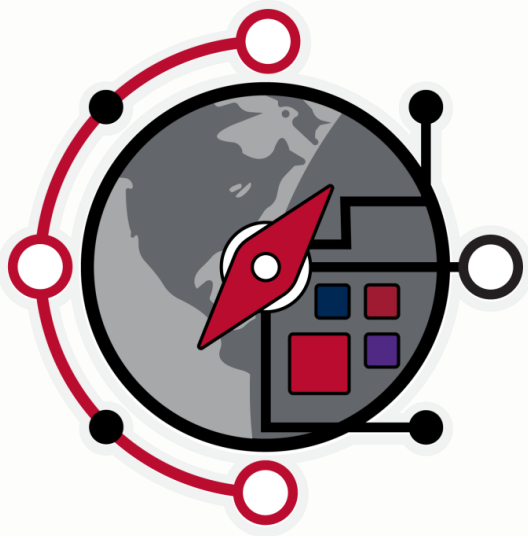




Center for
Optimized Modern
Parallel Adaptive
System Software

Adaptive Interface Models for Massively Parallel Hydrodynamics

STEVE SHKOLLER · UNIVERSITY OF CALIFORNIA, DAVIS

JASON STEWART · UNIVERSITY OF NEW MEXICO

INTEGRATED PROJECT: HYDRODYNAMIC SOLVER METHODS

COMPASS ANNUAL REVIEW · OCTOBER 29, 2026

Year 1: adaptive meshing delivered

DELIVERED

Z-model adaptive meshing

Roadmap: Year 1

3D z-model with adaptive meshing delivered to UNM for parallelization

STARTED EARLY

Model topological transitions

Roadmap: Year 2

Merger and pinch-off working in 3D; used routinely in 2D and axisymmetric runs

FIRST STEP

Compressible interface model

Roadmap: Year 1

Variable-density plume model and a new 1D shock model, to be merged with the z-model for shock–interface interaction

UNDER WAY AT UNM

Parallelize the z-model

Roadmap: Years 2–3

C++ port with a fast multipole solver (Jason Stewart)

Mixing happens on thin interfaces that roll up

Thin interfaces control mixing

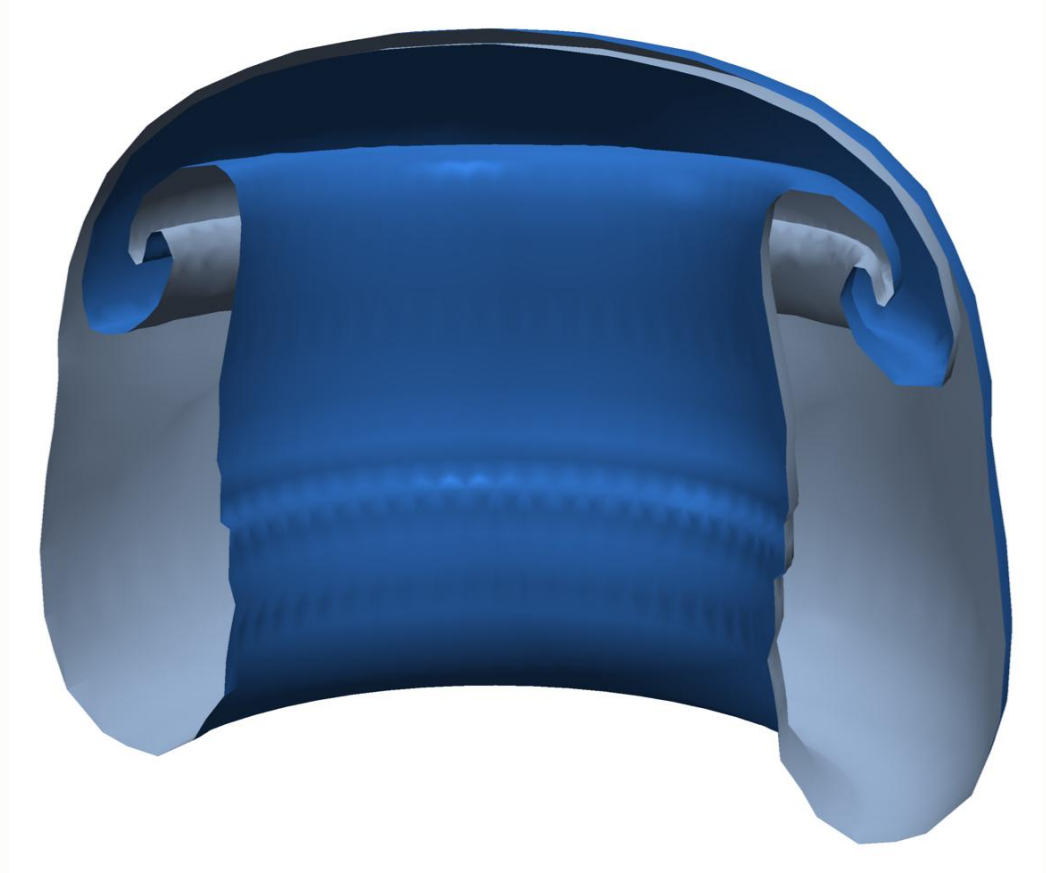
Rising plumes, Rayleigh–Taylor and Richtmyer–Meshkov mixing: the action is on a surface that folds and rolls up.

Grid codes pay for the whole volume

Resolving a thin rolling sheet needs fine cells everywhere, and numerical diffusion smears the roll-up.

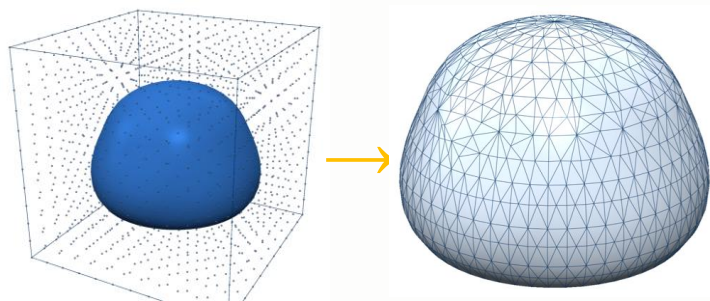
The z-model follows only the interface

It keeps the full roll-up at a tiny fraction of the cost.



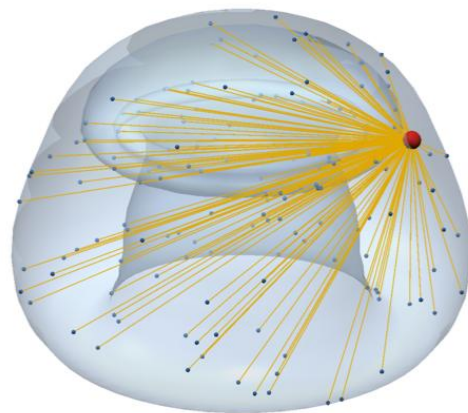
3D z-model: a rising bubble, cut open to show its rolled-up rim

Three ideas make the z-model fast



1 Follow the surface, not the volume

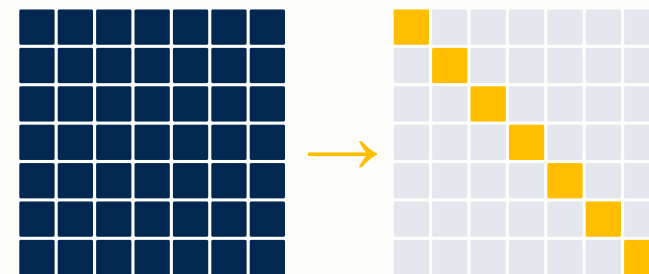
A 3D grid with n cells per side has n^3 cells; the interface needs only about n^2 points. The saving is a factor of n , and it grows as the resolution is refined.



2 Velocity by a nonlocal elliptic inversion

$$\bar{u} = \nabla \times (-\Delta)^{-1} \omega$$

The interface carries vorticity ω , and the velocity is the Biot–Savart integral of ω over the whole surface. Numerically it is an N-body sum, like gravity between stars: tree codes, fast multipole methods and GPUs apply directly.



Euler: implicit
solve every step

z-model:
explicit update

3 New asymptotics in the nonlocality

The exact Euler vortex-sheet equations contain an extra nonlinear time-derivative term, so each step is an implicit solve over the whole surface, essentially impossible in 3D. Asymptotics in the nonlocality, not in amplitude or slope, prove the term can be removed with extreme accuracy: an explicit vortex method that is easily solved.

From the Euler equations to the z-model

Velocity of the interface

$$\bar{u} = \nabla \times (-\Delta)^{-1} \omega$$

$$\bar{u}(s) = \frac{1}{4\pi} \iint \omega(s') \times \frac{z(s) - z(s')}{(\varepsilon^2 + |z(s) - z(s')|^2)^{3/2}} ds', \quad \omega = \mu_2 \partial_1 z - \mu_1 \partial_2 z$$

regularized at length ε (a vortex blob); an N-body sum over the surface

Exact Euler equations for the vortex sheet

$$\partial_t z = \bar{u}$$

$$\partial_t (\mu_\alpha + 2A \bar{u} \cdot \partial_\alpha z) = A \partial_\alpha (|\bar{u}|^2 - \frac{1}{4} h^{\beta\gamma} \mu_\beta \mu_\gamma - 2g z^3)$$

The red term contains **the time derivative of μ inside the Biot–Savart integral**. Every step requires solving a nonlinear Fredholm integral equation over the whole surface: essentially impossible in 3D.

The z-model

$$\partial_t z = \bar{u}$$

$$\partial_t \mu_\alpha = A \partial_\alpha (|\bar{u}|^2 - \frac{1}{4} h^{\beta\gamma} \mu_\beta \mu_\gamma - 2g z^3)$$

Asymptotics in the nonlocality prove the red term can be removed with extreme accuracy: **each step is explicit, one Biot–Savart sum**.

It also admits new physics:

- a dynamic Atwood number $A(s, t)$ carried by the interface
- a baroclinic torque in the vorticity equation
- surface viscosity, to continue past the sheet singularity
- merger and pinch-off by topological surgery

z : position of the interface; μ : vortex-sheet density; A : Atwood number; g : gravity; h : surface metric.

R. Granero-Belinchón and S. Shkoller, Multiscale Model. Simul. (2017); G. Pandya and S. Shkoller, J. Fluid Mech. (2023).

Orders of magnitude less computing

3,900×

fewer unknowns

A grid with n cells per side has n^3 cells; the surface needs about n^2 points, so the saving grows with resolution. Rising bubble: 7,700 surface points versus 30 million grid cells as small as its smallest triangles.

40,000×

faster on equal hardware

Cooling-tower plume. 3D axisymmetric (so 2D) grid code: 4,000 CPUs for 160 h to simulate 1,000 s. Axisymmetric z-model: 105 s in 46 min on a laptop, already 22× faster in wall-clock time. Spread over the same 4,000 processors: about 40,000× faster (estimate).

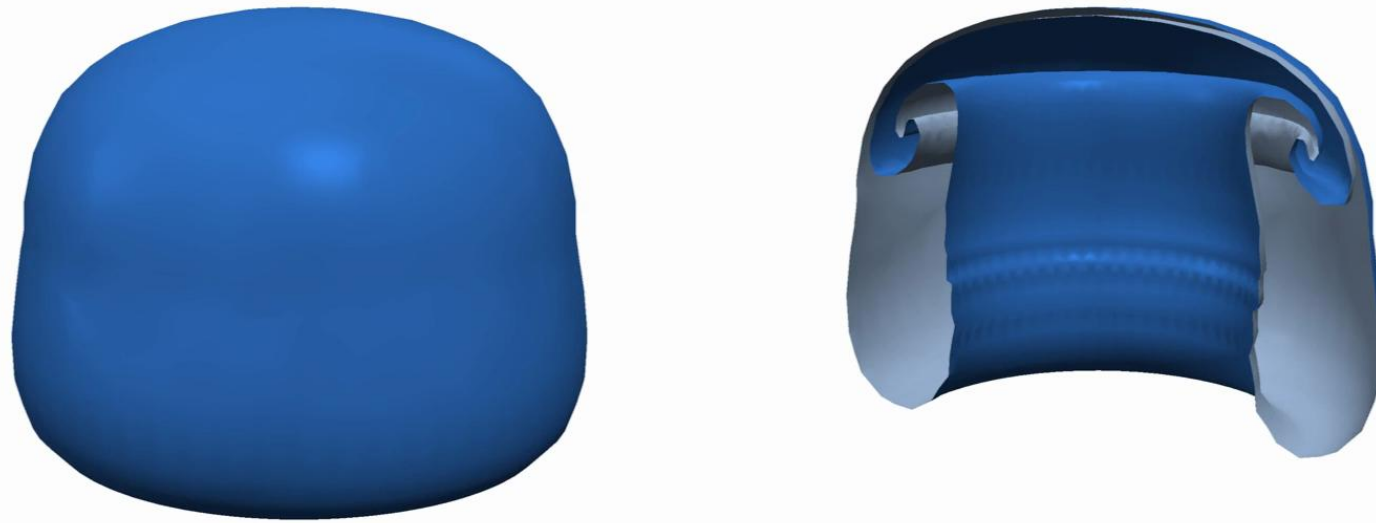
1.4 h

3D roll-up on one core

The rising bubble of the next slide, from rest to full roll-up, on a single laptop core, with no parallelization.

Laptop: Apple M3 Max, about 2 CPU cores used; all times measured. Grid count: uniform grid over the region the bubble sweeps, cells as small as the finest 1% of its triangle edges. Plume estimate: perfect parallel efficiency, each CPU counted as one laptop core, averaged over the first 105 s (about 8,000× over its last 5 s).

A rising 3D bubble rolls up into a mushroom cap



15,400 triangles

at the end of the movie, finest where the skirt rolls up

1.4 hours on one core

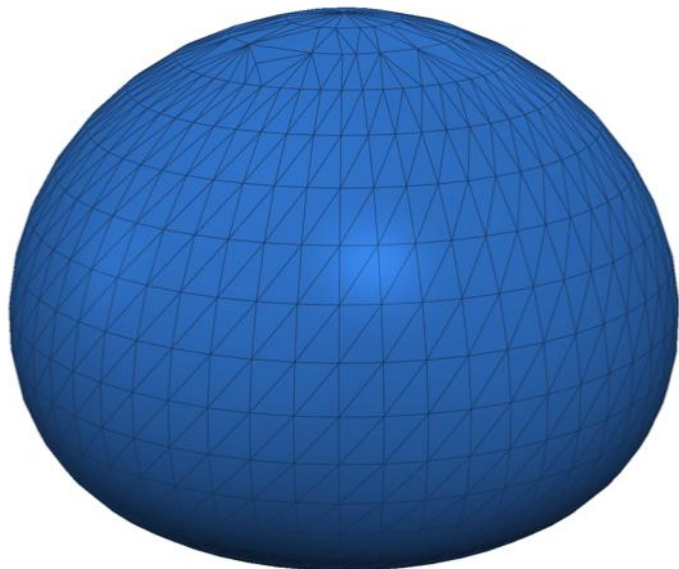
of a laptop, with no parallelization

Validated by linear theory

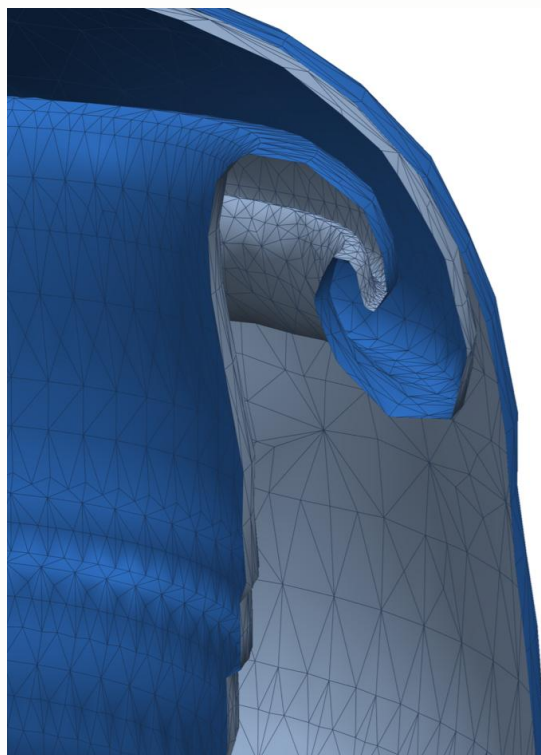
σ : growth rate of the disturbance; g : gravity; k : its wavenumber. Rayleigh–Taylor theory gives $\sigma = \sqrt{(Agk)}$, so $\sigma/\sqrt{A} = \sqrt{(gk)}$ depends on g and the bubble's shape but not on the Atwood number A . Runs from $A = 0.2$ to 0.9 (density ratios up to 19:1), same bubble: $\sigma/\sqrt{A} = 9.3$ for every A , within 7%.

Rising bubble, Atwood number 0.2: outside (left) and cut open (right)

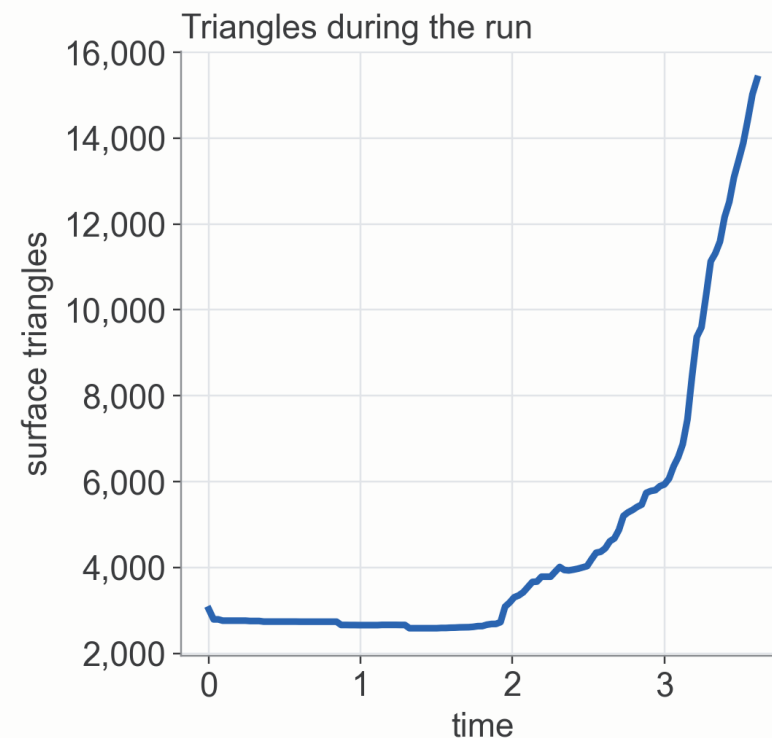
Resolution goes where the sheet rolls up



$t = 1.5$: 2,600 triangles, nearly uniform



$t = 3.6$: 15,400 triangles, finest at the spiral tip



The mesh refines where the sheet curves sharply or comes close to itself, and coarsens elsewhere. The enclosed volume is conserved to round-off.

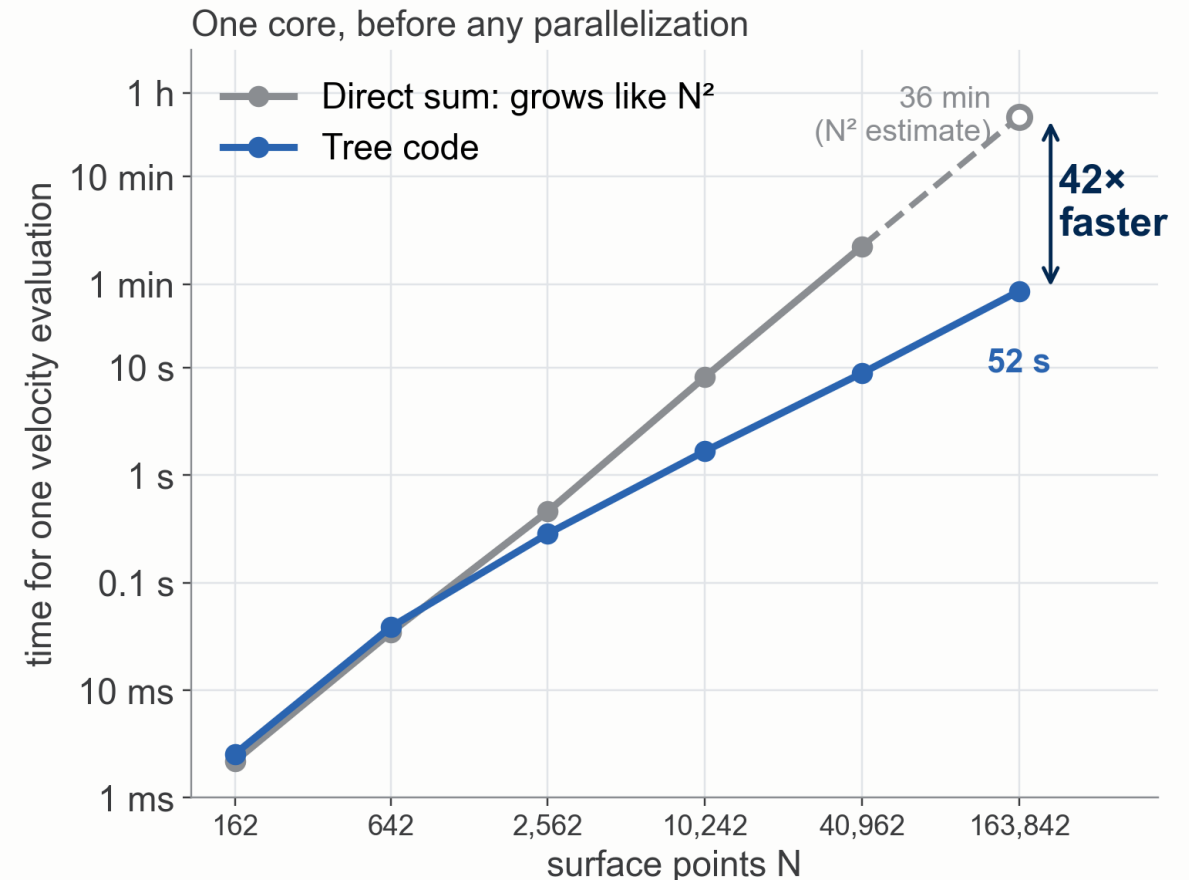
Delivered to UNM for massive parallelization

The zmodel3d-amr package

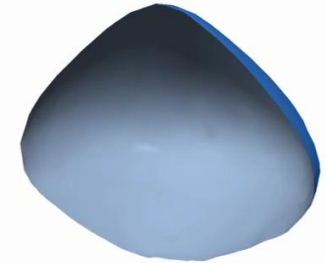
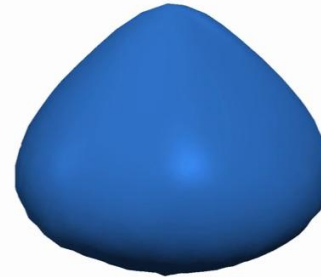
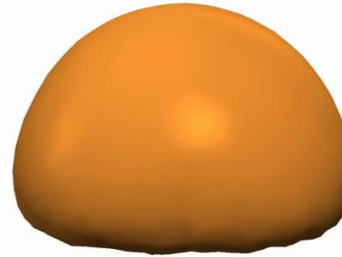
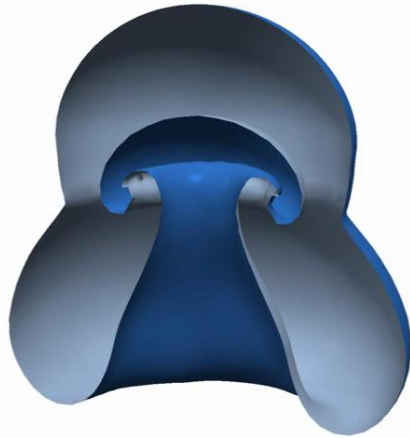
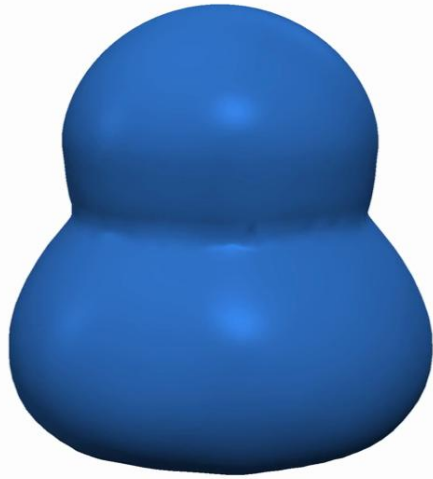
- Self-contained 3D z-model with adaptive meshing
- 8,200 lines of Python, 36 tests
- A guide to the physics, and a guide to parallelization with timings of each part of the algorithm
- Delivered to Patrick Bridges' team, July 2026

Where the time goes

Most of the time is spent computing the Biot–Savart velocity. The velocity at each surface point is a separate sum over all other points, so these sums can be divided among many processors. Once that is fast, the adaptive mesh is the largest remaining cost.



Interfaces merge and break apart



Merger: a fast bubble catches a slower one and they fuse

Pinch-off: a thin neck is cut; the halves recoil into droplets and rise

Surgery fires only when two sheets come closer than the smallest resolved scale, with safeguards against spurious events. **Merging sub-grid folds also caps the cost:** 10× cheaper, and the gap keeps growing.

Cooling-tower plume: a body of revolution

$t = 105 \text{ s}$



The problem

LANL's HIGRAD cooling-tower case: air heated by 30 K inside a 152 m tower rises into a stratified atmosphere.

The model

Axisymmetric z-model: the plume surface is a stack of vortex rings, and the tower wall and the heated band are part of the model.

Checked

Against classical results for vortex rings and interfacial waves, to within 0.5 to 5%.

Fast

105 s of plume rise in 46 minutes on a laptop: per simulated second, faster than the 3D axisymmetric (so 2D) grid code on 4,000 CPUs.

Variable density as an interface evolution

$$A = \frac{\rho^+ - \rho^-}{\rho^+ + \rho^-} \quad \frac{DA}{Dt} = \frac{N^2}{2g} \dot{x} - \lambda A, \quad \lambda = \frac{2\alpha w}{R(x)}$$

Density lives on the interface

A measures the density jump across the interface. It is now a field carried by the interface: each point has its own A , which changes as the point moves.

Stratification

The atmosphere is stably stratified ($N = 0.013/\text{s}$): rising air meets lighter surroundings and loses buoyancy.

Entrainment

Mixing with surrounding air dilutes the density jump at the Morton–Taylor–Turner rate λ (w : rise speed, R : plume radius). $\alpha = 0.1$ is the classical value from laboratory plumes, not fitted to the reference run.

First step toward compressible flow

Evolving density on the interface is the first ingredient of the compressible model, to be coupled to a shock model for Richtmyer–Meshkov.

No entrainment: a jet

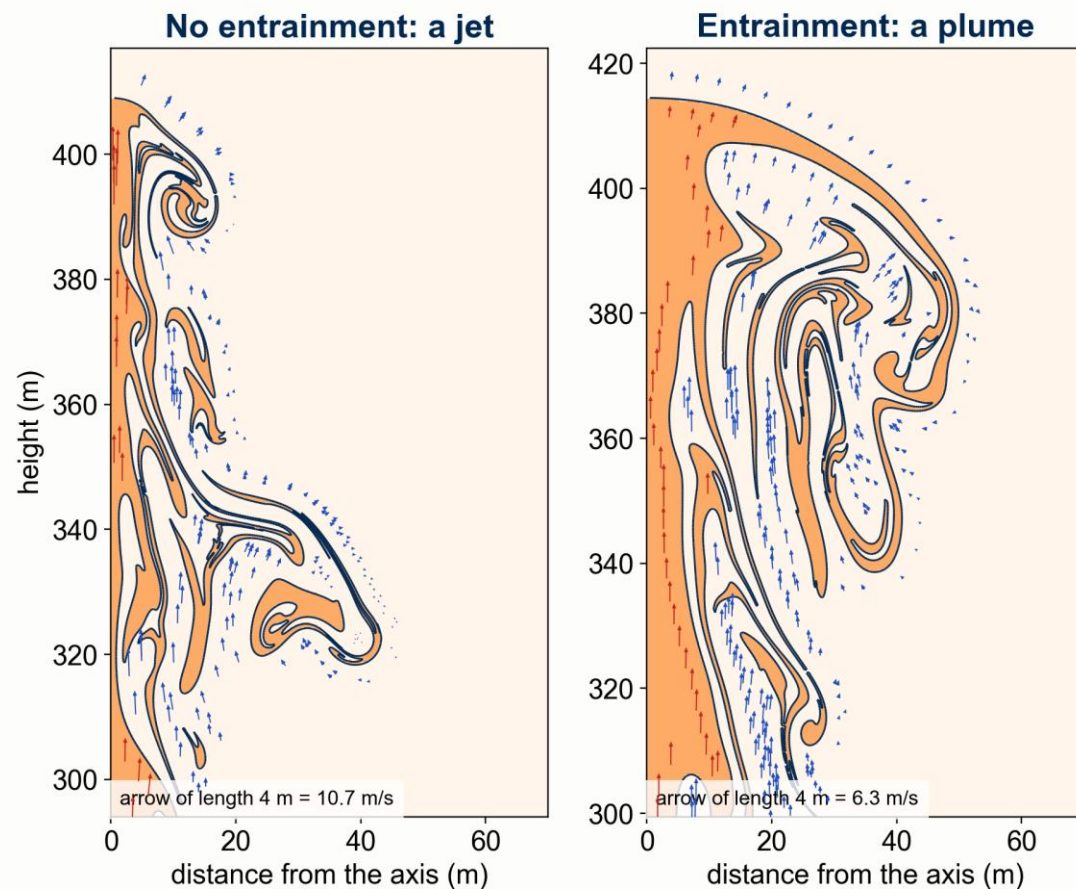


Entrainment: a plume



undiluted hot air  fully mixed

Entrainment: the plume draws in surrounding air



Air velocity 3 m off the interface: red inside the plume, blue outside

The cap entrains

Under the rim of the cap, surrounding air is drawn inward and upward into the plume. Without entrainment it is pushed aside.

Spreads like a plume

The plume widens by 0.15 m per meter of rise; plume theory predicts 0.12 for $\alpha = 0.1$.

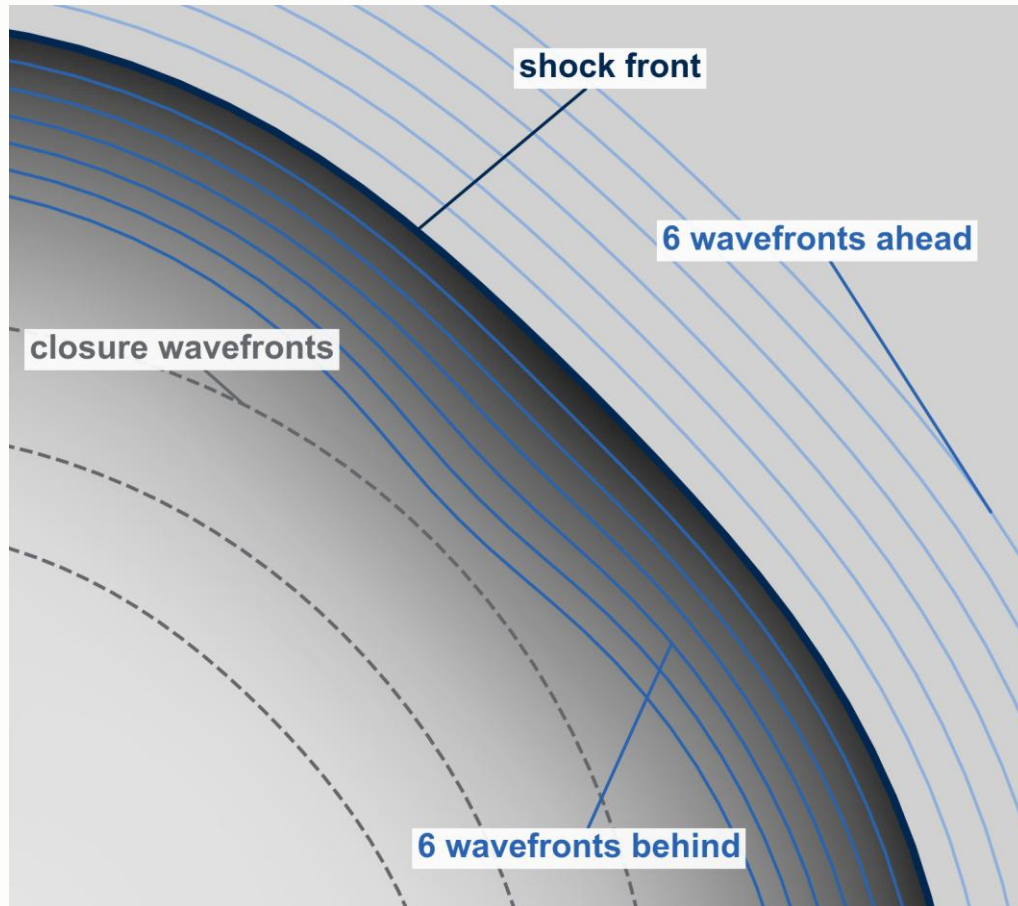
3.4× more entrained air

The head holds 285,000 m³ of entrained air, against 83,000 m³ without entrainment.

Slows as plume theory predicts

Rise speed decays with height like z to the power $-1/3$. At 400 m the theory gives 3.3 m/s and the model gives 3 to 3.5 m/s.

The 1D shock model: a shock front as an interface



Diverging front: the model's curves on the 2D Euler pressure (gray)

The shock as a front

Unknowns live on the shock front and a few wavefronts on either side: 19 curves, each a function of one variable along the front.

Jump conditions in front, characteristics behind

The front carries the exact jump conditions. Behind it, the wavefronts evolve W_n and the other differentiated Riemann variables: the characteristic information that strengthens or weakens the shock.

GSD is an extreme low-order model

Whitham's geometrical shock dynamics moves the front alone, by one algebraic relation between Mach number and ray-tube area. It has no access to the flow behind the shock.

Next: merge with the z-model

A shock front meeting a material interface: shock–interface interaction (Richtmyer–Meshkov) at the cost of interface models.

Differentiated Riemann variables: a local system on wavefronts

Differentiated Riemann variables

$$\begin{aligned}\dot{W}_n &= (\mathcal{N} \cdot \partial_n u + \partial_n \sigma - \frac{\sigma}{2\gamma} \partial_n k) \circ \psi, & \dot{A}_n &= (\mathcal{T} \cdot \partial_n u) \circ \psi, \\ \dot{Z}_n &= (\mathcal{N} \cdot \partial_n u - \partial_n \sigma + \frac{\sigma}{2\gamma} \partial_n k) \circ \psi, & \dot{S}_n &= (\partial_n k) \circ \psi\end{aligned}$$

Normal parts; the tangential parts use $\partial\tau$. $\mathcal{N}$, $\mathcal{T}$: normal and tangent of the wavefront.

Geometry of the fast wavefronts

$$\begin{aligned}\partial_t h &= g^{\frac{1}{2}} \Lambda_3, & \hat{\partial}_t h_{,2} &= g \left(\frac{1+\alpha}{2} \dot{W}_\tau + \frac{1-\alpha}{2} \dot{Z}_\tau + \frac{\alpha}{2\gamma} \Sigma \dot{S}_\tau \right), \\ \hat{\partial}_t J_g &= \frac{1+\alpha}{2} J_g \dot{W}_n + \frac{1-\alpha}{2} J_g \dot{Z}_n + \frac{\alpha}{2\gamma} \Sigma J_g \dot{S}_n\end{aligned}$$

Position h , slope $h_{,2}$ and Jacobian J_g ; the shock forms where J_g reaches zero.

Fast family

$$\frac{1}{\Sigma} \hat{\partial}_t (J_g \dot{W}_n) + \alpha J_g \hat{\partial}_2 \dot{A}_n - \frac{\alpha}{2} P_d g^{-1} \hat{\partial}_2 h_{,2} = P_{\dot{W}_n}$$

Only the derivative along the wavefront (blue) appears; P : polynomials in the unknowns.

Each wavefront evolves by itself

The geometry and the fast family need no derivative across wavefronts, only along them: every wavefront is a one-dimensional object.

Shock formation without singularities

The shock forms where the Jacobian vanishes, while the differentiated variables stay finite. The model opens its shock exactly there.

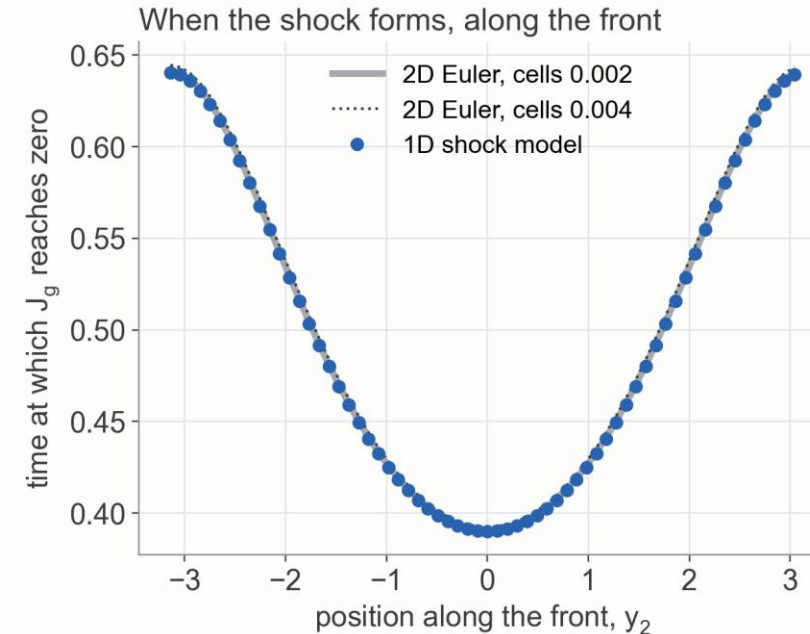
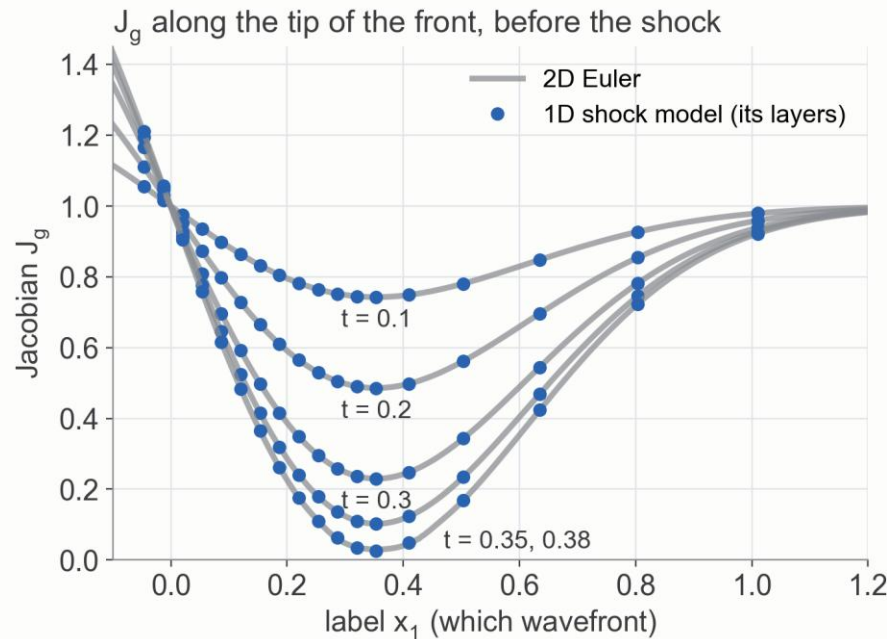
Ideal for an interface model

Only the slow and contact waves cross wavefronts, and only by transport, so a thin band of wavefronts at the shock suffices.

Local, unlike the z-model

Only derivatives, no Biot–Savart integral: each point needs only its neighbors along the curve. Ideal for massive parallelization.

The model predicts when and where the shock forms



A shock forms where wavefronts meet

Neighboring fast wavefronts converge and their Jacobian goes to zero. The model carries the Jacobian by its own equation, with no derivative across wavefronts.

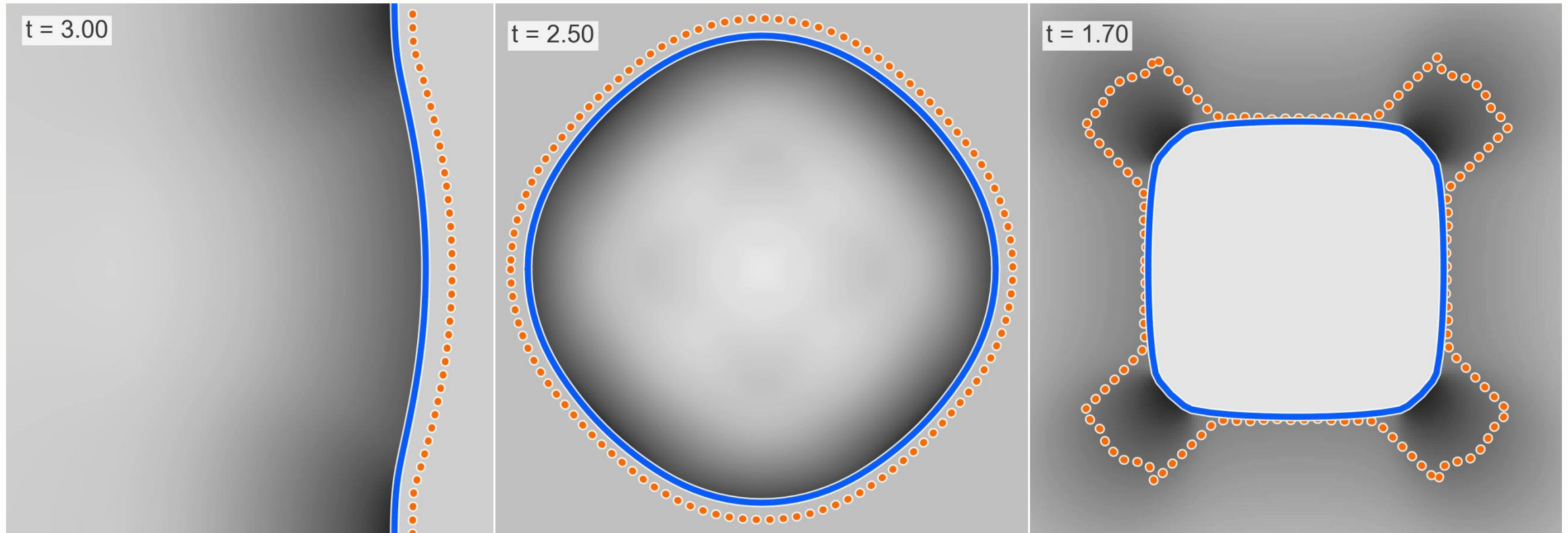
Started from smooth initial data

No Euler state is used. Each point of the front starts its shock when the Jacobian nearly vanishes, and the front then propagates as before.

Formation time within 0.001

of 2D Euler along the whole front, while the two Euler grids differ by 0.0035. On plane pulses, within 0.5% of the exact formation time.

Planar, diverging and converging shock fronts



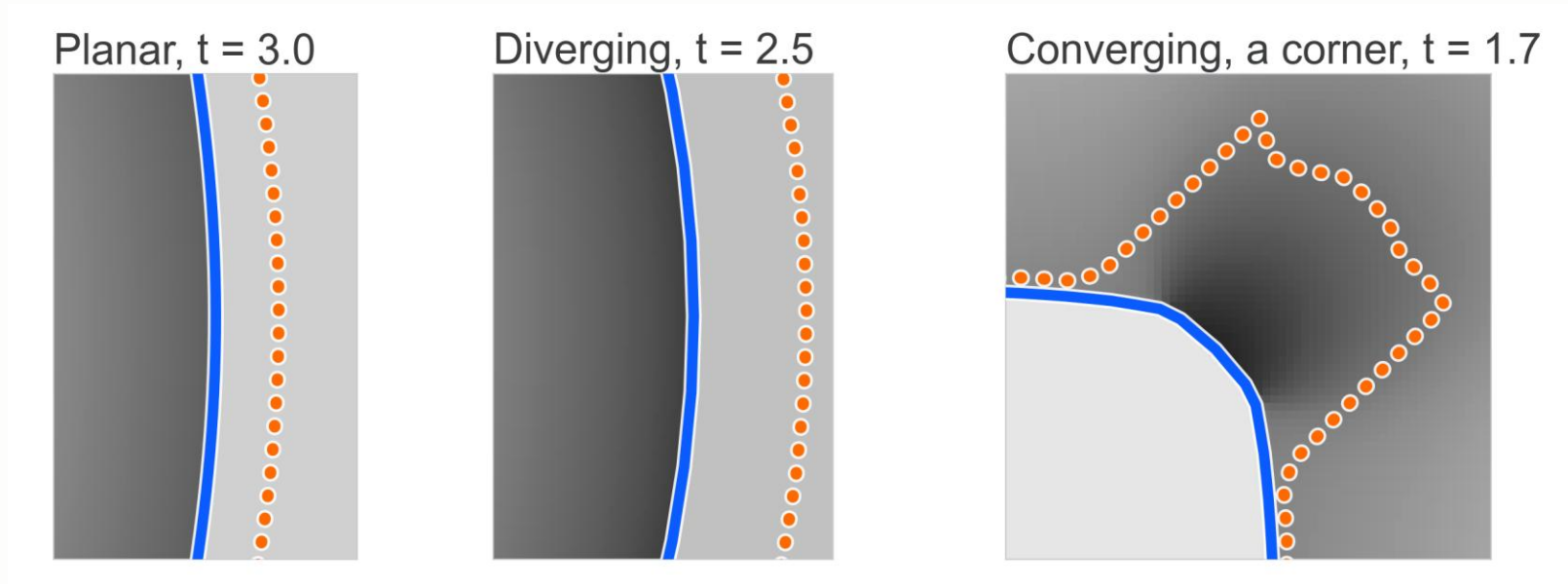
Planar front, Mach 1.4 to 1.9

Diverging front, Mach about 2

Converging front

Gray: 2D Euler pressure. **Blue: 1D shock model.** **Orange dots: classical model (GSD).** All started from the same 2D Euler state.

Matches 2D Euler at a hundredth of the cost



Final fronts, zoomed. Gray: 2D Euler pressure. Blue: 1D shock model. Orange dots: classical model (GSD).

0.005

largest front error

against 2D Euler on all three tests;
the perturbation history to 0.4%

0.3 – 0.5

error of GSD

no access to the flow behind the
shock; false corners on the
converging front

100 – 350×

faster on one core

than the 2D Euler simulation of the
same accuracy

1,000 – 4,000×

projected, once compiled

260× fewer points and 12–16×
fewer time steps, at the same cost
per point per step

Summary and next year

Delivered

3D z-model with adaptive meshing, now being parallelized at UNM



Started early

Merger and pinch-off of 3D interfaces

New

Cooling-tower plume with variable density carried on the interface



New

1D shock model: 2D shock fronts at a hundredth of the cost



Next year

Condensation · full 3D plume · parallel topological transitions with UNM · shock model merged with the z-model for shock–interface interaction · clean restart of the compressible model



Supported by the NNSA PSAAP IV program through the COMPASS center.

Next

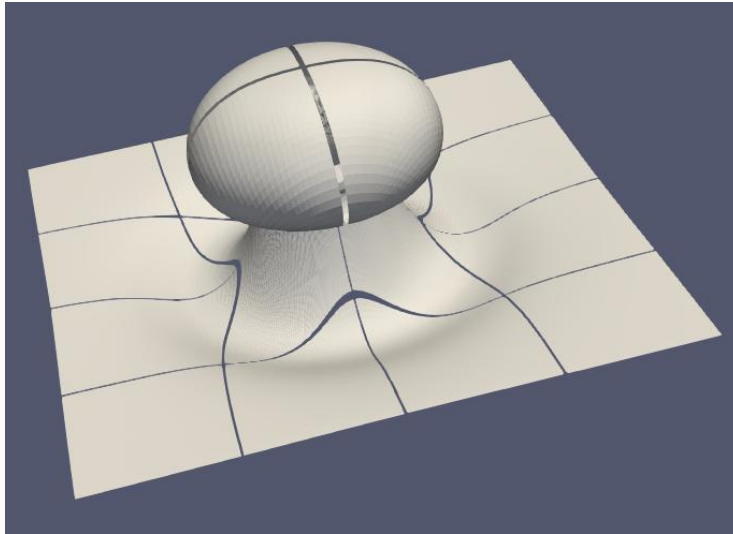
Parallel Implementation

Jason Stewart, University of New Mexico

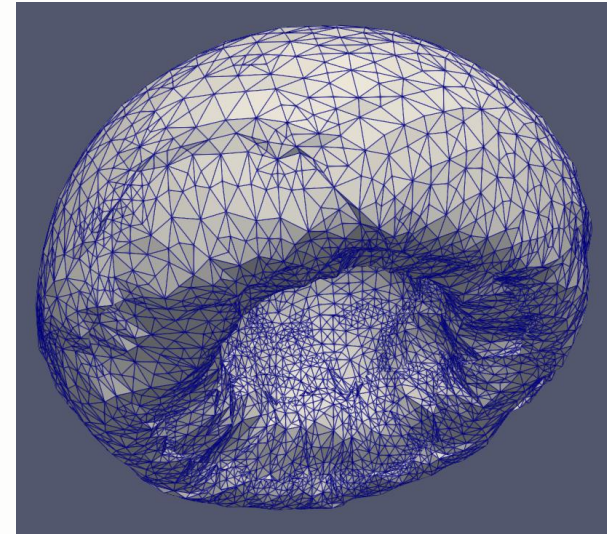


Redesigned Beatnik from the ground up

- Too much refactoring needed to move from structured, open surfaces to unstructured, closed surfaces with updated mathematics.
- Core principles unchanged:
 - Far field force solve.
 - Neighbor vertex haloing to compute surface derivatives.



Original Model



Redesigned Model

We introduce two new supporting libraries

Tessera: Unstructured mesh support

- Face, edge, and vertex data stored in three Cabana array of structures of arrays. (AoSoAs)
- Initial distributed mesh build.
- Refinement performed by splitting edges that become too long compared to neighbors.
- Build halos:
 - Rebuild needed after each mesh update, either edge splitting or mesh quality updates.
- Mesh quality operations still need completing.
 - Collapse edges too short.
 - Flip edges to keep vertex valence near 6.

Canopy: Far-field solve via the fast multipole method

- Input: positions (N, 3) and charges (N, 3). Charges are area-weighted vector sources per owned Tessera vertex.
- Output: potential (N, 3), gradient (N, 3, 3). Beatnik uses the gradient for velocity or the Riesz scalar calculations.
- Bin particles into an adaptive octree. ParMETIS load-balances the tree cells.
- Particles migrate to the owner of their leaf, which gives a decomposition separate from Tessera's.

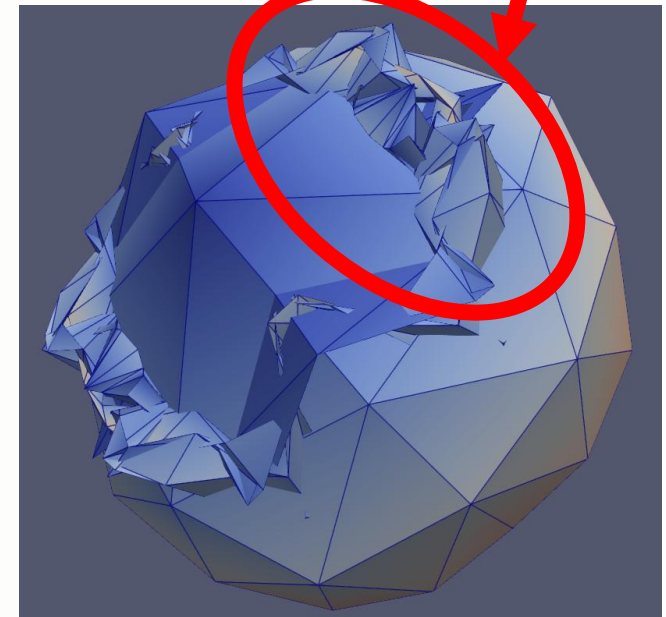
Status: Profiling and optimizing the FMM solve

- Building the interaction list between FMM cells is the bottleneck.
“The FMM halo build.”
 - See poster for details!
- Avoid by minimizing changes to tree topology.

Next steps:

- Complete mesh quality development to profile the FMM with many points on the bubble.
- Time stepping too slow if initial conditions have too many points.
- However, many points are needed for advanced rollup.
- Current profiling uses simple gravity solve for 500,000+ particles.
- Again, see poster for details!

AMR needed; the surface rollup intersects its upper boundary



The rising bubble without AMR