Knoxville, [3]Oak Ridge National Laboratory

PRESENTED AT:



1. MOTIVATION



What are the science goals?

- To understand how massive $(M \gtrsim 8\,M_{\odot})$ stars explode as core-collapse supernovae (CCSNe), and the observables they produce

Why are these goals important?

- CCSNe synthesize and distribute elements from oxygen to iron, influencing galactic chemical evolution
- CCSNe are expected to produce gravitational waves in frequency ranges of, e.g., aLIGO/Virgo/KAGRA
- CCSNe are environments containing trapped neutrinos of all flavors: useful for probing fundamental neutrino physics
- CCSNe may contribute to r-process elements
- CCSNe provide information about neutron stars, including
  - Mass
  - Spin
  - Natal kicks
  - Magnetic fields

How are we trying to achieve our goals?

- Developing a code to simulate the neutrino-radiation hydrodynamics of CCSN explosions on leadership-class supercomputers

What makes our effort different?

- We use the discontinuous Galerkin (DG) method for spatial discretization (see box 2), whereas most other codes use the finite volume method

- We approximate general relativity with the conformally-flat approximation (see box 3)

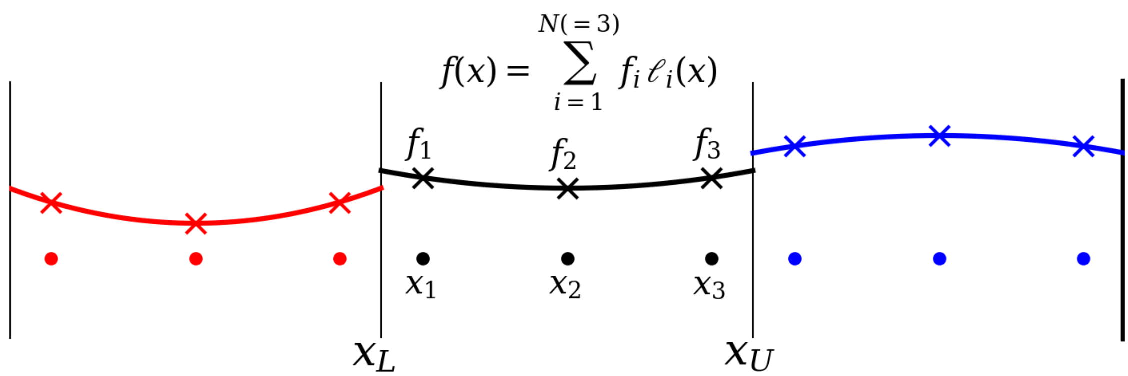
2. HYDRODYNAMICS SOLVER

thornado

- **t**oolkit for **h**igh-**or**der **n**eutrino-r**ad**iation hydr**o**dynamics
- Publicly available here: https://github.com/endeve/thornado (http://github.com/endeve/thornado)
- Uses discontinuous Galerkin (DG) methods to discretize in phase-space
- Uses explicit and implicit-explicit Runge—Kutta methods to discretize in time

What is the DG method?

- A high-order numerical method for solving partial differential equations
- Assumes a local polynomial representation of the solution in each element
- Solutions can be discontinuous at element interfaces, which aids the code's ability to represent shocks and contacts
- In `thornado`, the polynomial basis is formed from the Lagrange polynomials, $\ell_i\left(x\right)$
- A sequence of elements for a 1D field (e.g., mass density) is shown in the image below for a 3rd order method
  - The solid dots are the local grid points (nodes)
  - The x symbols mark the nodal values of the solution
  - The solution in each element is a linear combination of the nodal values and the Lagrange polynomials



Why use the DG method?

- Able to resolve strong shocks and achieve high-order accuracy in regions of smooth flow
- High-order methods achieve desired accuracy with fewer degrees of freedom (number of elements times number of nodes per element)
- Uses a compact stencil (nearest-neighbors only), irrespective of the order used
- Achieves good scalability on parallel computing architectures

4. RESULTS

Yahil Self-Similar Collapse

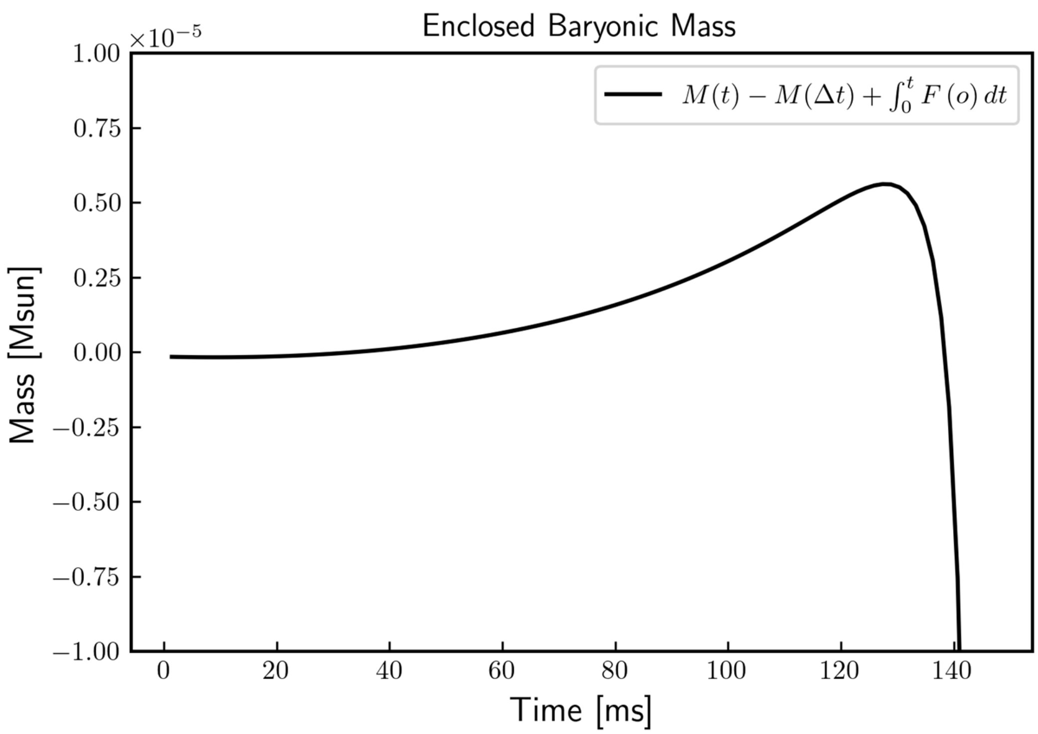
- Collapse of an idealized stellar core with polytropic $\left(P \propto \rho^{\Gamma}\right)$ equation of state
- Video showing time evolution of: mass density (top-left), fluid velocity (top-right), conformal factor (bottom-left, dashed), lapse function (bottom-left, solid), and shift vector (bottom-right)
- c is the speed of light

[VIDEO] https://www.youtube.com/embed/IseEWgjxVls?rel=0&fs=1&modestbranding=1&rel=0&showinfo=0

- As measured by an observer infintely far away, collapse happens faster with CFA because of GR effects
  - Top: mass density, bottom: fluid velocity

[VIDEO] https://www.youtube.com/embed/k3qm4oji3BY?rel=0&fs=1&modestbranding=1&rel=0&showinfo=0

- Accounting for mass flowing in from the boundary, this problem should conserve baryonic mass
  - We find agreement to better than $10^{-5}\,M_{\odot}$
  - Not machine precision because not solving mass equation in full conservation form
  - The drop at the end is due to an inability to resolve all of the mass being localized to one point



TOV star

- We were not able to complete this analysis before the poster session
- Main issue is related to boundary conditions for the gravity solver

Gravitational Collapse with Nuclear EOS

- Collapse of the inner 8000 km of a $15\,M_{\odot}$ progenitor (Woosley & Heger (2007))
- Using tabulated, nuclear equation of state (SFHo; Steiner et al., (2013))
  - As implemented in weaklib (publicly available here: https://github.com/starkiller-astro/weaklib (https://github.com/starkiller-astro/weaklib))
- Multi-panel movie showing time evolution of: mass density (top-left), fluid velocity (top-right), conformal factor (bottom-left, dashed), lapse function (bottom-left, solid), and shift vector (bottom-right)

[VIDEO] https://www.youtube.com/embed/PS_leI6sABc?rel=0&fs=1&modestbranding=1&rel=0&showinfo=0

- Again, when compared with Newtonian result, collapse happens faster because of GR effects according to an observer infintely far away
  - Top: pressure, middle: temperature, bottom: electron fraction

[VIDEO] https://www.youtube.com/embed/1H24nFnWOTg?rel=0&fs=1&modestbranding=1&rel=0&showinfo=0

3. GRAVITY SOLVER

Poseidon

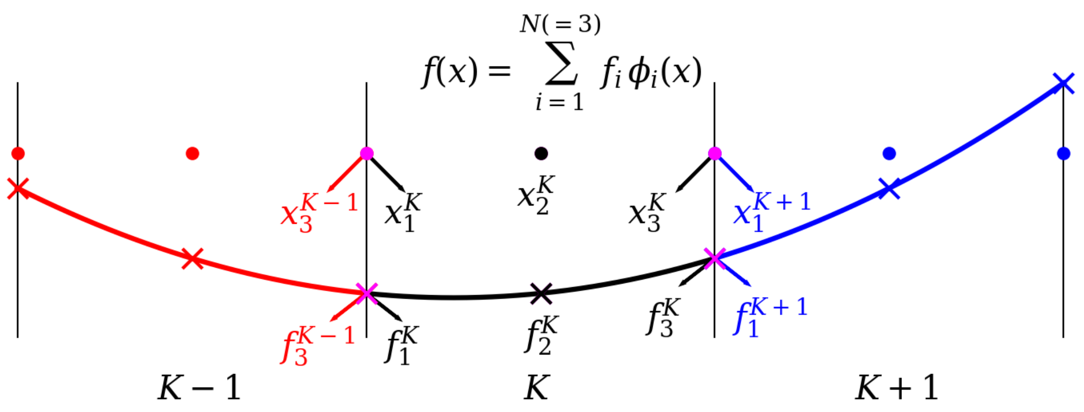
- Solves the Einstein field equations under the conformally-flat approximation (CFA) to general relativity
  - The CFA spacetime metric is shown below, where
    - α is the lapse function
    - β^i is the i^{th} component of the shift vector
    - ψ is the conformal factor
    - $\overline{\gamma}_{ij}$ is the ij^{th} component of the conformal 3-metric, which is chosen to be flat and diagonal (e.g., $\left(1, r^2, r^2 \sin^2 \theta\right)$ for spherical coordinates)

$$[g_{\mu\nu}] = \begin{pmatrix} -\alpha^2 + \beta_i \beta^i & \beta_1 & \beta_2 & \beta_3 \\ \beta_1 & \psi^4\, \overline{\gamma}_{11} & 0 & 0 \\ \beta_2 & 0 & \psi^4\, \overline{\gamma}_{22} & 0 \\ \beta_3 & 0 & 0 & \psi^4\, \overline{\gamma}_{33} \end{pmatrix}$$

- Poseidon uses maximal slicing
  - Avoids code crashing due to singularity
- Given the fluid fields at time t, Poseidon solves the field equations for α, β^i, and ψ
- CFA has been shown to be applicable to CCSN simulations (Dimmelmeier et al. (2002))
- Uses continuous Galerkin (CG) method

What is the CG Method?

- Similar to DG method (see box 2), except
  - CG demands continuity of solution across element interfaces
- A sequence of elements for a 1D field (e.g., lapse function) is shown in the image below using a 3rd order method
  - The solid dots are the grid points (nodes)
  - The x symbols mark the nodal values of the field
  - The solution is a linear combination of the nodal values and the basis functions (e.g., Lagrange polynomials)
  - Note that adjacent elements share a node



5. CONCLUSIONS/FUTURE WORK

Summary

- Developing `thornado`, a DG solver for neutrino radiation-hydrodynamics equations under CFA
- Coupled `thornado` to Poseidon, a CFA gravity solver being developed by Nick Roberts of UT-Knoxville
- Evolved realistic $15\,M_{\odot}$ progenitor up to bounce

Conclusions

- DG methods seem suitable for CCSN hydrodynamics
- We're ready for the next steps

Future Work

- Evolve collapsed progenitor through bounce and beyond
- Multi-dimensional simulations
- Couple to our DG, CFA neutrino transport solver being developed by Zack Elledge of UT-Knoxville
- CCSN simulations!

AUTHOR INFO

Samuel J. Dunham[1,2]

Eirik Endeve[2,3]

J. Nick Roberts[2]

Anthony Mezzacappa[2]

[1]Department of Physics and Astronomy, Vanderbilt University, 6301 Stevenson Center, Nashville, TN 37235

[2]Department of Physics, University of Tennessee at Knoxville, 1408 Circle Dr, Knoxville, TN 37996

[3]Computer Science and Mathematics Division, Oak Ridge National Laboratory, 1 Bethel Valley Rd, Oak Ridge, TN 37830

ABSTRACT

We present recent results from thornado, a radiation-hydrodynamics code we are developing primarily for application to core-collapse supernova (CCSN) simulations, using high-order-accurate Runge–Kutta discontinuous Galerkin methods [1]. We have validated our code against test problems in both special and general relativistic (GR) regimes with curvilinear coordinates using stationary spacetimes [2]. Here, we present results from test problems involving dynamical spacetimes, which are achievable after coupling thornado to Poseidon [3], the latter of which solves the 3+1 Einstein equations assuming the conformally-flat approximation (CFA) [4] to GR. Specifically, we evolve the self-similar collapse of a spherically symmetric star with a polytropic equation of state (EoS) [5]. We compare these results with those obtained using the non-relativistic hydrodynamics and gravity solvers of thornado and Poseidon, respectively [6]. Preliminary results show that the two solutions agree until central densities reach about $10^{11}\,\mathrm{g\,cm^{-3}}$ at which point the relativistic solution begins to collapse at a faster rate. The results from this problem represent a significant step toward realistic CCSN simulations in conformally-flat GR with thornado. We also show results relating to the stability of a TOV star assuming a stationary spacetime, as well as results when that assumption is relaxed, and compare with published results in [7]. These encouraging results demonstrate that thornado has been successfully extended to GR and coupled to Poseidon. In this poster, we describe thornado and our choice of numerical methods, show our latest results, and discuss future work, including incorporation of a tabulated nuclear EoS.

[1] Cockburn, B. et al. 2001 J. Sci. Comp. 16:173

[2] Dunham, S. J. et al. 2020 J. Phys.: Conf. Ser. 1623 012012

[3] Roberts, J. et al. (in prep.)

[4] Wilson, J. R. et al. 1996 Phys. Rev. D, 54, 1317

[5] Yahil, A. 1983 Ap. J., 265:1047

[6] Endeve, E. et al. 2019 J. Phys.: Conf. Ser. 1225 012014

[7] Cheong, P. C.-K. et al. 2020 Classical and Quantum Gravity, 37:14

REFERENCES

Background image: https://apod.nasa.gov/apod/ap200906.html (https://apod.nasa.gov/apod/ap200906.html)

Picture in Box 1: https://apod.nasa.gov/apod/ap190906.html (https://apod.nasa.gov/apod/ap190906.html)

Steiner et al., 2013 ApJ **774** 17

Dimmelmeier et al., 2002 A&A, **388** 917

Pochik et al., 2021 ApJS (accepted) arXiv:2011.04680

Woosley, S. E., & Heger, A. 2007, PhR, 442, 269