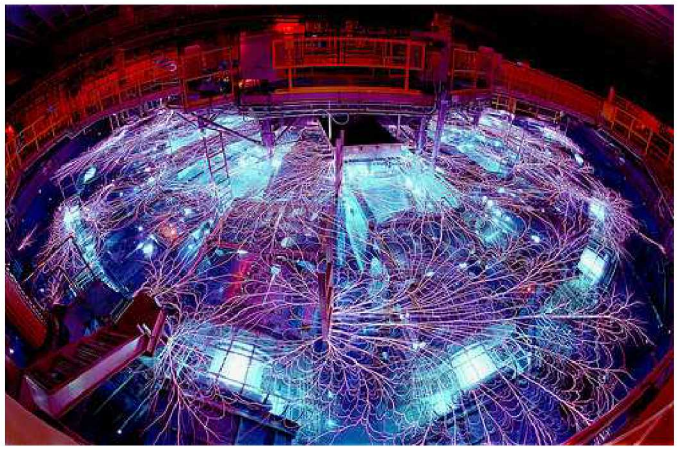
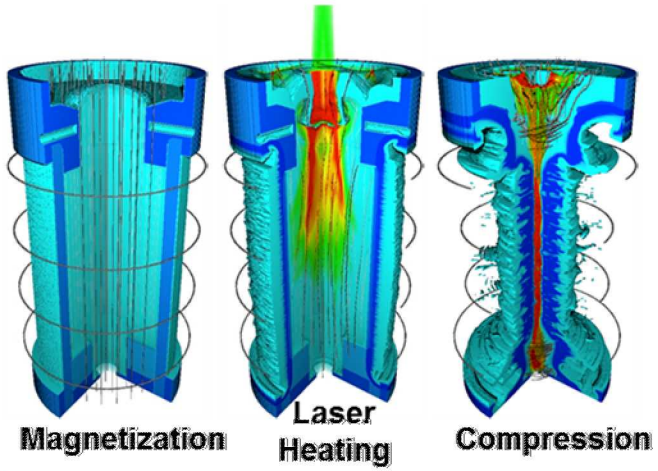
SAND2020-7901C





FLEXO Development Activities

Presenter: Tom Voth, SNL/NM Org. 1443 (Computational Multiphysics)

FLEXO Team: K. Beckwith, S. Bond, B. Granzow, N. Hamlin, H. Hanshaw, G. Hansen, C. Jennings, E. Love, M. Martin, L. Shulenburger, A. Stagg

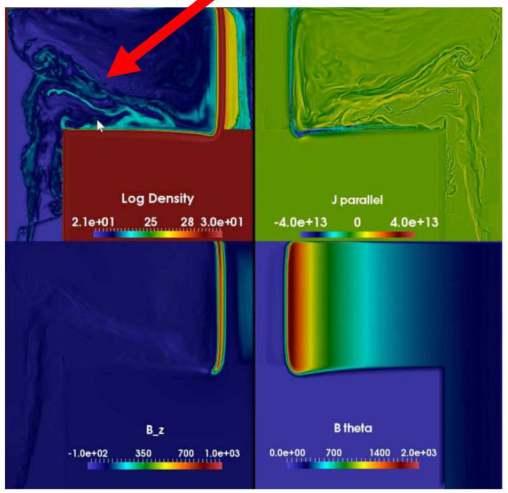
Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

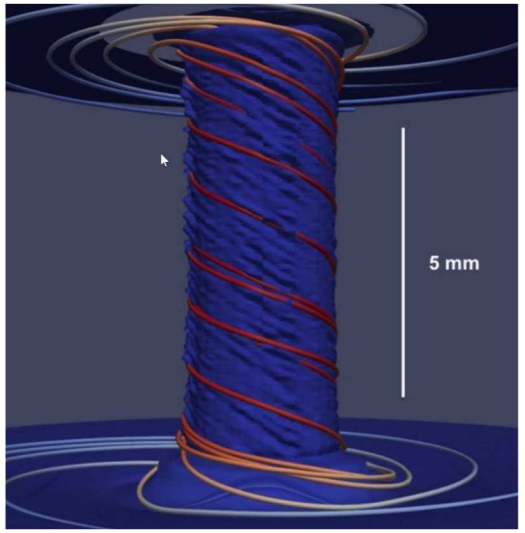
PERSEUS/FLEXO Overview:

- **PERSEUS:** A generalized Ohm's Law (XMHD), FORTRAN90, Discontinuous Galerkin (DG) code. Originally developed at Cornell (Martin, Seyler) and licensed to SNL. Numerous publications demonstrating the need for XMHD physics in modeling of pulsed power systems.

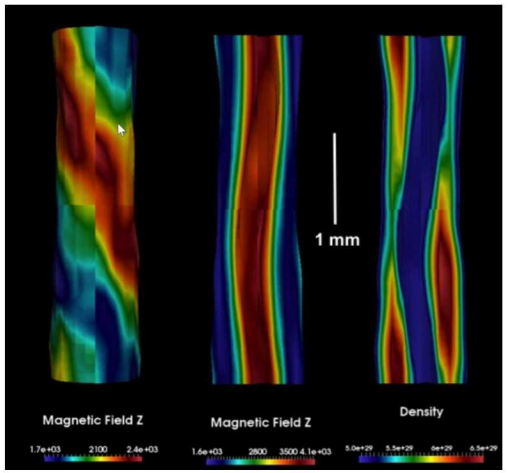
- **FLEXO:** A new C++ XMHD code (Flux Limited Extended Ohm's law) based on PERSEUS. Developed at SNL with new capabilities: multi-material equation of state (EOS), adaptive mesh refinement (AMR). All code developed from birth to support MPI-X (GPU/threaded) advanced computer architectures.

1) Feed plasma transport requires XMHD due to low densities





2) XMHD predicts helical instability in 3D calculations due to feed plasma driving flux compression in MagLIF

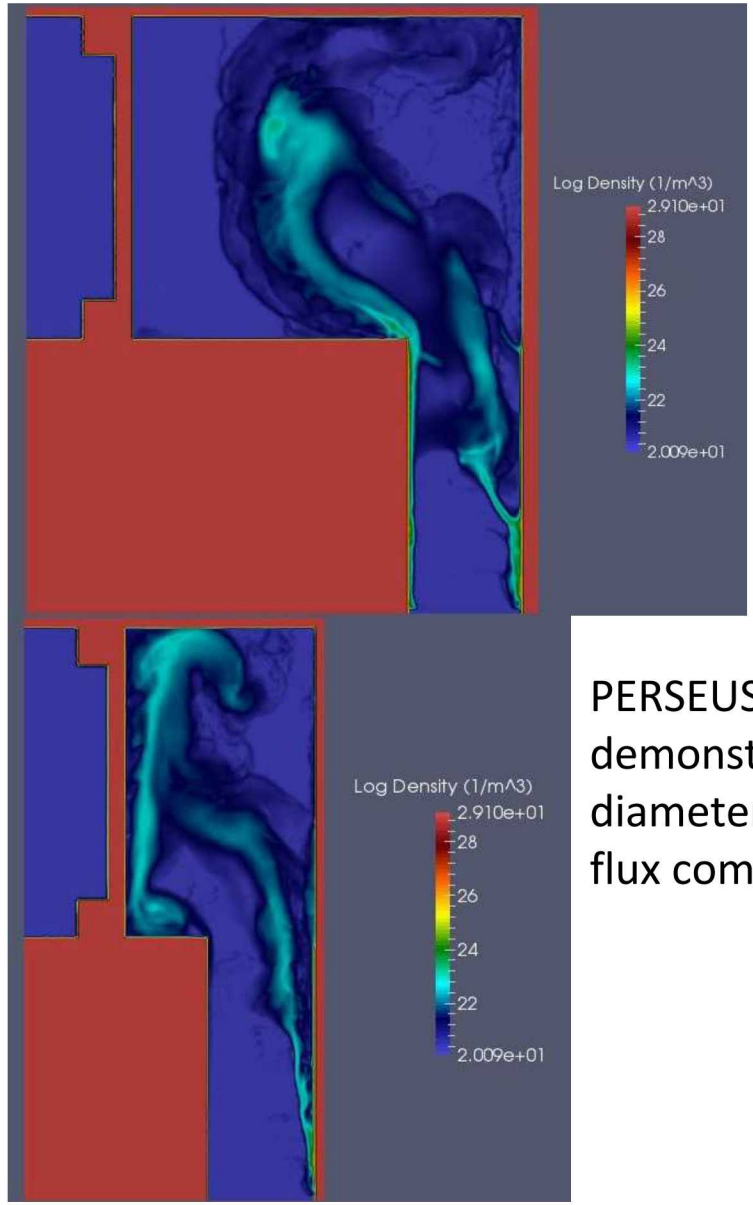


3) Low density feed plasma (~10^18/cc) changes morphology and stability of liner stagnation

PERSEUS has demonstrated its predictive abilities for Z-target physics.

- PERSEUS is the design code for experiments to investigate the origins of the helical instability in MagLIF.

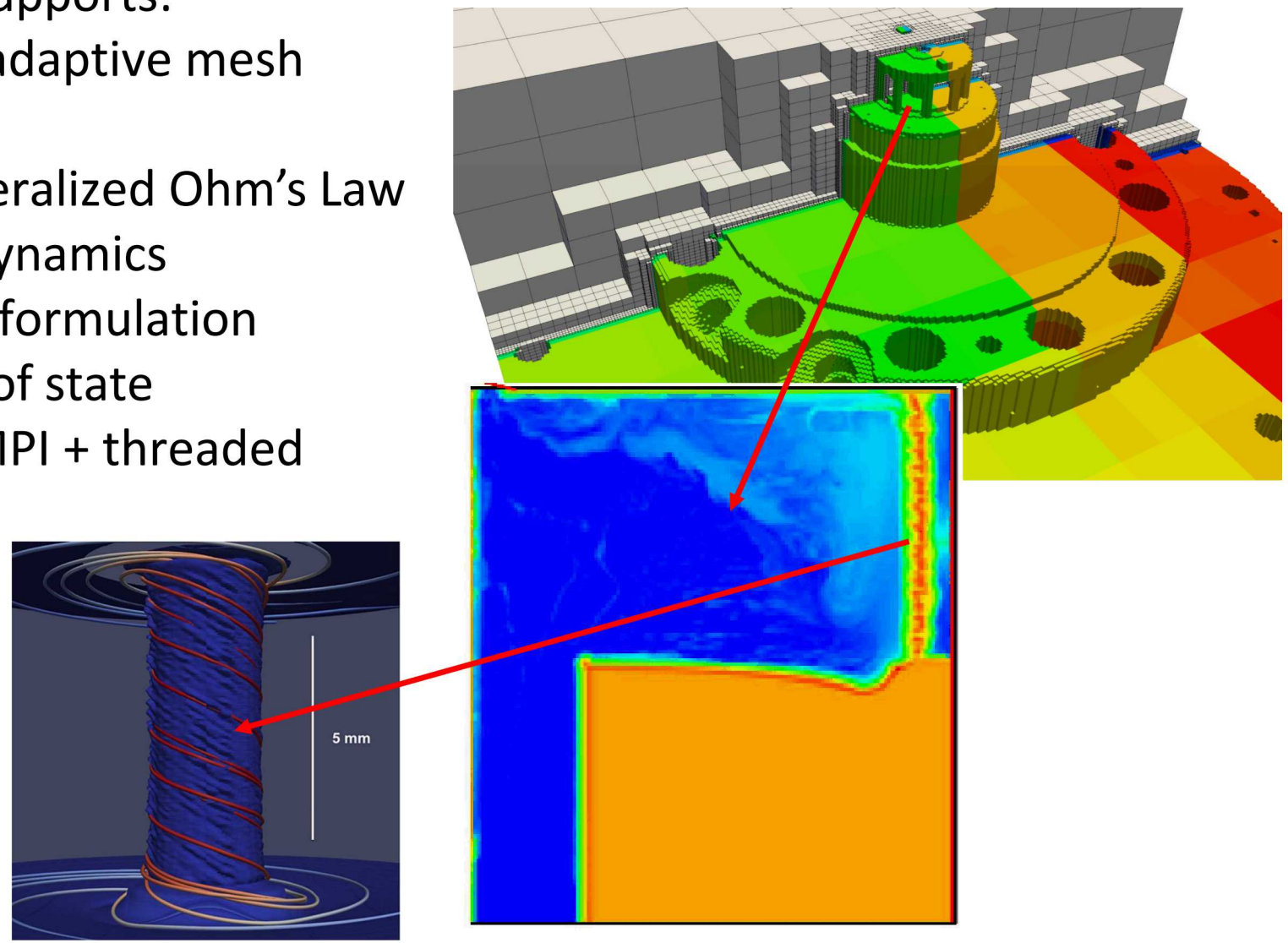
- PERSEUS validation tests on Z experimental results are bringing new insights:
  - MagLIF simulations show steeper helical pitch on liner surface for larger flux compression volume.
  - Results are closer to qualitative agreement with experiment when modeling influence of Hall physics on B-field diffusion into low-density plasma.



PERSEUS demonstrates can diameter effects on flux compression.

FLEXO is being developed to provide increased fidelity:

- Develop a framework that supports:
  - Structured block-based adaptive mesh refinement (AMR)
  - for the execution of generalized Ohm's Law (G.O.L.) magnetohydrodynamics
  - with a multiple material formulation
  - using tabular equations of state
  - that can execute using MPI + threaded parallelism



PERSEUS/FLEXO Algorithm:

Governing Equations:

$$\partial_t[\rho] + \nabla \cdot [\rho \mathbf{v}] = 0$$

$$\partial_t[\rho \mathbf{v}] + \nabla \cdot [\rho \mathbf{v} \otimes \mathbf{v} + p\mathbf{I}] = \mathbf{J} \times \mathbf{B}$$

$$\partial_t[E] + \nabla \cdot [\mathbf{v}(E + p)] = \mathbf{v} \cdot (\mathbf{J} \times \mathbf{B}) + \eta \mathbf{J}^2$$

$$\partial_t[\mathbf{B}] + \nabla \times \mathbf{E} = \mathbf{0}$$

$$\partial_t[\mathbf{E}] - c^2 \nabla \times \mathbf{B} = -c^2 \mu_0 \mathbf{J}$$

$$\partial_t[\mathbf{J}] = (n_e e^2/m_e) \left[\mathbf{E} + \mathbf{v} \times \mathbf{B} - (n_e e)^{-1} \mathbf{J} \times \mathbf{B} - \eta \mathbf{J}\right]$$

Time advancement strategy:

Explicit predictor:

$$\rho^* = \rho^n - \Delta t \, \nabla \cdot [\rho \mathbf{v}]$$

$$[\rho \mathbf{v}]^* = [\rho \mathbf{v}]^n - \Delta t \, \nabla \cdot [\rho \mathbf{v} \otimes \mathbf{v} + p\mathbf{I}]^n + \Delta t \, [\mathbf{J} \times \mathbf{B}]^n$$

$$\mathcal{E}^* = \mathcal{E}^n - \Delta t \, \nabla \cdot [\mathbf{v}(\mathcal{E} + p)]^n + \Delta t \, [\mathbf{v} \cdot (\mathbf{J} \times \mathbf{B}) + \eta \mathbf{J}^2]^n$$

$$\mathbf{B}^* = \mathbf{B}^n - \Delta t \, [\nabla \times \mathbf{E}]^n$$

$$\mathbf{E}^* = \mathbf{E}^n + \Delta t \, [c^2 \nabla \times \mathbf{B}]^n$$

$$\mathbf{J}^* = \mathbf{J}^n$$

Implicit (element-local) corrector:

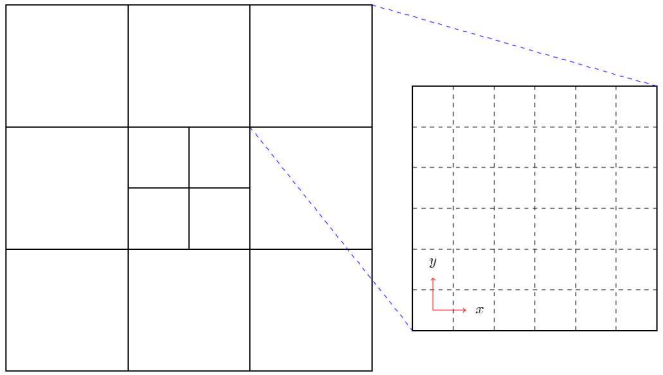
$$\mathbf{E}^{n+1} = \mathbf{E}^* - \Delta t \, c^2 \mu_0 \mathbf{J}^{n+1}$$

$$\mathbf{J}^{n+1} = \mathbf{J}^* + \Delta t \left(\frac{n_e e^2}{m_e}\right) [\quad \mathbf{E}^{n+1} + \mathbf{v}^* \times \mathbf{B}^* - \frac{1}{n_e e} \mathbf{J}^{n+1} \times \mathbf{B}^* - \eta^* \mathbf{J}^{n+1} \,]$$

For details see: Seyler, Martin, Physics of Plasmas **18**, 012703 (2011)

FLEXO Framework: Structured block AMR

- Based on an MPI-parallelized octree
- Each leaf in the octree is an (i,j,k) structured grid -> “block”
- Physics is evaluated over each block
- Fluxes across block boundaries couple the block physics
- Shared memory parallelism -> kernels over block entities (cells, faces, …)
- MPI parallelism -> solution information transferred at block boundaries
- Block adjacencies restricted to at most one finer or coarser level (2-1 interfaces)

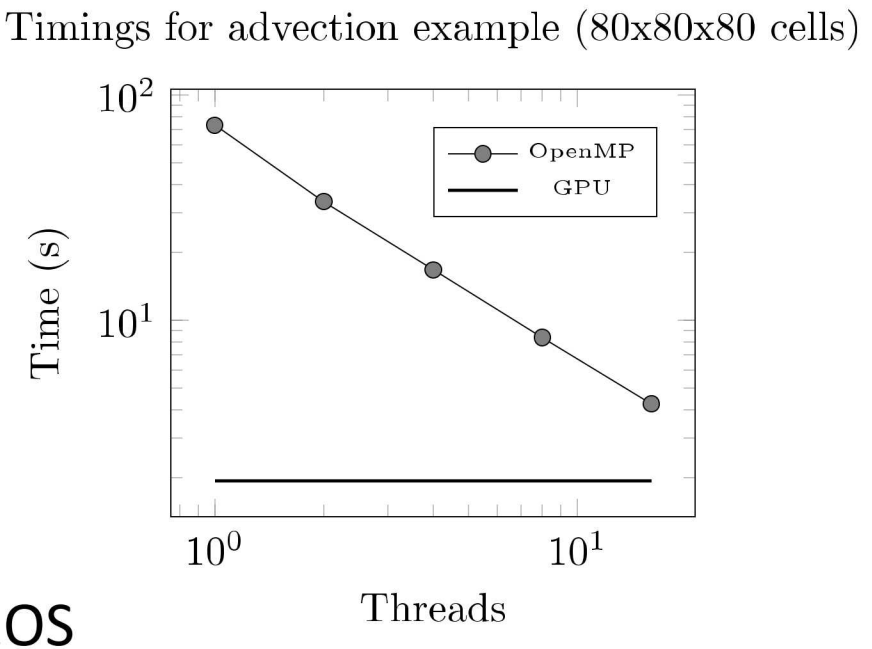


FLEXO Framework: Threaded Parallelism

- Thread-based parallelism occurs over block entities (cells, faces, …)
  - A physics functor is written over entities
  - A parallel for each construct (“hpc::grid_for_each”) is called for the functor
  - Mixing + matching execution spaces (OpenMP, CUDA,…) a possibility, not yet exercised

```
auto f = [=] HPC_DEVICE (hpc::vector3<int> const& cell_loc) {
  int const cell = hpc::get_index(cell_loc, cell_grid);
  for (int basis = 0; basis < nbasis(polynomial_order); ++basis) {
    resid(cell, RHO, basis) += 1.0;
  }
};
hpc::grid_for_each(hpc::execution::device, cell_grid, f);
```

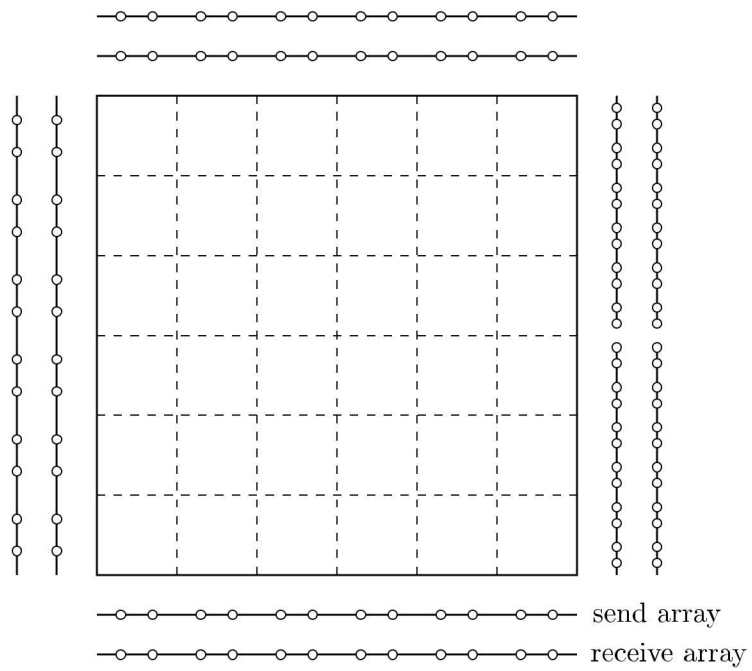
- Allow tabular EOS on GPUs
  - Leverage NG capable library “mu”
  - Ability to import SESAME tables into C++ data structures
  - Performs bisection to find appropriate cell in table
  - Performs bilinear interpolation in table cell to evaluate EOS



FLEXO Framework: MPI Parallelism

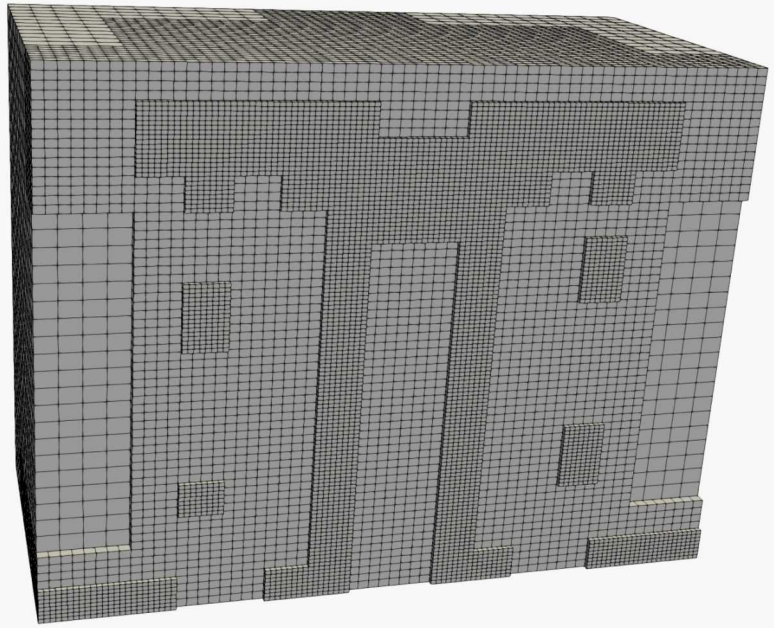
- Octree stored on every MPI rank (Athena approach)
  - Data only initialized for owned blocks
  - Ownership determined by a Z-curve ordering of leaf nodes in octree
  - Z-curve also to be used for dynamic load balancing (in progress)

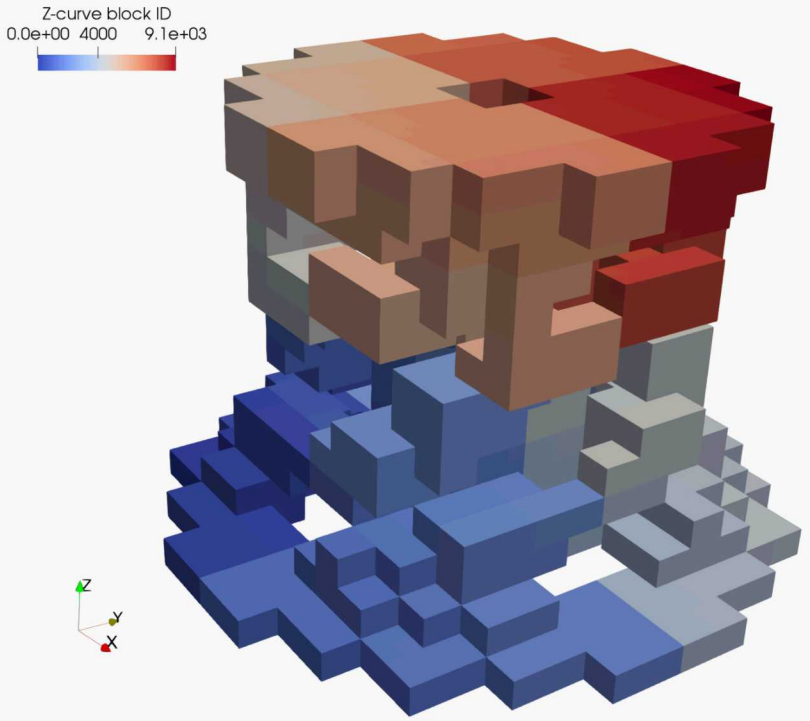
- MPI parallelism achieved via transfer of block border data:
  - The DG solution interpolated to QPs on block border
  - Borders classified as:
    - “standard” -> adjacent block at same depth
    - “coarse to fine” -> adj. block at depth + 1
    - “fine to coarse” -> adj. block at depth - 1
    - “boundary” -> no adjacent block
  - Fluxes computed exactly at AMR interfaces
    - Eliminates need for prolongation / restriction of solution

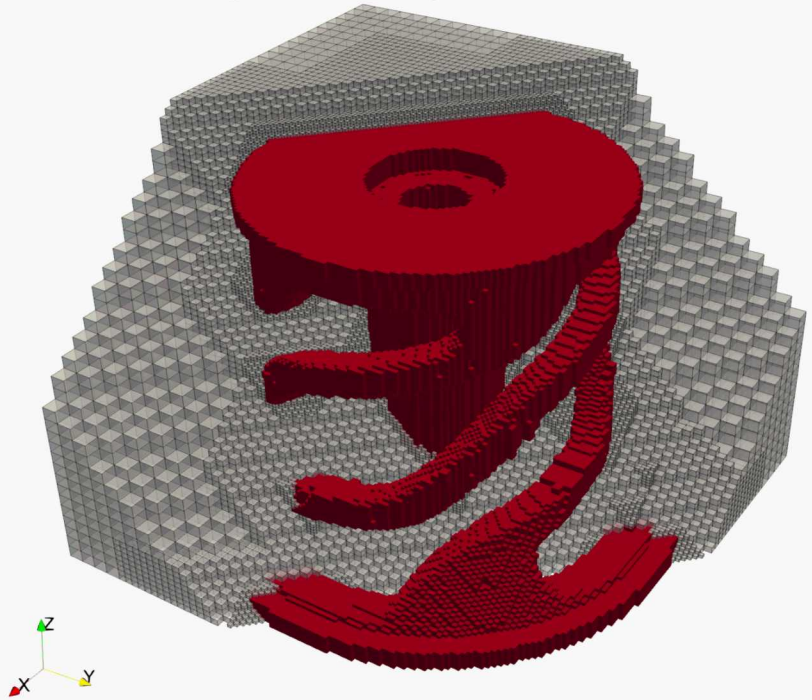


FLEXO Framework: Refinement to STL geometry

- Simple example case:
  - STL screw-pinch geometry with ~60,000 triangles
  - Start with 2x2x2 octree mesh
  - Refine a block if any cell center in the block is `interior' to STL geometry
  - 'Interior'-ness determined by a ray-shooting point in polygon type algorithm
  - Repeat 5x
  - Initialize density based on cell center being `interior' or `exterior' to STL geometry







Examining stabilization approaches:

- Discontinuous Galerkin approaches:
  - Slope limiters:
    - Can be applied (synchronously or **asynchronously**) directly to conserved variables or **via primitive variables**
      - Barth-Jesperson with minmod (more robust) and MC (less diffusive)
      - Vertex (Kuzmin)
      - Moment
  - Artificial Viscosity Methods (P > 0)
  - Positivity limiter

D. Kuzmin / Journal of Computational and Applied Mathematics 233 (2010) 3077–3085

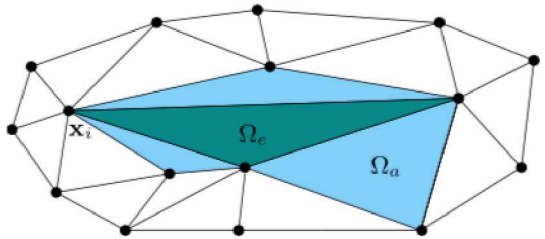


Fig. 1. Vertices and neighbors of Ω_e on a triangular mesh.

- High-resolution Finite Volume:
  - Minmod, MC, Superbee, Anti-diffusion (Harten et al.) etc.

Moment limiter: L. Krivodonova,"Limiters for high-order discontinuous Galerkin methods", JCP 226 (2007) 879-896.

Compatible gradient limiting

$$d\mathbf{C} = \mathbf{J} d\mathbf{P}$$

Limited conserved-variable slopes

Limited primitive-variable slopes

$$\mathbf{C} = \begin{bmatrix} \rho \\ \rho u \\ E \end{bmatrix} \qquad \mathbf{P} = \begin{bmatrix} \rho \\ u \\ p \end{bmatrix} \qquad J_{ij} = \frac{\partial C_i}{\partial P_j}$$

ideal gas:

$$\mathbf{J} = \begin{bmatrix} 1 & 0 & 0 \\ u & \rho & 0 \\ \frac{1}{2}u^2 & \rho u & \frac{1}{\gamma - 1} \end{bmatrix}$$

”Hierarchical slope limiting in explicit and implicit discontinuous Galerkin methods”, D. Kuzmin, *JCP* 257 (2014) 1140-1162.

*discussions with Ed Love (Dept. 1351)

HRFV ideal for shocks:

Sod: Traditional Multi-material test. Exact solution is available.

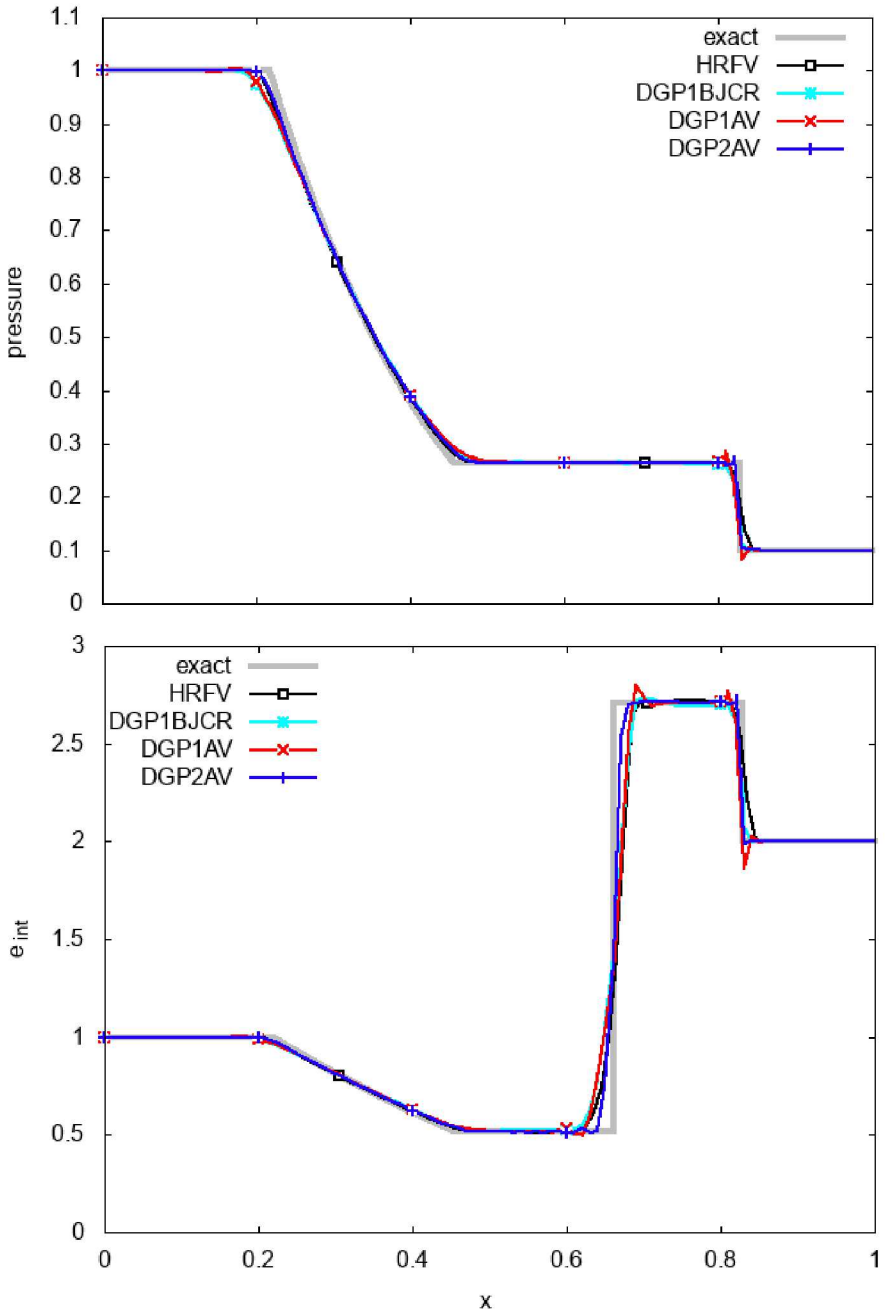
- HRFV captures the shock without oscillations.
- DG methods need additional dissipation for discontinuities (shock and contact here).
- All methods considered here smear the contact. Not surprising as contacts are not self-steepening.
- All methods appear to converge to the analytic solution with refinement.

$$\frac{\partial \alpha_i}{\partial t} + \nabla \cdot (\alpha_i \mathbf{v}) = \begin{cases} \alpha_i \nabla \cdot \mathbf{v} \\ \text{or} \\ \frac{\alpha_i K_i}{\overline{K}} \nabla \cdot \mathbf{v} \end{cases}$$

$$\frac{\partial \alpha_i \rho_i}{\partial t} + \nabla \cdot (\alpha_i \rho_i \mathbf{v}) = 0$$

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v} + P\mathbf{I}) = 0$$

$$\frac{\partial \rho E}{\partial t} + \nabla \cdot (\mathbf{v}\,(\rho E + P)) = 0$$



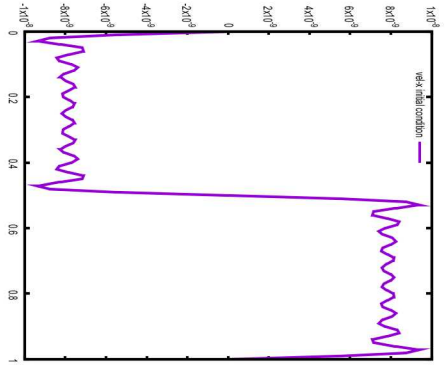
DG ideal for smooth waves

Dispersive Euler: single-material. Good test of methods on smooth data. Exact solution is available.
(http://ammar-hakim.org/sj/je/je5/je5-dispersive-eqns.html)

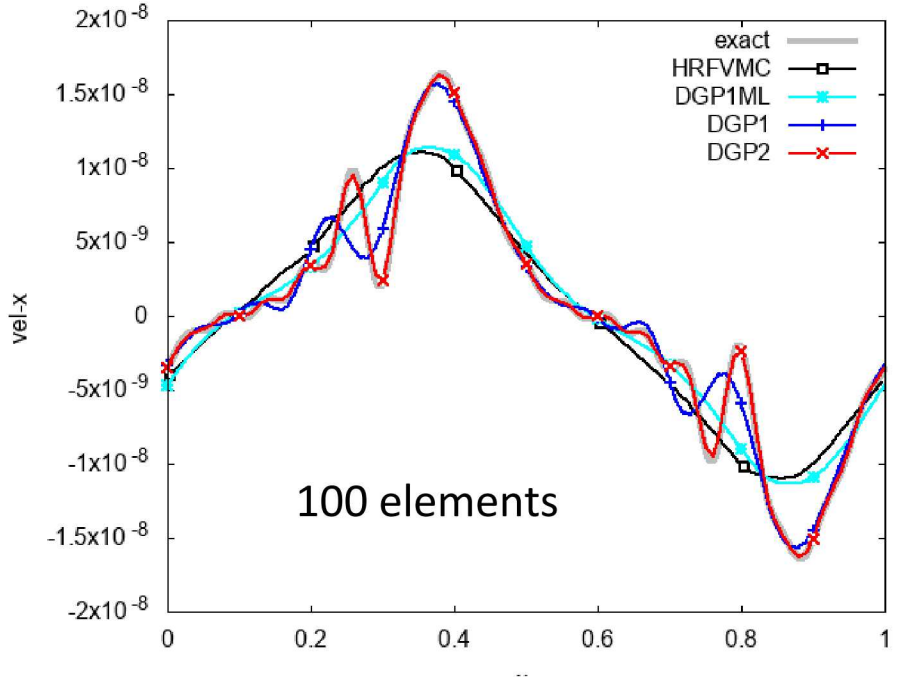
$$\frac{\partial \mathbf{Q}}{\partial t} + \nabla \mathbf{F} = \mathbf{S}$$

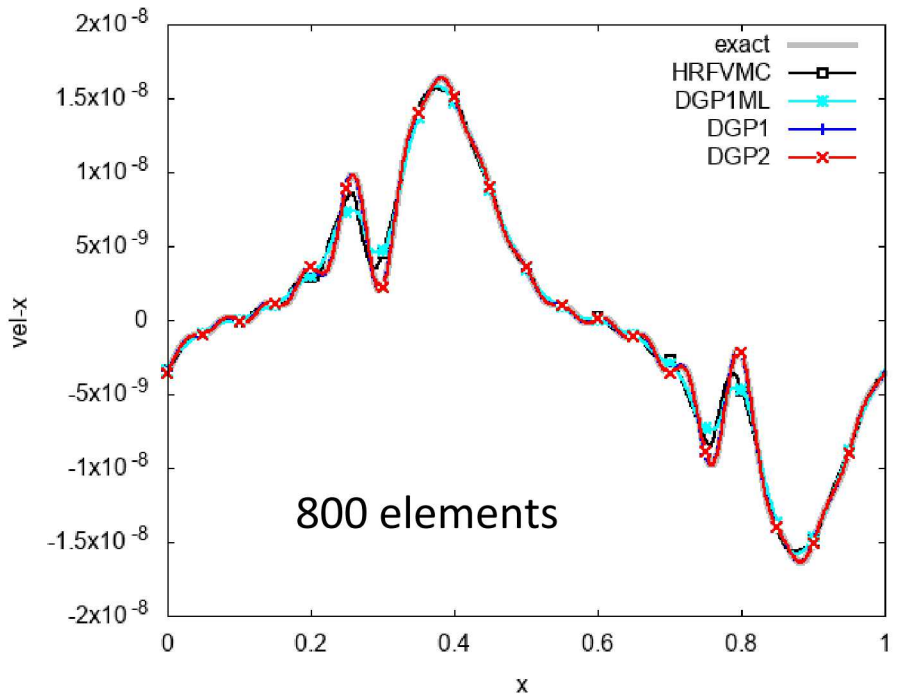
$$\mathbf{Q} = [\rho, \rho u, \rho v, \rho E]^T$$

$$\mathbf{S} = [0, \lambda v B_z, -\lambda u B_z, 0]^T$$



- HRFV misses much of the frequency content at lower resolutions.
- DG (without non-linear stabilization) demonstrates excellent results at low resolution.
- DG limiters degrade accuracy relative to unlimited approach.
- All methods appear to converge to analytic solution with refinement.





Hybrid DG/FV:

Can we apply different methods in different regions (or physics) according to their strengths?

- DG performs well in smooth regions but is not robust at shocks/discontinuities (fails immediately for Leblanc shock tube with AV approach).

- FV performs well at shocks/discontinuities but requires significant refinement there when using HRFV.

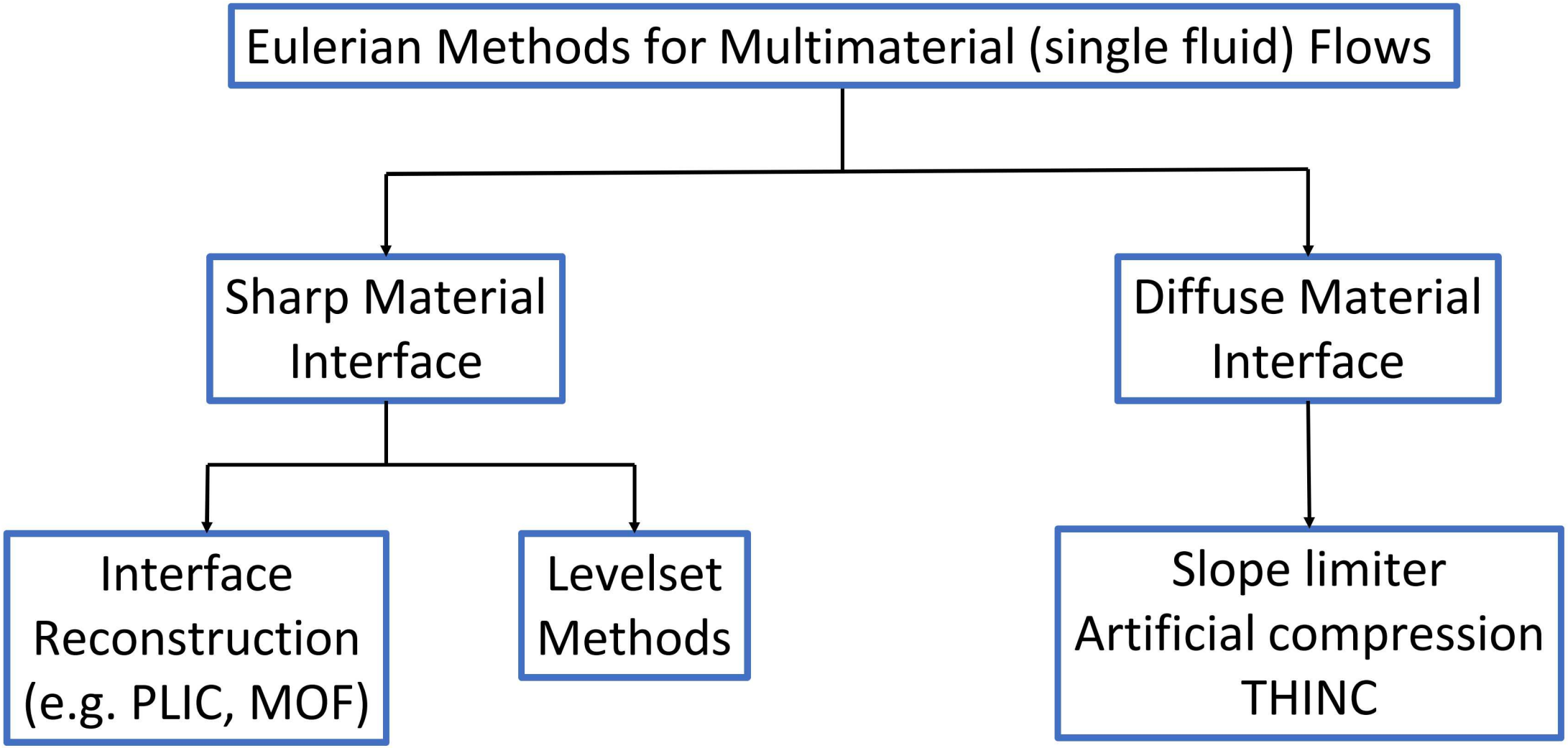
    - Hybrid DG / sub-grid FV: DG in smooth regions and (refined) FV for discontinuities.
        - Indicator can be used for selection of FV elements
        - Selection of FV regions would vary in space and time

    - Hybrid FV-hydro / DG-magnetics.
        - Mass, momentum and energy via FV
        - Maxwell eqns. Via DG.

Testing with FLEXI code (Ref).

Implementing directly in PERSEUS

A taxonomy of Eulerian multimaterial:



Explore diffuse multimat. methods:

- Most approaches are derivatives of Baer and Nunziato (1986).

- Five Equation Model (Allaire et al., 2002 and many others)
  - Single velocity/momentum equation.
  - Strong enforcement of pressure equilibrium.
  - Additional evolution equation(s) for volume fraction and individual component mass conservation statements.
  - Separate energy equations per component are not used, partitioning inherent in pressure equilibration.
- Six Equation Model (Saurel et al., 2009)
  - Single velocity/momentum equation.
  - Multiple pressures (one per material) with relaxation step.
  - Multiple internal energy equations (one per material).
  - Evolution equation for material volume fractions.
  - Additional equation for mixture total energy.

Five Equation Evolution System:

Volume fraction: $\frac{\partial \alpha_i}{\partial t} + \mathbf{v} \cdot \nabla \alpha_i = 0$ ⟹ $\frac{\partial \alpha_i}{\partial t} + \nabla \cdot (\alpha_i \mathbf{v}) = \alpha_i \nabla \cdot \mathbf{v}$

Treat as conservation + source

Mass conservation: $\frac{\partial \alpha_i \tilde{\rho}_i}{\partial t} + \nabla \cdot (\alpha_i \tilde{\rho}_i \mathbf{v}) = 0$

Momentum conservation: $\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v} + P\mathbf{I}) = 0$

Total energy conservation: $\frac{\partial \rho E}{\partial t} + \nabla \cdot (\mathbf{v}\,(\rho E + P)) = 0$

Closure model:

$$\sum_j \alpha_j = 1$$

$$\rho = \sum_j \alpha_j \tilde{\rho}_j$$

$$e = \sum_j \alpha_j \tilde{\rho}_j \; \tilde{e}_j \; [\tilde{\rho}_j, p]$$

$$p = p_k$$

HLLC Approximate Riemann Solver:

We use HLLC solver with some modifications to treat multiple materials:

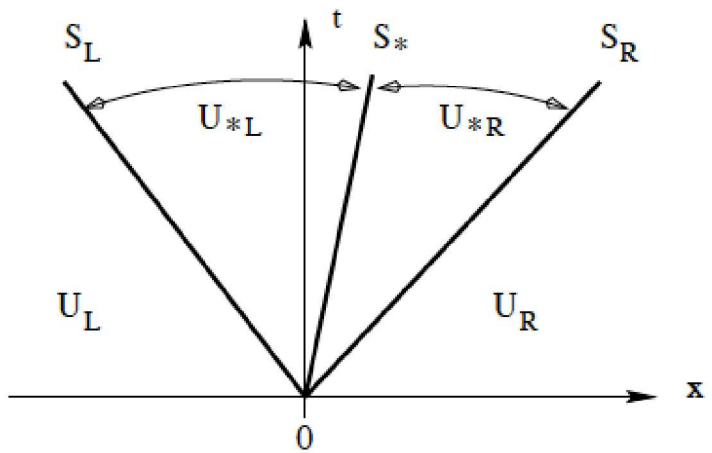
$$\hat{\mathbf{F}} = \begin{cases} \mathbf{F}_L & \text{if} \quad 0 \le S_L \\ \mathbf{F}_L + S_L\left(\mathbf{U}_{*L} - \mathbf{U}_L\right) & \text{if} \quad S_L \le 0 \le S_* \\ \mathbf{F}_R + S_R\left(\mathbf{U}_{*R} - \mathbf{U}_R\right) & \text{if} \quad S_* \le 0 \le S_R \\ \mathbf{F}_L & \text{if} \quad S_R \le 0 \end{cases}$$

$$S_L = \min\left(u_L - c_L, u_R - c_R\right)$$

$$S_R = \min\left(u_L + c_L, u_R + c_R\right)$$

$$c^2 = \frac{\sum_j \tilde{\rho}_j \alpha_j c_j^2}{\sum_j \tilde{\rho}_j \alpha_j}$$

$$S_* = \frac{p_R - p_L + \rho_L u_L\left(S_L - u_L\right) - \rho_R u_R\left(S_R - u_R\right)}{\rho_L\left(S_L - u_L\right) - \rho_R\left(S_R - u_R\right)}$$



$$\mathbf{U}_{*K} = \begin{cases} \alpha_{iK} \\ \left(\alpha_i \tilde{\rho}_i\right)_K \frac{\left(S_K - u_K\right)}{\left(S_K - S_*\right)} \\ \rho_K S_* \frac{\left(S_K - u_K\right)}{\left(S_K - S_*\right)} \\ \rho_K \frac{\left(S_K - u_K\right)}{\left(S_K - S_*\right)}\left(\frac{E_K}{\rho_K} + \left(S_* - u_K\right)\left[S_* + \frac{p_K}{\rho_K\left(S_K - u_K\right)}\right]\right) \end{cases}$$

Batten et al., 1997; Saurel et al., 2009; Toro et al., 1994

Care needed in solving VF evolution:

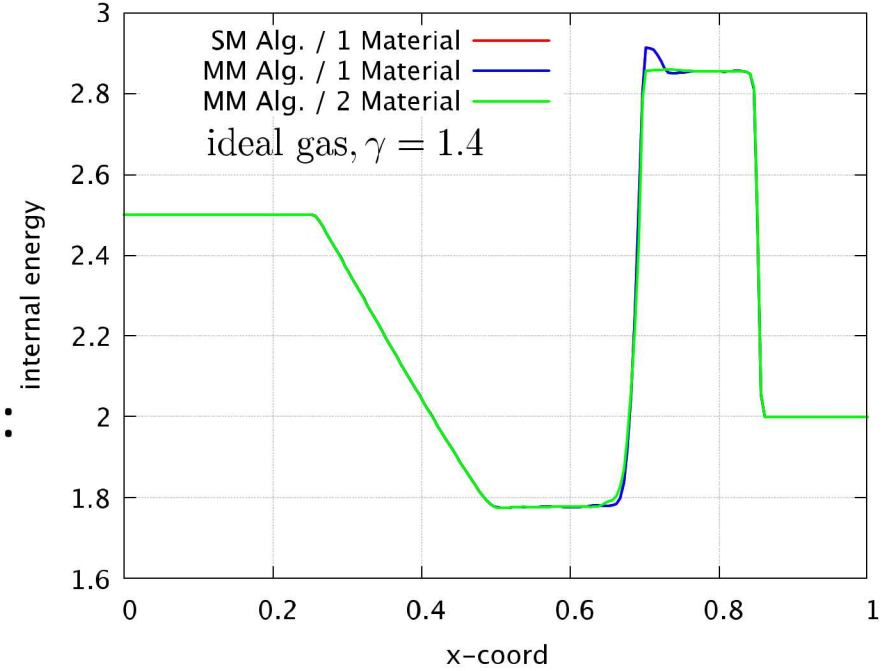
$$\frac{\partial \alpha_i}{\partial t} + \nabla \cdot (\alpha_i \mathbf{v}) = \alpha_i \nabla \cdot \mathbf{v}$$

Require recovery of single material formulation:

$$\alpha = 1 \qquad \frac{\partial \alpha}{\partial t} = 0$$

Analysis of discretization demonstrates requirements:

$$\frac{\partial \alpha}{\partial t} = \frac{\Delta t}{\Delta x}\left[\left(\hat{F}_L - \hat{F}_R\right) + \alpha^n \left(v_R^* - v_L^*\right)\right]$$



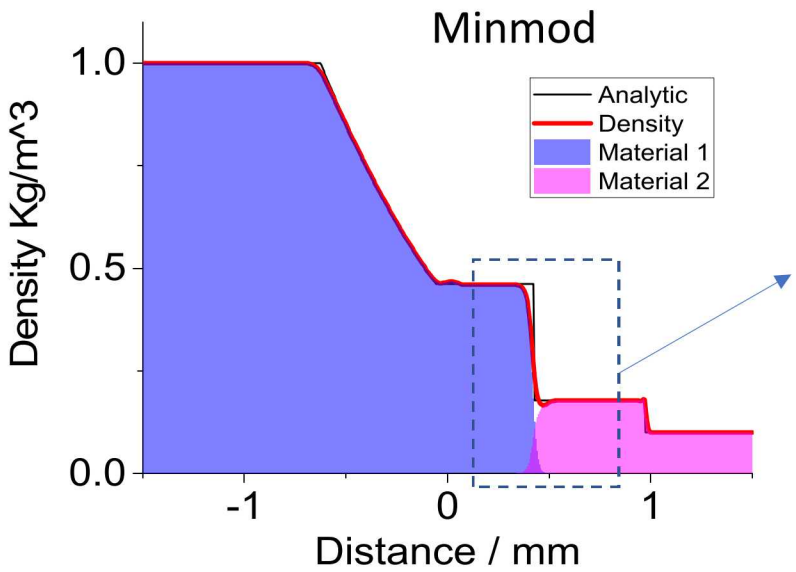
Sod with single and multimaterial algorithms.

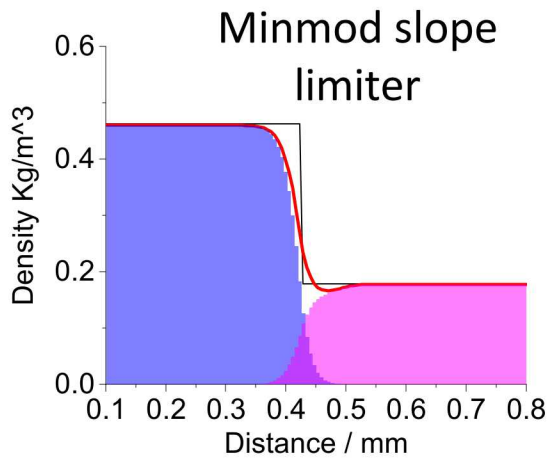
Assume LLF:

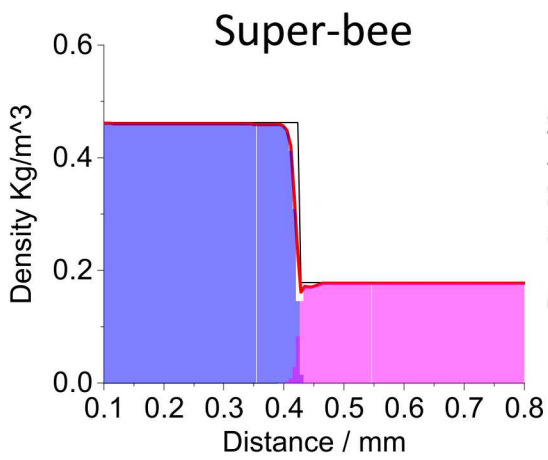
$$\hat{F}_L = \frac{1}{2}\left[v_L^- \alpha_L^- + v_L^+ \alpha_L^+\right] + S\left[\alpha_L^- - \alpha_L^+\right] = \alpha \frac{1}{2}\left[v_L^+ + v_L^-\right]$$

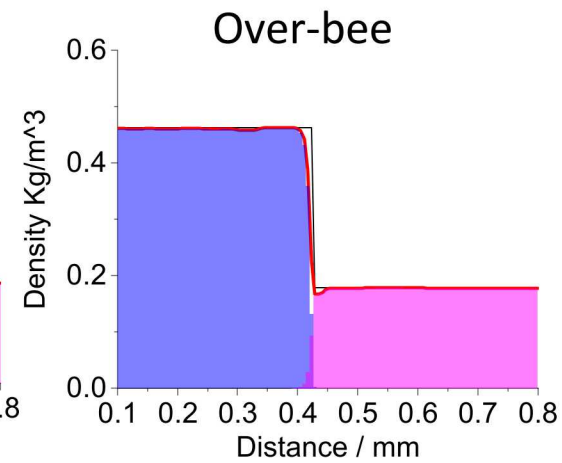
$$v^* = \frac{(v^+ + v^-)}{2} \quad \Leftarrow \quad \alpha\left[v_L^+ + v_L^-\right] - \alpha\left[v_R^+ + v_R^-\right] + \alpha\left(v_R^* - v_L^*\right) = 0$$

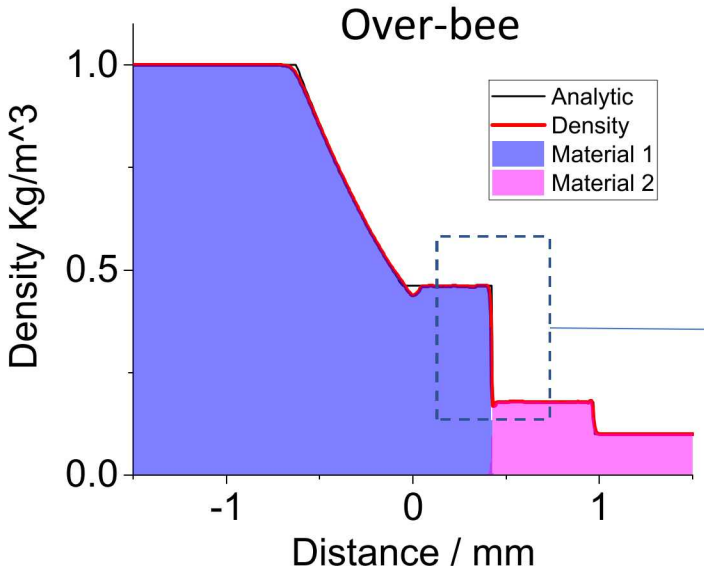
Five equation model behavior

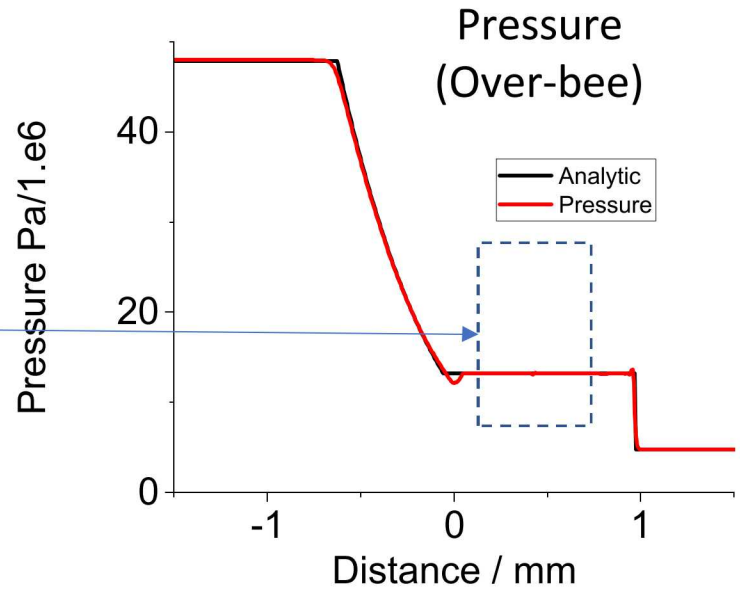








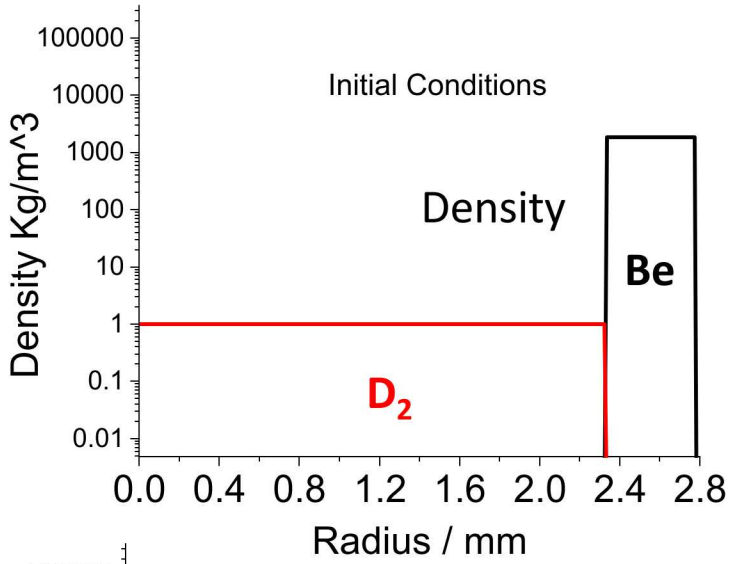




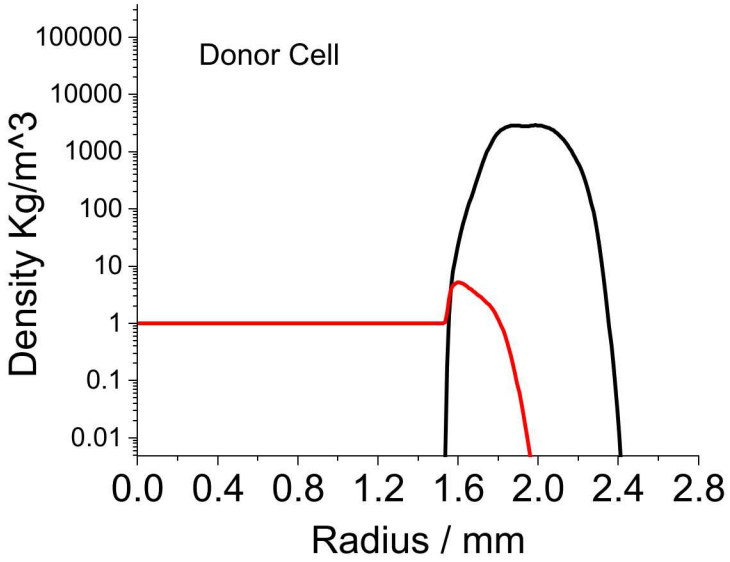
Constant pressure maintained over 2 material interface

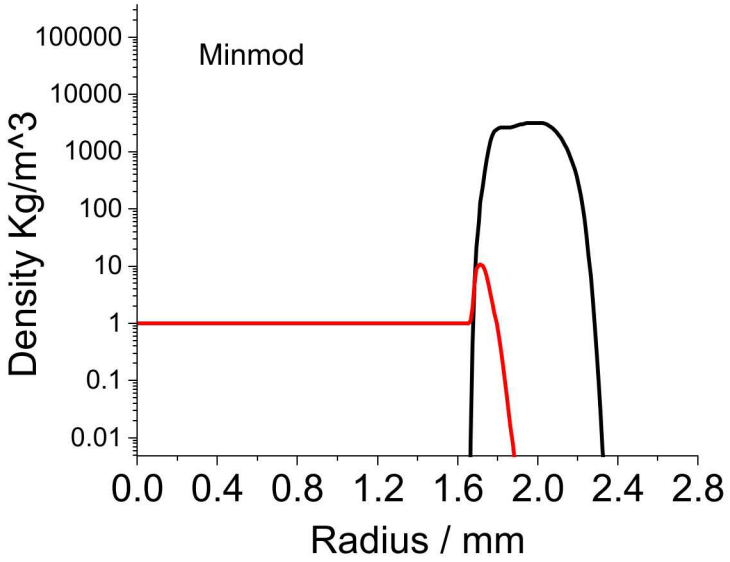
Multimaterial (5eqn model)

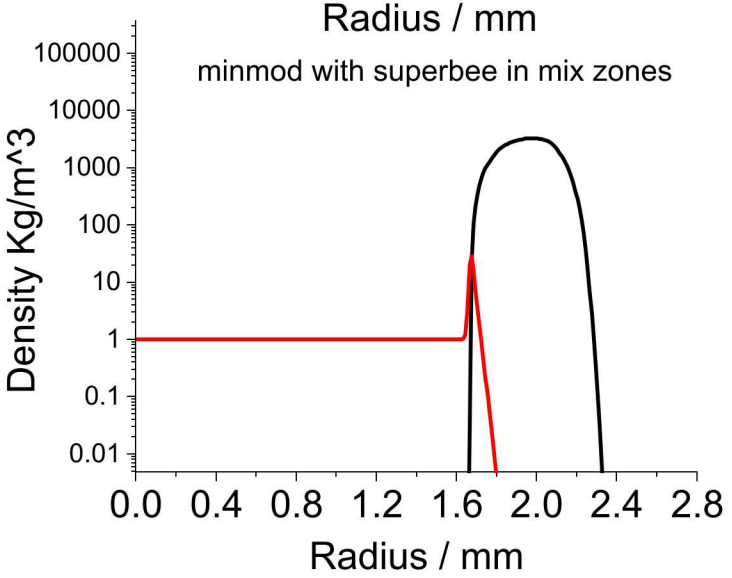
Studying applicability and implementation of finite volume 5 equation model to magnetically driven MagLif style liner implosion (shocked Beryllium liner driving into ~1mg/cc D_2 gas Driven with Sesame EOS & Conductivities).



90ns into 100ns rise to 20MA on sin^2 current profile





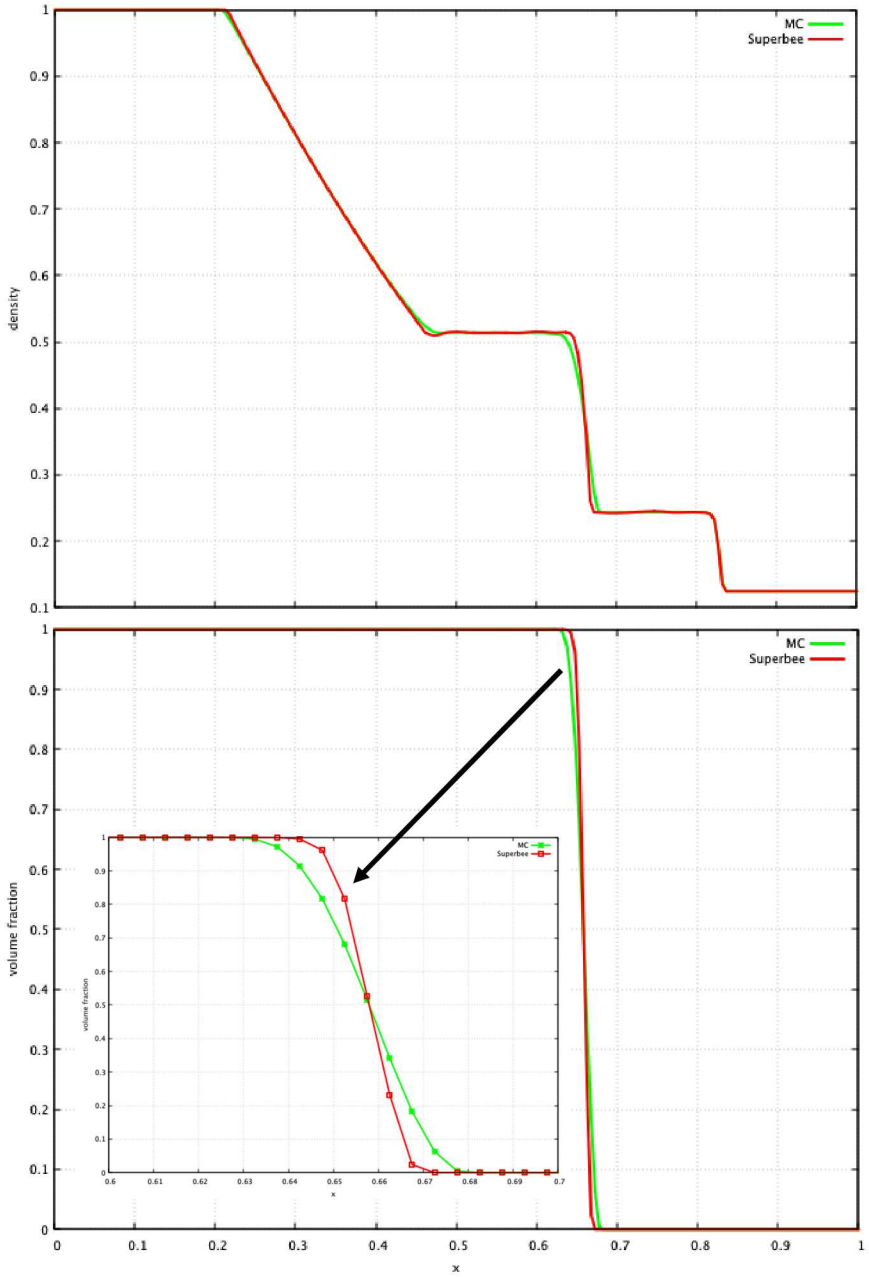


Thoughts on multimat:

- Properly resolving contacts is surprisingly(?) difficult as they do not have physical sharpening as do shocks. This becomes more problematic when dealing with multiple materials and mixing.

- Diffuse interface methods used here allows some inter-mixing of material with the amount dictated by choice of discretization.
  - Simple sharpening schemes (e.g. Super-bee, Over-bee, etc.) can help.
  - "Adaptivity" at contact can help.
  - Artificial compression methods that can provide O(h) refinement on contacts are now being .

- Strong enforcement of pressure continuity is problematic for mixing of "real" (e.g. tabular EOS-base) materials. Starting to implement/test pressure relaxation approaches (Miller and Puckett, 1996; Saurel et al., 2009).



Conclusions:

- Numerous developments are underway.

- Framework:
  - Block-based mesh refinement now available.
  - Designed to be NGP (MPI-GPU) for performant code base.
  - PERESUS algorithms are being implemented.
- Discretizations:
  - Optimal choice depends on physics regime.
  - Appropriate method may require a hybrid approach.
- Multimaterial:
  - Exploring diffuse methods.
  - Studying approaches that address contact capturing and state equilibration.

- Developing a broad set of tools and expertise that can be applied strategically.

Many mixture models:

- Ideal mixing rule (rarely used):

$$p = \sum_i x_i \boxed{p_i\left[\rho, T\right]}$$

EOS call

- Law of partial pressures (Dalton's Law):

$$p = p_1\left[\rho_1, T\right] = p_2\left[\rho_2, T\right] = ... = p_m\left[\rho_m, T\right]$$

pure material density: $\tilde{\rho}_i = \dfrac{m_i}{V_i}$

partial density: $\rho_i = \dfrac{m_i}{V_T}$

mass fraction: $x_i = \rho_i / \rho$

volume fraction: $\alpha_i = \dfrac{V_i}{V_T}$

- Law of additive volumes (Amagat's Law; many hydro-codes):

$$p = p_1\left[\tilde{\rho}_1, T\right] = p_2\left[\tilde{\rho}_2, T\right] = ... = p_m\left[\tilde{\rho}_m, T\right] \qquad \rho = \sum_i \alpha_i \tilde{\rho}_i$$

- Magyar and Mattsson (2013) suggest that Amagat's Law is superior though more recent work (Wayne et al., 2020) suggests neither is a good representation for shock-hydro problems.
- None of these methods are completely satisfactory for numerical hydrodynamics. None-the-less, we will use law of additive volumes to obtain mixture pressure, density and internal.