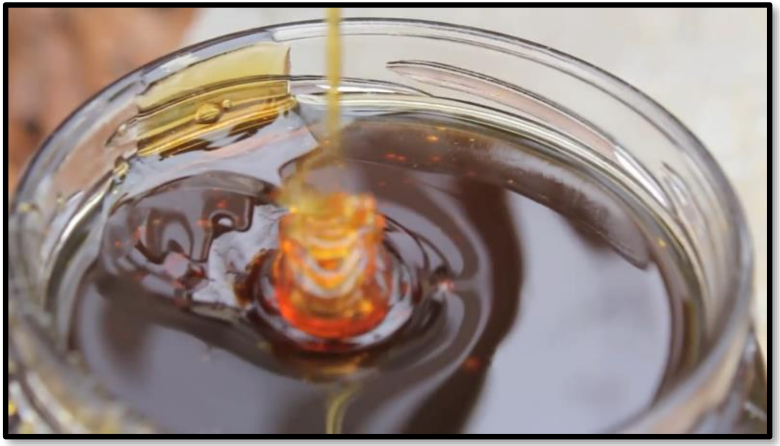
Viscous Threads (and Sheets and Volumes)

Christopher Batty

UNIVERSITY OF WATERLOO

Joint work with Mridul Aanjaneya, Robert Bridson, Ryan Goldade, Egor Larionov, & Yipeng Wang.

Characteristic Behaviours of (Real) Viscous Liquids

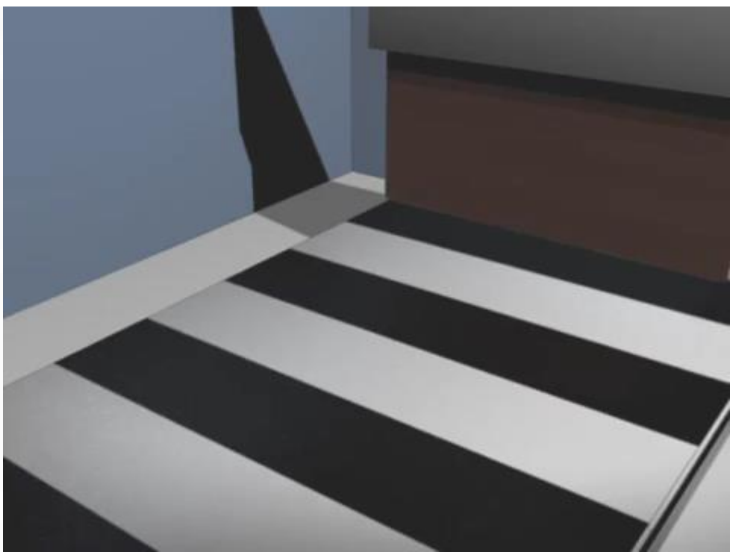


Rope Coiling

Buckling & Folding

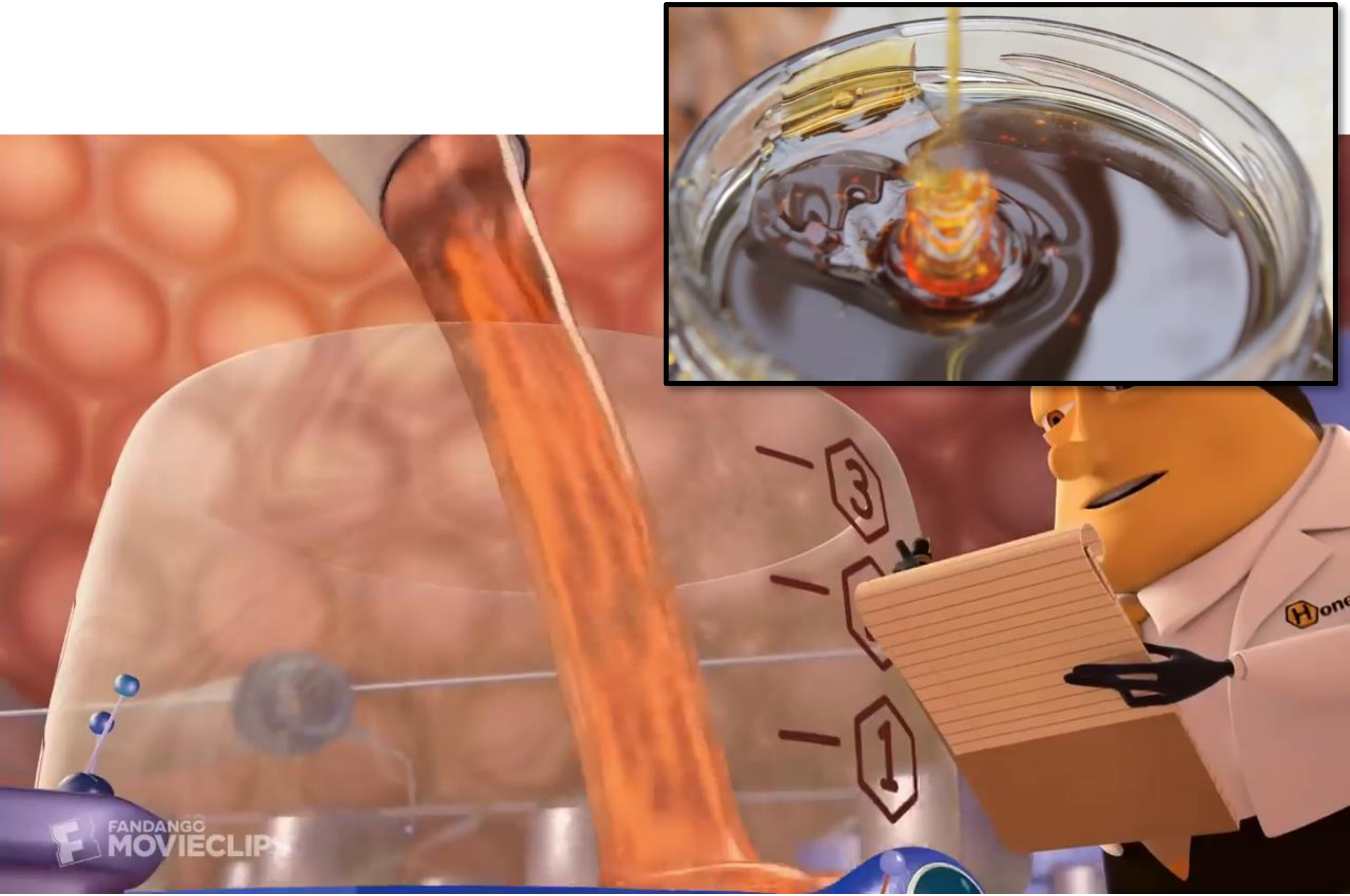
Specialized Approaches

Grinspun & co. developed rod- and shell-based models for thin threads and sheets.



Viscous Liquid on Film, c.2007

[Bee Movie, 2007]


3
1
FANDANGO
MOVIECLIPS

Our Aims

1. Fully general three-dimensional model
2. Favour simplicity & efficiency
3. Capture folding & coiling!

1\. & 2. suggest a Cartesian grid, finite difference approach.

How to achieve 3.?

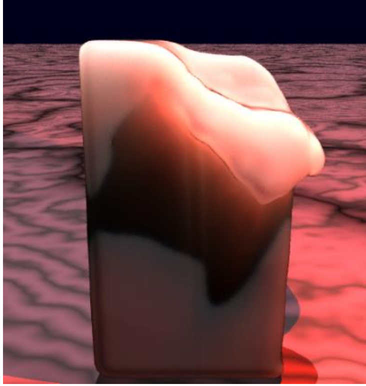
Standard Fluid Animation Framework

Apply operator-splitting to incompressible Navier-Stokes eq’ns, yielding four distinct steps/PDEs:

1. Advect velocities
2. Integrate body forces
3. Integrate viscosity
4. Enforce incompressibility

Approximate each PDE with finite differences.

Early Work in Graphics (Carlson et al. 2002)



Simplest possible viscosity model!

Component-wise diffusion ("Laplacian smoothing"):

$$\frac{\partial \boldsymbol{u}}{\partial t} = \frac{\mu}{\rho} \nabla \cdot \nabla \boldsymbol{u}$$

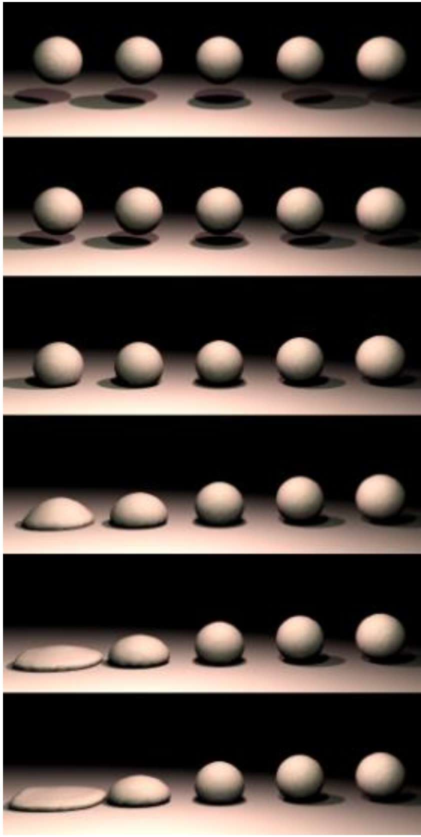
- Easy, stable with backward Euler. ☺
- Erroneous boundary conditions caused falling liquid to freeze in mid-air… ☹

Slightly Better Boundaries (Falt & Roble 2003)

Apply more "reasonable" Neumann BC to the diffusion model:

$$\nabla \boldsymbol{u} \cdot \boldsymbol{n} = \boldsymbol{0}$$

- Correct translational motion. ☺
- For spatially varying viscosity coefficients, diffusion form is wrong. Need coupling between velocity components! ☹



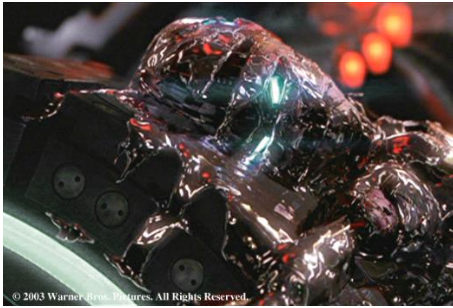
Low to High Viscosity

Variable Coefficients (Rasmussen et al. 2004)

Variable coefficient viscosity equations, integrated via IMEX scheme:

$$\frac{\partial \boldsymbol{u}}{\partial t} = \frac{1}{\rho} \nabla \cdot \mu(\overset{\text{Implicit}}{\nabla \boldsymbol{u}} + \overset{\text{Explicit}}{\nabla \boldsymbol{u}^T})$$

- Valid for variable viscosity. ☺
- Simple Neumann BC ($\nabla \boldsymbol{u} \cdot \boldsymbol{n} = \boldsymbol{0}$) still incorrect: destroys rotational motion. ☹



[Terminator 3, 2003]

Rotation comparison

Expected behavior
(rotation allowed)

Erroneous boundary conditions
(only shearing allowed)

Our Work

1. Improve boundaries to allow rotation & recover buckling [Batty et al. 2008]
2. Introduce octree discretization for efficiency [Goldade et al. 2019]
3. Avoid splitting error to recover coiling [Larionov et al. 2017]

Theme: Exploit variational form of the viscosity PDE(s) to simplify discretization.

Viscosity PDE

Velocity evolution: $$\rho \frac{\partial \boldsymbol{u}}{\partial t} = \nabla \cdot \boldsymbol{\tau}$$

Viscous stress tensor: $$\boldsymbol{\tau} = \mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T)$$

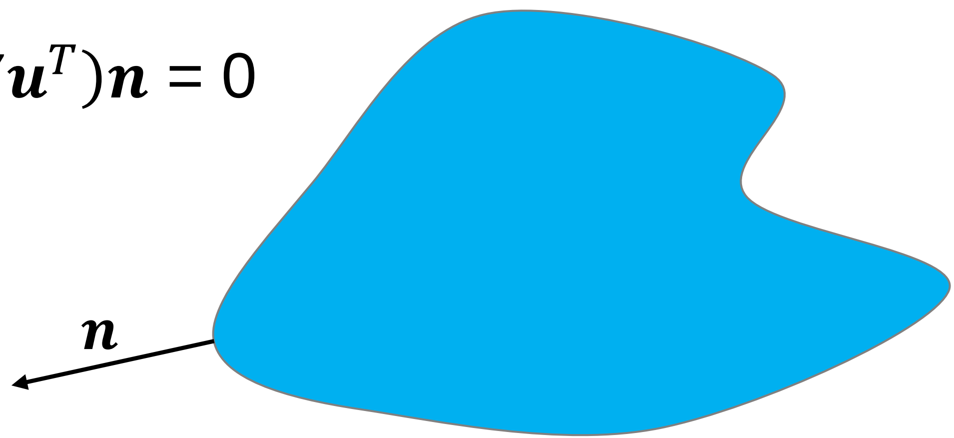
Desired boundary conditions…

- Solids: $\boldsymbol{u} = \boldsymbol{u}_{solid}$
- Liquid/Air Surface: **zero traction**

Zero Traction BC

Surrounding “air” (void) should apply no ghost force (traction $\boldsymbol{t}$) on liquid.

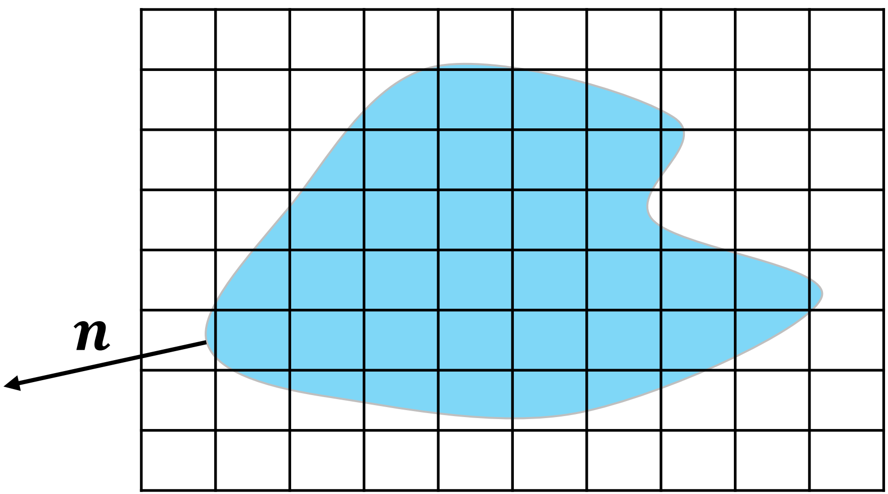
$$\begin{aligned} \boldsymbol{t} &= \boldsymbol{\tau}\boldsymbol{n} = 0 \\ &= \mu(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^T)\boldsymbol{n} = 0 \end{aligned}$$



Discretization Challenges

Regular grid / finite differences enable various efficiencies & implementation simplicity.

But complicate treating irregular geometry.



Our Approach: Variational Form

Replace PDE with equivalent variational expression & exploit natural boundary conditions.

$$\rho\,\frac{(\mathbf{u}^{\mathrm{n+1}} - \mathbf{u}^{\mathrm{n}})}{\Delta t} = \nabla \cdot \boldsymbol{\tau} \quad \text{and} \quad \boldsymbol{\tau} = \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)$$

$$\underset{\mathbf{u}^{\mathrm{n+1}}}{\operatorname{argmin}} \iiint_\Omega \left(\frac{\rho}{2\Delta t} \|\mathbf{u}^{\mathrm{n+1}} - \mathbf{u}^{\mathrm{n}}\|_2^2 + \mu \left\| \frac{\nabla\mathbf{u}^{\mathrm{n+1}} + (\nabla\mathbf{u}^{\mathrm{n+1}})^T}{2} \right\|_F^2 \right) dV$$

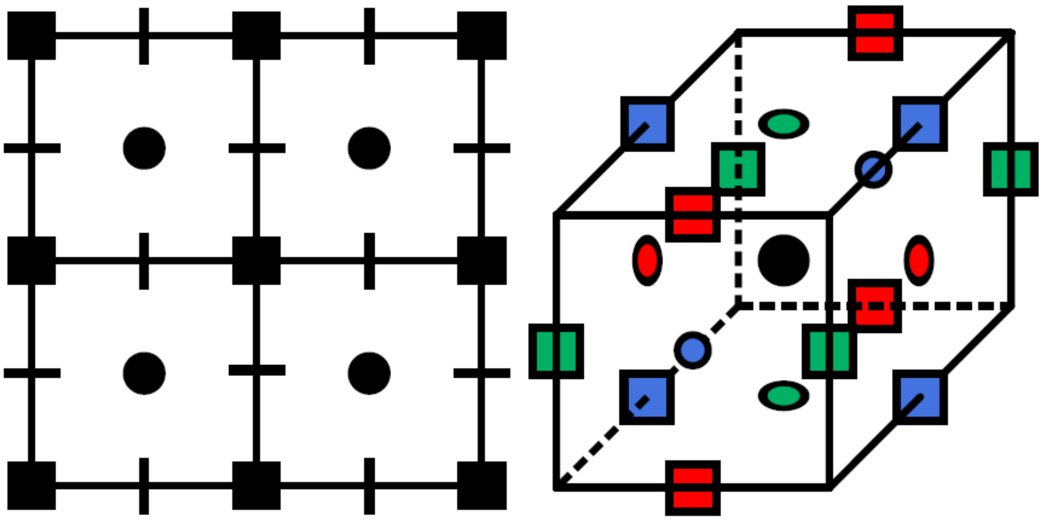
Natural BC

$$\underset{\mathbf{u}^{\mathrm{n+1}}}{\operatorname{argmin}} \iiint_{\Omega} \left(\frac{\rho}{2\Delta t} \|\mathbf{u}^{\mathrm{n+1}} - \mathbf{u}^{\mathrm{n}}\|_2^2 + \mu \left\| \frac{\nabla \mathbf{u}^{\mathrm{n+1}} + (\nabla \mathbf{u}^{\mathrm{n+1}})^T}{2} \right\|_F^2 \right) dV$$

The “natural” boundary conditions of this form are the desired zero traction, $\mu(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T) \cdot \boldsymbol{n} = 0$.

Just approximate the integral and optimize; no need to explicitly enforce BC.

Approximating Derivatives – Centered Finite Differences on Staggered Grid



2D Grid 3D Grid Cell

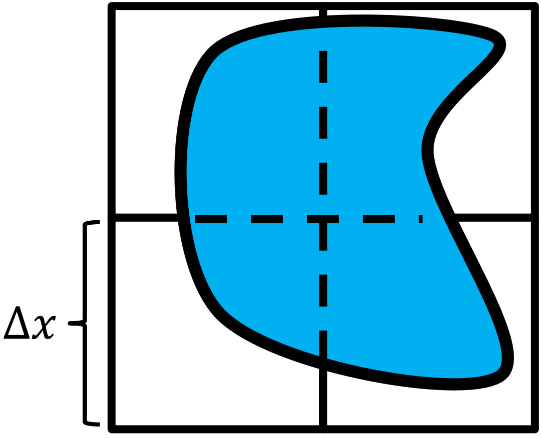
Approximating the Integral

Simply assume piecewise constant data per cell.

$$\int_{\Omega} q dV \approx \sum_{Cells} q_{cell} \cdot V_{cell}$$

where

$V_{cell} = (\Delta x)^3 \cdot (\text{fluid fraction})$.

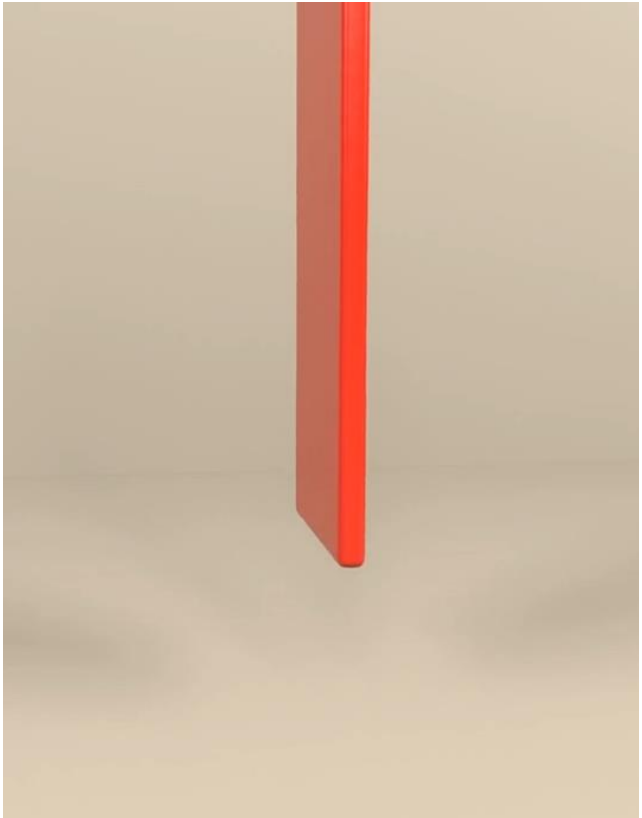


No need to approximate normals, enforce boundary constraints, or use a conforming mesh.

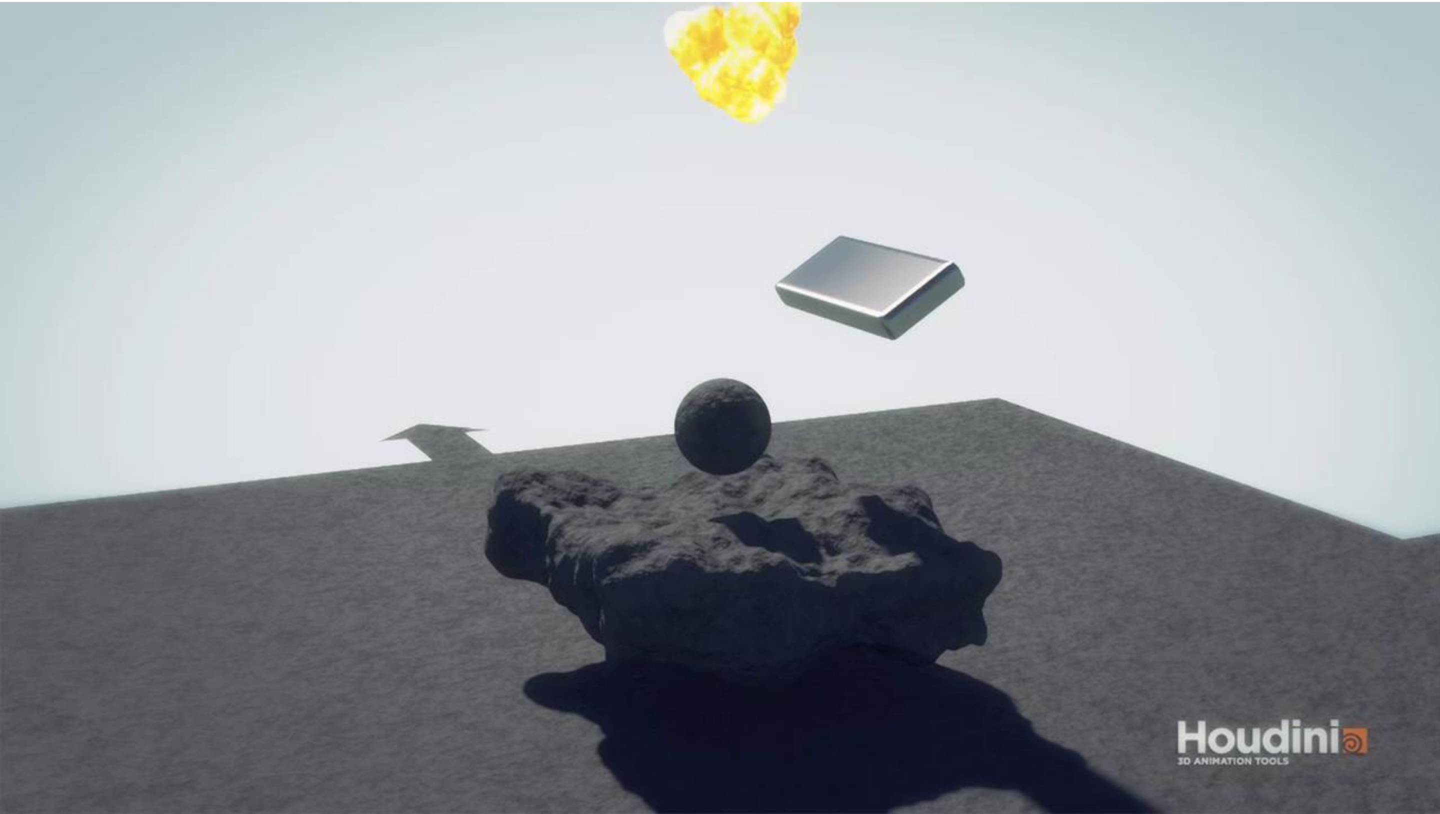
Buckling!

- Recovers the desired buckling effect.
- Reduces to standard finite differences on interior regions.

- Bonus: quadratic variational form yields a symmetric (pos. def.) linear system.



Industrial Adoption - Houdini

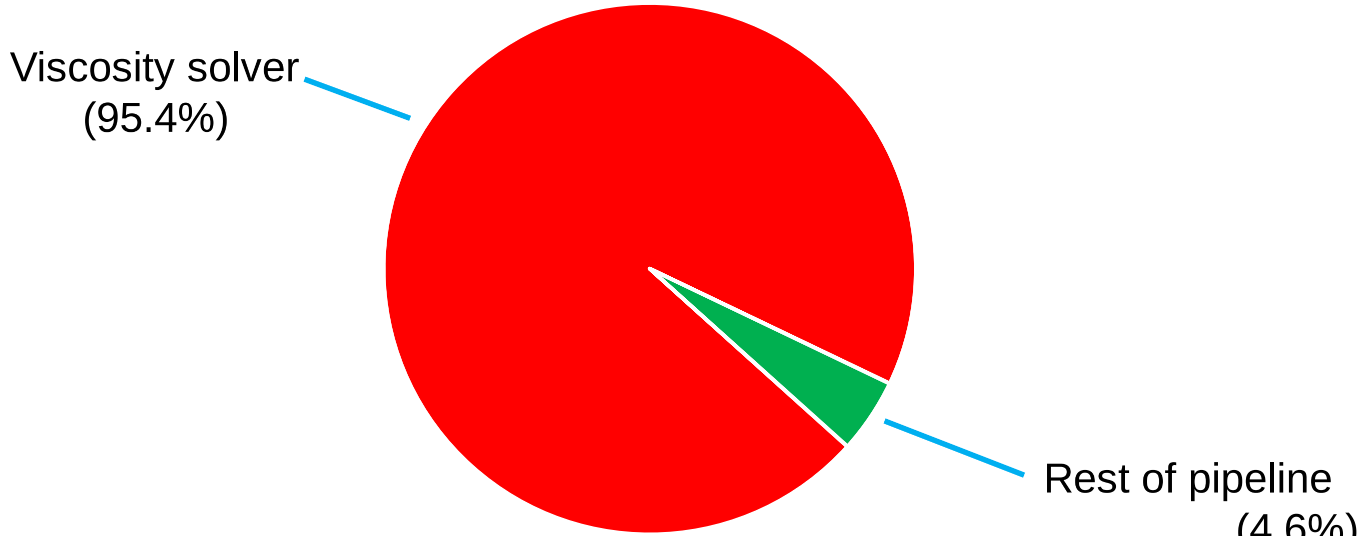


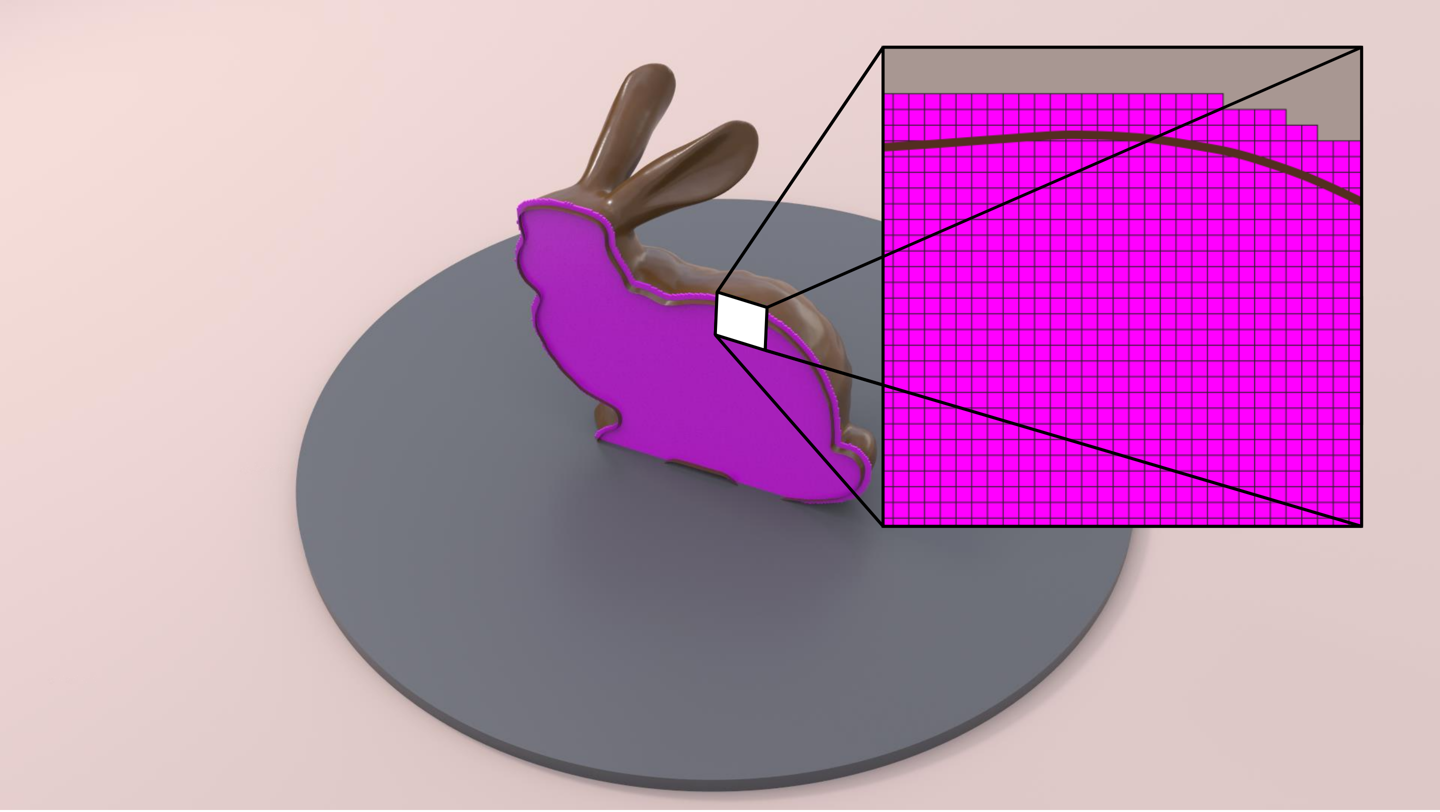
Our Work

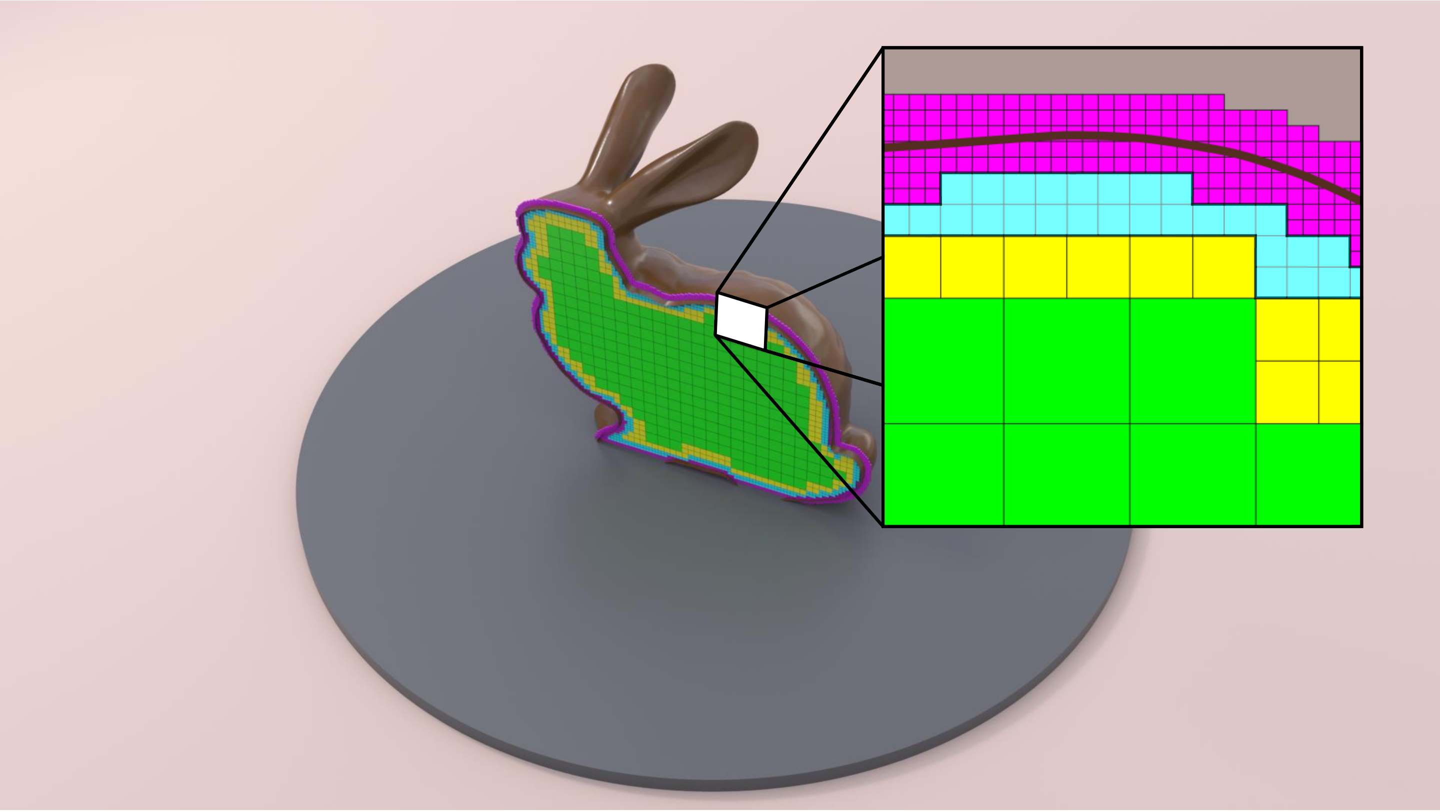
1. Improve boundaries to allow rotation & recover buckling [Batty et al. 2008] ✓
2. Introduce octree discretization for efficiency [Goldade et al. 2019]
3. Avoid splitting error to recover coiling [Larionov et al. 2017]

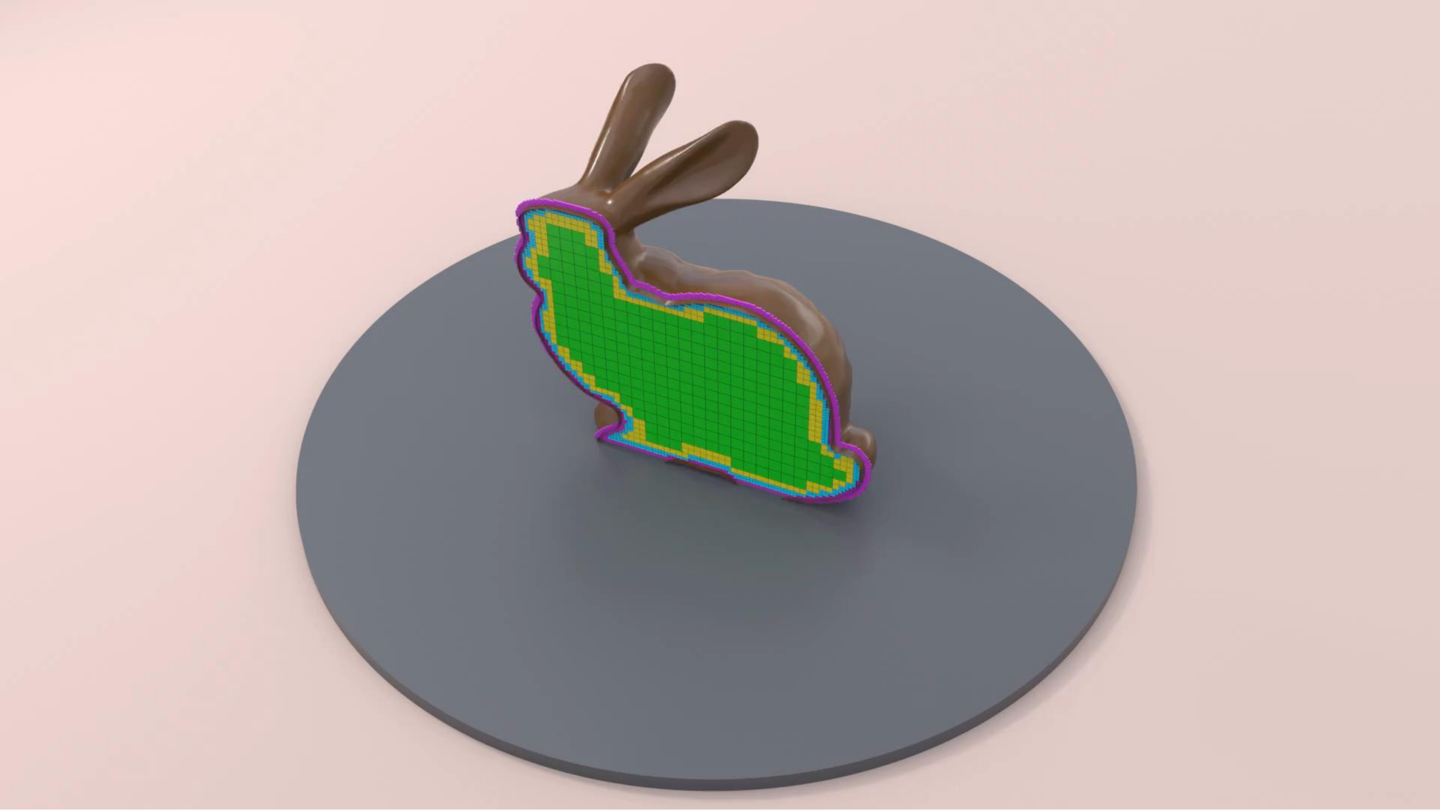
Theme: Exploit variational form of the viscosity PDE(s) to simplify discretization.

Melting Bunny – Pipeline Breakdown



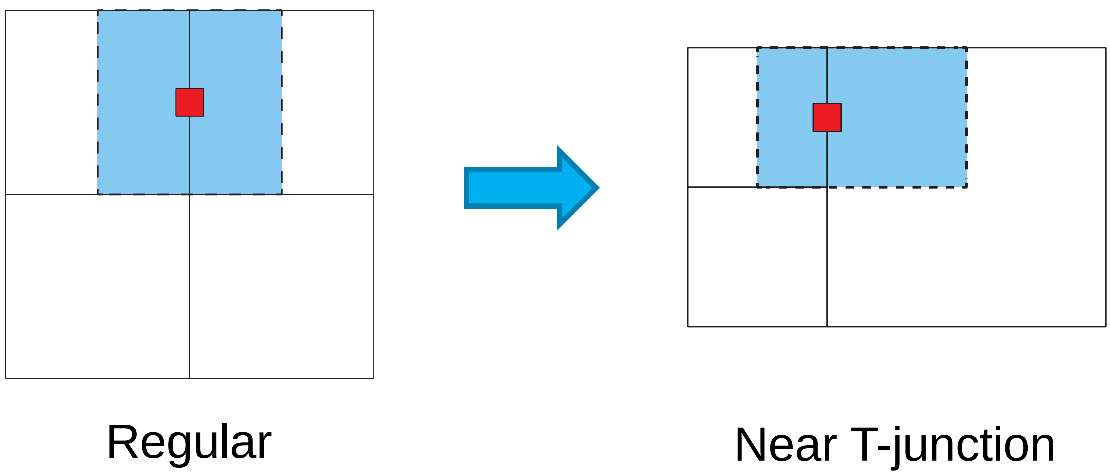






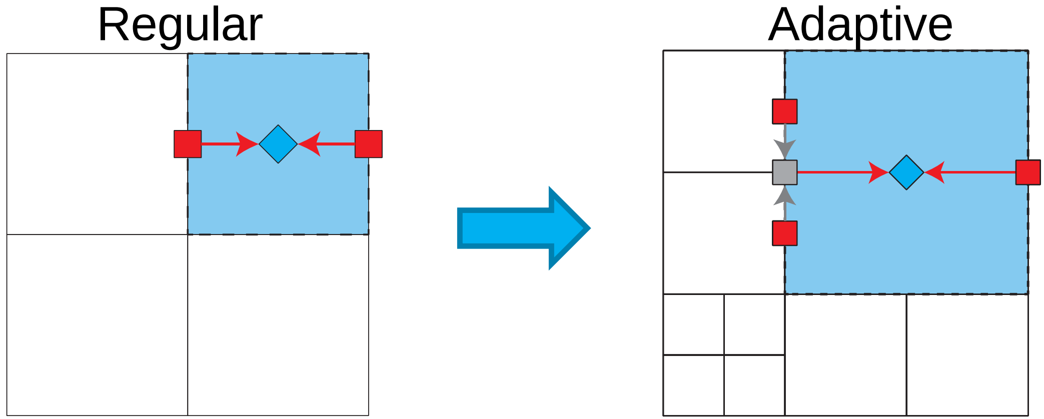
Two Key Changes for Adaptivity

1) Modify the control volumes (cyan) for estimating the integrals.



Two Key Changes for Adaptivity

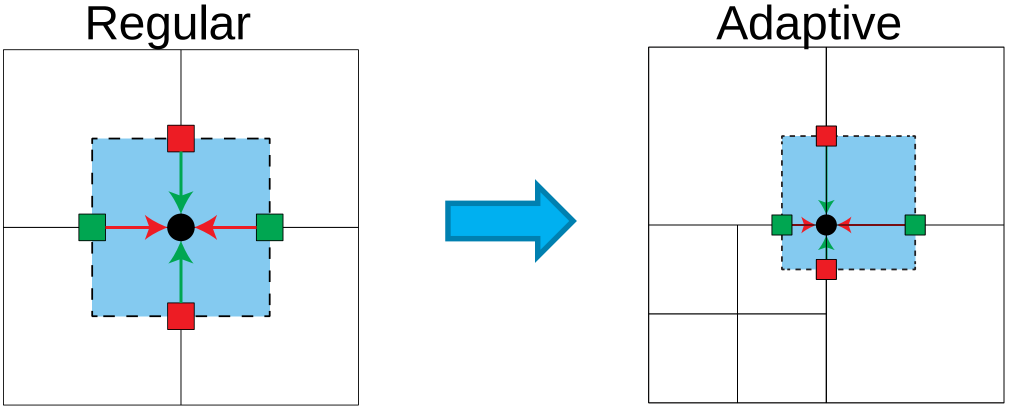
2) Modify finite difference stencils near T-junctions.



$$\text{e.g., } \tau_{xx} = 2\mu\frac{\partial u}{\partial x}$$

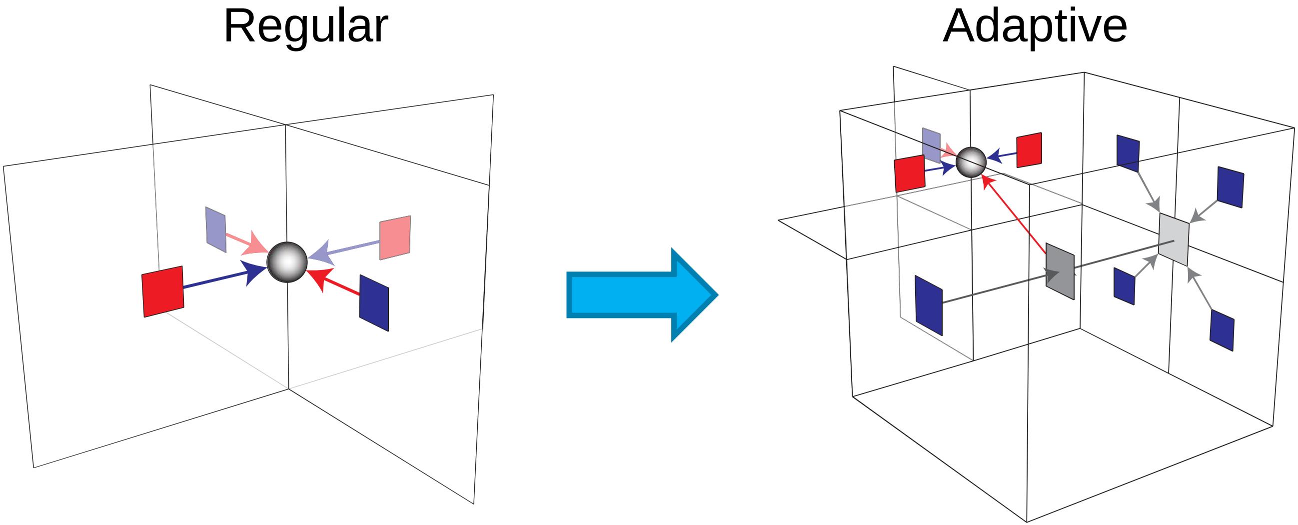
Two Key Changes for Adaptivity

2) Modify finite difference stencils near T-junctions.



$$\text{e.g., } \tau_{xy} = \mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)$$

3D is subtler, but same idea



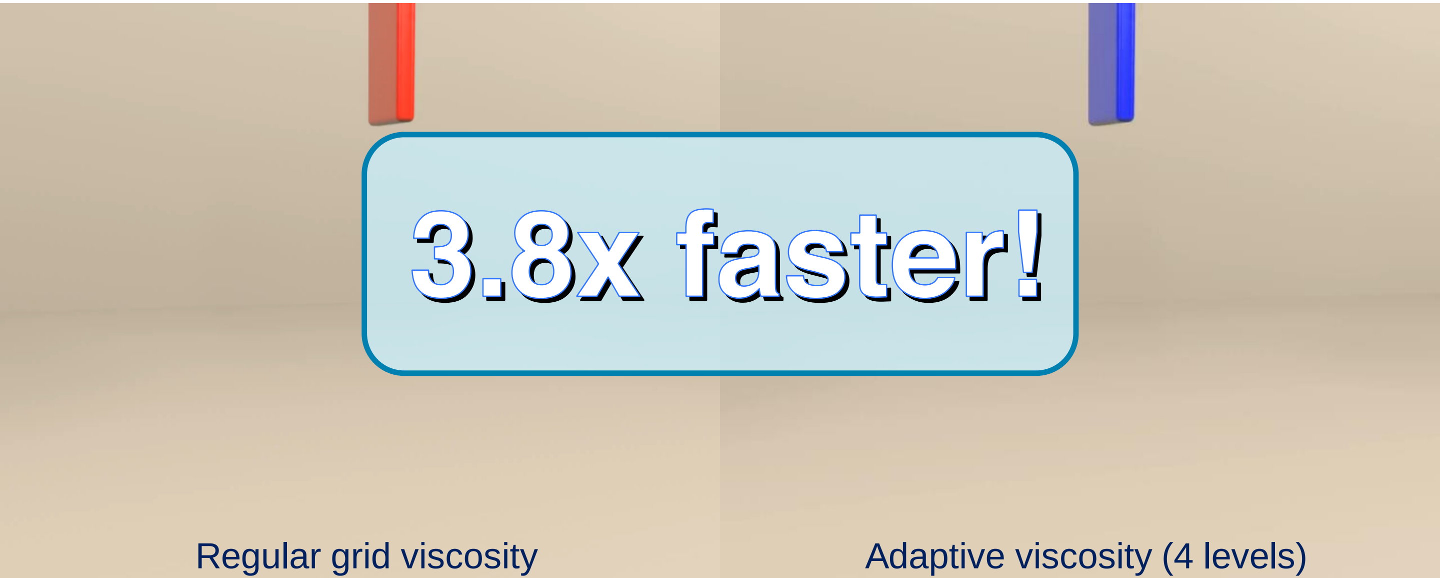
$$\text{e.g., } \tau_{xz} = \mu\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right)$$

Benefits of Variational Form

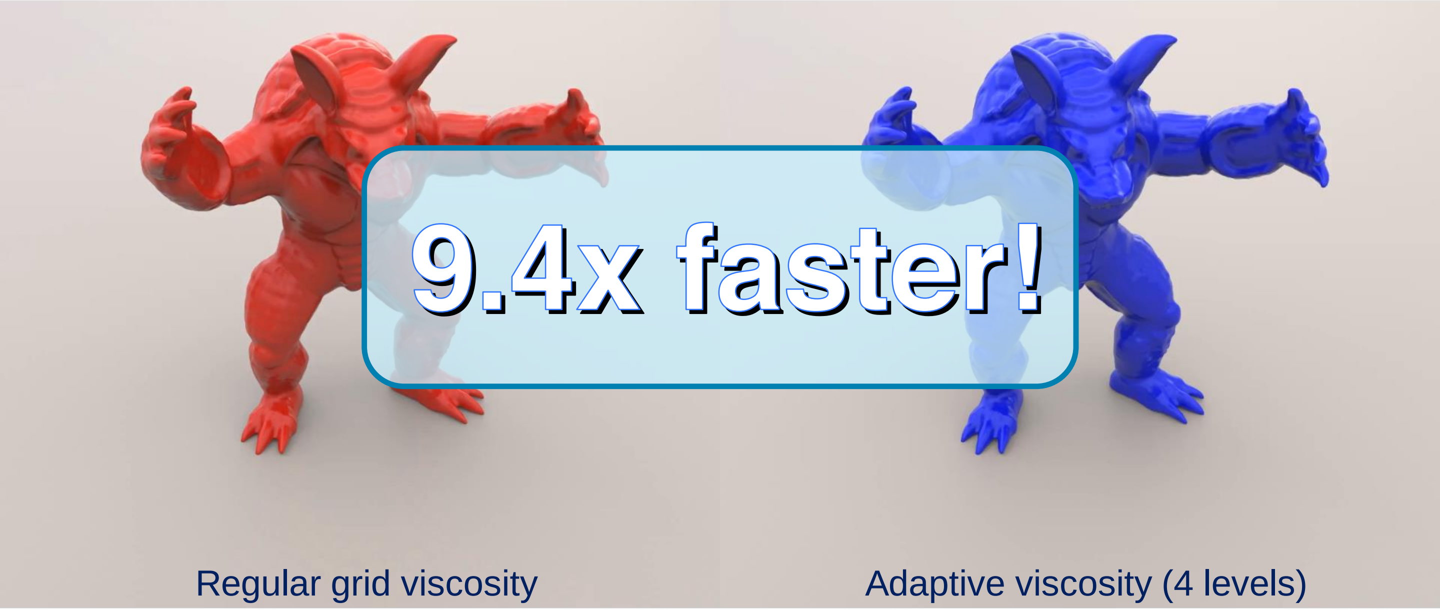
…versus naïve octree finite difference:

- Symmetry preserved automatically.
- Only discretize one derivative operator (vector gradient), rather than two (vector gradient **and** tensor divergence).

Viscous Buckling Comparison



Gooey Armadillo Comparison



Animation speed shown proportional to simulation time.

Our Work

1. Improve boundaries to allow rotation & recover buckling [Batty et al. 2008] ✓
2. Introduce octree discretization for efficiency [Goldade et al. 2019] ✓
3. Avoid splitting error to recover coiling [Larionov et al. 2017]

Theme: Exploit variational form of the viscosity PDE(s) to simplify discretization.

What about coiling?

With the method shown so far, only semi-random buckling occurs.

The expected consistent cylindrical coiling behavior.

Remaining issue

Our earlier operator splitting treats pressure and viscosity **separately**, applying zero traction to each.

1. Advect velocities
2. Integrate body forces
3. Integrate viscosity
4. Enforce incompressibility

However…

Correct zero traction boundaries require pressure and viscous stress to **balance**!

Full Navier-Stokes equations

Momentum: $\rho \frac{Du}{Dt} = -\nabla p + \nabla \cdot \tau$

Incompressibility: $\nabla \cdot u = 0$

Viscous Stress: $\tau = \mu(\nabla u + \nabla u^T)$

Total fluid stress is $\sigma = -pI + \tau$

Traction-free condition is $(-p\boldsymbol{I} + \boldsymbol{\tau})\boldsymbol{n} = 0$

Solution – Stokes solver

Don't split! (as much)

Solve for pressure and viscosity **simultaneously** (i.e., solve an unsteady Stokes problem).

$$\rho \frac{\partial \boldsymbol{u}}{\partial t} = -\nabla p + \frac{1}{\rho} \nabla \cdot \mu (\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T)$$

subject to $\nabla \cdot \boldsymbol{u} = 0$

with $(-p\boldsymbol{I} + \boldsymbol{\tau})\boldsymbol{n} = \boldsymbol{0}$ on $\partial\Omega$.

Solution – Stokes solver

This form…

$$\max_{p,\tau} \min_{\vec{u}} \iiint_{\Omega_L} \frac{\rho}{2}\|\vec{u} - \vec{u}^*\|^2 - \Delta t p \nabla \cdot \vec{u} + \Delta t \tau : \left(\frac{\nabla \vec{u} + (\nabla \vec{u})^T}{2} \right) - \frac{\Delta t}{4\mu}\|\tau\|_F^2$$

…*naturally* gives correct coupled boundary conditions, $(-pI + \tau)\boldsymbol{n} = 0$.

Again, apply our variational / finite difference approach again to discretize.

Modified simulation loop:

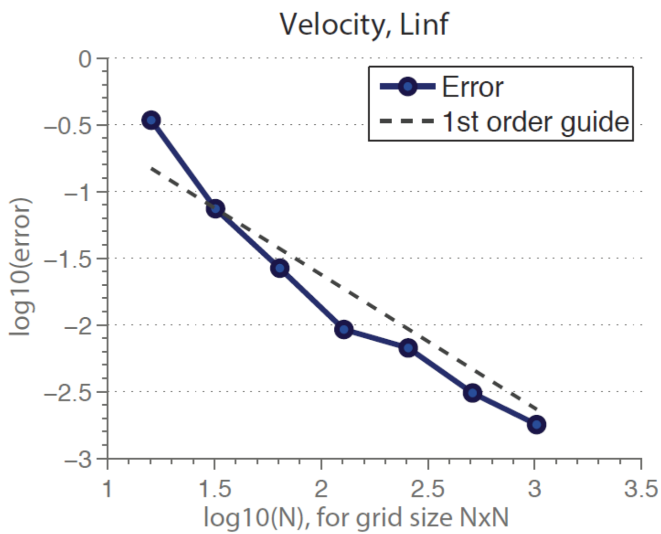
Only split NSE into 3 steps (not 4):

1) Advection
2) Apply body forces
3) Perform Stokes solve (viscosity and pressure together)

Results

Stokes Convergence: ~1st order in u

Grid	$\|u - u^h\|_\infty$	Order
16^2	3.4171E-001	
32^2	7.4246E-002	2.20
64^2	2.6593E-002	1.48
128^2	9.2292E-003	1.53
256^2	6.7182E-003	0.46
512^2	3.0843E-003	1.12
1024^2	1.7877E-003	0.79



Summary – Viscous Liquids

We developed a simple 3D Stokes-based finite difference solver to qualitatively capture viscous behaviours, that…

- Keeps the usual staggered regular grid.
- Stably avoids splitting errors.
- Recovers **folding** and **coiling** motion.
- Yields symmetric linear systems.
- Can be extended to adaptive grids.

Cost vs. (Visual) Quality Tradeoffs

Improved quality (i.e., coiling) is **very** costly…

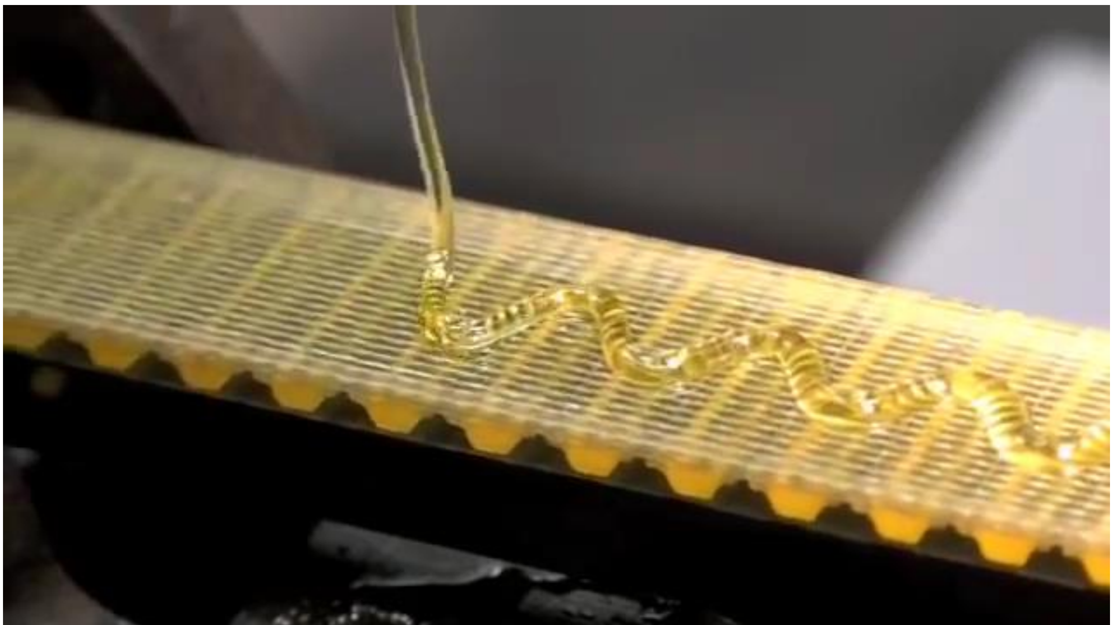
Considering both pressure & viscosity steps:

- Naïve diffusion form (2002) is 4 NxN SPD systems.
- Decoupled viscosity/pressure (2008) gives a 3Nx3N and a NxN SPD system.
- Our coupled Stokes form (2017) gives one 6Nx6N SPD system for pressure & stress.
- (cf. standard Stokes is a 4Nx4N indefinite system.)

Can one split pressure & viscosity and still recover coiling?

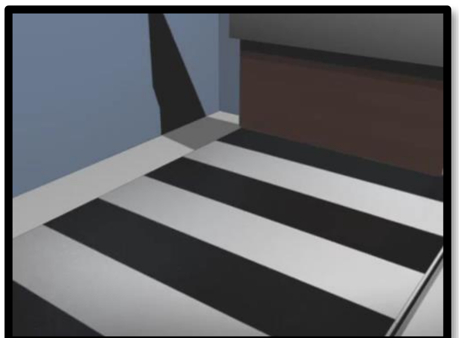
Viscous Sewing Machine

Rod-based models can recover all of these patterns – we haven't yet.

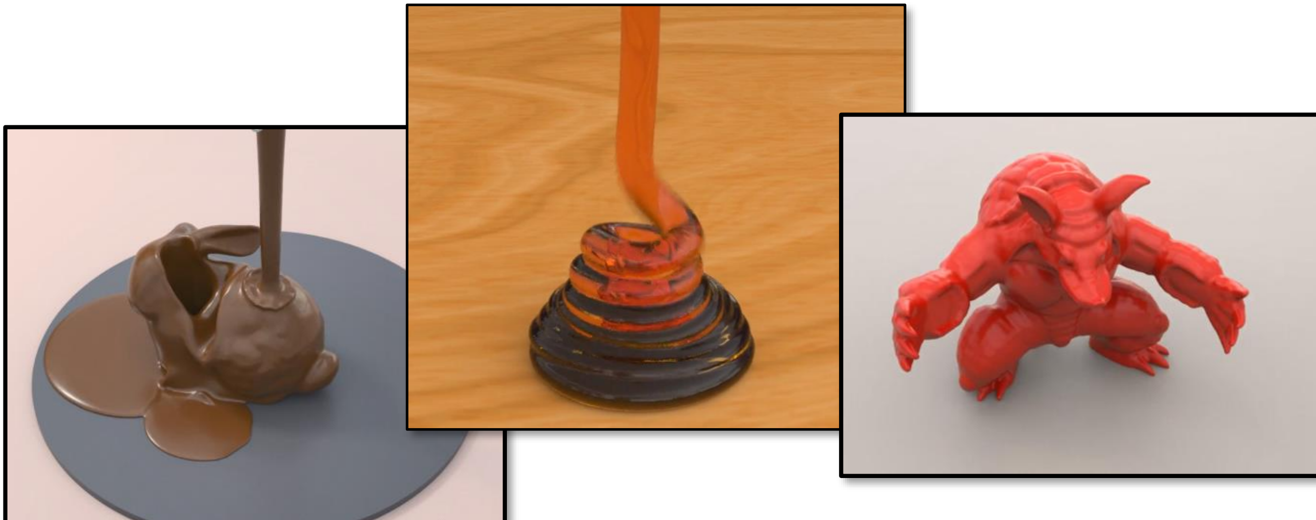


Hybrid Model

Could one combine thread, sheet, and volumetric models for efficiency?



Thanks!



Collaborators:

Mridul Aanjaneya, Robert Bridson, Ryan Goldade, Egor Larionov, & Yipeng Wang.

.

Funding & Equipment:

NSERC, Side Effects Software.