# *Computational Manufacturing: Toward Simulating Metal Casting and Welding Processes**

**Douglas B. Kothe**
**Computer and Computational Sciences Division**
**Los Alamos National Laboratory**
**Los Alamos, NM USA**
***dbk@lanl.gov***

***Supported by the USA Department of Energy Advanced Simulation and Computing Program**

# Project Team Members (Past/Present)

- **Los Alamos National Laboratory**
  - *Currently:* Jim Sicilian, Bryan Lally, Jeff Durachta, Ed Dendy, Sharen Cummins, Marius Stan, Kin Lam, Dave Korzekwa, Deniece Korzekwa, John Turner, Jamal Moyd-Yusof, Sriram Swaminarayan, Andrew Kuprat, Neil Carlson, Marianne Francois, Austin Minnich, Sam Johnson, Andre Starobin, Dan Tortelli
  - *Past:* Matthew Williams, Brian Vanderheyden, Mark Schraad, Frank Harlow, Nely Padial, Dana Knoll, Vince Mousseau, Jerry Brock, Hilary Abhold, Jay Mosso, John Cerutti, Bill Rider, Richard LeSar, Mike Johnson, Laura Crotzer, Mark Taylor, Dan Hartmann, David Cerutti, Robert Ferrell
- **Elsewhere (Academia, National Labs, Industry)**
  - Markus Bussmann, David Brown, Mike Steinzig, Damir Juric, Phyl Crandall, Christoph Beckermann, Anand Reddy, Larry DeChant, Jianzheng Guo, Jeff Marchetta, Michelle Murillo, Sam Lambrakos

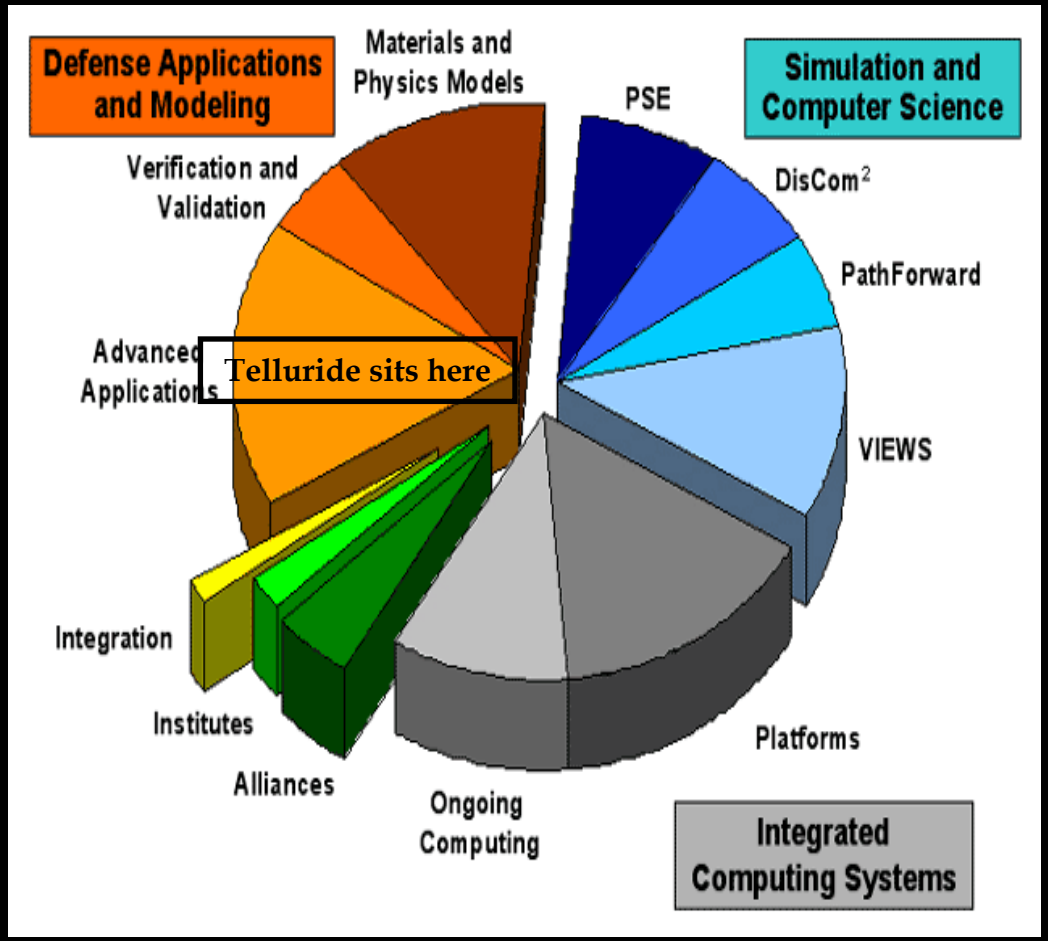
# Project Customers, Stakeholders, Users

- **Los Alamos National Laboratory (LANL)**
  - Telluride project team members: applications, V&V, development
  - Group MST-6: gravity casting (Sigma, CMR foundries), welding
  - Group NMT-5: gravity casting (TA-55 foundry); MST-6: welding
  - Group ESA-WMM: foam curing
- **Lawrence Livermore National Laboratory (LLNL)**
  - Die casting
- **Oak Ridge National Laboratory**
  - Casting microporosity
- **Y-12 National Security Complex**
  - Gravity casting, vacuum arc re-melting (VAR)
- **Department of Defense (DOD) Missile Defense Agency (MDA)**
  - Viscoelastic droplet breakup
- **Universities**
  - Colorado School of Mines, University of Ca-Irvine, Cornell, Memphis, UCLA, Iowa, Tennessee, Ohio State, Oregon State, Illinois
- **Navy**
  - Naval Research Laboratory: welding modeling
  - Naval Surface Warfare Center: droplet breakup

# The DOE ASCI Program

- **Objective: Meet the needs and requirements of the Stockpile Stewardship Program in**
  - *Performance:* Create predictive simulations of nuclear weapons systems to analyze behavior and assess performance in an environment without nuclear testing
  - *Safety:* Predict with high certainty the behavior of full weapons systems in complex accident scenarios
  - *Reliability:* Achieve sufficient, validated predictive simulations to extend the lifetime of the stockpile, predict failure mechanisms, and reduce routine maintenance → This project ("Telluride") fits in here
  - *Sustainability:* Use virtual prototyping and modeling to understand how new production processes and materials affect performance, safety, reliability, and aging → This project ("Telluride") fits in here
- **Useful ASCI URLs**
  - National Nuclear Security Administration (NNSA): *www.nnsa.doe.gov/asc/*
  - Los Alamos National Laboratory (LANL): *www.lanl.gov/projects/asci*
  - Lawrence Livermore National Laboratory (LLNL): *www.llnl.gov/asci*
  - Sandia National Laboratory (SNL): *www.sandia.gov/ASCI/index.htm*

# ASCI Program Structure



# Outline

- **Why simulate manufacturing operations?**
  - Overview: motivation, scope, examples, stakeholders, goals
- **Physical models for casting / welding processes**
  - A laundry list of phenomena
- **Numerical solution algorithms used**
  - Robustness and accuracy always at odds
- **Barriers to forward progress**
  - *Everything* is occurring simultaneously
- **Greatest problems, issues, and opportunities**
  - It's a new field: there are many

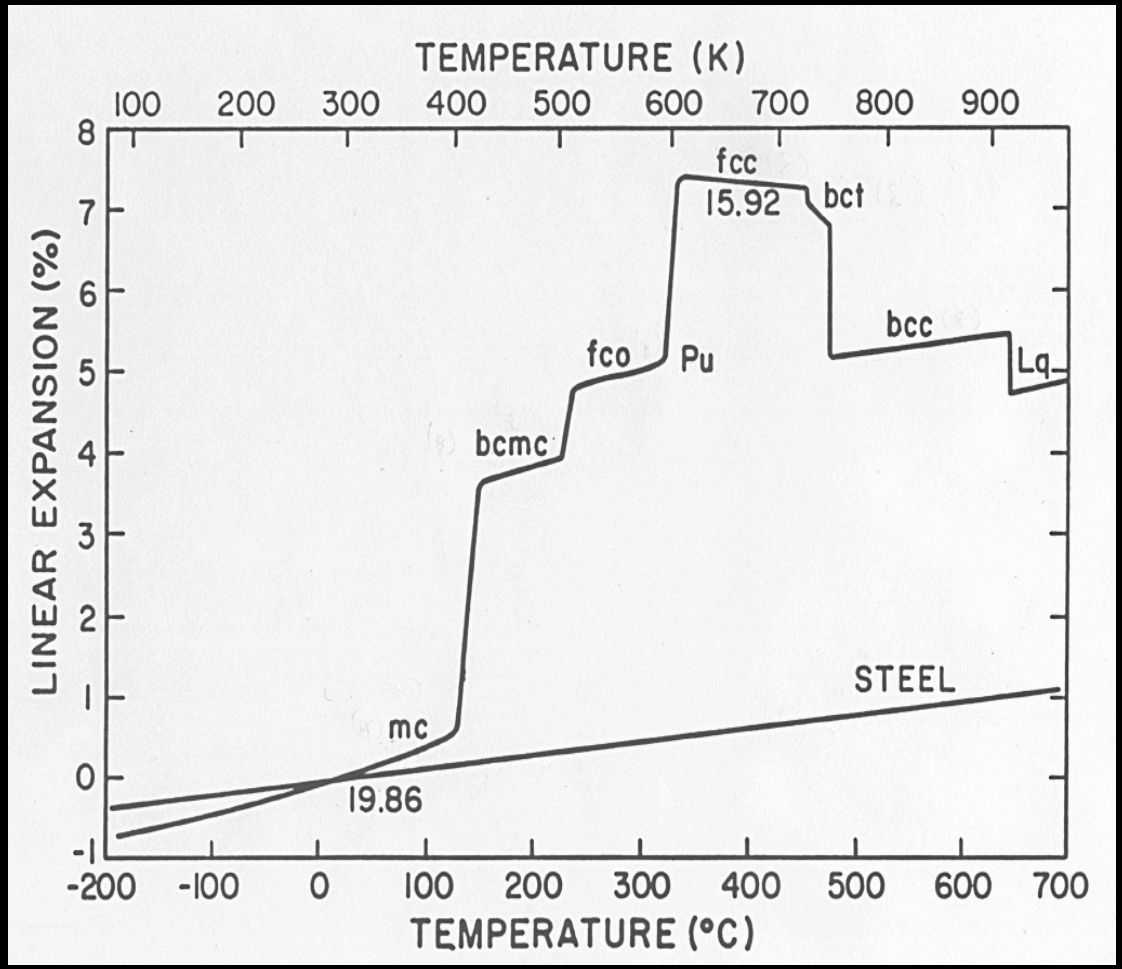
# Project Mission: Requirements

- **Deliver verified and validated computational manufacturing tools for key US Department of Energy (DOE) Complex operations**
- **Capture, within the computational tools:**
  - Realistic macroscopic models for free surface incompressible fluid flow, phase change heat transfer, material microstructure evolution, thermo-mechanical solid response, and electromagnetic effects
  - Robust, high-fidelity numerical algorithms for discrete finite volume solutions of nonlinear systems of PDEs on 3D, unstructured-mesh computational domains
  - Modular, extensible, maintainable, and portable software constructed with quality-assured software engineering practices
- **Realize efficient execution on computational platforms ranging from desktop systems to high performance computing systems typical of the DOE ASCI Program**
- **Target casting and welding operations within the DOE Complex**

# *Computational Manufacturing: Why?*

- **Many mission-essential manufacturing processes already exist within DOE Complex**
  - At National Laboratories such as LANL, SNL, LLNL, Y-12, KCP
- **New processes are needed to support future product requirements**
  - Re-manufacturing of Rocky Flats replacement components
- **These processes are *unique***
  - Involving specialized metal alloys and glove box operations
- **Past manufacturing process design…**
  - Was based on expert experience: trial-and-error, intuition
  - "Works", but is inefficient in time, energy, cost, and material
- **Can a simulation tool help?**
  - Reduce inefficiencies, provide insight, enable optimization?

# Plutonium: Not Your Typical Industrial Metal



# *Casting Simulation: Impact List*

- **Easiest: Provide qualitative insight**
  - About what has just happened
  - By answer basic “what if” questions
- **Easier: Guide macroscopic process parameters**
  - Gating & feeding rate & geometry, mold & charge preheat
  - Mold geometry, cooling boundary conditions
- **Hard: Capturing microstructural effects**
  - Thermal history, alloy micro- & macro-segregation
  - Correlation-based grain size and orientation predictions
- **Harder: Predictive, quantitative accuracy**
  - Part geometry: residual stress/distortion, porosity, cracks
  - Material microstructure specifications: grain characterization
- **Hardest: Process optimization**
  - Based on part/material design criteria
  - Based on other criteria such as energy, cost, etc.

# *Casting Simulation Tools: Their Inabilities**

- Inclusion formation (due to oxidation during pouring) and final location in casting
- Hot tears
- Mold penetration (with effects of coatings), sand defects
- Leakers due to porosity , inclusions, etc.
- Automated, optimized design of filling system
- Simple geometric tools during initial concept development
- Integration of casting models, nondestructive evaluation standards, and service performance analysis
- Incorporation of more foundry variables such as mold hardness, metal chemistry variations, etc.
- Prediction of final casting dimensions, taking into account all part/mold shrinkages/stresses, etc.
- Completing simulations in a realistic amount of wall clock time

*** Prof. Christoph Beckermann (Univ. of Iowa, 7/00)**

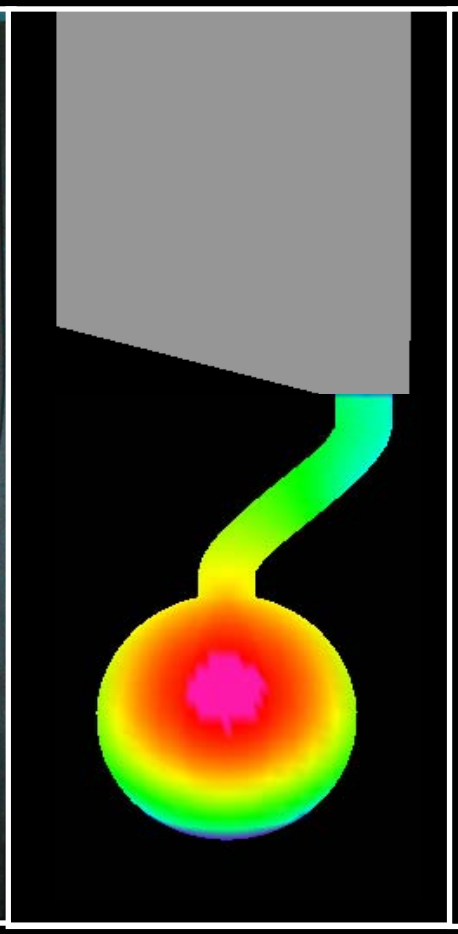
# *Casting Simulation Tools: Their Abilities**

- **Heat transfer**

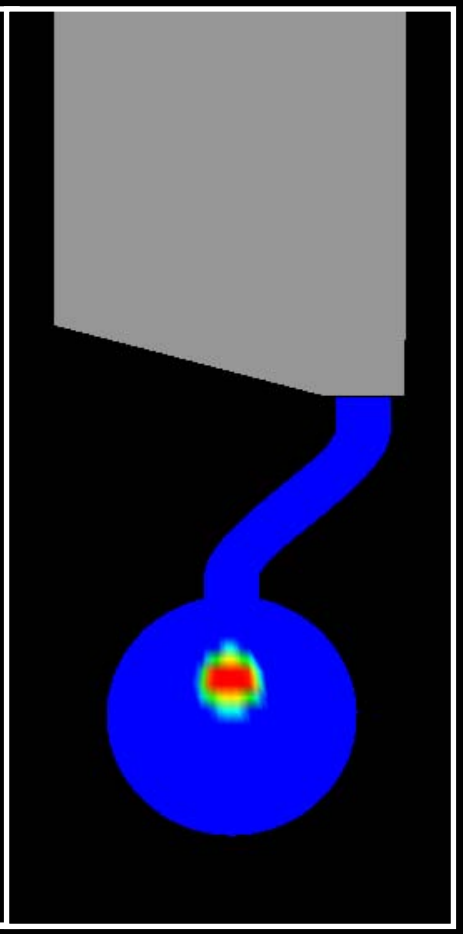
*** Prof. Christoph Beckermann (Univ. of Iowa, 7/00)**

# Application: Aluminum Ball Casting

Photo of aluminum ball casting showing porosity



Computed temperature distribution during solidification.



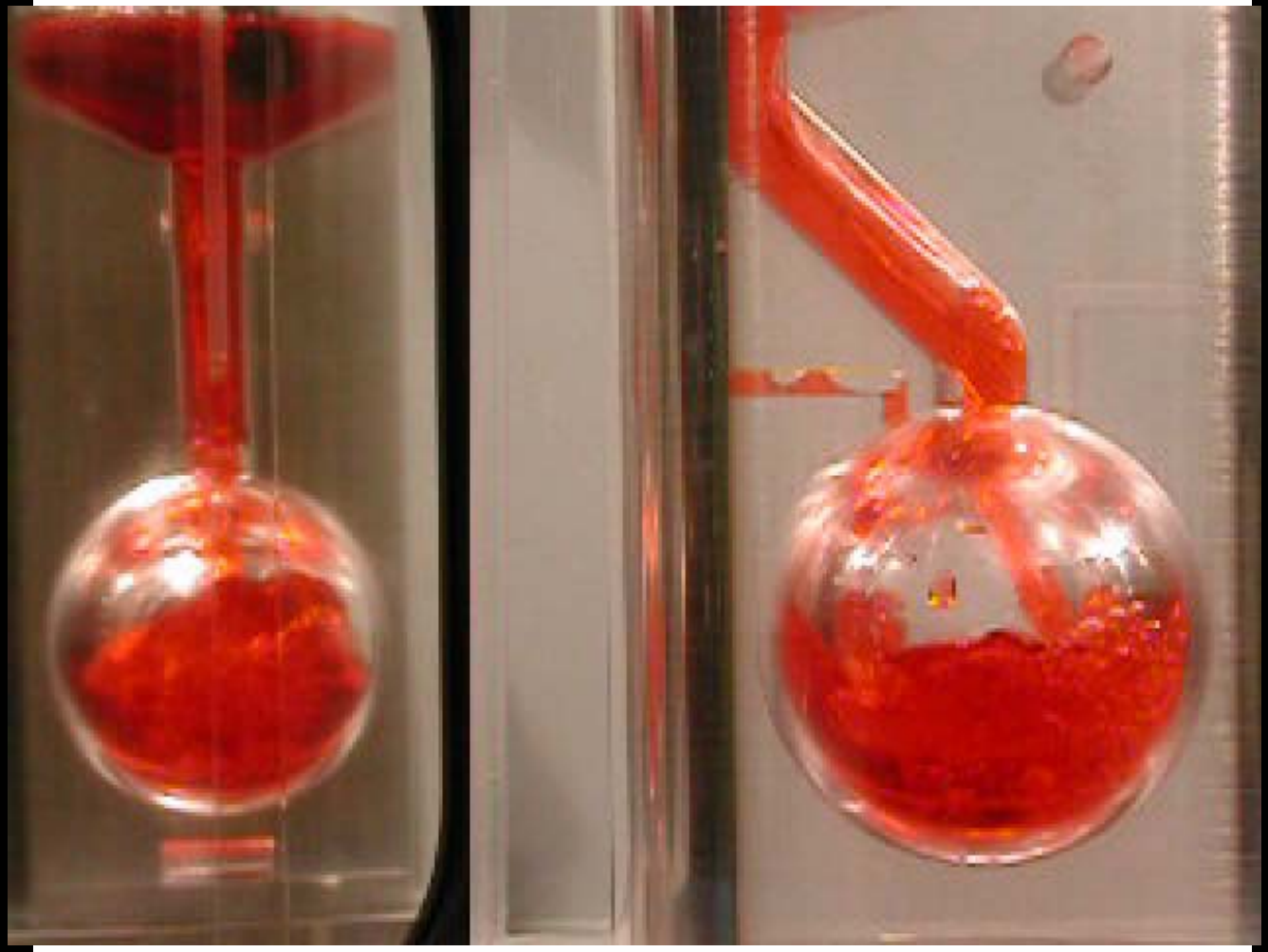
Computed volume fraction distribution; red is last to solidify

A ball is a difficult shape to cast since the last place to solidify will tend to be toward the center of the ball and will typically exhibit shrinkage porosity as shown.

Aluminum is a challenging metal to cast as it has 7 % shrinkage on solidification.

Basic phase change heat transfer simulations really do make a difference! If ICs and BCs are known, the last point to solidify is an attainable simulation result.

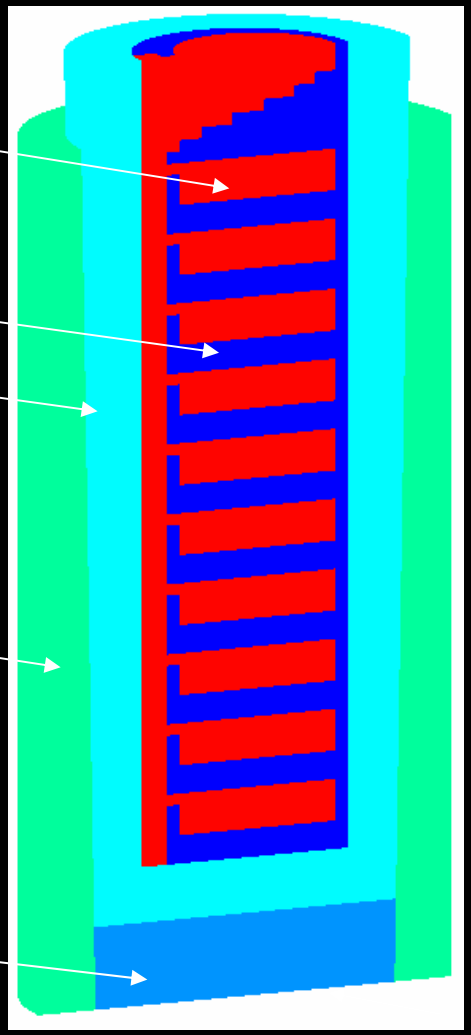
# Mold Filling Experiments, Simulations



water pouring into lexan mold

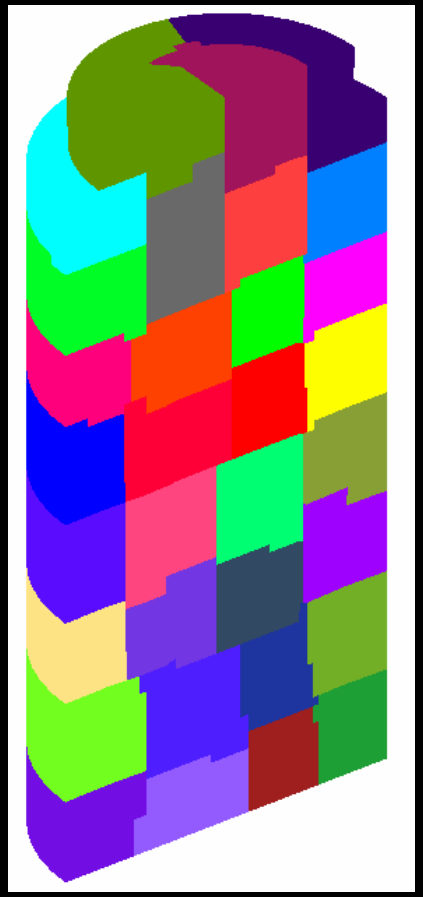
side view front view

# Application: "Puck" Castings




- Mesh is half symmetry with 38364 cells and 41847 nodes.
- All mold parts are graphite
- Simulation starts with metal at 950°C and graphite at 650°C
- Simulations are used to predict the cooling rate and expected grain size.
- Typical CPU times (4-PE Compaq ES40)
  - 21264 Alpha processor (667 MHz)
  - 2 PEs – 318 µs/cell/cycle
  - 4 PEs – 110 µs/cell/cycle

# Application: "Puck" Solidification

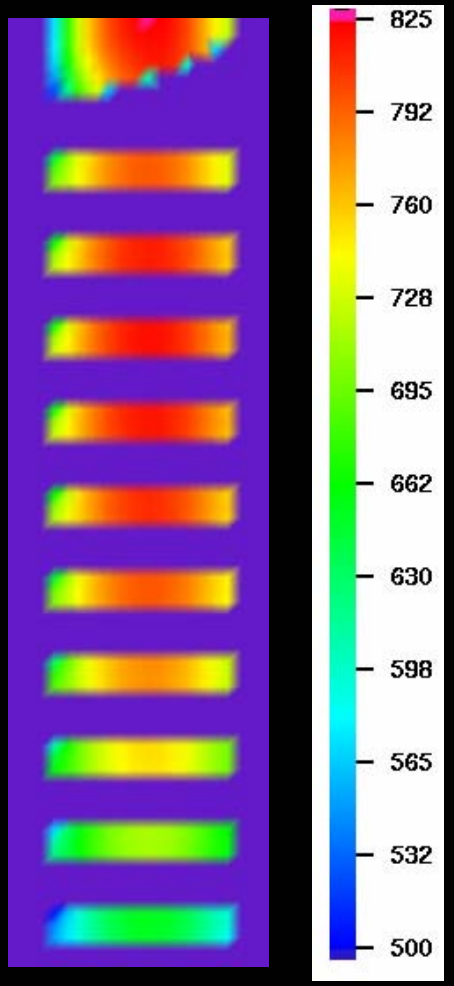


32 processor decomposition

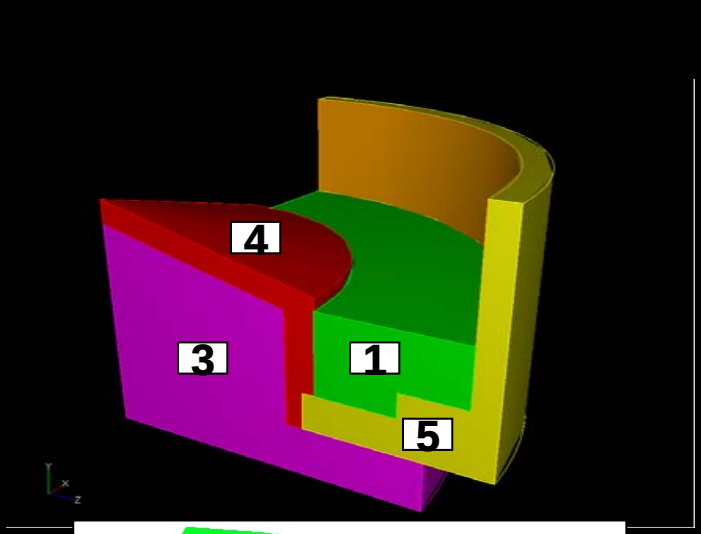
Local solidification time (time in mushy zone) and the cooling rate through the epsilon phase have been correlated to grain size and coring.

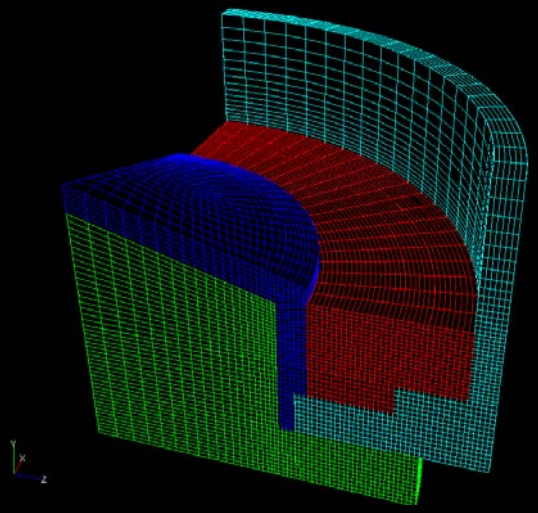
Simulations show variations between pucks as well as within each puck.

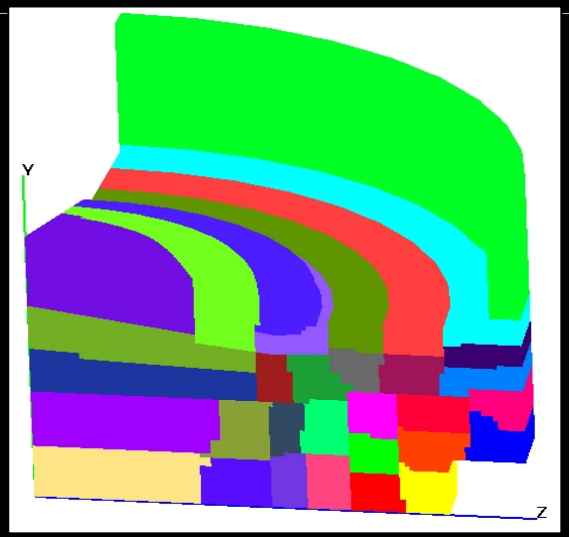
This information can be compared to other samples and be used in subsequent simulation codes.

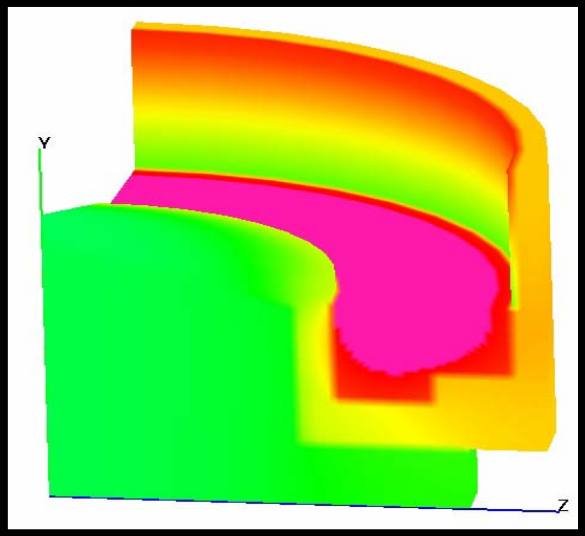


# SPR-III Reactor Ring Castings

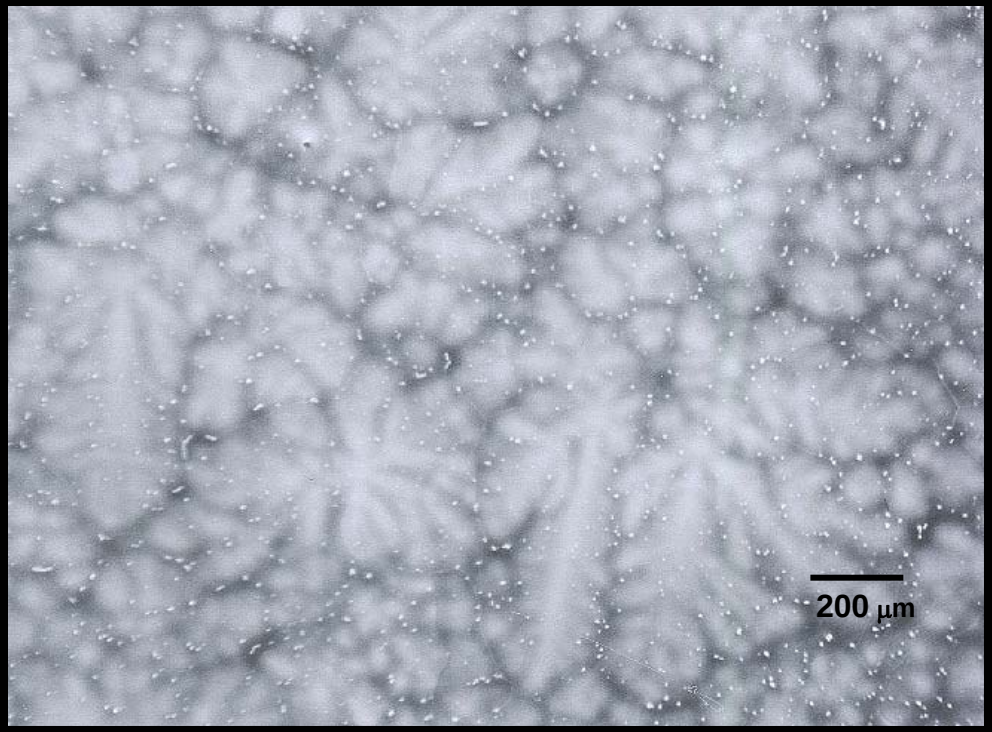


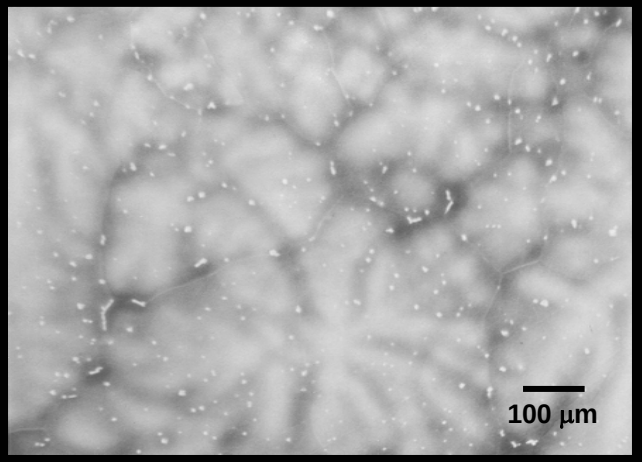



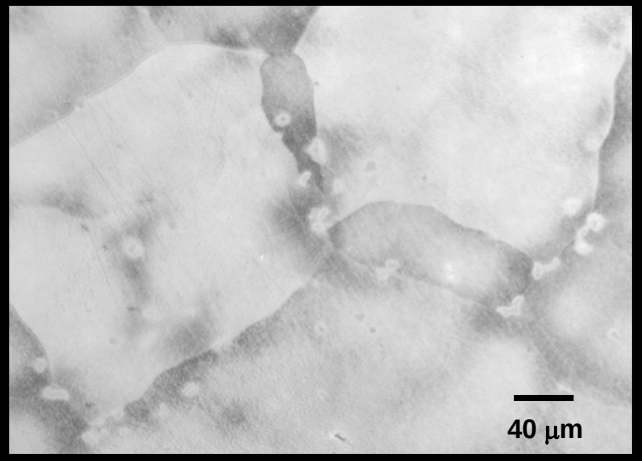


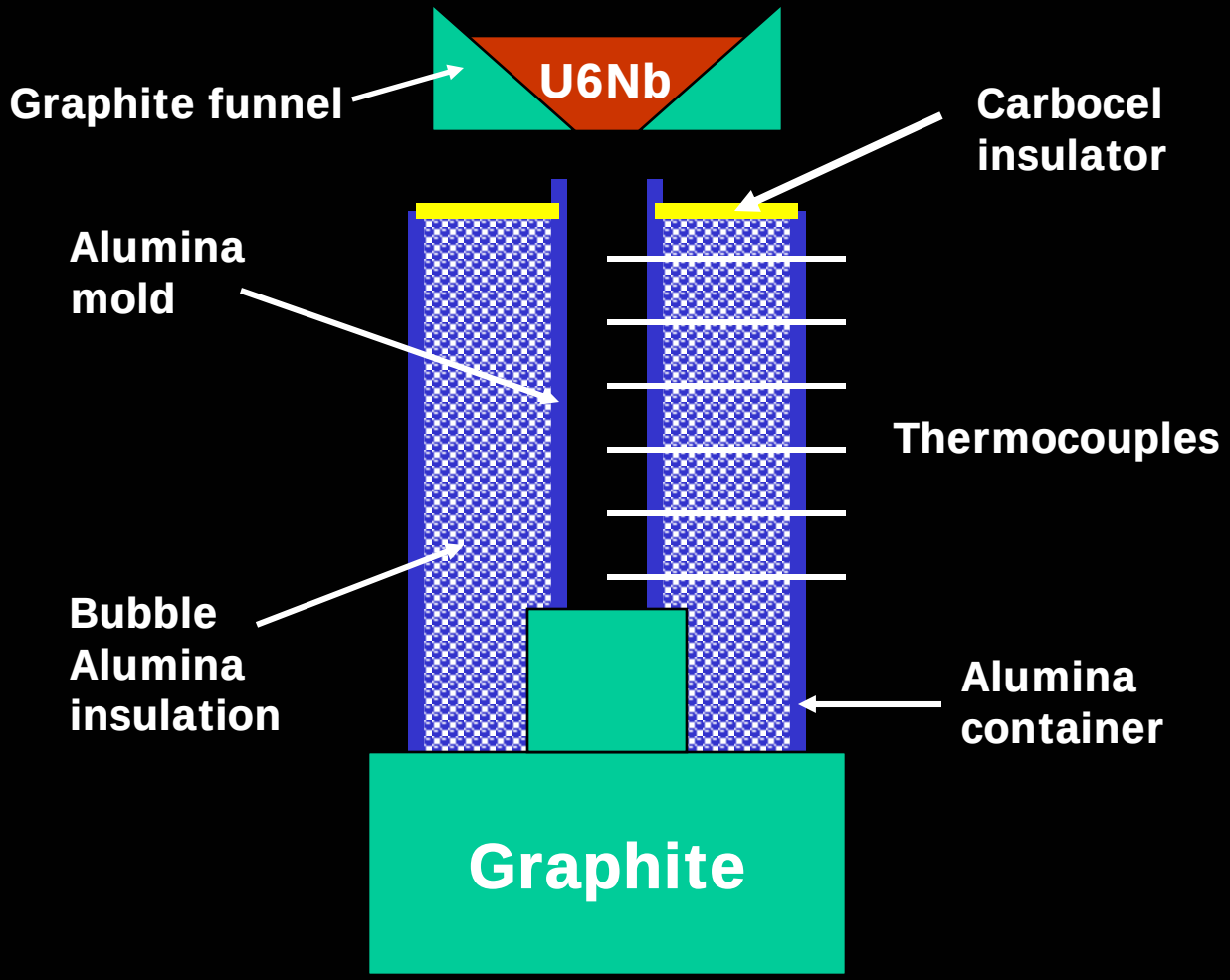
# SPR-III Ring Microstructure: Can This be Predicted?



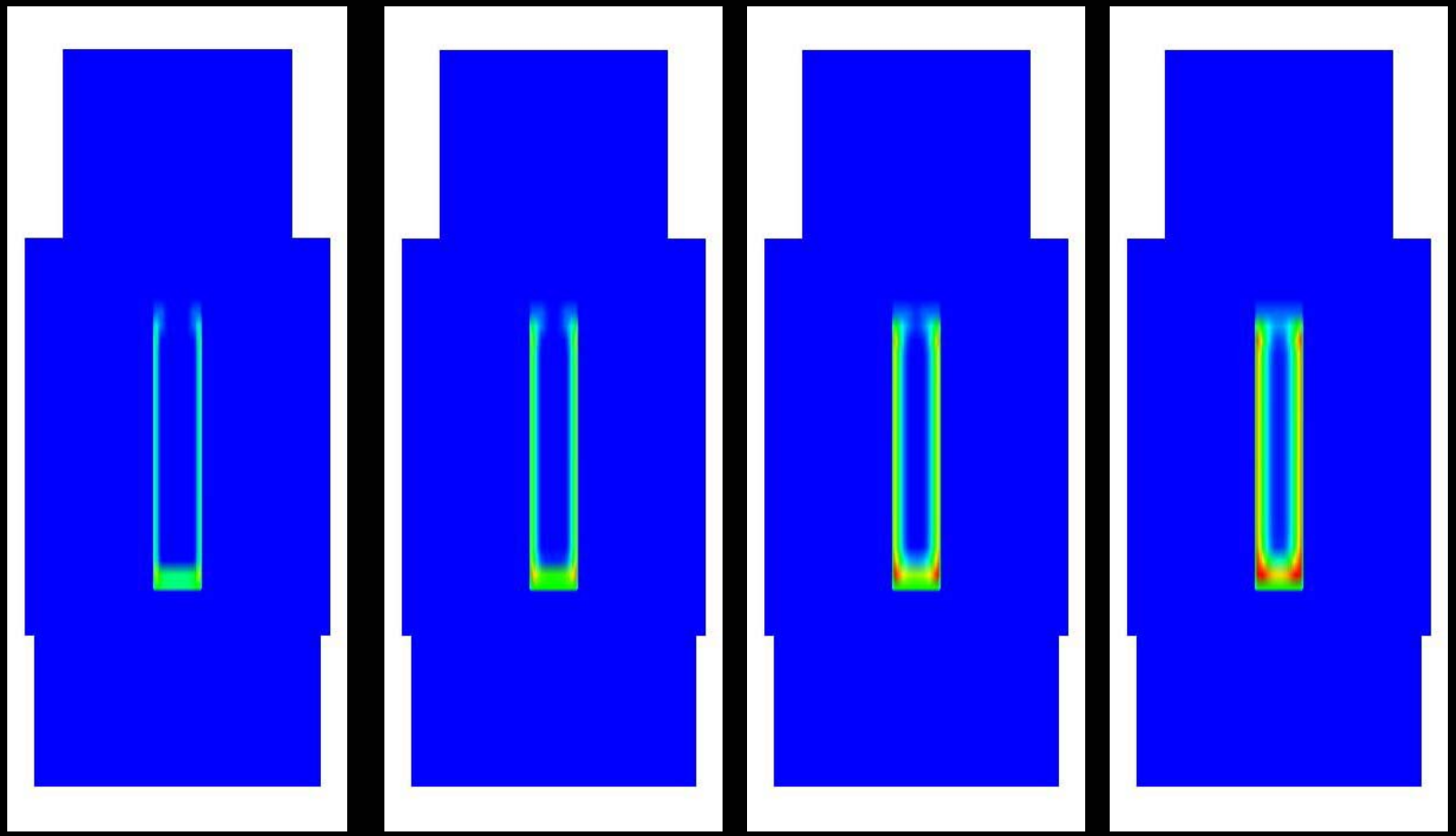
# LANL Directional Solidification Exp



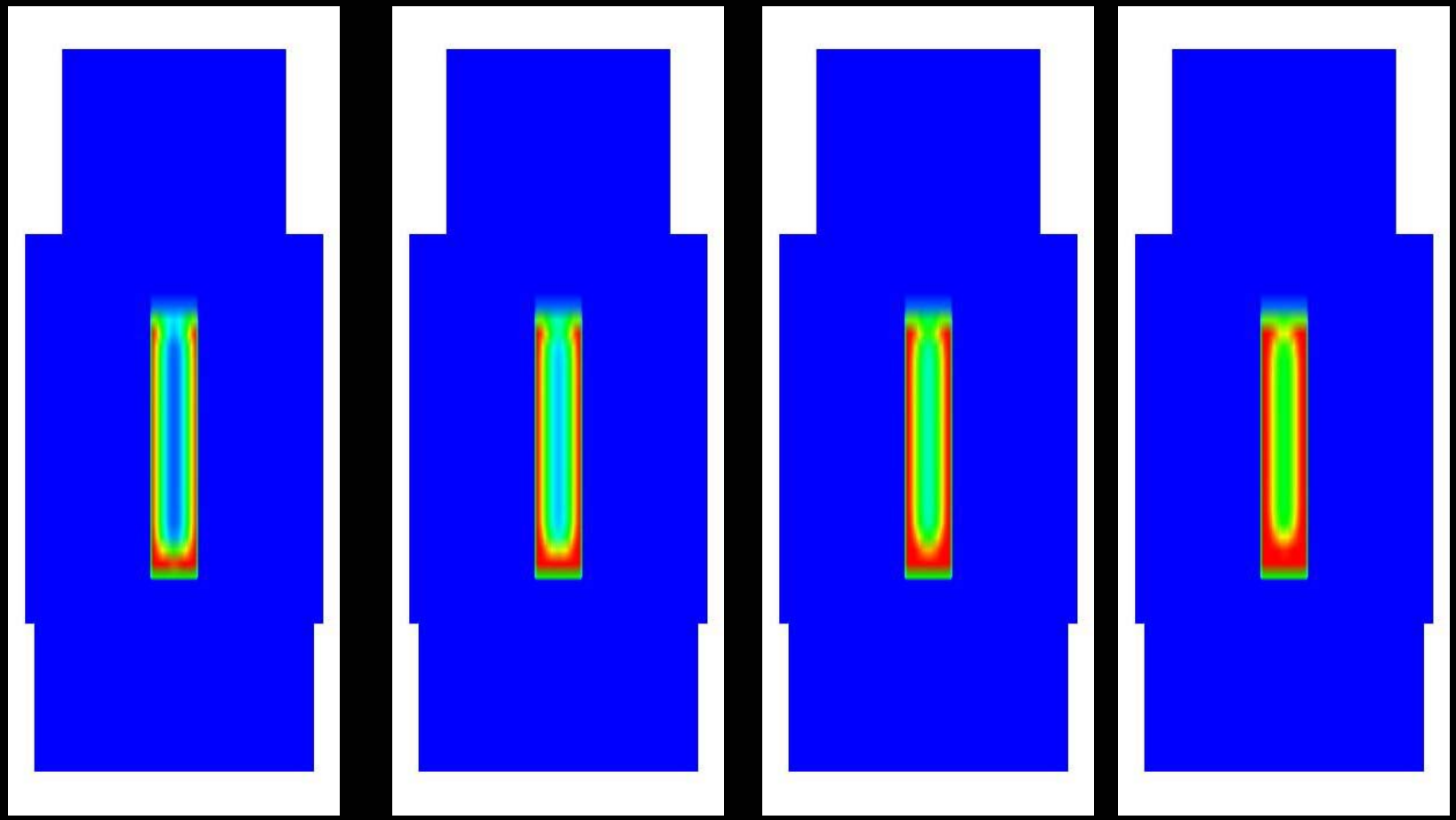

- Unheated mold
- U6Nb Temperature 1350-1475°C
- Type C and K Thermocouples
- Ideally unidirectional solidification

# LANL Directional Solidification

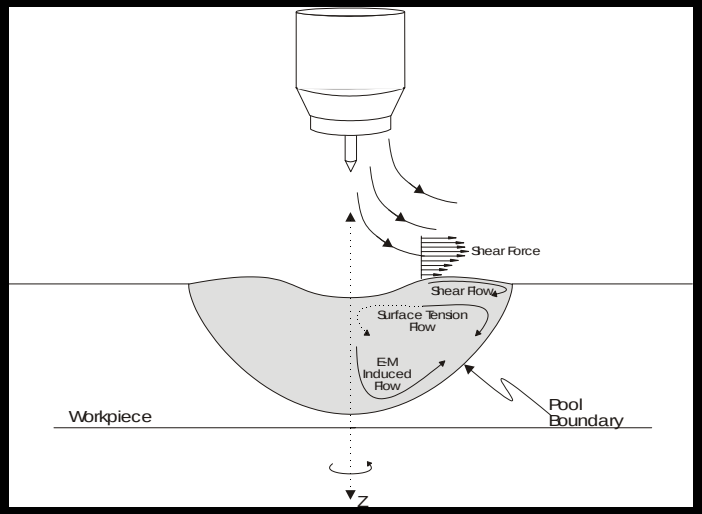
# Simulation Results Indicate a Non-Axial Solidification Front Propagation



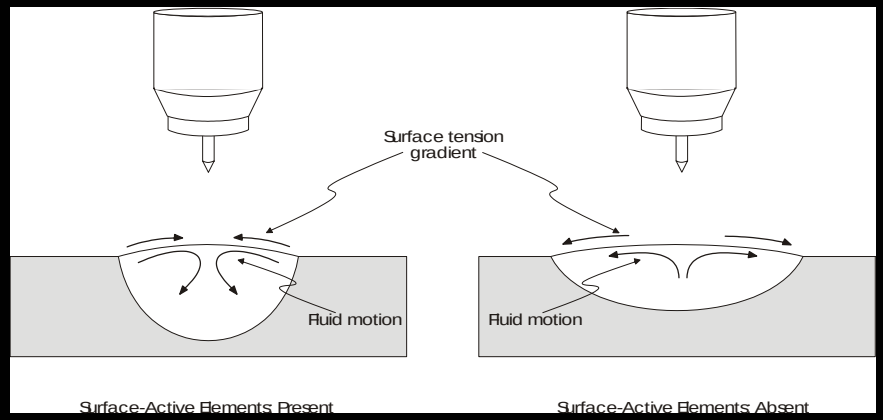
# Simulation Results Indicate a Non-Axial Solidification Front Propagation



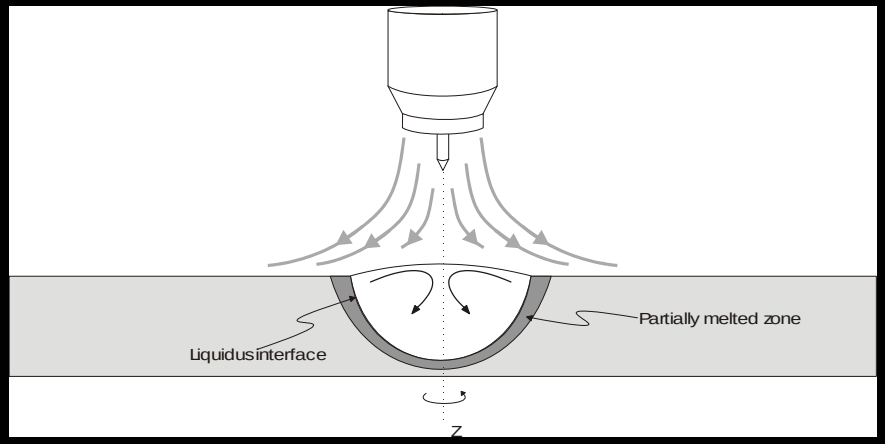
# Welding Phenomena: Convection Flows and Traveling Heat Sources



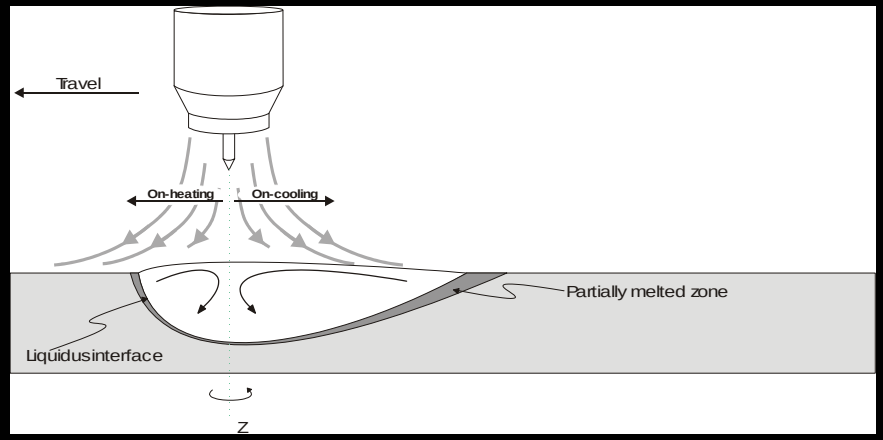

Forces affecting the convection flows inside the weld pool.




Marangoni convection model.



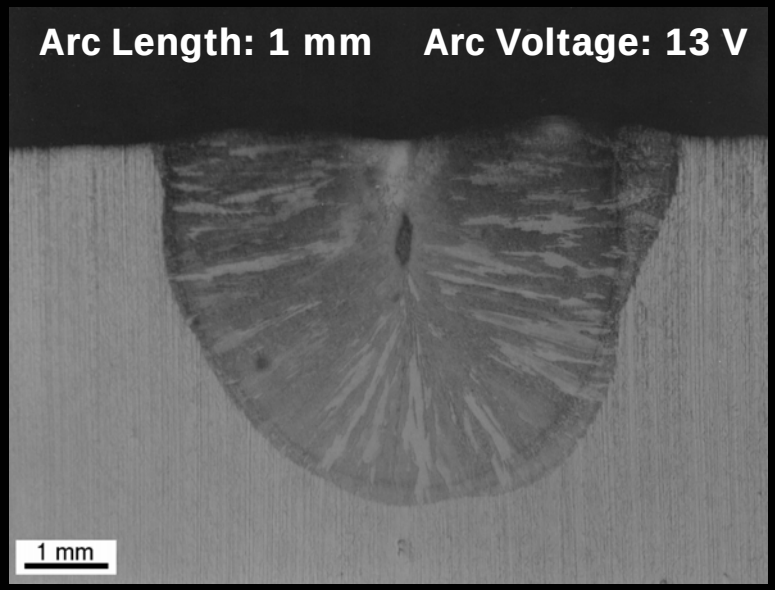

Pool geometry for a stationary heat source.

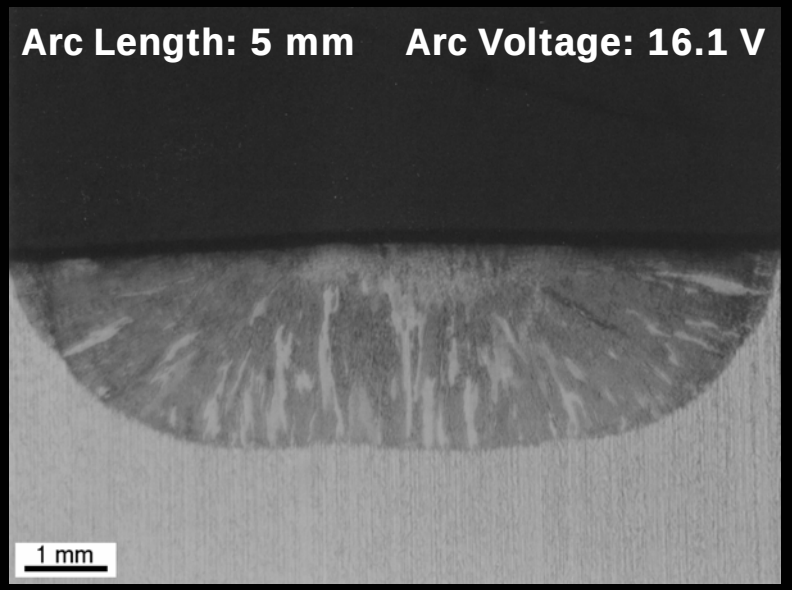



Pool geometry for a traveling heat source.

# *Useful Welding Models Must Predict Weld Pool Dimensions*

## GTAW Spot Weld Experiments on SS304L

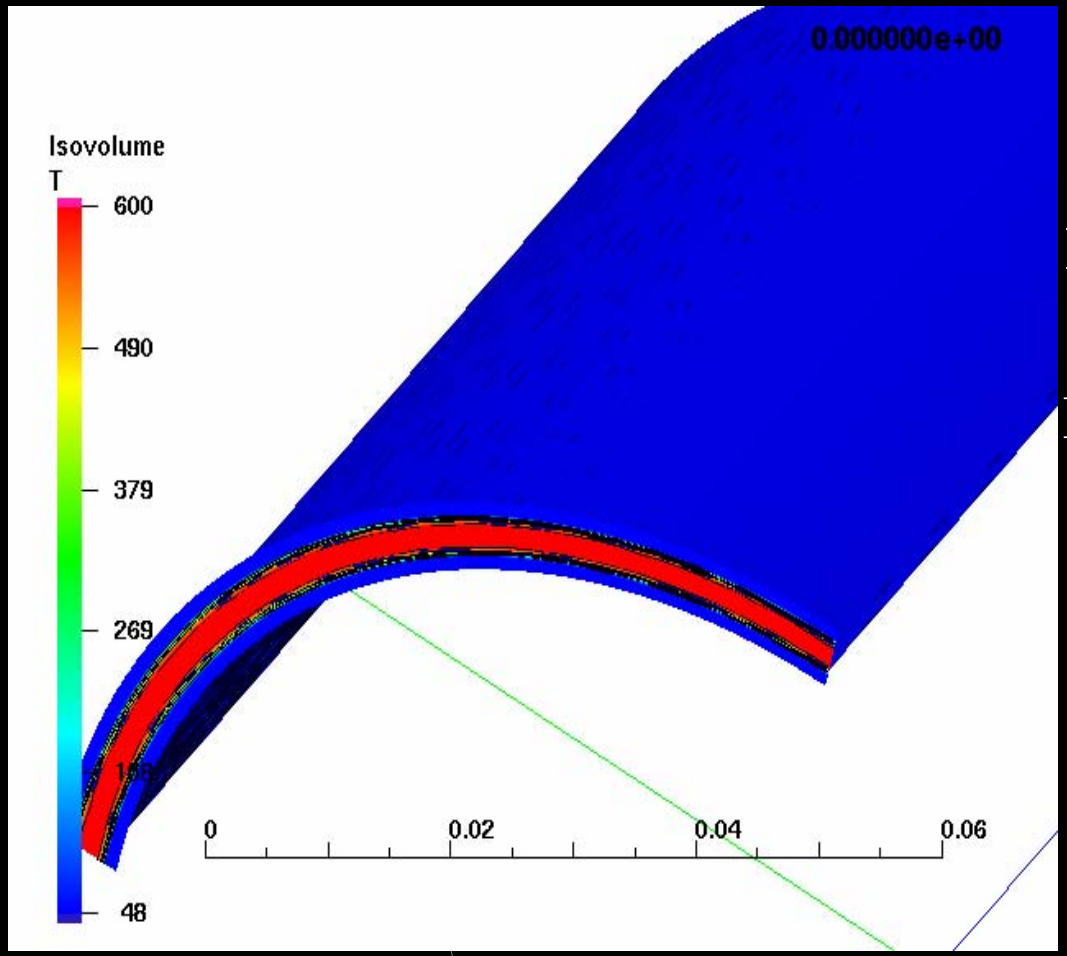



**Constant parameters**

- **Arc current: 180 A** **Beam on time: 5 s**
- **Shield gas: Argon** **Flow rate: 25 scfh**

# Pipe Butt Weld Simulation




Weld pool profile

2" Schedule 10

Aluminum 6061 Pipe

Power to work piece: 700W

Travel Speed: 47 ips

Front face shows weld pool shape (volume fractions)

Outer pipe shell show temperature profiles in Celsius

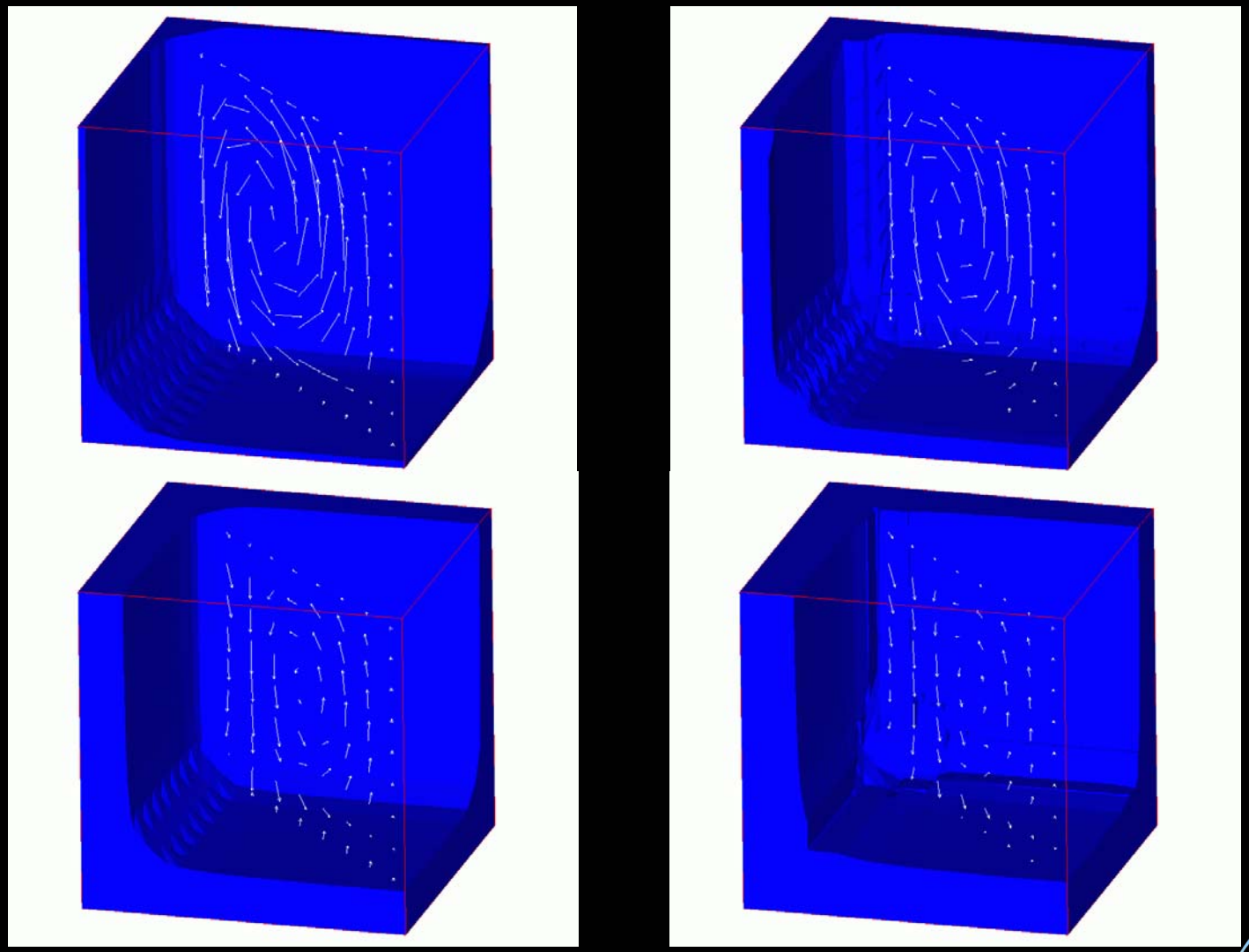
Mesh: 20x25x50

Void background

# Physical Models: Overview of Requirements

- **Heat transfer**
  - Conductive, convective, inductive
  - Radiative on boundaries
- **Phase change**
  - Liquid/solid, solid/solid transitions
  - Liquid/vapor transitions?
  - Pure materials, alloys having n species
  - Arbitrarily-complex phase diagrams
- **Incompressible fluid flow**
  - Boussinesq approximation for thermal and solutal buoyancy
  - Turbulence models for fill
  - Porous media models in mush
- **Interface kinematics and dynamics**
  - Free surfaces, fluid/solid and fluid/fluid interfaces
  - Surface tension and phase change forces
- **Thermo-mechanical material response**
  - Residual stress and distortion
  - Elastic/plastic, creep, phase change
- **Quasi-static MHD**
  - Inductive (Joule) RF heating
  - Magnetic (Lorentz) stirring
- **Material microstructure**
  - Nucleation and growth models
  - Homogenization models

# 3D Solidifying Flow



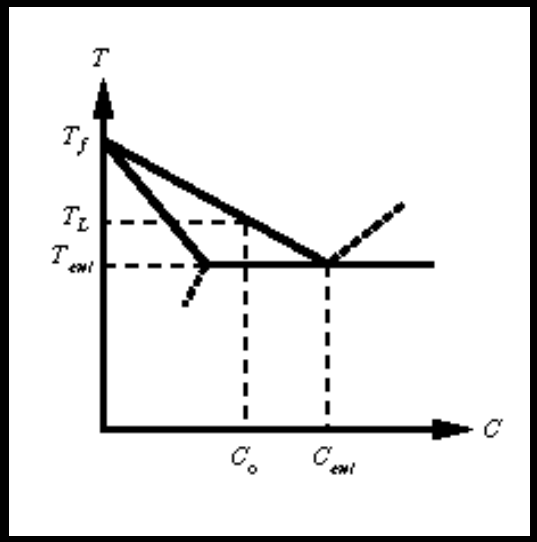
# Physical Model for Heat Transfer: A Mixture Enthalpy Formulation

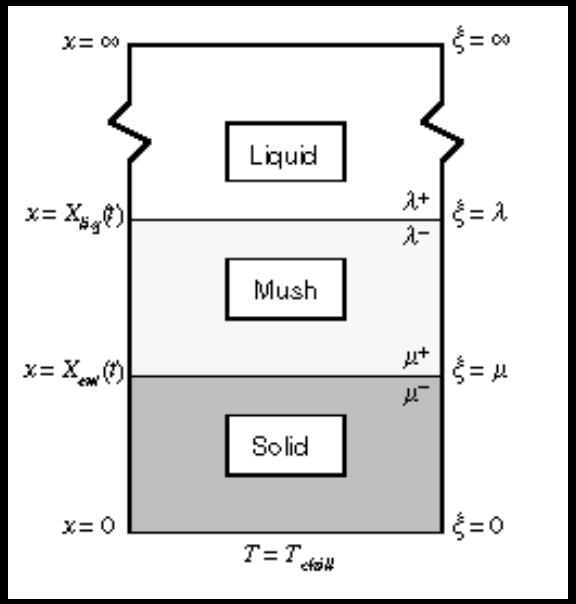
$$\frac{\partial}{\partial t}\langle \rho h \rangle + \nabla \cdot \left[ \mathbf{u} \langle \rho h \rangle \right] - \nabla \cdot \left[ k \nabla T(h) \right] - S(h) = 0$$

$$\langle \rho h \rangle (T) = \sum_k \rho_k \varepsilon_k h_k(T); \quad h_k(T) = h_k^{\mathrm{ref}} + \int_{T_k^{\mathrm{ref}}}^{T} C_{\mathrm{p},k}(T) dT$$

- Thermodynamic equilibrium: $T_k = T$ for all phases k
- Enthalpy nonlinearities are present because of:
  - Nonlinear temperature dependence of specific heats $C_p(T)$ and conductivities k(T)
  - Jumps in reference latent heats $h^{ref}$ at phase change boundaries
- Solid phases (k=s) are not allowed to flow: $u_l = u$; $u_s = \mathbf{0}$
- Phase fronts are *captured* (ε within a cell) rather than *tracked*
- Microstructure nucleation/growth models: classical (JMAC) or DNS inspired

# Alloy Solidification: Micro-Segregation Model




**Voller recently found similarity solutions for this system:**

**Int. J. Heat Mass Transfer, 40, 2869 (1997)**

$$\frac{\partial}{\partial t}[\rho h]+\nabla\cdot(\rho_l h_l \mathbf{u}_l)=\nabla\cdot(\nabla \kappa T)$$

$$[\rho h]=\varepsilon_s\rho_s h_s+(1-\varepsilon_s)\rho_l h_l$$

$$h_s=\int C_{ps}dT\ ;\ \ h_l=\int C_{pl}dT+L$$

**mixture enthalpy transport**

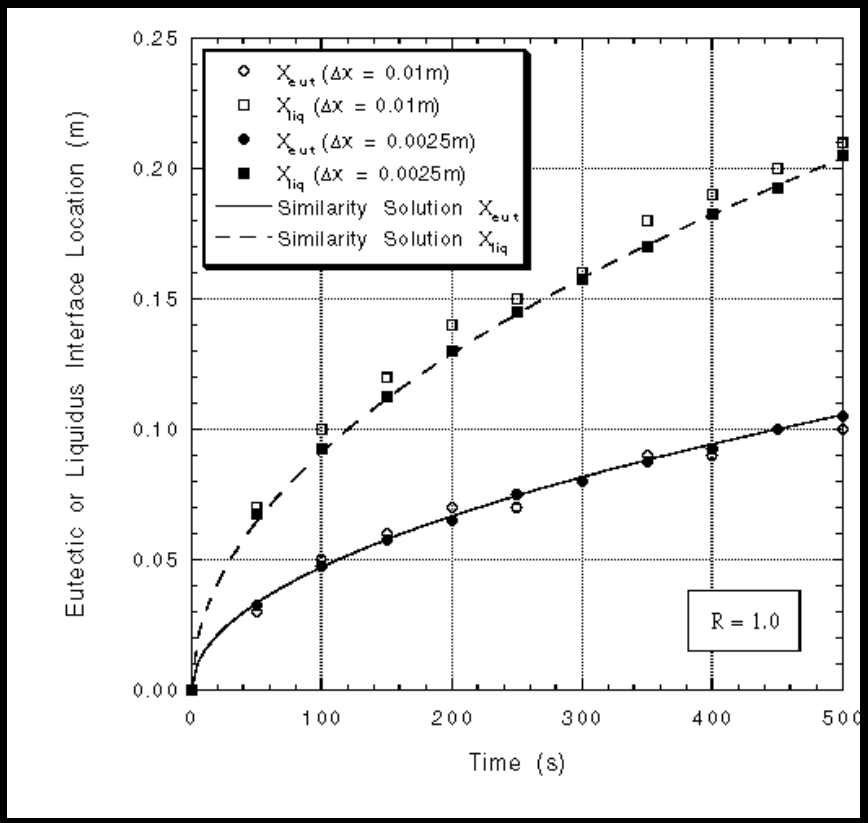
$$\varepsilon_l\rho_l\frac{\partial C_l}{\partial t}+\varepsilon_l\rho_l u_l\cdot\nabla C_l=(C_l-C_{si})\frac{\partial}{\partial t}(\varepsilon_s\rho_s)+\frac{S_v\rho_s D_s}{l_s}(C_s-C_{si})$$

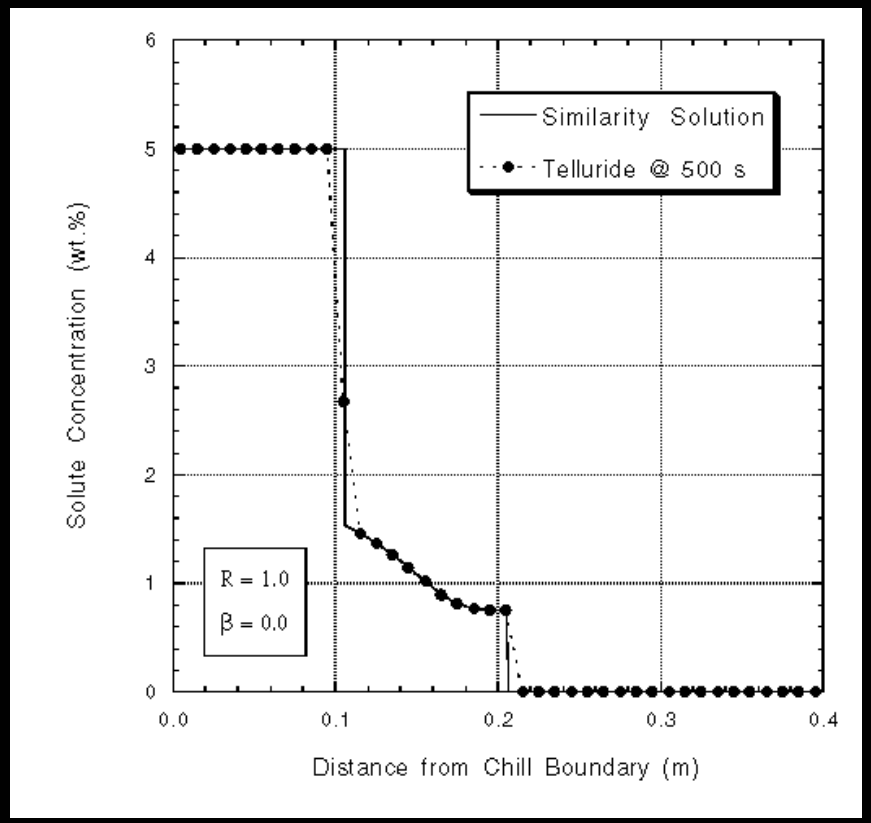
$$\varepsilon_s\rho_s\frac{\partial C_s}{\partial t}=(C_{si}-C_s)\frac{\partial}{\partial t}(\varepsilon_s\rho_s)+\frac{S_v\rho_s D_s}{l_s}(C_{si}-C_s)$$

$$T=T_m+m_l C_{li}\ ;\ \ C_{si}=kC_{li}$$

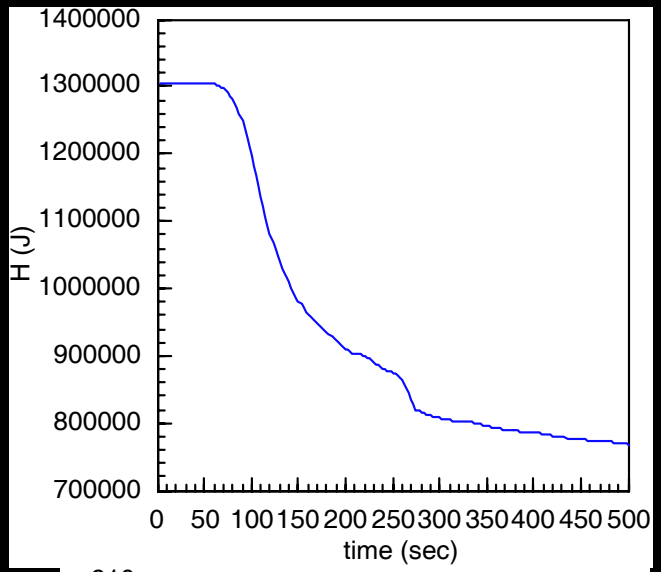
**alloy species transport and phase diagram**

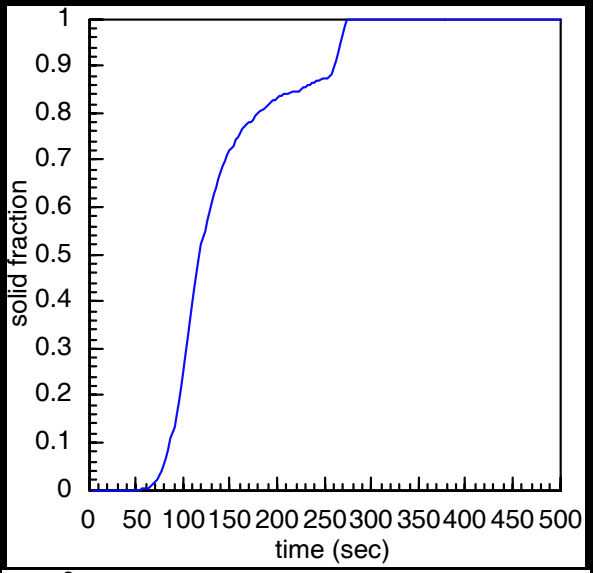
# Alloy Solidification Verification

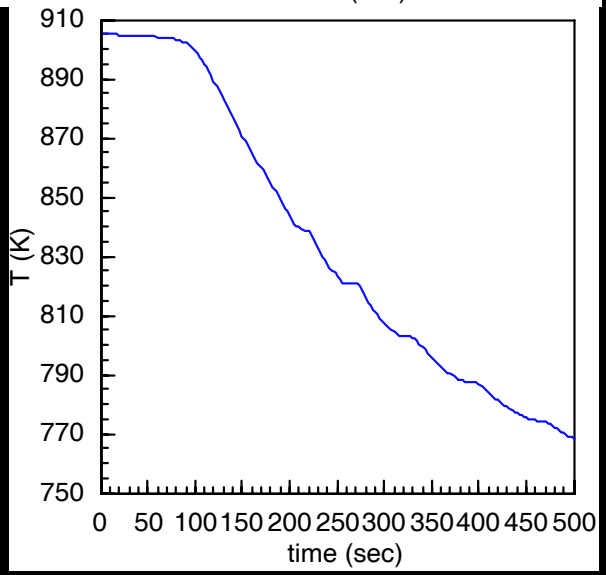


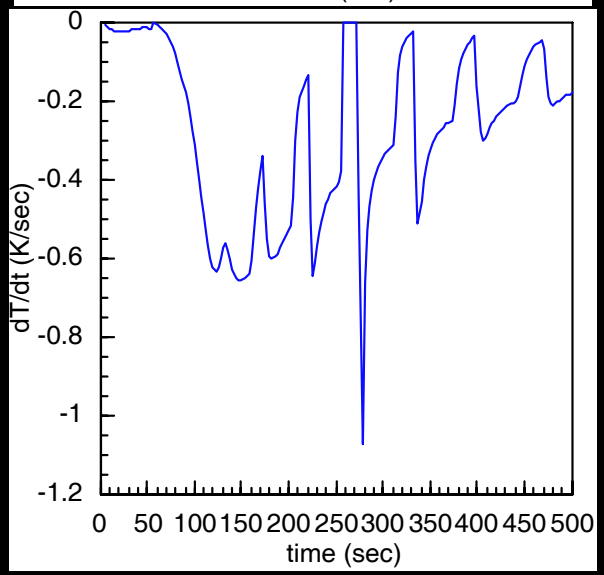



# Capturing the Correct Thermal History



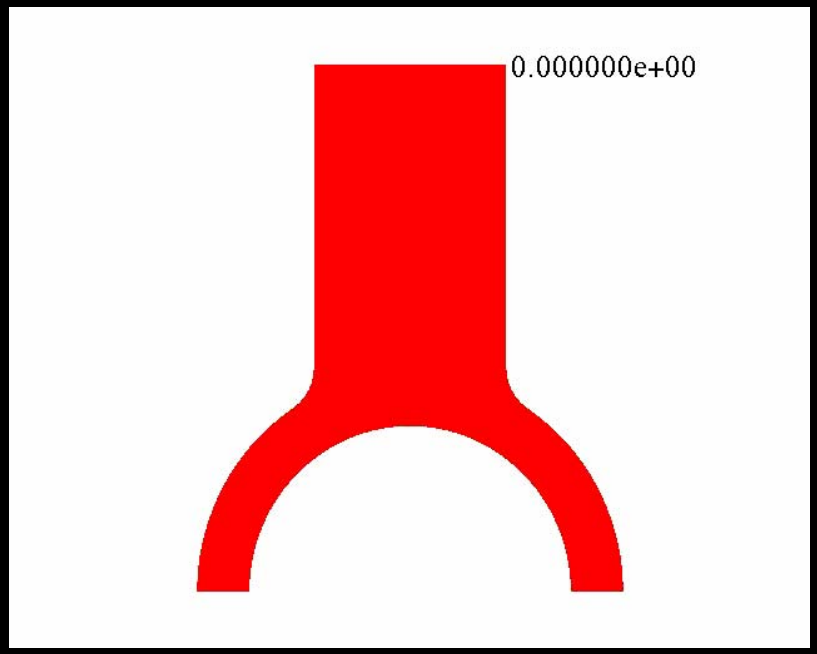



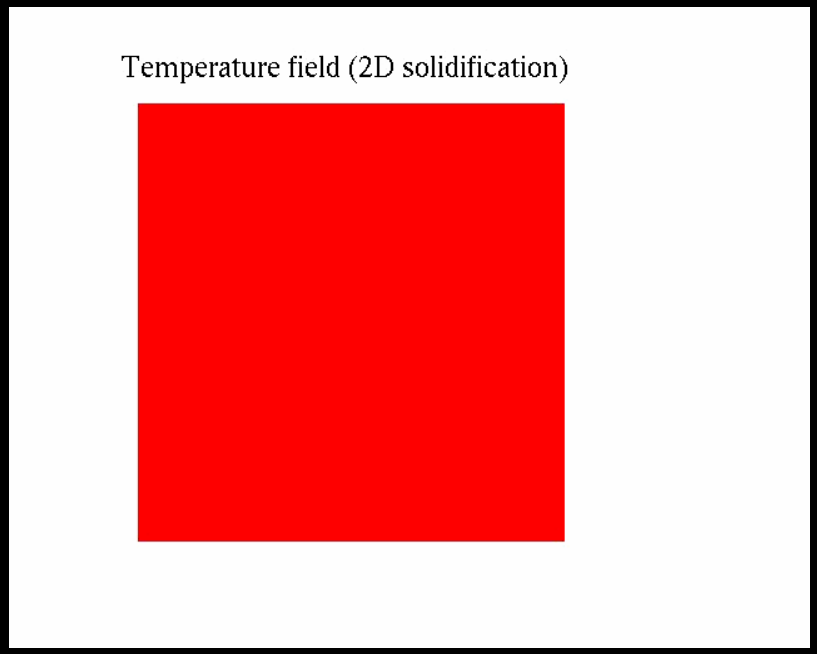

**Micro-structural estimates depend on correct thermal history, yet macroscopic simulations often produce noisy results.**

**This is a result of capturing the phase front rather then tracking it.**

**Solution: more intelligent, "one-sided" fluxes.**

# Phase Change Heat Transfer Examples



# Basic Hemi: Thermocouple Layout

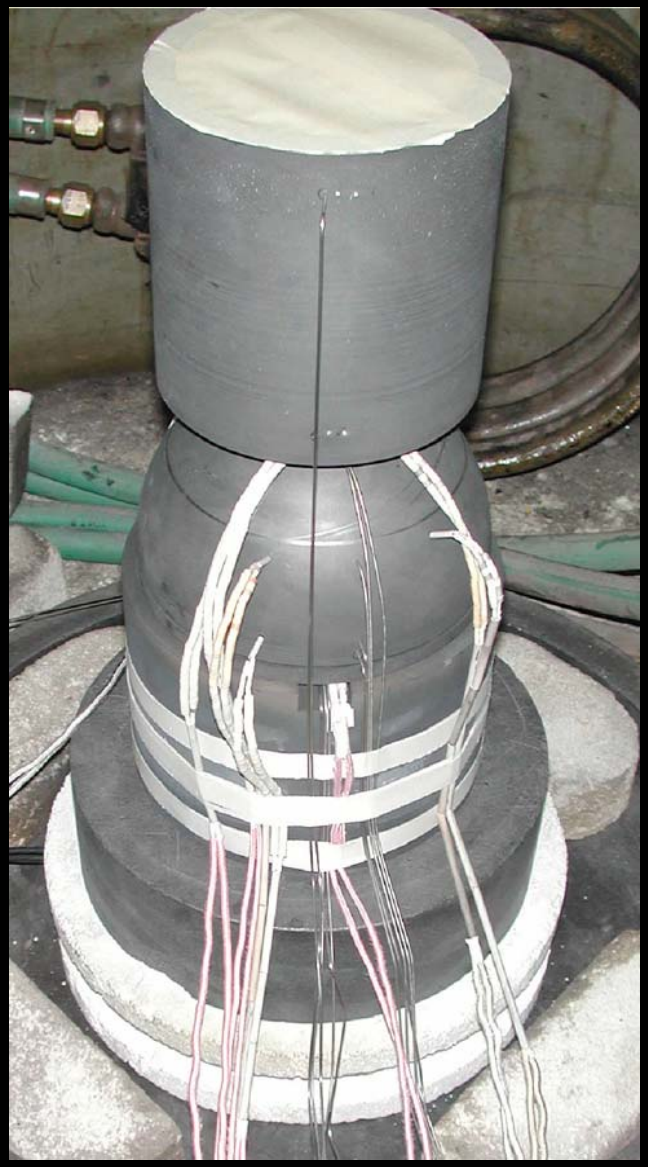
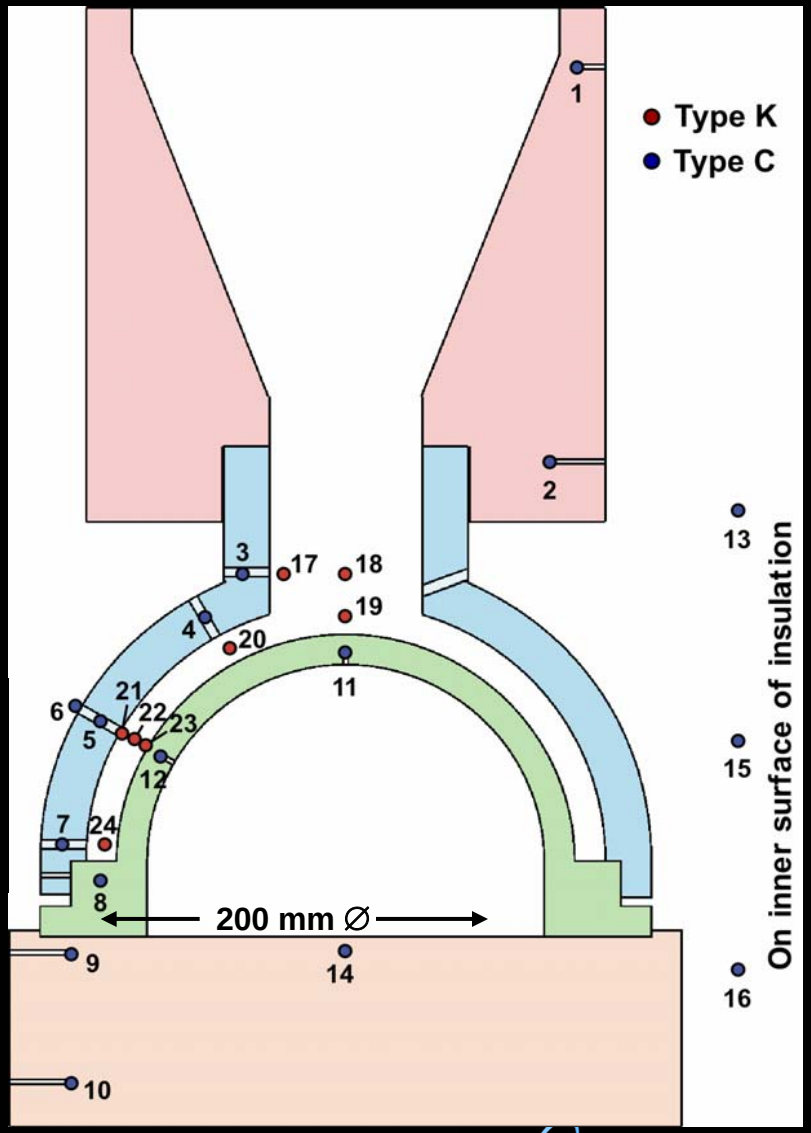
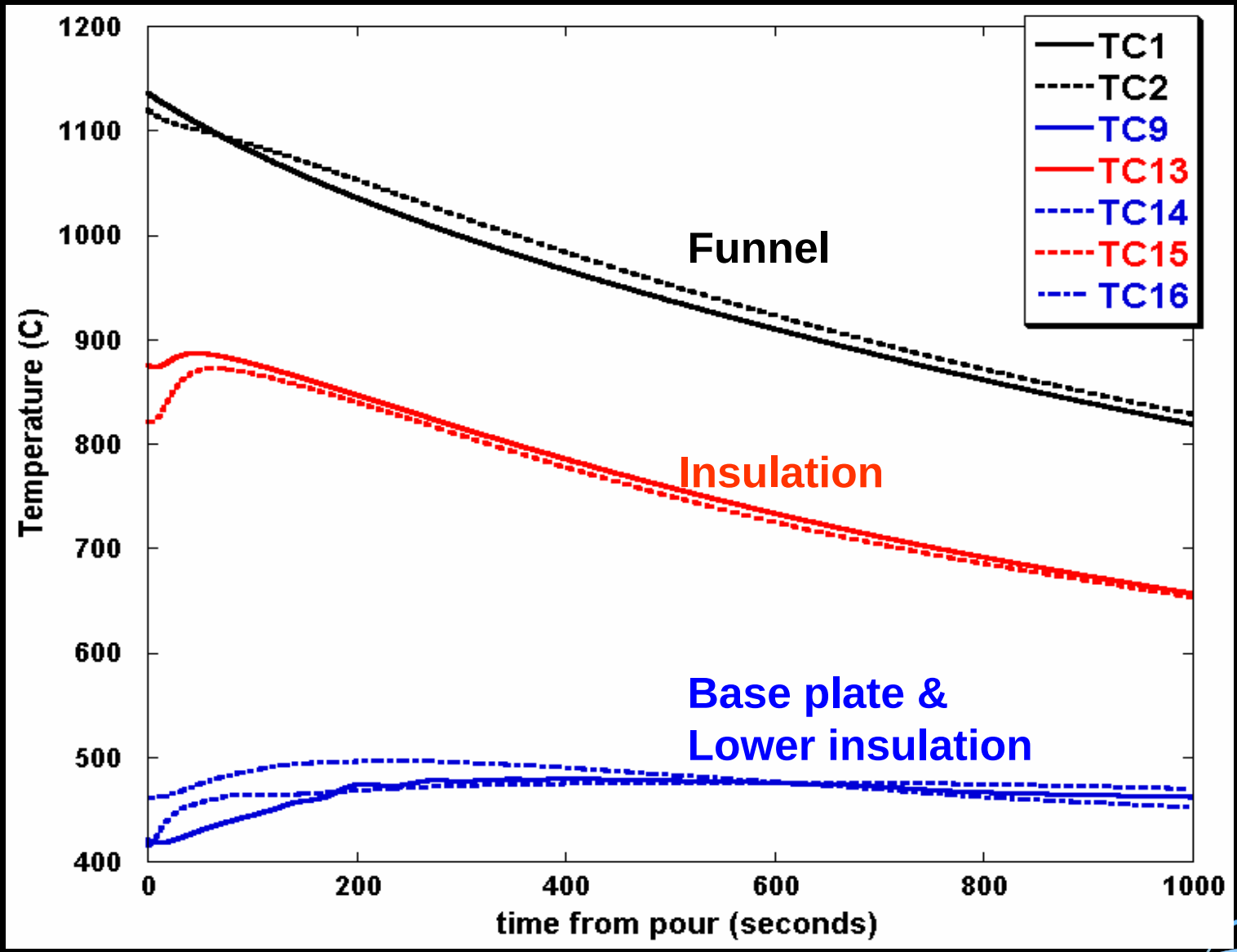


Photo of TC's
17, 18, 19
from the top

Thermocouple positions showing the TCs used to determine the boundary conditions for the model.



# Basic Hemi: BC Thermocouples



# Basic Hemi: Simulation Initial and Boundary Conditions

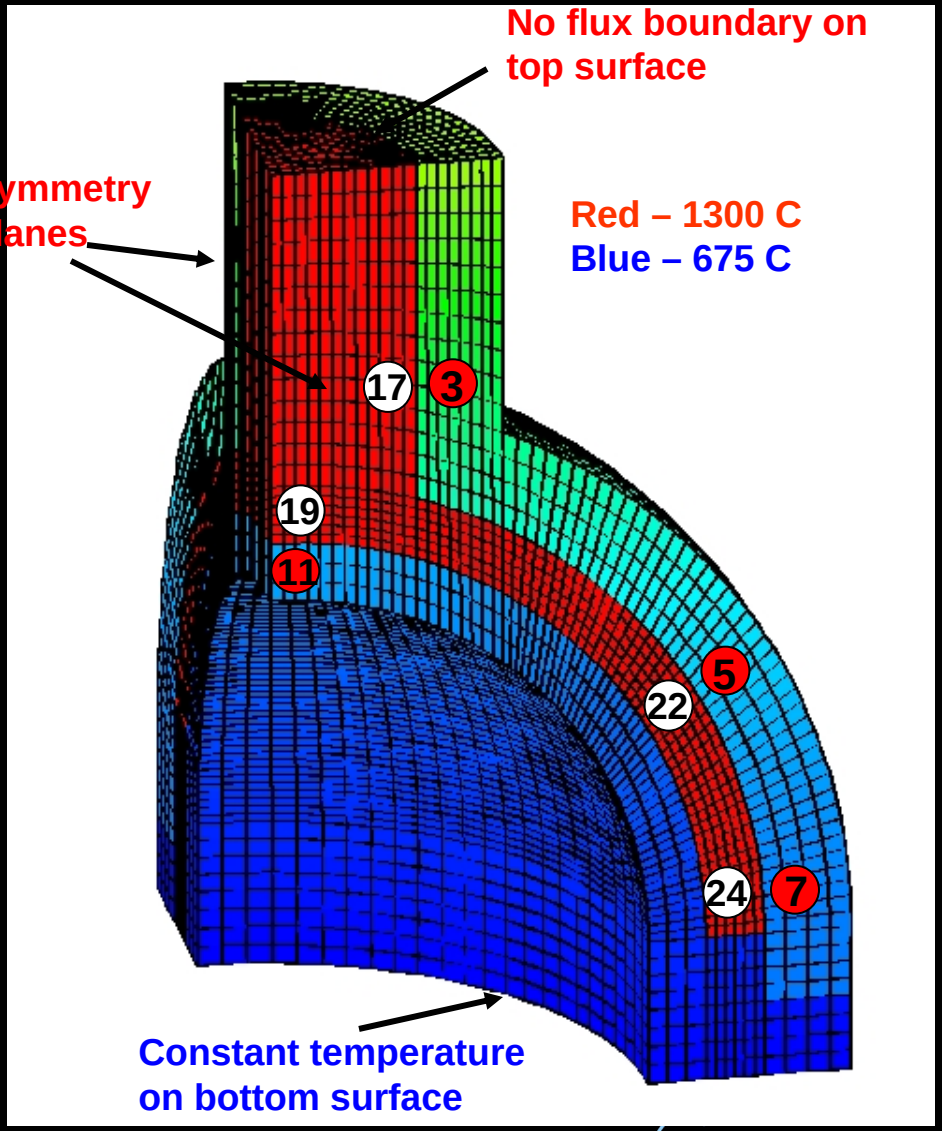
**Initial mold temperature varies to match starting conditions Uranium 1300C**

**Electromagnetics needed to predict starting temperatures**

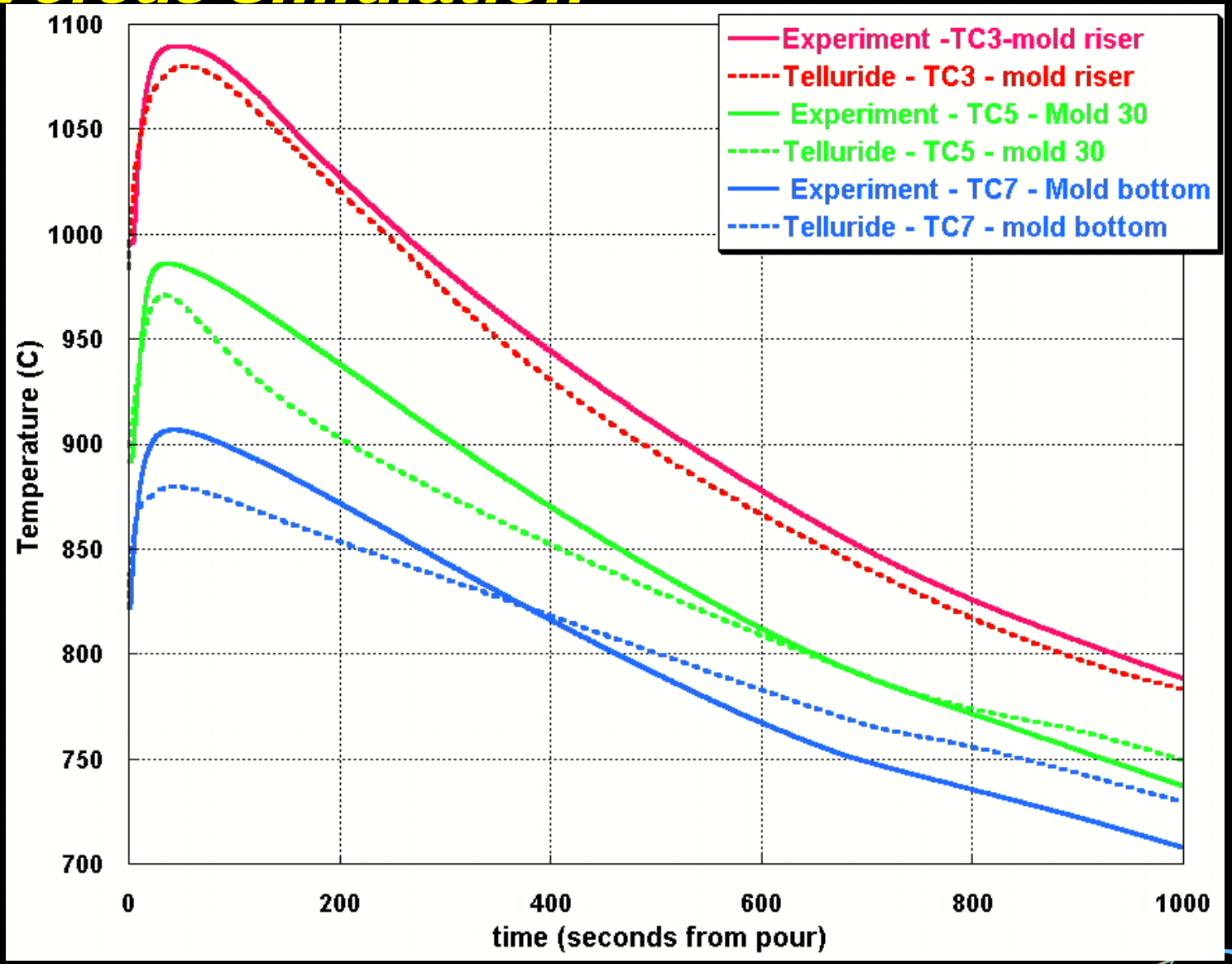
**Heat transfer coefficient: metal and graphite:**
**Varies positionally to match TC's**
**htc= 2000 W/m²K metal – core**
**htc= 1000 W/m²K metal – case**

**Radiation boundary condition:**
**Varies positionally to match TC's from boundary**
**ϵ = 0.2 core ϵ = 0.4 case**

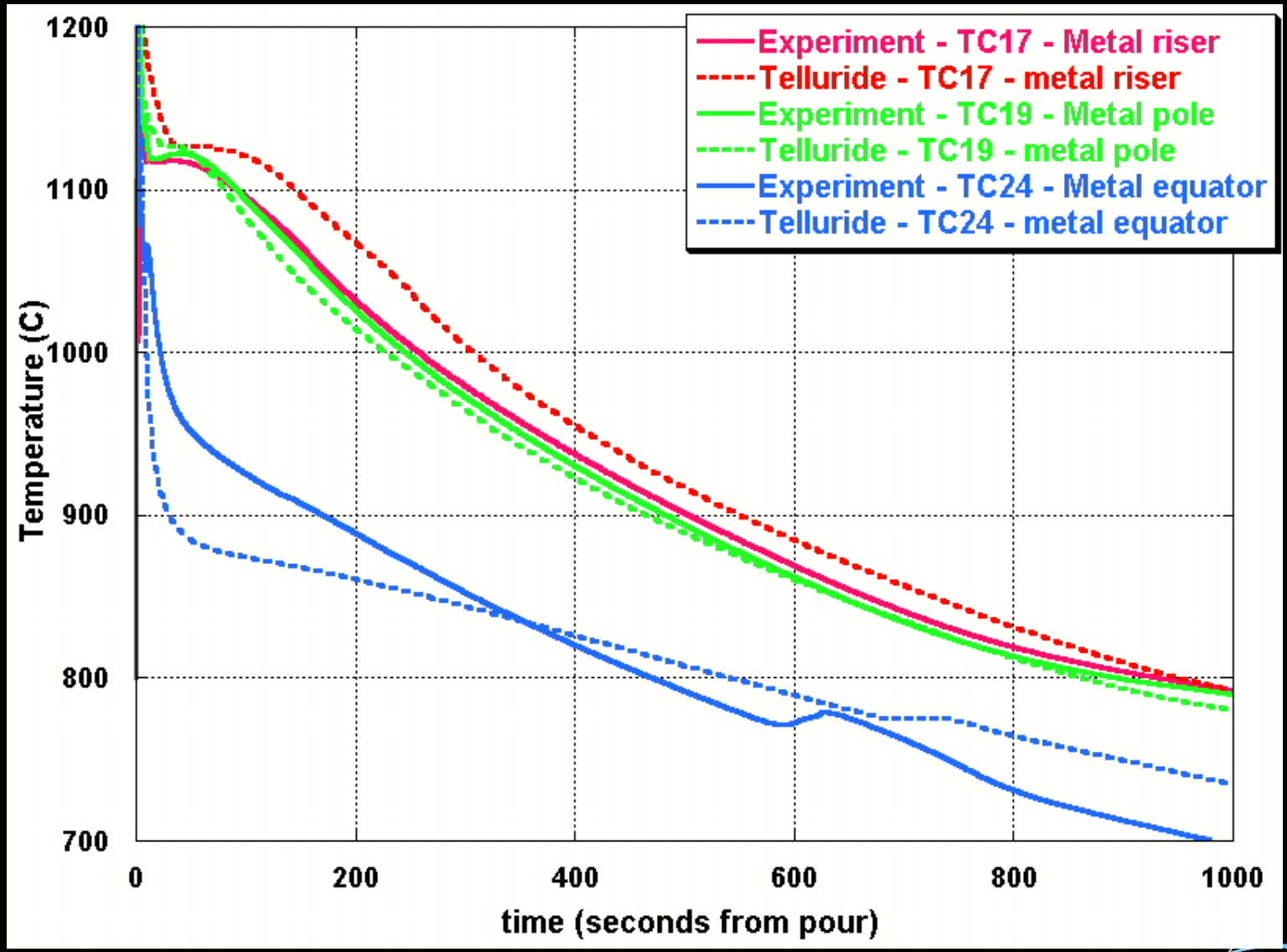
**23489 cells – heat transfer only**
4 processors ES45 – 6.0 hours for 1000 seconds simulation time
Average CPU time/cell/cycle 179.3 μs



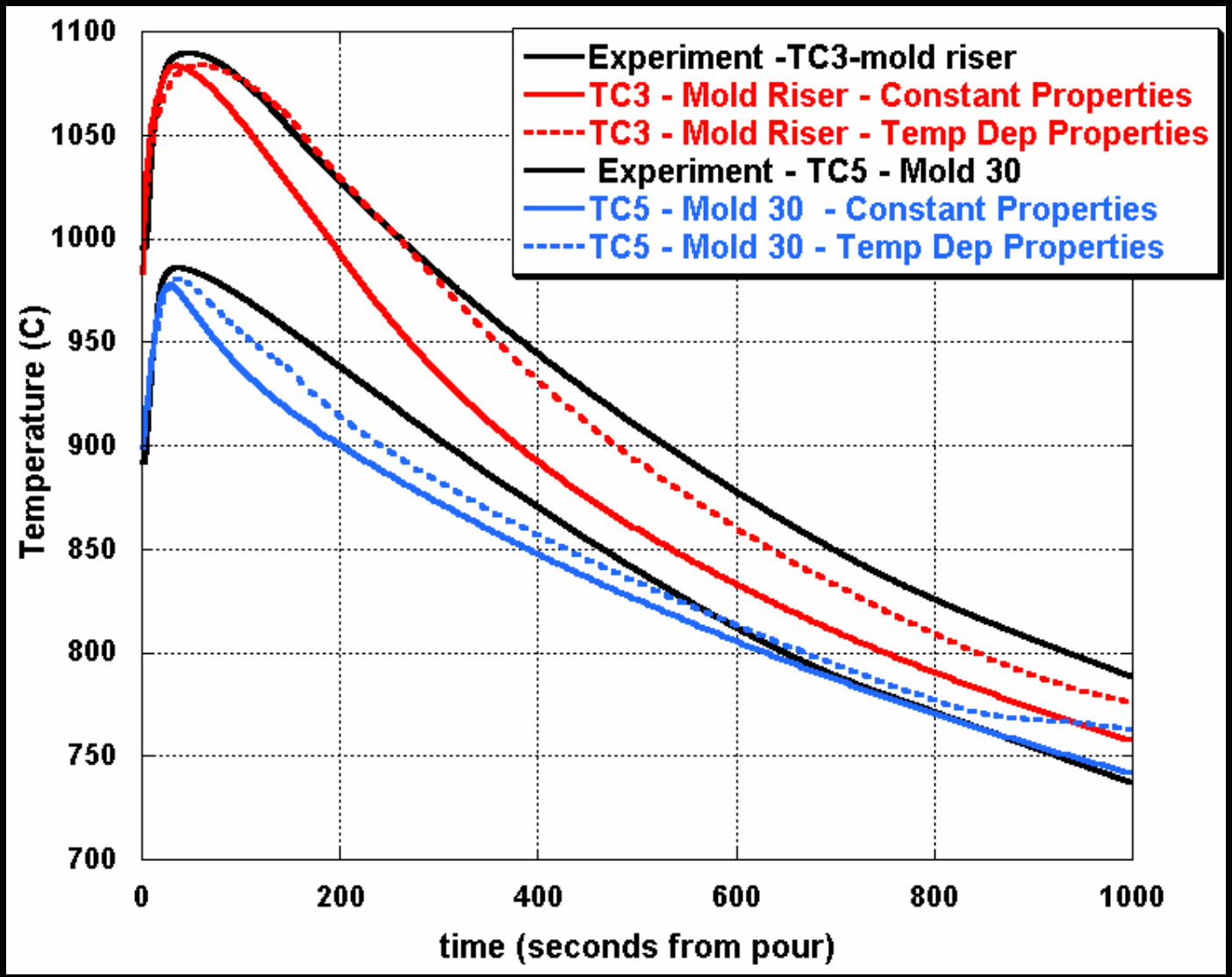
# Basic Hemi: Mold Thermocouples Versus Simulation



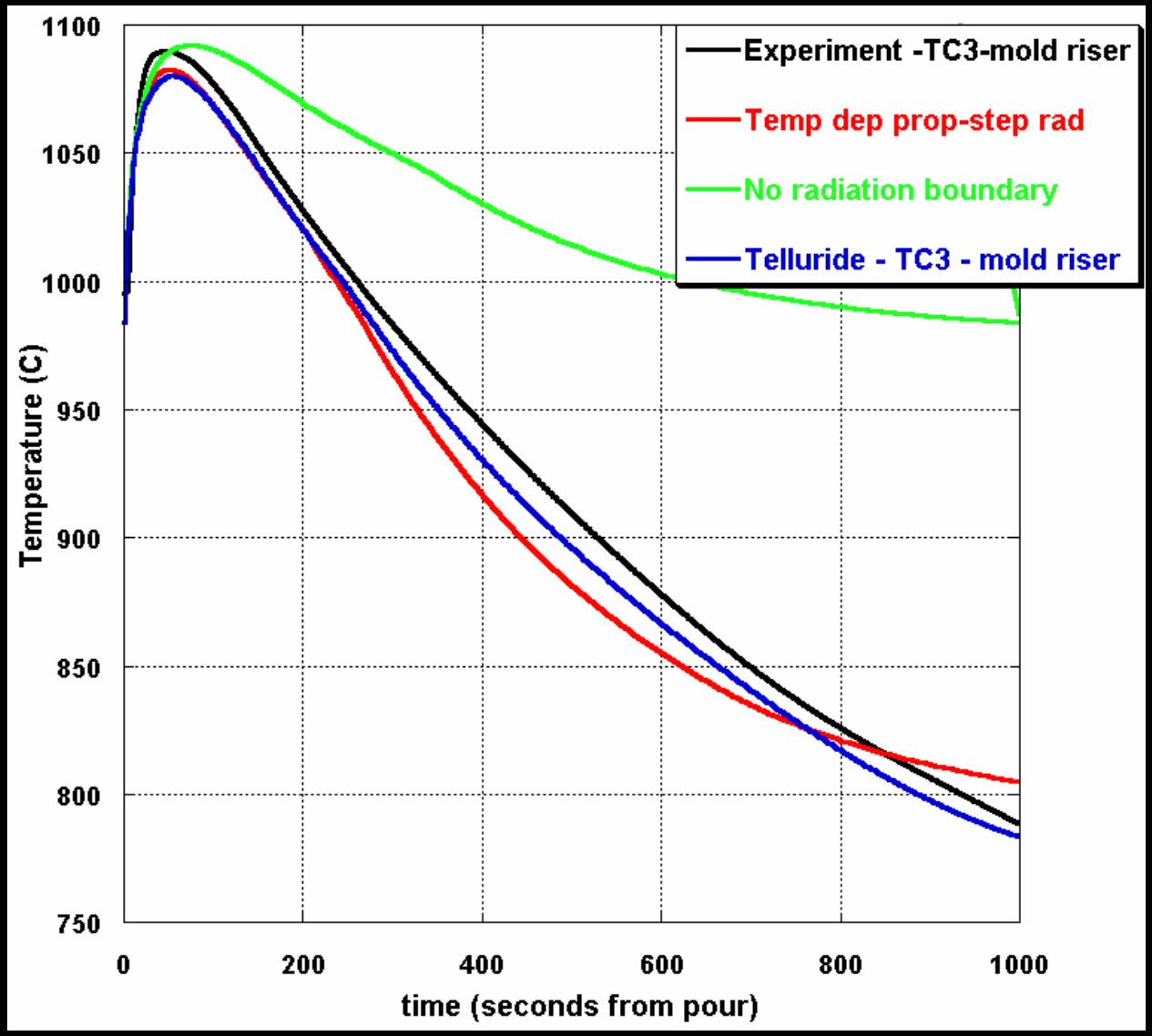
# Basic Hemi: Metal Thermocouples Versus Simulation



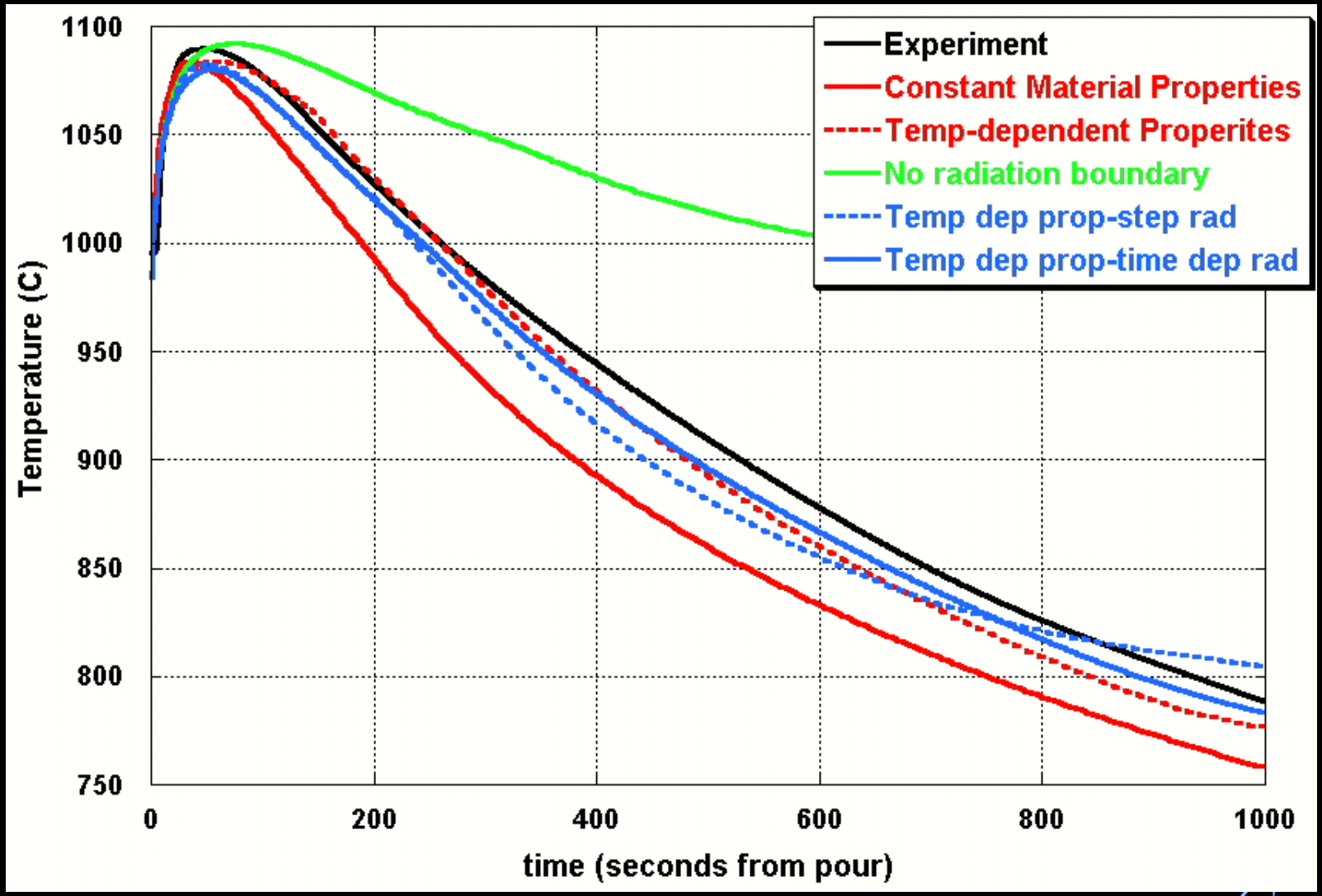
# Basic Hemi: Effect of Temperature Dependent Conductivity on Mold Response



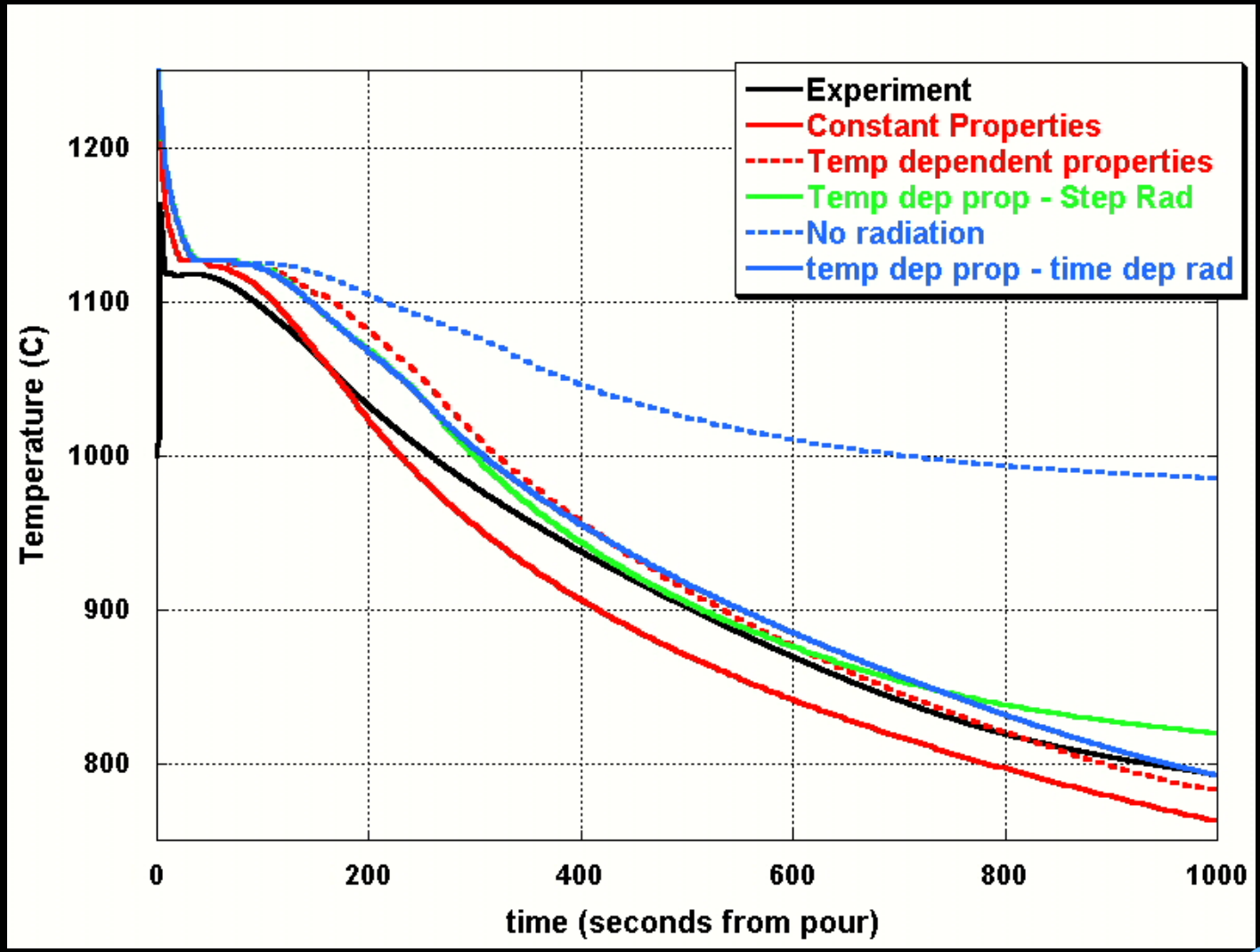
# Basic Hemi: Effect of Radiation BCs on Mold Response



# Basic Hemi: Mold Riser Thermocouple #3



# Basic Hemi: Metal Riser Thermocouple #17



# Physical Model for Solid Material Response: Quasi-static Thermal Elasticity

$$\frac{\partial \sigma_{ij}}{\partial x_j} = 0; \quad \sigma_{ij} = \lambda \varepsilon_{kk} \delta_{ij} + 2\mu \varepsilon_{ij} - (3\lambda + 2\mu)\alpha (T - T_0)\delta_{ij}$$

$$\varepsilon_{ij} = \frac{1}{2}\left( \frac{\partial d_i}{\partial x_j} + \frac{\partial d_j}{\partial x_i} \right)$$

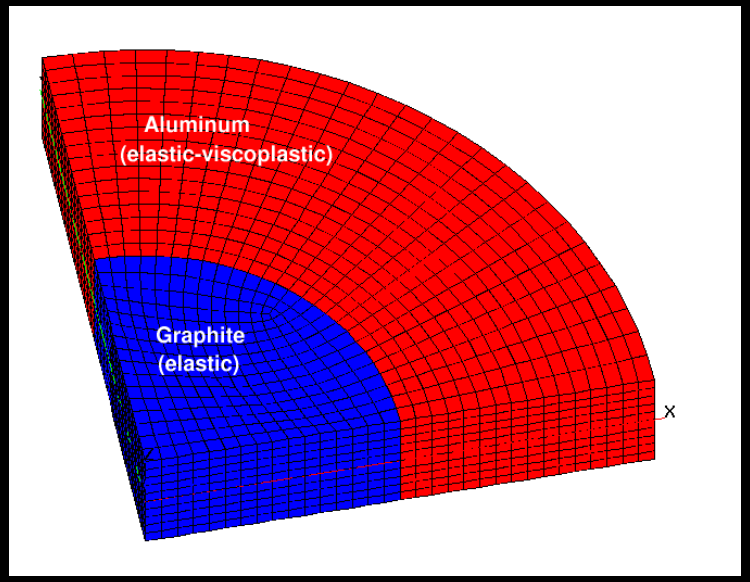
$$\frac{\partial}{\partial x_i}\left( \lambda \frac{\partial d_k}{\partial x_k} \right) + \frac{\partial}{\partial x_j}\left[ \mu \left( \frac{\partial d_i}{\partial x_j} + \frac{\partial d_j}{\partial x_i} \right) \right] - \frac{\partial}{\partial x_i}\left[ (3\lambda + 2\mu)\alpha (T - T_0) \right] = 0$$

- Displacements $d_i$ are assumed to be "small" (Eulerian frame)
- Material constants $\alpha$, $\lambda$, and $\mu$ are in general temperature dependent
- Can be easily expanded to include plasticity, viscoplasticity

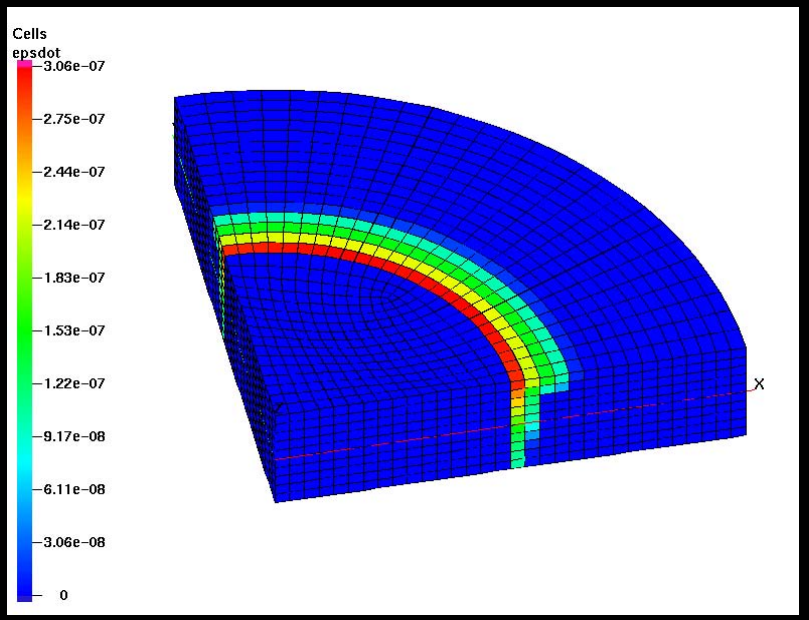
# Elastic-Viscoplastic Solid Mechanics

Cooling an aluminum alloy ring on a graphite plug from 600°K to 298°K

1/8 symmetry, uniform initial temperature, approximately isothermal cooling.



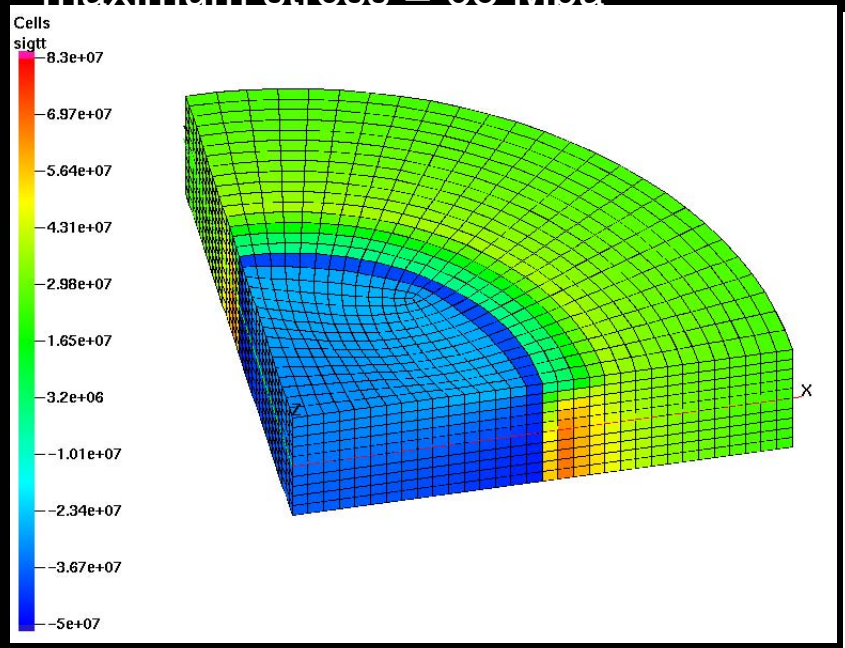

Plastic strain rate plot at 750 sec, T ~ 300°K



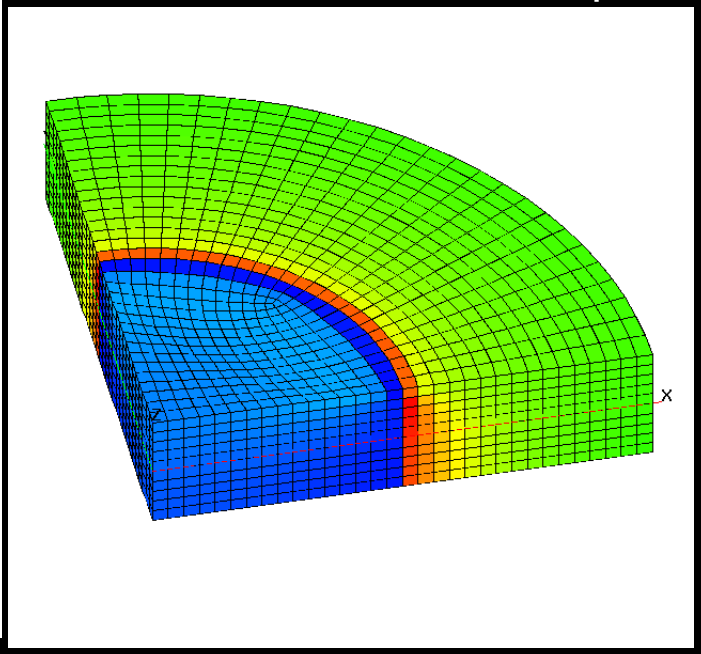
# Elastic-Viscoplastic Solid Mechanics

Comparison of hoop stresses between elastic-viscoplastic and elastic ring.

Viscoplastic -
maximum stress = 68 Mpa




Elastic -
maximum stress = 83 Mpa



# Physical Model for Flow: Incompressible Single-Fluid Navier Stokes

$$\frac{\partial \rho}{\partial t} + \mathbf{u} \bullet \nabla \rho = 0$$

$$\nabla \bullet \mathbf{u} = 0$$

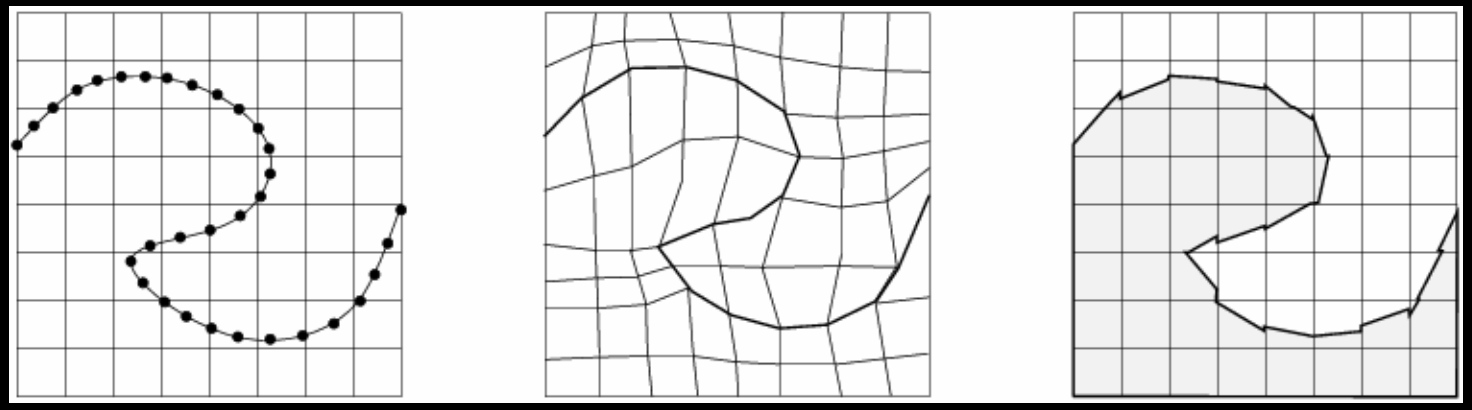
$$\frac{\partial}{\partial t}(\rho \mathbf{u}) + \nabla \bullet (\rho \mathbf{u}\mathbf{u}) - \nabla \bullet \boldsymbol{\tau} - \rho(C,T)\mathbf{g} - (\mathbf{F}_D + \mathbf{F}_s + \mathbf{F}_L) = 0$$

- Newtonian stress tensor $\tau$
- Boussinesq approximation yields $\rho(C,T)$ dependence on gravity force
- Solid phases (k=s) are considered rigid: $u_l = u$; $u_s = 0$
- Drag force $F_D$ for flow retardation in partial solid (mush) regions
- Surface tension force $F_s$ models uneven interfacial attraction effects
- Lorentz force $F_L$ models stirring effects of magnetic fields

# Algorithms: Overview

- **Unstructured mesh domain partitioning**
  - Each cell is a logical cube mapped into degenerate hexahedra: hexes, tets, prisms, pyramids
  - Fixed, Eulerian reference frame (for now)
- **Cell-centered collocated finite volume discretizations**
  - Nearest neighbor connectivity; formally 2nd order in space
  - Embrace weighted least squares for function reconstruction and some matrix operators (Laplacian)
- **Phase change heat transfer**
  - Implicit, nonlinearly-consistent NK-based enthalpy method
  - Locally nonlinear (point) systems for alloy solidification
- **Flow: semi-implicit, fractional step projection method**
  - Moving to fully implicit NK-based scheme as an option
  - Design constraint: complex topology interfacial flows with density ratios of 1 to infinity (void) across interface
- **Thermo-mechanical: node-based CV-FEM approach**
- **Electromagnetics: edge, face-based staggering**

# Interface Representations



Marker Particles  Adaptive Mesh  Volume Tracking

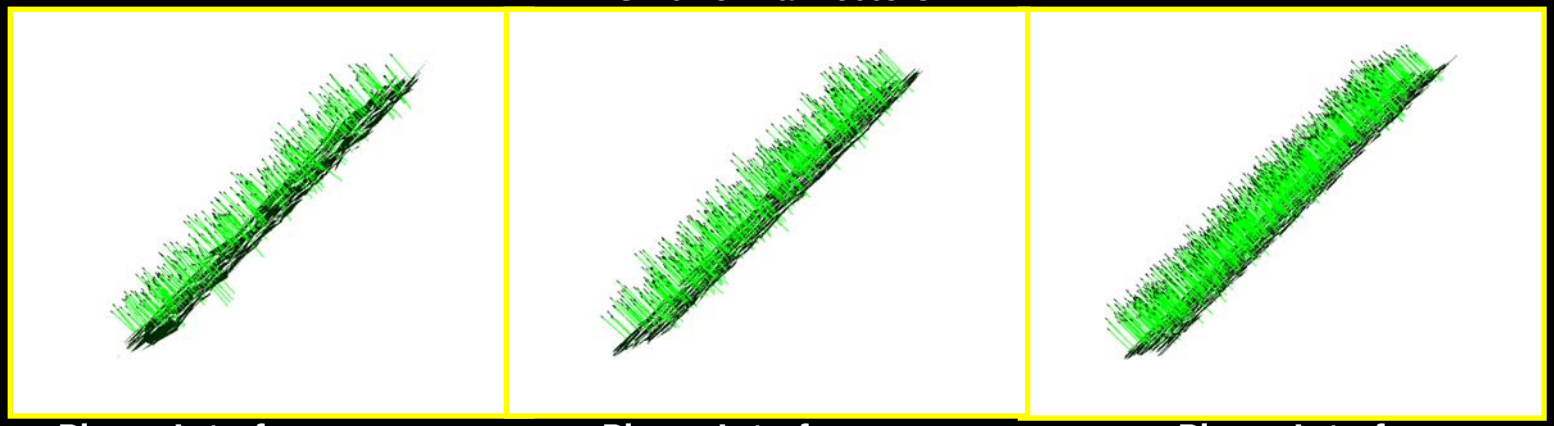
We have experience with many algorithms of interest for interface representation & kinematics

- PIC, SPH, Front tracking, volume tracking, level sets, phase field, continuum advection

Our current algorithm for interface kinematics is based on a volume tracking representation

# Algorithms: Verification of an Interface Representation

Unit normal vectors

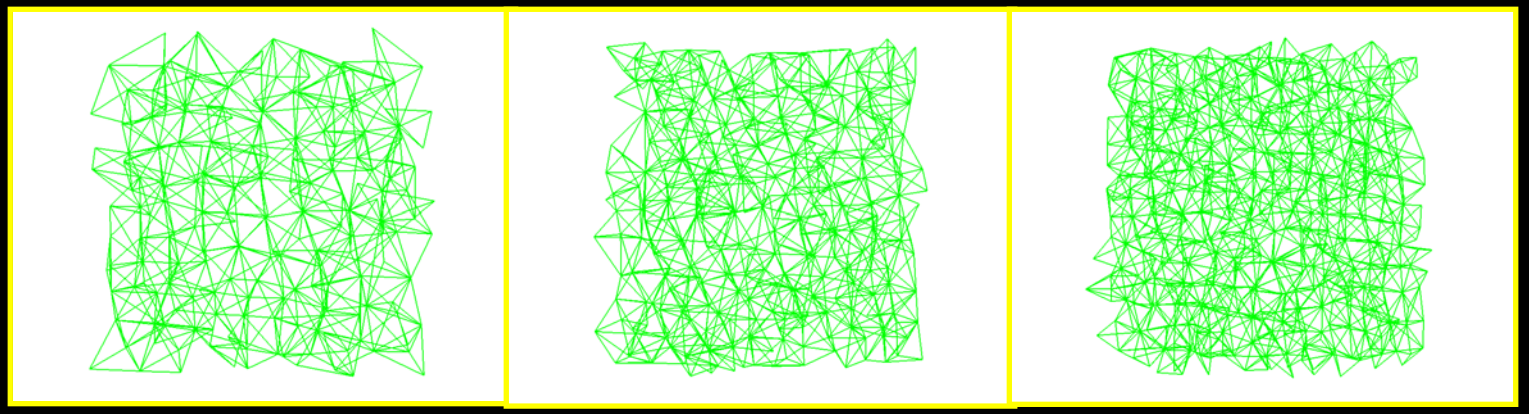


Planar Interface Reconstruction on 7454 tets

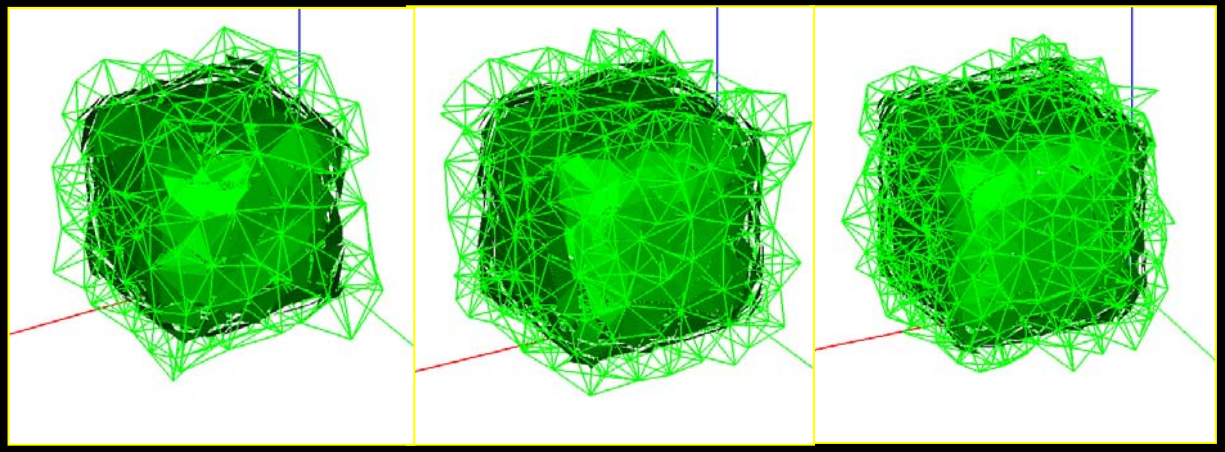
Planar Interface Reconstruction on 14298 tets

Planar Interface Reconstruction on 33153 tets

## Reconstructing Planar Interfaces on Unstructured Tetrahedral Meshes

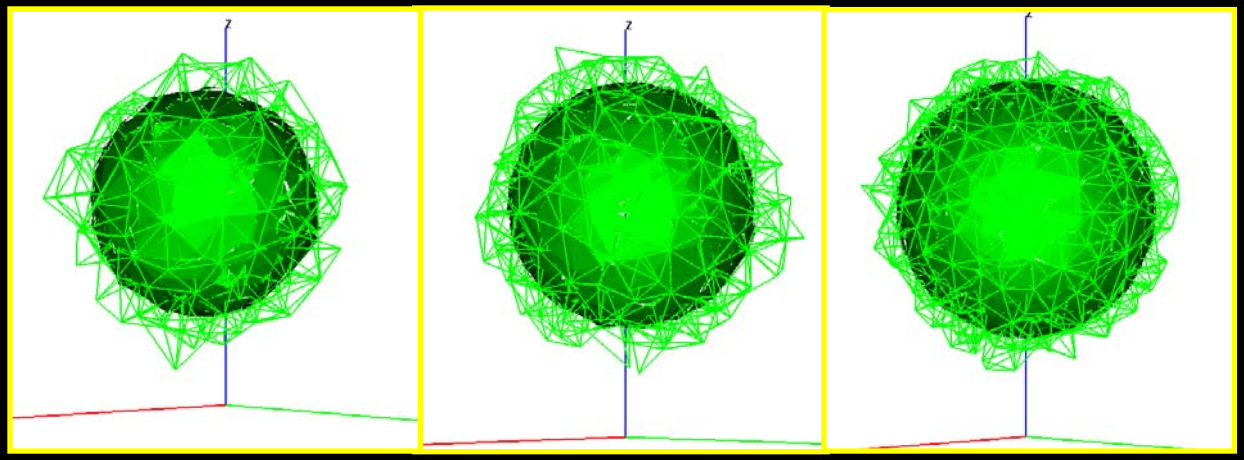


# Algorithms: Verification of an Interface Representation



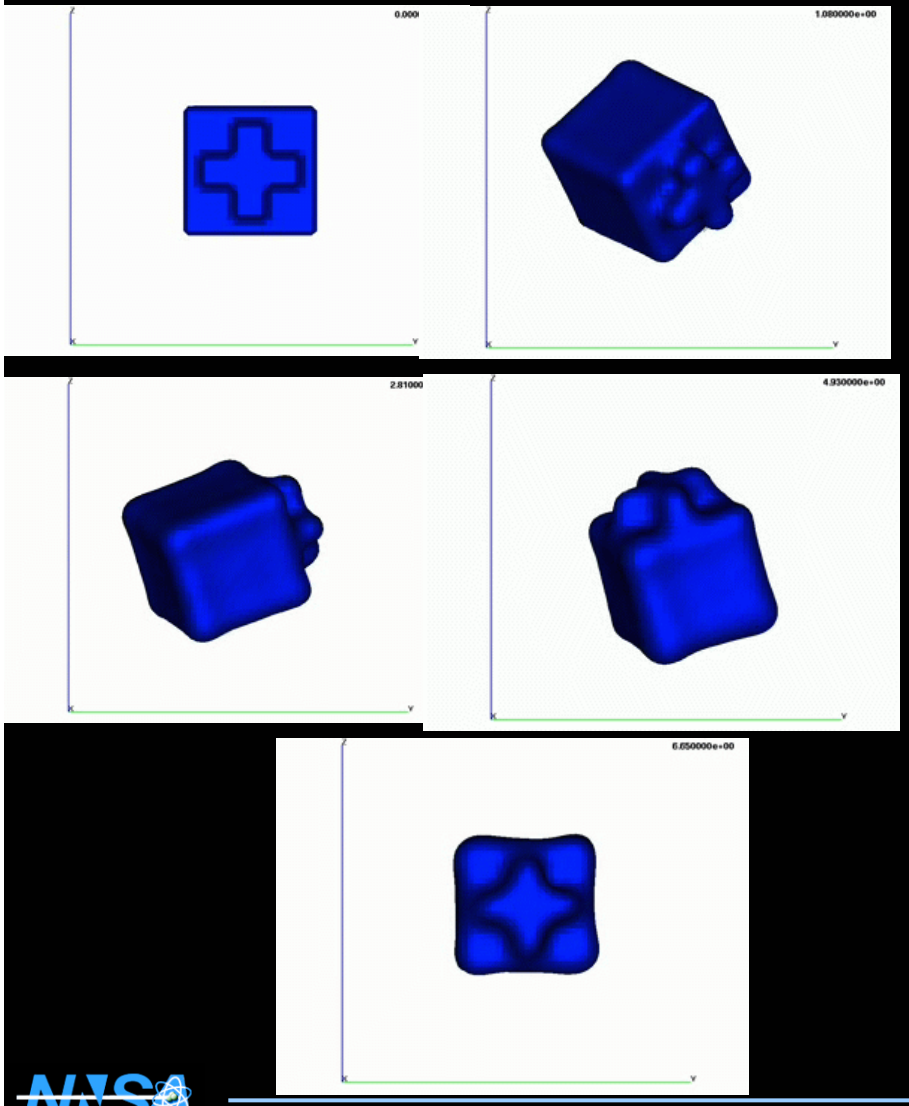
reconstructed interfaces for a cube

7475 tets 14298 tets 33153 tets



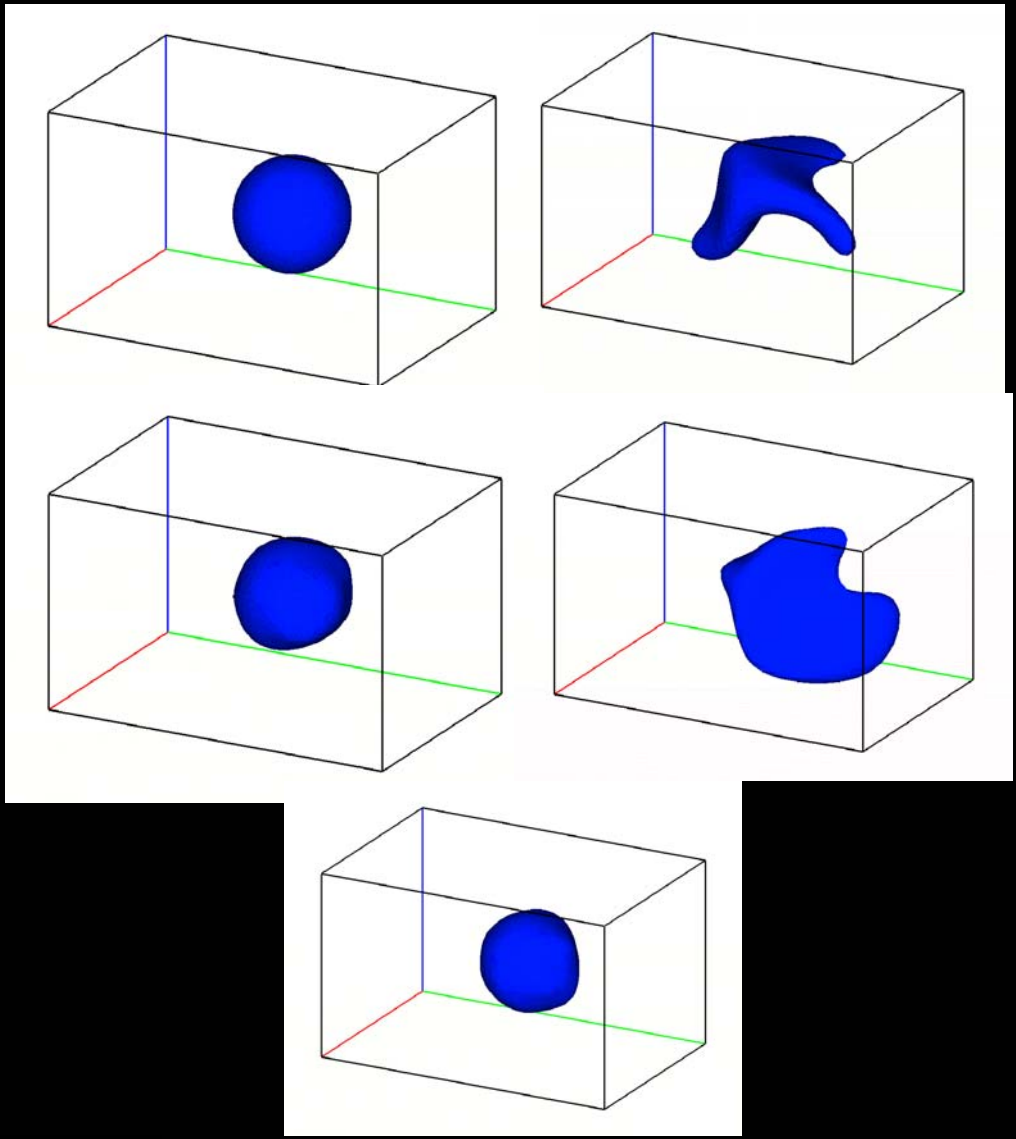
reconstructed interfaces for a sphere

# Volume Tracking Benchmarks: The Rotating Notch



- A standard volume tracking benchmark problem
- Independent of any flow alignment with the mesh
- Demonstrates tracking of regions with high (infinite) curvature
- 40x40x40 mesh with the unit normal to the interface determined by the convolution method

# Algorithms: Interface Kinematics Verification



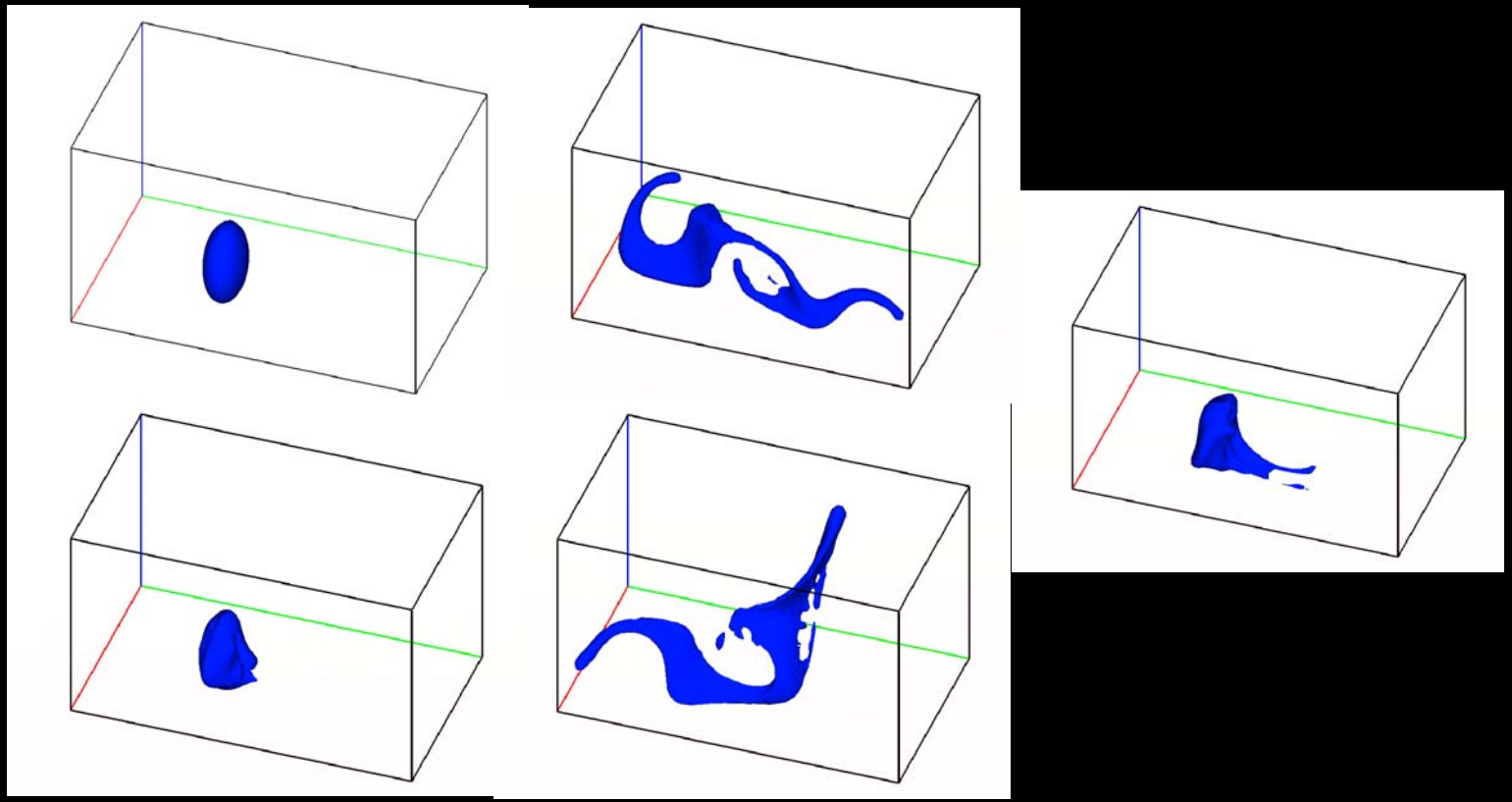
A spherical drop is placed into a velocity field which deforms the drop from initial configuration.

The velocity field is sinusoidal in time so the drop should return to its original shape at one half of a time period and again after one full time period.

Large distortions of the drop after one time period signify a poor algorithm for interface kinematics.

Quantitative error analysis possible at each time period.

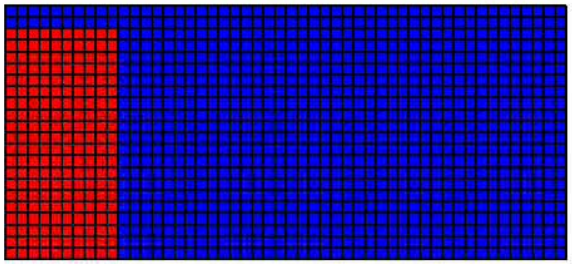
# Algorithms: Interface Kinematics Verification

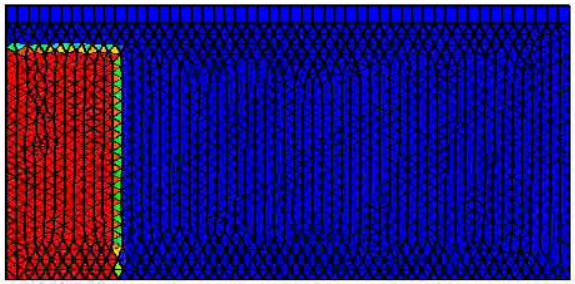


**Volume tracking tests that include tearing and breakup of fluid are probably the most demanding benchmarks.**

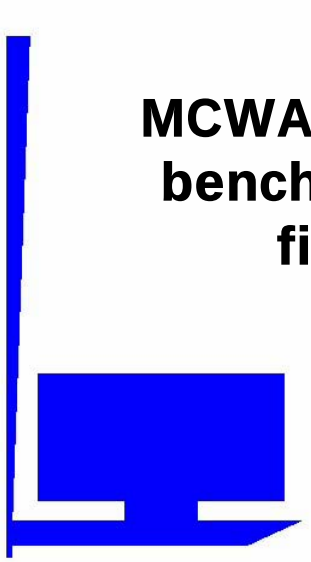
**The ellipsoid drop is placed in a deformation field that is sinusoidal in time. The drop should return to its original configuration after one full time period.**

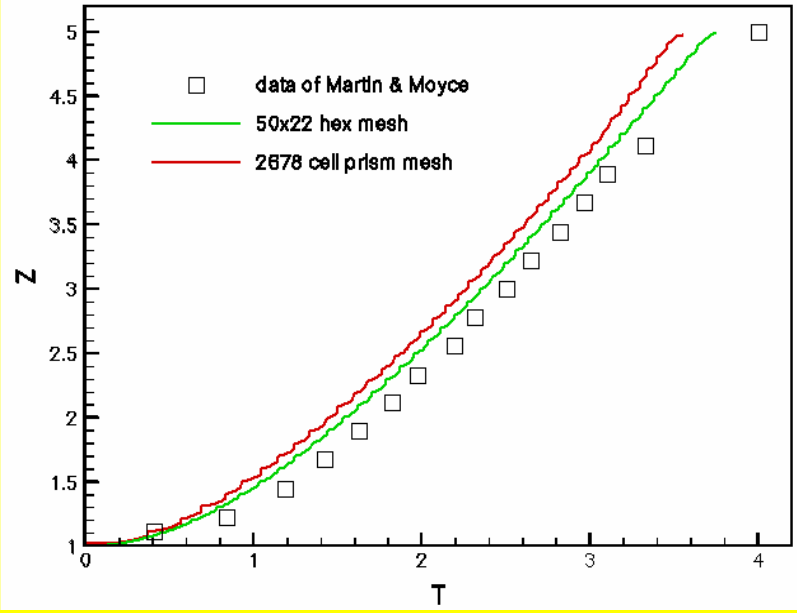
# Mold Filling Validation





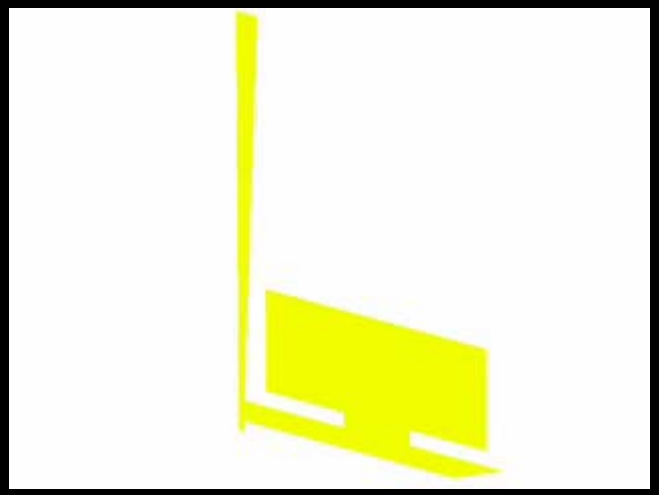
**Broken dam on quads and triangles**

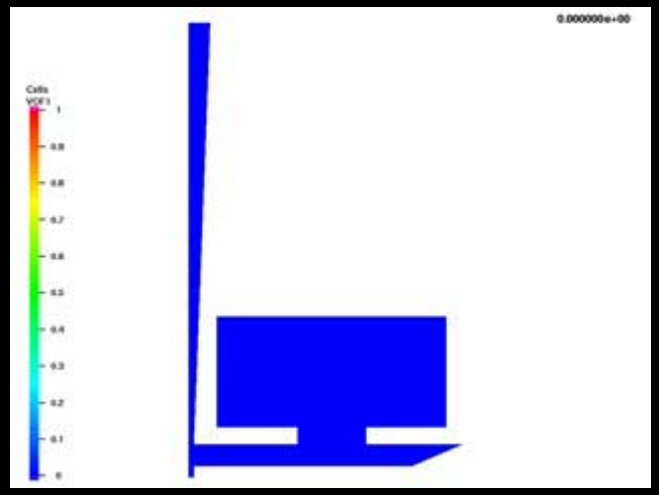


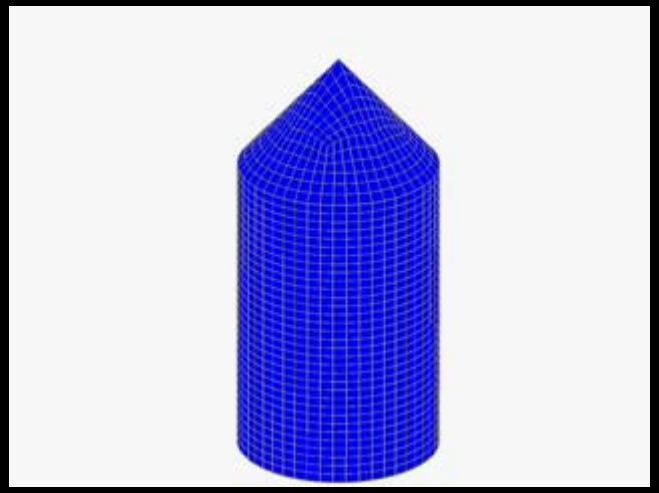

**aluminum/air density ratio – 2000:1**

# MCWASP '98: Benchmark Mold Filling

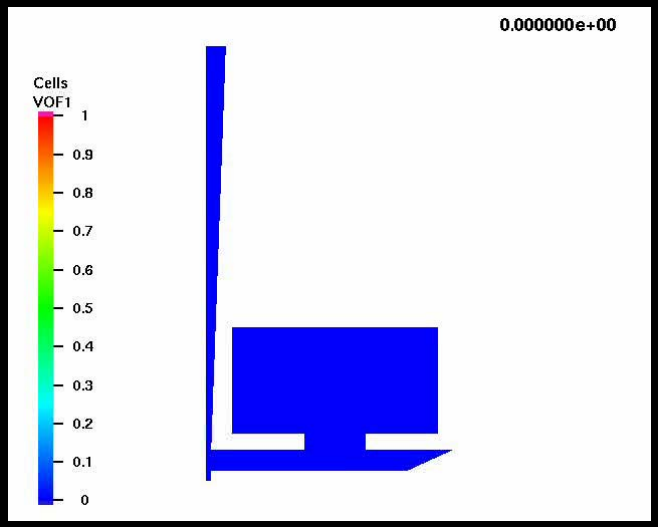





Cavity modeled as fluid with 2000x less density



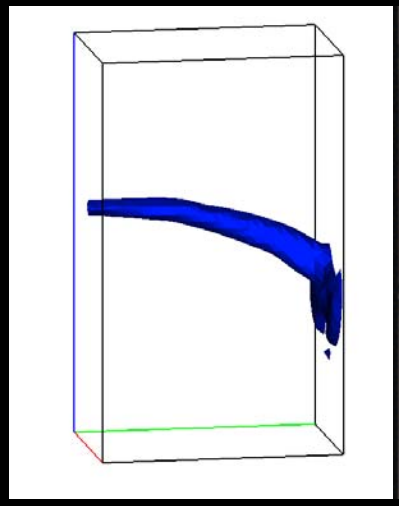
LANL cylindrical fill experiment

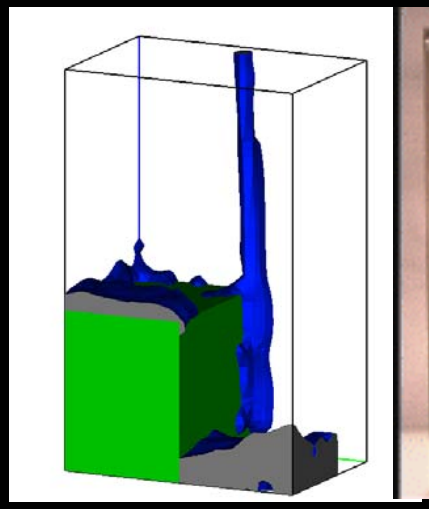



Cavity modeled as void

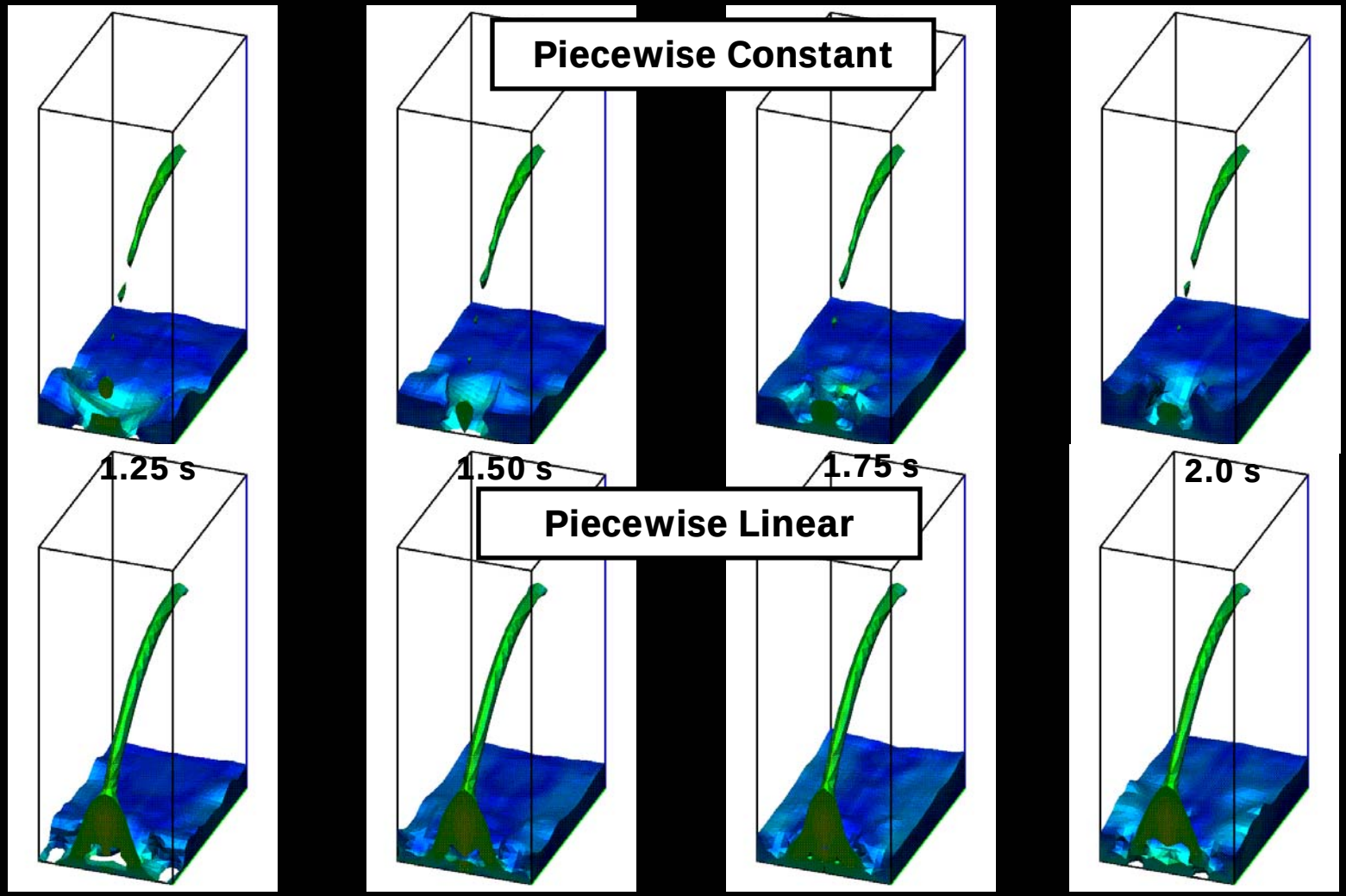
# Mold Filling Validation

- Sigma foundry transparent molds
  - Mold: simple Plexiglas box with various injection ports
  - High-speed digital video of free surface topology during fill
- Two fluid/fluid systems tested
  - Water/Air: High density ratio, energetic flow (Re ↑) with negligible surface tension effects (We ↑)
  - Gallium/Water: Low density ratio, laminar flow (Re ↓) with high surface tension effects (We ↓)

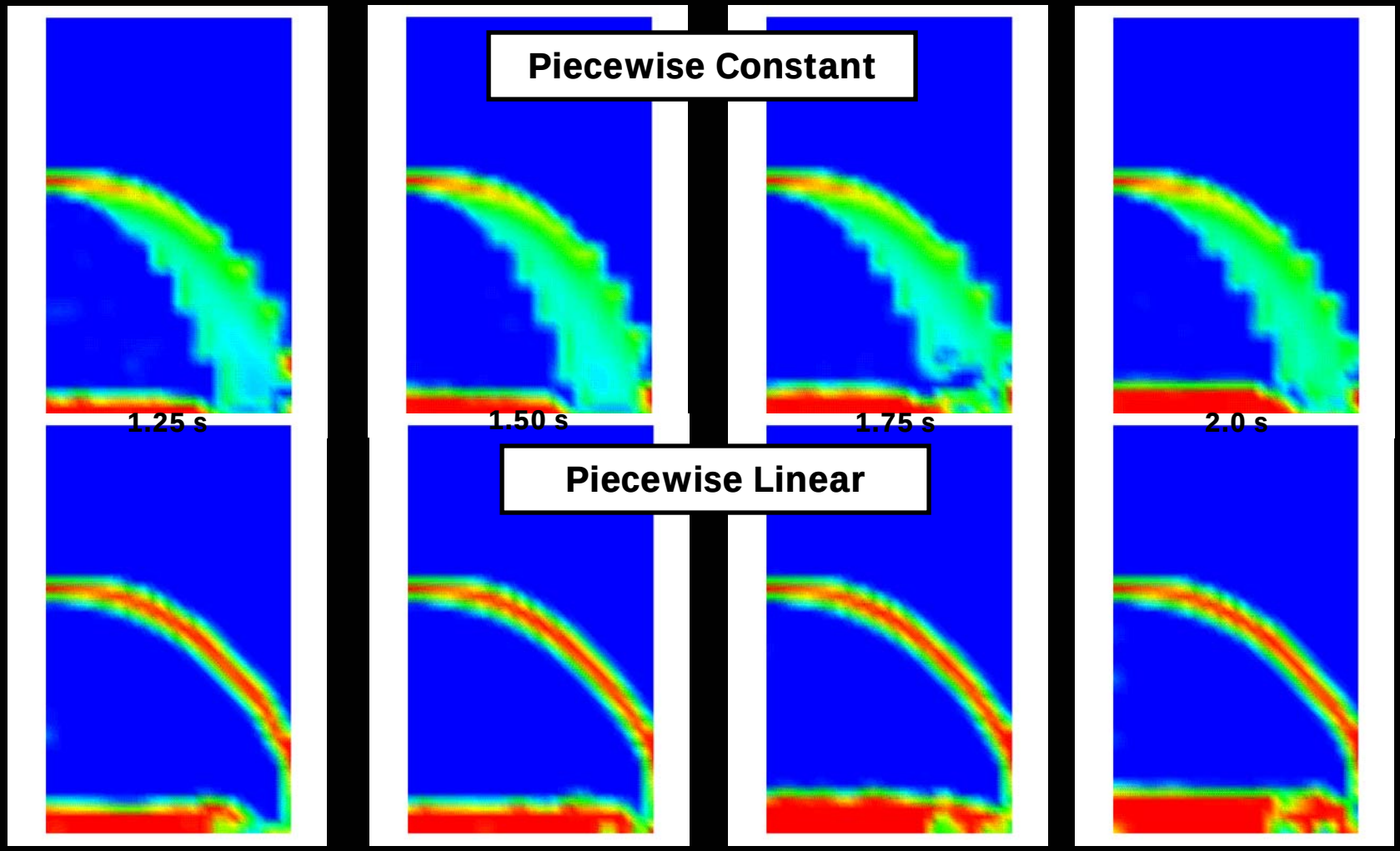




# Plexiglas Box Fill Experiment



# Plexiglas Box Fill Experiment



# The Basic Projection Flow Algorithm

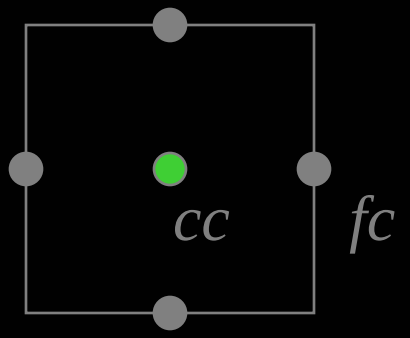
- **Predictor Step**

$$\rho^{n+1}\vec{u}^{*}_{cc} = \rho^{n}\vec{u}^{n}_{cc} + \delta t\left[-\rho^{n+1}_{cc}\left\langle \frac{\nabla P^{n}}{\rho^{n}_{fc}} - \frac{\vec{F}^{\,n}_{s}}{\rho^{n}_{fc}}\right\rangle_{cc}\right]$$

- **Velocity transfer to face**

$$\vec{u}^{*}_{fc} = \Lambda_{fc}\left(\vec{u}^{*}_{cc} + \delta t\left\langle \frac{\nabla P^{n}}{\rho^{n}_{fc}} - \frac{\vec{F}^{\,n}_{s}}{\rho^{n}_{fc}}\right\rangle_{cc}\right) - \delta t\left(\frac{\nabla P^{n}}{\rho^{n+1}_{fc}} - \frac{\vec{F}^{\,n}_{s}}{\rho^{n+1}_{fcc}}\right)$$

  - Rhie-Chow (1983) like procedure




**interpolation operators**

$$\left\langle \phi_{fc}\right\rangle_{cc} \quad \Lambda_{fc}\left(\phi_{cc}\right)$$

- **Poisson equation for pressure correction**

$$\nabla.\left(\frac{\nabla \delta P^{n+1}_{cc}}{\rho^{n+1}}\right) = \nabla.\left(\frac{\vec{u}^{*}_{fc}}{\delta t} + \frac{\delta \vec{F}^{\,n+1}_{s}}{\rho^{n+1}}\right)$$

$$\vec{u}^{n+1}_{fc} = \vec{u}^{*}_{fc} - \delta t\left(\frac{\nabla \delta P^{n+1}_{fc} + \delta \vec{F}^{\,n+1}_{s}}{\rho^{n+1}_{fc}}\right) \qquad \vec{u}^{n+1}_{cc} = \vec{u}^{*}_{cc} - \delta t\left\langle \frac{\nabla \delta P^{n+1}_{fc} + \delta \vec{F}^{\,n+1}_{s}}{\rho^{n+1}_{fc}}\right\rangle_{cc}$$

# Momentum Advection: Consistency with Interface Tracker is Essential

- Second order approximation for cells far from the interface, where the cell density does not change
- Near the interface, we use the mass fluxes calculated by the volume tracker to define momentum fluxes

**Old: inconsistent**

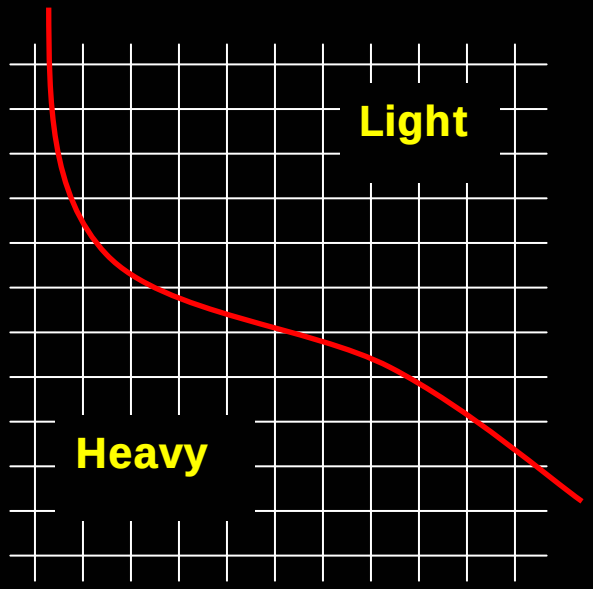
$$\delta t \int \nabla \cdot (\rho \boldsymbol{u}\boldsymbol{u})^n \, dV \approx \rho^{n+1/2} \sum_f \delta t \boldsymbol{A}_f \cdot \boldsymbol{u}_{fs}^n \left\langle \boldsymbol{u} \right\rangle_{f,upwind}^n$$

$$= \rho^{n+1/2} \sum_f \delta V_f \left\langle \boldsymbol{u} \right\rangle_{f,upwind}^n$$

**New: consistent**

$$\delta t \int \nabla \cdot (\rho \boldsymbol{u}\boldsymbol{u})^n \, dV \approx \sum_f \delta t \boldsymbol{A}_f \cdot \boldsymbol{u}_{fs}^n \left\langle \rho \boldsymbol{u} \right\rangle_{f,upwind}^n$$

$$= \sum_f \sum_k \delta V_f \left[ f_{k,f} \rho_k^n \left\langle \boldsymbol{u} \right\rangle_{f,upwind}^n \right]$$



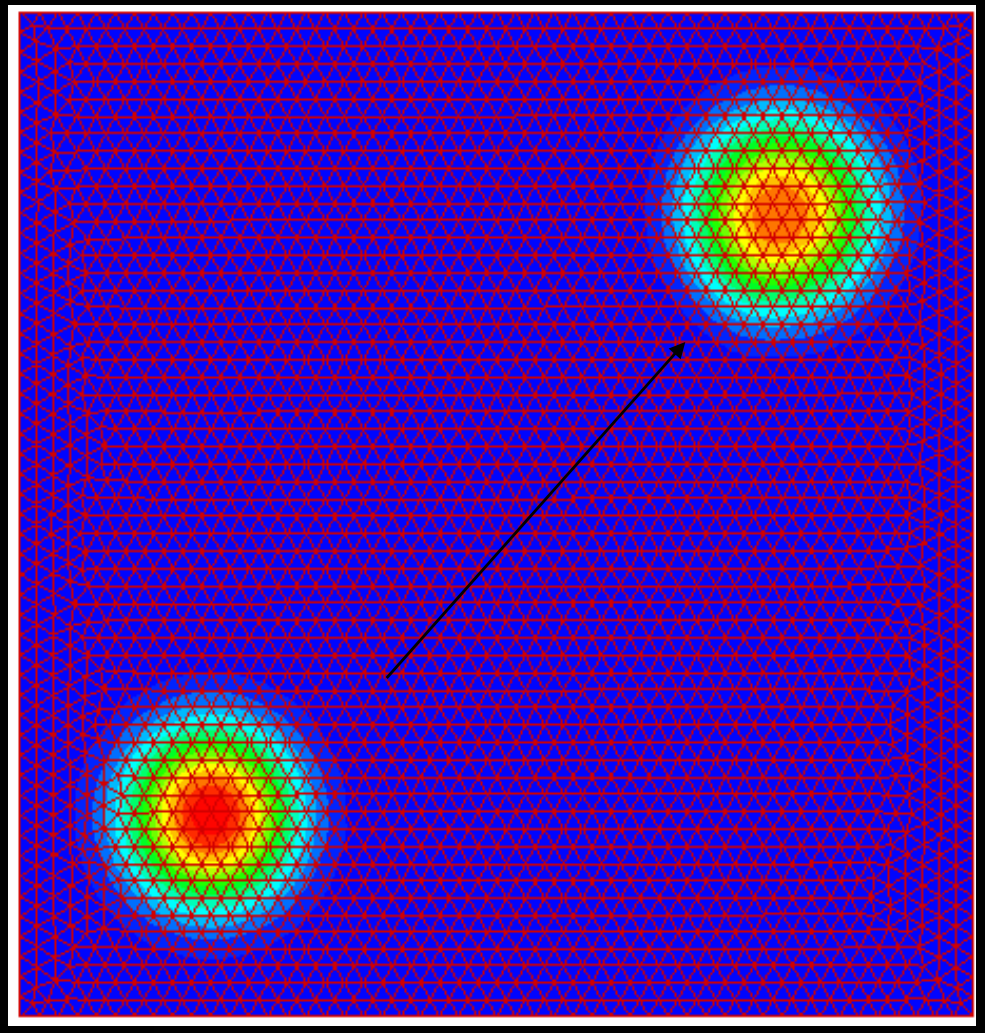
# Advection: Consistency with Interface Tracker is Essential for any Variable

- Second order approximation for cells far from the interface, where the cell density does not change
- Near the interface, we use the mass fluxes calculated by the volume tracker to define *all* advective fluxes

$$\delta t \int \nabla \cdot (\rho \mathbf{u} \phi)^n \, dV \approx \sum_f \delta t \mathbf{A}_f \cdot \mathbf{u}_{fs}^n \langle \rho \phi \rangle_f^n$$
$$= \sum_f \sum_k \delta V_f \, f_{k,f} \rho_k^n \langle \phi \rangle_{k,f}^n$$

- $< \Phi >$ = upwind value of $\Phi$
- Examples for $\Phi$: enthalpy, species concentration

# A Simple Advection Test Problem



A simple test problem demonstrates many of the shortfalls of typical advection implementations

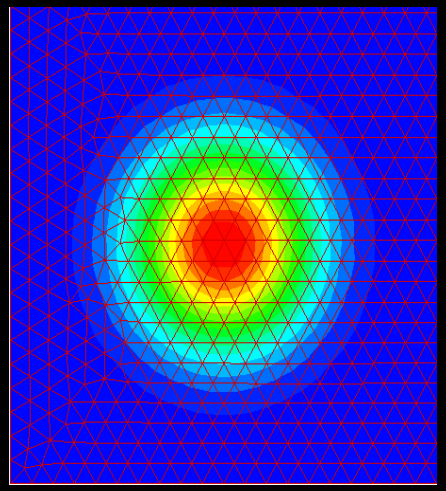
Advection of a concentration plume in a diagonal velocity field. Move the drop with a uniform velocity field from the lower left corner to the upper right.

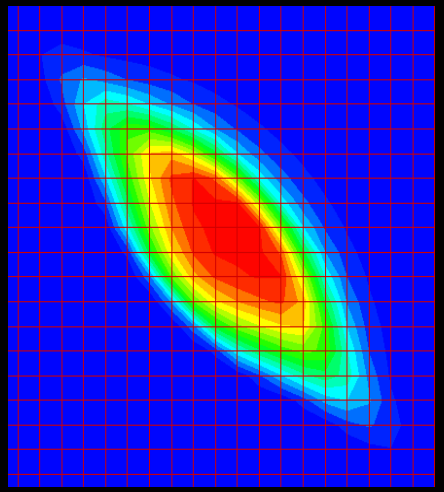
Does the drop shape change after translation? It should NOT!

Exact advection should move a Gaussian plume of material without altering shape or peak magnitude.

If it does, the errors are due solely to the numerical algorithm

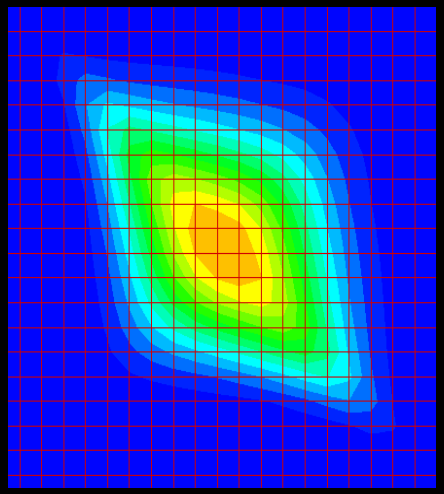
# Example Test Results

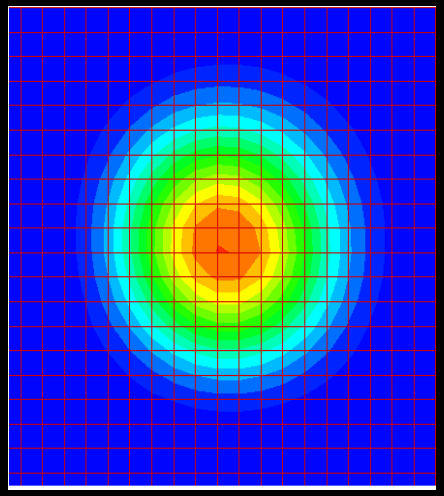


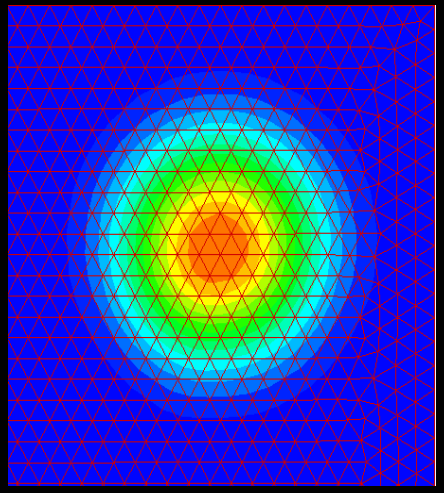


Close up view of the final distribution. Notice the “pancaking” for what is usually claimed as second order. Compare to the latest high order implementation on structured and unstructured grids.

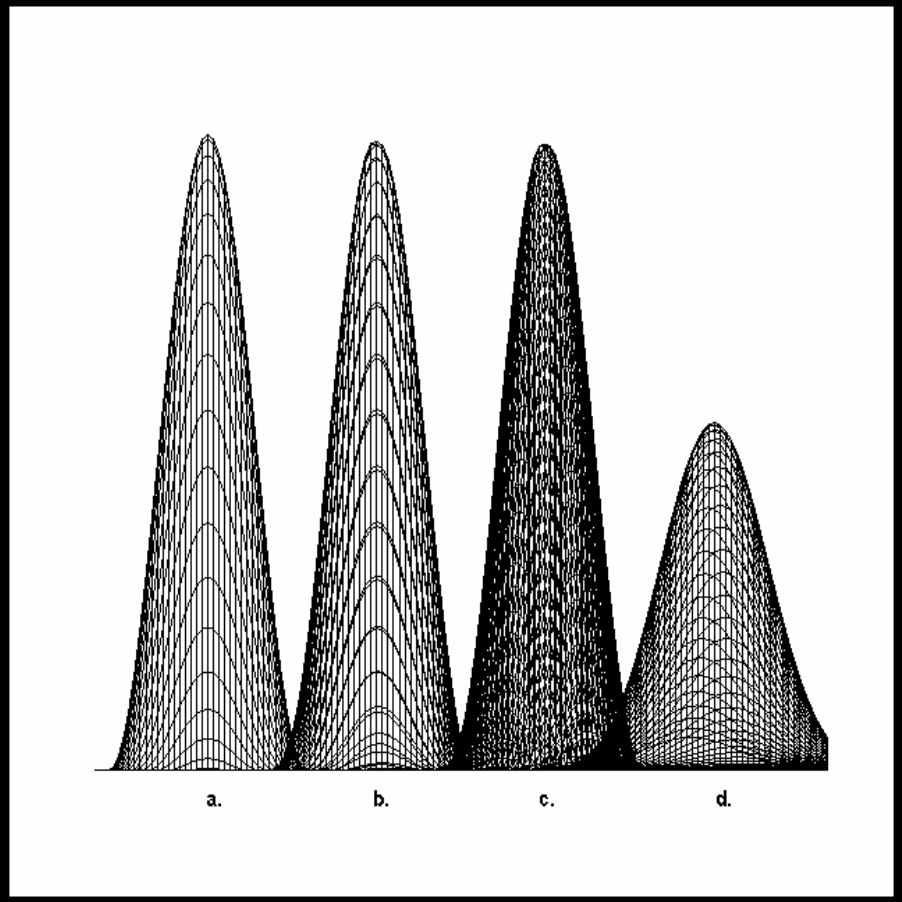
Accuracy? Stability?





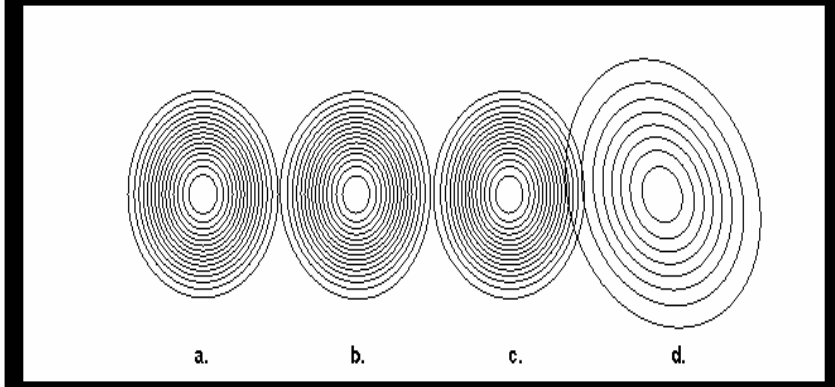


# Example Test Results



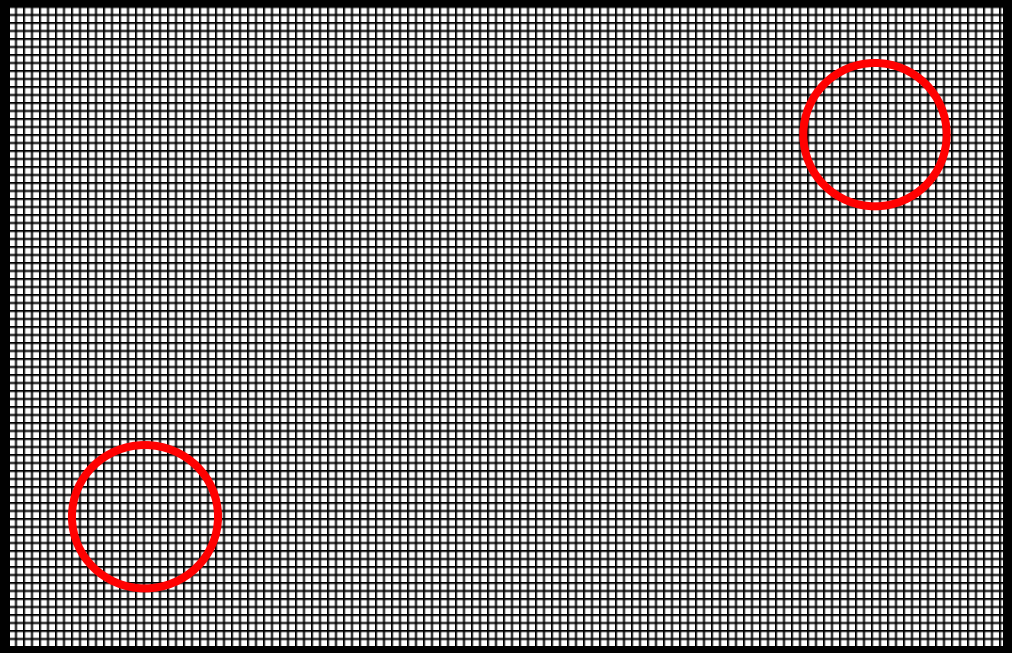

Another look showing the peak and shape preservation properties of the newly implemented method compared to 1st order which we see is overly diffusive.

a : Initial condition
b, c : High order fully multidimensional
d : First order



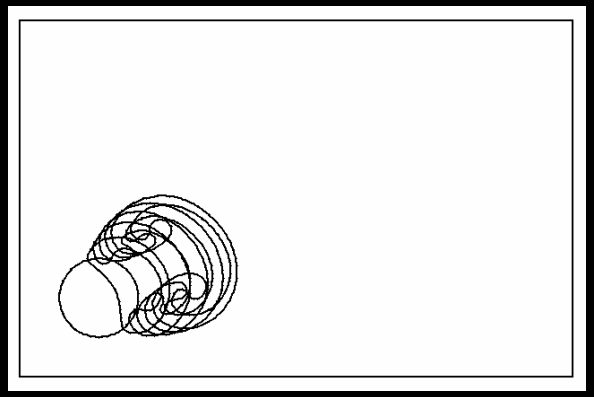
# *Droplet Translation: A Good Test*

- **Heavy 2D drop, initial velocity (2,1) in an ambient light fluid**

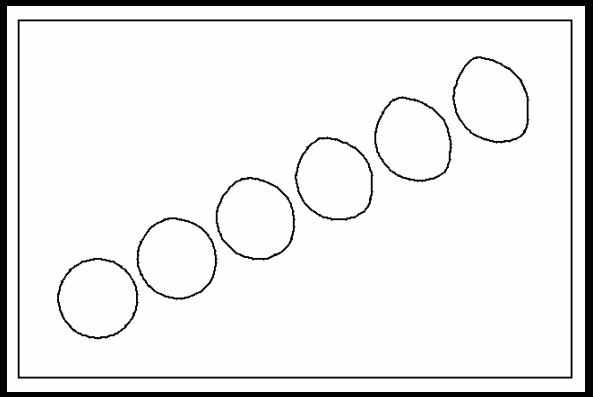


# Droplet Translation: Density Ratio Effects

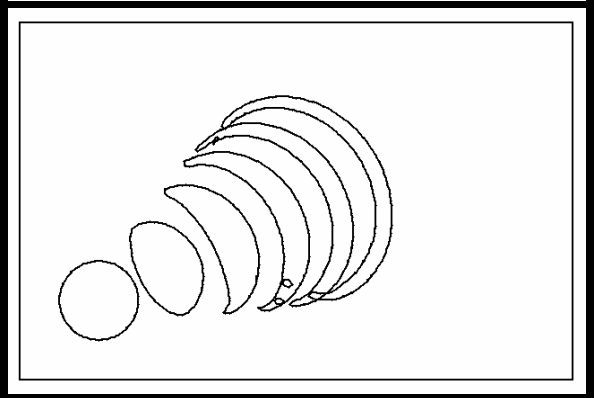
$\frac{\rho_L}{\rho_G} = 1$



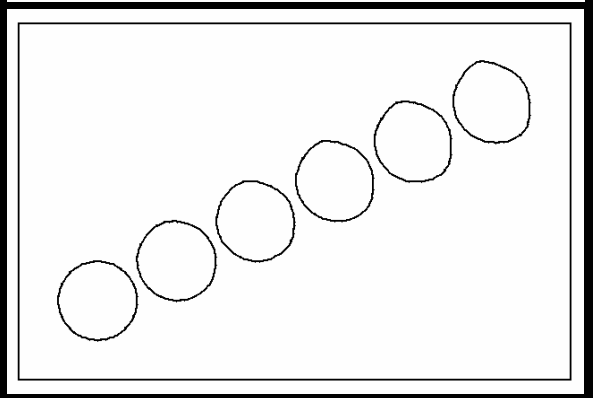
1000



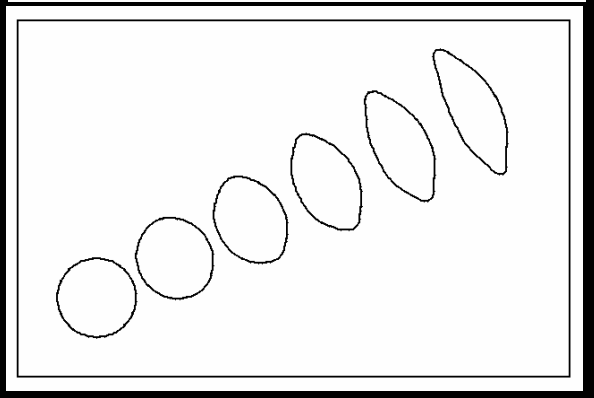
10



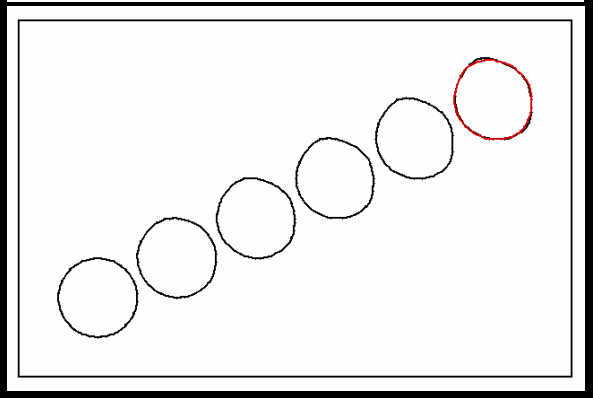
$10^6$



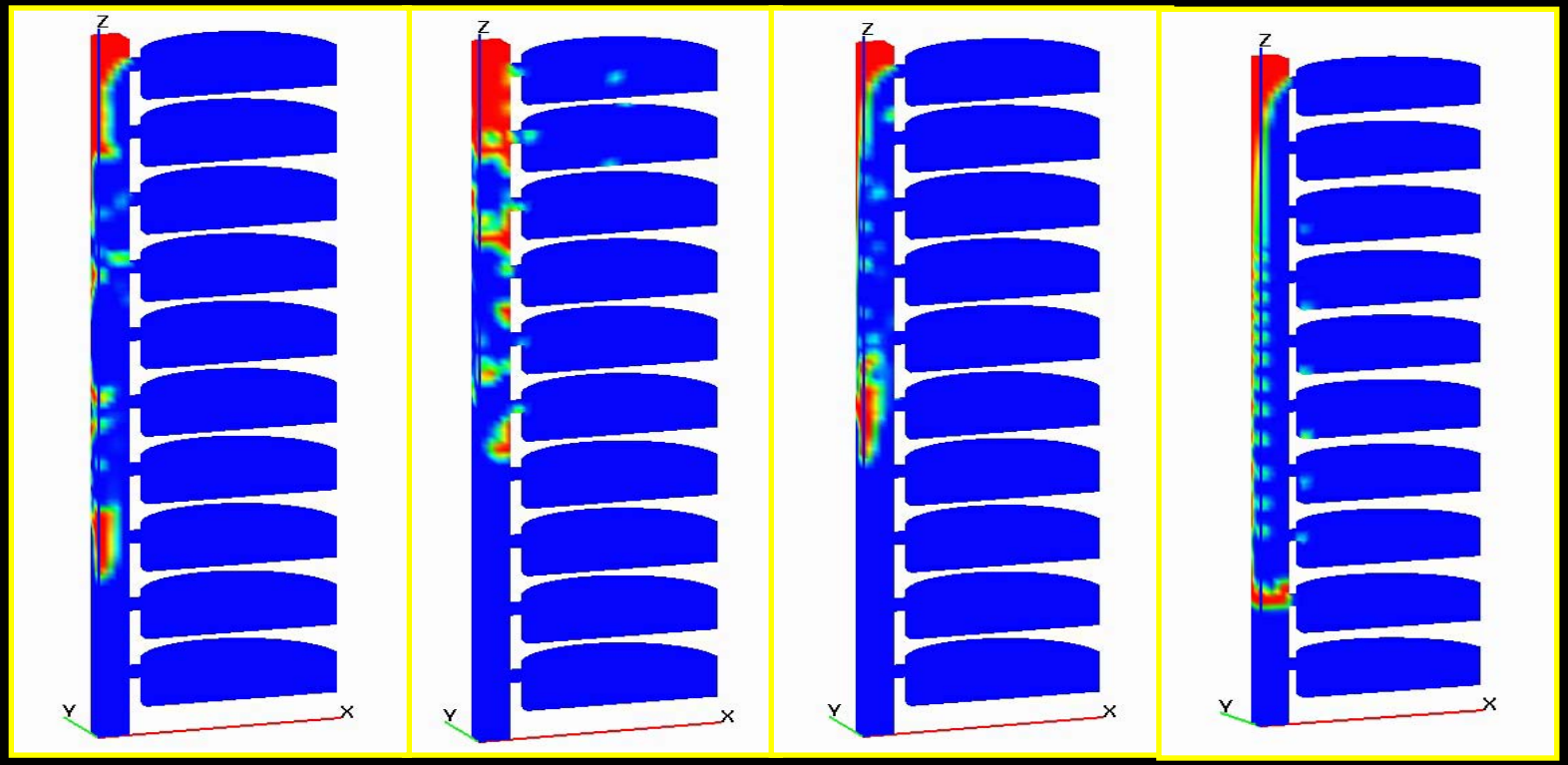
100



$10^9$



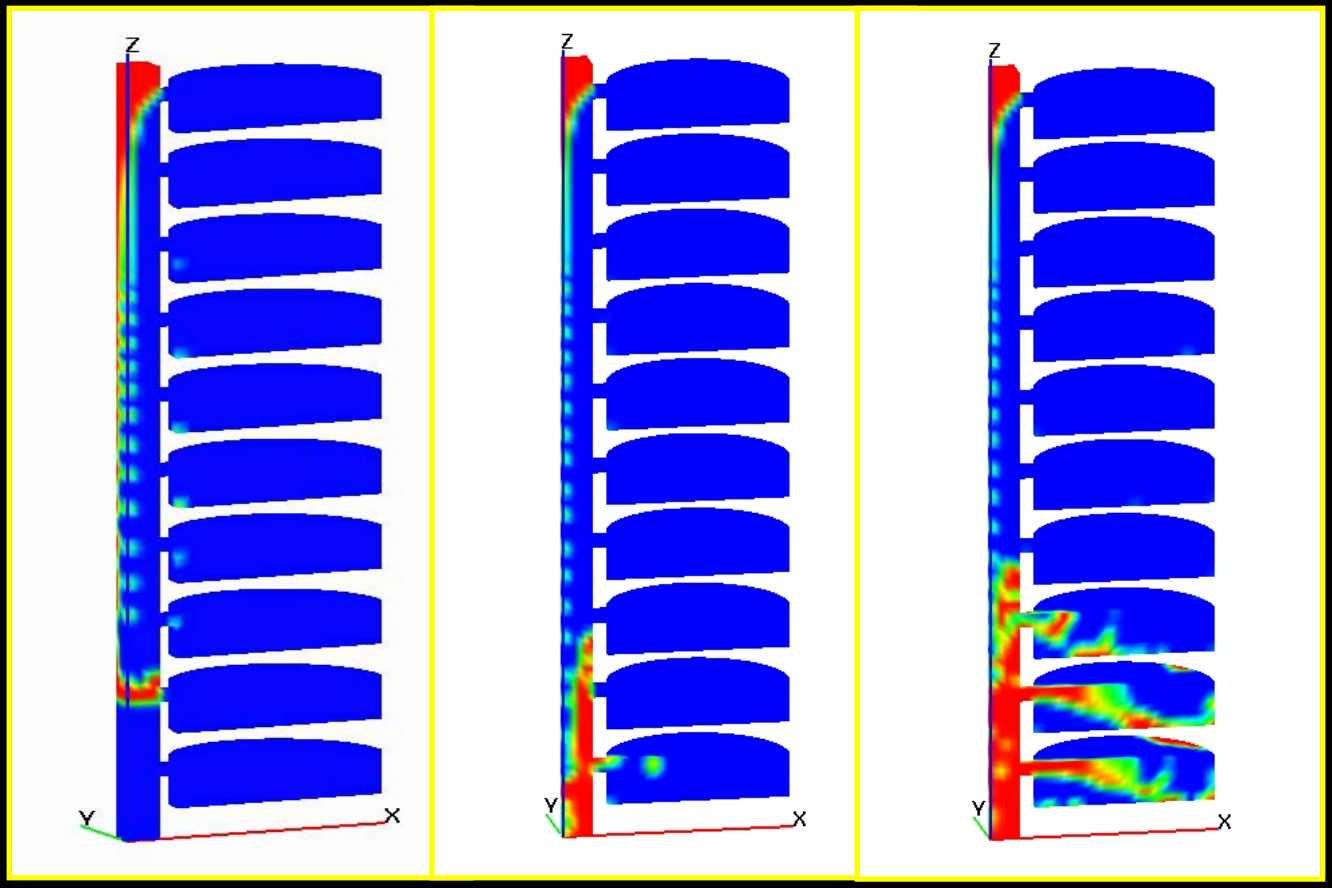
# Application: Puck Mold Fill




Bottom 4 pucks vented | Bottom 4 and top 2 pucks vented | All pucks vented | With void model

**Dramatic differences in filling patterns can seen at 0.2 seconds depending on vent location and number. These simulations motivated the development of a new void model.**

# Application: Puck Mold Fill




0.2 seconds 0.3 seconds 0.6 seconds

**A new void model used here was necessary for success!**

# Surface Tension: The CSF Model

$$\vec{F}_S = \sigma\kappa\hat{n}\delta \quad \Rightarrow \quad \vec{F}_S = \sigma\kappa\nabla f$$

- For static case, the balance between pressure and surface tension force should be verified

$$\nabla P = \sigma\kappa\nabla f$$

- Requires pressure and surface tension terms to be considered together and discretized at same location
- Many ways to estimate curvature: convolution of volume fractions, height function constructed from volume fractions, via a distance function from volume fraction
- Surface tension modeling is particularly important in the simulation of weld pool dynamics

# 2D Viscous Drop Oscillations

$$x^2/9 + y^2/4 = 1$$

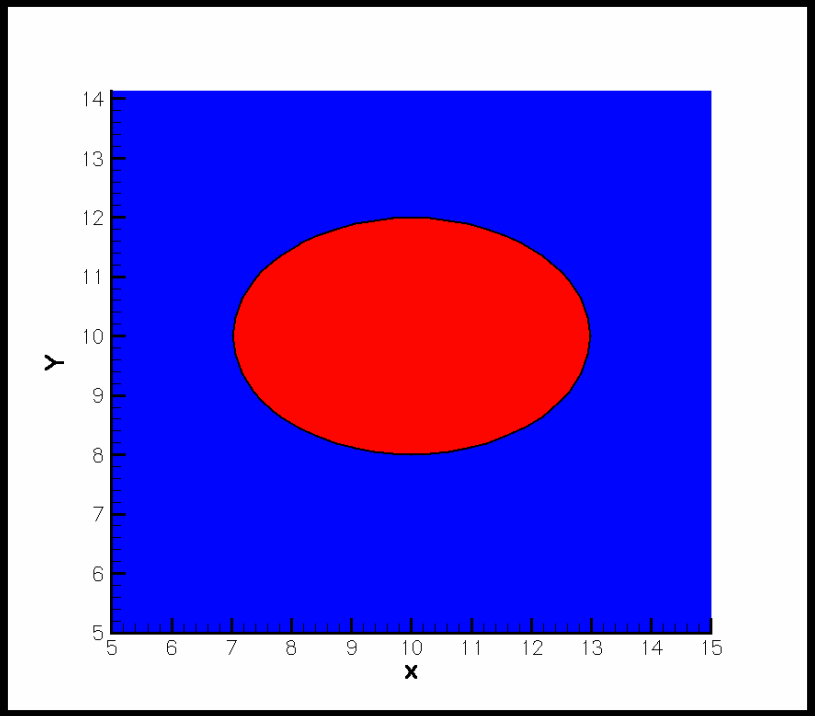
- **Initial shape ellipse**
  **(38.5% deviation of circle of R=2.55)**
  **[Torres and Brackbill 2000]**
- **Natural frequency of oscillations**

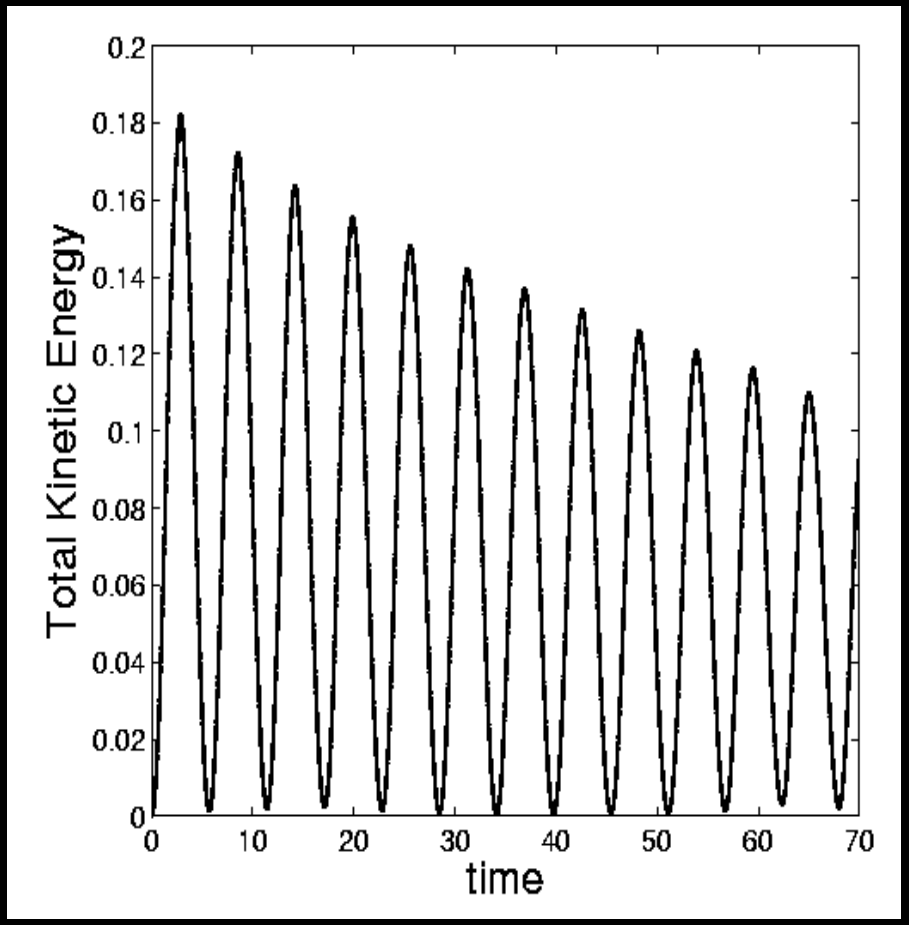
$$\omega_n^2 = \frac{\left(n^3 - n\right)\sigma}{\left(\rho_{in} + \rho_{out}\right)R^3}$$

- **Mode 2 shape oscillations**
  - period $T_{th.}$ = 10.5, $T_{cal.}$ = 11.4 => 8.6 % error
- **64x64 grid, 20x20 domain**
- **$\sigma$=1, $\rho_{in}$=1 $\rho_{out}$=0.01, $\mu_{in}$= 0.01, $\mu_{out}$ =5x10$^{-5}$**

# 2D Viscous Drop Oscillations

Animation until time=35



# *Linear and Nonlinear Solver Philosophy*

- **Embrace Krylov methods**
  - Because they are robust, reliable, and *matrix-free*
- **But improve their efficiency**
  - By introducing better scalability through multi-level preconditioning
- **While maintaining simplicity as constrained by our 3D unstructured mesh topology**
  - The preconditioning operator must be simple, easy to form, and easy to store
  - Reduce mesh connectivity and accuracy order in the preconditioner
  - Formulate two-level solve with piecewise constant prolongation and restriction

# Implicit Time Discretization of PDE Model Leads to Elliptic and Parabolic Systems

- Most of our PDEs follow this general form: $$\frac{\partial \varphi}{\partial t} - \nabla \bullet \left[ D(\phi) \nabla \phi \right] = S(\phi)$$

- Projection method for incompressible Navier-Stokes leads to a pressure Poisson equation: $$\nabla \bullet \left( \frac{\delta t}{\rho} \nabla p \right) = \nabla \bullet \tilde{\mathbf{u}}$$

- Density ratios of 1 to $10^4$ possible across interfaces

- Mixture enthalpy formulation yields a typical diffusion equation: $$\frac{\partial}{\partial t} \langle \rho h \rangle - \nabla \bullet \left[ k \nabla T(h) \right] = S(h)$$

- Conductivity ratios of 1 to 10 typical

# Node-Centered Displacement Field Solutions: A CV-FEM Discretization on the Median Mesh

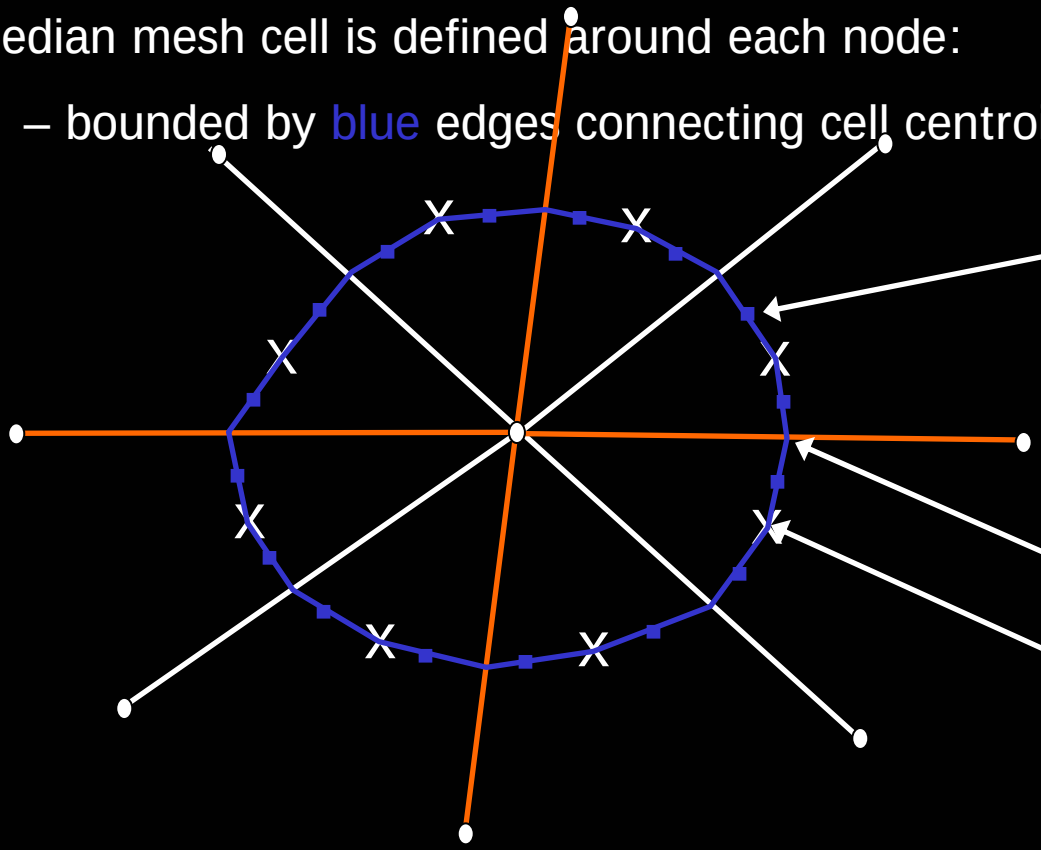
- Control volume formulation leads to a discrete sum over the integration points p on the median mesh surface:

$$\int \frac{\partial \boldsymbol{\sigma}_{ij}}{\partial x_j} dV \approx \sum_p (\boldsymbol{\sigma}_{ij} \bullet \hat{\mathbf{n}}_j A)_p = 0$$

- The stress tensor $\sigma$ at each integration point depends upon first order spatial derivatives of the nodal displacement field at each point p
- These derivatives are evaluated by inverting a 3x3 Jacobian tensor at each integration point p
- Expensive to do implicitly!

# Node-Centered Displacement Field Solutions: A CV-FEM Discretization on the Median Mesh

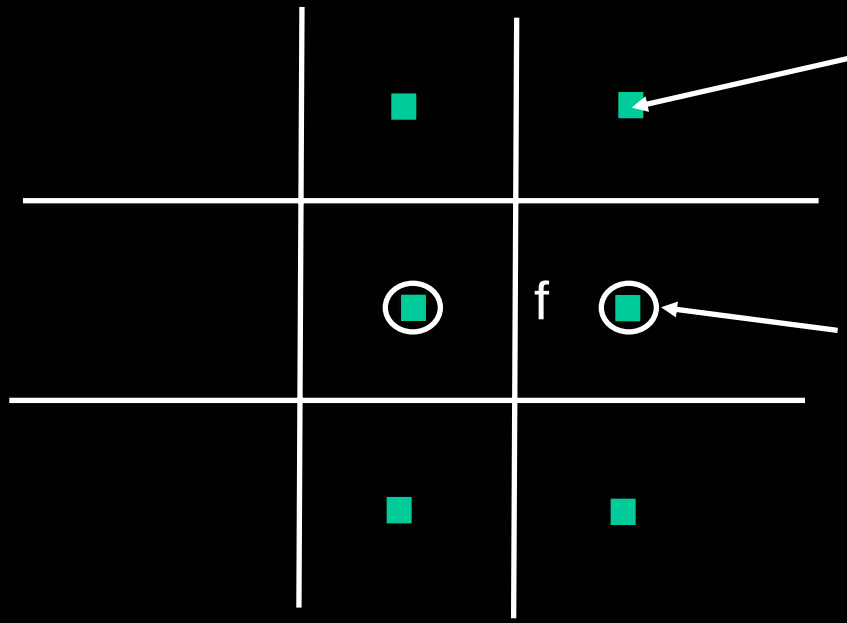
- Median mesh cell is defined around each node:
  - bounded by blue edges connecting cell centroids and edge bisectors




- For the preconditioning operator:
  - Nodes connected by white edges are ignored
  - Nodes connected by orange edges are used

# Pressure Poisson Equation Discretization

$$\int \nabla \bullet \left( \frac{\delta t}{\rho} \nabla p \right) dV \approx \frac{\delta t}{V} \sum_f \left( \frac{1}{\rho} \nabla p \right)_f \bullet \hat{\mathbf{n}}_f A_f$$




Cell that is used for actual face f gradient operator

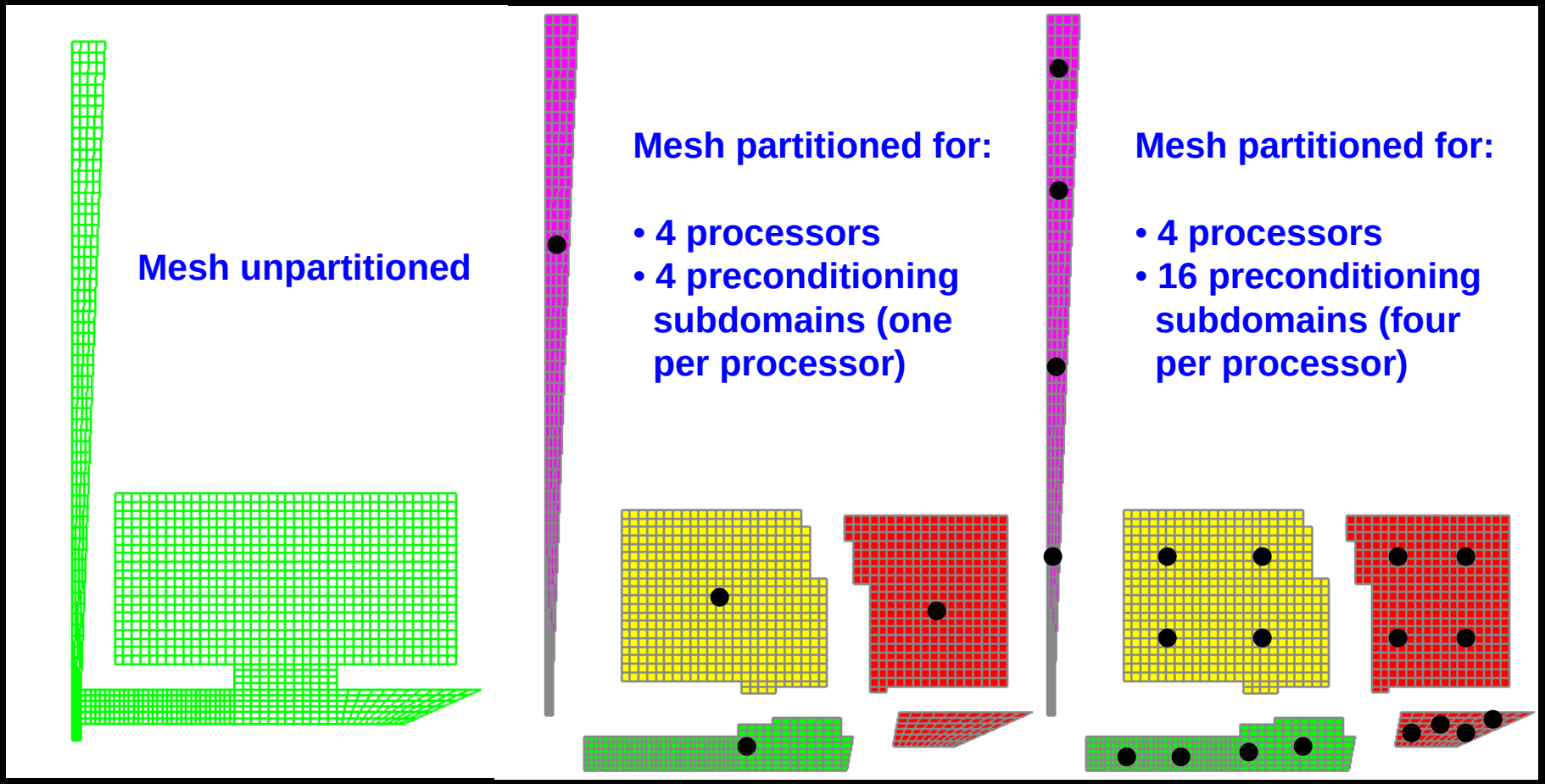
Cell that is used for both actual and preconditioned face f gradient operator

# Two-Level Preconditioner for Krylov-Based Linear Solution Methods

- **Partition mesh into N subdomains (N >= number of processors)**
- **Multigrid idea: coarsen to one equation per subdomain**
- **Number of coarse equations = number of subdomains = N**
- **Algebraic coarsening through the matrix elements; Do not rediscretize physics on the coarse mesh**
- **Easy to implement - *even on unstructured meshes***

| | | |
|---|---|---|
| Precondition (ILU) | $A_f\, x_f = b_f$ | $\Rightarrow x_f^1$ |
| Evaluate residual | $res_f = b_f - A_f\, x_f^1$ | |
| Restrict | $b_c = \mathbf{R}\, res_f$ | |
| | $A_c = \mathbf{R}\, A_f$ | |
| Solve coarse eqtns | $A_c\, \delta x_c = b_c$ | $\Rightarrow \delta x_c$ |
| Prolong | $x_f^2 = x_f^1 + \mathbf{P}\, \delta x_c$ | $\Rightarrow x_f^2$ |
| Evaluate residual | $res_f = b_f - A_f\, x_f^2$ | |
| Smooth (ILU) | $A_f\, \delta x_f = res_f$ | $\Rightarrow \delta x_f$ |
| Update | $x_f^3 = x_f^2 + \delta x_f$ | $\Rightarrow x_f^3$ |

# Benchmark Problem: Mesh




- => Preconditioning subdomain coarse grid point

# *Two-Level Preconditioning on Multiple Subdomains Per Processor Yields Substantial Efficiency Gains*

| Subdomains | Processors | Total Iterations | Total Time (seconds) | Speedup |
|---|---|---|---|---|
| 0 | 1 | 2989 | 14760 | 1 |
| **64** | **1** | **250** | **1275** | **12** |
| 0 | 2 | 2781 | 6900 | |
| 2 | 2 | 1180 | 3000 | 1 |
| **64** | **2** | **250** | **640** | **5** |

- **0 subdomains means no 2-level preconditioning.**
- **Speedup is relative to previous "best case".**

# *Nonlinear Solution Algorithm*
# *The Newton-Krylov (NK) Method*

**Wish to solve $F_i(u) = 0$, where F is a nonlinear system of equations and u is a state vector**

- **Newton's method typically requires the Jacobian matrix:**

$$J_{i,j} = \partial F_i(u) / \partial u_j$$

- **$J_{i,j}$ can be**
  - **difficult / costly to form, ill-conditioned, expensive to store**
- **By using a *Krylov* algorithm to solve $-J_{i,j}\delta u_j = F_i$**
  - **the action of $J_{i,j}$ occurs only via matrix-vector products**
  - **approximate by finite differences (v = Krylov vector):**

$$J_{i,j}v_j \cong [F_i(u+\varepsilon v) - F_i(u)] / \varepsilon$$

- **Newton performance *without directly forming* the Jacobian**

# JFNK: Jacobian-Free Newton-Krylov Methods

- JFNK is used for all heat transfer and phase change simulations
  - Novel solidification front "CFL" time step control
  - Multiple simultaneous phase changes can occur in a single simulation
- JFNK is implemented as a general Black Box Newton-Krylov (BBNK) nonlinear solver library
  - Developer provides nonlinear function $F_n(x_j)$ for n equations and j unknowns
  - Developer provides approximate function $\widetilde{F}_n(x_j)$ for preconditioner
- JFNK can be used for incompressible Navier-Stokes solutions
  - Use semi-implicit fractional step algorithm used a preconditioner
  - Also possible: solidifying flow via JFNK by combining Navier-Stokes, heat transfer, and phase change packages
  - Fully coupled implicit solutions without restrictive time step constraints
- JFNK used for nucleation & growth model prototyping

# Nonlinear NK Phase Change Algorithm Performance

- **Simulation details**
  - $\delta t$ = 1.0/0.20/0.04 and 2.5/0.50/0.10 for mesh sizes of $32^3/64^3/128^3$
  - 3D alloy solidification
- **Simulation results (iterations as a function of problem size)**

| Mesh | PEs | Nonlinear/linear (small $\delta t$) | Nonlinear/linear (large $\delta t$) |
|---|---|---|---|
| $32^3$ | 8 | 35/184 | 47/316 |
| | 16 | 35/194 | 47/329 |
| | 32 | 35/196 | 47/339 |
| | 64 | 35/201 | 47/356 |
| $64^3$ | 8 | 33/173 | 41/292 |
| | 16 | 33/183 | 41/307 |
| | 32 | 33/184 | 41/318 |
| | 64 | 33/186 | 41/329 |
| $128^3$ | 64 | 29/187 | 38/334 |
| | 96 | 29/188 | 38/335 |

# Adaptive Time Step Maintenance of Dynamical Accuracy on Nonlinear Systems

- If a fully implicit method (JFNK) is used for the nonlinear systems, then an arbitrarily large integration time step can be used
  - But what if the solution admits sharp fronts and the time-dependent front location is an important feature to resolve and account for?
  - What is the largest time step that can be used without sacrificing dynamical resolution of the front?
- A novel and simple idea introduced recently by D. Knoll and W. Rider (LANL) solves this problem
  - Given some nonlinear solution, e.g., enthalpy H
  - Assume the solution has a hyperbolic form with a characteristic wave speed $\mathbf{v_f}$: $\frac{\partial H}{\partial t} + \boldsymbol{v}_f \bullet \nabla H = 0$
- Find the wave speed via solution of $|\boldsymbol{v}_f| = |\partial H / \partial t| / |\nabla H|$
- Restrict the time step via a "front CFL condition": $\delta t < C\, \delta x / |\boldsymbol{v}_f|,\ 0 < C \le 1$
- Properly predicts thermal and solidification wave speeds

# Algorithm Enablers

- Better models for interface kinematics on fixed grids
  - Need robust and accurate methods for modeling complex topology interface kinematics (“graceful termination”)
  - Liquid/gas, liquid/liquid, liquid/solid
- High-resolution discrete vector calculus operators
  - Accurate and convergent operators on arbitrary meshes
  - Constrained, weighted least squares near external boundaries and internal, moving interfaces?
- Scalable linear solution (Ax=b) methods
  - Multi-level preconditioned Krylov techniques?
- Scalable nonlinear solution [F(x)=0] methods
  - Multi-level preconditioned Newton-Krylov (NK) techniques
  - NK methods are attractive because Jacobian-free (JF) !
- Adaptive temporal accuracy for dynamical solutions that admit sharp fronts (enthalpy)

# More Algorithm Enablers

- Parallelism
  - Brings more CPU horsepower and memory to 3-D simulations
  - Portable, general parallel paradigm that does not sacrifice single processor performance?
  - Paradigm that is best suited for platforms of today/tomorrow?
- Coupled, implicit solutions for the entire nonlinear multi-physics equation system
  - More research is need to enhance robustness and scalability
- Discrete particle models
  - Useful for "history-dependent" physics (freckles?), gross deformation
- Interface dynamics and interfacial flows
  - Continuum surface tension forces, wetting models, and geometric approximations for flow quantities & advection near the interface
  - "Smart" interface estimates in melt/solid exchange terms
  - Approximating moving and stationary rigid obstacles whose features might be too fine to resolve

# Issues Related to Casting Simulations: Notes From One of Our Power Users

Many codes can model fluid flow and heat transfer but to accurately simulate many DOE castings more physics is needed.

- Initial mold temperatures – only outer edge of mold couples with the induction field, the much of the mold stack is heated by conduction and radiation
  - Electromagnetic model
  - Radiation model with view factors
- Time/stress/position dependent heat transfer coefficient
- Stresses can develop within the casting to cause breakage of the mold and/or hot tearing of the metal
  - Thermo-mechanical model
- Microstructure/Alloy solidification–
  - Macrosegregation model
  - Dendritic growth and microsegregation model

# Barriers: Physics

- Coupling the micro and macro length scales
  - Common theme in other fields (turbulence, climate modeling)
  - Averaged models inspired by DNS data sets and homogenization theories (*mushy zone transport*)
  - Can micro and macro simulations be coupled simultaneously?
- Phase change heat transfer
  - Non-equilibrium kinetics
  - Microstructural nucleation and growth models
  - Solid state transitions and their role in stress/distortion buildup
  - Generalization to arbitrary equilibrium phase diagram relationships
  - Boundary condition models for internal gaps
- Material response
  - Creep and hot tearing models
  - Microporosity growth models
  - Solid state transition effects (TRIP models)

# Barriers: Algorithms, Computer Science

- **Algorithms**
  - Finite-width interface representation on fixed meshes
  - Discretization around interfaces where physics change
    - Imposition of constraints, reduction of order
  - Linear/Nonlinear solution algorithms
    - Matrix-free Krylov and Newton-Krylov methods are great
      - A lot still depends on physics-based preconditioning
      - How much can pushed into “black boxes”?
    - When to couple and when not to couple? Cost/benefit?
  - Boundary condition treatments
  - Balancing robustness with accuracy; cost/benefit?
- **Computer Science**
  - Single PE execution time; >80% of cycles spent moving data
  - Efficient software processes and practices
    - Frameworks, environments, in-situ prototyping
  - Finding the right mix of efforts devoted to software engineering and physics models & algorithms

# *Conclusions*

- **Computational manufacturing (casting, welding) is a field in its infancy**
  - Only ~15 years old
- **Tremendous challenges**
  - Physical models
    - Coupling the macro and micro length scales
  - Numerical algorithms
    - Efficient, accurate solutions of systems of nonlinear PDEs
- **We welcome your interest and participation**
  - Contact the project @ telluride-info@lanl.gov
  - Contact me @ dbk@lanl.gov
- **A great project for CSGF students!**
  - In the past we've supported 7 PhD students and 8 Postdocs