Simulation of Multi-Material Flows Using a Finite Element Riemann Solver
(LA-UR-13-26801)

T. R. Canfield, J. I. Waltz, N. R. Morgan, M. R. J. Charest, L. D. Risinger Jr. and J. G. Wohlbier
Los Alamos National Laboratory
Los Alamos, NM 87585

Outline

- Chicoma Hydrodynamic Equations
- Multi-Material Formulation
- Hyper-C Implementation
- Results
- Conclusions

Chicoma
3D unstructured mesh code

- Tetrahedral Finite Element Mesh
- Edge based with node centric data
- Eulerian, Lagrangian and (ALE)
- Mesh is moved by adjusting velocity rather than moving points
- Adaptive grid refinement

Chicoma Hydrodynamic Equations
Eulerian Flux Conservative Form

$$\frac{\partial U}{\partial t} + \frac{\partial F_j}{\partial x_j} = 0$$

$$U = \begin{pmatrix} \rho \\ \rho u_i \\ \rho E \end{pmatrix} \quad F_j(U) = \begin{pmatrix} \rho u_j \\ \rho u_i u_j + p\delta_{ij} \\ u_j(\rho E + p) \end{pmatrix}$$

Chicoma Hydrodynamic Equations
Spatial Operators and Discretization of Edges

$$M_L^v \frac{dU^v}{dt} + \sum_{e\in v}\left[D_j^{vw}\left(F_j^v + F_j^w\right) + \left|\lambda^{vw}\right|\left({}^{+}U^w - {}^{+}U^v\right)\right] = 0$$

$$D_j^{vw} = \frac{1}{2}\sum_{\Omega_h \in e}\int_{\Omega_h}\left(\frac{\partial N^v}{\partial x_j}N^w - N^v\frac{\partial N^w}{\partial x_j}\right)d\Omega_h$$

$${}^{+}U^v = U^v + \frac{1}{2}\left[\frac{1}{2}(1-\zeta)\varphi\left(r^v\right)\delta_1 + \frac{1}{2}(1+\zeta)\varphi\left(\frac{1}{r^v}\right)\delta_2\right]$$

$${}^{+}U^w = U^w - \frac{1}{2}\left[\frac{1}{2}(1+\zeta)\varphi\left(r^w\right)\delta_1 + \frac{1}{2}(1-\zeta)\varphi\left(\frac{1}{r^w}\right)\delta_2\right]$$

Chicoma Hydrodynamic Equations

Limiters (only 2nd order accurate):

$$r^v = \frac{\delta_2 + \varepsilon}{\delta_1 + \varepsilon}, \quad r^w = \frac{\delta_2 + \varepsilon}{\delta_3 + \varepsilon},$$

$$\delta_2 = U^w - U^v$$

$$\delta_1 = 2x_i^{vw} \frac{\partial U^v}{\partial x_i} - \delta_2$$

$$\delta_3 = 2x_i^{vw} \frac{\partial U^w}{\partial x_i} - \delta_2$$

$$x_i^{vw} = x_i^w - x_i^v$$

Multi-Material Formulation
Building Blocks:

- Basic Data Structures
  - Single Phase solid, liquid, gas or plasma
  - Multi-Phase Materials
    - Equilibrium EOS with all phases
    - EOS and strength for each phase
  - Multiple Materials in the volume surrounding a node
    - Single phase solid, liquid, gas or plasma
    - Multi-Phase materials

Multi-Material Formulation
Building Blocks:

- EOS
  - Analytic:
    - gamma-law gas
    - Mie-Grüneisen
  - SESAME (EOSPac6)

- Strength (Under Construction)
  - Analytic:
    - Lindemann Melt Temperature
    - PTW, Steinberg-Guinan
  - EOSPac6:
    - Cold Shear Modulus
    - Melt/Freeze Temperature

Multi-Material Formulation
Volume and Mass Fractions:

In a mixed region

$$\phi_k = \frac{V_k}{V}$$

$$m_k = \frac{M_k}{M}$$

with the following constraints

$$\sum_k \phi_k = 1$$

$$\sum_k m_k = 1$$

Multi-Material Formulation
Volume Averaged Stress and Strain:

In a mixed region

$$\bar{p} = \sum_k \phi_k p_k$$

$$\bar{\underset{\sim}{\sigma}} = \sum_k \phi_k \underset{\sim}{\sigma}_k$$

$$\bar{\mu} = \sum_k \phi_k \mu_k$$

$$\bar{\underset{\sim}{\varepsilon}} = \sum_k \phi_k \underset{\sim}{\varepsilon}_k$$

Above holds regardless of the closure model.

Multi-Material Formulation
Closure Models:

- Intent is to use method that is locally appropriate
- Closure Models
  1. Uniform Strain
  2. Tipton like pressure equilibration/relaxation
  3. P-T Equilibration

Hyper-C Implementation in Chicoma
Volume Fraction Advection

- Take advantage of the efficient flux machinery available in Chicoma.
- Advection of volume fraction rather use expensive interface reconstruction.
- To do this we solve this equation

$$\frac{\partial \phi}{\partial t} + \nabla \cdot \left(\phi \underset{\sim}{u} \right) = 0 \quad \begin{array}{ll} \phi = 1 & \text{Zone is full} \\ 0 < \phi < 1 & \text{Zone is partially filled} \\ \phi = 0 & \text{Zone is empty} \end{array}$$

Hyper-C Implementation in Chicoma
Volume Fraction Advection

Volume Fractions	
ϕ^a	Acceptor cell
ϕ^d	Donor cell
ϕ^f	Interface between acceptor and donor cell
ϕ^u	Upwind neighbor cell

Hyper-C Implementation in Chicoma
Consider the Flow on an Edge Connecting Two Nodes

Define

$$r^k = \frac{\phi^k - \phi^u}{\phi^a - \phi^u}$$

$$\phi^k = r^k\left(\phi^a - \phi^u\right) + \phi^u$$

At the interface between and acceptor and donor cell we have

$$\phi^f = r^f\left(\phi^a - \phi^u\right) + \phi^u$$

Hyper-C Implementation in Chicoma

Maximum value of r^f

The maximum value that r^f can assume is

$$r^f_{\max} = \begin{cases} \min\left(1, \frac{1}{c} r^d\right) & 0 \le r^d \le 1 \\ r^d & \text{otherwise} \end{cases}$$

with

$$c = |\underset{\sim}{u}|^f \frac{\Delta t}{l}$$

$$r^d = \frac{\phi^d - \phi^u}{\phi^a - \phi^u}$$

Hyper-C Implementation in Chicoma
Ubbink-Quickest Scheme:

Modified with the ultimate quickest scheme (U-Q) for tangential advection in multidimensional flows

$$r^f_{U-Q} = \begin{cases} \min\left(\dfrac{8cr^d + (1-c)(6r^d+3)}{8}, r^f_{\max}\right) & 0 \le r^d \le 1 \\ r^d & \text{otherwise} \end{cases}$$

Hyper-C Implementation in Chicoma
Ubbink-Issa Blending:

Blending of Hyper-C and U-Q limiters:

$$r^f = \alpha r^f_{\max} + (1-\alpha) r^f_{U-Q}$$

$$\cos\theta = \frac{\underset{\sim}{\nabla}\phi^d \cdot \underset{\sim}{n}^f}{\left|\underset{\sim}{\nabla}\phi^d\right|}$$

$$\alpha = \min\left(\kappa \cos^2\theta, 1\right)$$

A scalar advection test problem has been used to evaluate advection methods for material interfaces

- Solve linear advection equation on 3D cylindrical tube 100 units in length:

$$\frac{\partial \phi}{\partial t} + \underset{\sim}{u} \cdot \nabla \phi = 0$$

- Initial conditions:

$$\phi(\underset{\sim}{x}, 0) = \begin{cases} 1.0 & z \leq 50 \\ 0.5 & z > 50 \end{cases}$$

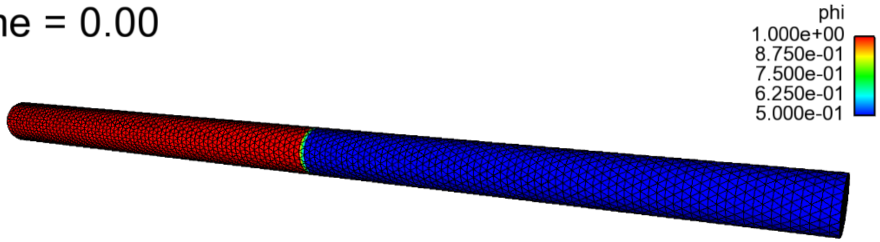
$$u_x = u_y = 0, \qquad u_z = 1$$

- Exact solution:

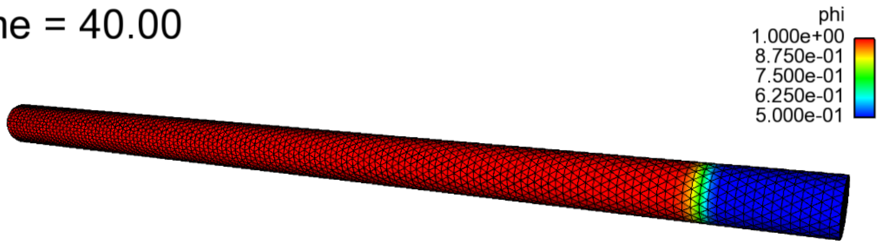
$$\phi(\underset{\sim}{x}, t) = \begin{cases} 1.0 & z \leq 50 + u_z t \\ 0.5 & z > 50 + u_z t \end{cases}$$

- Evaluate error at *t = 5* (images show *t = 40*)

Time = 0.00

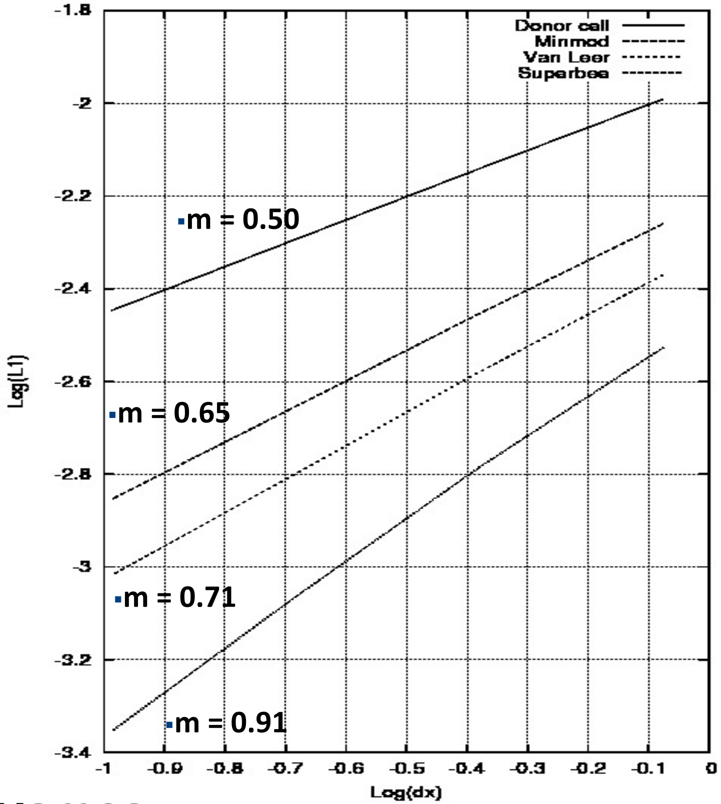


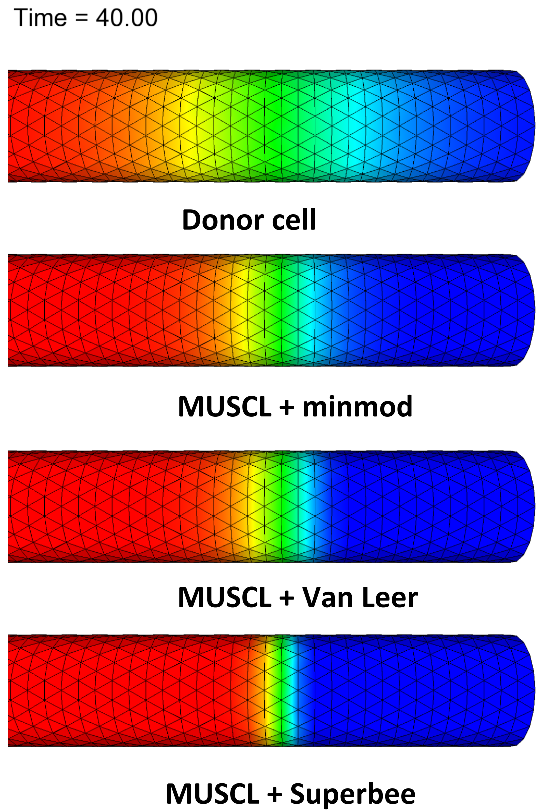
Time = 40.00



An initial implementation of the hyper-C method performs as expected

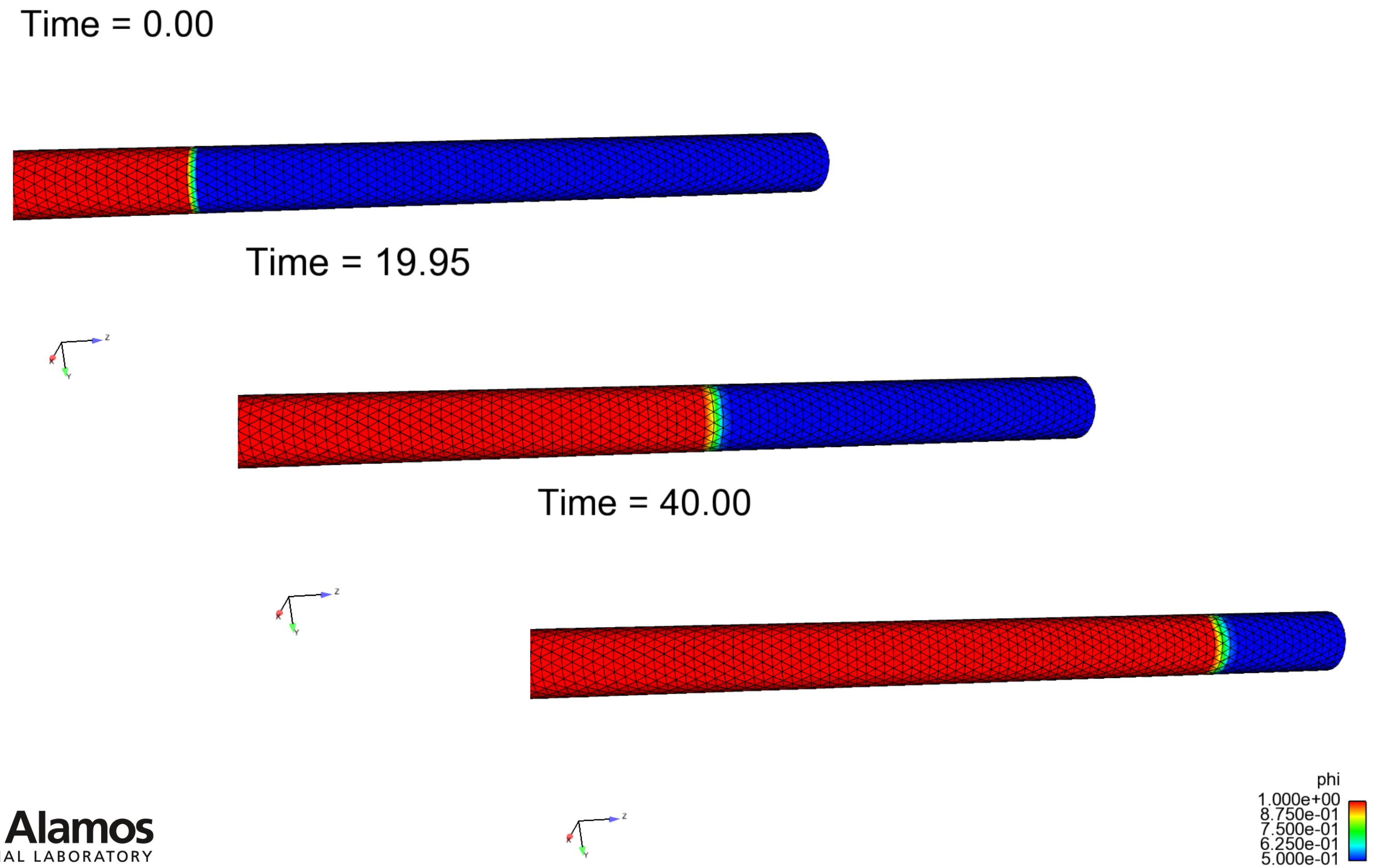
- The hyper-C method reduces numerical diffusion and maintains 1^{st}-order convergence





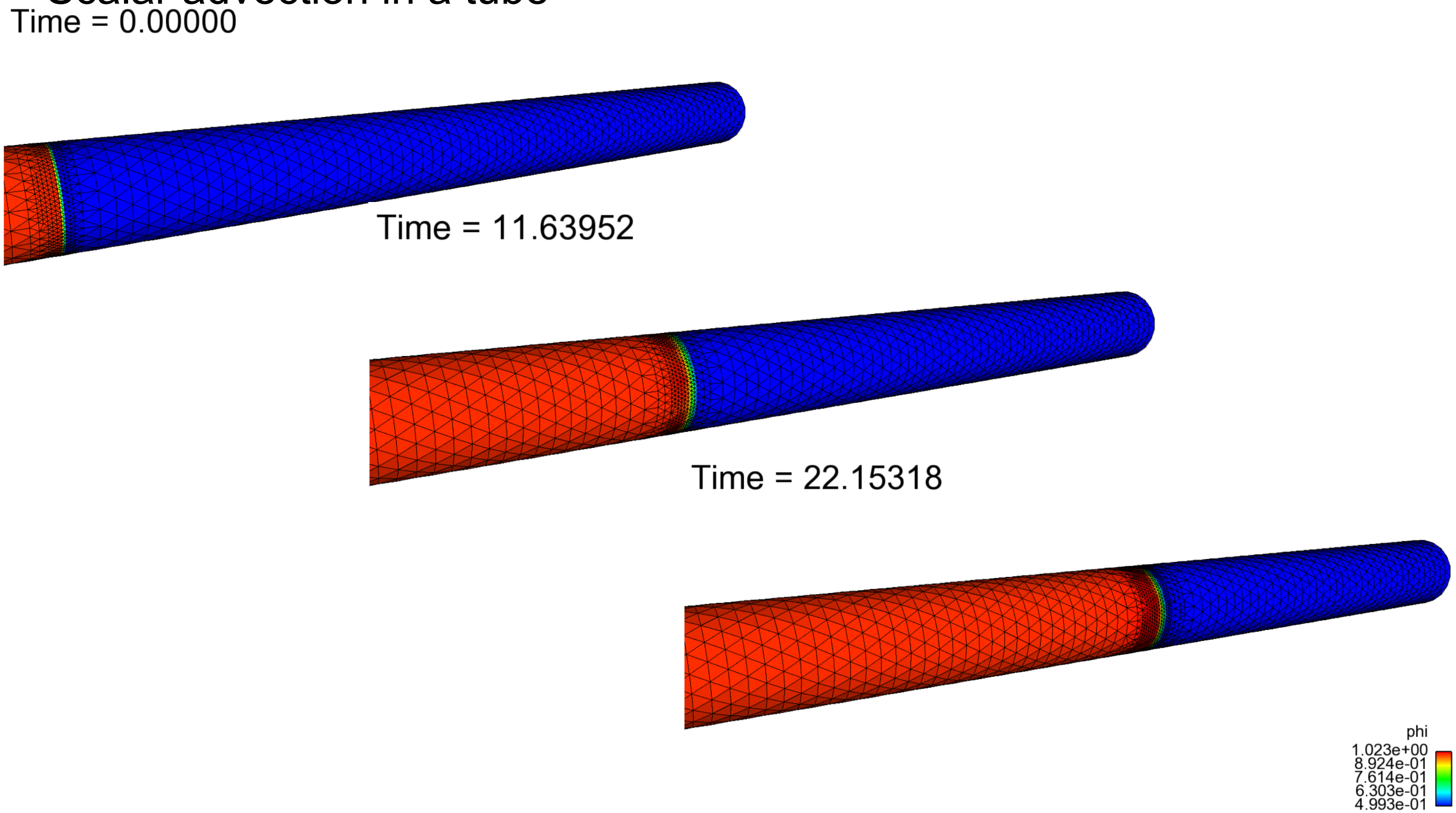
Results with the Modified Hyper-C Advection Strategy

- Scalar advection in a tube



Results with the Modified Hyper-C Advection Strategy with AMR

- Scalar advection in a tube



Conclusions

- We have implemented a variation of the Hyper-C method on our tetrahedral finite element grid
- Results produce sharp interfaces that are contained within the volume surrounding a node.
- We are studying convergence and accuracy on various problems with the method in Chicoma