COL781: Computer Graphics

36. Solids and Fluids

Fei et al. 2019

Announcements

Assignment 3 demos postponed to next week (Mon-Wed afternoons),
sign up again on the same sheet

Assignment 4 deadline extended to Monday, 22 April

Back to the deformable rod

- Positions: $\mathbf{x}(s)$ where $s \in [0, L]$
- Mass of differential segment: $\mathrm{d}m = \rho\, \mathrm{d}s$
- Relative length of segment: $\|\mathbf{x}(s+\mathrm{d}s) - \mathbf{x}(s)\|/\mathrm{d}s = \|\partial\mathbf{x}/\partial s\|$
- **Strain** $\varepsilon = \|\partial\mathbf{x}/\partial s\| - 1$

positions $\mathbf{x}(s)$ → strain ε → tension τ → force d$\mathbf{f}$
→ PDE $\mathrm{d}^2\mathbf{x}/\mathrm{d}t^2 = \cdots$ → discretize

Simpler approach:

positions $\mathbf{x}(s)$ → strain ε → potential energy U → discretize!

$$U = \int_0^L \frac{1}{2} k \epsilon^2 \,\mathrm{d}s$$

Discretize space:

- Sample points $s_0, s_1, \ldots, s_N$ with positions $\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_N$
- $\dfrac{\partial \mathbf{x}}{\partial s} \approx \dfrac{\mathbf{x}_{i+1} - \mathbf{x}_i}{\Delta s}$
- $U = \displaystyle\int_0^L \frac{1}{2} k \epsilon^2 \, \mathrm{d}s \approx \sum_{i=0}^{N-1} \frac{1}{2} k \left(\left\| \frac{\mathbf{x}_{i+1} - \mathbf{x}_i}{\Delta s} \right\| - 1 \right)^2 \Delta s$

The elastic energy is a sum of several terms, each corresponding to a connection between adjacent particles *i* and *i*+1.

…We've just reinvented mass-spring systems!

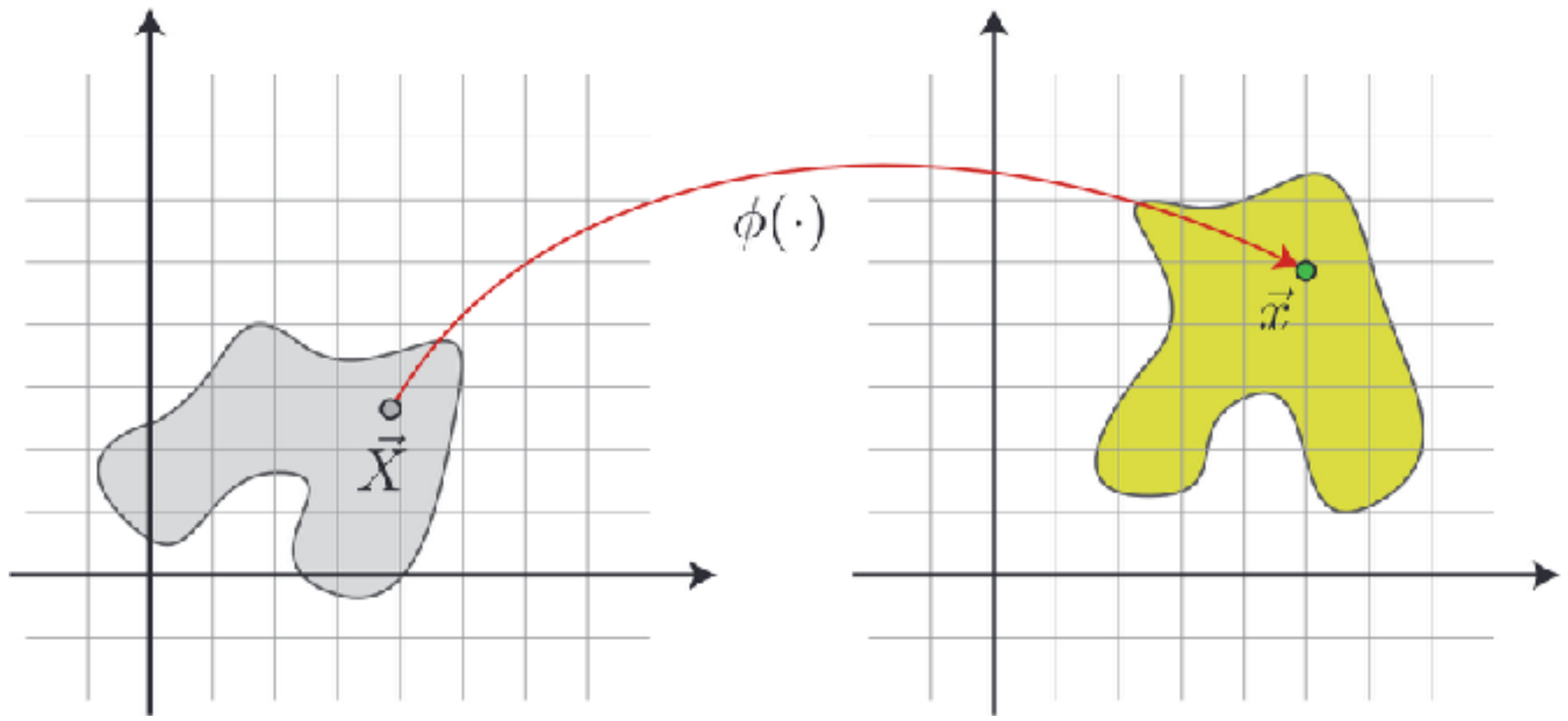
Now we can derive the equations of motion as usual:

- Net force on particle $i = -\frac{\partial U}{\partial \mathbf{x}_i} = -\sum_{j=0}^{N-1} \frac{\partial}{\partial \mathbf{x}_i}\left(\cdots\right)$
- $\frac{\mathrm{d}^2\mathbf{x}_i}{\mathrm{d}t^2} = -m_i^{-1}\frac{\partial U}{\partial \mathbf{x}_i}$

What was the point?

- Behaves consistently with resolution (changing number of particles *N*)
- Generalizes naturally from 1D rods to 2D sheets and 3D volumes!

Elastic deformation



- Deformation is a map **x**(**X**) from the rest shape defined in a reference domain

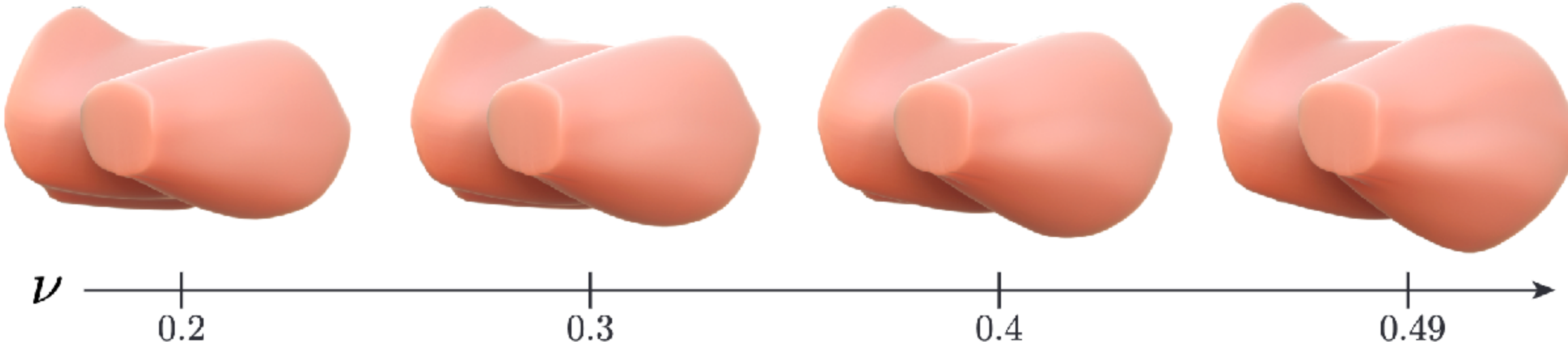
- Amount of stretch is described by its Jacobian: the deformation gradient **F**

$$\mathbf{F} = \begin{bmatrix} \frac{\partial \mathbf{x}}{\partial X_1} & \frac{\partial \mathbf{x}}{\partial X_2} & \cdots \end{bmatrix}$$

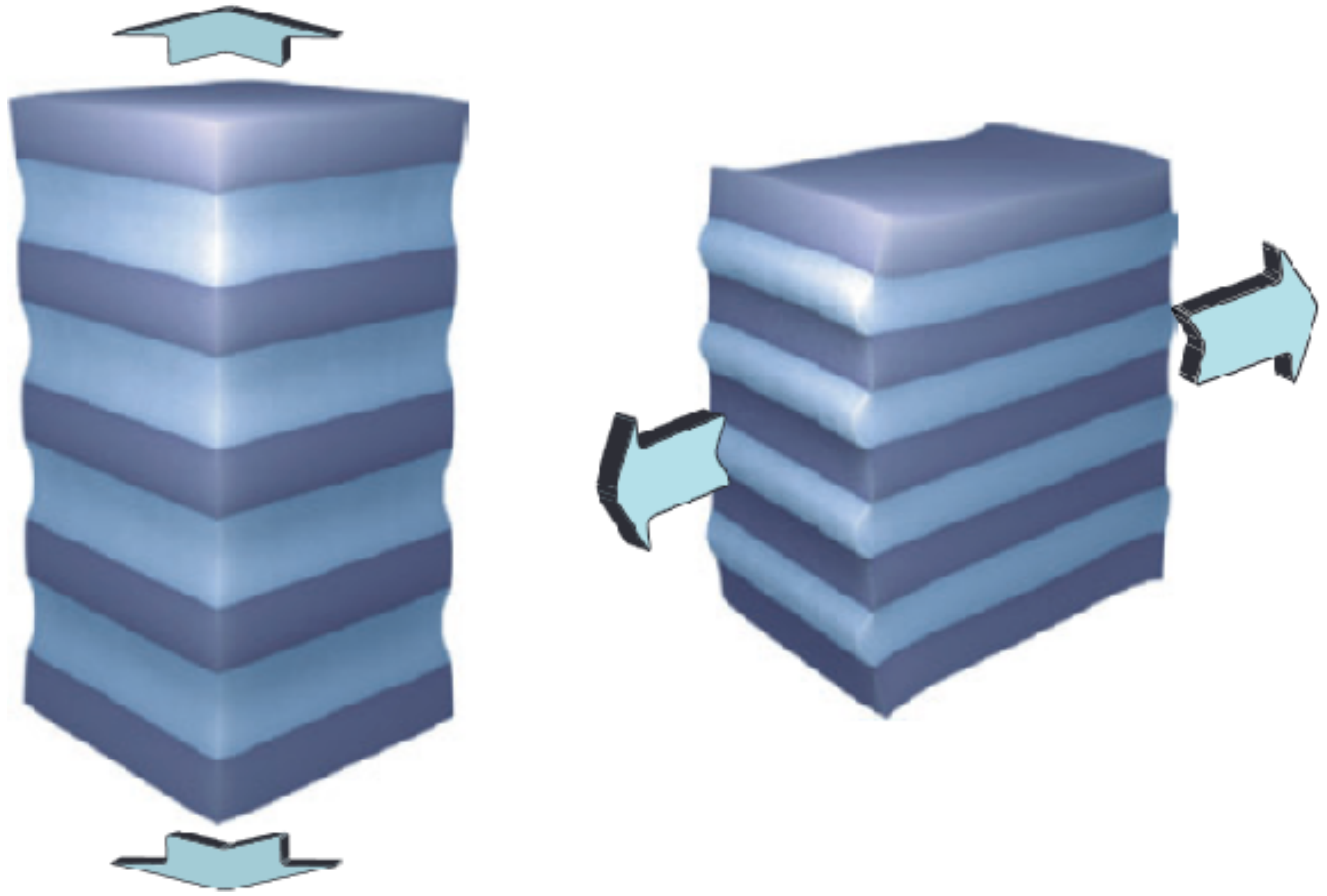
- Elastic energy is given by a (material-dependent) strain energy density function Ψ

$$U = \iiint \Psi(\mathbf{F})\,\mathrm{d}V$$

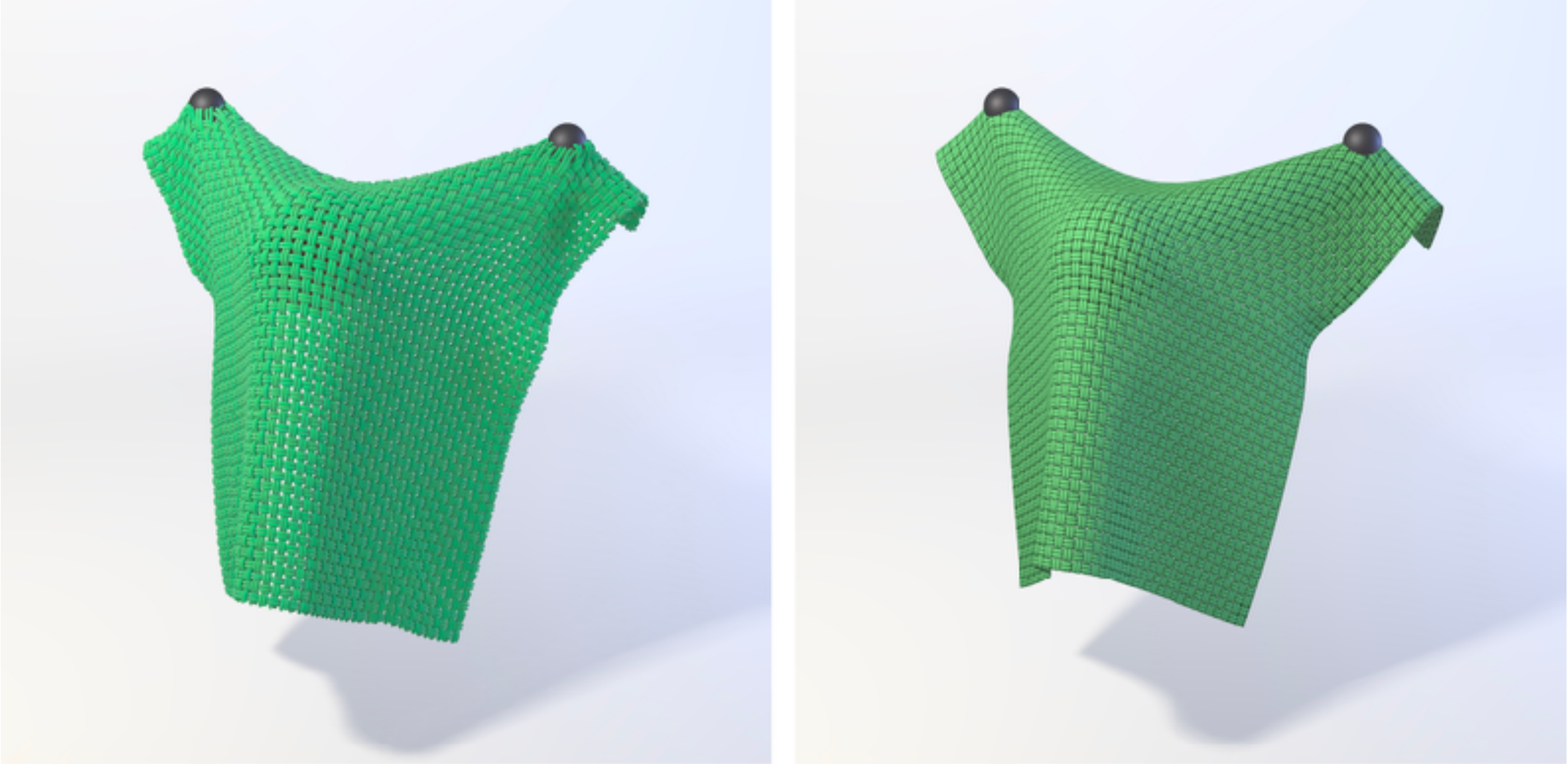
Choice of strain energy density $\Psi(\mathbf{F})$ determines material behaviour, including volume preservation (Poisson's ratio), anisotropy, and all other effects



Smith et al. 2018



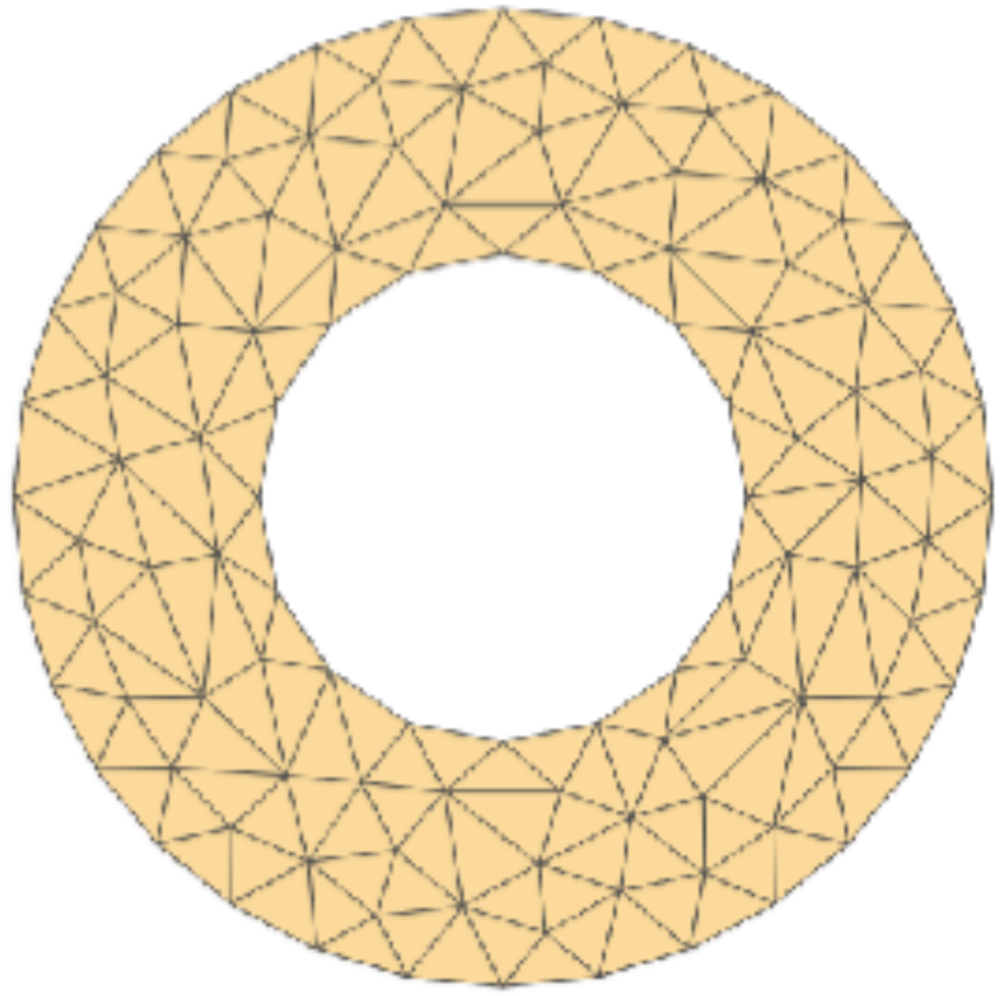
Kharevych et al. 2009

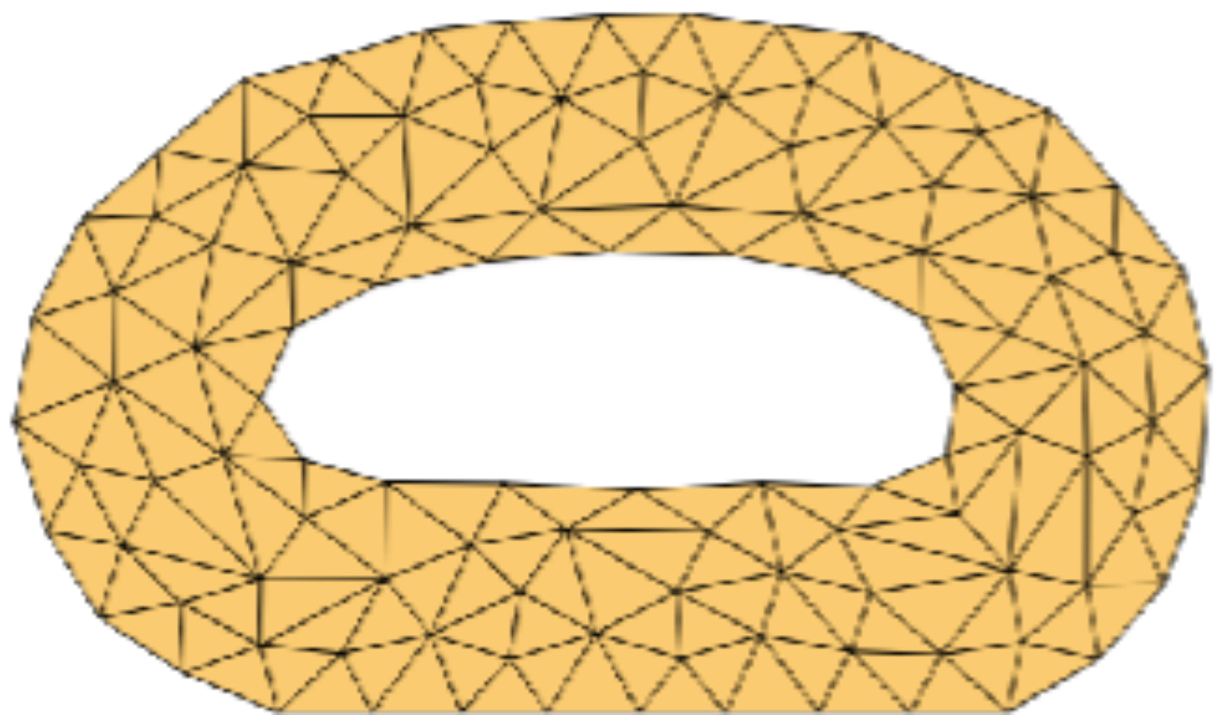


Sperl et al. 2020

The finite element method

- Discretize the reference domain using a mesh
- On each element (triangles in 2D, tetrahedra in 3D) interpolate **x**(**X**) and compute **F** = d**x**/d**X**
- Total energy $U = \sum_{\text{element } j} \Psi(\mathbf{F}_j)\, V_j$
- Then proceed as usual!





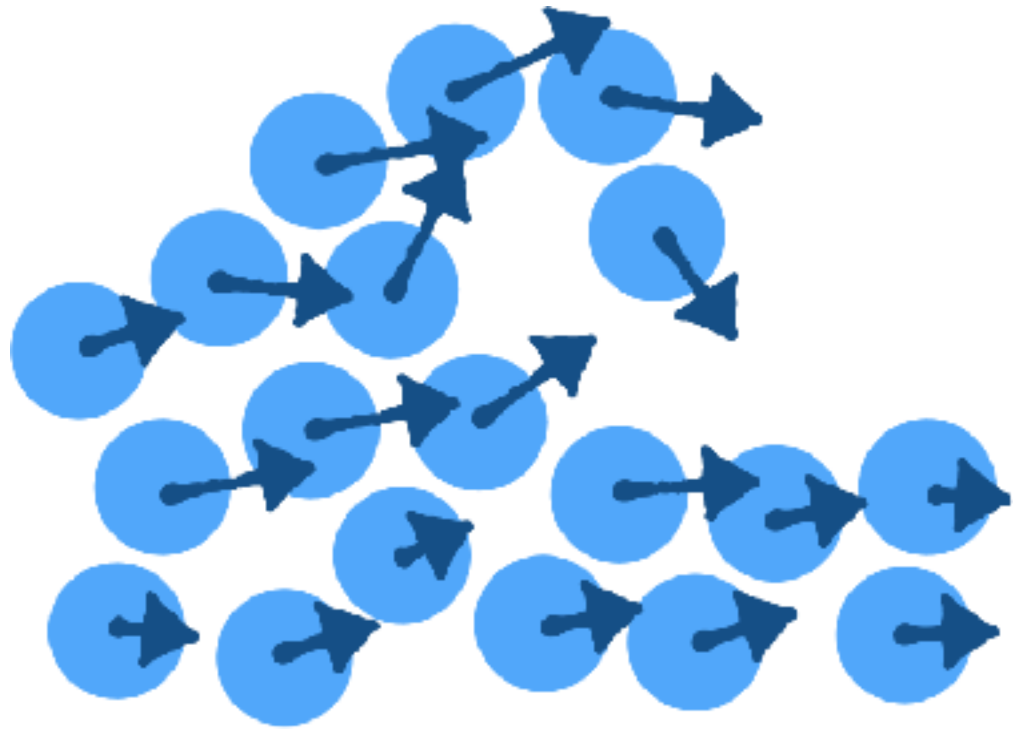
(This is just the tip of the iceberg regarding FEM. But enough to make it work!)

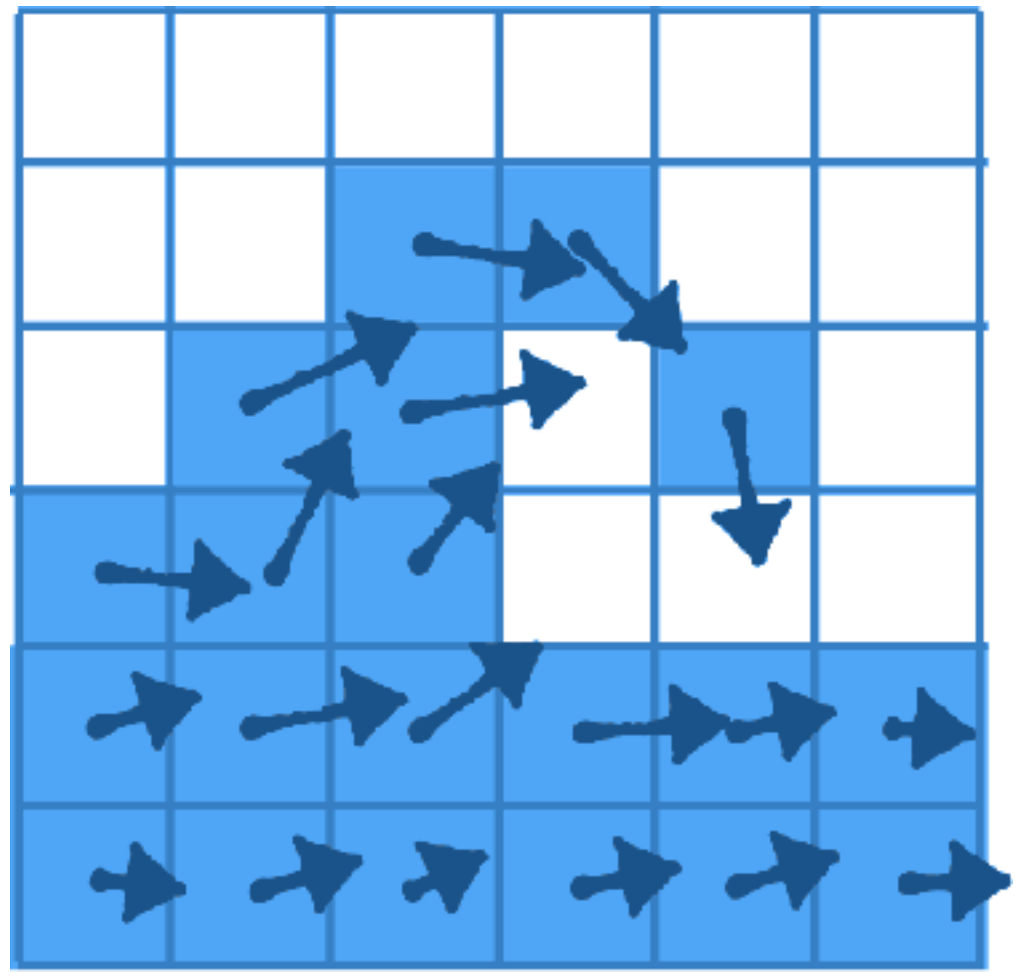
Fluids

No rest shape, so no reference space **X** needed.
No deformation map **x**(**X**), no time derivative **v**(**X**) = $\dot{\mathbf{x}}$(**X**)

Still need **v** as a function of **x** though: the velocity field

Can discretize using particles or a grid:



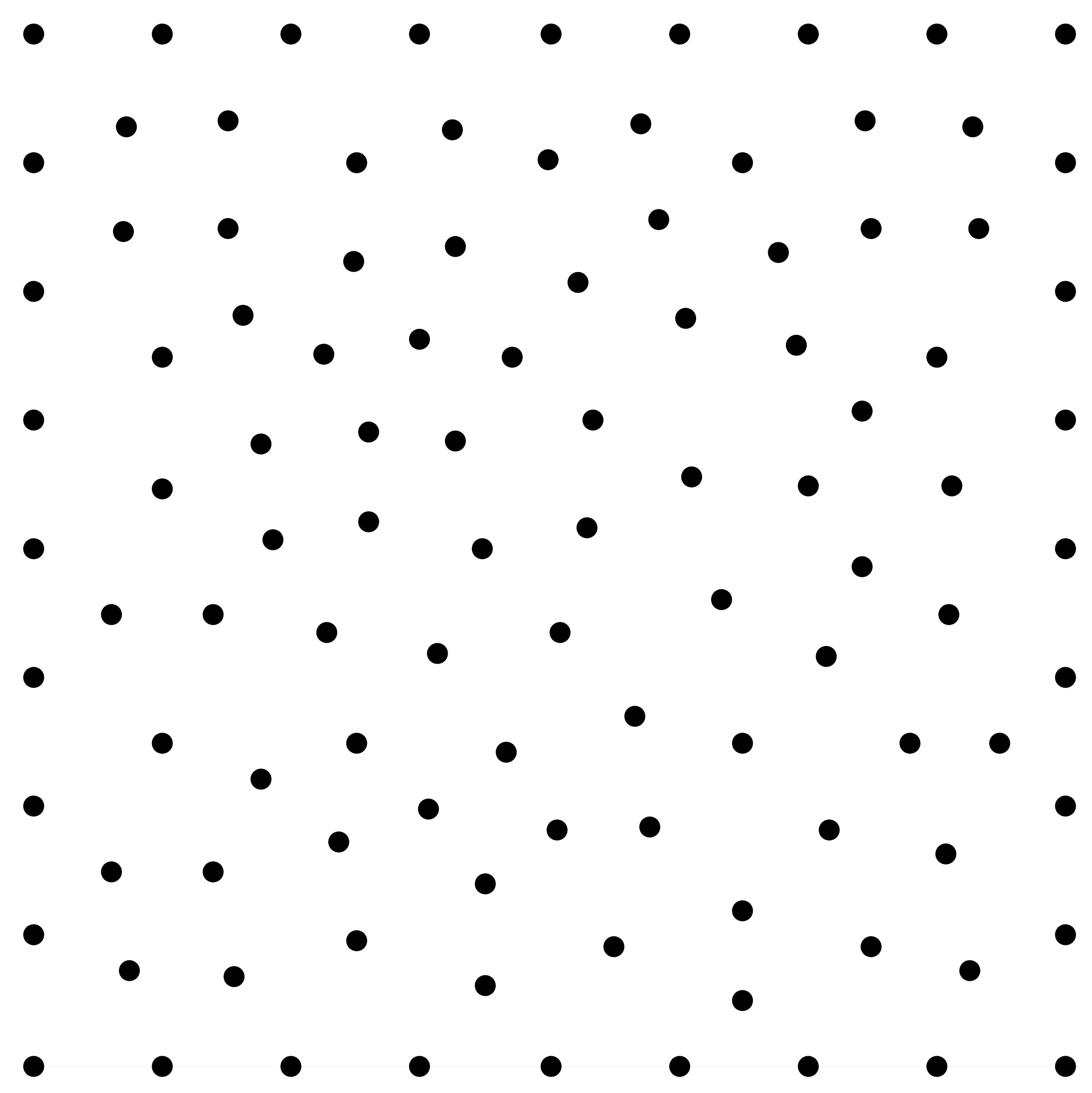


Let's just discretize space using a collection of particles

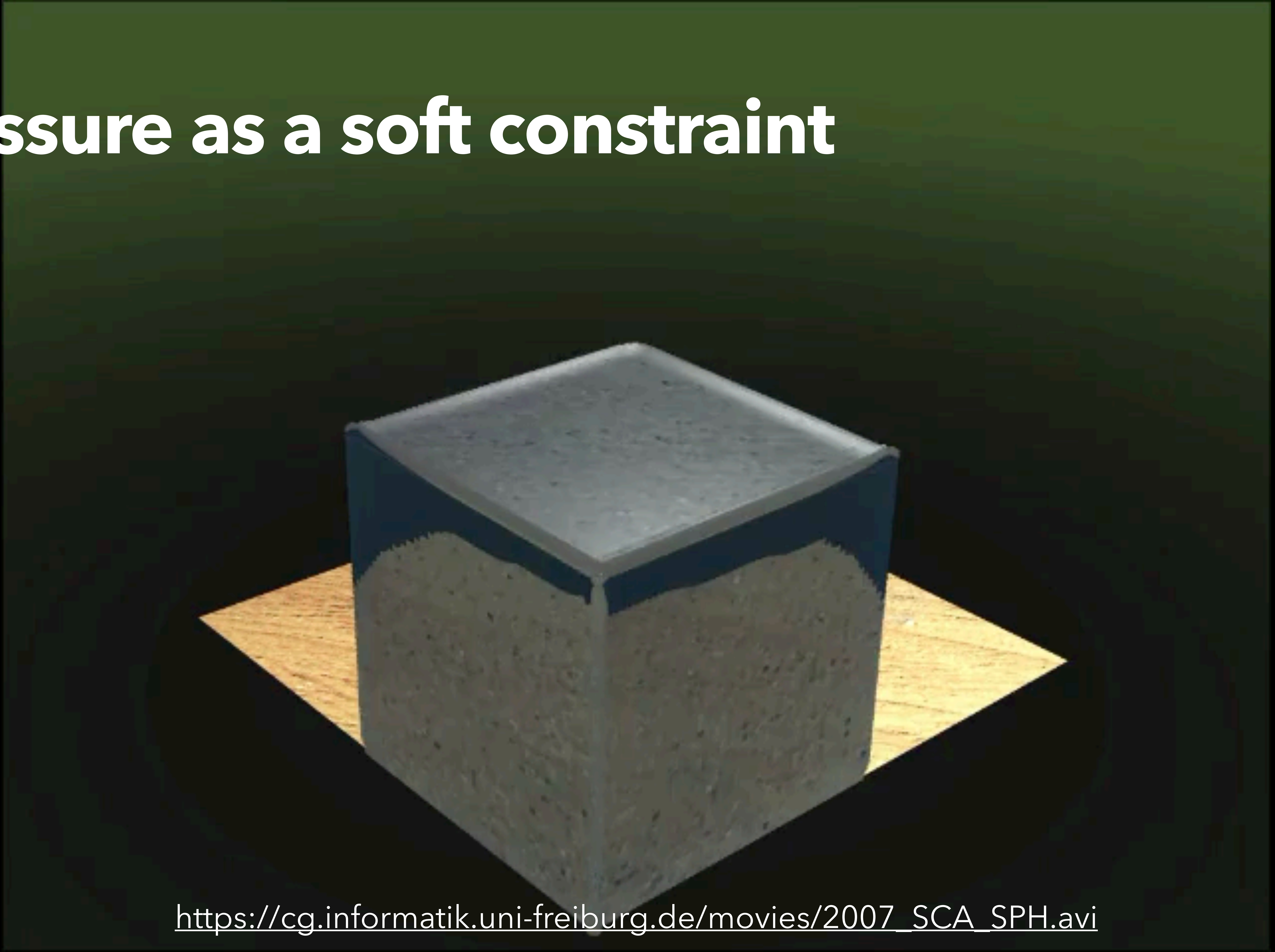
- Velocity field $\mathbf{v}(\mathbf{x}) \to$ samples $\mathbf{v}_1, \mathbf{v}_2, \ldots$ at $\mathbf{x}_1, \mathbf{x}_2, \ldots$
- Also let the particles move with velocity $\mathbf{v}_i$

What are the forces/constraints acting on the fluid?

- Most important: density $\rho(\mathbf{x}) = \text{const}$
- The pressure $p(\mathbf{x})$ is just the corresponding constraint force!



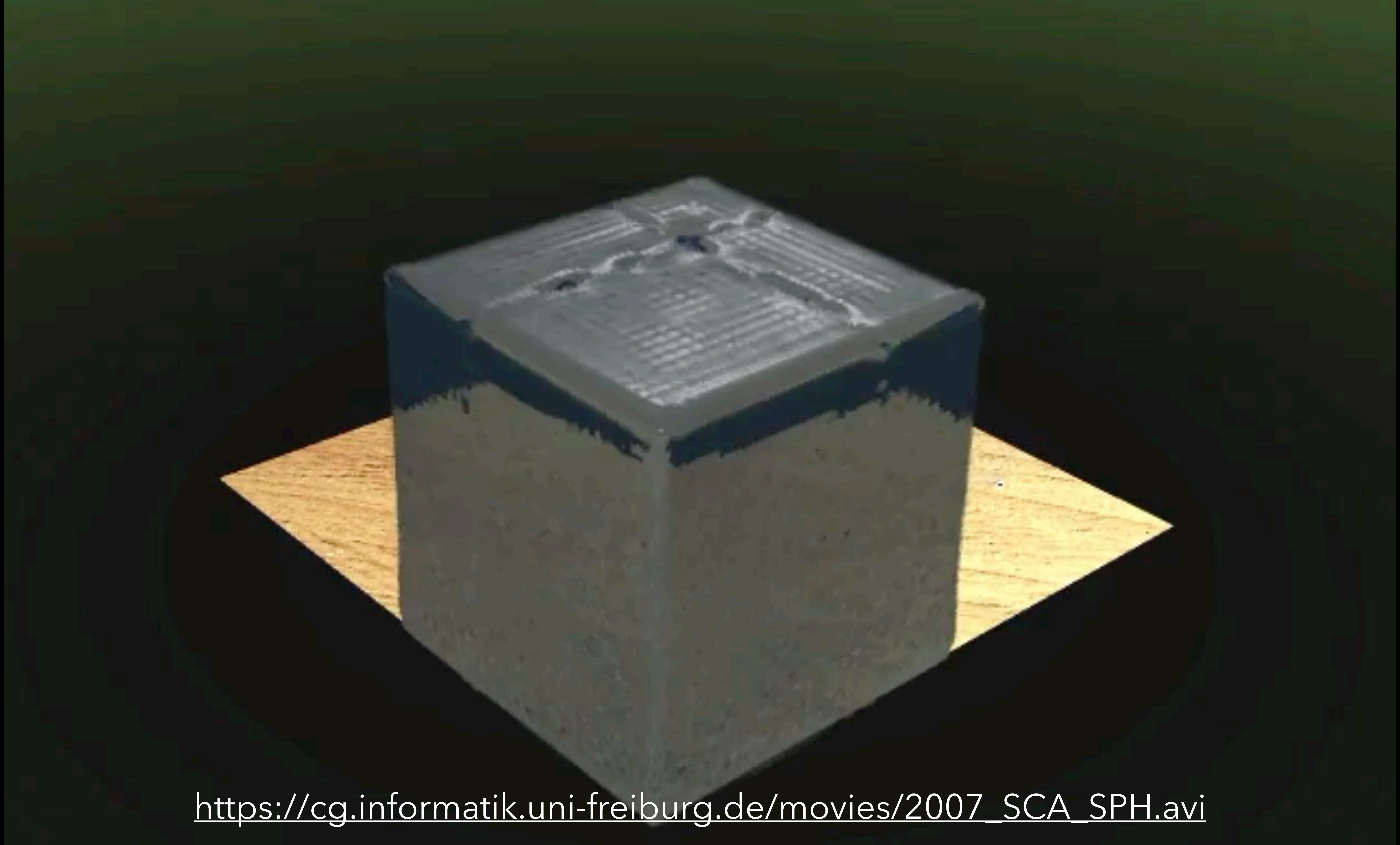
Pressure as a soft constraint



https://cg.informatik.uni-freiburg.de/movies/2007_SCA_SPH.avi

Becker & Teschner 2007

Pressure as a harder constraint



https://cg.informatik.uni-freiburg.de/movies/2007_SCA_SPH.avi

Becker & Teschner 2007

One way to discretize:

- Density at a particle: ρ_i = number of nearby particles $\mathbf{x}_j$

$$= \sum_j w(\|\mathbf{x}_i - \mathbf{x}_j\|)$$

- Density constraint: $c_i(\mathbf{q}) = \sum_j w(\|\mathbf{x}_i - \mathbf{x}_j\|) - \rho_0 = 0$

- Constraint force: $p_i \nabla w$ acting on all nearby particles

This leads to purely particle-based fluid simulation (smoothed particle hydrodynamics, position-based fluids, …)

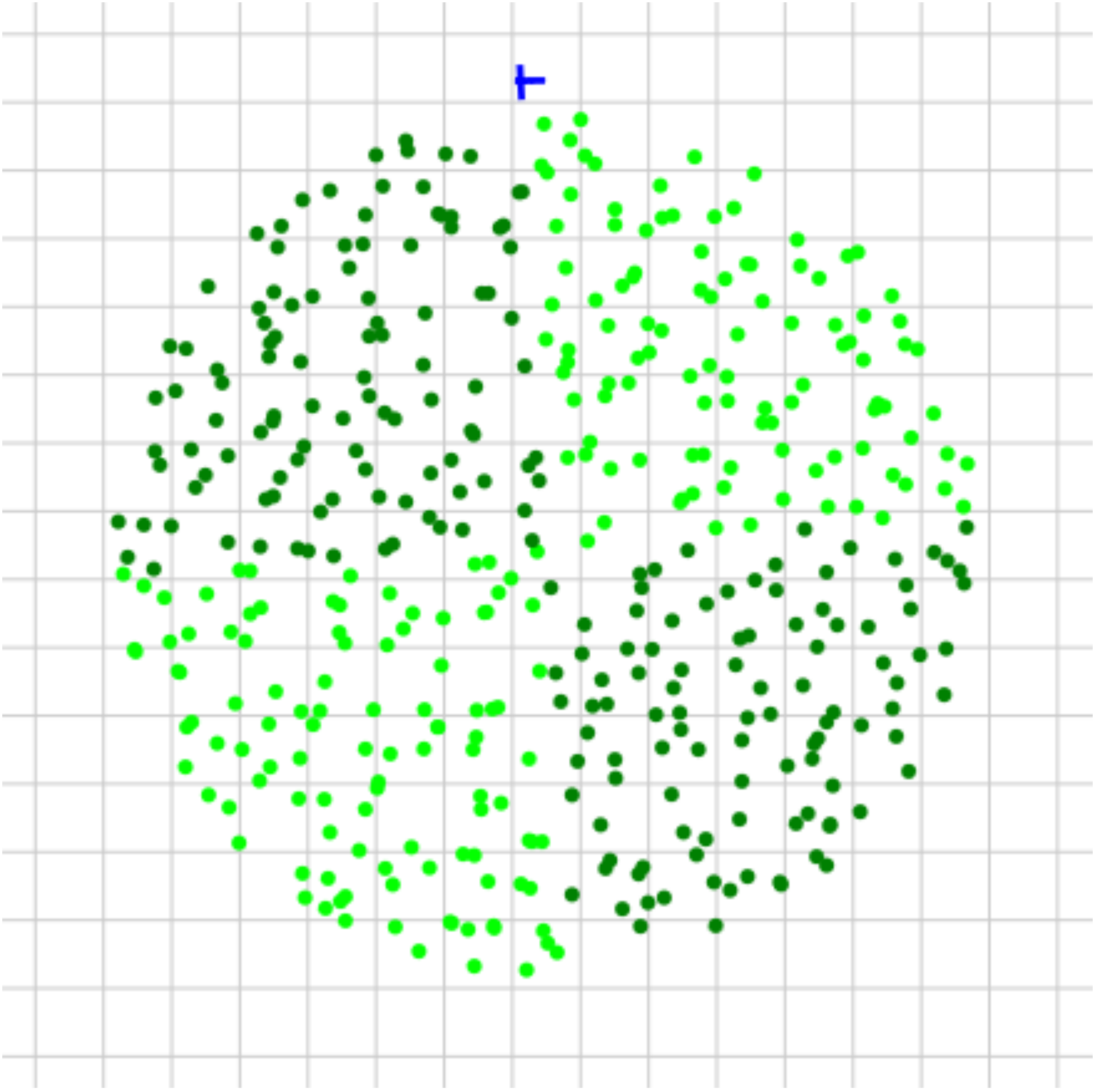
Another idea:

- Density $\rho(\mathbf{x})$ = const $\Rightarrow$ velocity divergence $\nabla \cdot \mathbf{v}(\mathbf{x}) = \frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \cdots = 0$

- There should be no net inflow or outflow in any region

Hybrid / particle-grid / particle-in-cell methods:

- Keeping track of nearest neighbours is expensive, let's compute forces on a grid instead
- At each grid cell, set $\mathbf{v}_{ij}$ = average velocity of nearby particles
- Compute divergence $\nabla \cdot \mathbf{v}$ using finite differences
- Constraint force = $-\nabla p$. Solve for p_{ij} over entire grid so that new velocity has zero divergence: $\nabla \cdot (\mathbf{v} - \nabla p) = 0$
- Interpolate pressure force $-\nabla p$ to particles



Where to learn more

Simulation in general:

- Witkin & Baraff, *Physically Based Modeling* (2001)
- Bargteil & Shinar, *An Introduction to Physics-Based Animation* (2019)

Elastic bodies:

- Kim & Eberle, *Dynamic Deformables* (2022)

Contact handling:

- Andrews et al., *Contact and Friction Simulation for Computer Graphics* (2022)

Fluids:

- Bridson & Müller-Fischer, *Fluid Simulation for Computer Animation* (2007)