Lecture 20:

Introduction to Fluid Simulation

Computer Graphics and Imaging

UC Berkeley CS184/284A

Example: Fluids

Macklin and Müller, Position Based Fluids TOG 2013

Example: Fluids

Macklin and Müller, Position Based Fluids TOG 2013

Problem Setup

Lagrangian Formulation

- Where in space did this material move to?
- Commonly used for solid materials

Eulerian Formulation

- What material is at this location in space?
- Commonly used for fluids
  - Why: Because fluids don’t remember their shape

Problem Discretization

Grids

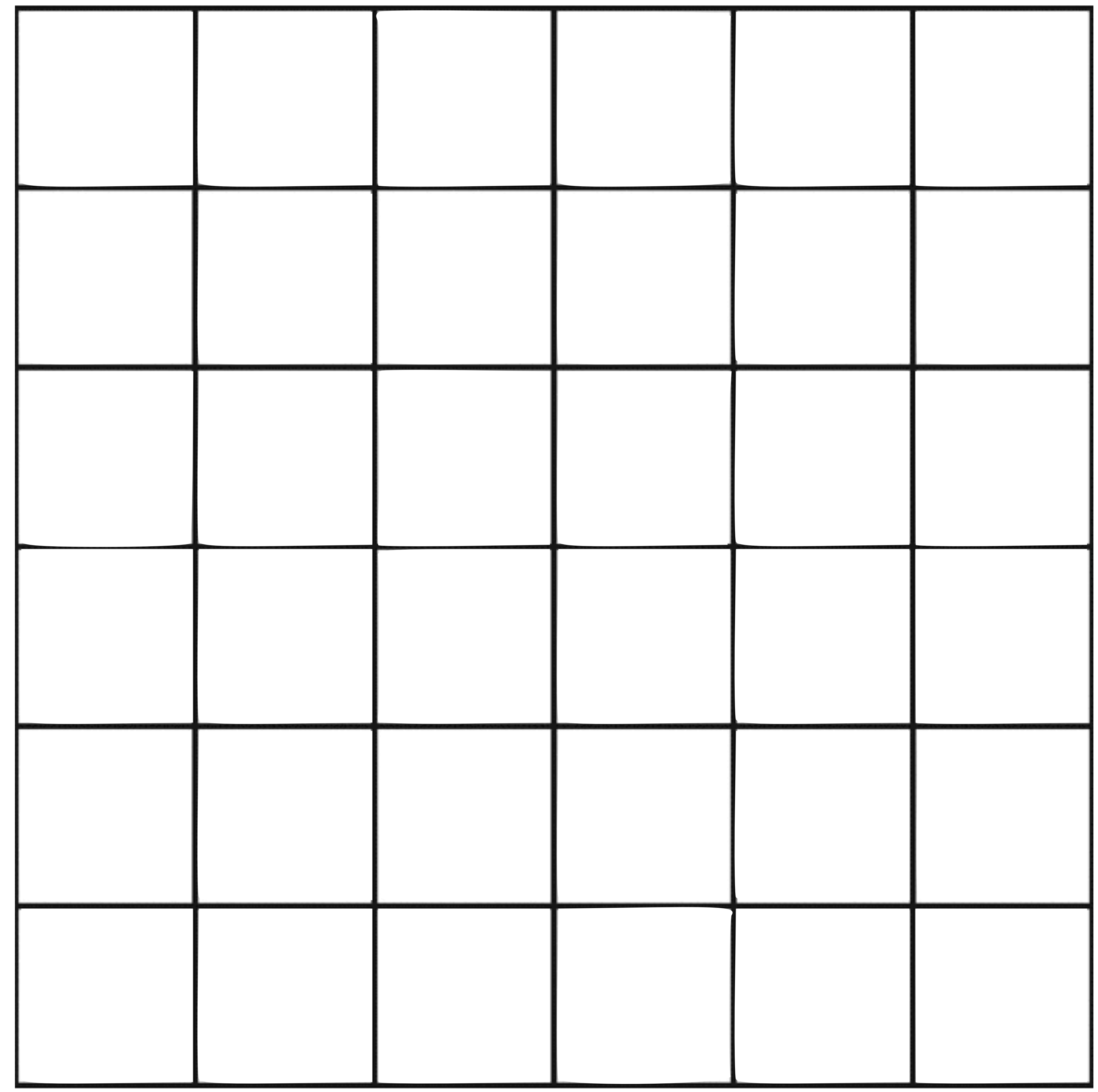
- Store quantities on a grid
- Fluid move "through" grid
- Scales reasonably well to large systems
- Surface tracking is challenging

Particles

- Fluid defined by locations of particles
- Inter-particle forces create fluid behavior
- Scaling to large systems not simple
- Surface tracking less difficult

Many popular methods combine grids and particles

Fluid Grid



Store Fluid State On Grid

- Velocity
- Pressure
- Density

Staggered Grid

- Bilinear interpolation
- Seems odd at first
- Very useful
- Non-staggered produces unstable checkerboard

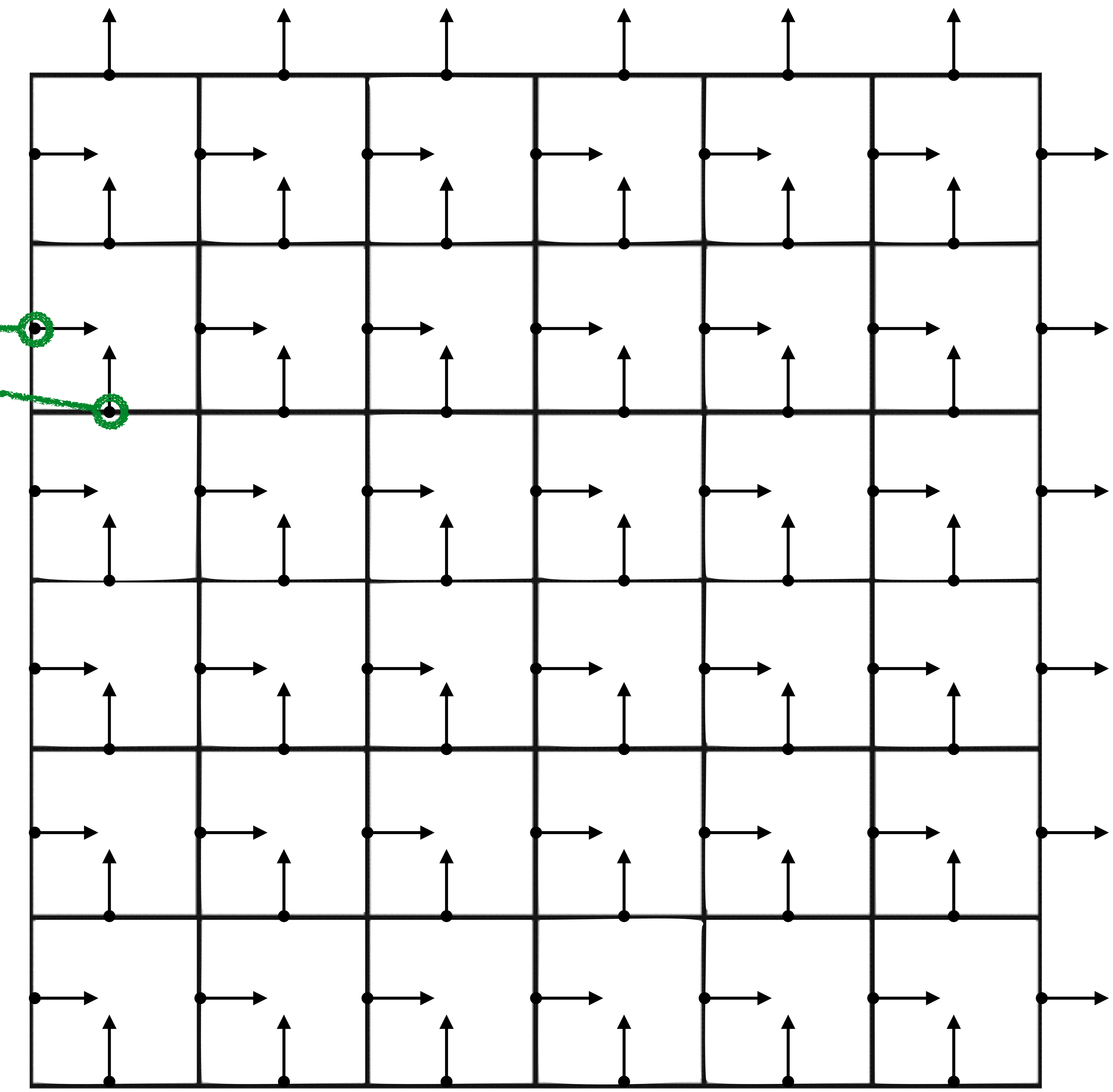
Fluid Grid

Store Fluid State On Grid

- Velocity
- Pressure
- Density

Staggered Grid

- Bilinear interpolation
- Seems odd at first
- Very useful
- Non-staggered produces unstable checkerboard



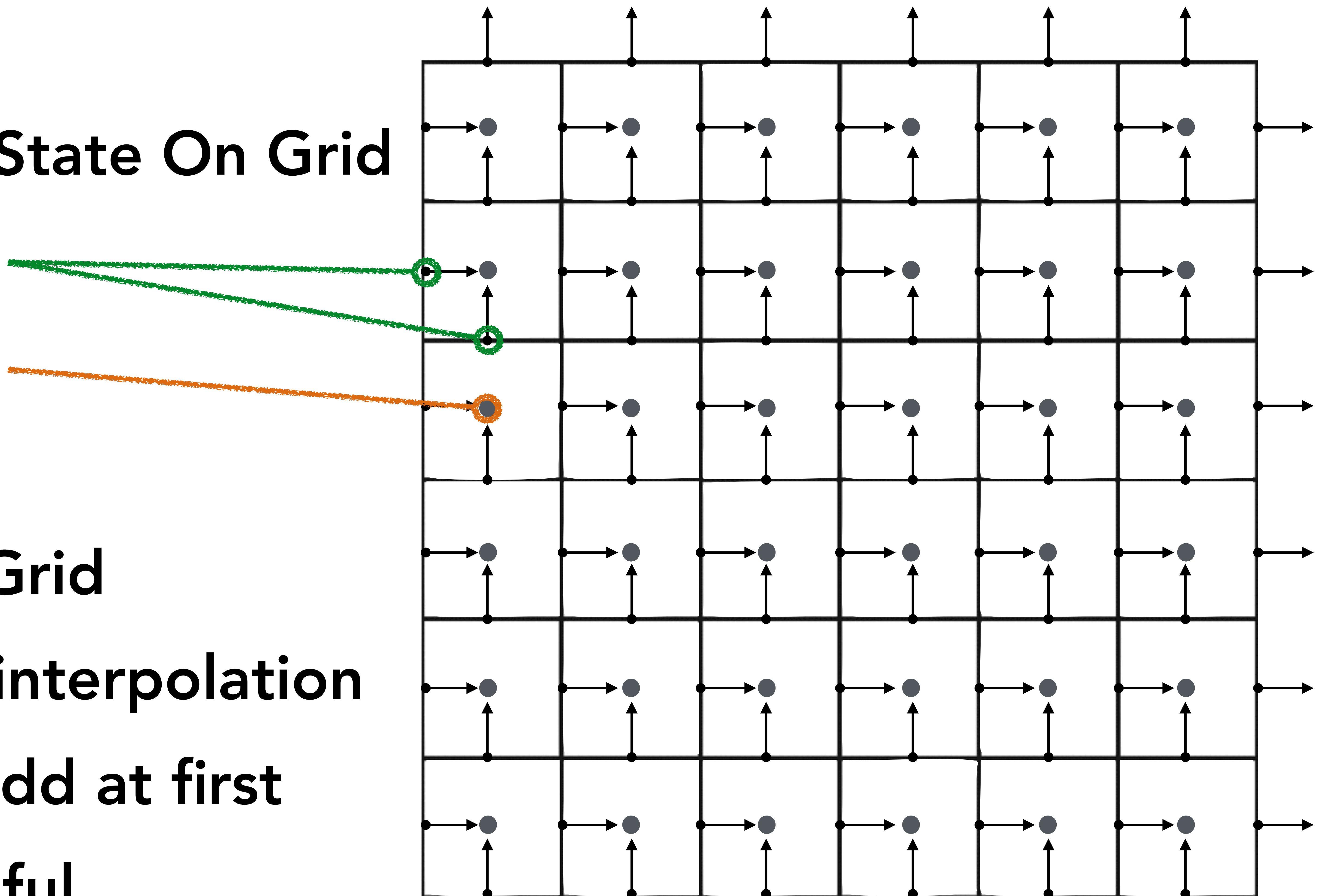
Fluid Grid

Store Fluid State On Grid

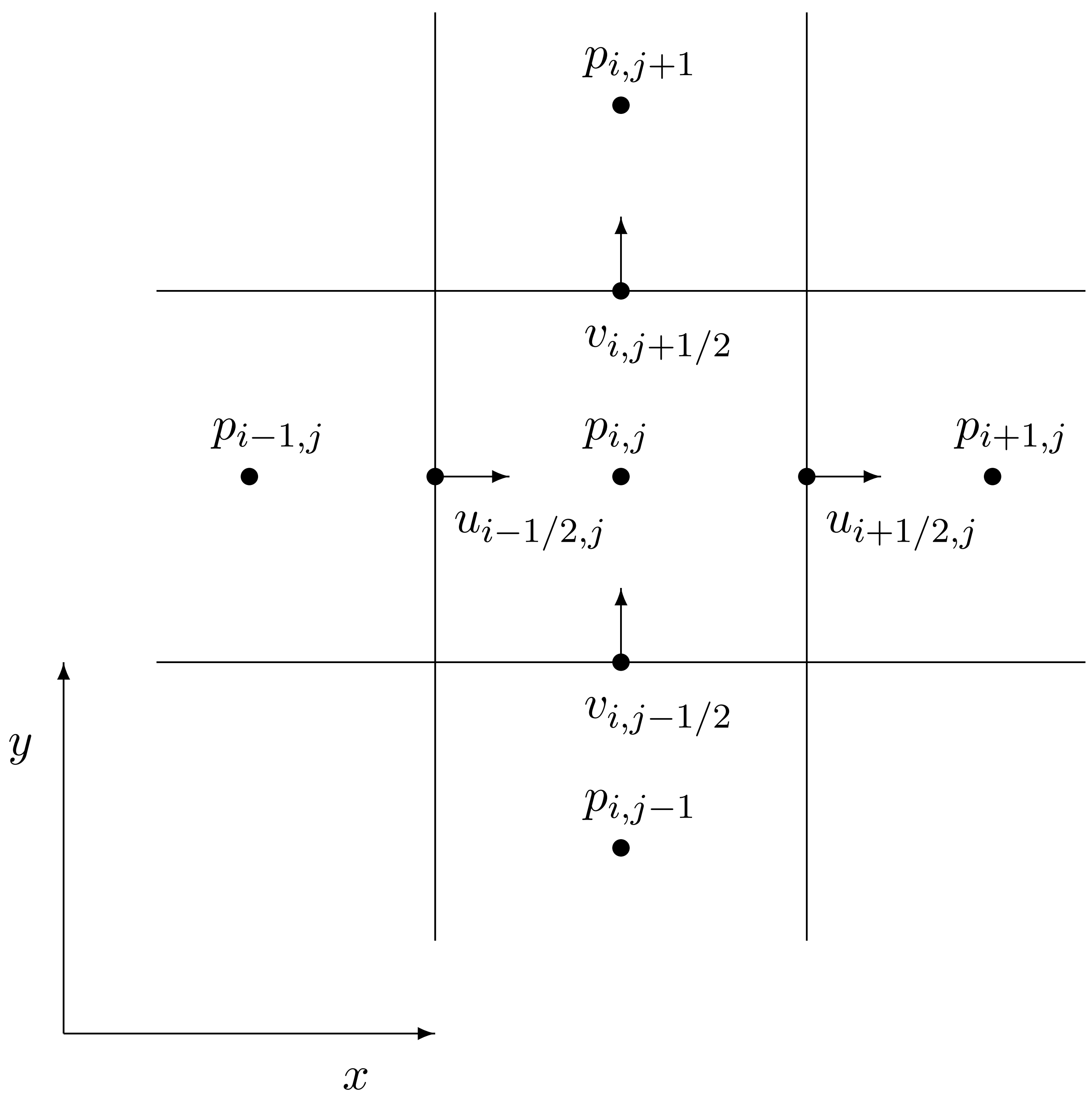
- Velocity
- Pressure
- Density

Staggered Grid

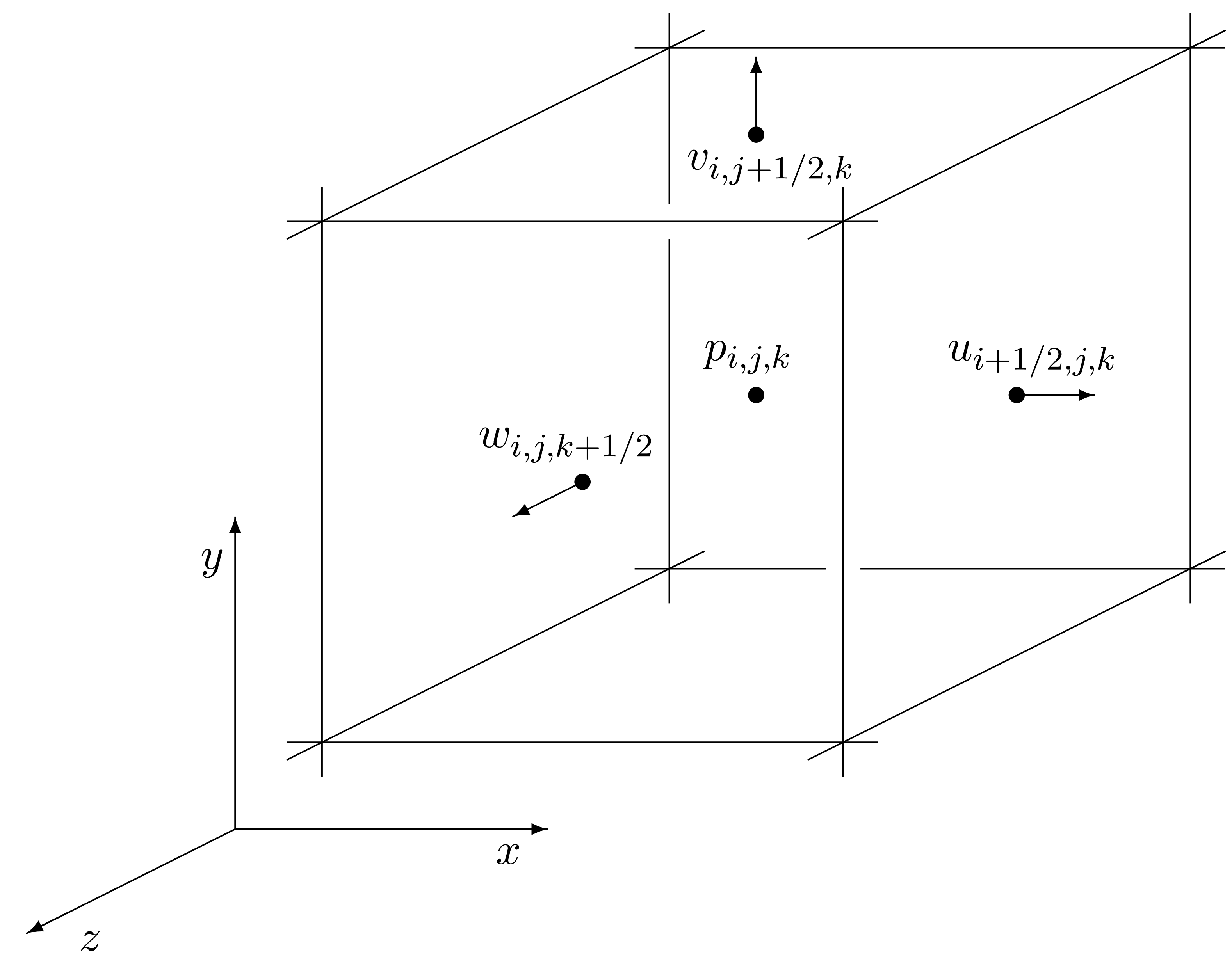
- Bilinear interpolation
- Seems odd at first
- Very useful
- Non-staggered produces unstable checkerboard



Fluid Grid



2D Staggered Grid

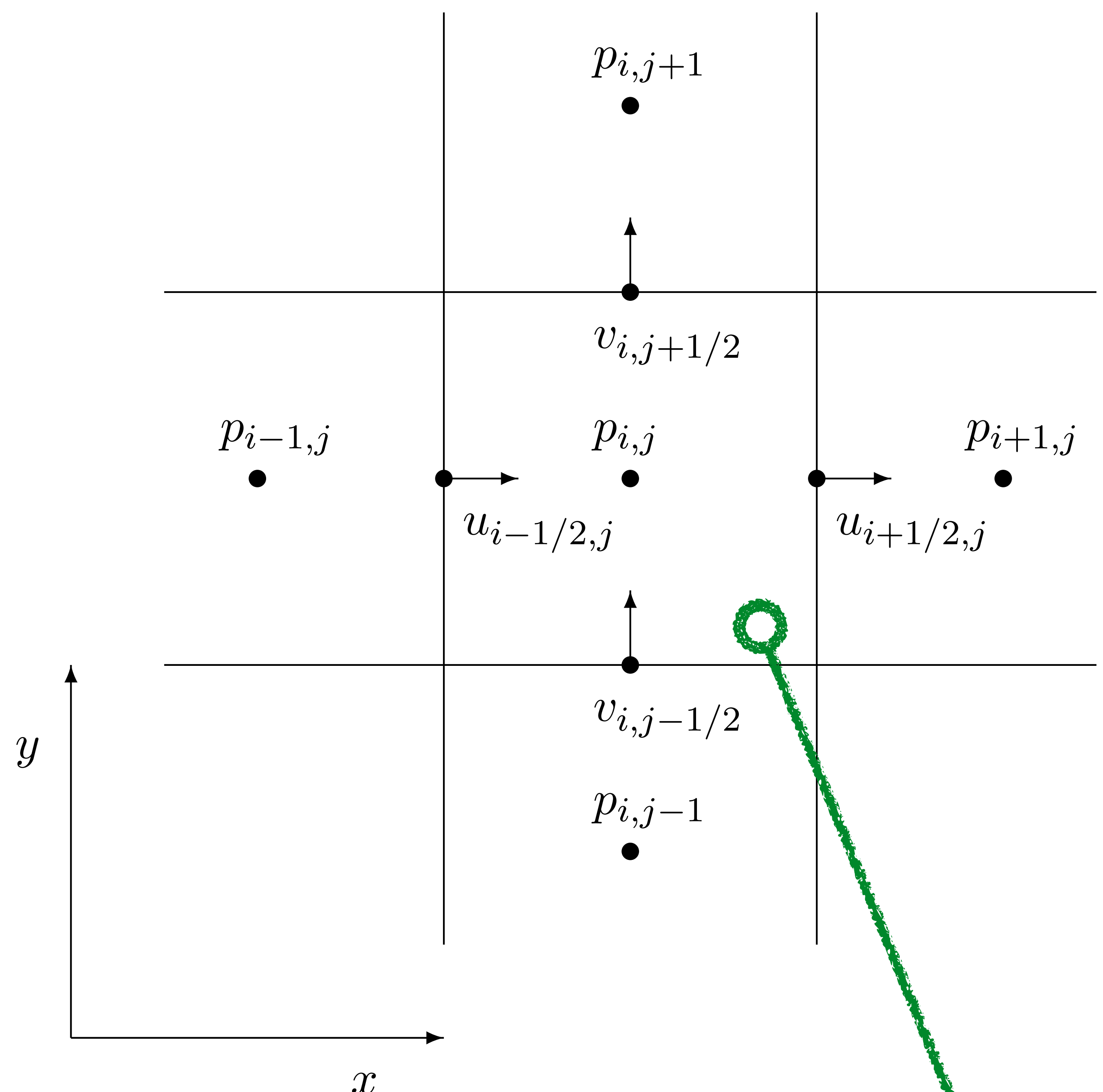


3D Staggered Grid

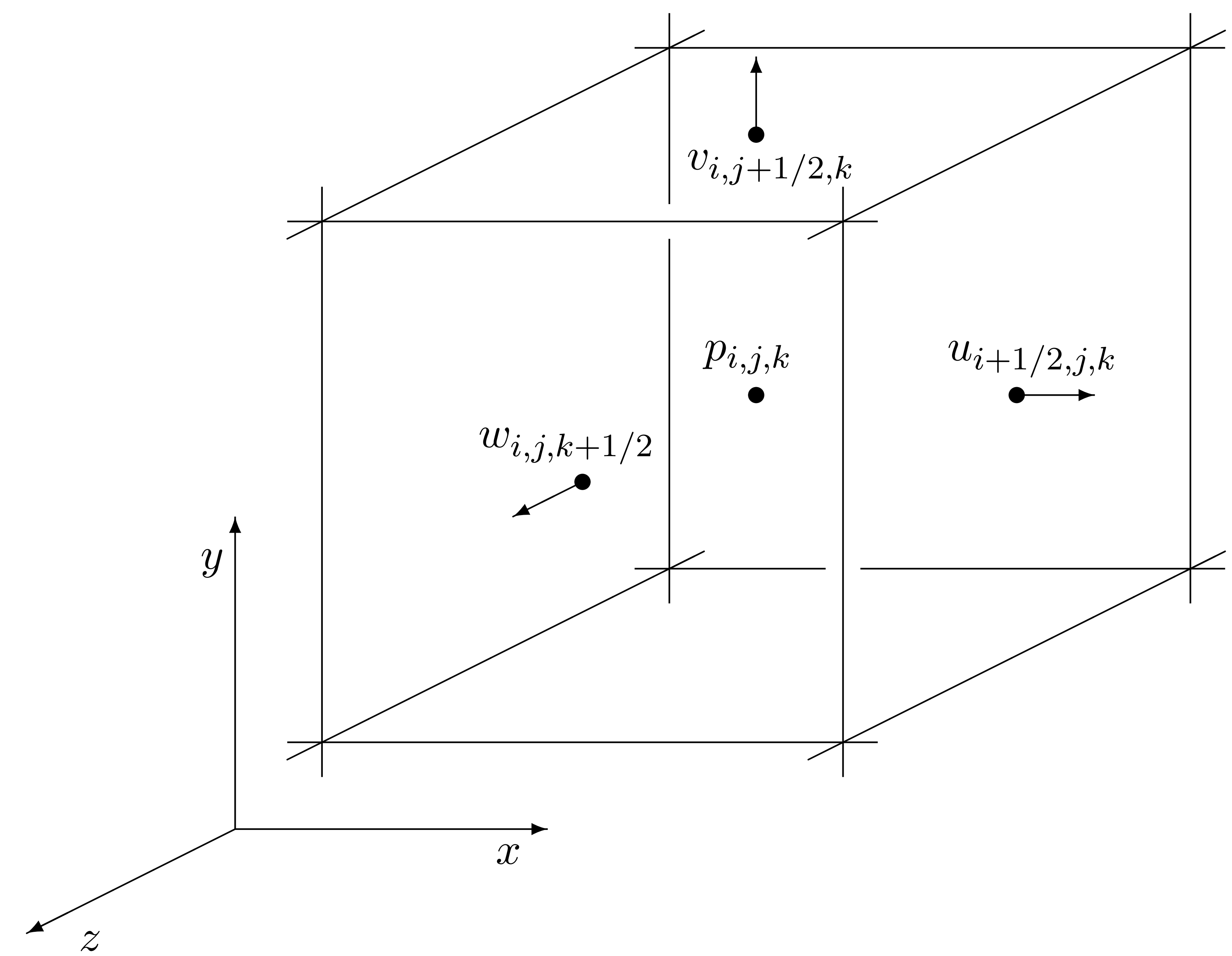
$$\mathbf{u} = \mathbf{u}(x, y)$$
$$p = p(x, y)$$

$$\mathbf{u} = \mathbf{u}(x, y, z)$$
$$p = p(x, y, z)$$

Fluid Grid



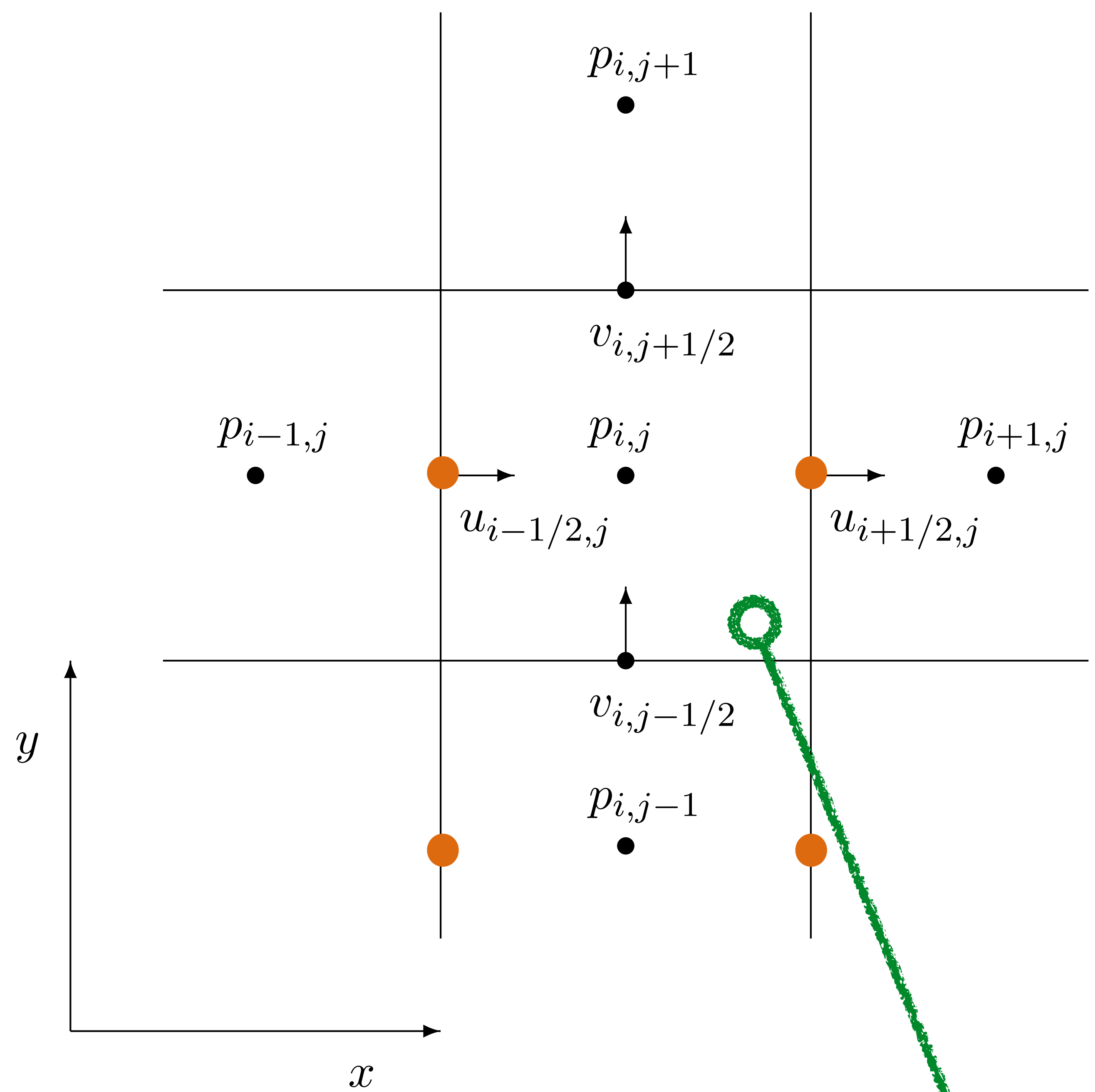
2D Staggered Grid



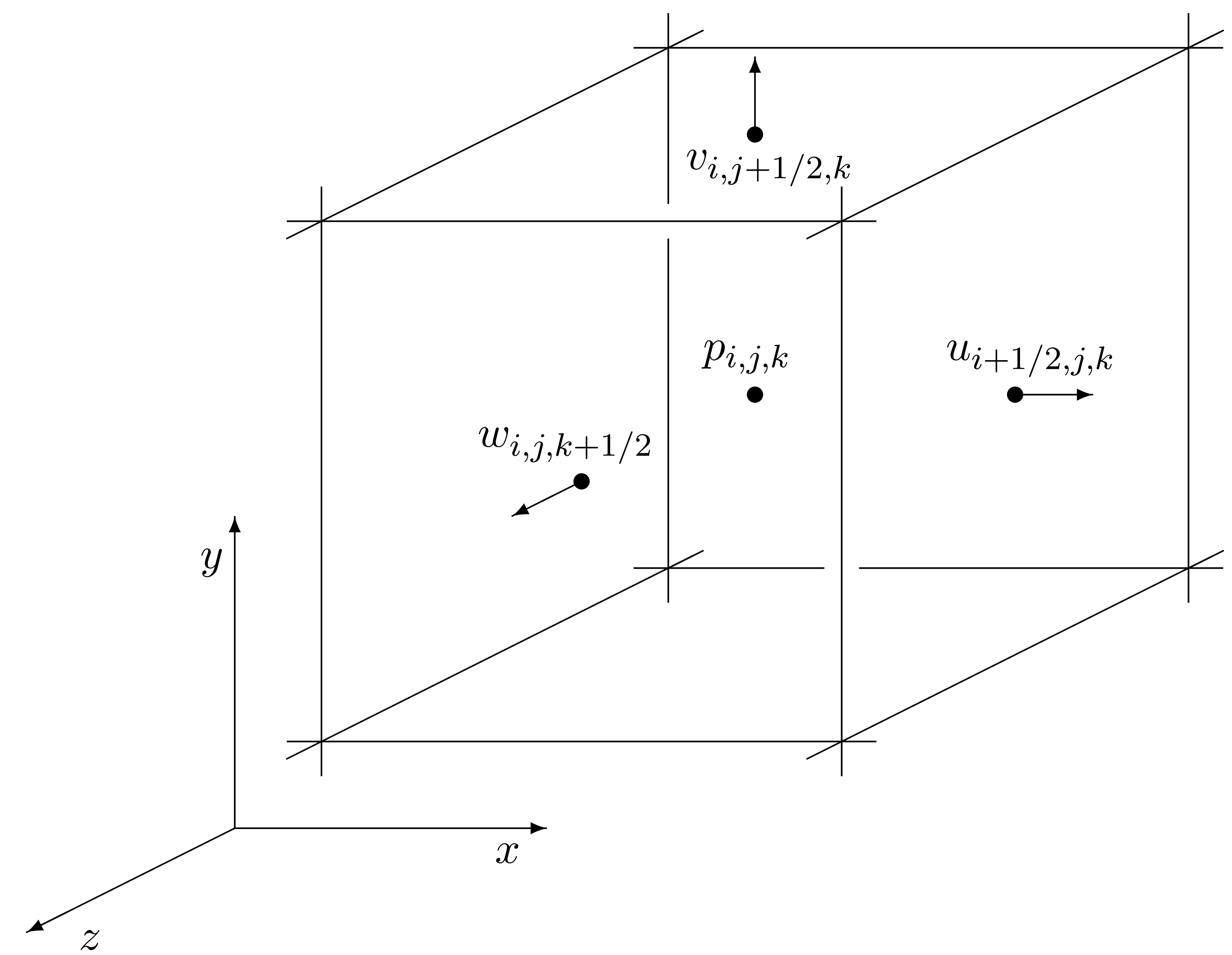
3D Staggered Grid

$$\mathbf{u} = \begin{bmatrix} u \\ v \end{bmatrix}$$

Fluid Grid



2D Staggered Grid

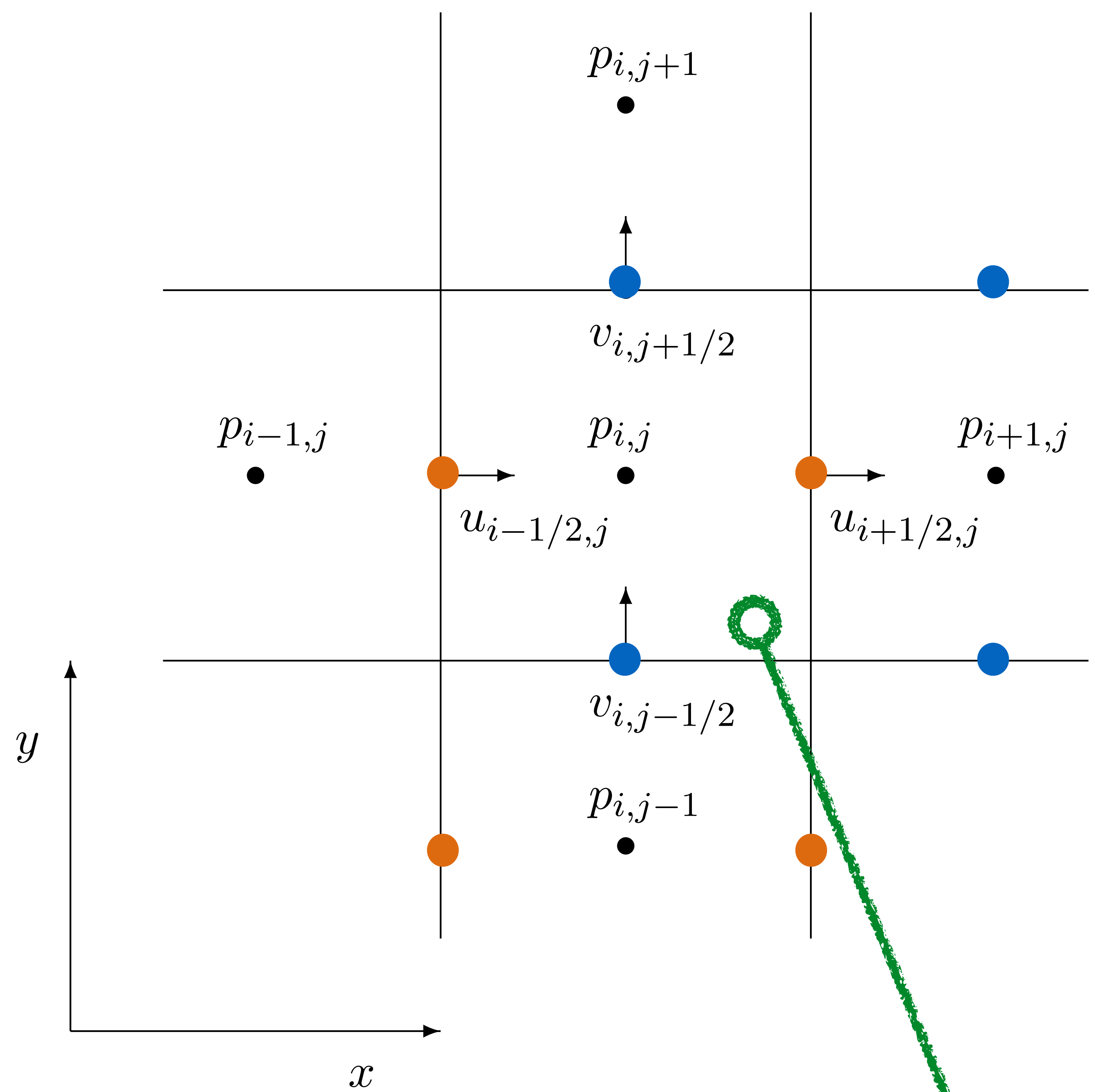


3D Staggered Grid

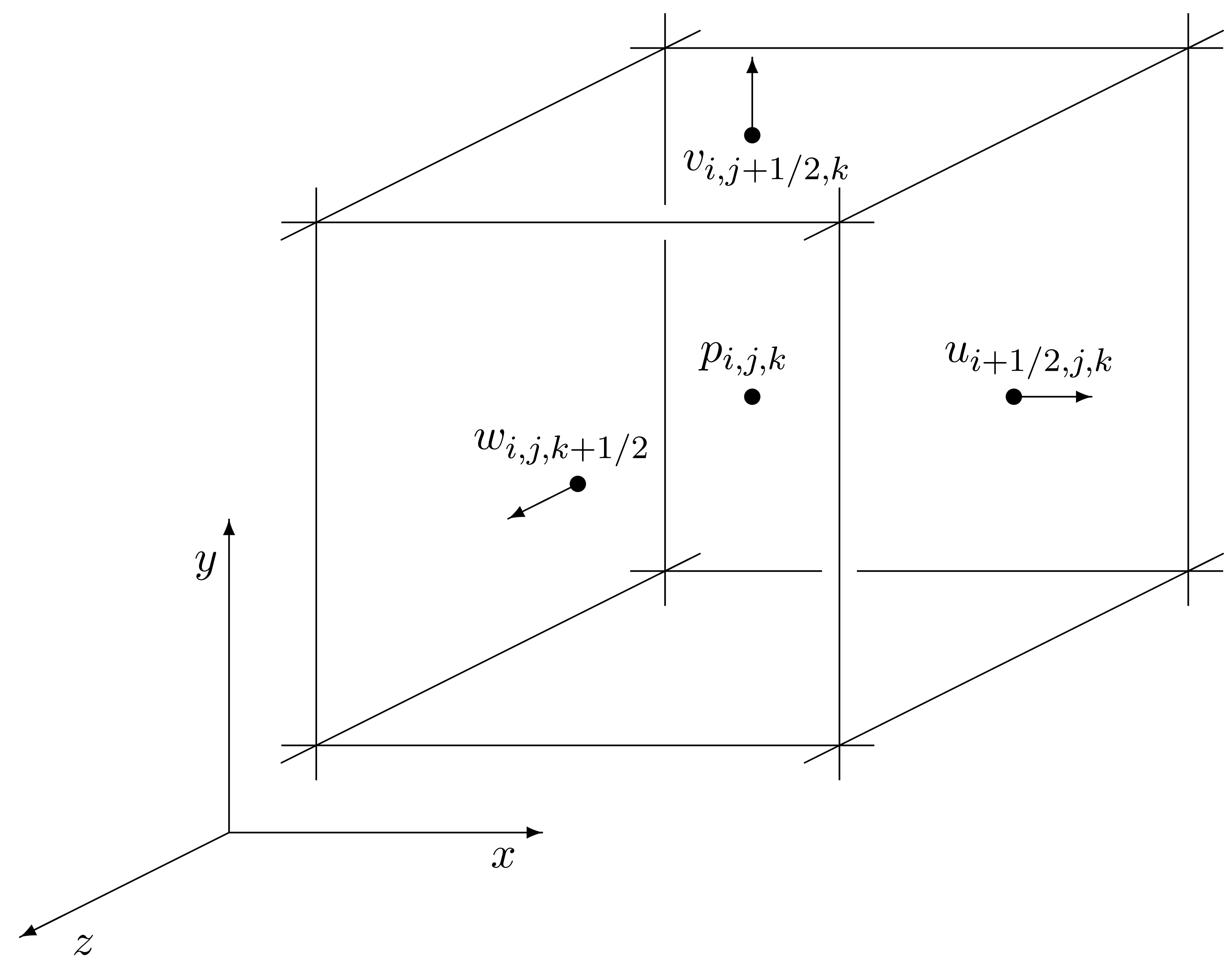
$$\mathbf{u} = \begin{bmatrix} u \\ v \end{bmatrix}$$

$$\mathbf{u}_x = u = \textbf{BiLinear}(\cdot, \cdot, \cdot, \cdot)$$

Fluid Grid



2D Staggered Grid



3D Staggered Grid

$$\mathbf{u} = \begin{bmatrix} u \\ v \end{bmatrix}$$

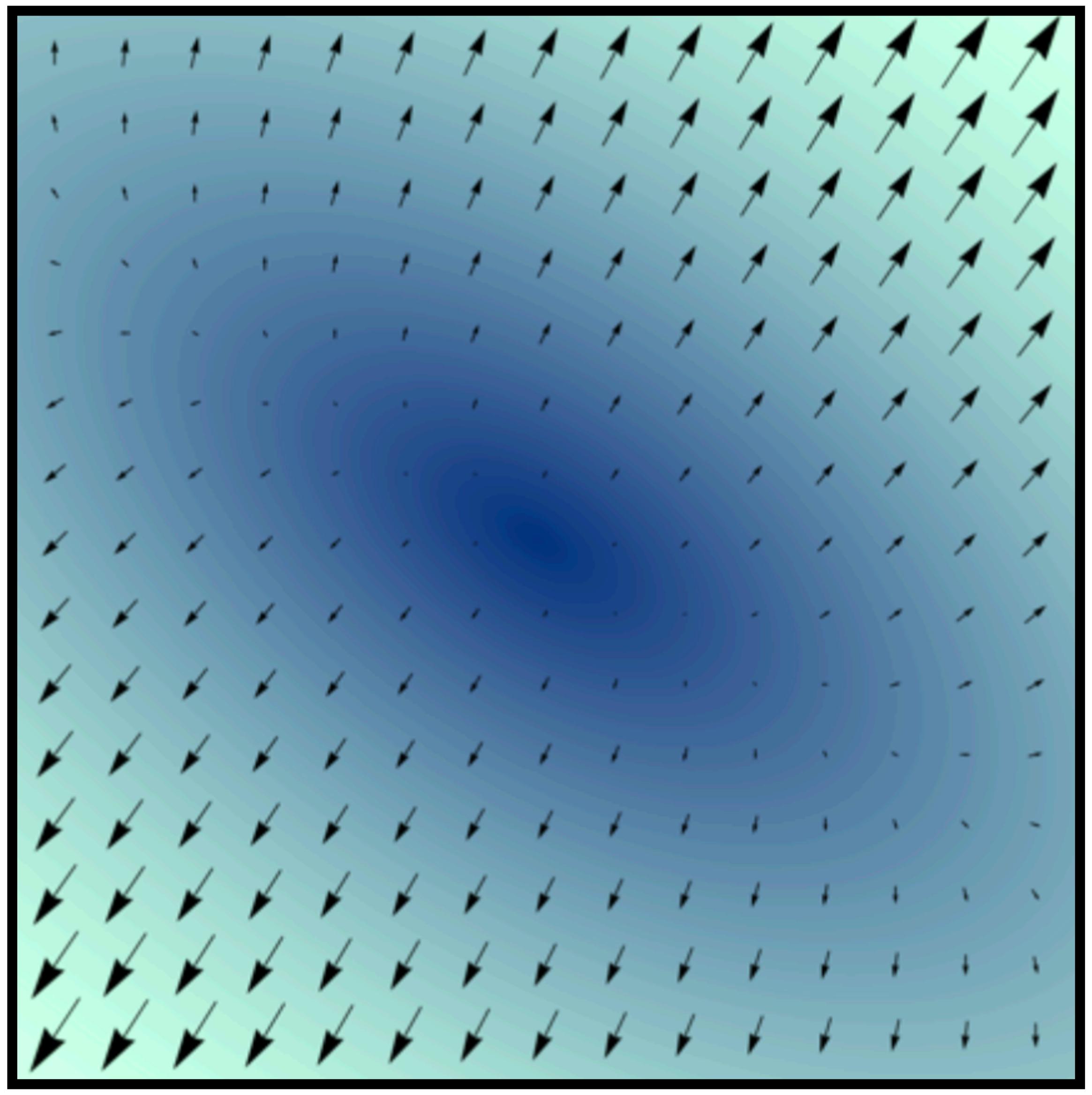
$\mathbf{u}_x = u =$ **BiLinear(** · , · , · , · **)**

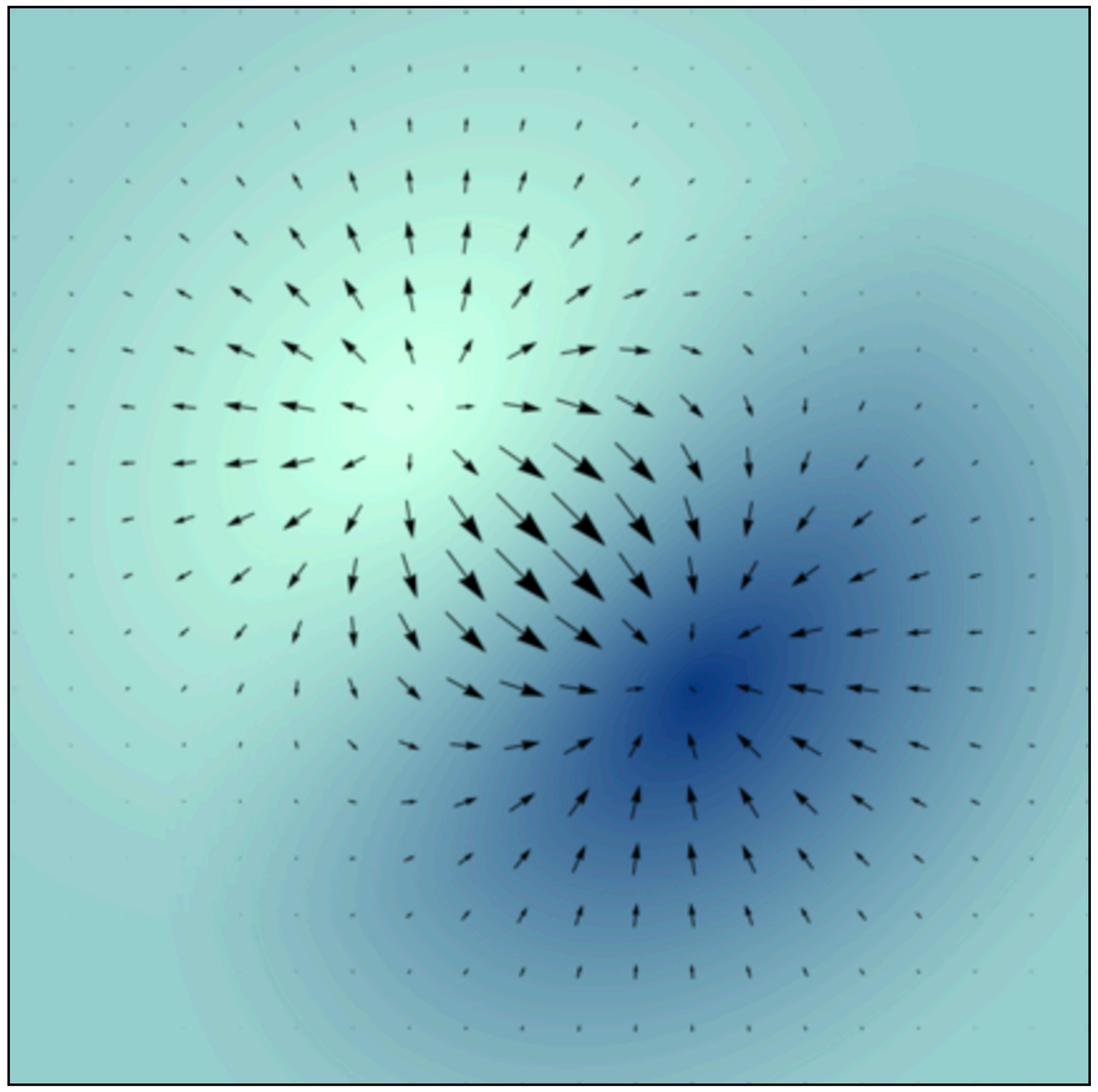
$\mathbf{u}_y = v =$ **BiLinear(** · , · , · , · **)**

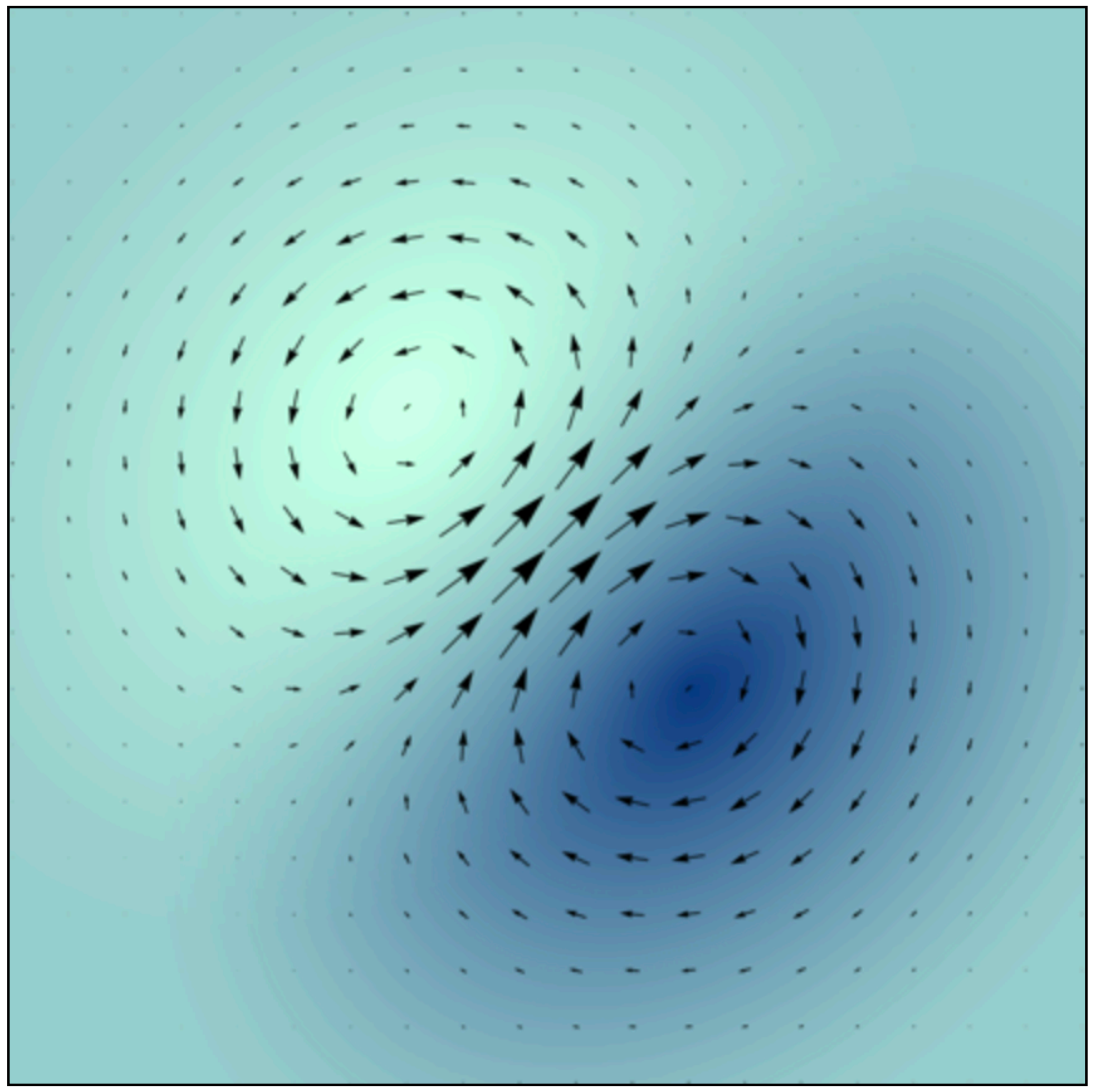
Ren Ng, James O'Brien

Vector Fields

$$\mathbf{v} = \mathbf{v}(x, y)$$
$$p = p(x, y)$$







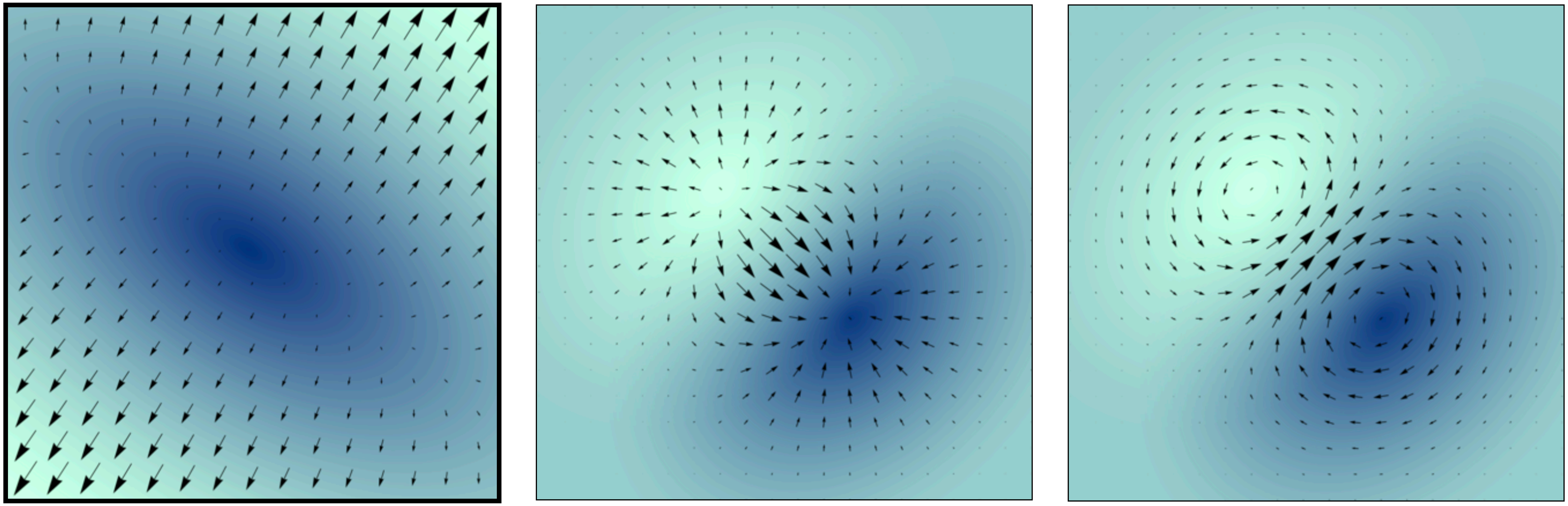
Gradient:

Direction of greatest change

$$\textbf{grad}(p(x,y)) = \nabla p\,|_{x,y} = \begin{bmatrix} \frac{\partial p}{\partial x} \\ \frac{\partial p}{\partial y} \end{bmatrix}$$

Vector Fields

$$\mathbf{v} = \mathbf{v}(x, y)$$
$$p = p(x, y)$$



Gradient:

Direction of greatest change

$$\textbf{grad}(p(x,y)) = \nabla p\,|_{x,y} = \begin{bmatrix} \frac{\partial p}{\partial x} \\ \frac{\partial p}{\partial y} \end{bmatrix}$$

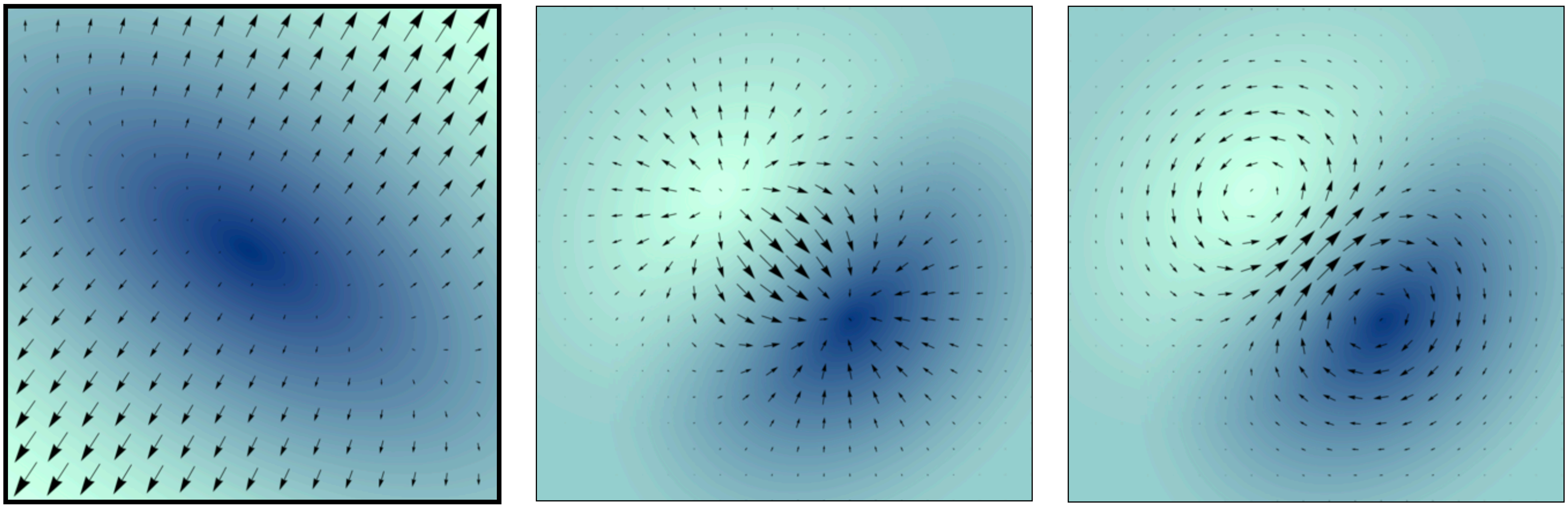
$$\textbf{grad}(p) = \nabla p$$

The ∇ is a differential operator, like $\frac{\partial}{\partial x}$, but a vector

$$\nabla = \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix}$$

Vector Fields

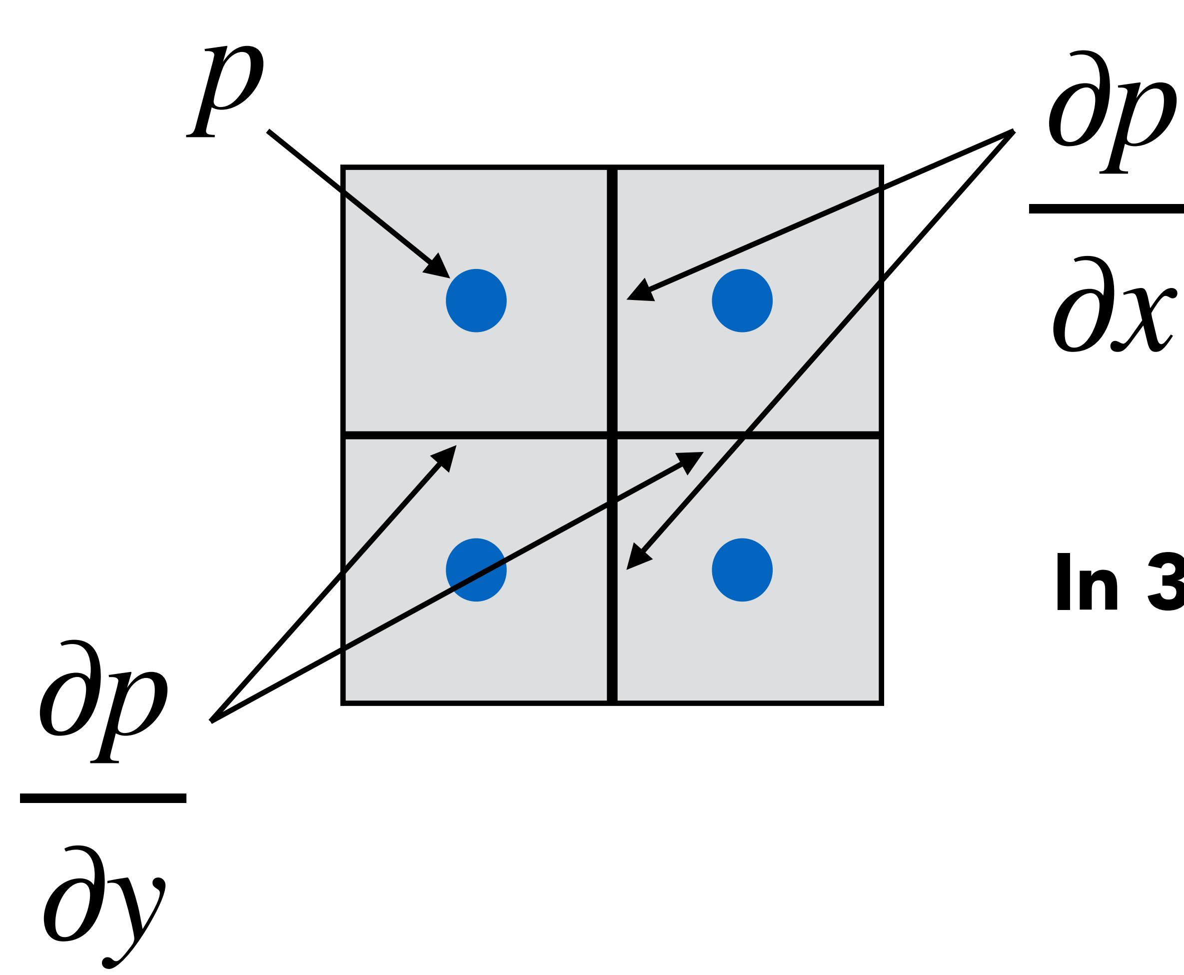
$$\nabla = \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} \qquad \mathbf{v} = \mathbf{v}(x, y) \qquad p = p(x, y)$$



Gradient:

Direction of greatest change

$$\textbf{grad}(p) = \nabla p = \begin{bmatrix} \frac{\partial p}{\partial x} \\ \frac{\partial p}{\partial y} \end{bmatrix}$$

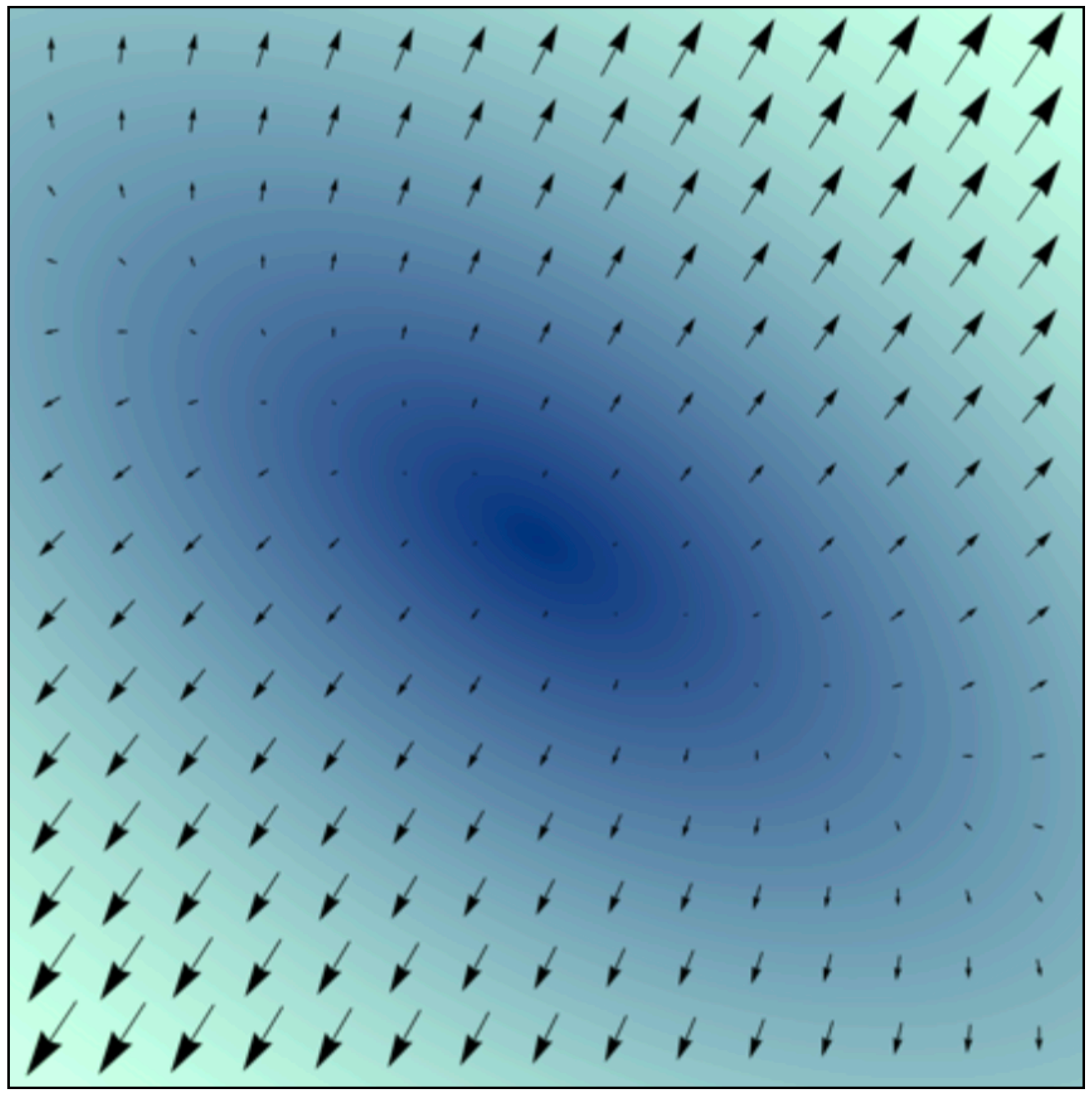


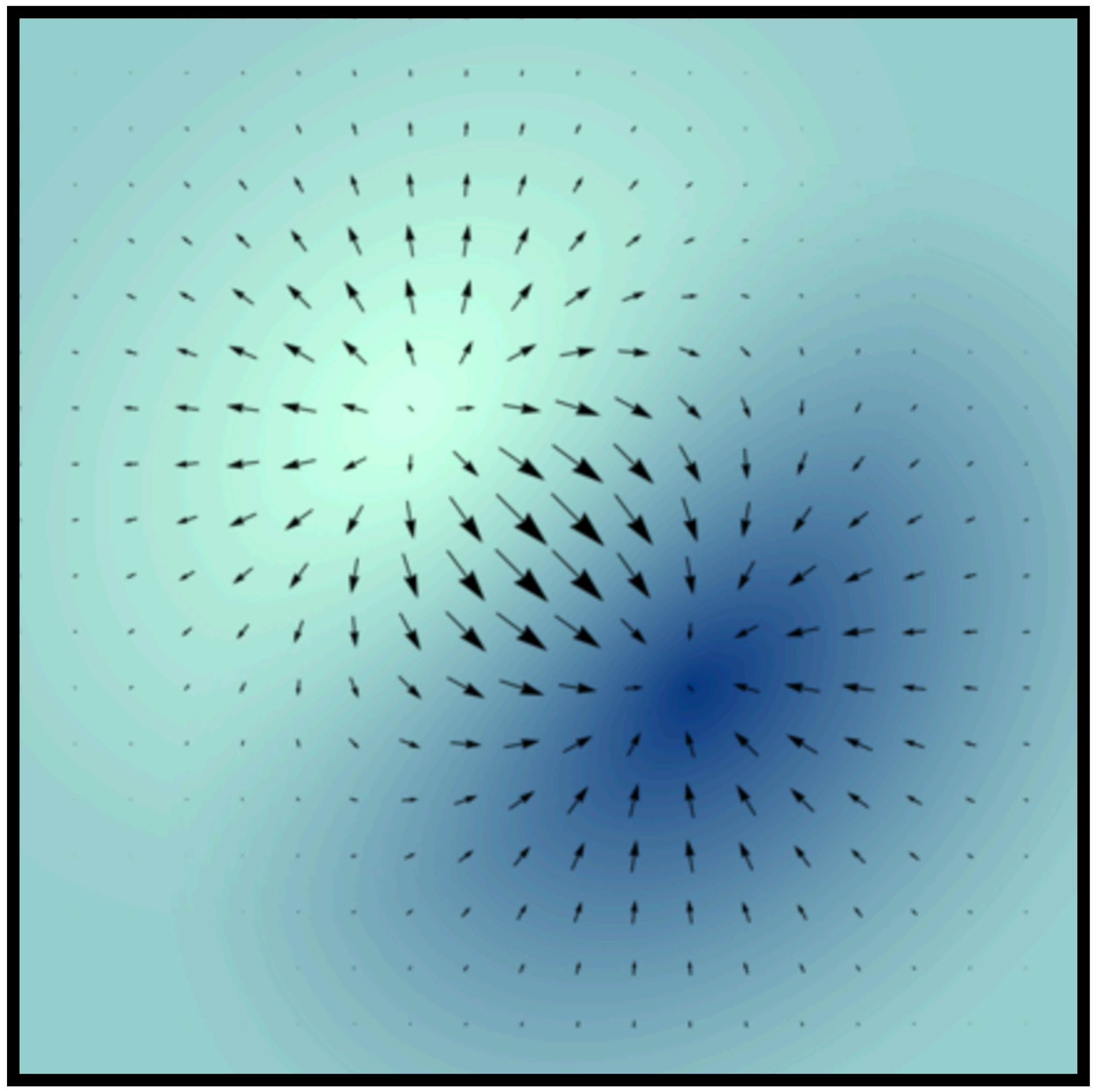
In 3D, cell centers and faces

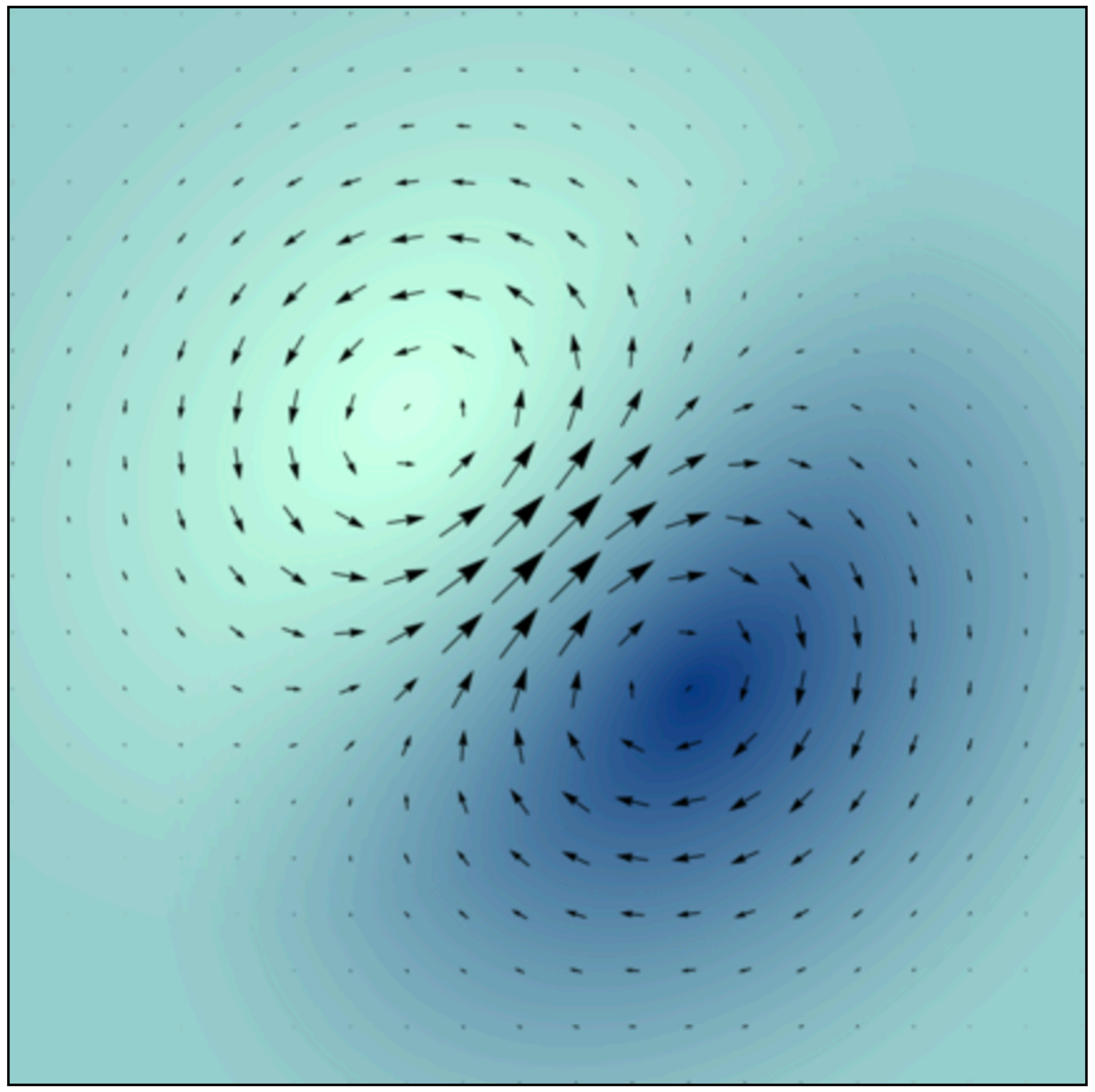
Vector Fields

$$\nabla = \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix}$$

$$\mathbf{v} = \mathbf{v}(x, y)$$
$$p = p(x, y)$$







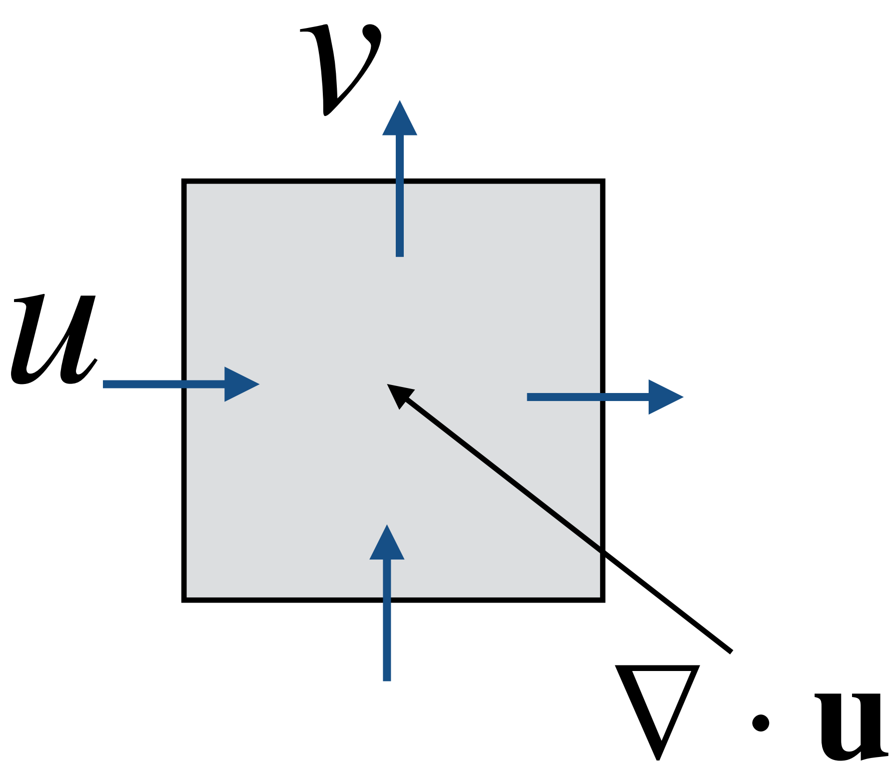
Gradient:
Direction of greatest change

$$\textbf{grad}(p) = \nabla p = \begin{bmatrix} \frac{\partial p}{\partial x} \\ \frac{\partial p}{\partial y} \end{bmatrix}$$

Divergence:
Net flow in or out of region

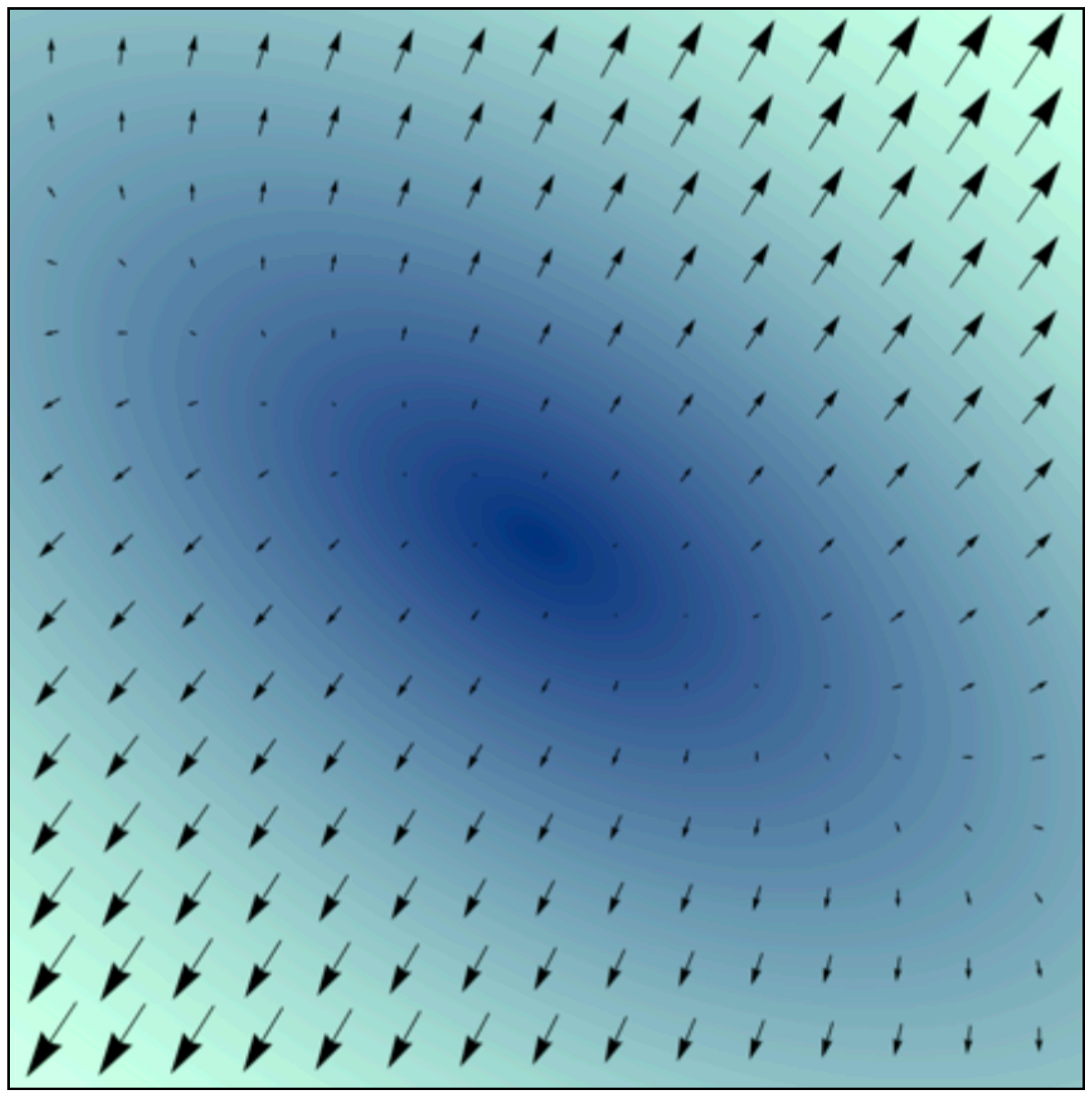
$$\textbf{div}(\mathbf{u}) = \nabla \cdot \mathbf{u} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}$$

In 3D, cell centers and faces

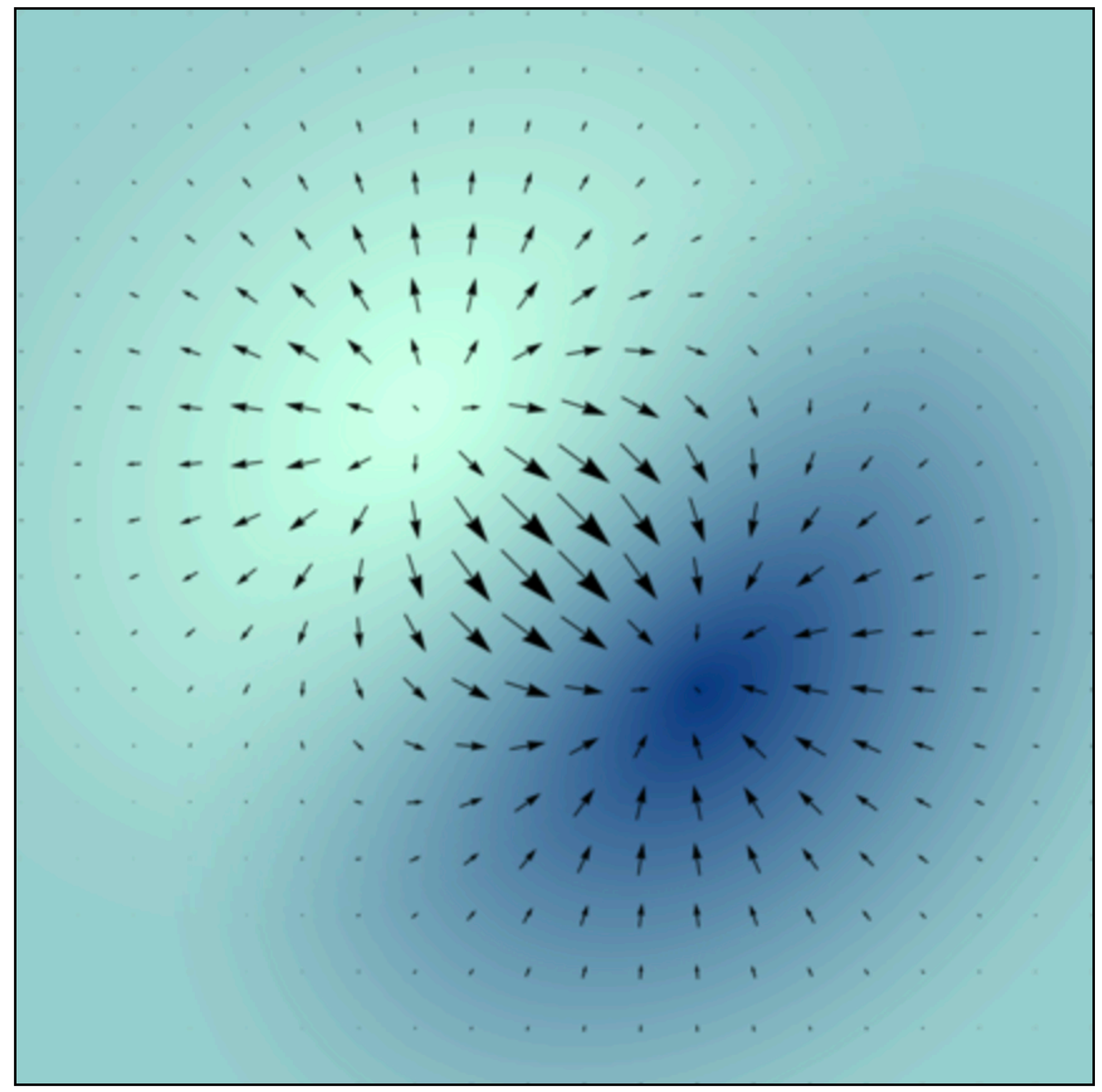


Vector Fields

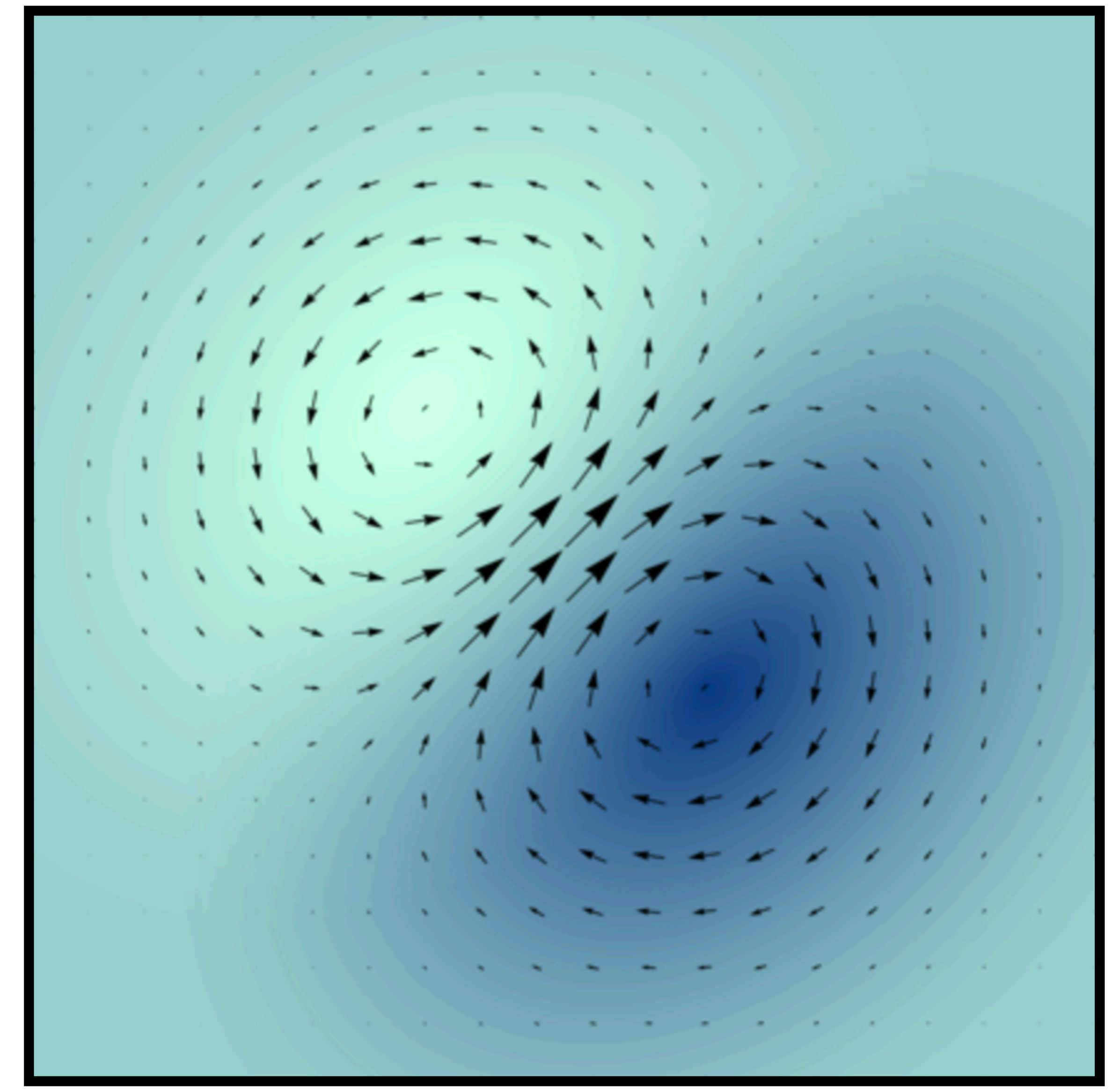
$$\nabla = \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} \qquad \mathbf{v} = \mathbf{v}(x, y) \qquad p = p(x, y)$$



Gradient:
Direction of greatest change



Divergence:
Net flow in or out of region

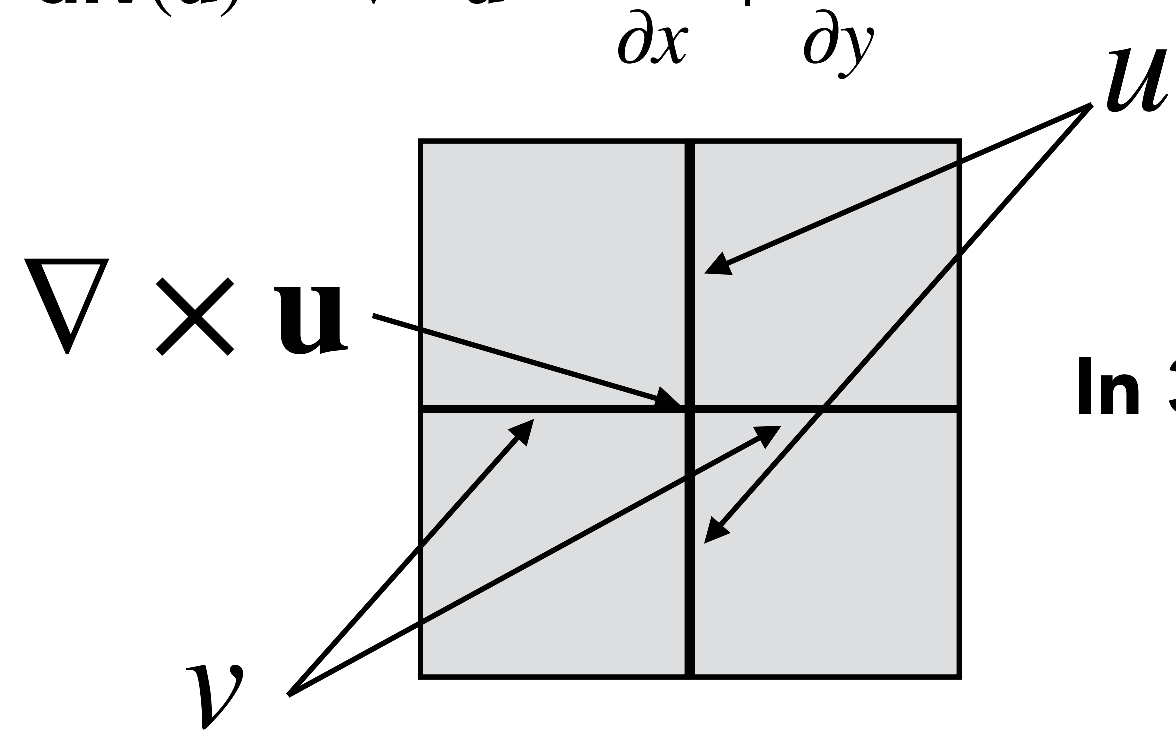


Curl:
Circulation around point

$$\textbf{grad}(p) = \nabla p = \begin{bmatrix} \frac{\partial p}{\partial x} \\ \frac{\partial p}{\partial y} \end{bmatrix}$$

$$\textbf{div}(\mathbf{u}) = \nabla \cdot \mathbf{u} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}$$

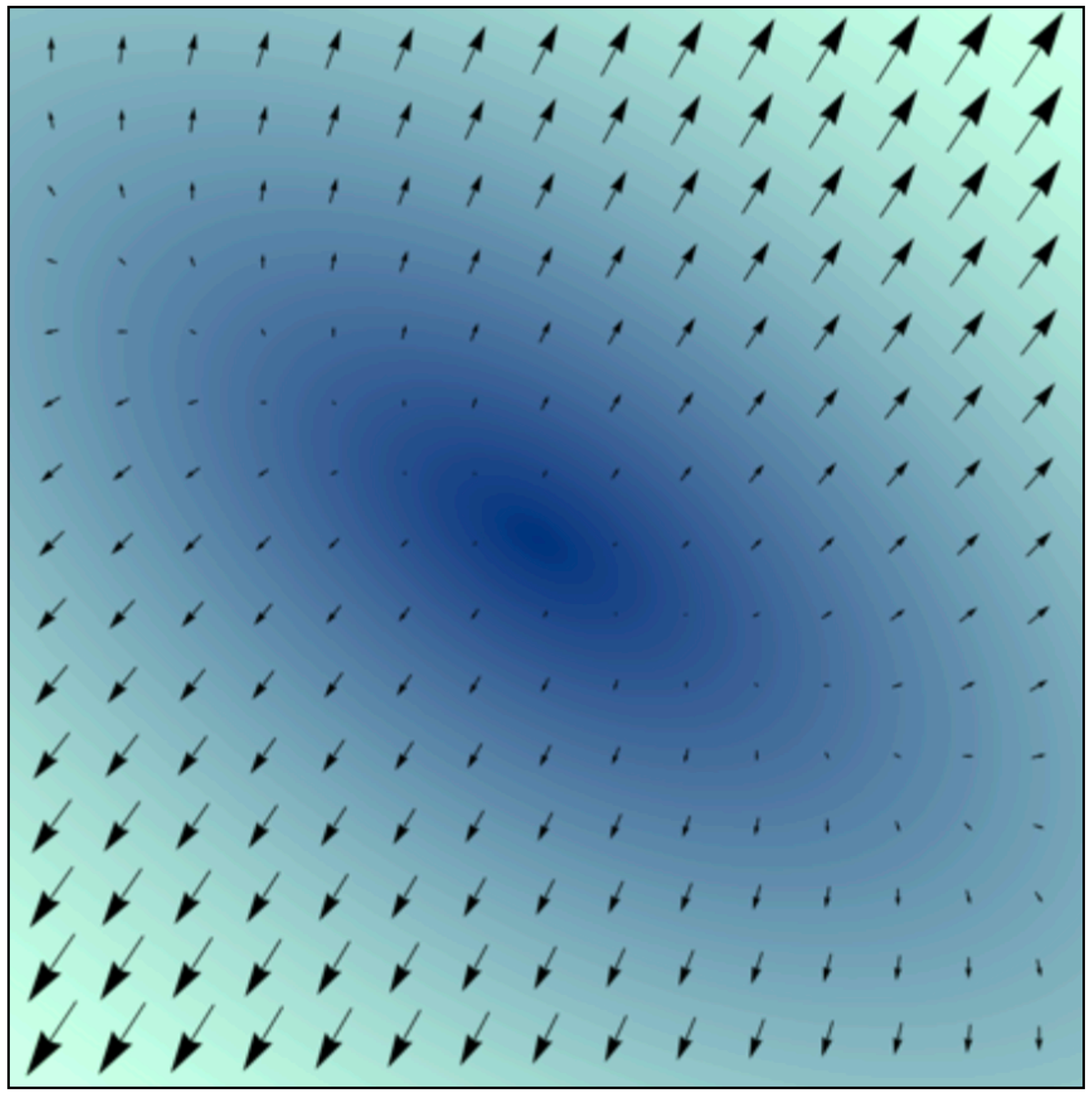
$$\textbf{curl}(\mathbf{u}) = \nabla \times \mathbf{u} = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$$



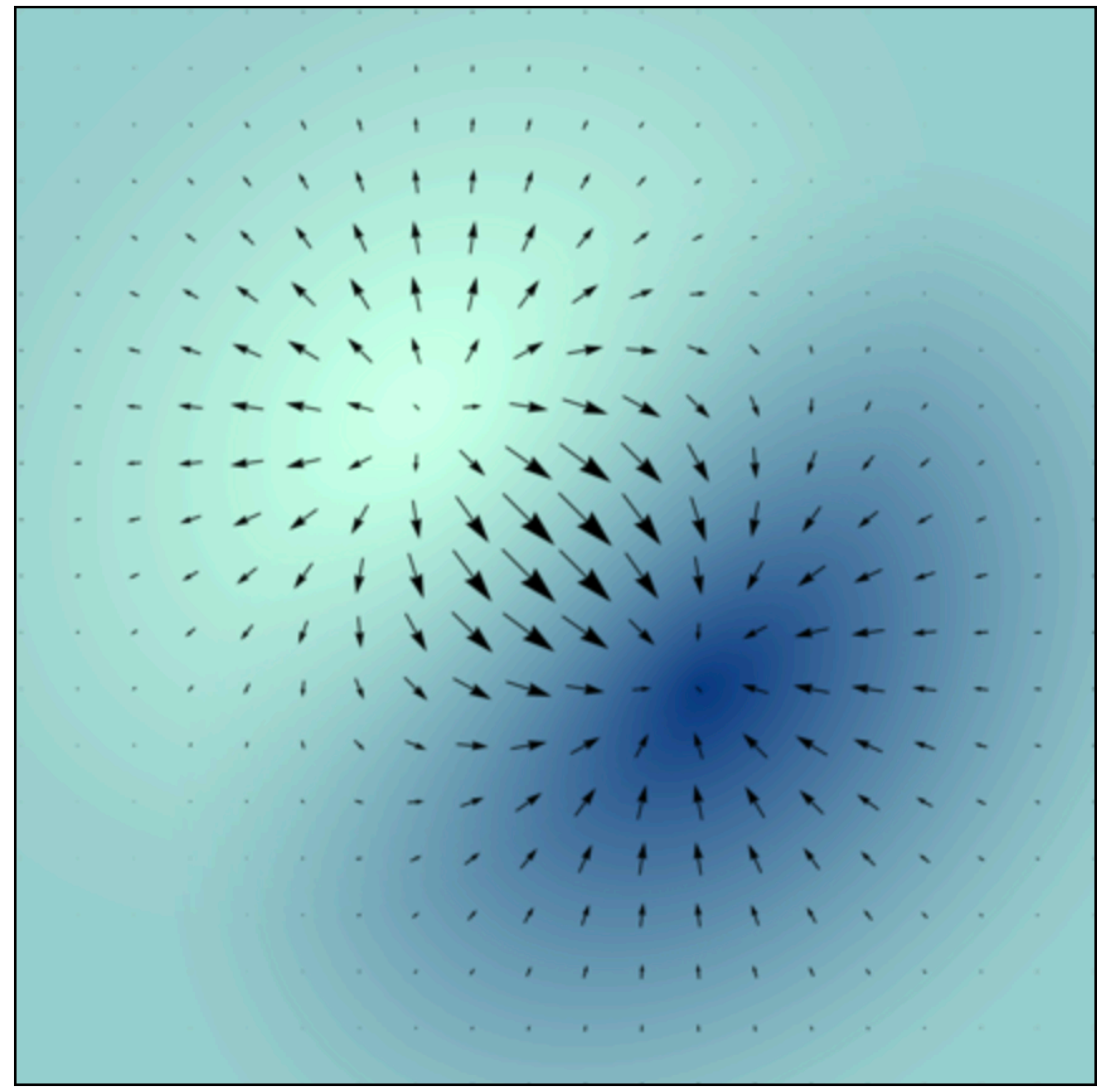
In 3D, cell faces and edges

Vector Fields

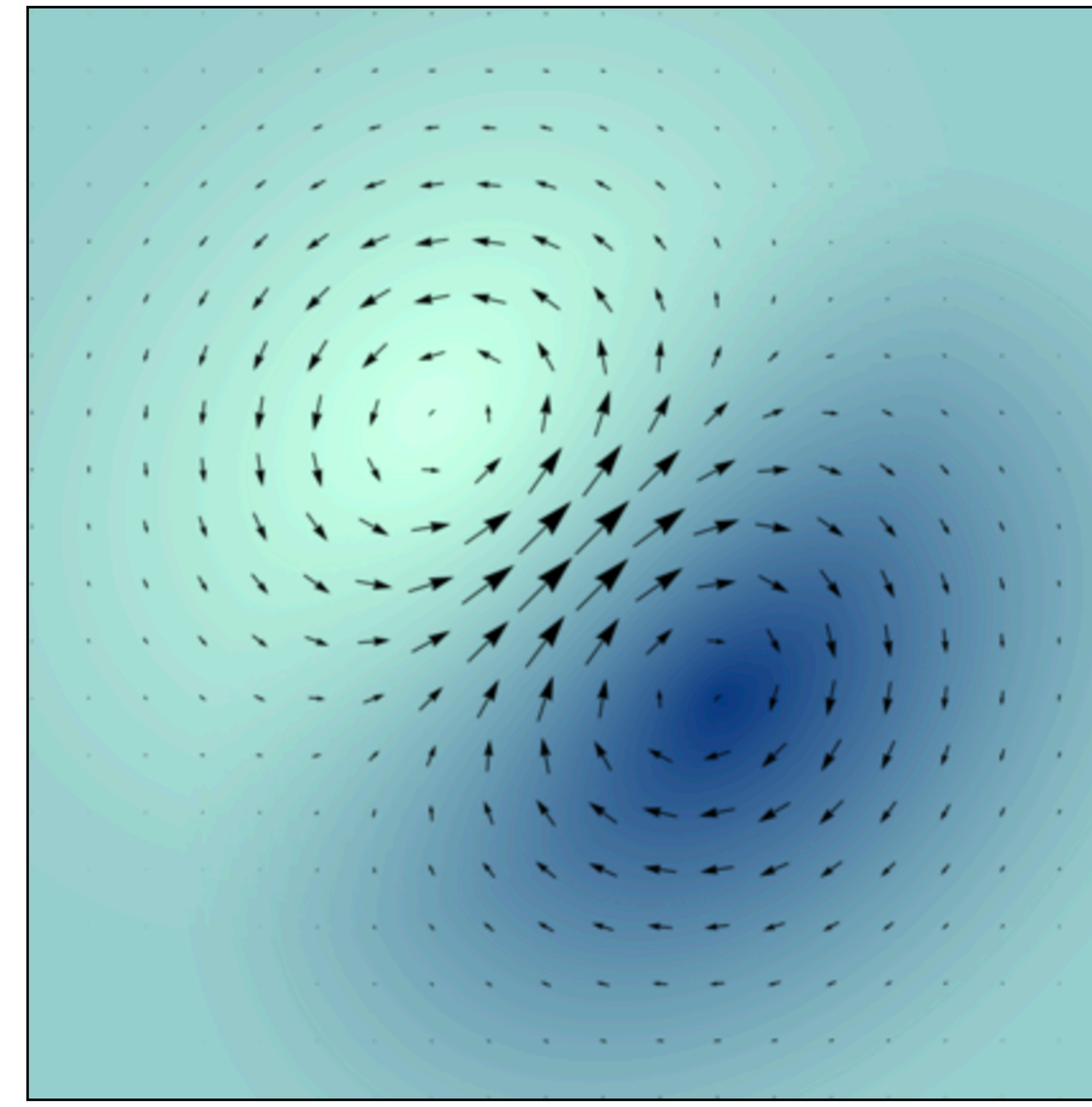
$$\nabla = \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} \qquad \mathbf{v} = \mathbf{v}(x, y) \qquad p = p(x, y)$$



Gradient:
Direction of greatest change



Divergence:
Net flow in or out of region



Curl:
Circulation around point

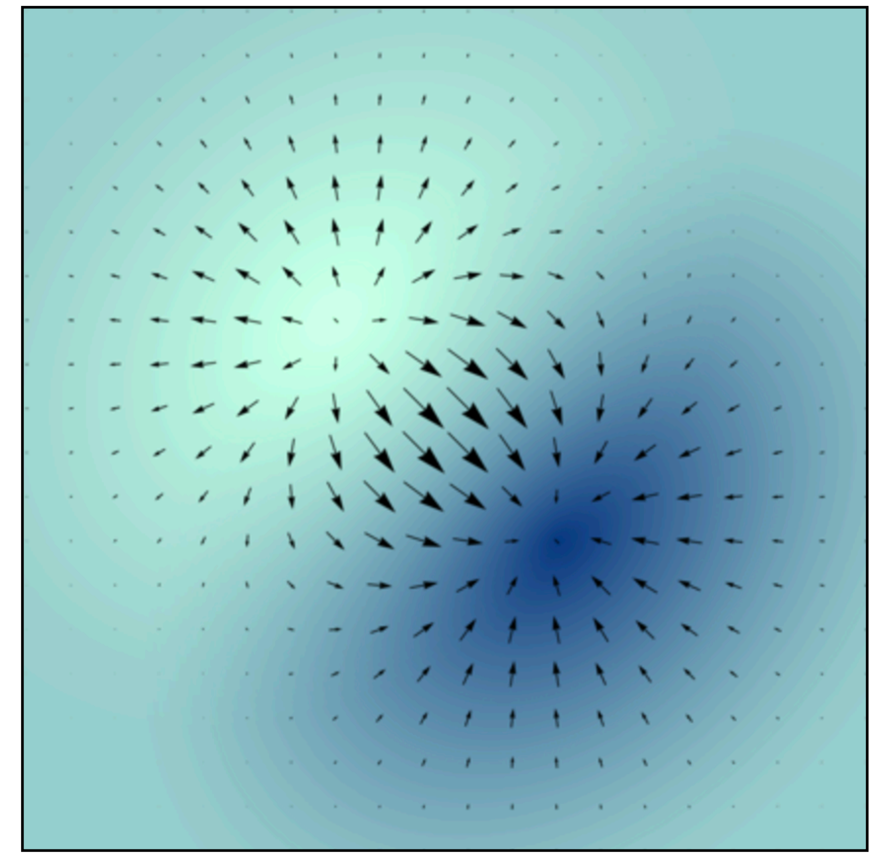
$$\textbf{grad}(p) = \nabla p = \begin{bmatrix} \frac{\partial p}{\partial x} \\ \frac{\partial p}{\partial y} \end{bmatrix}$$

$$\textbf{div}(\mathbf{u}) = \nabla \cdot \mathbf{u} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}$$

$$\textbf{curl}(\mathbf{u}) = \nabla \times \mathbf{u} = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$$

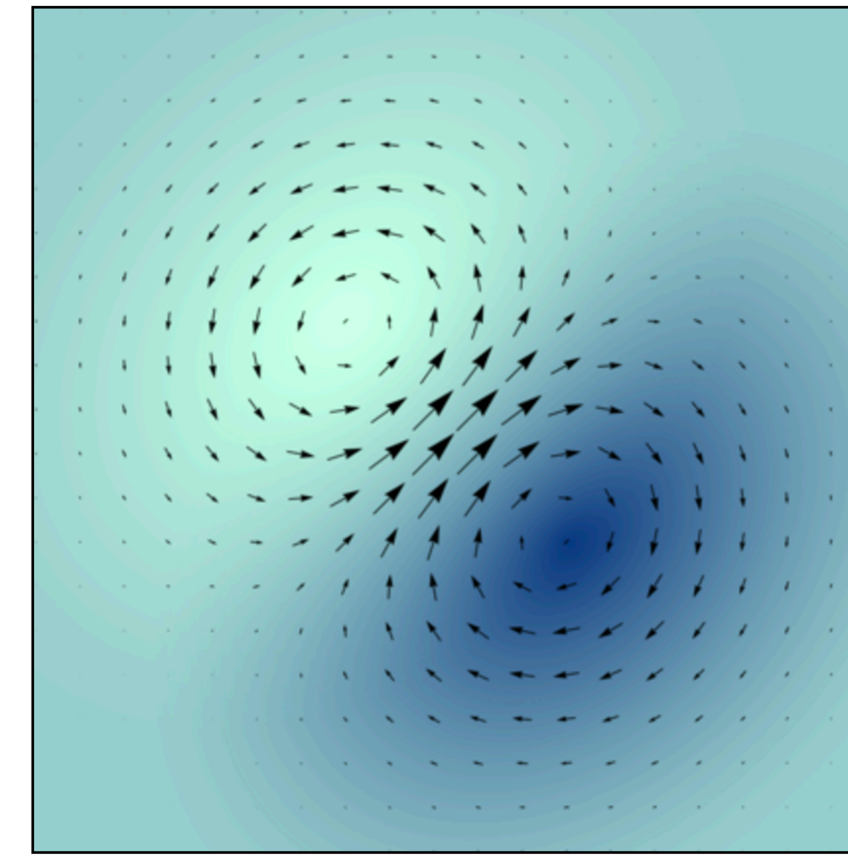
In 3D, curl is vector stored at edges

Vector Fields



Divergence:
Net flow in or out of region

$$\textbf{div}(\mathbf{u}) = \nabla \cdot \mathbf{u} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}$$

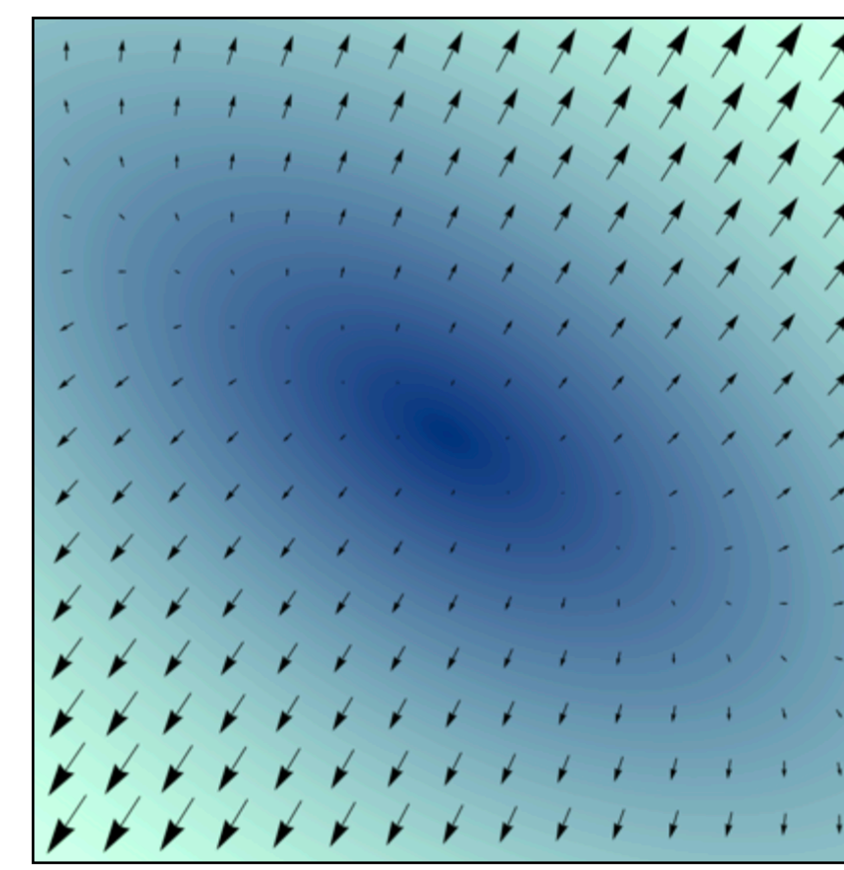


Curl:
Circulation around point

$$\textbf{curl}(\mathbf{u}) = \nabla \times \mathbf{u} = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$$

Laplacian:
Difference from the neighborhood average

$$\nabla^2 = \nabla \cdot \nabla = \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right)$$



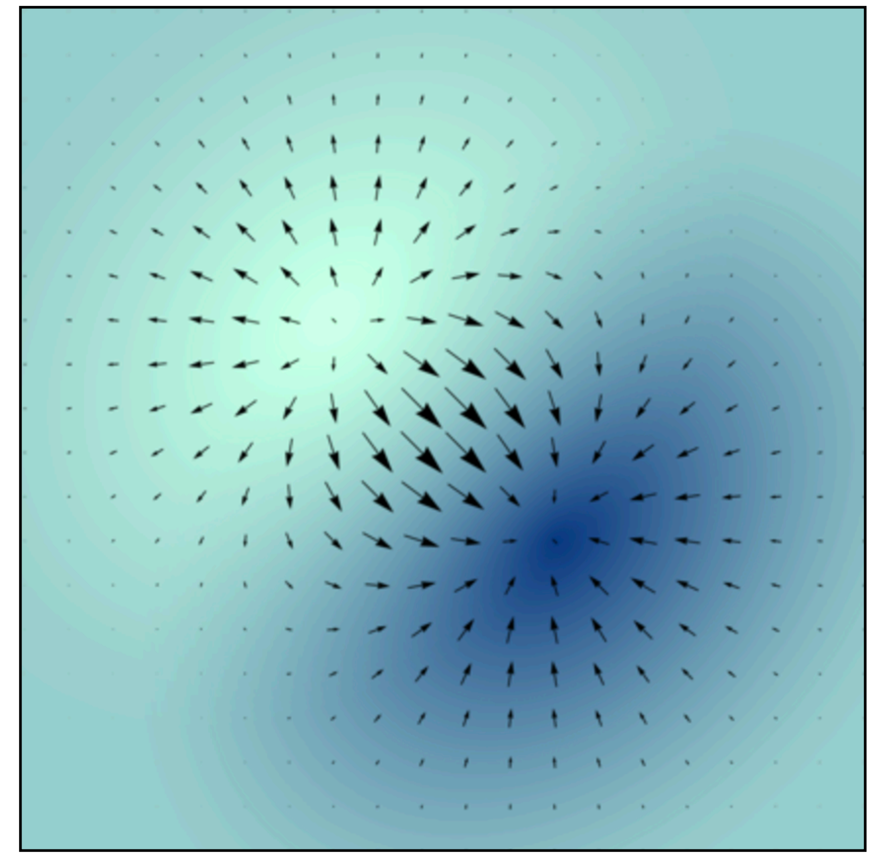
Gradient:
Direction of greatest change

$$\textbf{grad}(p) = \nabla p = \begin{bmatrix} \frac{\partial p}{\partial x} \\ \frac{\partial p}{\partial y} \end{bmatrix}$$

Vector Fields

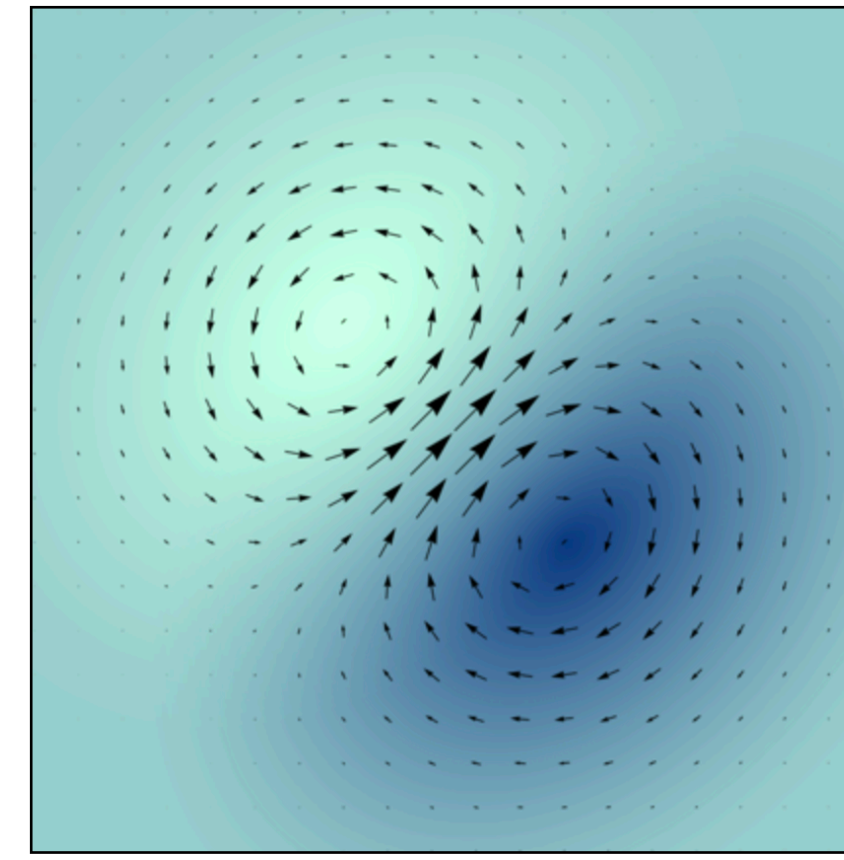
Laplacian:
Difference from the neighborhood average

$$\nabla^2 = \nabla \cdot \nabla = \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right)$$



Divergence:
Net flow in or out of region

$$\textbf{div}(\mathbf{u}) = \nabla \cdot \mathbf{u} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}$$



Curl:
Circulation around point

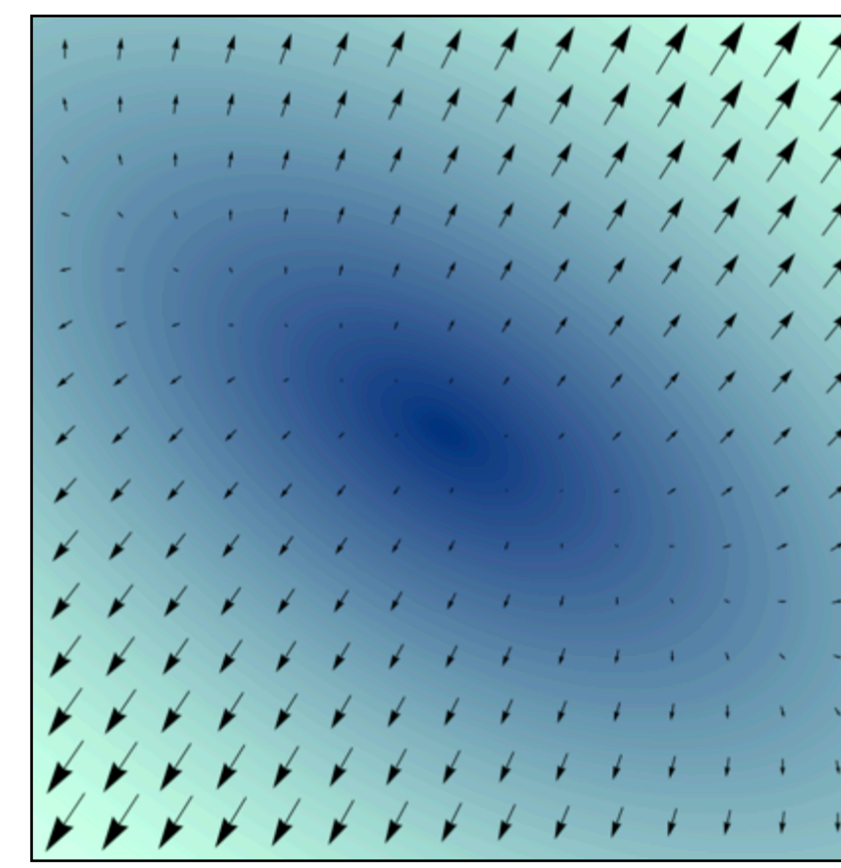
$$\textbf{curl}(\mathbf{u}) = \nabla \times \mathbf{u} = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$$

Directional Derivative:
How a quantity changes as point of observation moves

$$(\mathbf{u} \cdot \nabla) = \left(u_x \frac{\partial}{\partial x} + u_y \frac{\partial}{\partial y} \right)$$

How fast are we moving in x direction?

How does something change as we move in the x direction?



Gradient:
Direction of greatest change

$$\textbf{grad}(p) = \nabla p = \begin{bmatrix} \frac{\partial p}{\partial x} \\ \frac{\partial p}{\partial y} \end{bmatrix}$$

"Vector Analysis" - lots of fun math

Navier–Stokes Equations (N-SE)

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u} \cdot \nabla)\mathbf{u} - \frac{\nabla p}{\rho} + \frac{\nu}{\rho}\nabla^2 \mathbf{u} + \frac{\mathbf{f}_f}{\rho}$$

Change in fluid velocity

Advection

Pressure

Viscosity

Field forces (*e.g.*: gravity)

ρ **is density**

ν **is viscosity**

Bad Solver

- Store velocity ($\mathbf{u}$) and density (ρ) on staggered grid
- Compute pressure (p) as function of density
- Use N-SE to update velocities
- Update densities $\dot{\rho} \propto -(\mathbf{u} \cdot \nabla)\rho + \nabla \cdot (\mathbf{u}\,\rho)$
- Repeat until end of simulation

Problem: Pressure waves move fast so this explicit method must use very small timesteps or go unstable.

Problem: Advection term also limits time step based on speed of fluid. (Bulk speed of fluid is generally less than wave speed.)

Incompressible Fluids

Replace pressure forces with constraints

- No more pressure waves
- This is another projection method!

Divergence is net in-/out-flow

- Constrain divergence to be zero by projection
  - $\nabla \cdot \mathbf{u} = 0$

Split advection term off from the rest of N-SE and use semi-Lagrangian advection.

"Stable Fluids" by Jos Stam, SIGGRAPH 99

Incompressible Fluids

Separate problems terms from the rest:

$$\Delta \mathbf{u} = \Delta t \left(-(\mathbf{u} \cdot \nabla)\mathbf{u} - \frac{\nabla p}{\rho} + \frac{\nu}{\rho} \nabla^2 \mathbf{u} + \frac{\mathbf{f}_f}{\rho} \right)$$

$$\Delta \mathbf{u} = \Delta t \left(\Delta \mathbf{u}_a + \Delta \mathbf{u}_p + \frac{\nu}{\rho} \nabla^2 \mathbf{u} + \frac{\mathbf{f}_f}{\rho} \right)$$

$$\mathbf{u}^* = \mathbf{u}^t + \Delta t \left(\frac{\nu}{\rho} \nabla^2 \mathbf{u} + \frac{\mathbf{f}_f}{\rho} \right)$$

Unprojected and unadvected new velocities

Incompressible Fluids

Separate problems terms from the rest:

$$\mathbf{u}^* = \mathbf{u}^t + \Delta t \left(\frac{\nu}{\rho} \nabla^2 \mathbf{u} + \frac{\mathbf{f}_f}{\rho} \right)$$

In general we will have $\nabla \cdot \mathbf{u}^* \neq 0$

Use pressure to correct this:

$$\nabla \cdot \left(\mathbf{u}^* + \Delta \mathbf{u}_p \right) = \nabla \cdot \mathbf{u}^* + \nabla \cdot \Delta \mathbf{u}_p = 0$$

$$\Delta \mathbf{u}_p = -\Delta t \frac{\nabla p}{\rho}$$

$$\nabla \cdot \mathbf{u}^* = \Delta t \nabla \cdot \frac{\nabla p}{\rho}$$

Incompressible Fluids

Separate problems terms from the rest:

$$\mathbf{u}^* = \mathbf{u}^t + \Delta t \left(\frac{\nu}{\rho} \nabla^2 \mathbf{u} + \frac{\mathbf{f}_f}{\rho} \right)$$

$$\nabla \cdot \mathbf{u}^* = \Delta t \nabla \cdot \frac{\nabla p}{\rho}$$

$$\frac{\Delta t \nabla^2}{\rho} p = \nabla \cdot \mathbf{u}^*$$

$$\mathbf{A}\,\mathbf{x} = \mathbf{b}$$ Solve for pressure.

Density is now constant, so it can move past the divergence operator.

Incompressible Fluids

Add pressure correction to get projected, but not advected, velocities:

$$\mathbf{u}^{+} = \mathbf{u}^{*} - \frac{\Delta t \, \nabla^{2}}{\rho} p$$

Solving for pressure

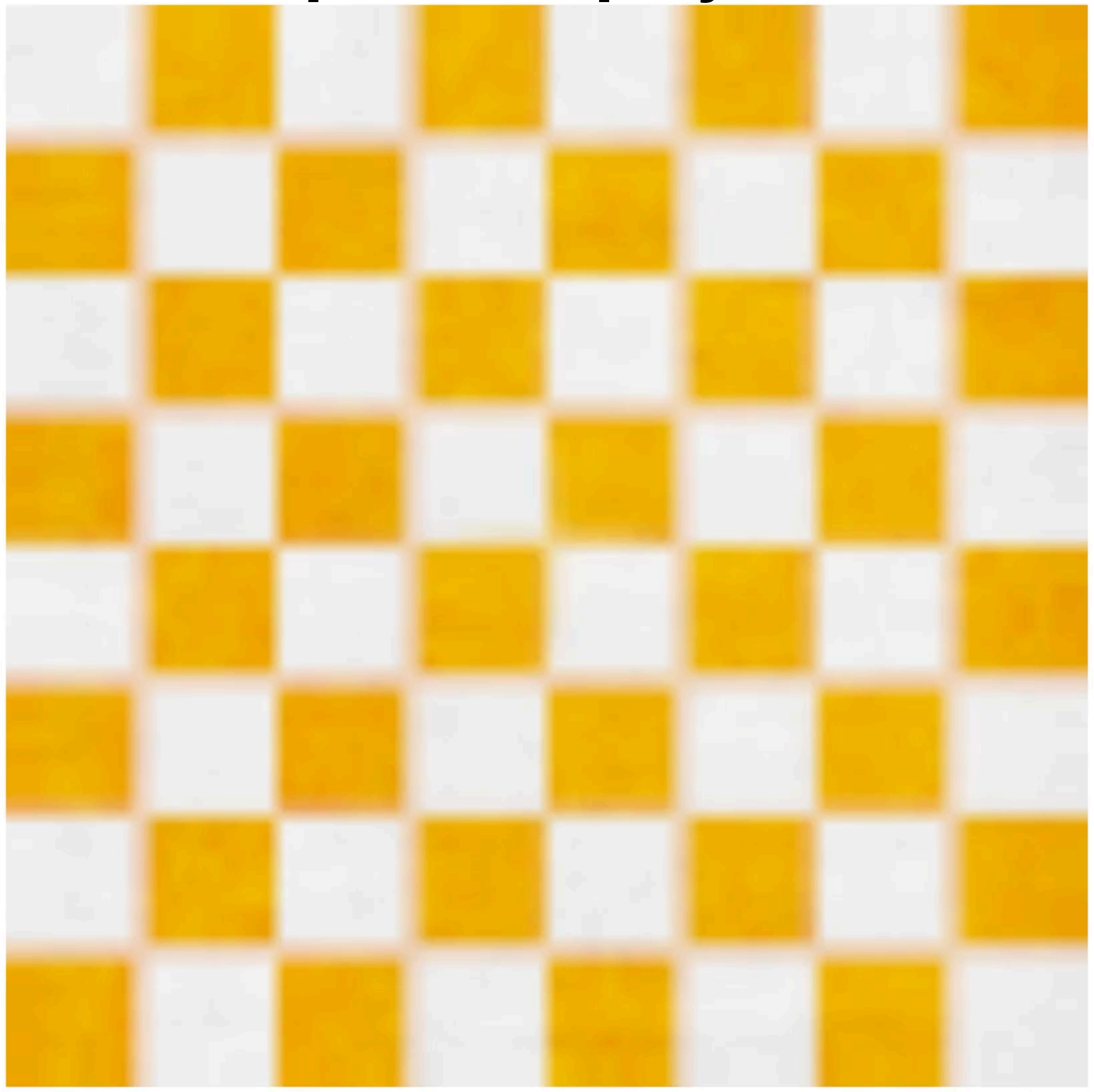
- Successive over-relaxation
  - Easy to understand and implement, but slow
- Pre-conditions conjugate gradient
  - Widely used, reasonably fast
  - [Modified] Incomplete Cholesky for preconditioned
- Other problem-specific methods

Incompressible Fluids

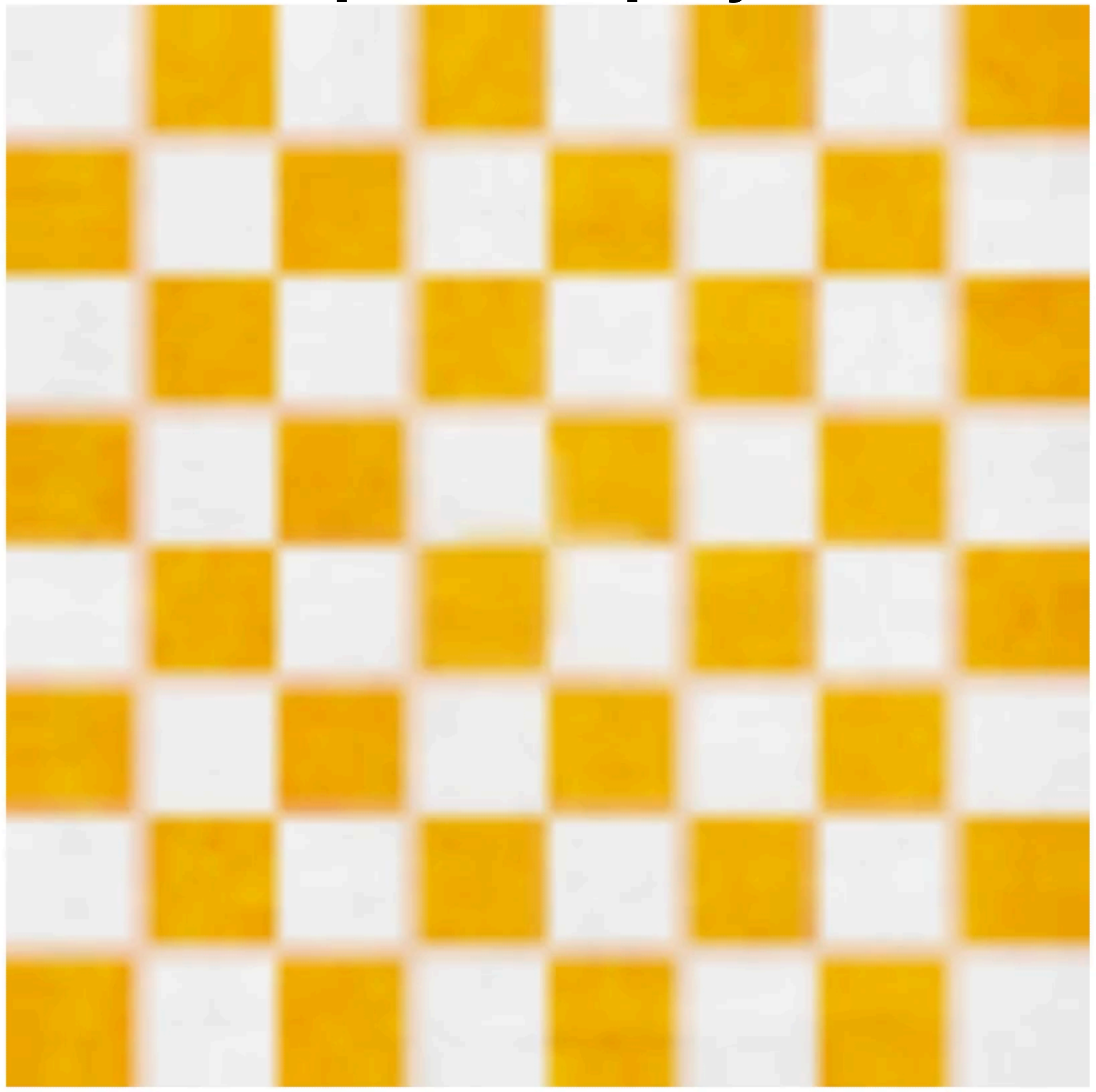
Add pressure correction to get projected, but not advected, velocities:

$$\mathbf{u}^{+} = \mathbf{u}^{*} - \frac{\Delta t \, \nabla^{2}}{\rho} p$$

No pressure projection



With pressure projection

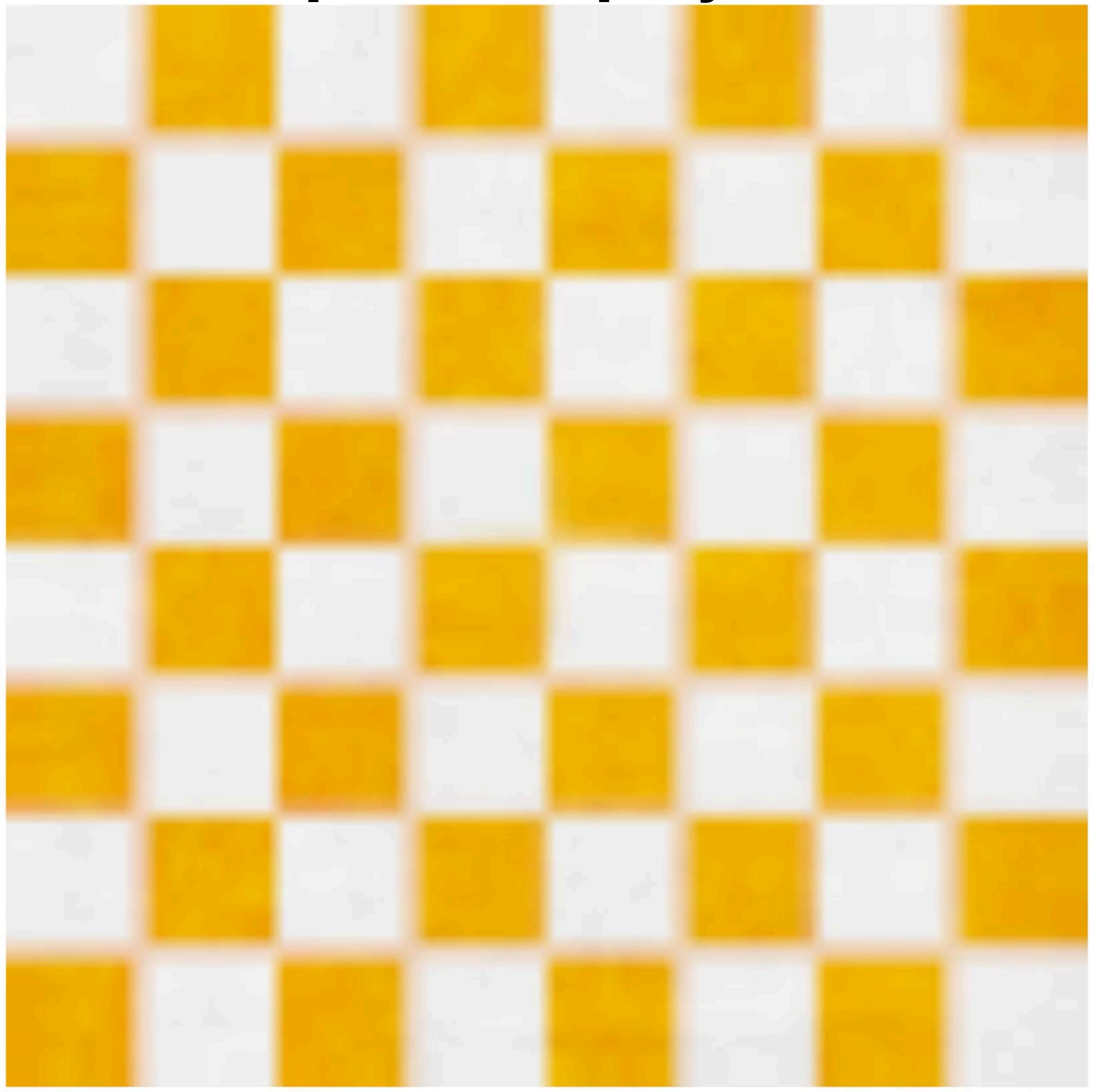


Incompressible Fluids

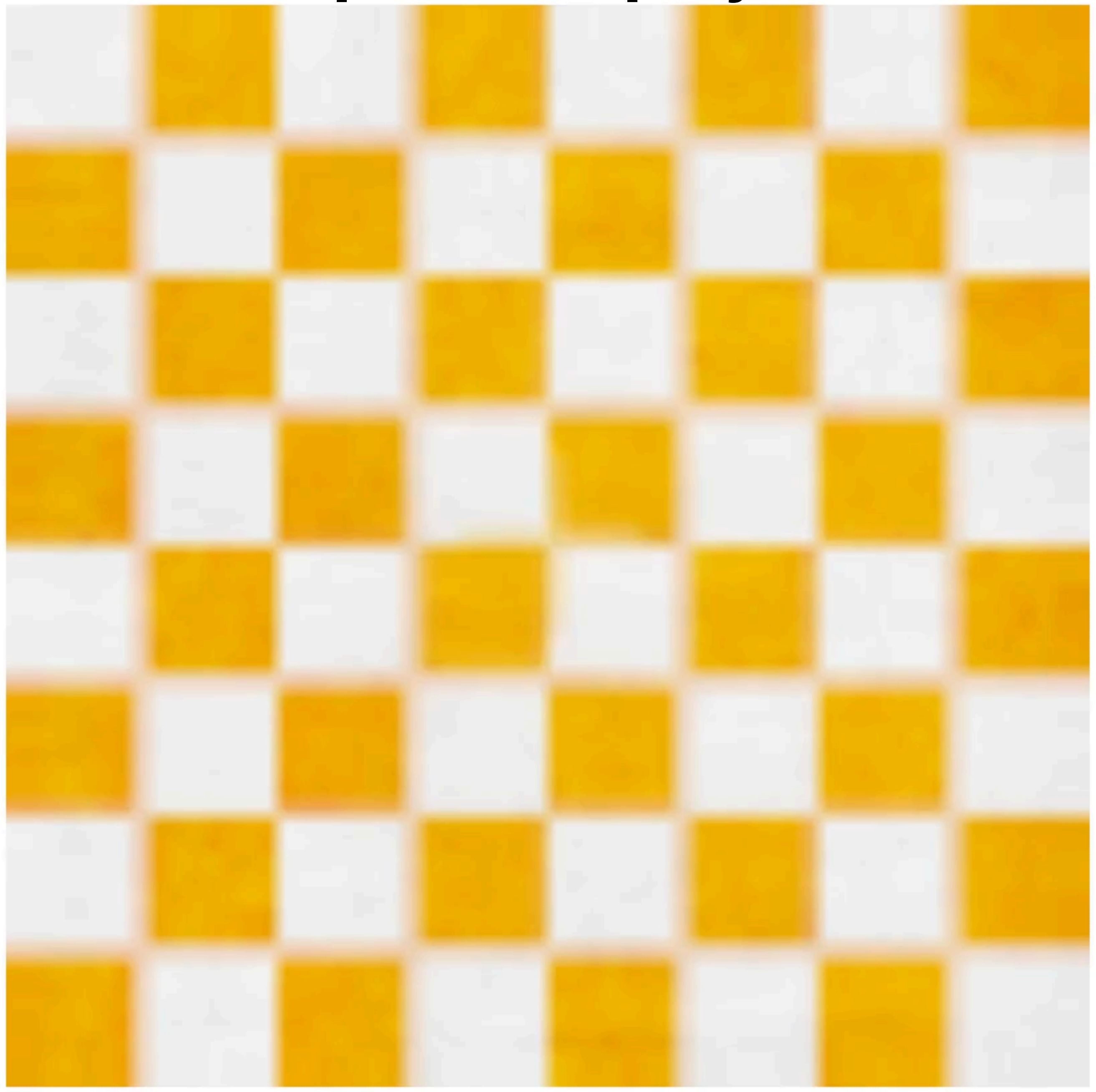
Add pressure correction to get projected, but not advected, velocities:

$$\mathbf{u}^+ = \mathbf{u}^* - \frac{\Delta t \nabla^2}{\rho} p$$

No pressure projection



With pressure projection

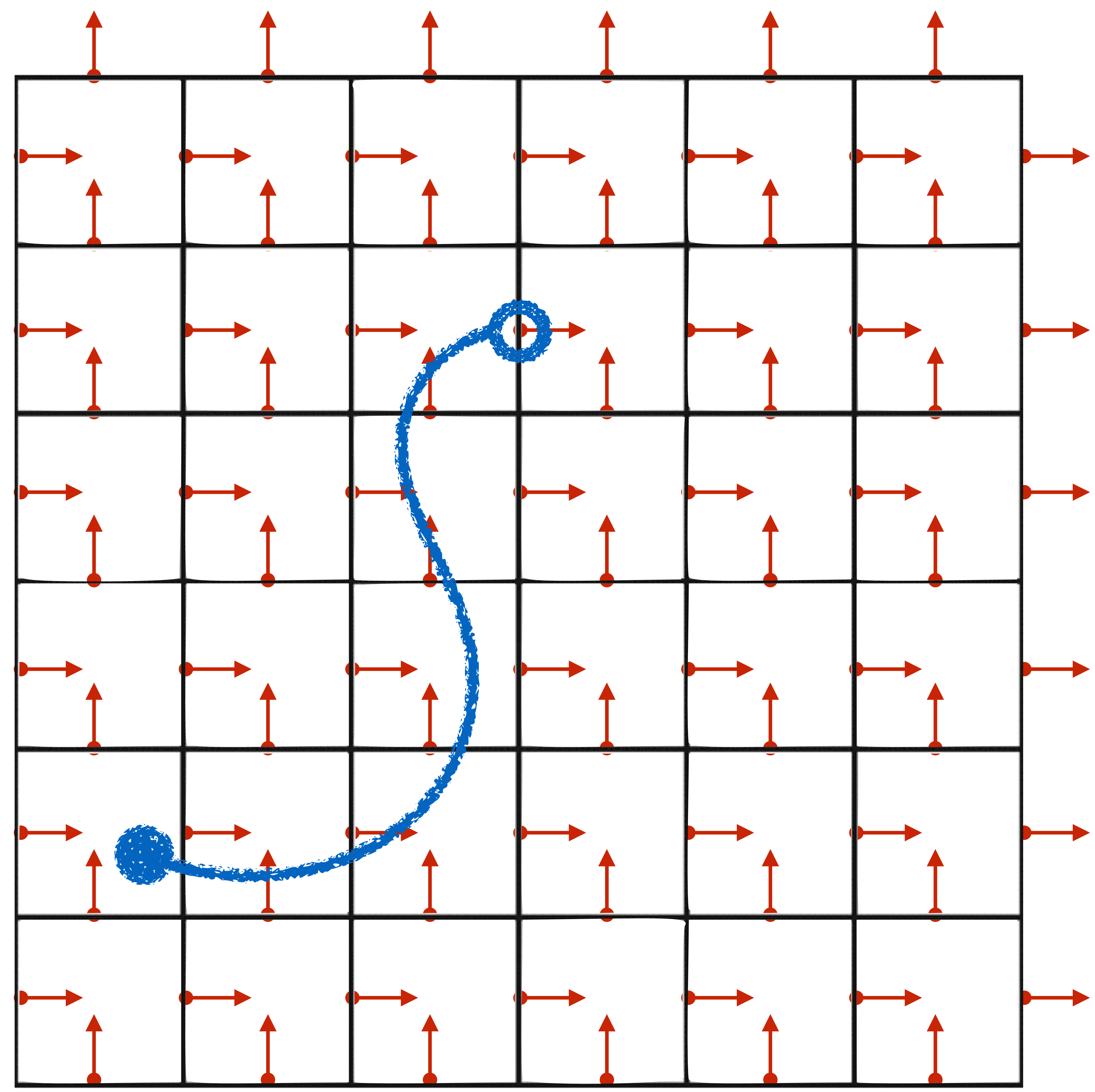


Semi-Lagrangian Advection

(A method of characteristics)

Instead of using 2nd order advection term, pick up the values and move them!

- For each location
  - Track backward through grid for Δt
  - Interpolate value
  - Copy to new location



Note: This works for other quantities besides velocity.

Note: Vector values should be rotated based on flow, but most people don't do this.

Note: Backtrace is done in one or more substeps.

Semi-Lagrangian Advection

Final velocity is:

$$\mathbf{u}^{t+\Delta t} = \text{advect}\left(\mathbf{u}^* - \frac{\Delta t\,\nabla^2}{\rho}p\right)$$

Unconditionally stable

Large steps introduce extra damping

- Viscosity term often omitted as unwanted

Stable Fluids

Demo by Amanda Ghassaei

https://apps.amandaghassaei.com/gpu-io/examples/fluid/

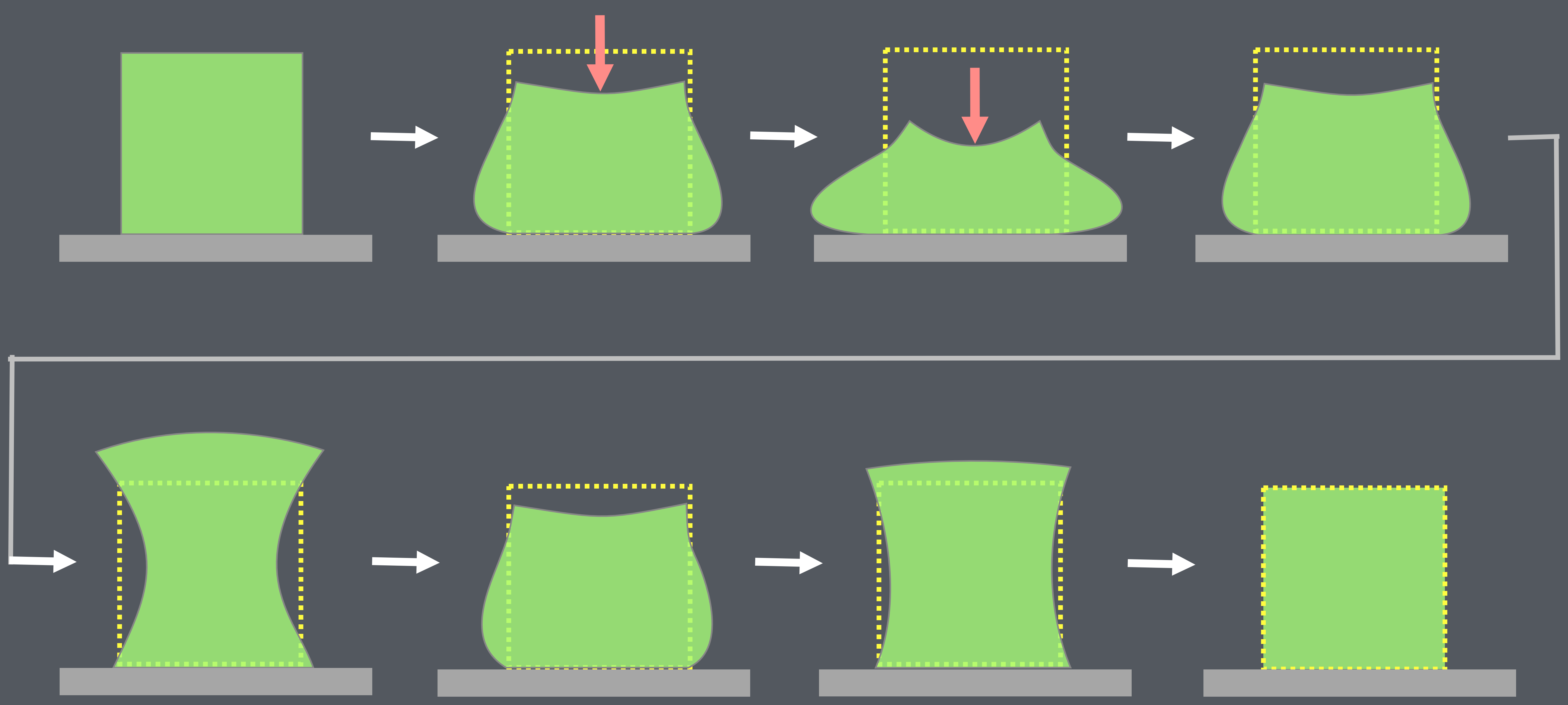
Things to notice:

- In pressure view you can see grid cells
  - You don't see them when simulation is rendered!
- Note how much damping there is
- Note how pressure changes as cursor is moved

Viscoelasticity

Ideal Elastic Solids

- Deformations are recoverable



Viscoelasticity

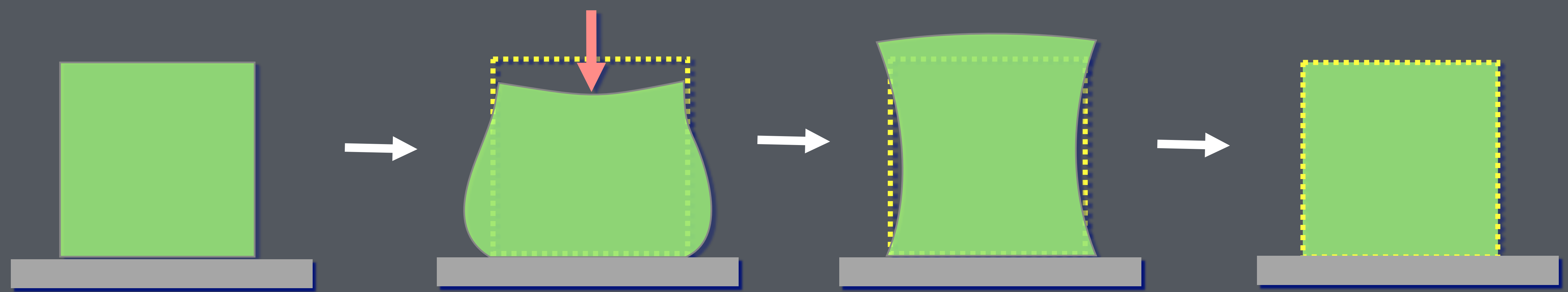
Ideal Fluids

- Deformations are permanent

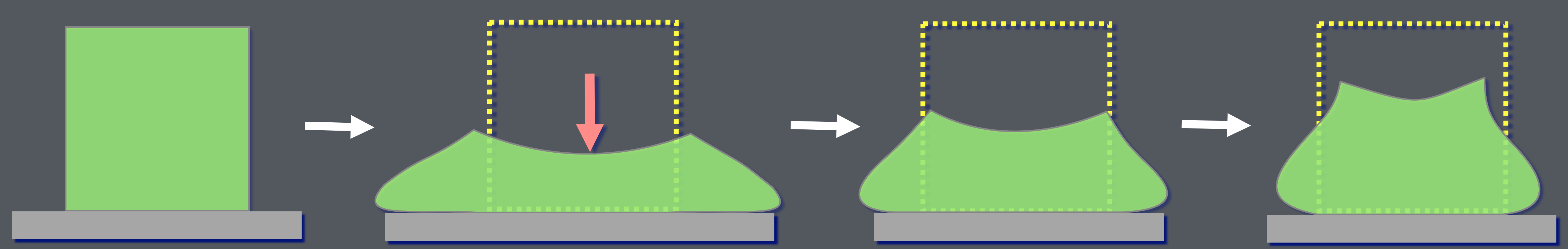
Viscoelasticity

Viscoelastic Fluids

1. Elastic deformation until a limit (yield point)



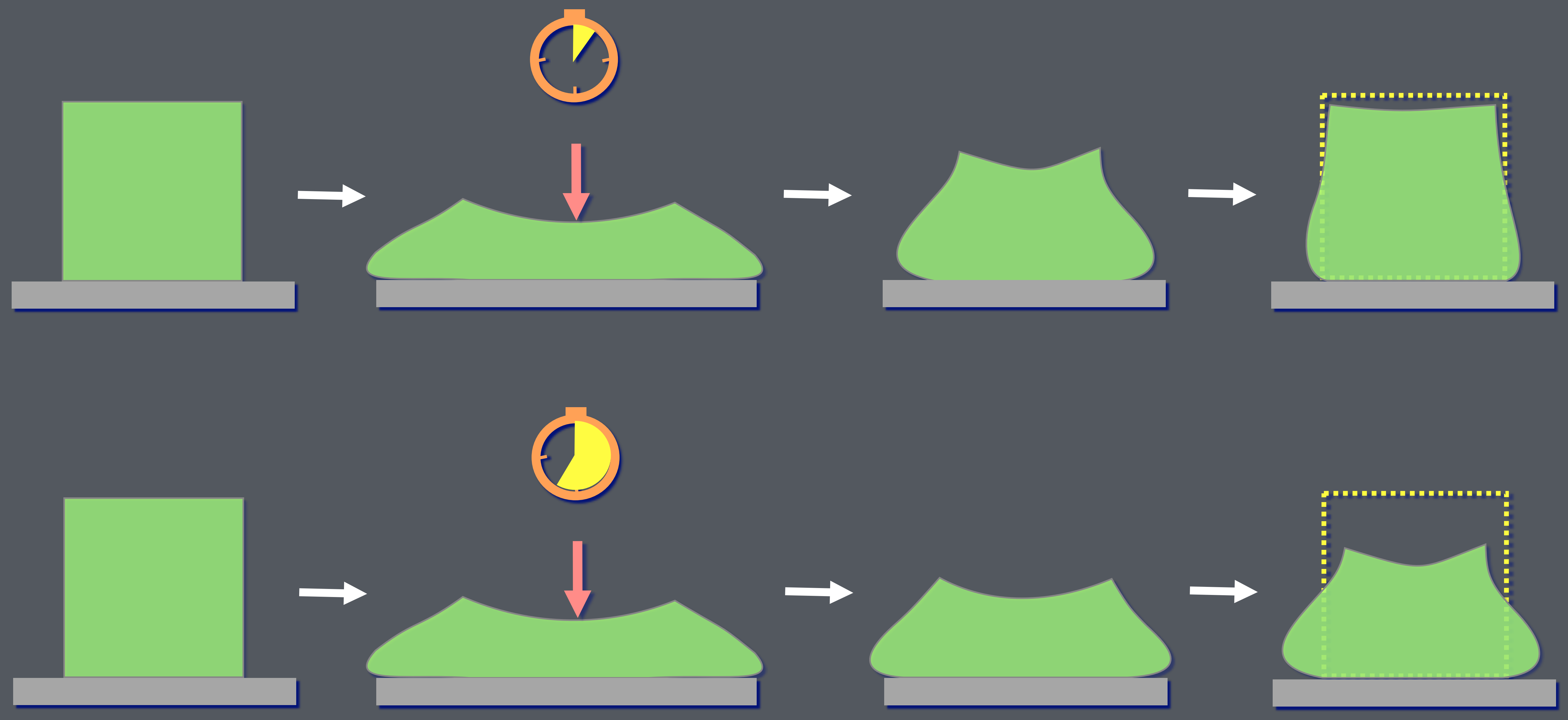
2. Some permanent (plastic) deformation after the limit



Viscoelasticity

Creep

- Plastic deformation occurs over time



Viscoelasticity

Strategy

- Start with an Eulerian based fluid simulation
- Add elastic behavior:
  - Elastic forces
  - Plastic yielding
  - Creep

Extra material for 284A

Viscoelasticity

Fluid Simulation

- Navier-Stokes
  (Incompressible)

$$\frac{\partial \boldsymbol{u}}{\partial t} = -(\boldsymbol{u} \cdot \nabla)\boldsymbol{u} - \frac{\nabla p}{\rho} + \frac{\mu_v}{\rho}\nabla^2 \boldsymbol{u} + \frac{\boldsymbol{f}}{\rho}$$

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

Viscoelasticity

Viscoelastic Fluid Simulation

- Modified Navier-Stokes
  (Incompressible)

$$\frac{\partial \boldsymbol{u}}{\partial t} = -(\boldsymbol{u} \cdot \nabla)\boldsymbol{u} - \frac{\nabla p}{\rho} + \frac{\mu_v}{\rho}\nabla^2 \boldsymbol{u} + \frac{\boldsymbol{f}}{\rho} + \boxed{\frac{\mu_e \nabla \cdot \boldsymbol{\epsilon}}{\rho}}$$

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

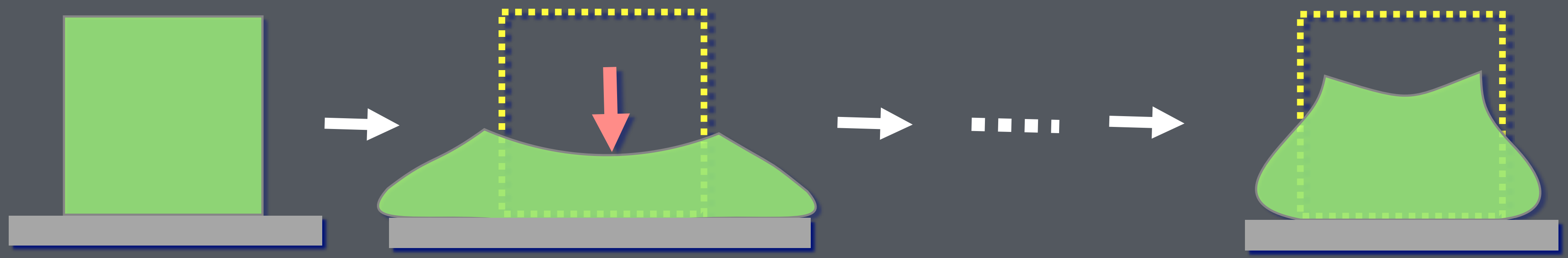
elastic forces

- Need to update strain $\boldsymbol{\epsilon}$

Viscoelasticity

Decomposing Strain

- Strain measures total deformation
- Plastic strain: permanent deformation
- Elastic strain: recoverable (temporary) deformation



$$\epsilon^{\text{Tot}} = \underline{\epsilon^{\text{Elc}}} + \epsilon^{\text{Plc}}$$

Viscoelasticity

Rate of Strain

- No deformation function available in Eulerian formulation
- Instead compute strain-rate and integrate

$$\dot{\epsilon}^{\mathsf{Tot}} = \frac{1}{2}\left(\nabla\, \boldsymbol{u} + (\nabla\, \boldsymbol{u})^{\mathsf{T}}\right)$$

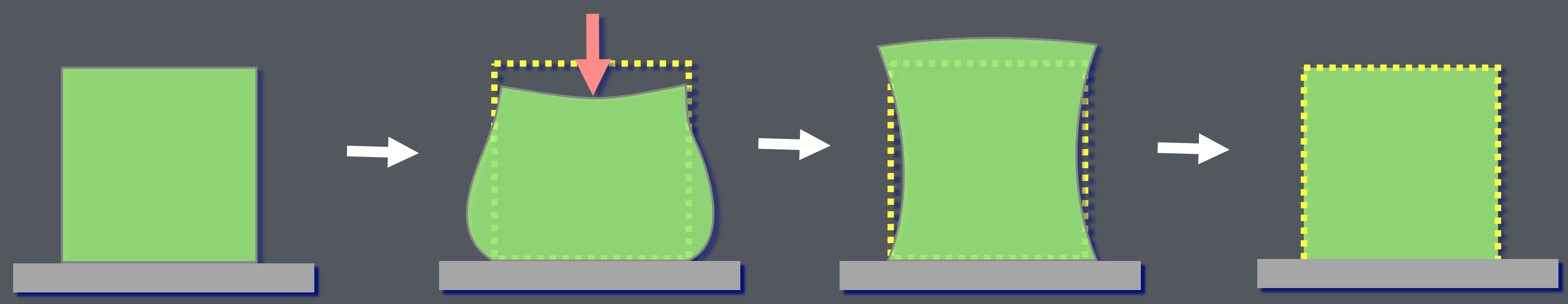
$$\epsilon^{\mathsf{Tot}} = \epsilon_0^{\mathsf{Tot}} + \int_0^t \dot{\epsilon}^{\mathsf{Tot}}\, \mathrm{d}t$$

Viscoelasticity

Strain Update

γ: Plastic Yield Point

Case 1: $\|\epsilon^{\text{Elc}}\| \leq \gamma$



$$\dot{\epsilon}^{\text{Tot}} = \dot{\epsilon}^{\text{Elc}} + \dot{\epsilon}^{\text{Plc}}$$

$$\dot{\epsilon}^{\text{Plc}} = 0$$

$$\Rightarrow \quad \boxed{\dot{\epsilon}^{\text{Elc}} = \dot{\epsilon}^{\text{Tot}}}$$

Viscoelasticity

Putting It Together

$$\dot{\epsilon}^{\text{Elc}} = \begin{cases} \dot{\epsilon}^{\text{Tot}} & : \quad \|\epsilon^{\text{Elc}}\| \leq \gamma \\ \dot{\epsilon}^{\text{Tot}} - \alpha \frac{\epsilon^{\text{Elc}}}{\|\epsilon^{\text{Elc}}\|} \left(\|\epsilon^{\text{Elc}}\| - \gamma \right) & : \quad \|\epsilon^{\text{Elc}}\| > \gamma \end{cases}$$

where: $\dot{\epsilon}^{\text{Tot}} = \frac{1}{2} \left(\nabla u + (\nabla u)^{\mathsf{T}} \right)$

$$\dot{\epsilon}^{\text{Elc}} = \frac{1}{2} \left(\nabla u + (\nabla u)^{\mathsf{T}} \right) - \alpha \frac{\epsilon^{\text{Elc}}}{\|\epsilon^{\text{Elc}}\|} \max \left(0, \|\epsilon^{\text{Elc}}\| - \gamma \right)$$

Integrate & Advect ($\dot{\epsilon}^{\text{Elc}}$)

Plastic Flow Rate (α)

Plastic Yield Point (γ)

Viscoelasticity

Effects of α and γ

$\gamma = 0.0$ $\alpha = 25$ | $\gamma = 0.1$ $\alpha = 25$ | $\gamma = 0.5$ $\alpha = 25$

$\gamma = 0.0$ $\alpha = 500$ | $\gamma = 0.1$ $\alpha = 500$ | $\gamma = 0.5$ $\alpha = 500$

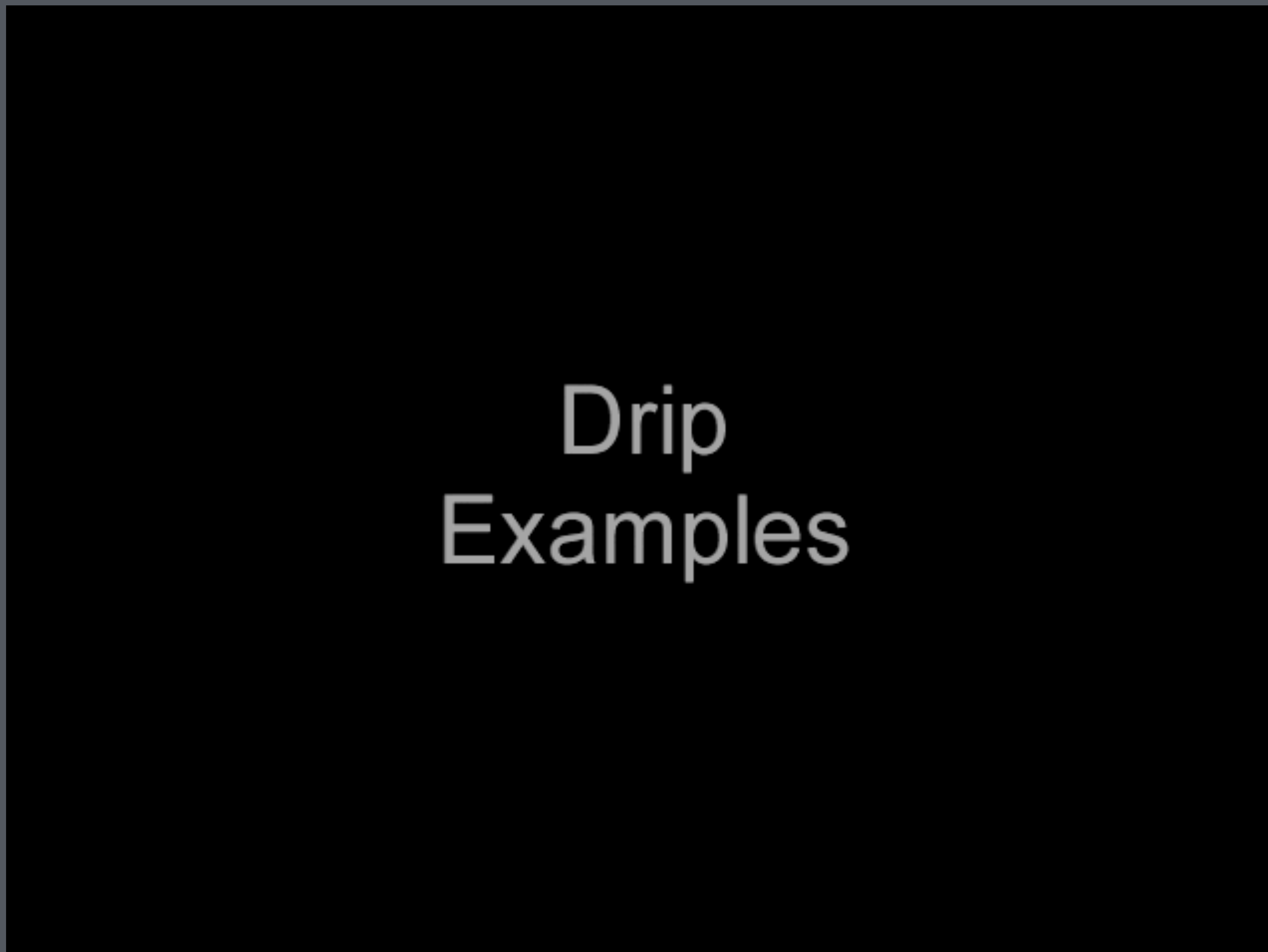
Viscoelasticity

Effects of α and γ

$\gamma = 0.0$ $\alpha = 25$ $\gamma = 0.1$ $\alpha = 25$ $\gamma = 0.5$ $\alpha = 25$

$\gamma = 0.0$ $\alpha = 500$ $\gamma = 0.1$ $\alpha = 500$ $\gamma = 0.5$ $\alpha = 500$

Viscoelasticity

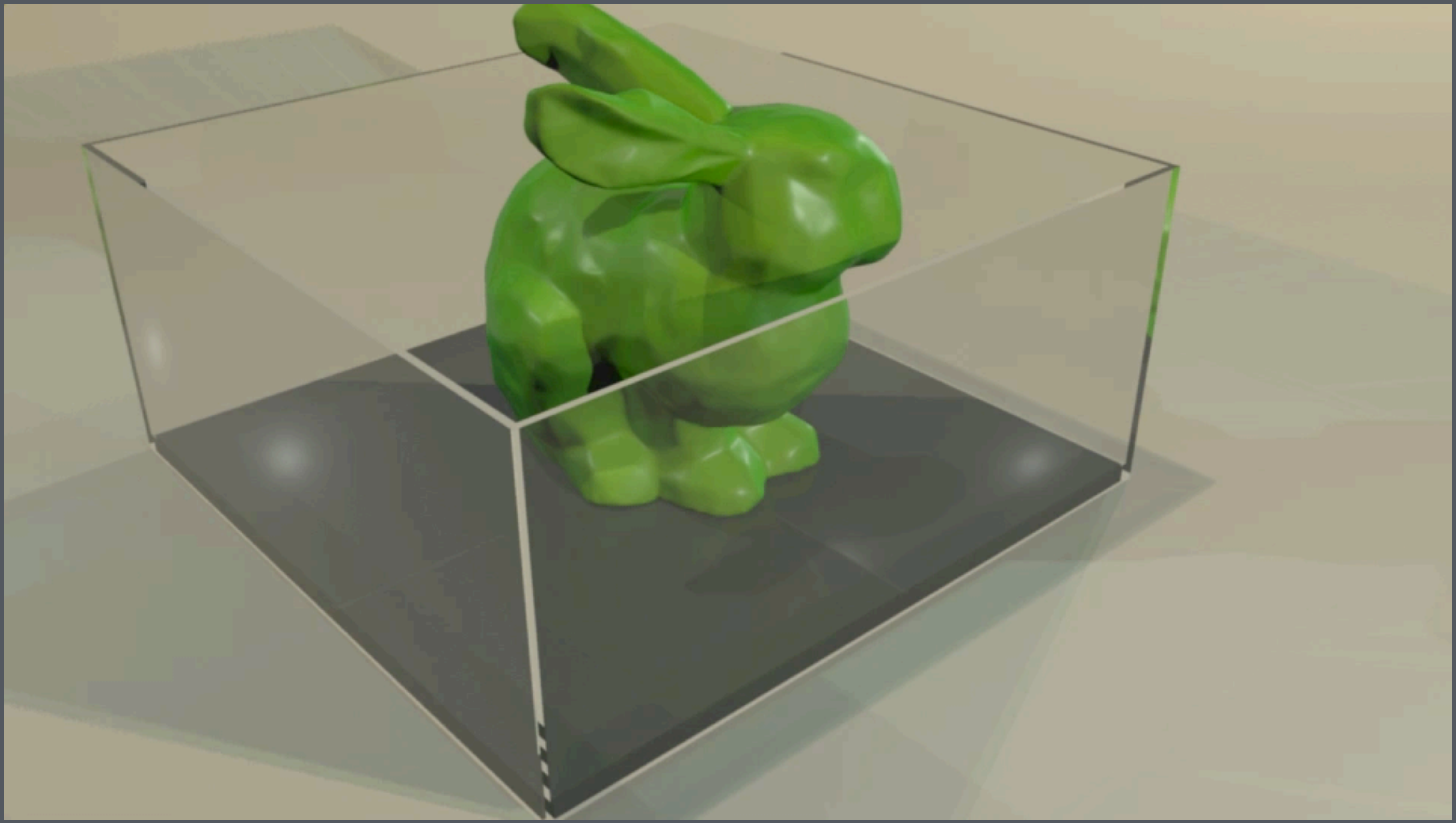


Goktekin, Bargteil, O’Brien

Viscoelasticity

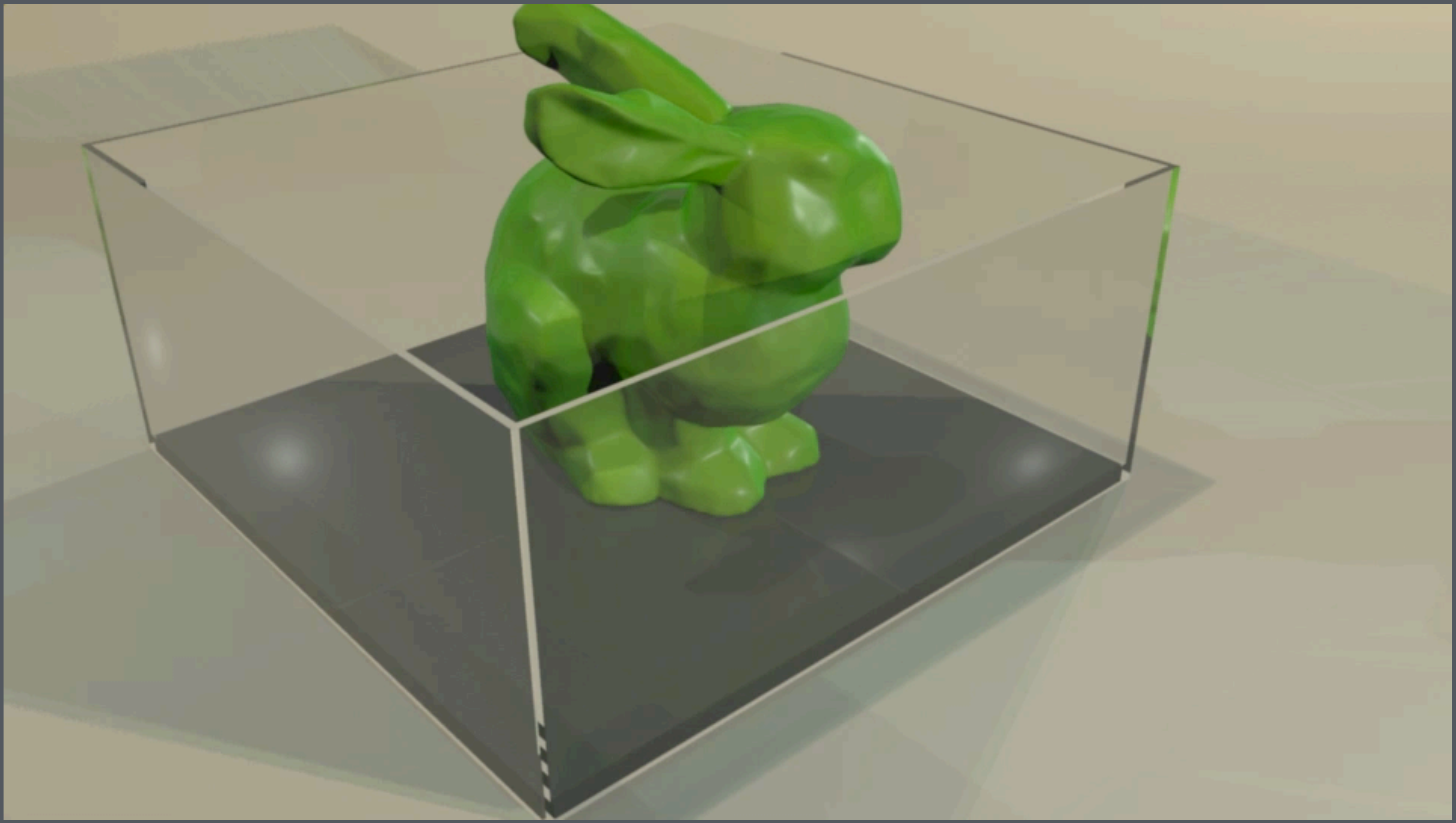
Goktekin, Bargteil, O'Brien

Other Effects: Melting



Clausen , Wicke, Shewchuk, O'Brien

Other Effects: Melting



Clausen , Wicke, Shewchuk, O'Brien

Other Effects: Strain Heating

Clausen , Wicke, Shewchuk, O'Brien

Other Effects: Strain Heating

Clausen , Wicke, Shewchuk, O'Brien

Other Effects: Strain Heating

Clausen , Wicke, Shewchuk, O'Brien

Other Methods

Particle-based fluids

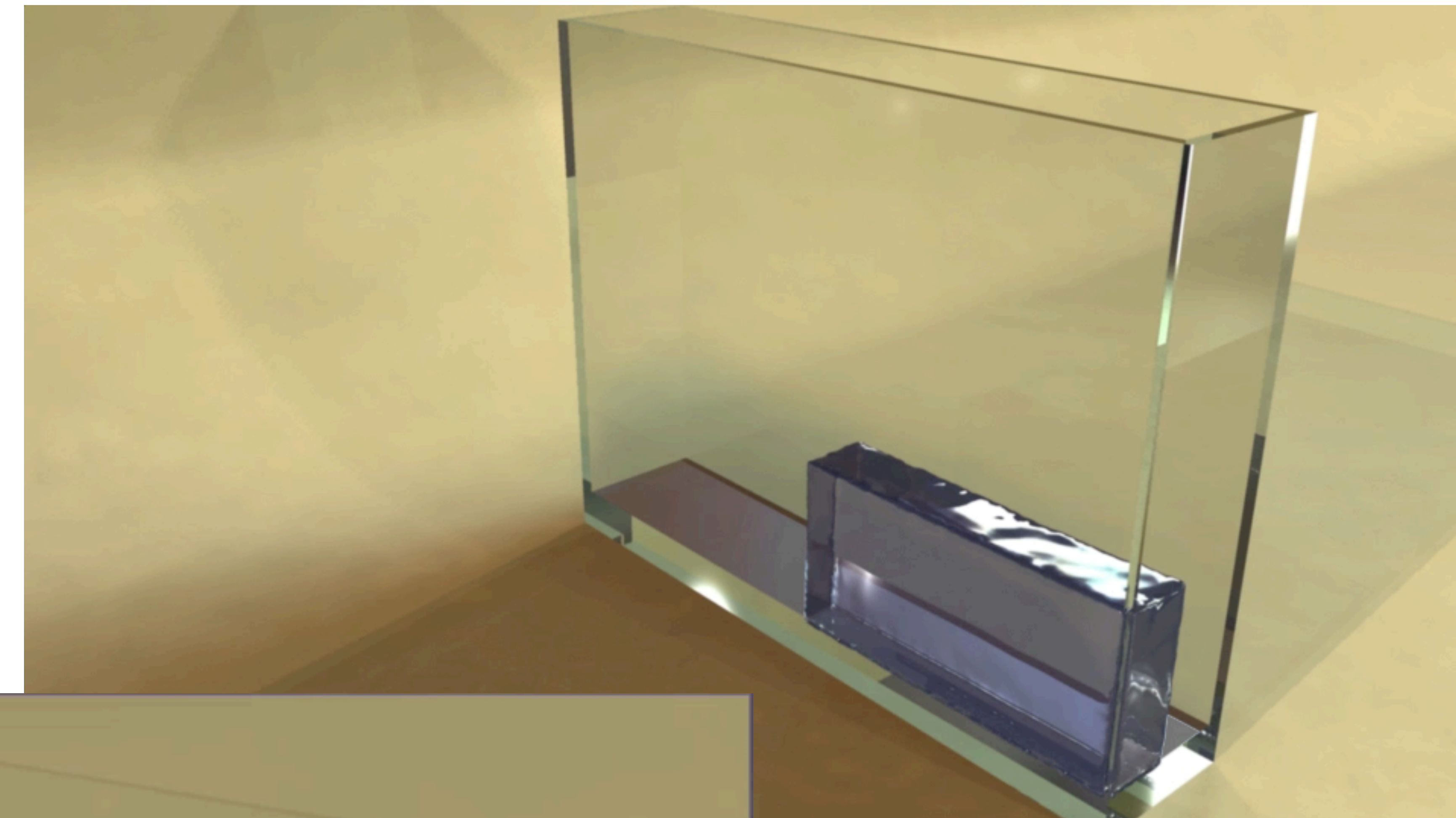
- SPH
  - Many variations
- FLIP/PIC

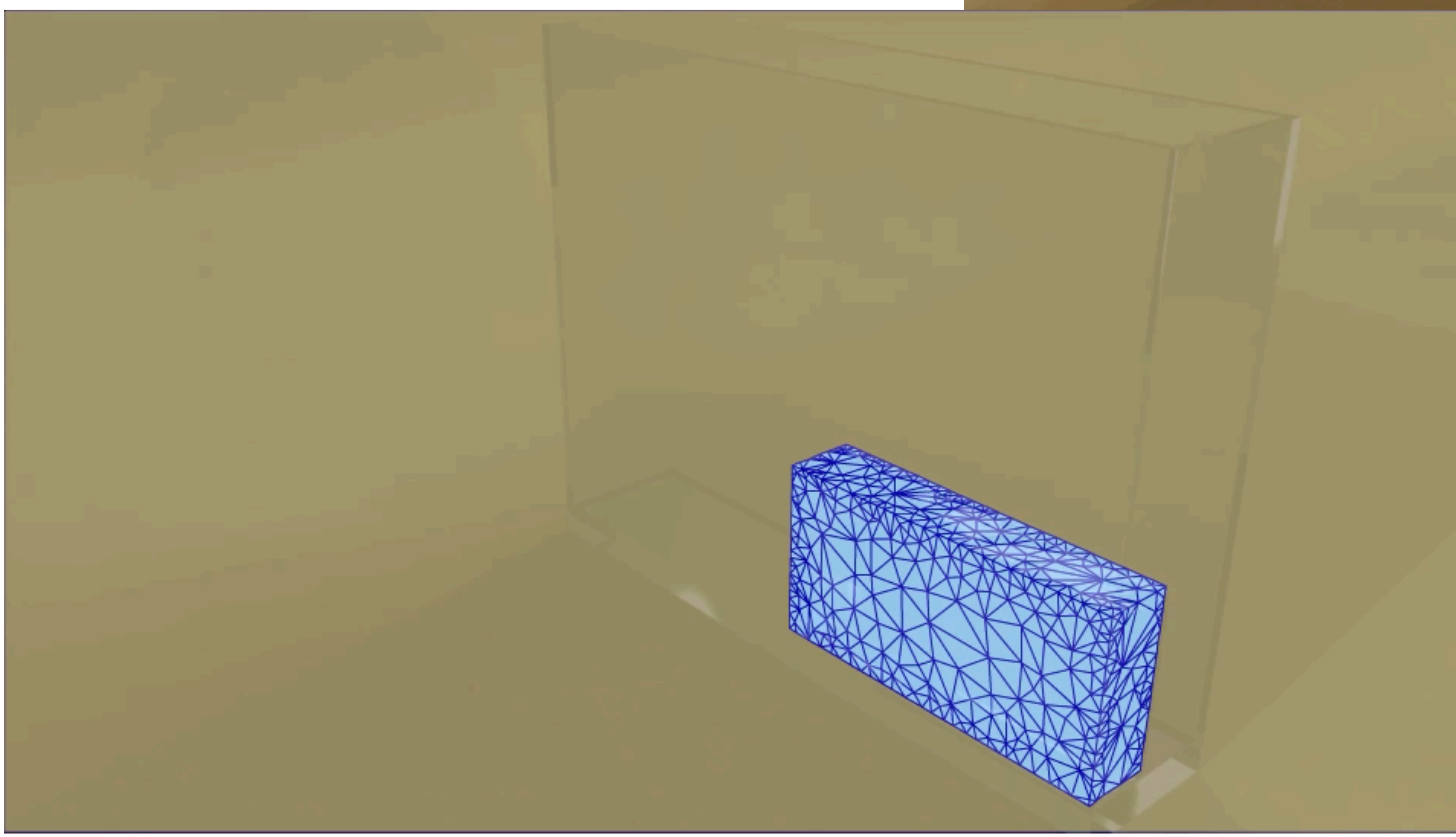
Mesh-based fluids

- Eulerian Meshes
- Lagrangian Meshes

Other Methods

Lagrangian Mesh Fluids

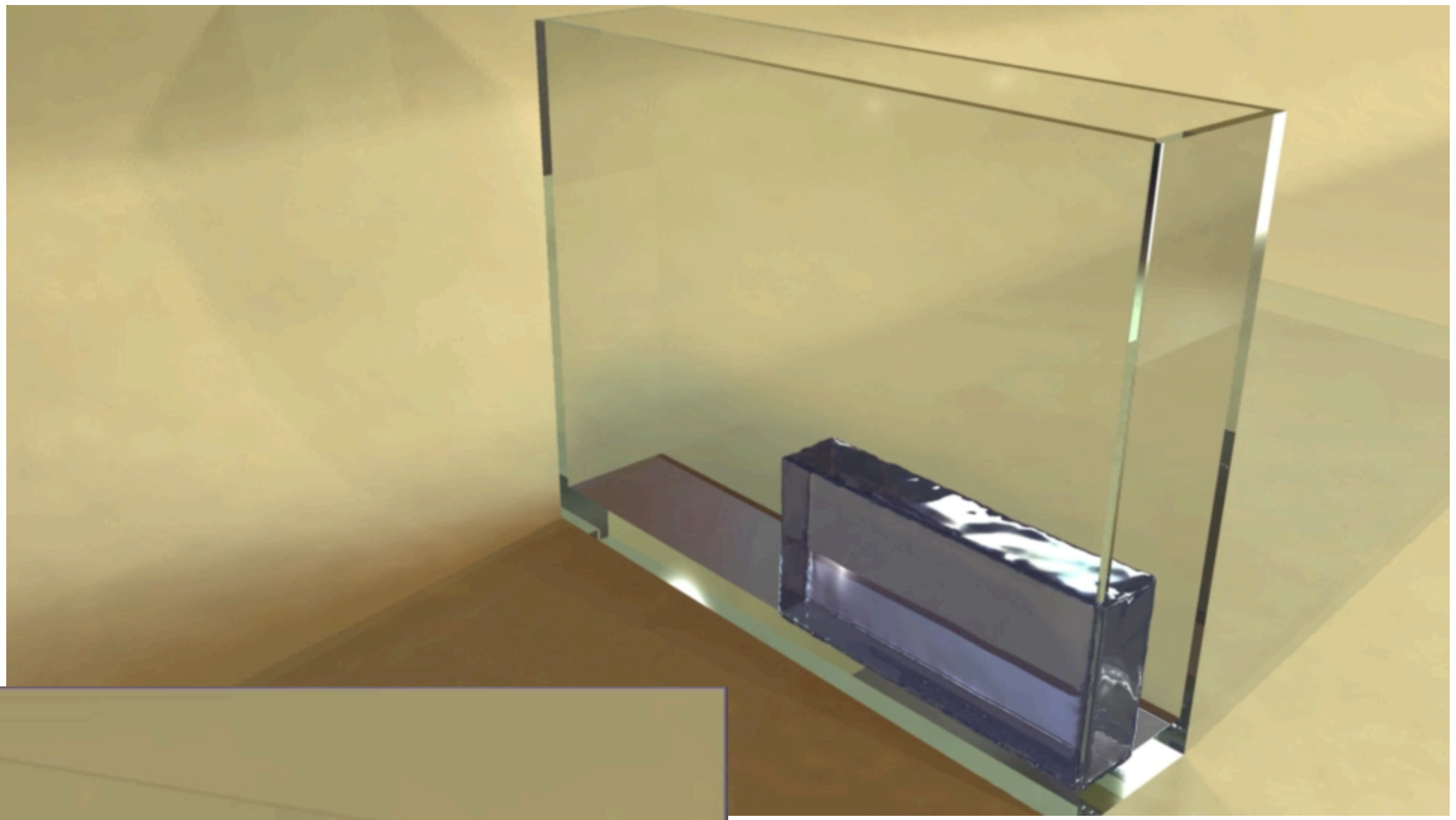


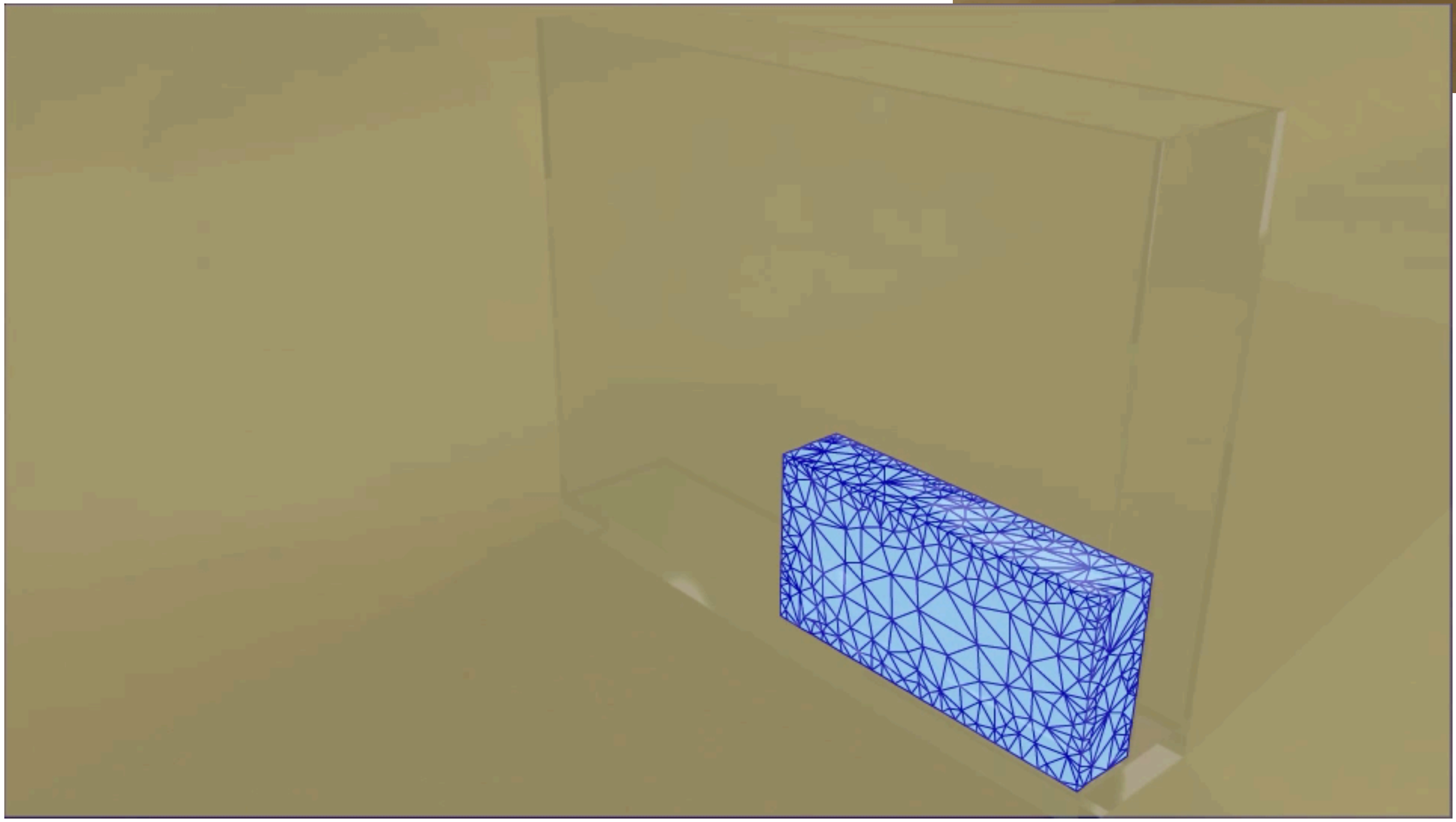


Clausen , Wicke, Shewchuk, O'Brien

Other Methods

Lagrangian Mesh Fluids





Clausen , Wicke, Shewchuk, O'Brien

Still to come:
Surface Tracking and Collisions

Acknowledgments

This slide set contain contributions from:

- Kayvon Fatahalian
- David Forsyth
- Pat Hanrahan
- Angjoo Kanazawa
- Steve Marschner
- Ren Ng
- James F. O'Brien
- Mark Pauly