Octree Frontend for Enzo-E in YT

Bolun Thompson
REU FoDoMMaT Visiting Scholar
Matthew Turk
Assistant Professor at the School of Information Sciences

July 6th, 2023

Volumetric Simulations

- Formation of stars and galaxies
- Nuclear reactor mechanics

Why?

Connects theories with data

What

- Enzo: **Adaptive Mesh Refinement** (AMR) astrophysical simulation code [1]
  - Multi-purpose
- Adaptive Mesh Refinement
  - Increase simulation accuracy in turbulent or interesting parts
  - Inconsistent

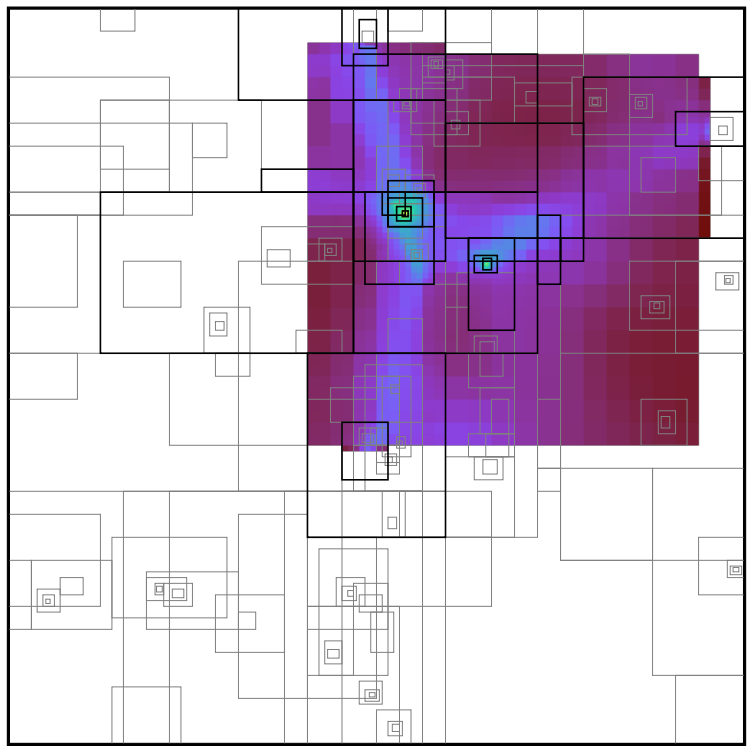

Figure 1: Patch-based AMR [2]

Enzo-E

- Enzo-E: Rearchitecture of Enzo for exascale computing [3]
  - Datasets $> 2048^3$ blocks (> 1 TB)
- Uses **array-of-octree** AMR
- Octree
  - Tree data structure to represent 3D space
  - Recursively subdivided
- Array-of-octree
  - 3D Array of Octrees
  - Easily parallelizable

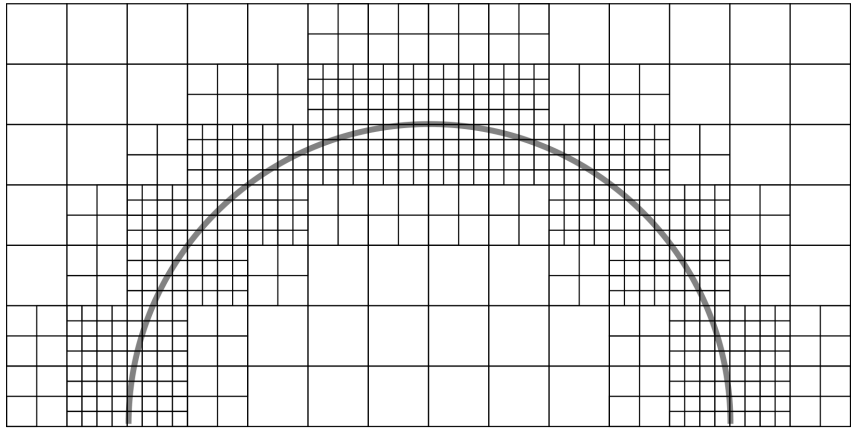

Figure 2: AMR Quadtree [4]

yt

- yt: Python analysis and visualization package [2]
- For any type of volumetric data
  - Both Enzo and Enzo-E

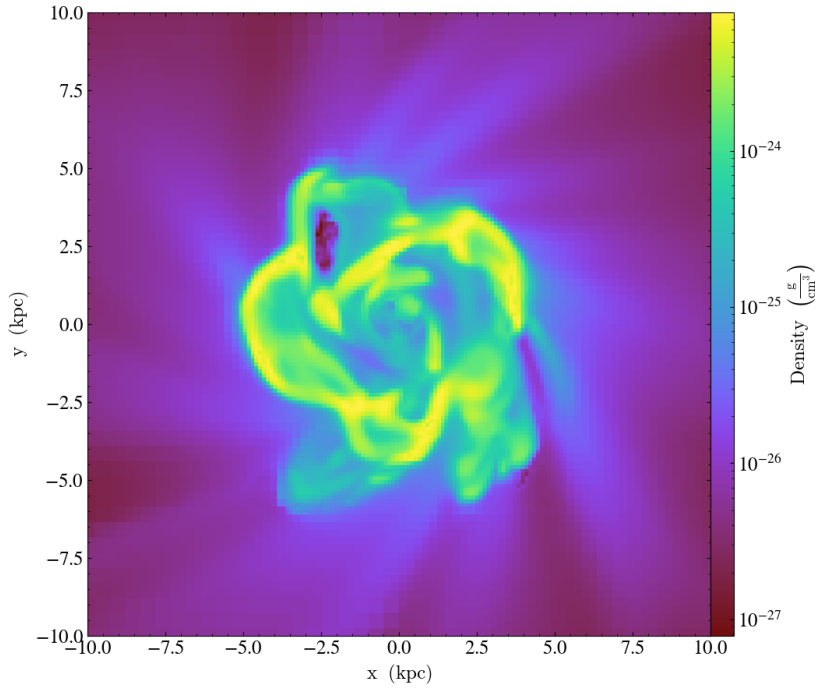

Figure 3: Slice over z axis of density

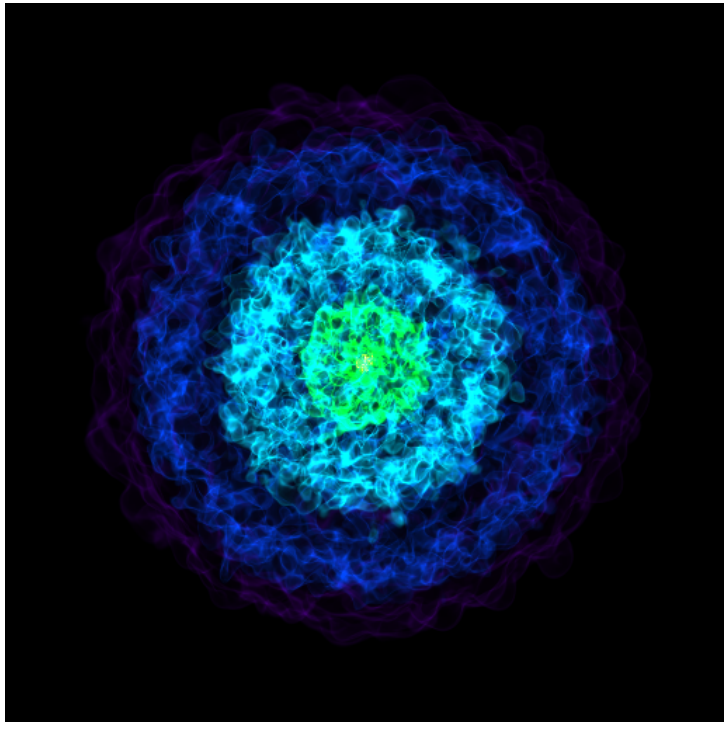

Figure 4: 3D visualization of density

Frontends

- yt supports a variety of **frontends** to load data
- Frontends can be implement in various formats
  - Grid-based
    - Inconsistent across the domain
  - Array-of-octree

Problem

Enzo-E analysis in yt is slow!

Problem

- Slow on large datasets
- Can't practically analyze enzo-e datasets of over 256^3 blocks
  - 256^3 blocks $\rightarrow$ multiple hours to load in the data
- Needs to analyze datasets of size 2048^3 blocks
  - ≈ 1 TB

Current Frontend

- Collection of grids
  - Each grid is a python object
    - ≈ 1 KB
- Largely single threaded

New frontend

- Array of Octree
- Multithreaded
- Each Oct is a C struct
  - At most 88 bytes

Result

- Faster
- Can analyze datasets of size 2048^3 blocks
  - ≈ 1 TB

Current Status

- New frontend is partially built
- Non refined data can be loaded in

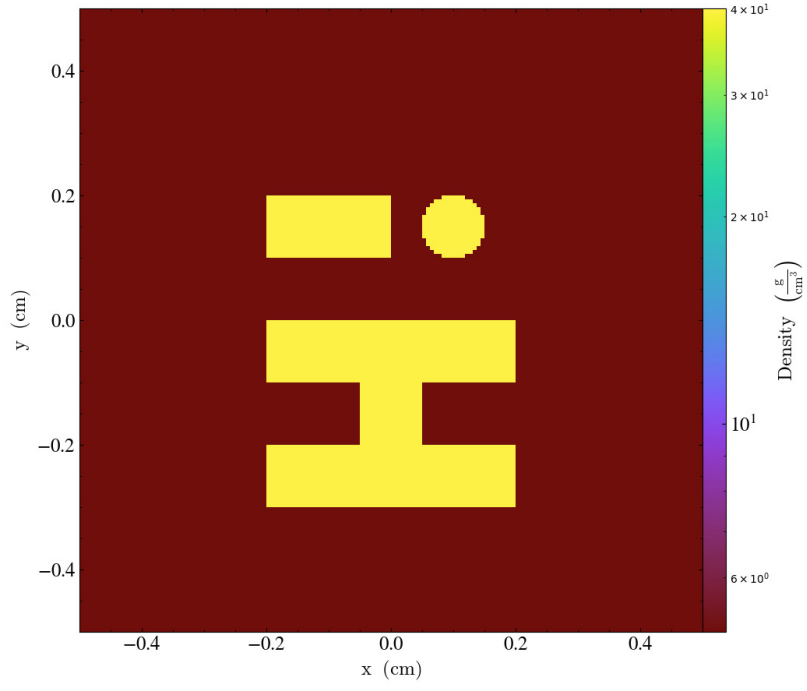

Figure 5: Incorrect visualization. The “Hi” is mirrored and rotated from what it should be.

Future Work

- New frontend is unoptimized
  - Initially slower
- Other simulation codes interpreted as grid-based, instead of oct-based

References I

1. **ENZO: AN ADAPTIVE MESH REFINEMENT CODE FOR ASTROPHYSICS** Greg L Bryan, Michael L Norman, Brian W OShea, Tom Abel, John H Wise, Matthew J Turk, Daniel R Reynolds, David C Collins, Peng Wang, Samuel W Skillman, ... Yuan Li and
The Astrophysical Journal Supplement Series (2014-03) https://doi.org/10.1088%2F0067-0049%2F211%2F2%2F19 DOI: 10.1088/0067-0049/211/2/19

2. **yt: A Multi-code Analysis Toolkit for Astrophysical Simulation Data** Matthew J Turk, Britton D Smith, Jeffrey S Oishi, Stephen Skory, Samuel W Skillman, Tom Abel, Michael L Norman
(2011-01) https://arxiv.org/abs/1011.3514 DOI: 10.1088/0067-0049/192/1/9

References II

3. **Computational cosmology and astrophysics on adaptive meshes using charm++** James Bordner, Michael L Norman (2018) https://arxiv.org/abs/1810.01319

4. **Adaptive mesh refinement in the fast lane** D Dunning, W Marts, RW Robey, P Bridges
Journal of Computational Physics (2020) https://www.sciencedirect.com/science/article/pii/S0021999119308988 DOI: https://doi.org/10.1016/j.jcp.2019.109193