Progress towards modeling hypersonic fluid-structure interactions with large structural deformations

Madeline Peck
Fluid Dynamics and Solid Mechanics Group, Los Alamos National Laboratory

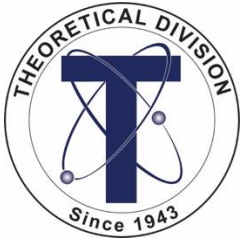
Advanced Modeling & Simulation (AMS) Seminar Series
NASA Ames Research Center, October 31st, 2024

LA-UR-24-31615

Acknowledgements

- Cartablanca++ Team
  - Duan Zhang
  - Jiajia Waters
  - Paul Barclay
  - Kyle Perez
  - Sara Calandrini
  - Jeremiah Brackbill
  - Christopher Long

- Funding
  - Advanced Simulation and Computing (ASC) program
  - Engineering and Technology Maturation (ETM), Delivery Environments program





Outline

- Introduction
  - Motivation, choice of numerical methods
- Discontinuous Galerkin (DG)
  - Methods, shock-capturing literature
  - Highlighted results
- Material Point Method (MPM)
  - Highlighted results
- Plans for fluid-structure interactions (FSI)
  - Multi-velocity method
- Summary

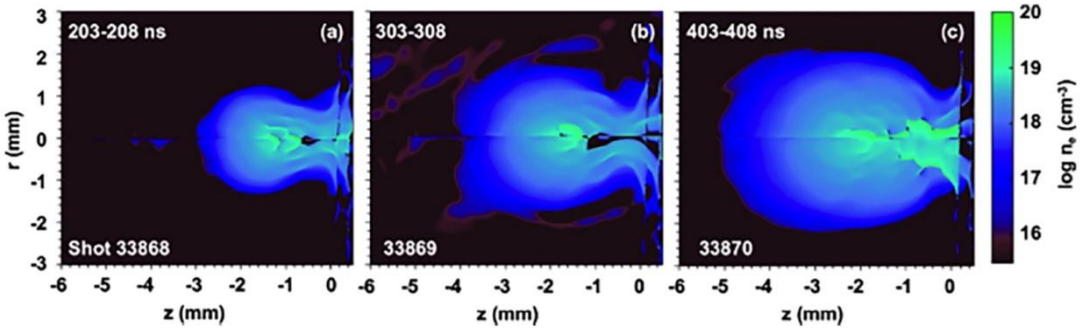
Motivation

- Blast effects are difficult to study experimentally
  - Destructive
  - Difficult to measure and expensive
- Strong pressure waves also characteristic of hypersonic flight, shocks
- Fluid interactions with surface could cause large deformations, damage

Obj: Develop numerical capability for simulating coupled fluid-structure interactions under extreme conditions with large structural deformations.

Motivation

- Example: Dual Axis Radiographic Hydrodynamic Test (DARHT) Facility
  - Problem: Modeling a solid target vaporized by an electron beam

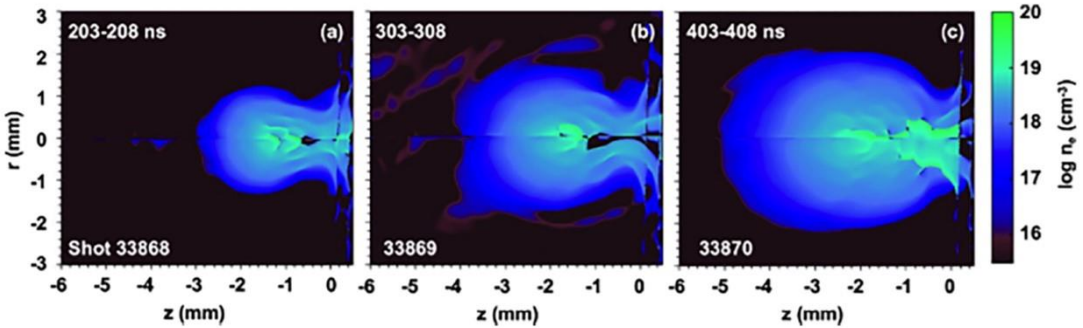


Coleman et al., Journal of Applied Physics (2022)

- Eulerian methods
  - Good for fluid/gas dynamics
  - Numerical diffusion at interfaces is an issue, unless methods like IBM and/or level set are used which induce mass, momentum and energy conservation issues.

Motivation

- Example: Dual Axis Radiographic Hydrodynamic Test (DARHT) Facility
  - Problem: Modeling a solid target vaporized by an electron beam



Coleman et al., Journal of Applied Physics (2022)

- Particle method
  - Particle methods can be used to ensure sharp interfaces.
  - Conservations (mass, momentum) can be built-in.

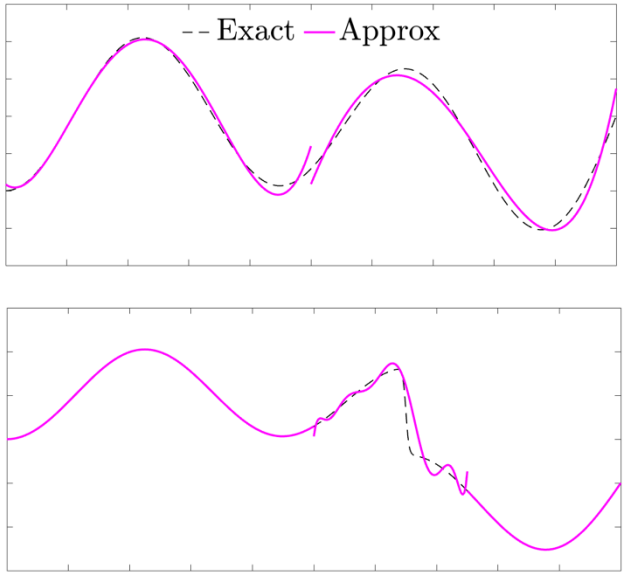
Numerical Methods for FSI

- Fluids: discontinuous Galerkin (DG), Eulerian approach
  - Arbitrary polynomials → high-order accurate
  - Superior parallel scalability due to local operator
  - Natural extension to h/p adaptivity
- Solids: material point method (MPM), Eulerian/particle approach
  - Handles large material deformation w/o mesh tangling
  - Material points carry history dependent information
- Interaction: multi-velocity formulation
  - Implicit interface tracking

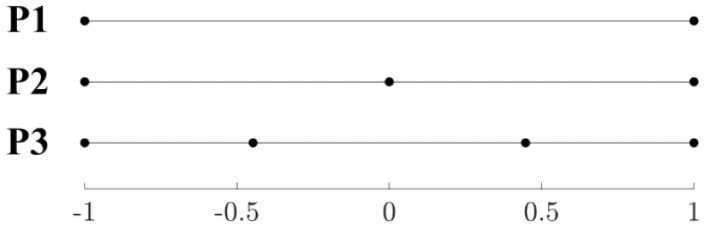
Discontinuous Galerkin
Methods

Discontinuous Galerkin FEM (DGFEM)

- Arbitrary polynomial order
- No continuity constraints between elements
  - Riemann problems
- Potential for excellent computational efficiency
  - Operators are local → parallelize well
  - unstructured meshes
  - h/p adaptation
- Shocks + high-order polynomials = oscillations



Cartablanca++



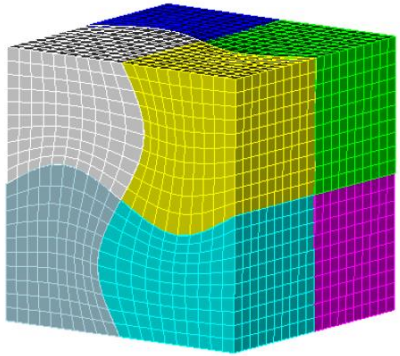
- DGFEM
  - Unstructured 3D hexahedral elements
  - Points at Legendre Gauss Lobatto nodes (tensor product)
  - Legendre polynomial basis functions
  - Flux functions: Lax Friedrichs, Roe, HLLC, HLLE
  - Explicit 3 stage 4th order SSP-RK
- Conservation Law
  - Inviscid, calorically perfect gas

$$\frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot \vec{\mathbf{F}} = 0.$$

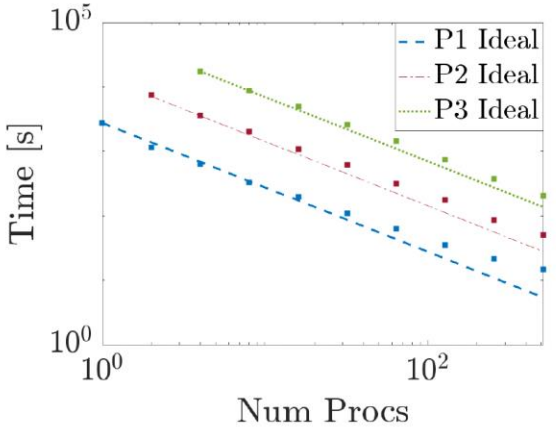
$$\mathbf{F} = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u(E+p) \end{pmatrix}, \quad \mathbf{G} = \begin{pmatrix} \rho v \\ \rho vu \\ \rho v^2 + p \\ \rho vw \\ v(E+p) \end{pmatrix}, \quad \mathbf{H} = \begin{pmatrix} \rho w \\ \rho wu \\ \rho wv \\ \rho w^2 + p \\ w(E+p) \end{pmatrix}.$$

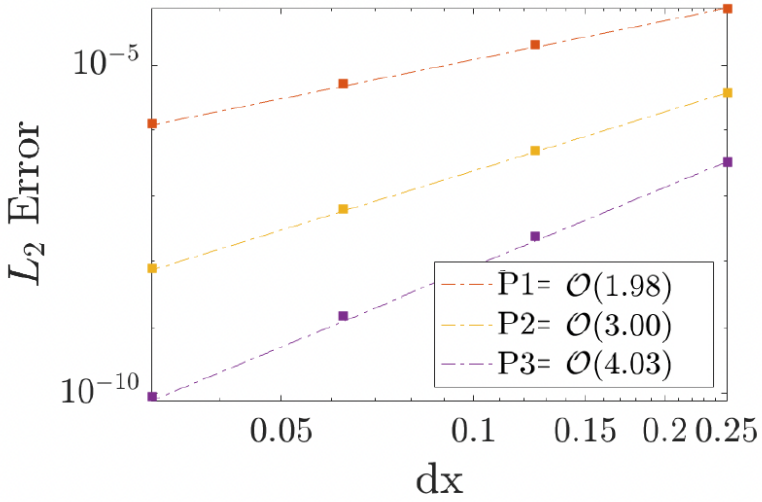
Hesthaven and Warburton, *Nodal discontinuous Galerkin methods: algorithms, analysis, and applications,* 2007

Cartablanca++ -- Verification & Scaling



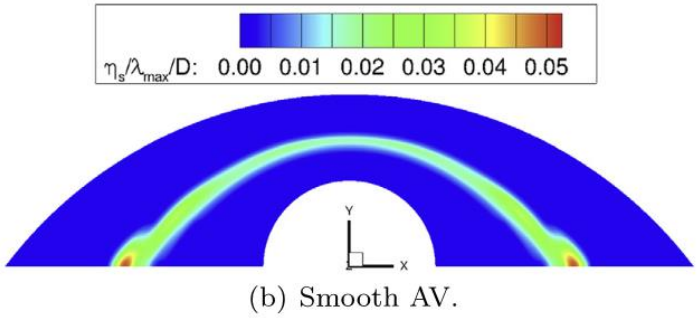
- Verification w/ Method of Manufactured Solutions (MMS)
  - Pure DGFEM algorithm
  - Expected order of convergence attained
- Scaling
  - MMS problem
  - Improves w/ P





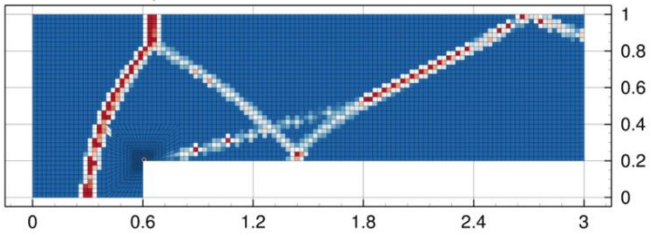
Shock-capturing overview

- Two important components:
  - Shock detection
  - Shock treatment
- Three families of strategies:
  - Shock-tracking: align cell boundaries with shocks
  - Targeted dissipation: add extra dissipation near shocks
  - Slope limiters: reduce variable slopes to constrain solution



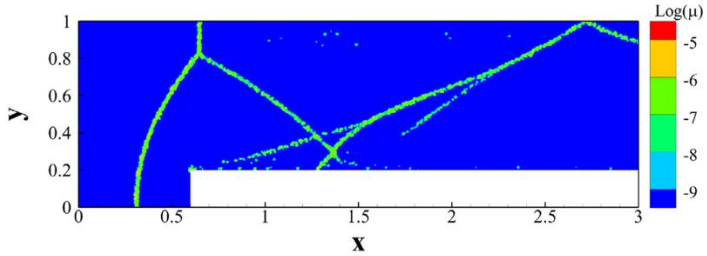
(b) Smooth AV.

Ching et al., Journal of Computational Physics 2019



Hennemann et al., Journal of Computational Physics (2021)

Yu & Hesthaven, Computers & Fluids (2022)



(c) *P4*

Artificial Viscosity (AV)

[1]Haga and Kuwai, Journal of Computational Physics (2019)
[2]Terashima et al., Computers & Fluids (2013)
[3]Bassi and Rebay, Journal of Computational Physics (1997)
[4]Yu and Hesthaven, Communications in Computational Physics (2020)

- Mimic physical viscosity to add additional dissipation around shocks
  - Coefficients for artificial bulk viscosity (μ_β) and mass (μ_χ) diffusion[1,2]
    - μ_β and μ_χ detect shocks and control AV magnitude
- Calculation procedure:
  - Compute gradients at nodes (BR1[3])
  - Compute pointwise μ_β, μ_χ

$$\mu_\beta = C_\beta h^2 \rho \min(\nabla \cdot \vec{\mathbf{u}}, 0) \qquad \mu_\chi = C_\chi \frac{ch^2}{\rho} |\nabla \rho|,$$

  - $C_\beta \approx 0.25$, $C_\chi \approx 0.01$ (tuning required)
  - Vertex-based smoothing procedure[4]
    - At a vertex, average all cell avg. values from cells that share the vertex
    - Linearly interpolate from vertices back to nodes

$$\mathbf{G}_{AV} = \mu_\beta \begin{pmatrix} 0 \\ (\nabla \cdot \vec{\mathbf{u}})\delta_{1j} \\ (\nabla \cdot \vec{\mathbf{u}})\delta_{2j} \\ (\nabla \cdot \vec{\mathbf{u}})\delta_{3j} \\ \vec{\mathbf{u}} \cdot (\nabla \cdot \vec{\mathbf{u}})\delta_{ij} \end{pmatrix} + \mu_\chi \begin{pmatrix} \nabla\rho \\ u\nabla\rho \\ v\nabla\rho \\ w\nabla\rho \\ \frac{1}{2}(\vec{\mathbf{u}} \cdot \vec{\mathbf{u}})\nabla\rho \end{pmatrix}$$

Slope Limiter

[1]Kuzmin, Journal of Computational and Applied Mathematics (2010)
[2]Nishikawa, AIAA-2022-1374 (2022)
[3]Ching et al., Journal of Computational Physics (2019)

- Rewrite polynomial using Taylor series basis function[1] $\rho(\mathbf{x}) = \overline{\rho} + \alpha^{\rho}\nabla\rho\Big|_{c} \cdot (\mathbf{x} - \mathbf{x_c})$
  - Requires linear basis – project from tri-linear
  - Essential to have accurate shock detection
    - Local application known to affect steady-state convergence
- Limiter constrain slopes by vertex neighbors' cell centered values.
  - Smooth limiter function in attempt to aid steady state convergence[2]

$$\alpha_v^{\rho} = \begin{cases} 1, & \text{if } \rho_v - \rho_c = 0, \\ \psi(y) & \text{if } |\rho_v - \rho_c| > 0, \end{cases} \qquad y = \max\left(\frac{\rho_c^{\min} - \rho_c}{\rho_v - \rho_c}, \frac{\rho_c^{\max} - \rho_c}{\rho_v - \rho_c}\right), \qquad \psi(y) = \begin{cases} 1, & \text{if } y > 2, \\ \frac{y^3+4y}{y^3+y^2+4} & \text{otherwise.} \end{cases}$$

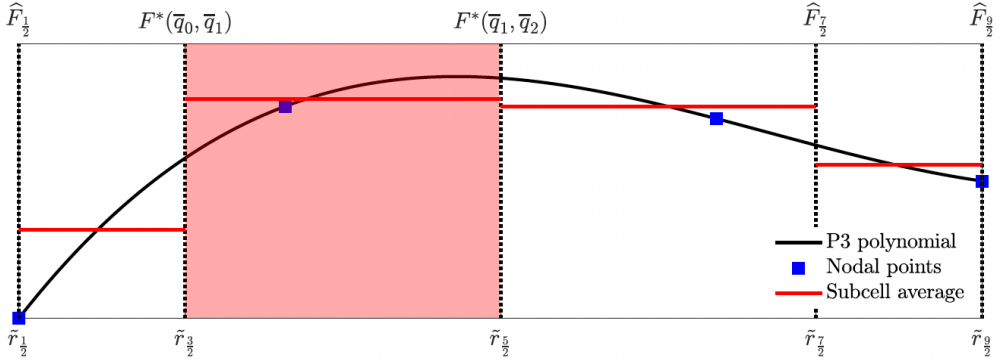
- Detection based on intra-element variation (usually of pressure)[3]

$$S = \sqrt{\frac{\int_{\Omega_e} (p/\overline{p} - 1)^2 d\mathcal{V}}{\mathcal{V}}}.$$

Subcell Finite-Volume Method

Vilar, Journal of Computational Physics (2019)

- Divide cell into $(P+1)^d$ subcells: Relate nodal points with subcell average
- Subcell flux is calculated as:
  - Reconstructed DG flux: $\hat{F}$
  - 1st order FV using subcell averages: $F^*(\bar{q}_0, \bar{q}_1)$
    - Substitute on "tagged" subcell faces
  - Tagging: discrete maximum principle, modified Ducros sensor, intraelement variation



Positivity Preserver

Zhang and Shu, Journal of Computational Physics (2010)

- "Last resort" slope limiter used in conjunction with AV, subcell
- Ensure minimum nodal density/pressure stay above a set floor (ϵ_{lim} = 10^{-8})
  - Preserved cell average of density, pressure during modification

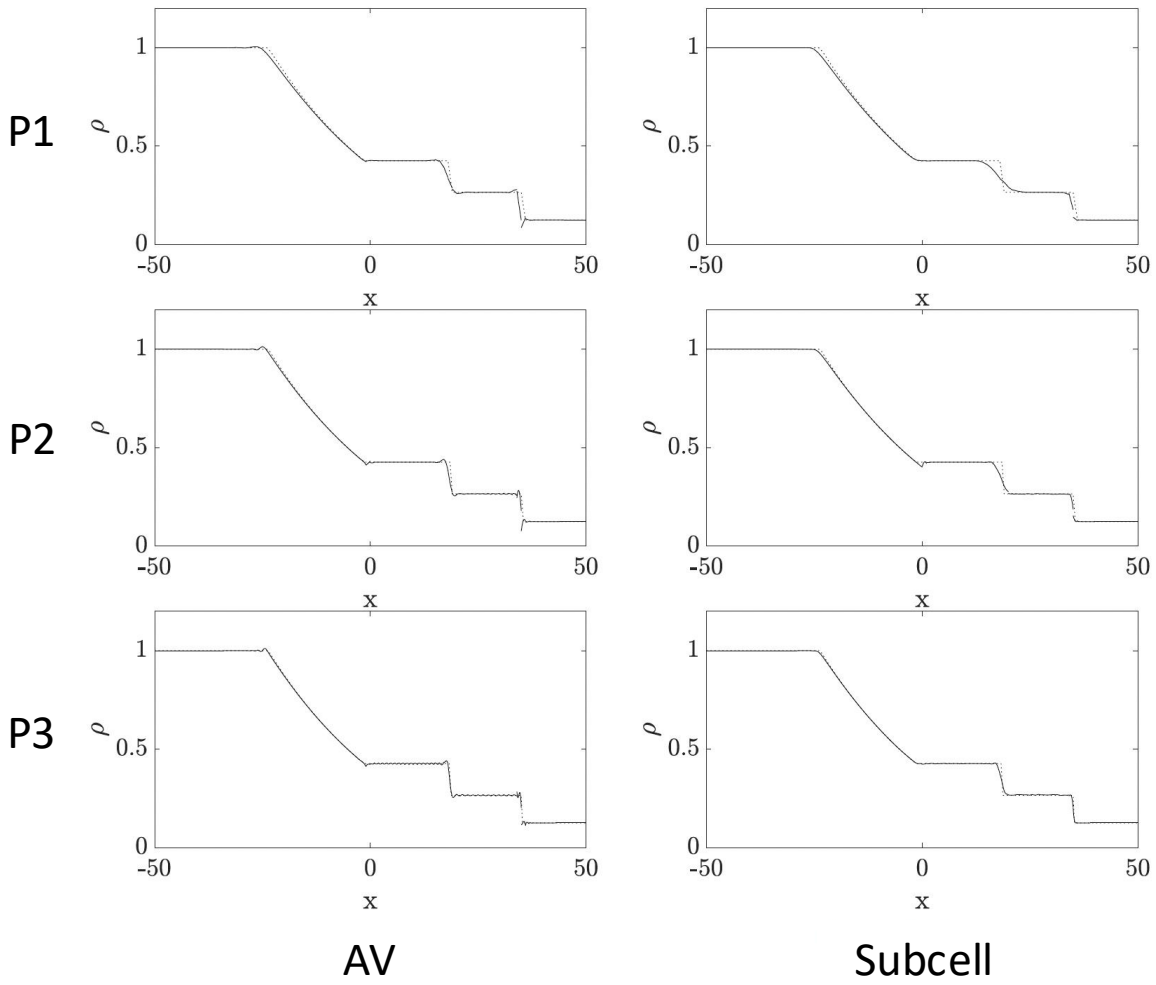
1. If $\rho_{min} < \epsilon_{lim}$, limit density: $\hat{\rho} = \theta_1(\rho - \bar{\rho}) + \bar{\rho}$
2. Recalculate pressure with modified state vector $\hat{q} = [\hat{\rho}, \rho\vec{u}, E]$
3. If $p_{min} < \epsilon_{lim}$, limit pressure: $p\left(\hat{\bar{q}}[\theta_2]\right) = \epsilon_{lim}$, where $\hat{\bar{q}} = \theta_2\left(\hat{q} - \bar{\hat{q}}\right) + \bar{\hat{q}}$
   1. Solve for θ_2 using Newton's method

$$\theta_1 = \min(1, \frac{\bar{\rho} - \epsilon_{lim}}{\bar{\rho} - \rho_{min}})$$

Discontinuous Galerkin
Results

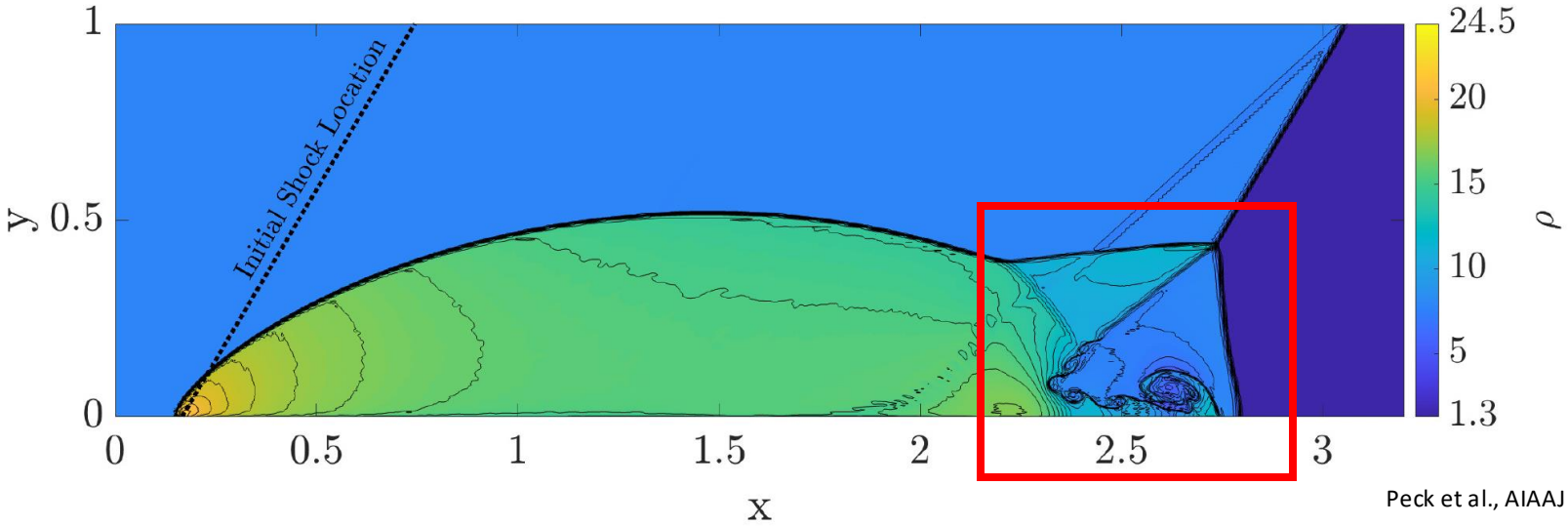
Sod Results

- Initial conditions
  - 8:1 density jump
  - 10:1 pressure jump
- Roe Flux
- 100 elements
- AV: $C_\chi = 0.01, C_\beta = 0.25$
  - No tuning of coefficients
- Subcell: DMP (density)



Double Mach Reflection

- Mach 10 shock inclined w.r.t slip surface (shock moves left to right)
  - Exact pre/post shock conditions prescribed at top boundary and inlet
- Goal: maintain stability while resolving vortical structures

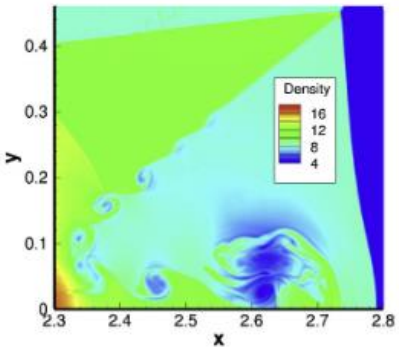


Peck et al., AIAAJ (2024)

Grid refinement

- Order fixed: P2
- Coarse: h = 1/100
- Fine: h = 1/240

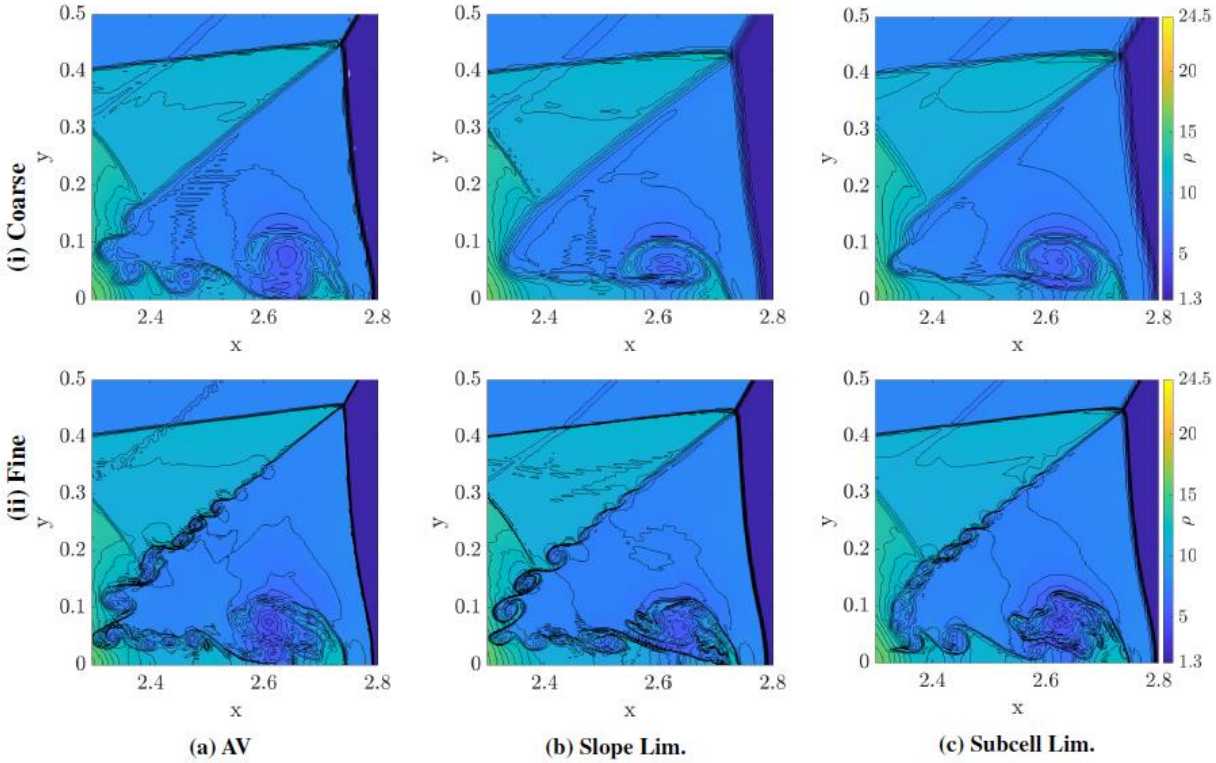
P2, Fine (Reference)



(b) Zoom-in on shear layer.

Ching et al., Journal of Computational Physics (2019)

Peck et al., AIAAJ (2024)

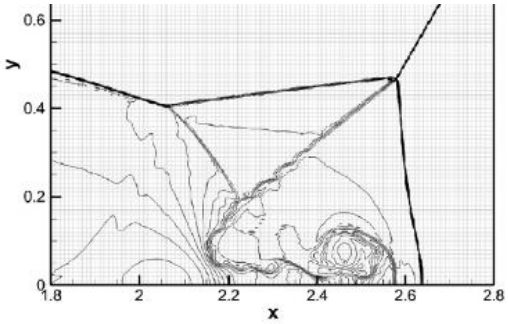


(a) AV (b) Slope Lim. (c) Subcell Lim.

Polynomial Refinement

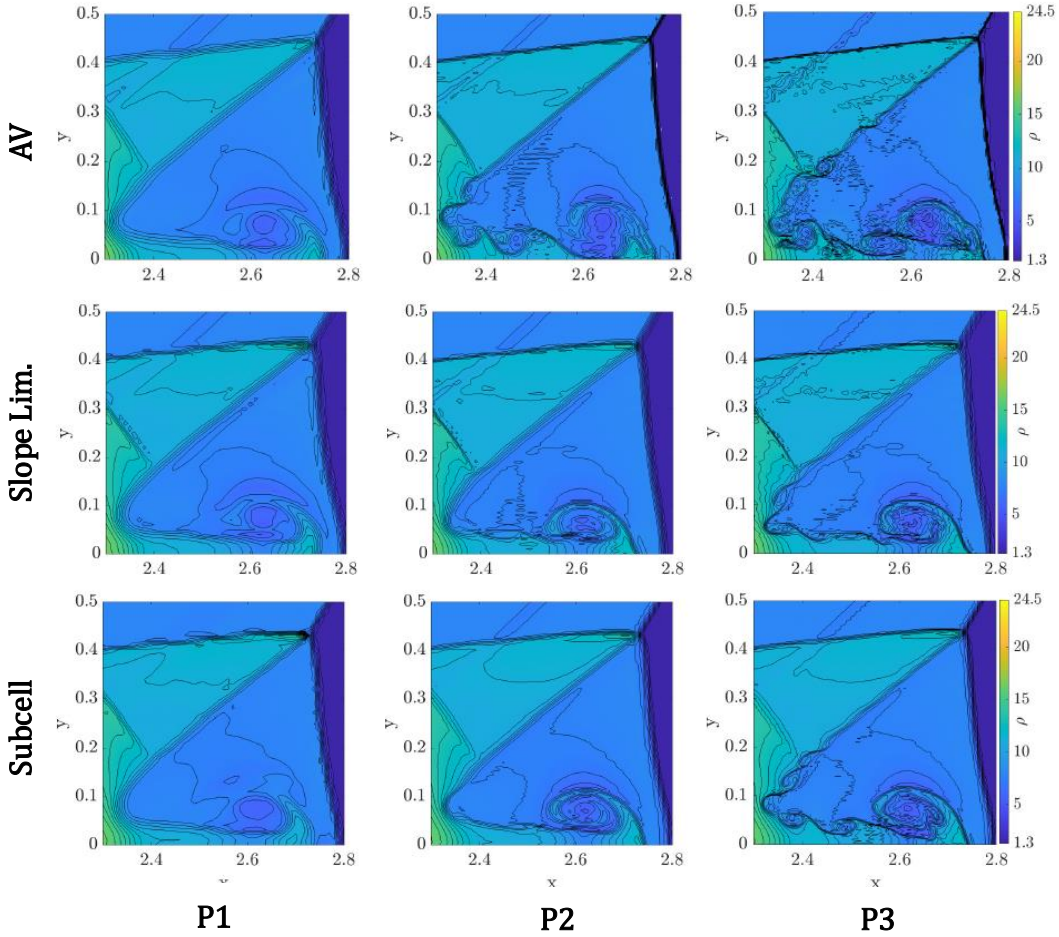
- Mesh fixed, h = 1/100

P2, Coarse (Reference)



Dumbser et al., Journal of Computational Physics (2014)

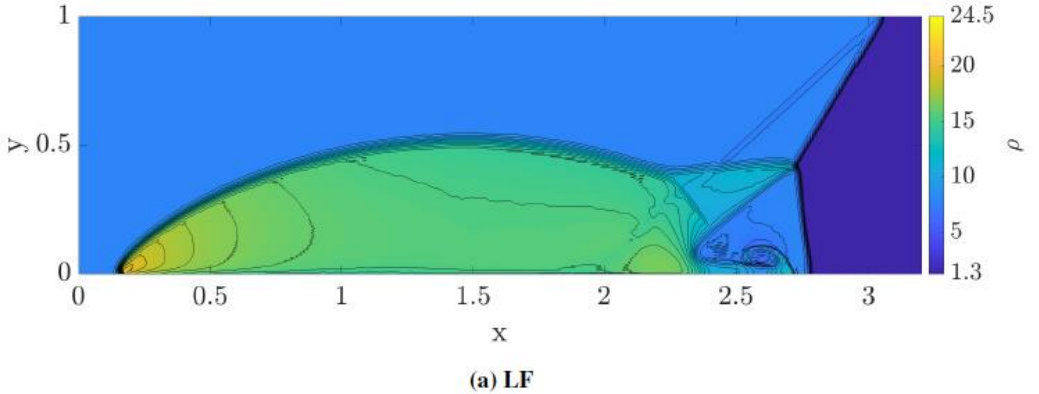
Peck et al., AIAAJ (2024)



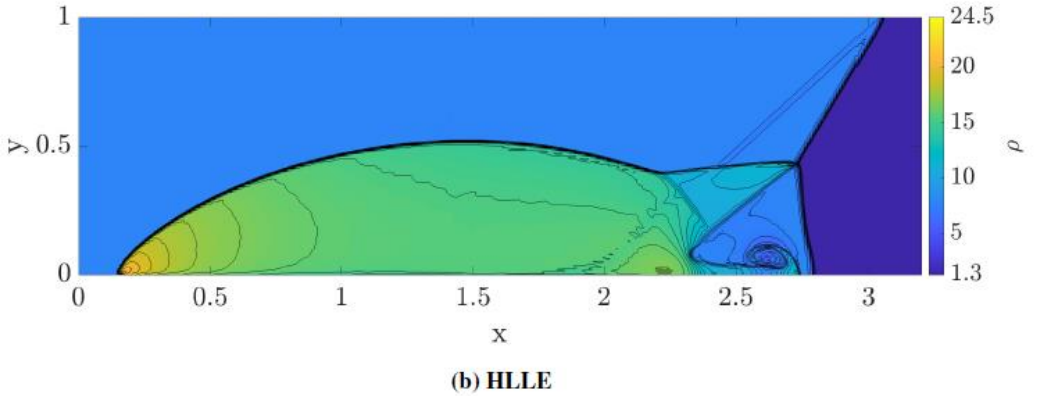
Effect of flux function on subcell limiter

Peck et al., AIAAJ (2024)

- Subcell limiter is sensitive to flux function
  - 1st-order finite volume flux formulation too dissipative with LF
  - Higher-order FV flux formulation is possible



(a) LF



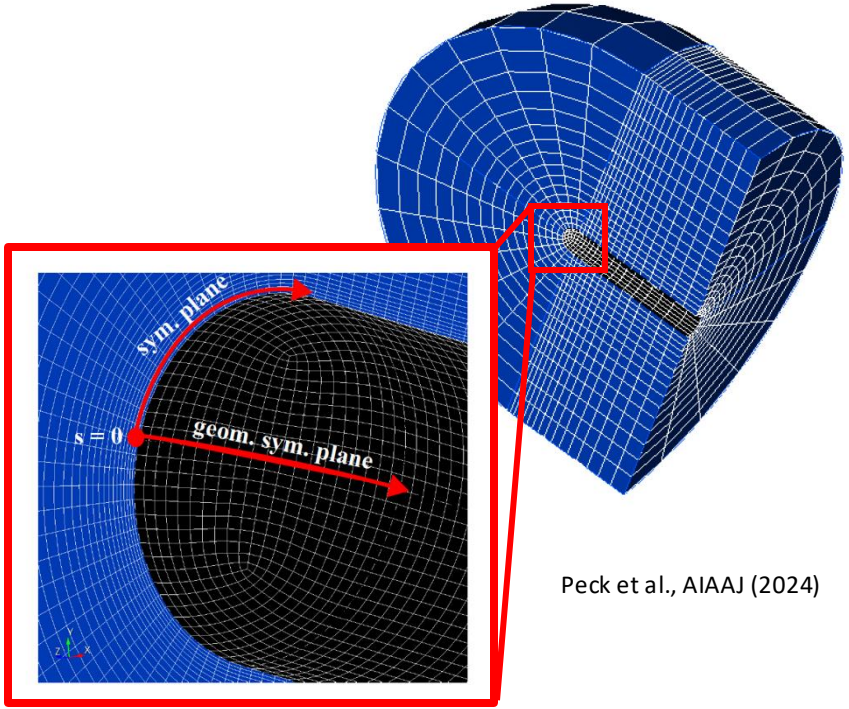
(b) HLLE

Inclined Blunt Body

- Cylinder inclined 20 deg
- Mach 5 freestream
  - T, p, U enforced at freestream boundaries
  - Initialize from freestream conditions everywhere
- Grids
  - Coarse: 150K, Fine: 450K
  - Single symmetry plane
  - Wall-normal grid clustering
  - No shock alignment

Freestream Conditions

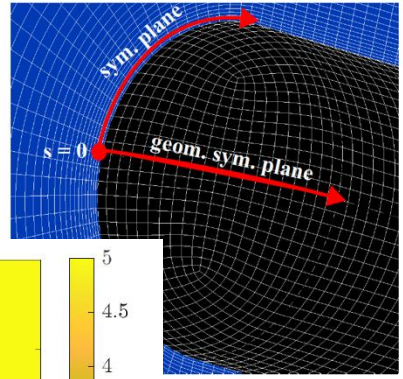
M_∞	T_∞ [K]	p_∞ [N/m^2]	U_∞ [m/s]
5.0	297.428	77.85e3	1730

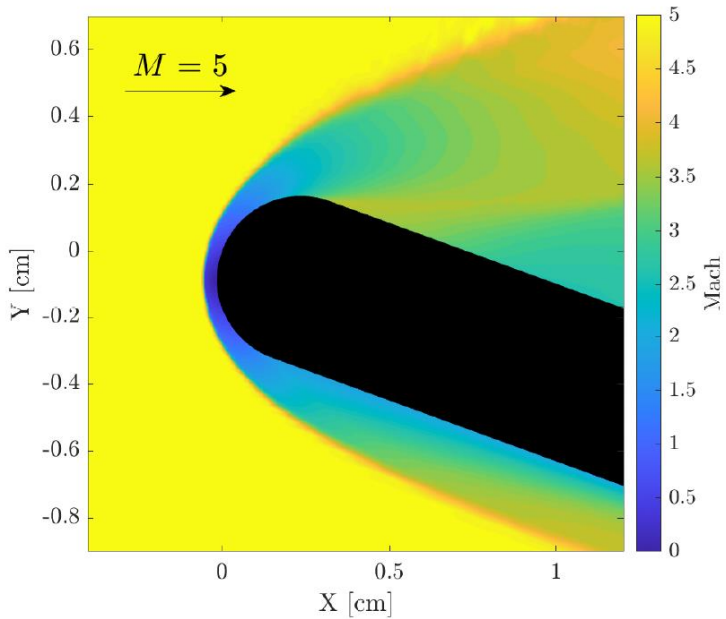


Peck et al., AIAAJ (2024)

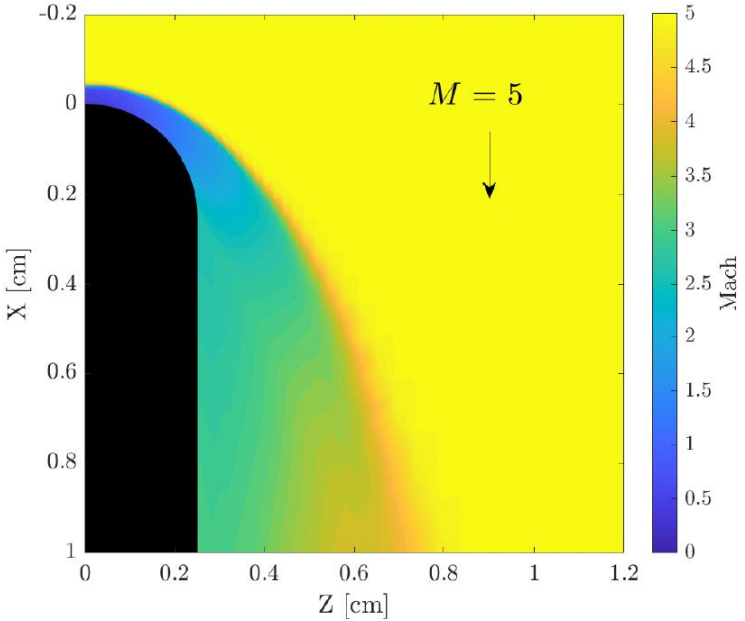
Inclined Blunt Body

Peck et al., AIAAJ (2024)





Sym. Plane



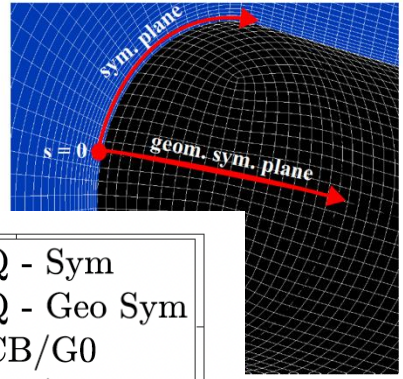
Geom. Sym. Plane

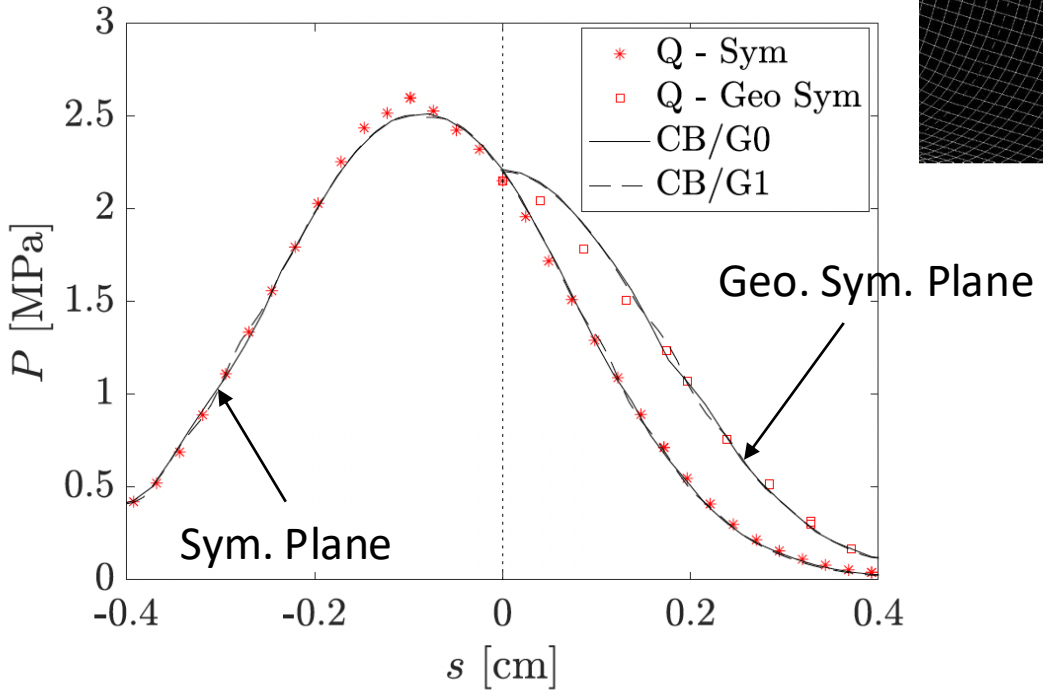
Inclined Blunt Body

- Quinoa[1] (Q)
  - P1 Galerkin FEM method
  - HLL Flux
  - 697K tetrahedral mesh
  - Van Leer's TVD limiter
- Cartablanca (CB)
  - P1 DGFEM
  - HLLE Flux
  - G0: 150K, G1: 450 K (hex)
  - **Subcell** limiter

[1]Pandare et al., Journal of Computational Physics 2023

Peck et al., AIAAJ (2024)

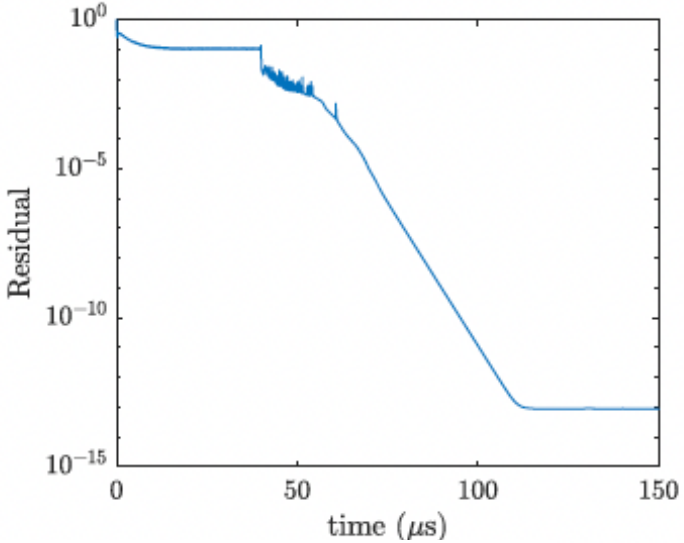


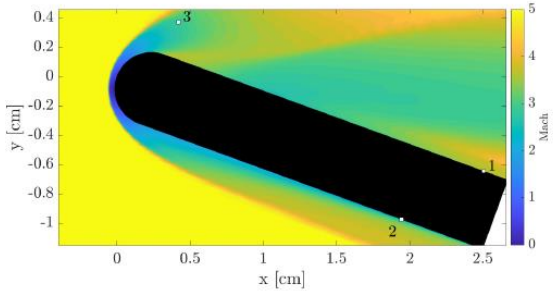


Steady State Results

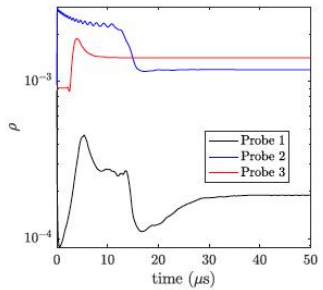
- Density residual monitors steady state
- Subcell limiter required frozen tags
  - Positivity preserver causes flickering
  - Accumulated cells to remove oscillations
  - Further work into better tagging methods needed.
- Slope limiter does not converge
  - Expected due to tag locality

Peck et al., AIAAJ (2024)





(a) Probe locations.

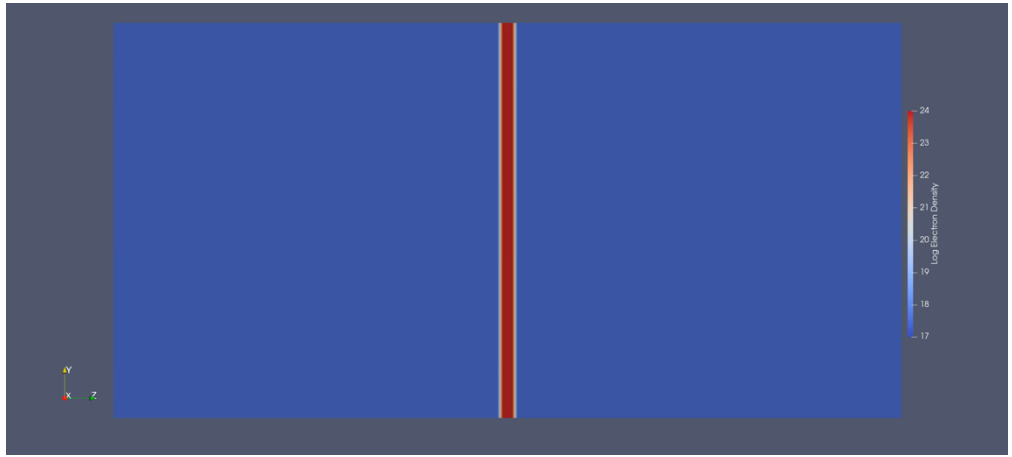


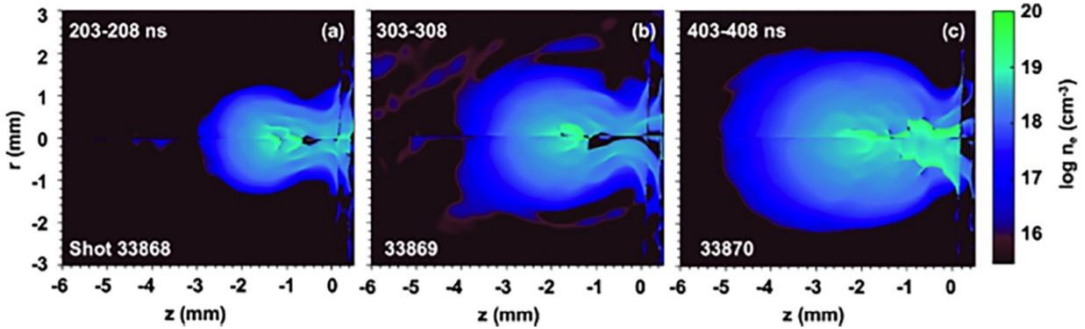
(b) Density.

DARHT (DG)

- Treat solid as dense ideal gas in equilibrium with the near-vacuum gas.
- Heating induces rapid expansion. Density, pressure, velocity have little noise due to shock capturing and flux model.
- Results obtain propagation speed similar to the experiment.
  - Used P1, slope limiter

Credit: Kyle Perez

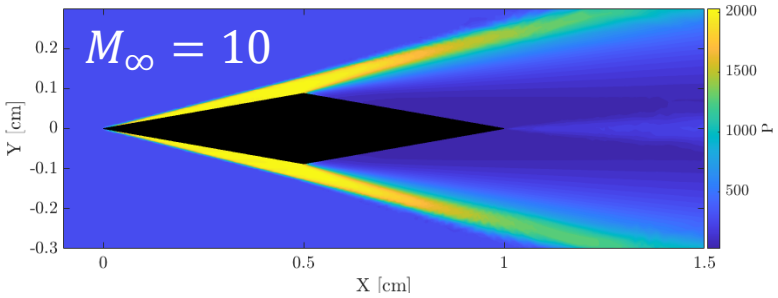




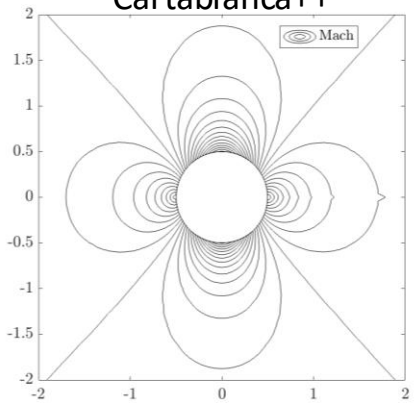
Coleman et al., Journal of Applied Physics (2022)

Additional benchmarks

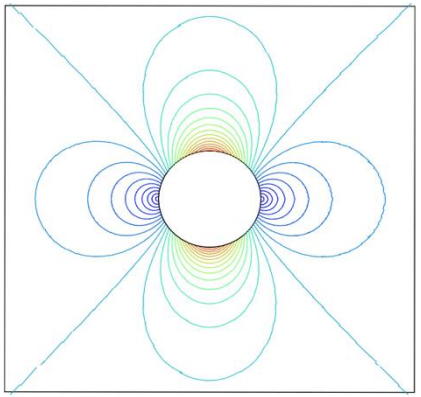
- Still considering additional test cases to improved tagging mechanisms
  - Range of conditions to verify algorithm robustness



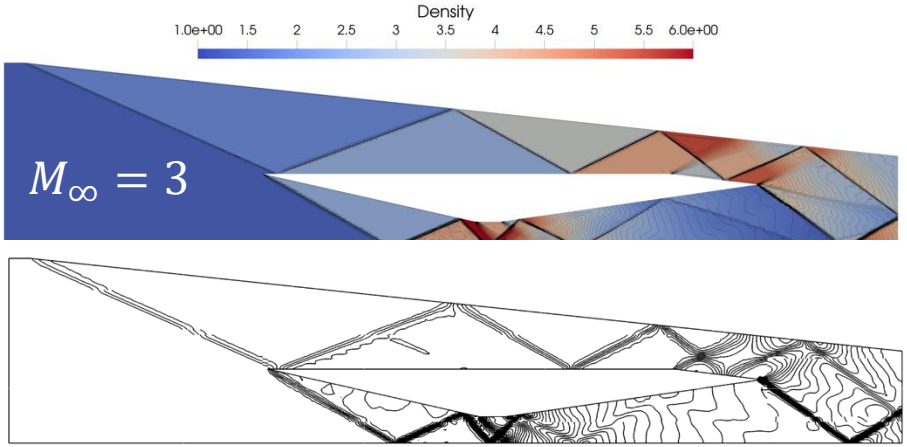
Cartablanca++



$M_\infty = 0.38$



Krivodonova & Berger,
Journal of Computational Physics (2006)



Hesthaven and Mönkeberg, Journal of Computational Physics: X (2021)

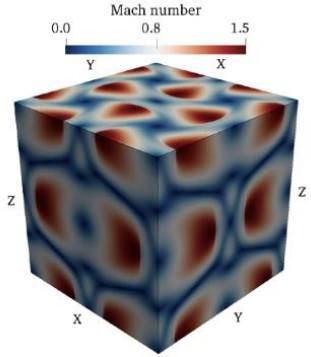
Navier—Stokes and Turbulence

[1]Bassi and Rebay, Journal of Computational Physics (1997)
[2]Vreman, Physics of Fluids (2004)

- Viscous terms implemented in Cartablanca++
  - BR1[1] scheme – sufficient for advection dominated problems
  - Extending shock-capturing algorithms
    - Some shock-detection schemes eliminated
    - High element aspect ratios
- Turbulence
  - Vreman SGS model[2] implemented
  - Can also run without LES model → “implicit LES”

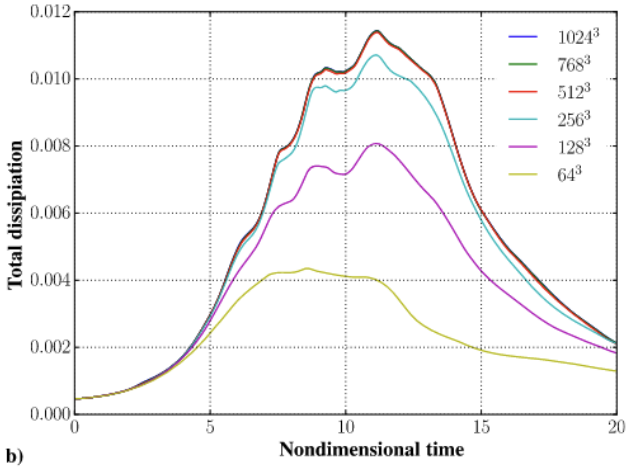
Supersonic Taylor Green Vortex

Luscher & Sandham, AIAAJ (2021)



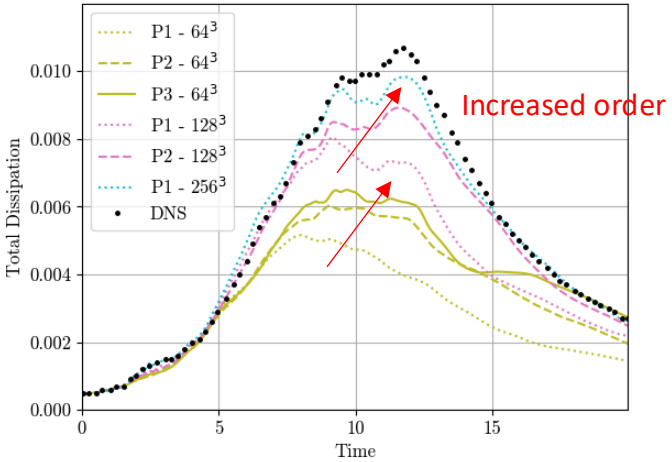
- Good qualitative comparison on grid convergence
  - Hold DOF constant as polynomial order (P) increases

Mach 1 – TENO6



Luscher & Sandham, AIAAJ (2021)

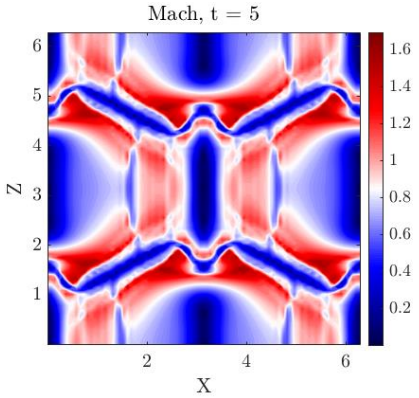
Mach 1.25 - DGFEM



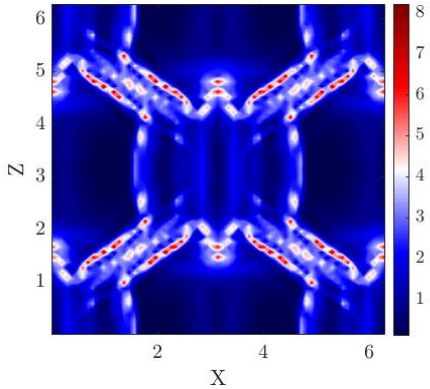
Cartablanca++

Supersonic Taylor Green Vortex

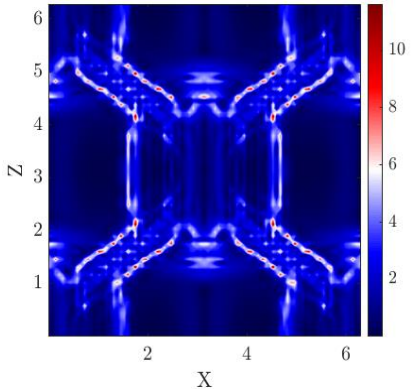
- Using subcell limiter
- Improvements in shock detection
  - Pressure variations on subcell scale
- Testing numerics
  - *Effect of flux function*, shock detection procedure



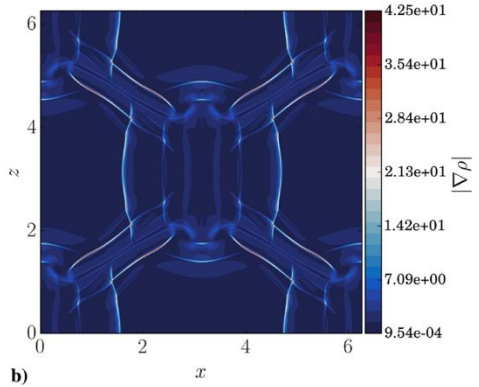
Magnitude of density gradient, t = 5



Cartablanca++, P1, DOF = 128^3



Cartablanca++, P2, DOF = 128^3



TENO6 DOF = 512^3

Luscher & Sandham, AIAAJ (2021)

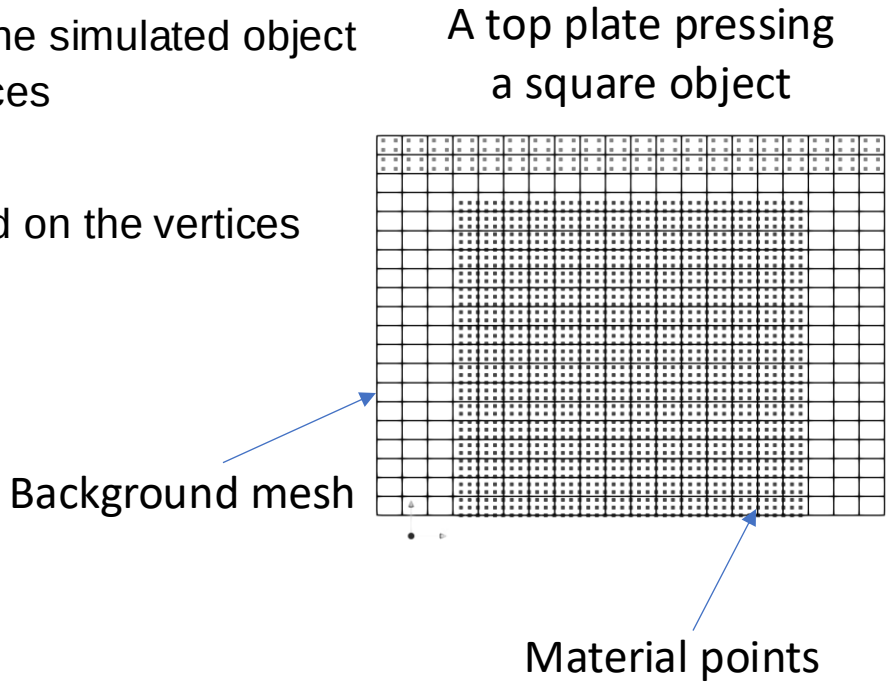
DG Summary

- High-order DG solver implemented in Cartablanca++
  - Robust shock-capturing shown up to Mach 10
  - High-order benefits retained in flows with strong shocks
  - Subcell limiter and slope limiter perform well at higher order
- Range of simulation capability
  - Hypersonic inviscid
  - Supersonic turbulence
  - Solid mechanics modeling
- Future developments
  - Implicit time stepping
  - h/p adaptation
  - GPU implementation

Material Point Method (MPM)

Material Point Method (MPM)

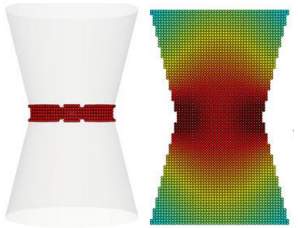
- Method is a hybrid Eulerian/particle method
- Material points
  - Lagrangian (material) points represent the simulated object
  - Serve as integration points for grid vertices
- Background mesh
  - Eulerian continuum equations are solved on the vertices



Why use MPM?

- Static mesh → no mesh tangling
- Mesh can be unstructured
- Material points (particles) represent material motion
- Material points can carry history dependent quantities

Brittle Failure



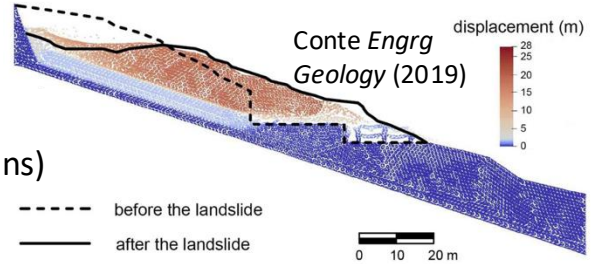
Barclay *Comp Part Mech* (2023)

Plasticity and Ductile Failure



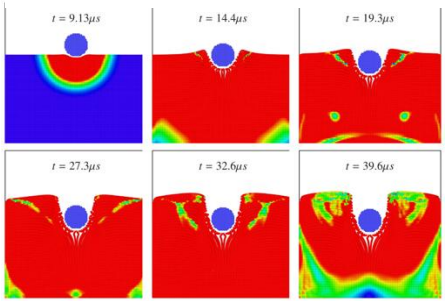
Long *Comput. Methods Appl. Mech. Engrg* (2016)

Landslides
(large deformations)



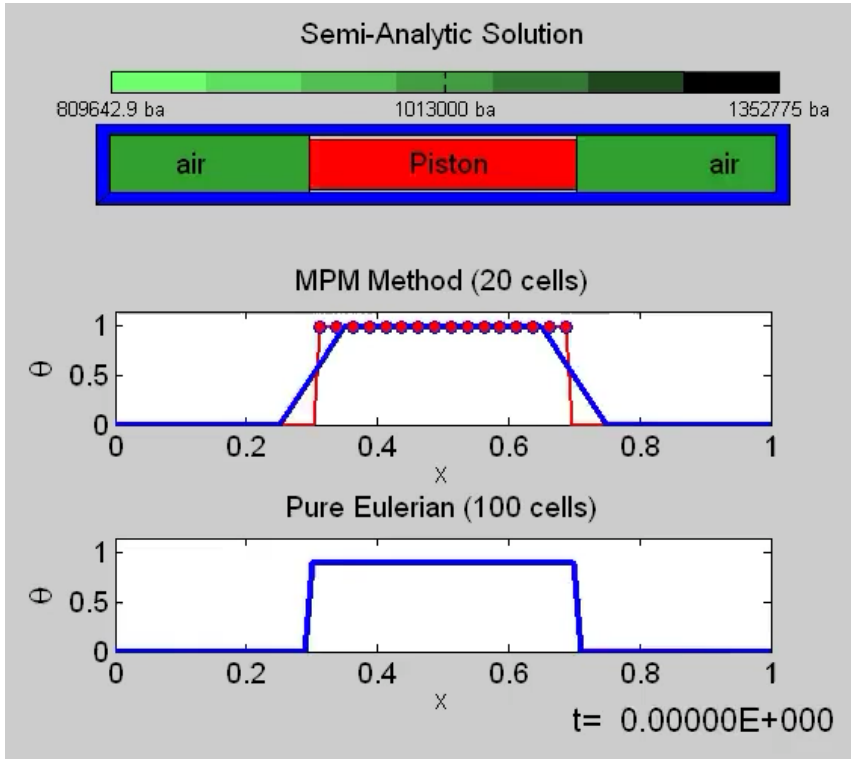
Conte *Engrg Geology* (2019)

Contact and Impact

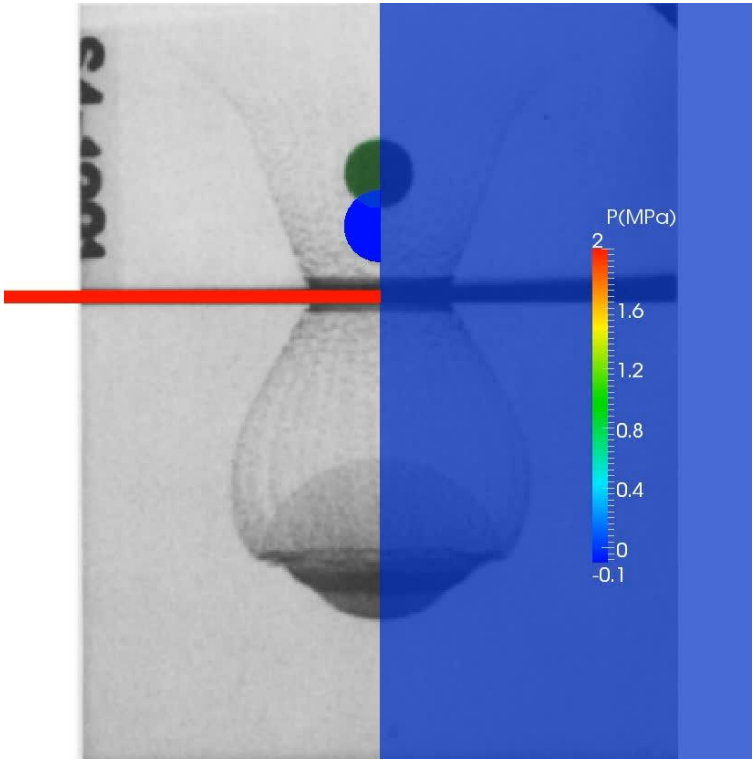


de Vaucorbeil *Comput. Methods Appl. Mech. Engrg* (2021)

Material interfaces



Credit: Duan Zhong Zhang

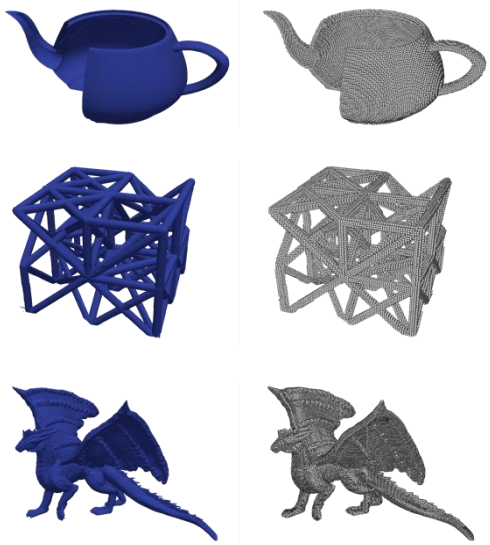


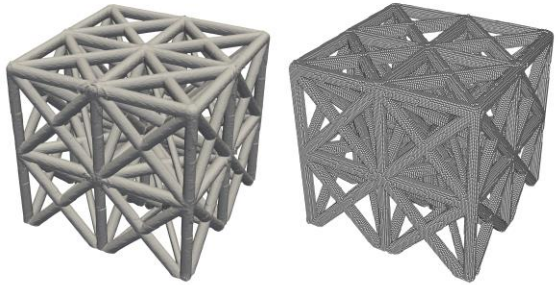
Aluminum ball hitting steel plate

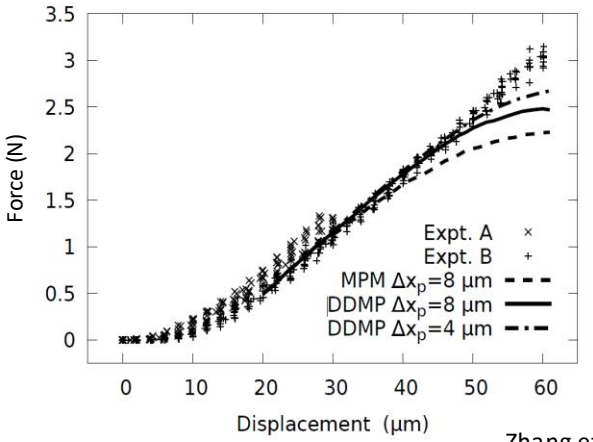
MPM Results

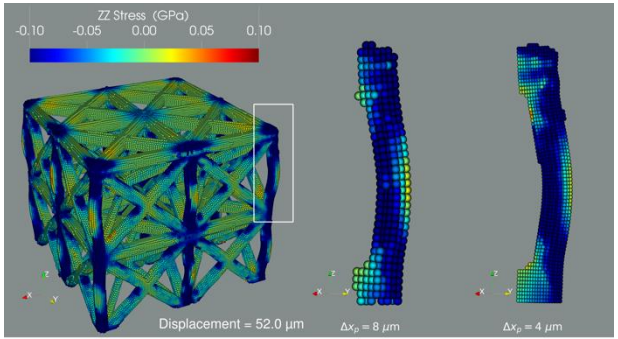
Credit: Paul Barclay

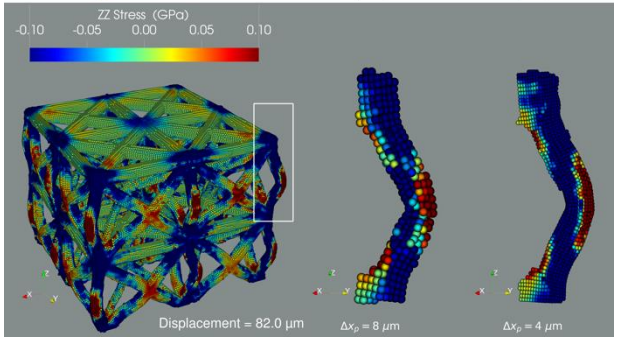
- Large deformation
- Complex shapes
  - STL file → matl. points







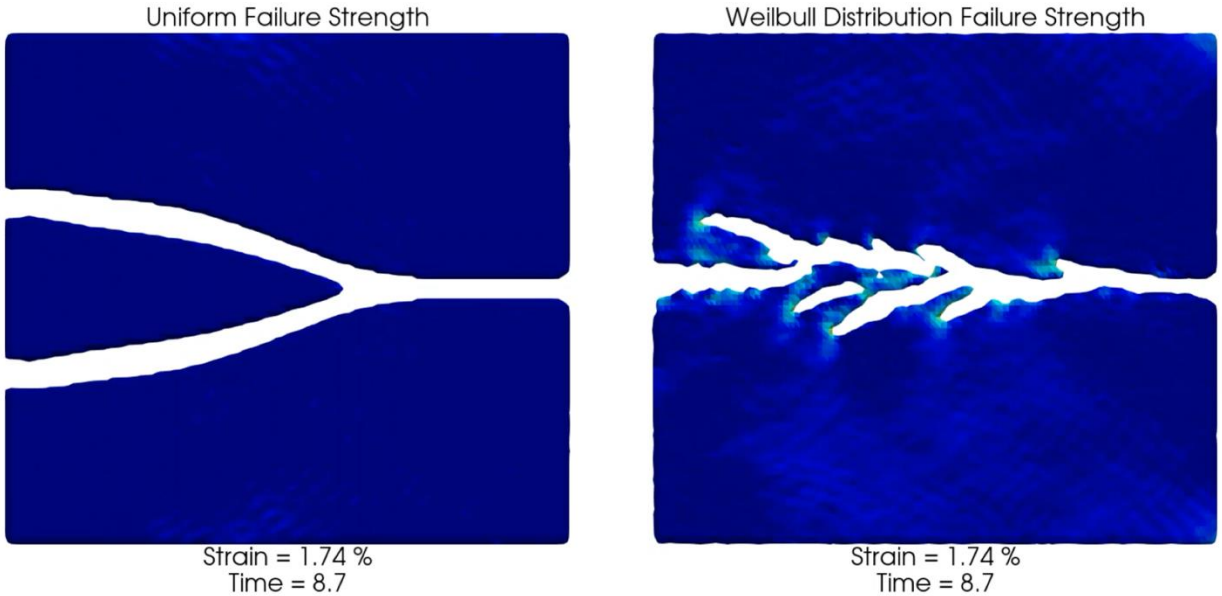


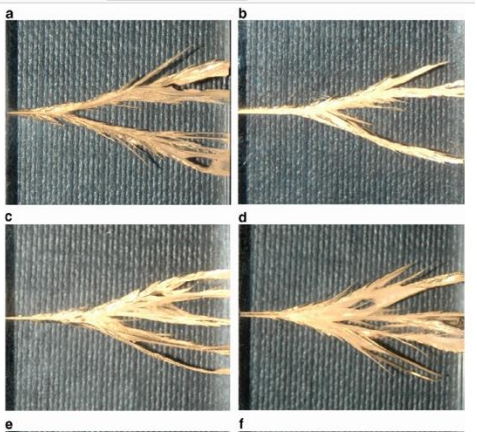


Zhang et al., Computational Particle Mechanics (2024)

MPM Results

- Crack propagation in brittle, ductile materials
  - Comparing uniform and non-uniform material



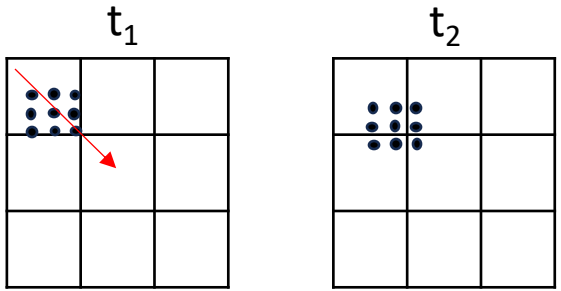


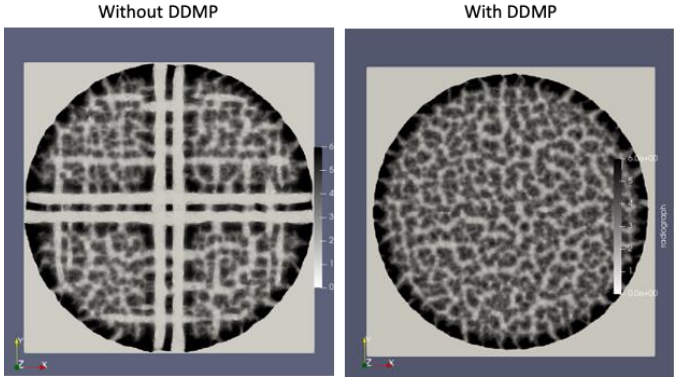
Murphy
Eng Fract Mech (2006)

Credit: Paul Barclay

Dual Domain Material Point (DDMP)

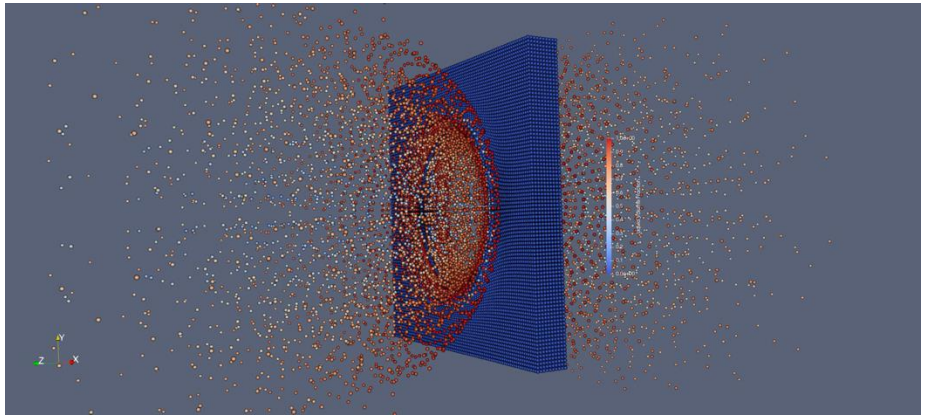
- Pure MPM has a cell-crossing problem
  - When particles cross a cell boundary, they generate numerical noise
  - Particularly an issue for simulating brittle materials.
- Many solutions proposed in the literature. We use DDMP.
  - Modifies the underlying shape function definition so that derivatives are continuously defined across cells
  - Fixes mesh imprinting in failure simulations





DARHT (MPM)

- Aluminum target, tons of particles, and plenty of space to expand.
- Heating induces a phase change and rapid expansion of aluminum fluid.
- MPM fails with large expansion
  - high density ratio → empty cells with no particles
  - Number of particles required is computationally infeasible.
  - Solution is to pass MPM results to DG.



Credit: Kyle Perez

MPM Summary

- MPM implemented in Cartablanca++
  - Material points provide sharp resolution of interfaces between materials
    - E.g., solid and a void, two materials colliding
  - Excellent description of materials undergoing large deformations/failures
- Large density gradients are expensive to compute
  - Particles represent the material
  - Need large number of particles to avoid empty mesh cells
  - In this case, ideal to hand off the calculation to another solver

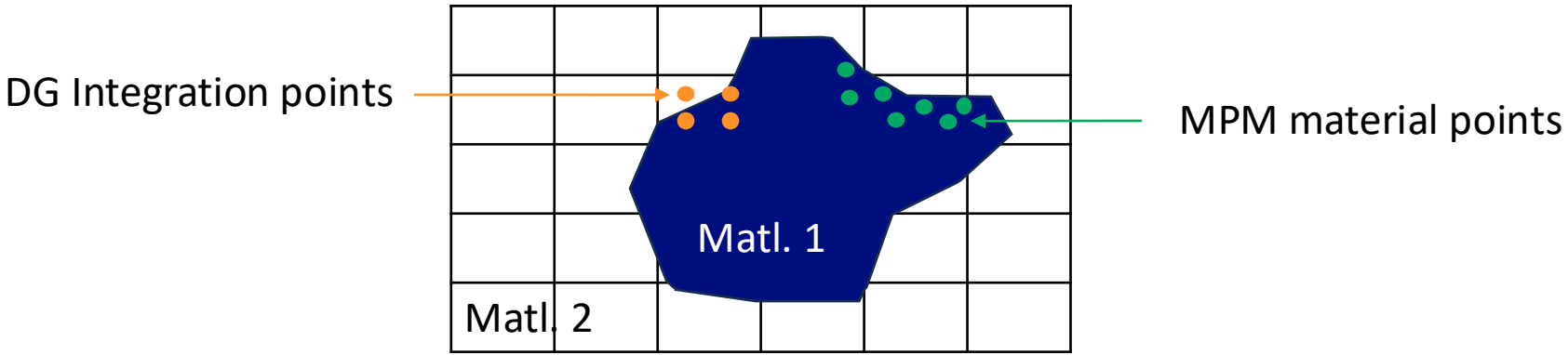
Interaction – Multi-velocity Formulation

Multi-velocity approach

- Reinterpret conservations by considering what $(\vec{x}, t)$ represents
  - It is a single spacetime point
    - $(\vec{x}, t)$ can only have one material
    - Interfaces require special methods to track
  - It is *a vicinity around* a spacetime point
    - $(\vec{x}, t)$ could have many materials → multi-velocity concept becomes meaningful
    - Solve additional mass conservation equation to access “volume fraction” (VOF)

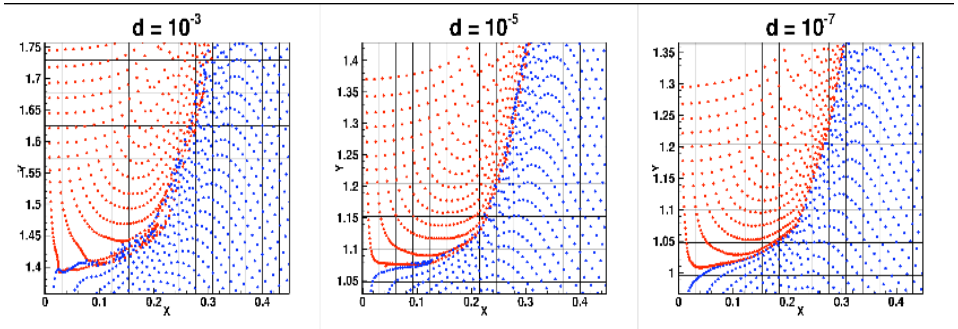
Multi-velocity approach

- Each material is assigned a solution method at initialization
  - E.g., material 1 is a solid and assigned MPM, material 2 is air and assigned DGFEM
- Conservation equations give each material VOF
  - If a material VOF > 0, the assigned solution method is active
  - Implicitly tracks interface
- Materials interact through momentum, energy exchange



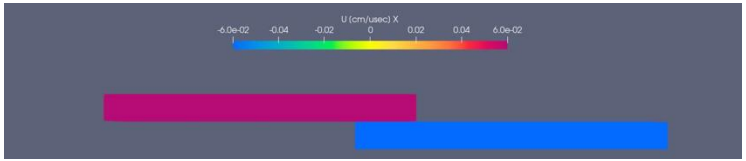
Material interaction

- Added “drag” force in momentum equations to model friction between materials
  - Drag is proportional to the velocity difference between materials
  - Analog in the energy equation to account for heat transfer

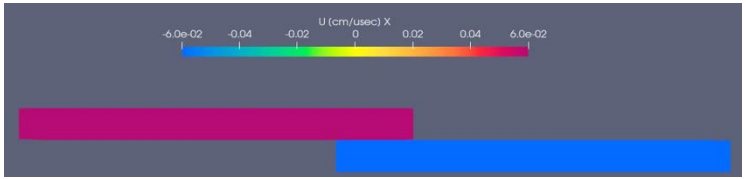


Credit: Duan Zhong Zhang

No friction



High friction



Credit: Jiajia Waters

Summary

Highlights of Cartablanca++

- Cartablanca++ contains both DGFEM and MPM solvers
  - DGFEM provides high-order capabilities for fluid dynamics
    - Different shock-capturing algorithms compared
      - Subcell and slope limiter perform well at high-order: robust while still resolving fine features
    - Continuing to improve DG robustness by extending to more benchmark problems
  - MPM provides solid mechanics capabilities
    - Sharp interface tracking between materials
    - Handles large deformations and fracture/failure of materials
- Multi-velocity formulations recently implemented in both solvers
  - Interaction framework (drag force, heat transfer) included
- Future work: simultaneous DG/MPM multi-velocity
  - Enables material interactions with different solvers assigned to materials

Thanks for listening! Any Questions?