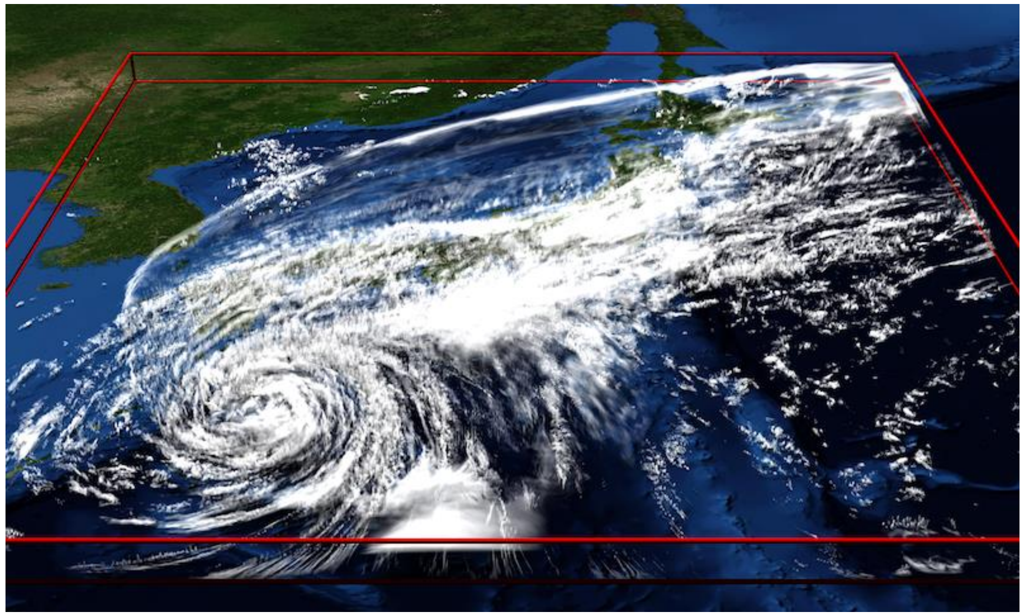
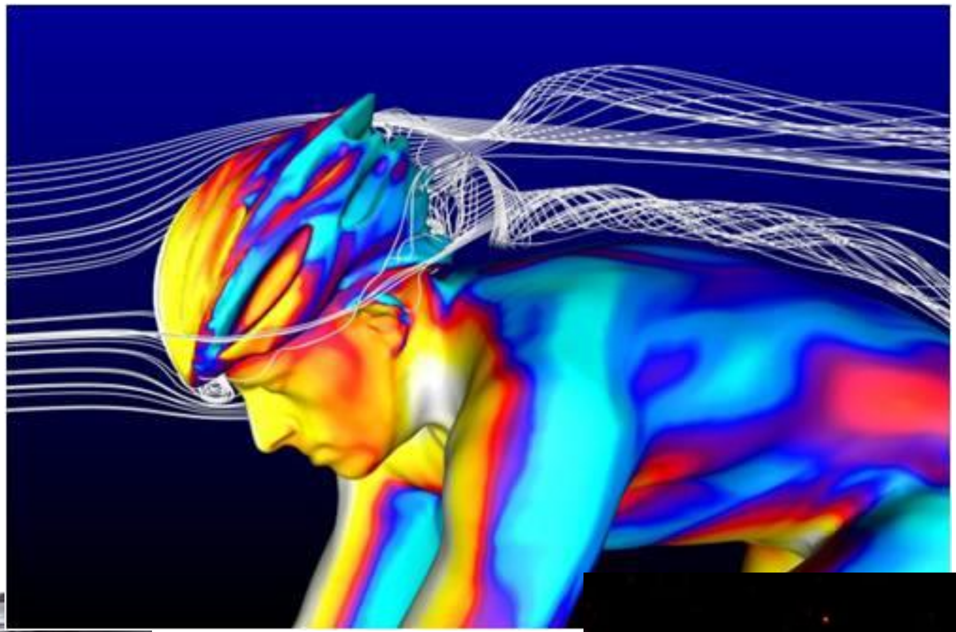
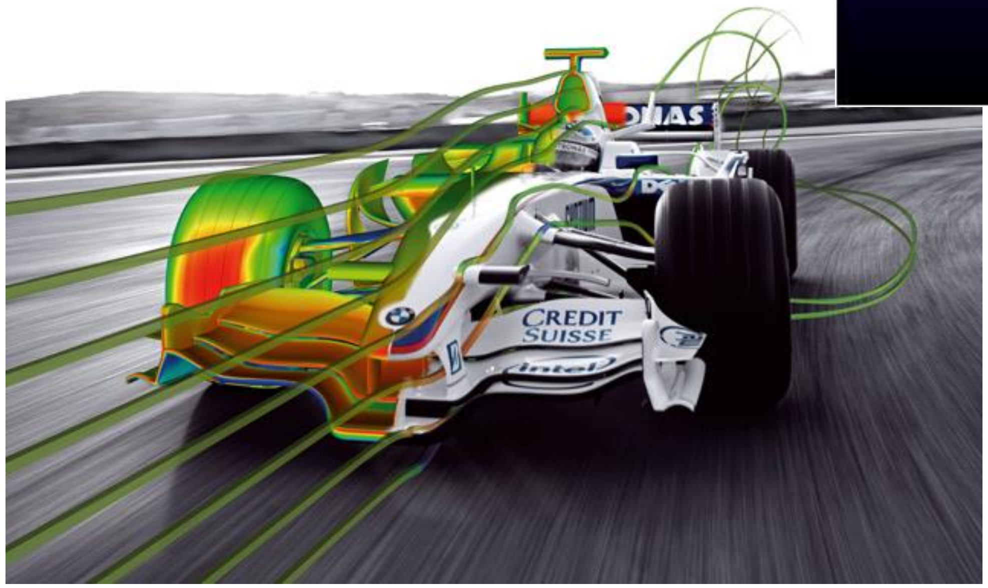
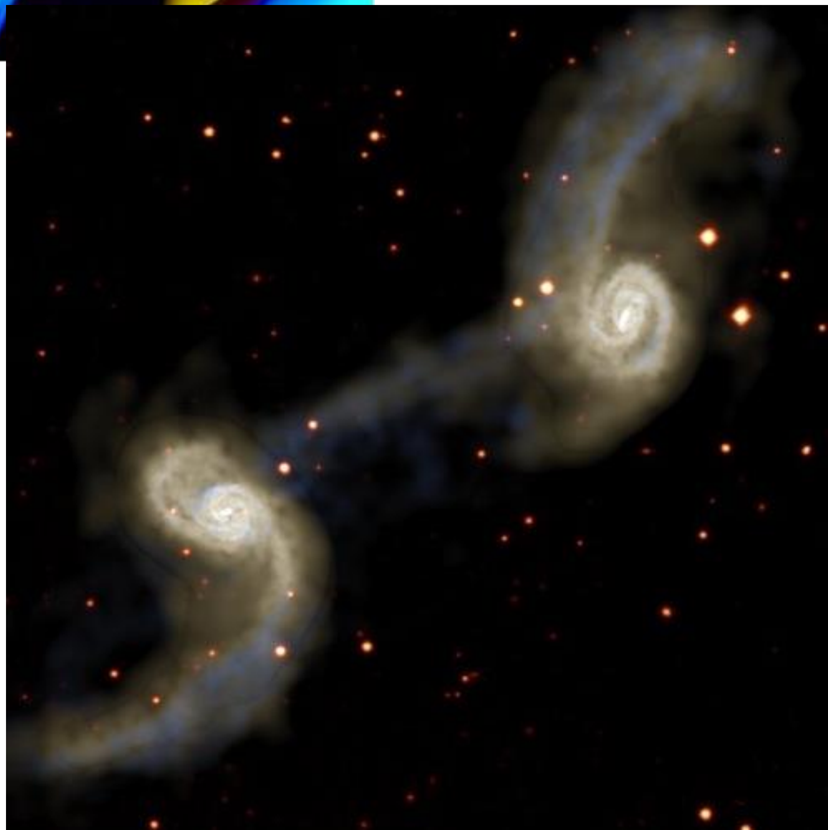
Fluid Simulation 1
Grid-based Methods

CFD







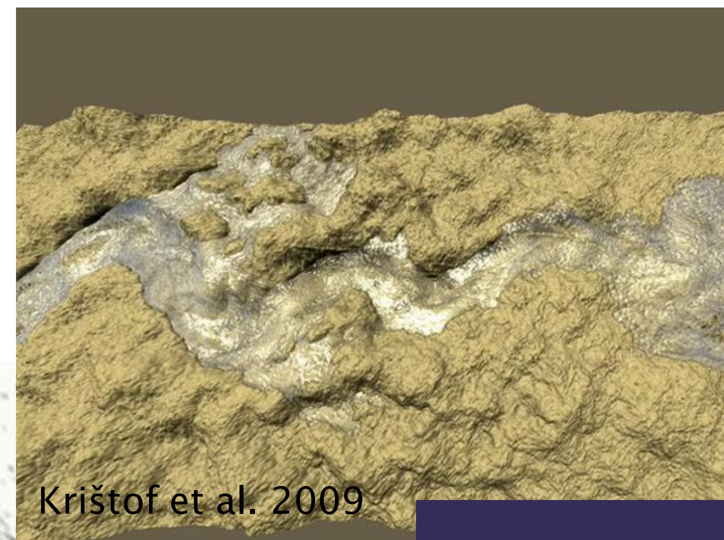


Graphics

Feature films

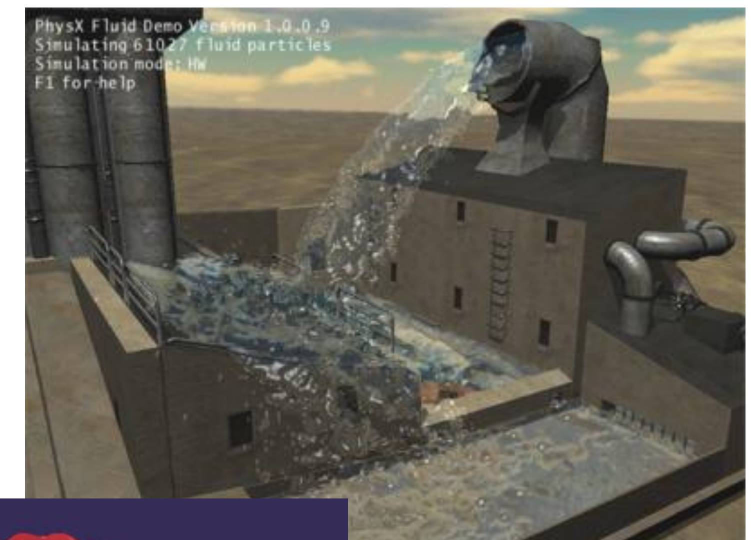
Warner Bros / ILM

Natural phenomena



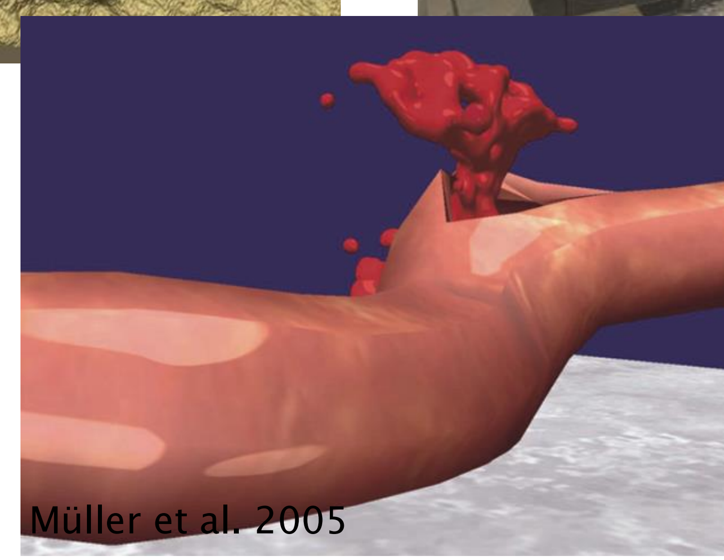
Krištof et al. 2009

Games



Fusion CI Studios

Commercials



Müller et al. 2005

Surgery simulators

Pre-computed ⟷ **Real-time**

Films

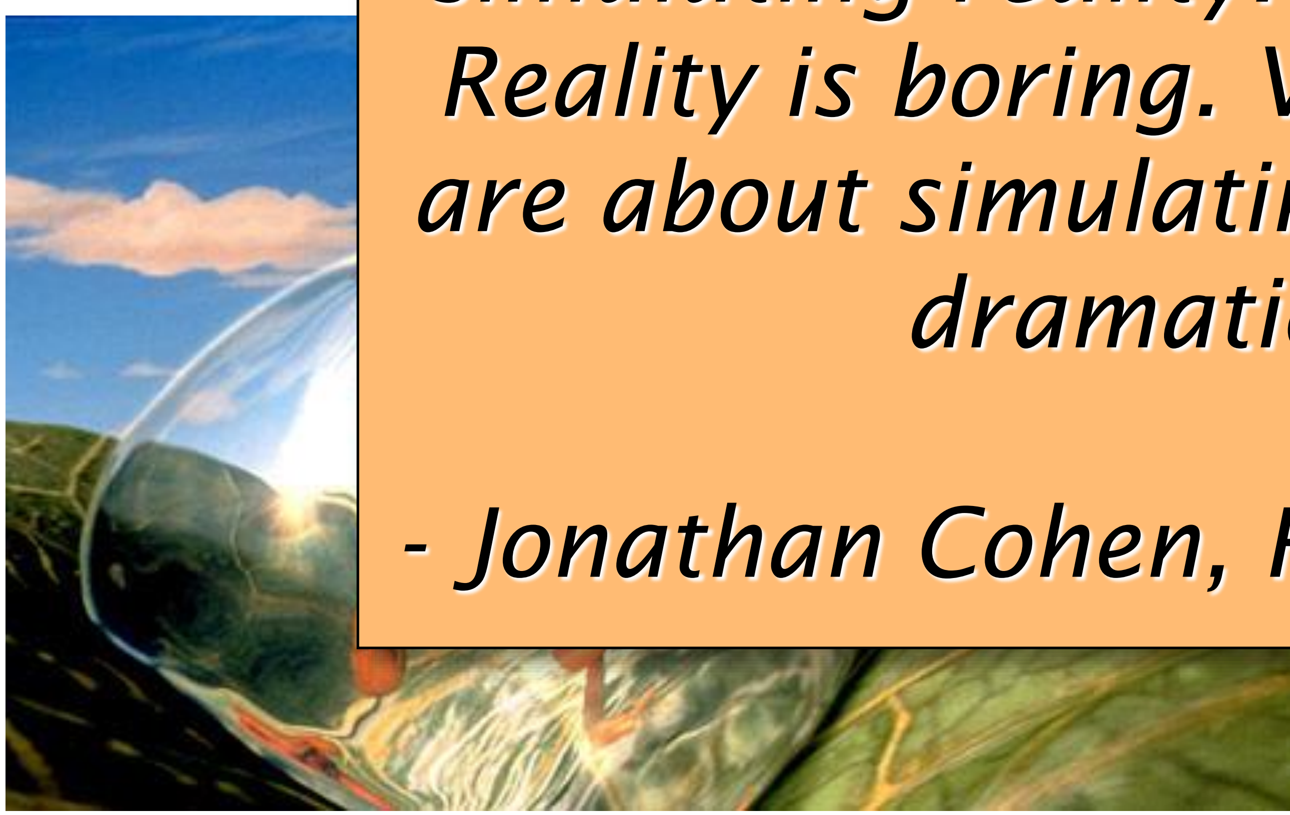
- Autodesk, Digital Domain, ILM, Next Limit, Rhythm & Hues, Scanline, Weta, ...

The Abyss, 89

Fluid effects were animated manually, e.g. digitally painting a water skin over the model

Films

- Aut Limit, Rhy



“It’s a common misconception that visual effects are about simulating reality. They’re not. Reality is boring. Visual effects are about simulating something dramatic,”

- Jonathan Cohen, Rhythm&Hues

th fluid

-> very time consuming
-> [Stam99]

Films

- ILM, Digital Domain: Control!
  -> fans, invisible whirlpools, hidden suction pumps
- Manipulate their virtual fluids to meet the directors' and animators' desires

Pirates of the Caribbean 3, 07

ILM / Standford

Films

- ILM, Digital Domain: Control!
  -> fans, invisible whirlpools, hidden suction pumps
- Manipulate their virtual fluids to meet the directors' and animators' desires

The Day After Tomorrow, 04

Foam, mist, simple animator control

Films

Simulating Whitewater Rapids in *Ratatouille*

"Before the director and layout department could …

"After character animation, the shots returned to the FX department for a final pass to add foam, spray, mist, and bubbles…"

… sensitive to initial conditions. Initial conditions were often impossible to reproduce due to changes in the sewer tunnel design during the several weeks since the coarse simulation was done. Moreover, the sequence of shots used pieces of the coarse simulation out of order, and in some cases the speed of the water mesh motion had been scaled up to match a desired camera move.

This meant that we had to construct new detailed simulations shot-by-shot, making them match the camera moves as well as possible

… induced shape changes as the bubbles moved through the water.

References

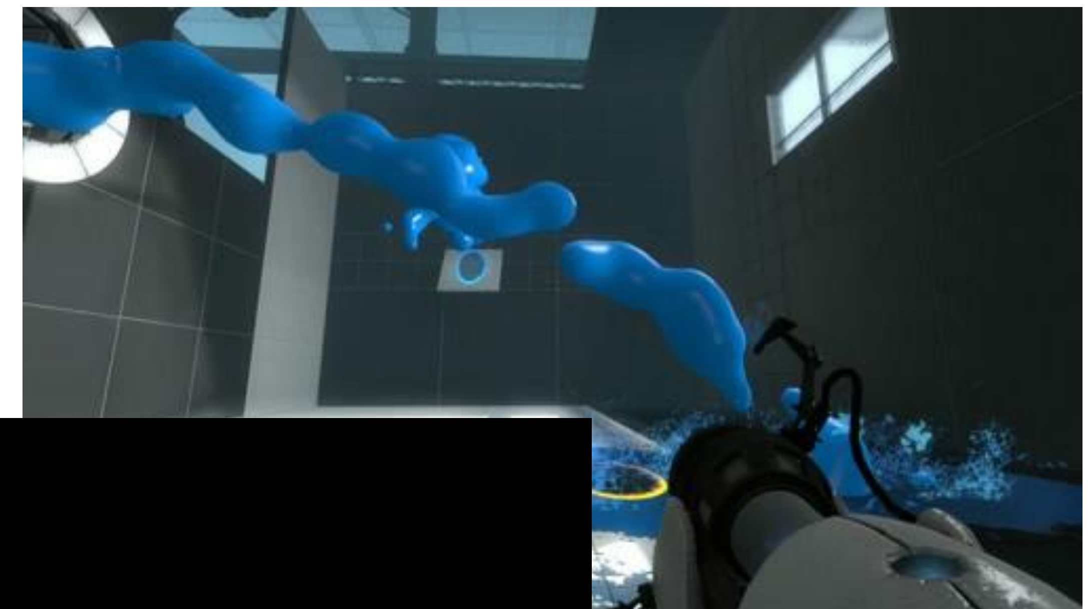
SHEN, C. 2007. Extracting temporally coherent surfaces from particles. *SIGGRAPH 2007 Sketches*.

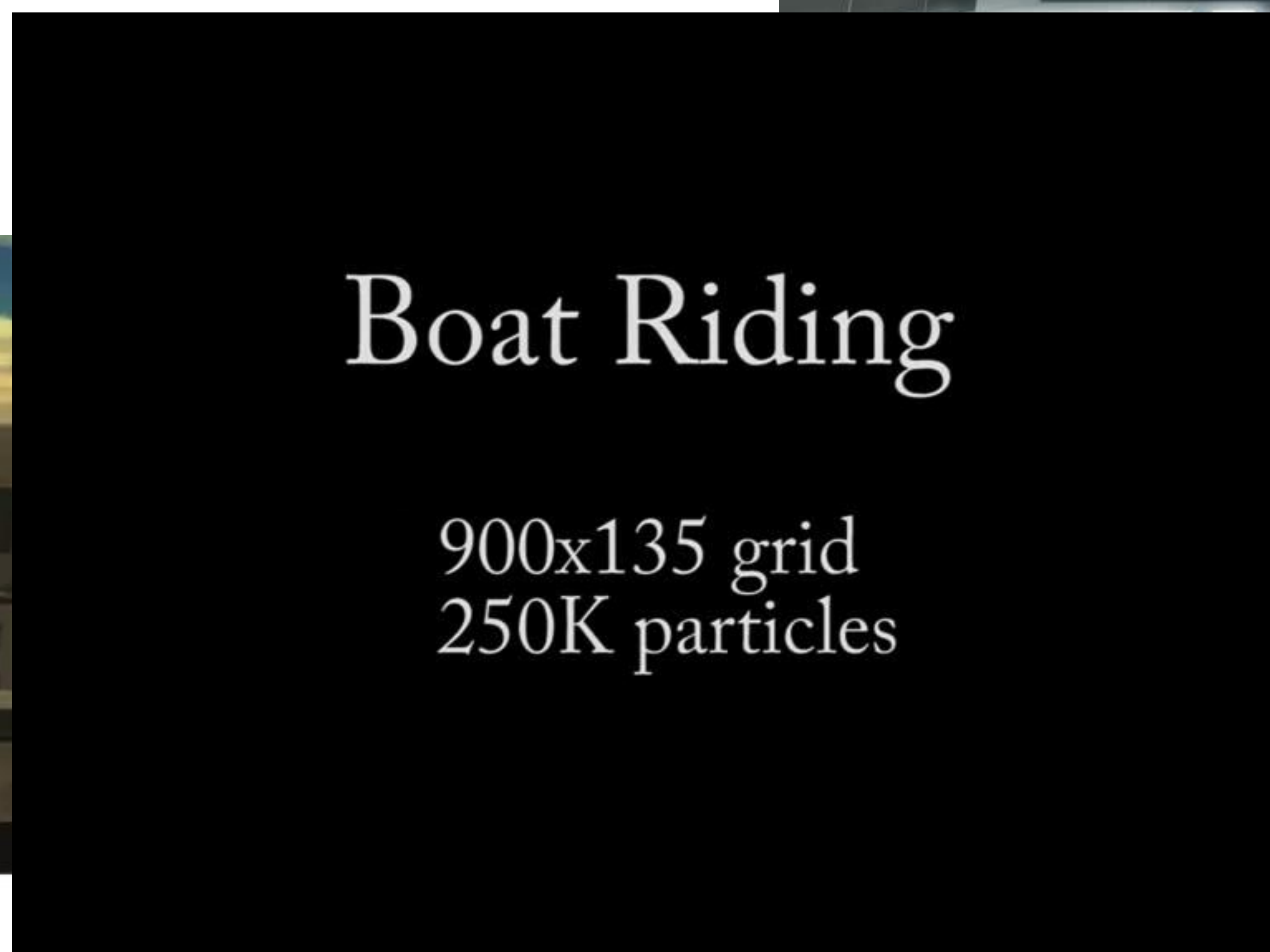
Films

Scanline VFX: 2012

Games

- **Real-time, Stability**

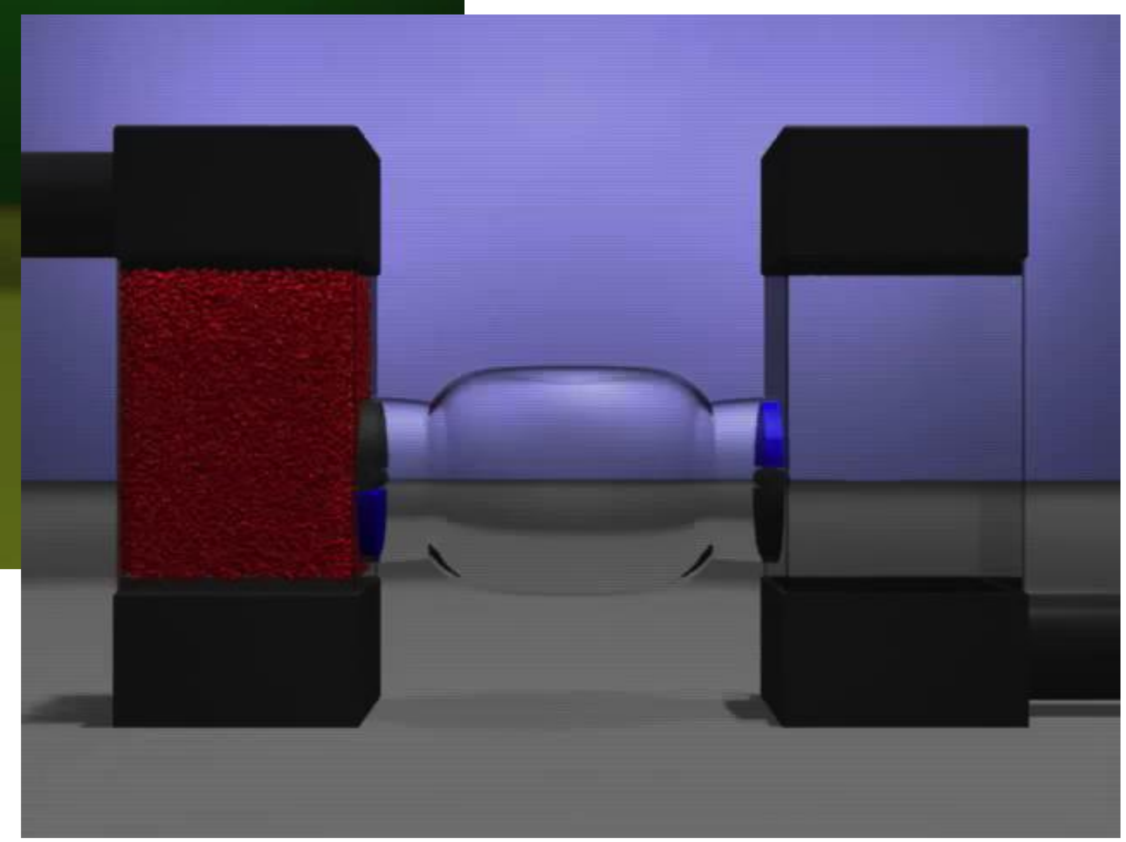




Project Motivation

[Hong et al., 07]

[Klingner et al., 06]



Outline

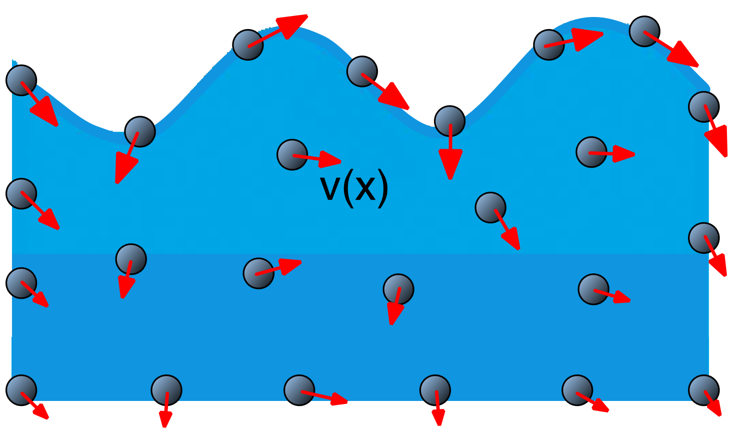
- Euler vs. Lagrange
- Navier-Stokes Equations
  - Conservation of momentum
  - Conservation of mass
- Numerical Solution
  - Discretization
  - Advection, external Forces, incompressibility

Outline

- Euler vs. Lagrange
- Navier-Stokes Equations
  - Conservation of momentum
  - Conservation of mass
- Numerical Solution
  - Discretization
  - Advection, external Forces, incompressibility

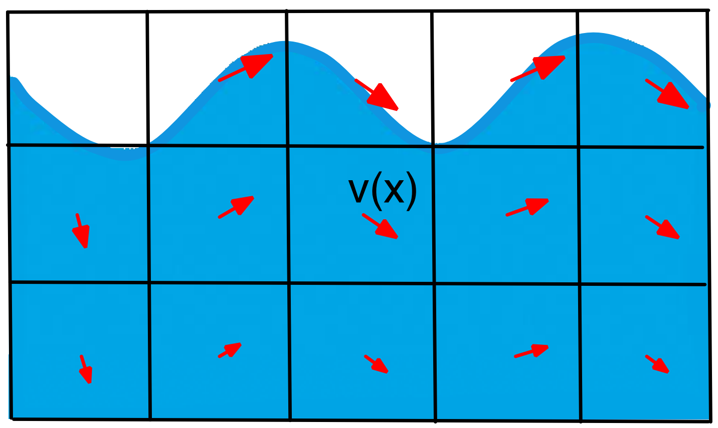
Spatial Discretization

Langrangian viewpoint



- Particles can move freely in space, carry quantities
- Fluid motion by moving particles

Eulerian viewpoint



- Fixed spatial locations
- Measure quantities as it flows past

You are in the balloon floating along with the wind, measuring the pressure, temperature, humidity etc of the air that's flowing alongside you

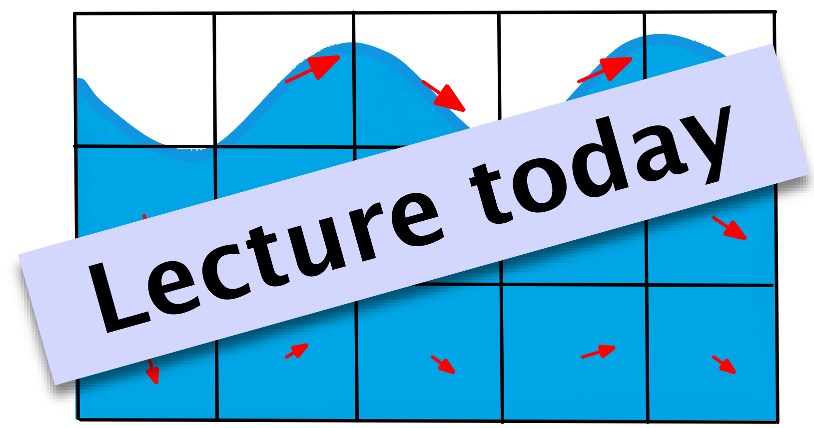
doing a weather report

You are stuck on the ground, measuring the pressure, temperature, humidity etc of the air that's flowing past

Lagrangian vs. Eulerian

Langrangian viewpoint

Eulerian viewpoint



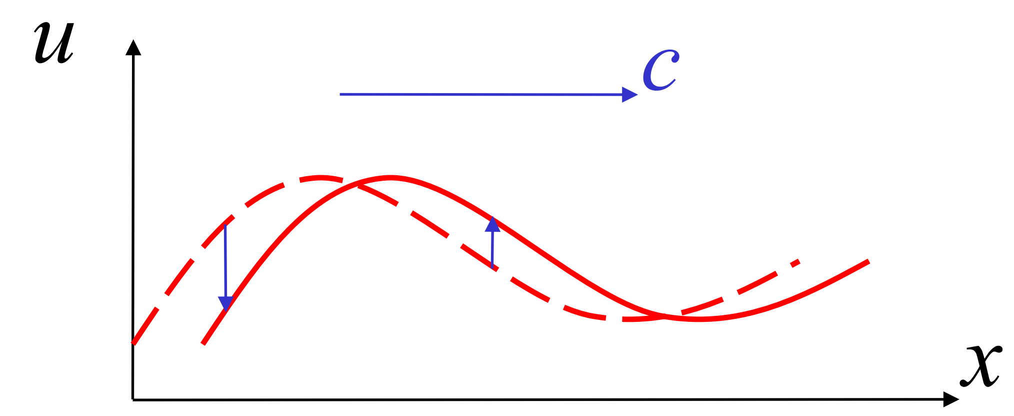
Lagrangian Models		Eulerian Models
−	Incompressibility	+
+/−	Complex Interactions	−
+	Splashes, droplets	−
+	Advection	−
−	Smooth Surfaces	+

Notation Reminder

- **Nabla operator:**
  - **Gradient (scalar -> vector):** $$\nabla u = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix}$$
  - **Divergence (vector -> scalar):** $$\nabla \cdot \mathbf{u} = u_x + u_y + u_z$$
- **Laplace operator (scalar->scalar):** $$\triangle u = \nabla^2 u = \nabla \cdot \nabla u$$ $$\triangle u = u_{xx} + u_{yy} + u_{zz}$$

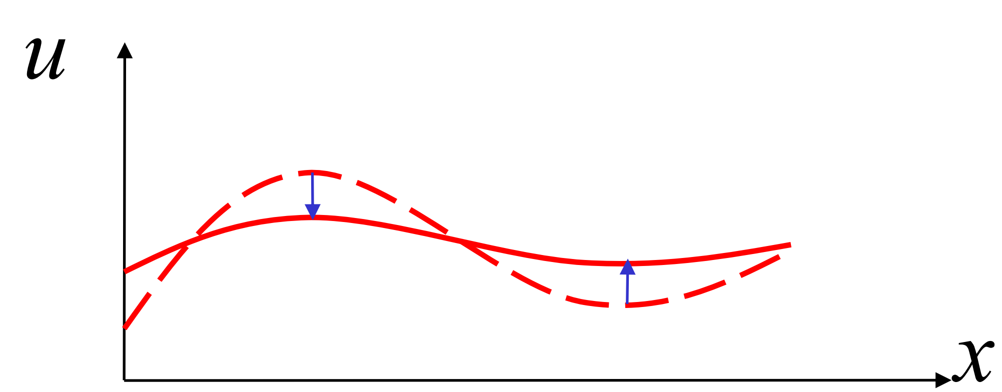
Notation Reminder

- **Advection 1D:** $$\frac{\partial u}{\partial t} = -c\frac{\partial u}{\partial x}$$



  - **Vector notation:** $$u_t = -c \cdot \nabla u$$

- **Diffusion 1D:** $$\frac{\partial u}{\partial t} = k\frac{\partial^2 u}{\partial x^2}$$



  - **Vector notation:** $$u_t = k\Delta u$$

Navier-Stokes Equations

- **2 differential equations describing velocity field over time**

Conservation of momentum:

$$\underbrace{\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}}_{\text{advection}} = \underbrace{-\frac{1}{\rho}\nabla p}_{\text{pressure}} + \underbrace{\nu \triangle \mathbf{u}}_{\text{viscosity}} + \mathbf{g}$$

Conservation of mass (incompressible):

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

Symbols

- **NS equations:**

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

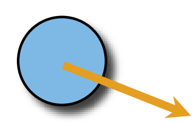
$$\underbrace{\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}}_{\text{advection}} = \underbrace{-\frac{1}{\rho}\nabla p}_{\text{pressure}} + \underbrace{\nu \triangle \mathbf{u}}_{\text{viscosity}} + \mathbf{g}$$

  - ***u*** **= velocity**
  - p **= pressure**
  - ρ **= density**
  - ***g*** **= gravity (body forces)**
  - ν **= viscosity (kinematic)**

Momentum Equation

$$\underbrace{\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}}_{\text{advection}} = \underbrace{-\frac{1}{\rho}\nabla p}_{\text{pressure}} + \underbrace{\nu \triangle \mathbf{u}}_{\text{viscosity}} + \mathbf{g}$$

- **Imagine single blob of fluid *p*:**



m = mass
V = volume
u = velocity

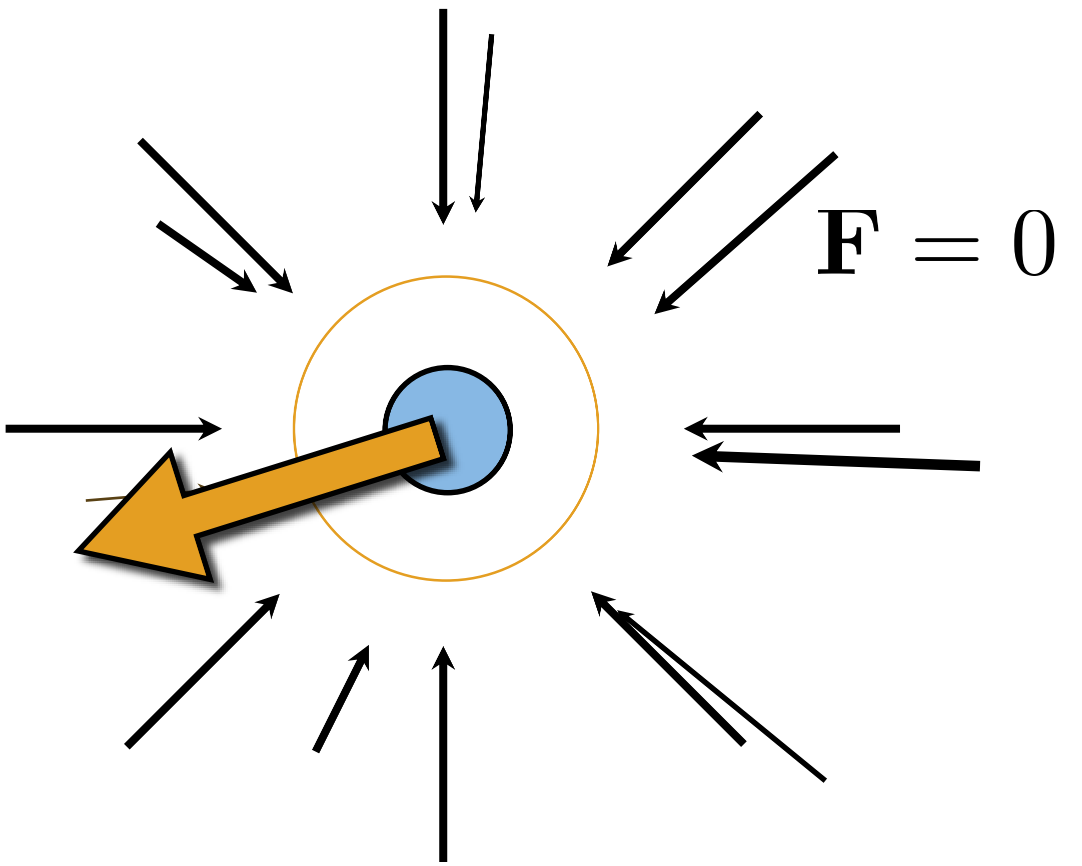
- **Start with Newton's second law:** $\mathbf{F} = m\mathbf{a}$

- **Rewrite as:** $\mathbf{F} = m\frac{D\mathbf{u}}{Dt}$

What forces act on p?

$$\underbrace{\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}}_{\text{advection}} = \underbrace{-\frac{1}{\rho}\nabla p}_{\text{pressure}} + \underbrace{\nu \triangle \mathbf{u}}_{\text{viscosity}} + \mathbf{g}$$

- **Gravity:** $m\mathbf{g}$
- **Pressure:**
  - **Consider surrounding fluid**
  - **Force: negative gradient -> towards largest pressure decrease / low pressure areas**

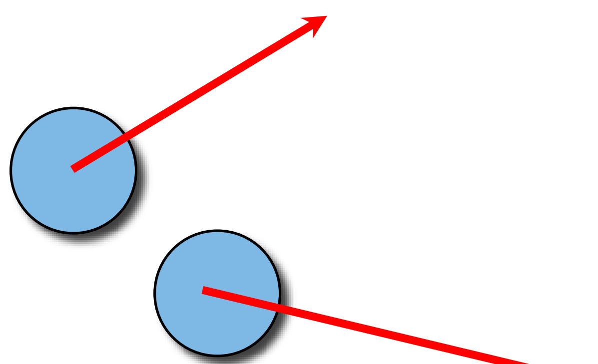


- **Compute as:** $-\nabla p \; V$

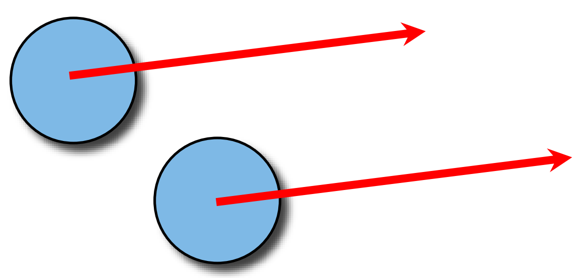
What forces act on p?

$$\underbrace{\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}}_{\text{advection}} = \underbrace{-\frac{1}{\rho}\nabla p}_{\text{pressure}} + \underbrace{\nu \triangle \mathbf{u}}_{\text{viscosity}} + \mathbf{g}$$

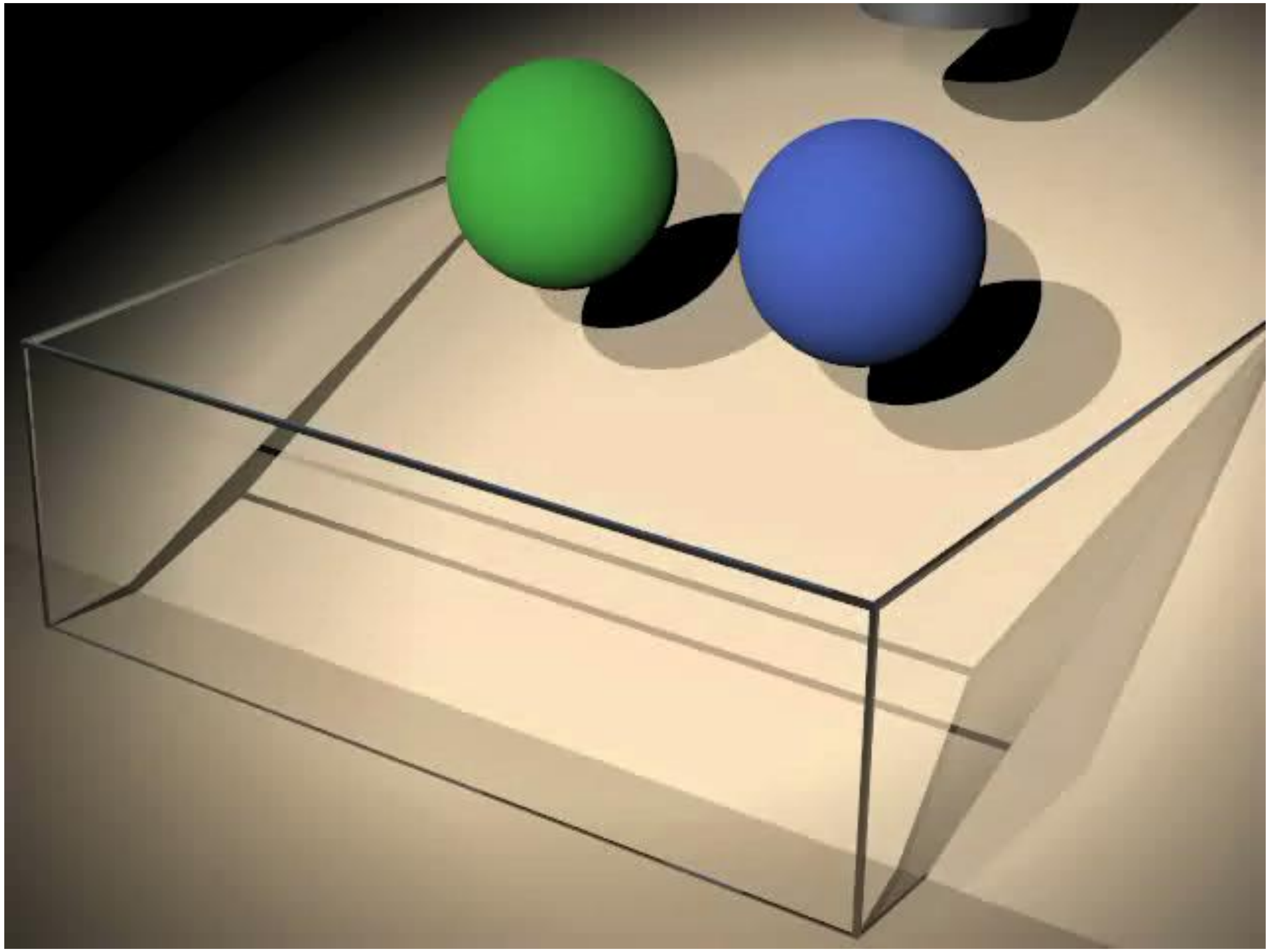
- **Viscosity**
  - **"Internal friction"**
  - **More accurately: diffusion of velocities**



- **Diffusion**
  - **Laplacian**
  - **Strength: dynamic viscosity coefficient**
  - **Compute with:** $V\mu\nabla \cdot \nabla\,\mathbf{u} = V\mu\triangle\mathbf{u}$



Viscous Material



Losasso et al., Multiple Interacting Liquids, Siggraph 06

Total Forces on ρ

(1) $$\mathbf{F} = m\mathbf{a}$$

Newton's 2nd law

(2) $$\mathbf{F} = m\frac{D\mathbf{u}}{Dt}$$

(3) $$m\mathbf{g} - V\nabla p + V\mu\triangle\mathbf{u} = m\frac{D\mathbf{u}}{Dt}$$

Forces so far.

(4) $$\rho\mathbf{g} - \nabla p + \mu\triangle\mathbf{u} = \rho\frac{D\mathbf{u}}{Dt}$$

Divide by V, density = m/V

(5) $$\mathbf{g} - \frac{1}{\rho}\nabla p + \frac{\mu}{\rho}\triangle\mathbf{u} = \frac{D\mathbf{u}}{Dt}$$

Divide by density

Total Forces on ρ (2)

- **So far:** $$\mathbf{g} - \frac{1}{\rho}\nabla p + \frac{\mu}{\rho}\triangle\mathbf{u} = \frac{D\mathbf{u}}{Dt}$$

- **Rearrange, use kinematic viscosity:**

$$\boxed{\frac{D\mathbf{u}}{Dt}} = -\frac{1}{\rho}\nabla p + \nu\triangle\mathbf{u} + \mathbf{g}$$

- **Let's look at the left hand side...**

Material Derivative

- **Change of a quantity *q(t,x)* during motion:**

$$
\begin{aligned}
\frac{d}{dt}\, q(t,\mathbf{x}) &= \frac{\partial q}{\partial t} + \frac{\partial q}{\partial \mathbf{x}} \cdot \frac{d\mathbf{x}}{dt} && \textbf{chain rule} \\
&= \boxed{\frac{\partial q}{\partial t} + \nabla q \cdot \mathbf{u}} && \textbf{use velocity} \\
&:= \frac{Dq}{Dt} && \textbf{material derivative}
\end{aligned}
$$

1 **how fast q is changing at fixed point in space x**
(temperature is decreasing because of cooling-down)

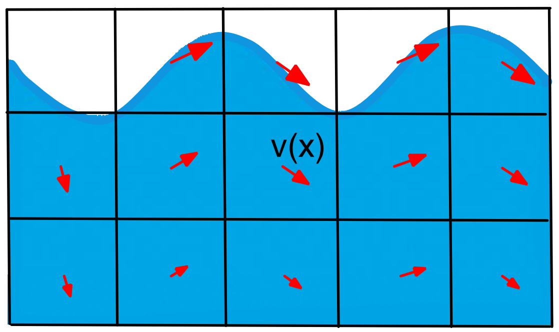
2 **inflow / outflow at x**
(temperature is changing because hot air is being replaced by cold air)

Viewpoints

how fast q is changing at fixed point in space x (temperature is decreasing because of cooling-down)	inflow / outflow at x (temperature is changing because hot air is being replaced by cold air)

- **Eulerian viewpoint:** 1 2

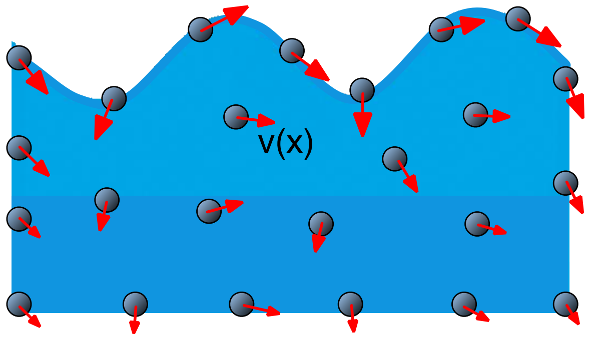
$$\frac{Dq}{Dt} = \frac{\partial q}{\partial t} + \nabla q \cdot \mathbf{u}$$



$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\frac{1}{\rho}\nabla p + \nu \triangle \mathbf{u} + \mathbf{g}$$

- **Lagrangian viewpoint:** 1

$$\frac{Dq}{Dt} = \frac{\partial q}{\partial t}$$



$$\frac{\partial \mathbf{u}}{\partial t} = -\frac{1}{\rho}\nabla p + \nu \triangle \mathbf{u} + \mathbf{g}$$

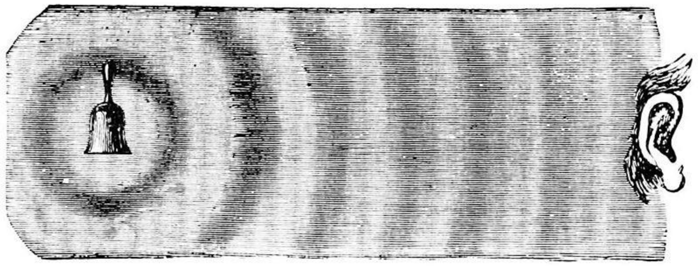
Viscosity

- Viscosity of water is close to zero
- Grids: Numerical inaccuracies introduce diffusion (manifest as viscosity)
  -> we can drop the viscosity term:

$$\underbrace{\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}}_{\text{advection}} = \underbrace{-\frac{1}{\rho}\nabla p}_{\text{pressure}} + \cancel{\underbrace{\nu \triangle \mathbf{u}}_{\text{viscosity}}} + \mathbf{g}$$

Note: **Particle simulations need viscosity to stabilize system!**

Incompressibility



- Water is not incompressible (sound waves)
- Very small volume change in water
- Small effect on how fluids move at macroscopic level
  → Irrelevant for animation
  → Water is *treated as incompressible*

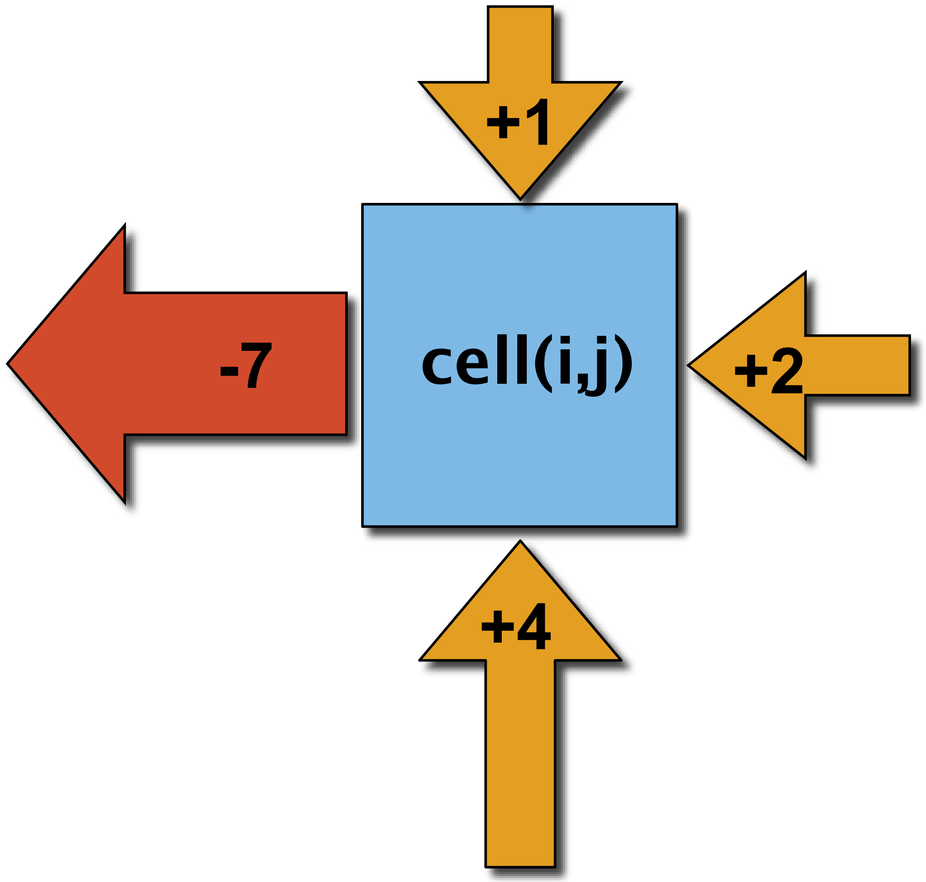
Incompressibility Condition

- Navier-Stokes Equations: Conservation of mass

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

- Divergence-free:
  - "what goes in somewhere, must go out somewhere else"
  - Holds for any volume in the fluid
  - ... also for a single cell

Per cell fluxes:



Divergence-Free

- Tricky part of simulating water: making sure that velocity field stays divergence-free
- Compute pressure to make the velocity field divergence free

$$\underbrace{\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}}_{\text{advection}} = \underbrace{\boxed{-\frac{1}{\rho}\nabla p}}_{\text{pressure}} + \underbrace{\nu \triangle \mathbf{u}}_{\text{viscosity}} + \mathbf{g}$$

Poisson Equation

- **We have:** $$\nabla \cdot \mathbf{u} = 0$$

$$\mathbf{u}'^{n} = f(\mathbf{u}^{n})$$ **// u' is the intermediate velocity after advection and applying viscosity and body forces**

$$\mathbf{u}^{n+1} = \mathbf{u}'^{n} - \Delta t \frac{1}{\rho} \nabla p$$

- **Thus:** $$\nabla \cdot \mathbf{u}^{n+1} = 0$$
- **Insert:** $$\nabla \cdot \mathbf{u}'^{n} - \Delta t \frac{1}{\rho} \nabla \cdot \nabla \;\; p = 0$$
- **Rearrange:** $$\nabla \cdot \nabla \;\; p = \frac{\rho}{\Delta t} \nabla \cdot \mathbf{u}'^{n}$$
- **...this is a *Poisson* equation, the right hand side is fully known!**

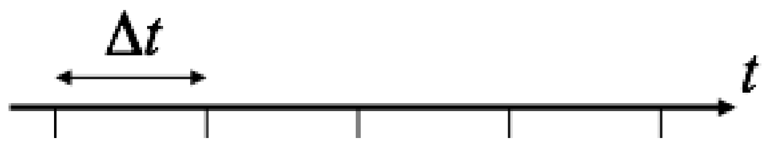
Outline

- Euler vs. Lagrange
- Navier-Stokes Equations
  - Conservation of momentum
  - Conservation of mass
- Numerical Solution
  - Discretization
  - Advection, external Forces, incompressibility
- Exercise

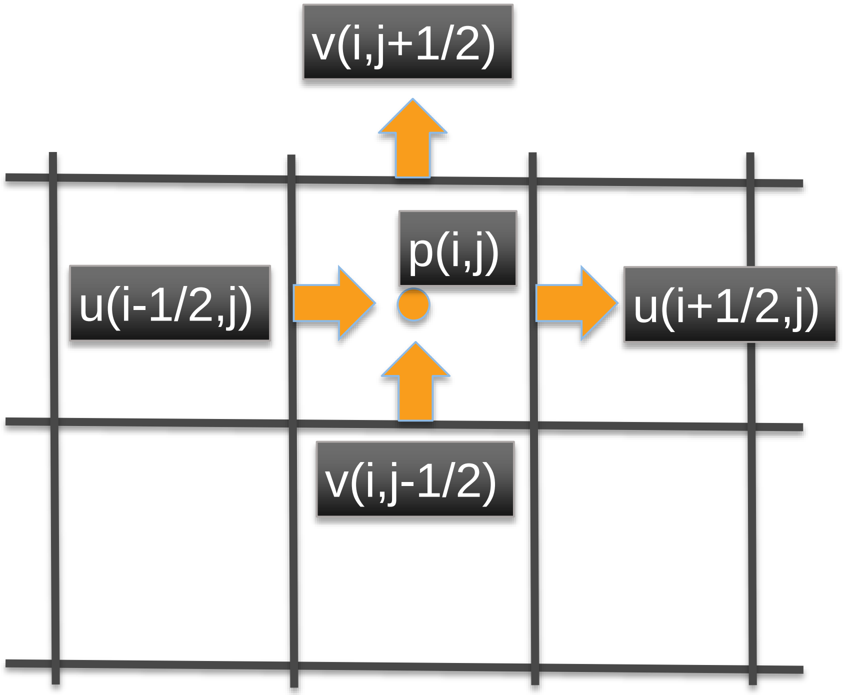
Discretization

- **Time discretization**

- **Fixed time step Δt**

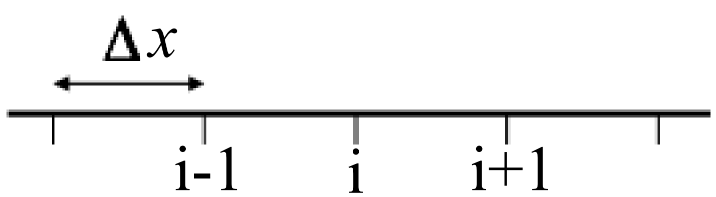


- **Space discretization**
  - **Staggered grid (MAC grid)**
  - **Cubical cells**
  - **Velocity components defined on faces of cells (staggering more stable)**
  - **Pressure defined at center**



Central Differences

- ***Central differences*** **to evaluate gradient and divergence**

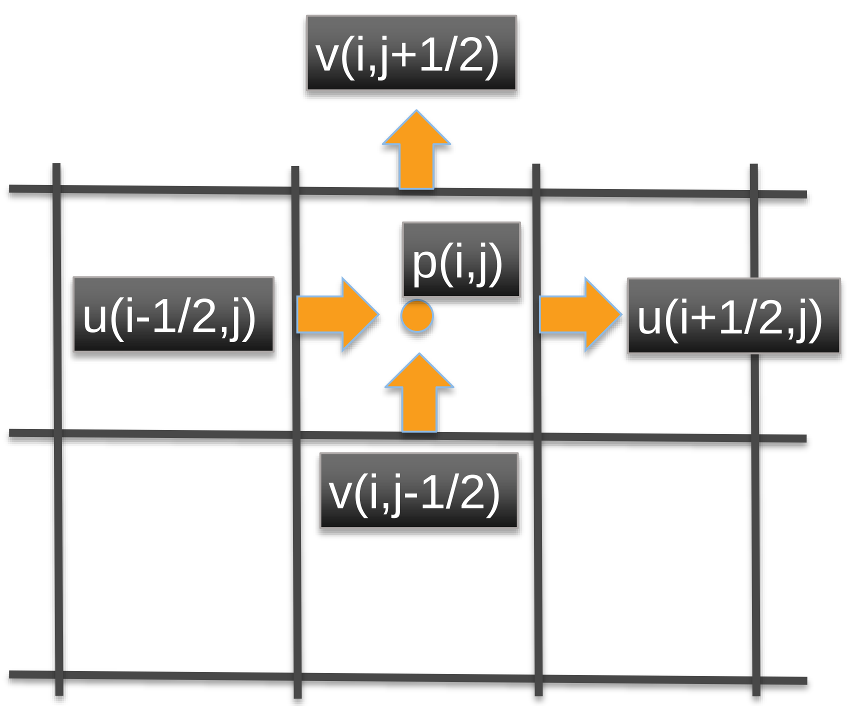


  - **First central difference -> unbiased, accurate O(Δx^2)** $$\frac{\partial q}{\partial x} = \frac{q_{i+1} - q_{i-1}}{2\Delta x}$$
  - q_i **can be different from** q_{i-1} **,** q_{i+1} **while central difference is still zero -> "null-space" problem: not only the constant function evaluates to zero**
  - **Forward or backward difference -> biased to the right/left, accurate O(Δx)** $$\frac{\partial q}{\partial x} = \frac{q_{i+1} - q_i}{\Delta x}$$

Staggered Form

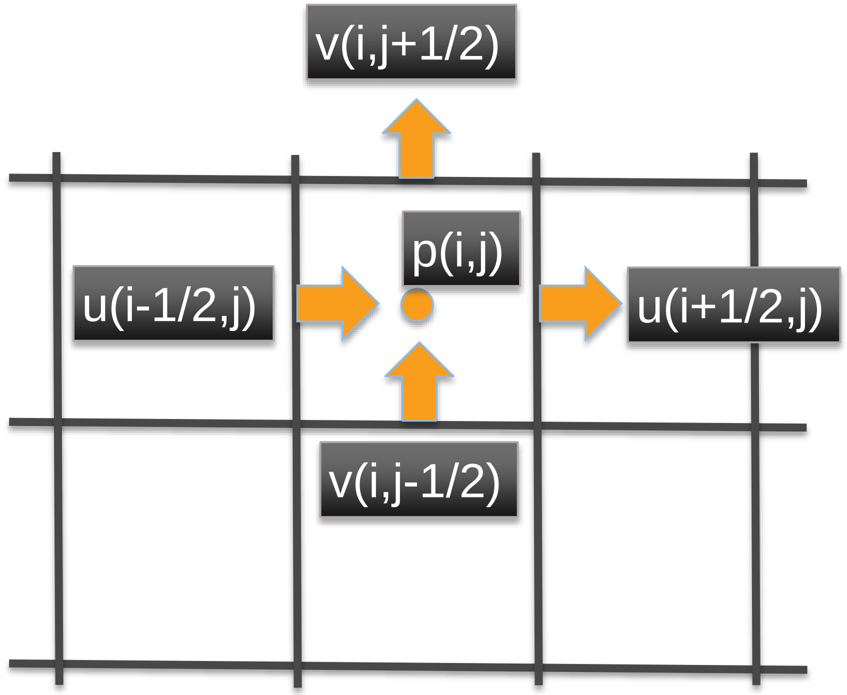
- **How can we get unbiased second-order accuracy of a central difference without null-space problem?**
- **Staggered: sample q at half-way points ->unbiased, second order, no null-space problem**

$$\frac{\partial q}{\partial x} = \frac{q_{i+1/2} - q_{i-1/2}}{\Delta x}$$



Staggered Form

- **Downside:**
- **Not having all quantities at the same spot makes some algorithms more difficult**
- **To evaluate velocity vector at a point we need interpolation**



Splitting

$$\underbrace{\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}}_{\text{advection}} = \underbrace{-\frac{1}{\rho}\nabla p}_{\text{pressure}} + \underbrace{\nu \triangle \mathbf{u}}_{\text{viscosity}} + \mathbf{g} \quad \& \quad \nabla \cdot \mathbf{u} = 0$$

- **Solve with operator splitting**
  - **If rate of change is a sum:** $$\frac{dq}{dt} = f_1() + f_2() + \ldots$$
  - **Add up components separately:** $$q' = q^n + \Delta t f_1()$$ $$q^{n+1} = q' + \Delta t f_2()$$
  - **Better: use specialized integration techniques for each**

Solution Overview

$$\underbrace{\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}}_{\text{advection}} = \underbrace{-\frac{1}{\rho}\nabla p}_{\text{pressure}} + \underbrace{\nu \triangle \mathbf{u}}_{\text{viscosity}} + \mathbf{g} \quad \& \quad \nabla \cdot \mathbf{u} = 0$$

- **Use conservation of momentum to update velocity field:**

1. **Advect velocities with themselves**
2. **Add external forces**

- **Ensure conservation of mass:**

3. **Choose pressure so that velocity field is divergence free**

The Navier-Stokes Equations

- **Conservation of momentum:**

$$\underbrace{\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}}_{\text{advection}} \;=\; \underbrace{-\frac{1}{\rho}\nabla p}_{\text{pressure}} + \underbrace{\nu \triangle \mathbf{u}}_{\text{viscosity}} + \mathbf{g}$$

1

- **Conservation of mass:**

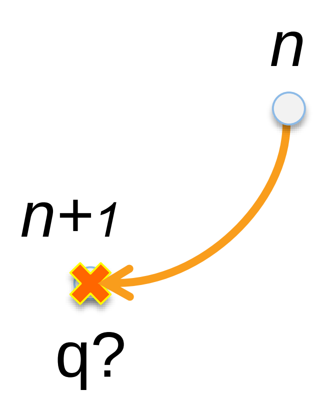
$$\nabla \cdot \mathbf{u} = 0$$

Advection

- **Solve:** $$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = 0$$ $$\frac{\partial q}{\partial t} + u\frac{\partial q}{\partial x} = 0$$
- **Move velocity field along itself**
- **Replacing derivatives with finite differences:**
  - ***Forward Euler*** **is unconditionally** ***unstable*** **-> no matter how small Δt, system always blows up**
  - **Even a more stable integration technique still gives us troubles**

Advection with Particles

- Use the so-called “Semi-Lagrangian” method
- Advection with particles trivial:
  - Particles move with the flow
  - New value of q at x is just old value of q of particle that ends up at x



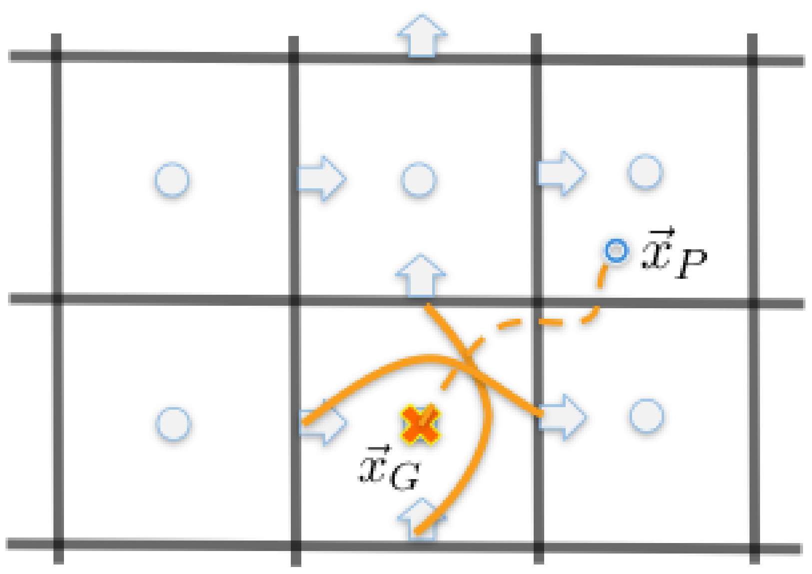
- Treat advection locally as a particle-based representation
- *Note*: additional variables such as smoke density, temperature are also advected

Semi-Lagrangian

- **Treat advection locally as a particle-based representation**

- **Determine velocity field at grid point**
- **Trace backward with velocity**

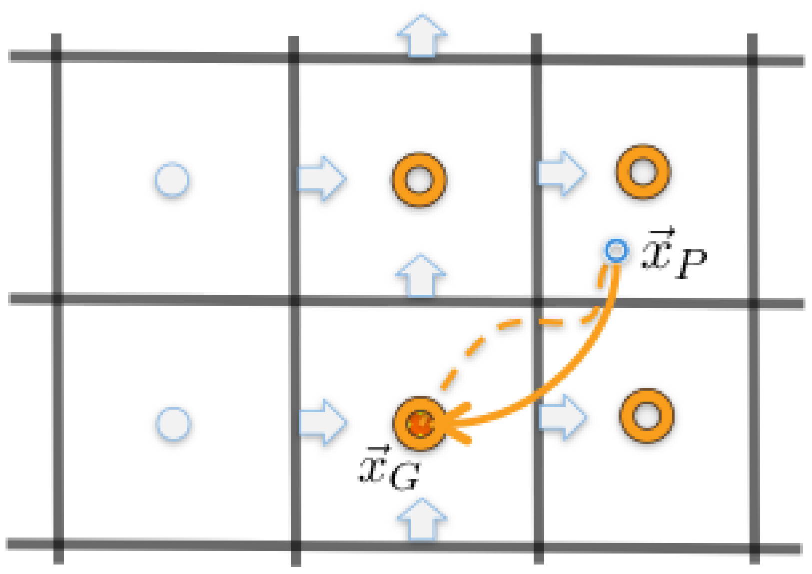
$$\vec{x}_P = \vec{x}_G - \Delta t\, \vec{u}_G$$



Semi-Lagrangian

- Treat advection locally as a particle-based representation

- Interpolate quantity q at source location
- Bilinear / trilinear interpolation
- Set new value of q at grid point



Properties of Semi-Lagrange Step

- In 1D: equal to finite differences with upwinding
- Upwinding: biasing a finite difference to the direction that flow is coming from
- Inaccurate (first order), but unconditionally stable
- Arbitrarily large time steps possible - but error and numerical dissipation will increase!

The Navier-Stokes Equations

- **Conservation of momentum:**

$$\underbrace{\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}}_{\text{advection}} = \underbrace{-\frac{1}{\rho}\nabla p}_{\text{pressure}} + \underbrace{\nu \triangle \mathbf{u}}_{\text{viscosity}} + \mathbf{g}$$

2

- **Conservation of mass:**

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

Add Forces

- **E.g., gravity - quite easy:**

  $$\mathbf{u}' = \mathbf{u} + \Delta t \mathbf{g}$$

- **More complicated settings: rigid body interactions, control methods...**

The Navier-Stokes Equations

- **Conservation of momentum:**

$$\underbrace{\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}}_{\text{advection}} = \underbrace{-\frac{1}{\rho}\nabla p}_{\text{pressure}} + \underbrace{\nu \triangle \mathbf{u}}_{\text{viscosity}} + \mathbf{g}$$

3

- **Conservation of mass:**

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

3

Make Divergence Free

- Advection & forces don't care whether velocity field ends up being divergence free
- Set pressures so that all cells satisfy $\nabla \cdot \mathbf{u} = 0$

3) Update velocities with pressure gradient

$$\mathbf{u}^{n+1} = \mathbf{u}^n - \Delta t \frac{1}{\rho} \nabla p$$

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

2) Solve for pressure

1) compute divergence

Discrete Pressure Gradient

3) Update velocities with pressure gradient

$$\mathbf{u}^{n+1} = \mathbf{u}^n - \Delta t \frac{1}{\rho} \nabla p$$

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

- **Compute pressure gradient with central differences for each velocity component**

$$u^{n+1}_{i+1/2,j} = u_{i+1/2,j} - \Delta t \frac{1}{\rho} \frac{p_{i+1,j} - p_{i,j}}{\Delta x}$$

$$v^{n+1}_{i,j+1/2} = v_{i,j+1/2} - \Delta t \frac{1}{\rho} \frac{p_{i,j+1} - p_{i,j}}{\Delta x}$$

- **Only applied to cells containing fluid**
- **Special treatment for boundary cells**

Discrete Divergence

- **Divergence in 2 dimensions:**

$$\mathbf{u}^{n+1} = \mathbf{u}^n - \Delta t \frac{1}{\rho} \nabla p$$

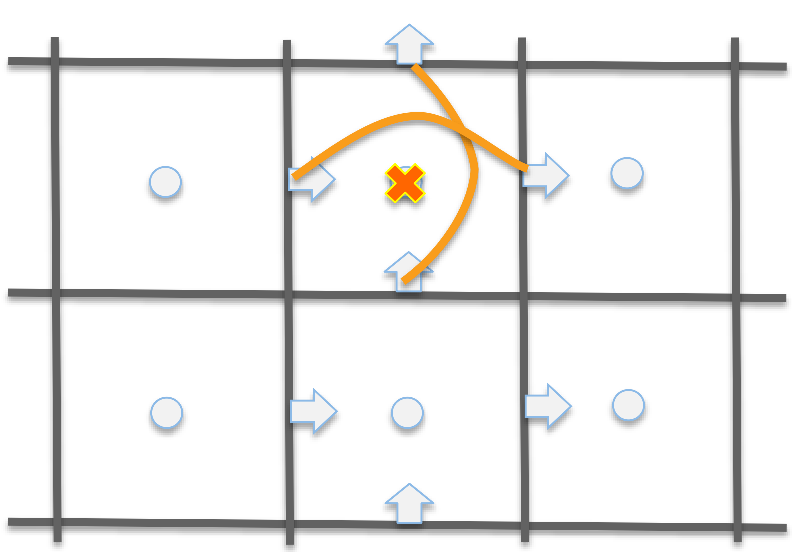
$$\nabla \cdot \nabla \quad p = \frac{\rho}{\Delta t} \nabla \cdot \mathbf{u}^n$$

$$\nabla \cdot \vec{u} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}$$

1) compute divergence

- **Central differences in 2D:**

$$(\nabla \cdot \vec{u})_{i,j} \approx \frac{u_{i+1/2,j} - u_{i-1/2,j}}{\Delta x} + \frac{v_{i,j+1/2} - v_{i,j-1/2}}{\Delta x}$$



Pressure Equation

- **For each cell we have a linear equation:**
  (note: we assume constant density)

$$\mathbf{u}^{n+1} = \mathbf{u}^n - \Delta t \frac{1}{\rho} \nabla p$$

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

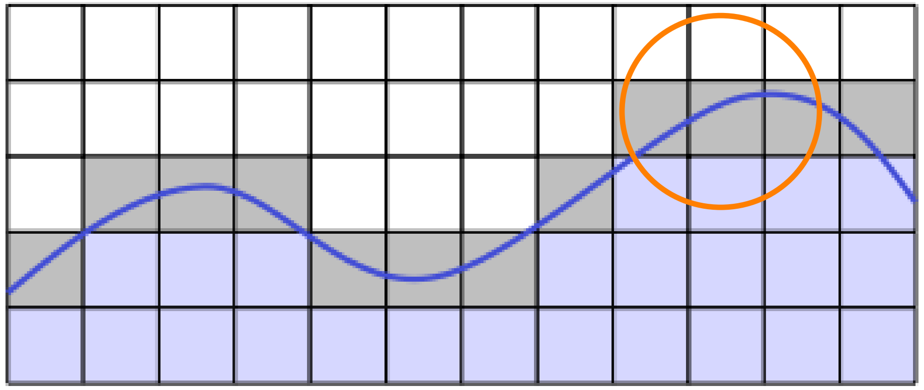
2) Solve for pressure

$$\frac{\Delta t}{\rho} \left(\frac{4p_{i,j} - p_{i+1,j} - p_{i,j+1} - p_{i-1,j} - p_{i,j-1}}{\Delta x^2} \right) = - \left(\frac{u_{i+1/2,j} - u_{i-1/2,j}}{\Delta x} + \frac{v_{i,j+1/2} - v_{i,j-1/2}}{\Delta x} \right)$$

- **Boundary cells involve pressure outside fluid**

Boundary Conditions

- Solid boundaries
  - Flow through wall should be zero
    -> pressure correction should also cause
    zero velocities $p_{wall} = p_{fluid}$



- Free surface boundaries
  - "ghost pressures" in air cells
    -> linear interpolation of p between fluid cell and air cell
    $p = 0$ at the actual fluid-air boundary

Matrix-Vector Form

$$\frac{\Delta t}{\rho}\left(\frac{4p_{i,j}-p_{i+1,j}-p_{i,j+1}-p_{i-1,j}-p_{i,j-1}}{\Delta x^2}\right)=-\left(\frac{u_{i+1/2,j}-u_{i-1/2,j}}{\Delta x}+\frac{v_{i,j+1/2}-v_{i,j-1/2}}{\Delta x}\right)$$

- **Put it into matrix-vector form** $Ap=d$
- **A is sparse and symmetric:**
  - **Each row corresponds to one cell (i,j)**
  - **Only 5 entries per row are non-zero (pij + 4 neighs)**
  - **Divide by** $\Delta t/(\rho\Delta x^2)$
  - **Diagonal entries non-zero, off-diagonal zero or -1 (neighbors)**

Solving the Pressure Equation

- **Easiest way:**
  - **Solve each equation in A separately**
  - **Iteratively solve for *p*. (*Note*: n denotes iterations here!)**

$$\tilde{p}_{i,j}^{n+1} = (\, d_{i,j} + \tilde{p}_{i+1,j}^{n} + \tilde{p}_{i-1,j}^{n} + \tilde{p}_{i,j+1}^{n} + \tilde{p}_{i,j-1}^{n})/4$$

- **Gauss-Seidel**
  - **Iterative method to solve a linear equation system**
  - **Similar to Jacobi method**
  - **Gauss-Seidel: uses updated values from *current* iteration step**
    Jacobi: uses only values from *last* iteration step

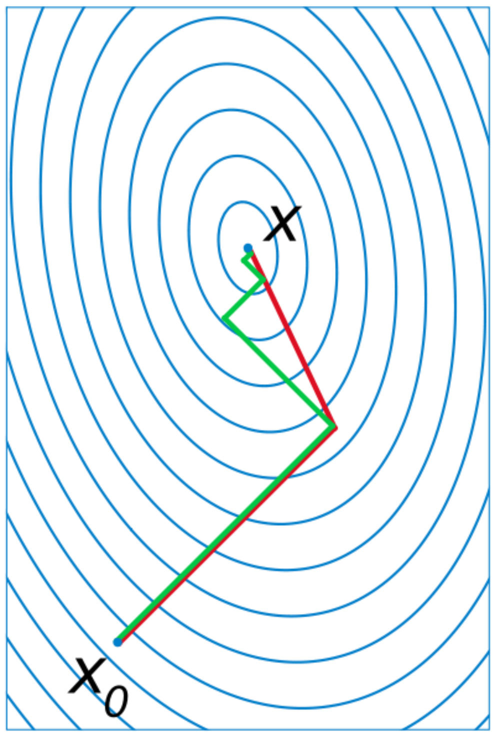
Convergence

- **Initial guess:**
  - **Rest state: pressure from last time step**
  - **Motion: Pressure can change significantly, use zero**

- **When to stop iterations?**
  - **Check residual:** $r = d - A\tilde{p}$
  - **Exact solution:** $r=0$
  - **Stop when** r **less than** *1e-5*

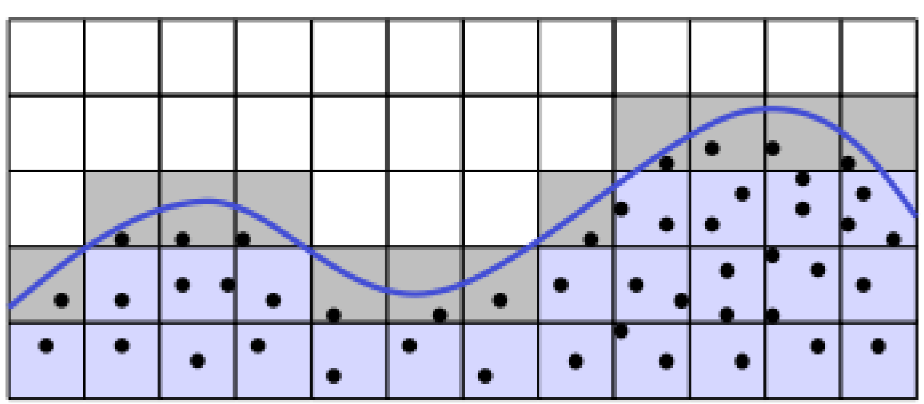
Conjugate Gradient

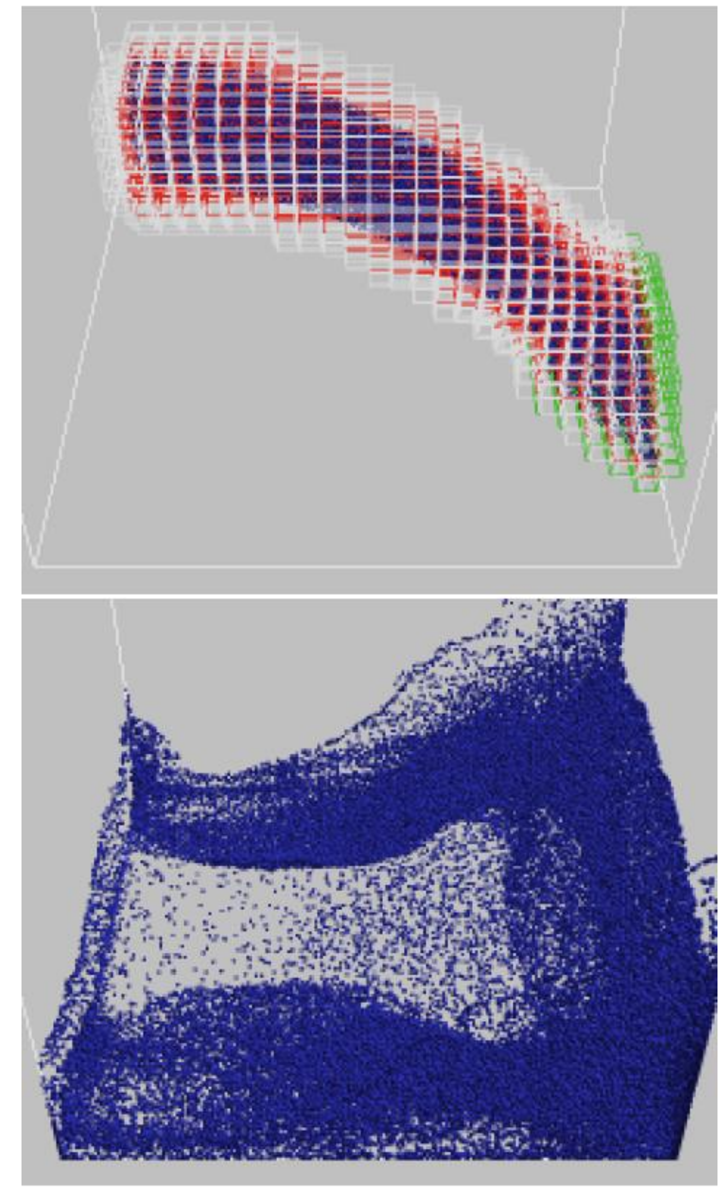
- Conjugate Gradient (CG)
  - Symmetric positive definite linear systems
  - Start with a guess
  - In each iteration improve it
  - Stop when error small enough
- The larger the grid, the longer it takes to converge
  -> Preconditioning (PCG) speed this up
  -> (Modified) Incomplete Cholesky



Marker particles

- Use mass-less marker particles
- Move markers using interpolated fluid velocity
- Mark cells containing any marker as fluid, rest as empty or solid
- Wrap surface around markers





Algorithm Summary

1. Update grid (mark cells)
2. Advance the velocity field
   - Advection (semi-Lagrange step for each velocity component)
   - Apply external forces (gravity, ...)
   - Calculate and apply pressure (divergence-free)
3. Move marker particles
4. Go to 1.

Further Reading

- Bridson et al., Fluid Course Notes, Siggraph 2006/2007 http://www.cs.ubc.ca/~rbridson/fluidsimulation/
- Cline et al., Fluid Flow for the Rest of Us: Tutorial of the Marker and Cell Method in Computer Graphics
- (Paper: Stam 1999: Stable Fluids)

Acknowledgements

Based on material by B. Thomaszewski, B. Solenthaler, M. Gross

ETH Zurich

Disney Research