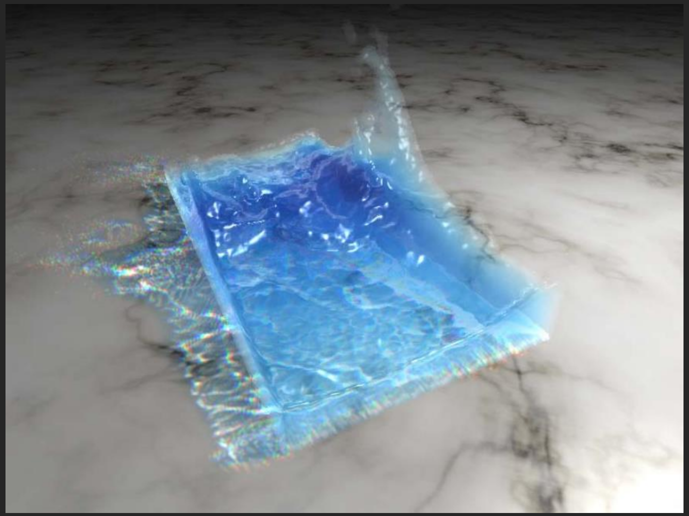
Fluid simulation

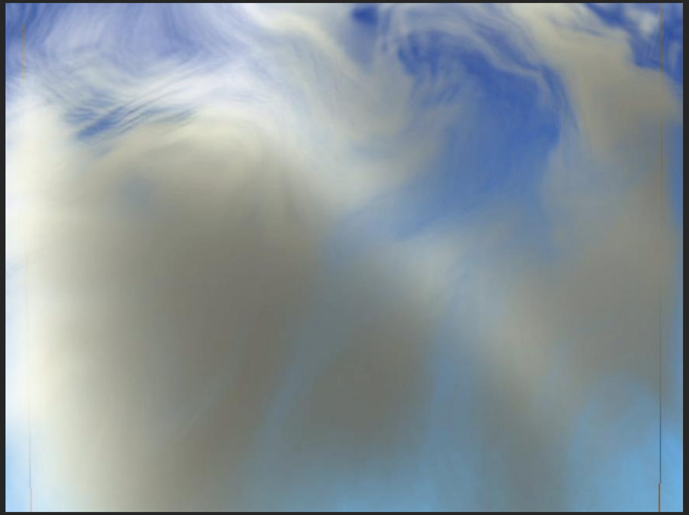
Marek Trtík

PA199

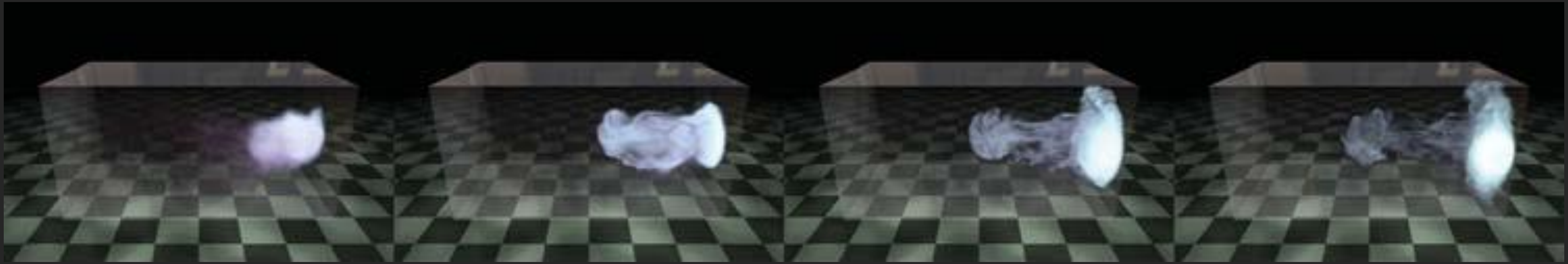
Motivation

Picture source: [2]





Picture source: [3]



Other pictures source: [5]

Outline

- **Euler approach**:
  - Fluid is modelled by a vector field, representing the velocity of the fluid.
- **Lagrange approach**:
  - Fluid is modelled by set of particles.
- **Smoothed Particle Hydrodynamics**:
  - Fluid is modelled by set of particles moved via a velocity vector field.
- **Hight-field surface approximation**:
  - Suitable for simulation of only fluid’s surface, e.g., lake or ocean surface.

Euler approach

Fluid Model

- Assumptions:
  - **Incompressible** fluid:
    - Volume of any subregion of the fluid is constant over time.
    - Represented by an **incompressible constraint**.
  - **Homogeneous** fluid:
    - The **density** of fluid is the same and **constant** in every region of the fluid and over time.

- **Navier-Stokes equations** model a fluid:
  - Fluid **velocity** (motion) represented by a **vector field** $\boldsymbol{u}(\boldsymbol{x}, t)$.
  - Fluid **pressure** represented by a **scalar field** $p(\boldsymbol{x}, t)$.
  - **Partial differential equations** define **changes** in the **vector field** $\boldsymbol{u}$ over time.

Navier-Stokes Equations

- The **momentum equations** (for each coordinate one):

$$\frac{\partial \boldsymbol{u}}{\partial t} = \underbrace{-(\boldsymbol{u} \cdot \nabla)\boldsymbol{u}}_{\text{advection}} \underbrace{- \frac{1}{\rho}\nabla p}_{\text{pressure}} + \underbrace{\nu \nabla^2 \boldsymbol{u}}_{\text{diffusion}} + \underbrace{\boldsymbol{g}}_{\text{external accel.}}$$

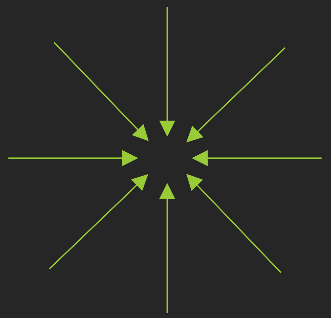
- The **incompressibility constraint**:

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

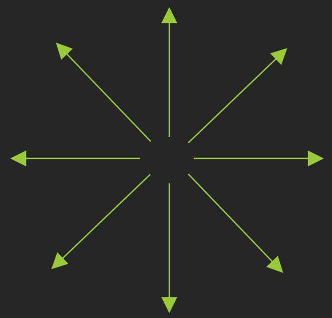
- Where (let $\boldsymbol{x} = (x, y, z)^{\mathsf{T}}$ be a position in space and t be simulation time):
  - $\boldsymbol{u}(\boldsymbol{x}, t) = \left(u(\boldsymbol{x}, t), v(\boldsymbol{x}, t), w(\boldsymbol{x}, t)\right)^{\mathsf{T}}$ is the **velocity vector field** of the fluid. (computed)
  - $p(\boldsymbol{x}, t)$ is a **pressure scalar field** of the fluid; used to **preserve incompressibility**. (computed)
  - ρ is the **density** of the fluid, e.g., water $\rho = 10^3 \frac{kg}{m^3}$.
  - ν is the **viscosity** (resistance to deformation) of the fluid, e.g., honey – high viscosity, water – low viscosity.
  - $\boldsymbol{g}(\boldsymbol{x}, t)$ is the **acceleration vector field** of forces acting on the fluid, e.g., gravity $\boldsymbol{g}(\boldsymbol{x}, t) = (0, 0, -10)^{\mathsf{T}} \frac{m}{s^2}$.
  - $\cdot$ is the dot product.

Gradient, Divergence and Laplacian

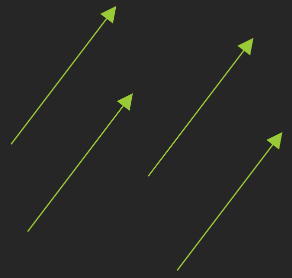
- Operator of spatial partial derivatives: $\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}\right)^{\mathrm{T}}$.
  - Identifies a direction of a maximum increase of a function at a given time.
  - Example: $\nabla p = \left(\frac{\partial p}{\partial x}, \frac{\partial p}{\partial y}, \frac{\partial p}{\partial z}\right)^{\mathrm{T}}$.
- Divergence operator: $\nabla \cdot \boldsymbol{u} = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}\right)^{\mathrm{T}} \cdot (u, v, w)^{\mathrm{T}} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}$.
  - Can only be applied to a vector field.



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



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



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

Gradient, Divergence and Laplacian

- Directional derivative: $\boldsymbol{u} \cdot \nabla = (u, v, w)^\mathsf{T} \cdot \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}\right)^\mathsf{T} = u\frac{\partial}{\partial x} + v\frac{\partial}{\partial y} + w\frac{\partial}{\partial z}.$
  - Therefore, $(\boldsymbol{u} \cdot \nabla)\boldsymbol{u} = \left(u\frac{\partial}{\partial x} + v\frac{\partial}{\partial y} + w\frac{\partial}{\partial z}\right)(u, v, w)^\mathsf{T} = \begin{pmatrix} u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} \\ u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} \\ u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z} \end{pmatrix}.$
- Laplacian operator: $\nabla^2 = \nabla \cdot \nabla = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}.$
  - Example: $\nabla^2 \boldsymbol{u} = \nabla \cdot \nabla\, \boldsymbol{u} = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}\right)(u, v, w)^\mathsf{T} = \begin{pmatrix} \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \\ \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2} \\ \frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2} \end{pmatrix}.$

Adding Custom Quantities

- Beside the fluid we also simulate other quantities, e.g., smoke density, temperature.
- Represent any such quantity q as another scalar/vector field.
- Add a related equation, how q changes in time:

$$\frac{\partial q}{\partial t} = -(\boldsymbol{u} \cdot \nabla) q + \nu \nabla^2 q + S$$

- Observe the similarity with the momentum equation:
  - Advection: $-(\boldsymbol{u} \cdot \nabla) q$
  - Diffusion: $\nu \nabla^2 q$
  - We do not have pressure term.
  - S can be used to simulate constant inflow of q into the fluid.

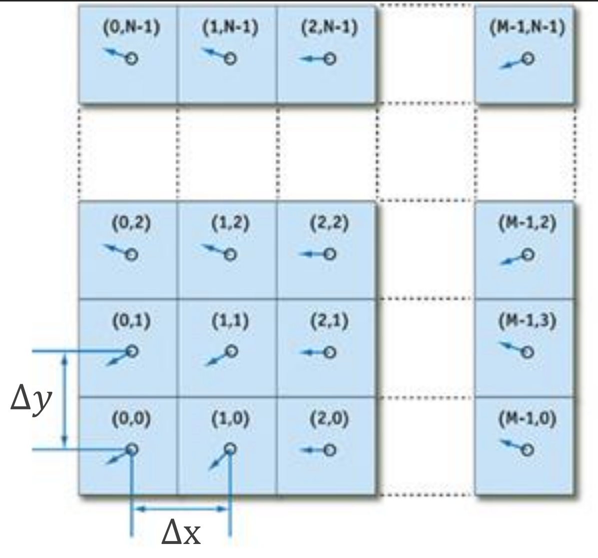
=> Methods for solving the momentum equation can be also applied for q equation.

Boundary Conditions

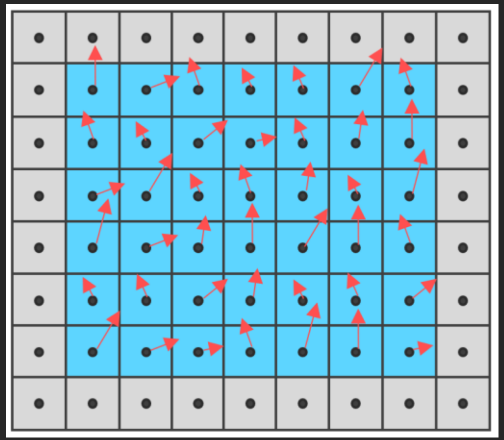
- The fluid can collide with:
  - Static solid objects, like walls.
  - Freely moveable solid objects, like piece of wood in water.
  - Another fluid, like oil stain on water surface. (not covered in this lecture)
- Our goal is to prevent the fluid to flow into the solid objects.
- Let $\boldsymbol{n}(\boldsymbol{x},t), \boldsymbol{u}_s(\boldsymbol{x},t)$ be the normal and velocity of the solid surface.
- The boundary constraint for:
  - **Low viscosity** fluid: $\boldsymbol{u}(\boldsymbol{x},t) \cdot \boldsymbol{n}(\boldsymbol{x},t) = \boldsymbol{u}_s(\boldsymbol{x},t) \cdot \boldsymbol{n}(\boldsymbol{x},t)$
  - **High viscosity** fluid: $\boldsymbol{u}(\boldsymbol{x},t) = \boldsymbol{u}_s(\boldsymbol{x},t)$
- We can use boundary condition to model **fluid source and/or sink**.

Discretize fields

- **Discretize** the space into a **regular grid**.
  For each cell i, j, k we store:
  - Fluid velocity: $\boldsymbol{u}_{i,j,k}$
  - Pressure: $p_{i,j,k}$
  - Any other field: $q_{i,j,k}$



2D grid



- Discretize **boundary conditions**:
  - Mark cells filled by solid objects, e.g., walls.

Discretize derivatives

- Use **finite differences** to approximate partial derivatives.
- Examples:

$$\nabla p_{i,j,k} = \left(\frac{p_{i+1,j,k} - p_{i-1,j,k}}{2\Delta x}, \frac{p_{i,j+1,k} - p_{i,j-1,k}}{2\Delta y}, \frac{p_{i,j,k+1} - p_{i,j,k-1}}{2\Delta z} \right)^{\mathsf{T}}$$

$$\nabla \cdot \boldsymbol{u}_{i,j,k} = \frac{\boldsymbol{u}_{i+1,j,k} - \boldsymbol{u}_{i-1,j,k}}{2\Delta x} + \frac{\boldsymbol{u}_{i,j+1,k} - \boldsymbol{u}_{i,j-1,k}}{2\Delta y} + \frac{\boldsymbol{u}_{i,j,k+1} - \boldsymbol{u}_{i,j,k-1}}{2\Delta z}$$

$$\nabla^2 q_{i,j,k} = \frac{q_{i+1,j,k} - 2q_{i,j,k} + q_{i-1,j,k}}{\Delta x^2} + \frac{q_{i,j+1,k} - 2q_{i,j,k} + q_{i,j-1,k}}{\Delta y^2} + \frac{q_{i,j,k+1} - 2q_{i,j,k} + q_{i,j,k-1}}{\Delta z^2}$$

Solving Equations

- Method of splitting:
  - Solve a complex equation by a sequence numerical integrations.

$$\frac{dq}{dt} = f(q) + g(q) \quad \rightarrow \quad \begin{aligned} \hat{q} &= q^t + \Delta t f(q^t) \\ q^{t+\Delta t} &= \hat{q} + \Delta t g(\hat{q}) \end{aligned}$$

  - The result is equivalent to a single integration:

$$\begin{aligned} q^{t+\Delta t} &= \hat{q} + \Delta t g(\hat{q}) \\ &= q^t + \Delta t f(q^t) + \Delta t g(q^t + \Delta t f(q^t)) \\ &= q^t + \Delta t f(q^t) + \Delta t (g(q^t) + \mathcal{O}(\Delta t)) \\ &= q^t + \Delta t (f(q^t) + g(q^t)) + \mathcal{O}(\Delta t^2) \\ &= q^t + \Delta t \frac{dq}{dt} + \mathcal{O}(\Delta t^2) \end{aligned}$$

Solving Equations

- We solve the momentum equation using the splitting method:

$$\frac{\partial \boldsymbol{u}}{\partial t} = -(\boldsymbol{u} \cdot \nabla)\boldsymbol{u} - \frac{1}{\rho}\nabla p + \nu \nabla^2 \boldsymbol{u} + \boldsymbol{g}$$

  - Start in the current state:

    $$\boldsymbol{w}_0(\boldsymbol{x}) = \boldsymbol{u}(\boldsymbol{x}, t)$$

  - Apply external accelerations $\boldsymbol{g}$:

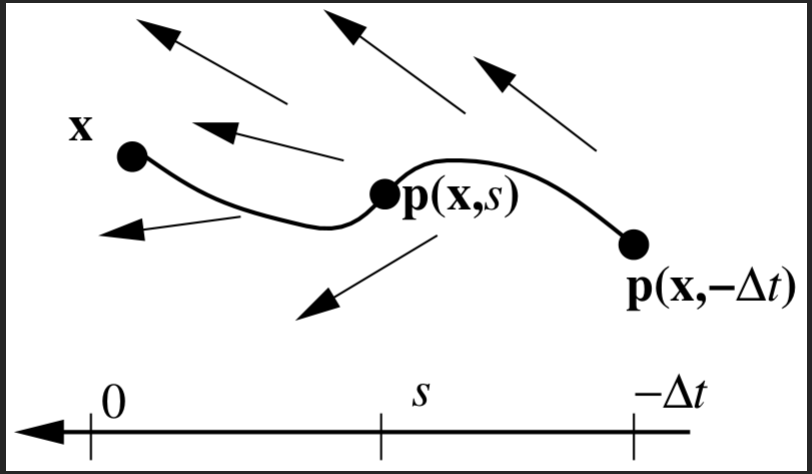
    $$\boldsymbol{w}_1(\boldsymbol{x}) = \boldsymbol{w}_0(\boldsymbol{x}) + \Delta t \boldsymbol{g}$$

    (forward Euler)

  - Apply fluid advection $-(\boldsymbol{u} \cdot \nabla)\boldsymbol{u}$:

    $$\boldsymbol{w}_2(\boldsymbol{x}) = \boldsymbol{w}_1(\mathbf{p}(\boldsymbol{x}, -\Delta t))$$

    (method of characteristics)



Picture source: [3]

The new velocity at $\boldsymbol{x}$ is the velocity that the particle had a time Δt ago at the location $\mathbf{p}(\boldsymbol{x}, -\Delta t)$ (going backward in time along $\mathbf{p}$).

Solving Equations

- We solve the momentum equation using the splitting method:

$$\frac{\partial \boldsymbol{u}}{\partial t} = -(\boldsymbol{u} \cdot \nabla)\boldsymbol{u} - \frac{1}{\rho}\nabla p + \nu\nabla^2\boldsymbol{u} + \boldsymbol{g}$$

  - Apply fluid viscosity $\nu\nabla^2\boldsymbol{u}$:

$$\boldsymbol{w}_3(\boldsymbol{x}) = \boldsymbol{w}_2(\boldsymbol{x}) + \Delta t\nu\nabla^2\boldsymbol{w}_3(\boldsymbol{x})$$

(backward Euler)

- Lastly, we must **compute** the pressure $-\frac{1}{\rho}\nabla p$ acceleration s.t. we remove divergence from $\boldsymbol{w}_3$, i.e., to satisfy the incompressibility:

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

Solving Equations

- **Helmholtz-Hodge Decomposition**: Any vector field $\boldsymbol{w}$ can be uniquely decomposed to a vector field $\boldsymbol{u}$ and a scalar field p satisfying:

$$\boldsymbol{w} = \boldsymbol{u} + \nabla p$$

  where $\boldsymbol{u}$ is a divergence free, i.e., $\nabla \cdot \boldsymbol{u} = 0$.

- When we apply divergence operator to both sides of the equation:

$$\nabla \cdot \boldsymbol{w} = \nabla^2 p$$

  we get a Poisson equation.

- Due to discretization, we get a sparse system of linear equations

  => Use, for example, Jacobi method.

- We use the computed pressure field to get the resulting fluid velocity:

$$\boldsymbol{u}(\boldsymbol{x}, t + \Delta\mathrm{t}) = \boldsymbol{w}_3(\boldsymbol{x}) - \nabla p$$

Euler approach

DEMO!

https://paveldogreat.github.io/WebGL-Fluid-Simulation/
http://haxiomic.github.io/projects/webgl-fluid-and-particles/

Lagrange approach

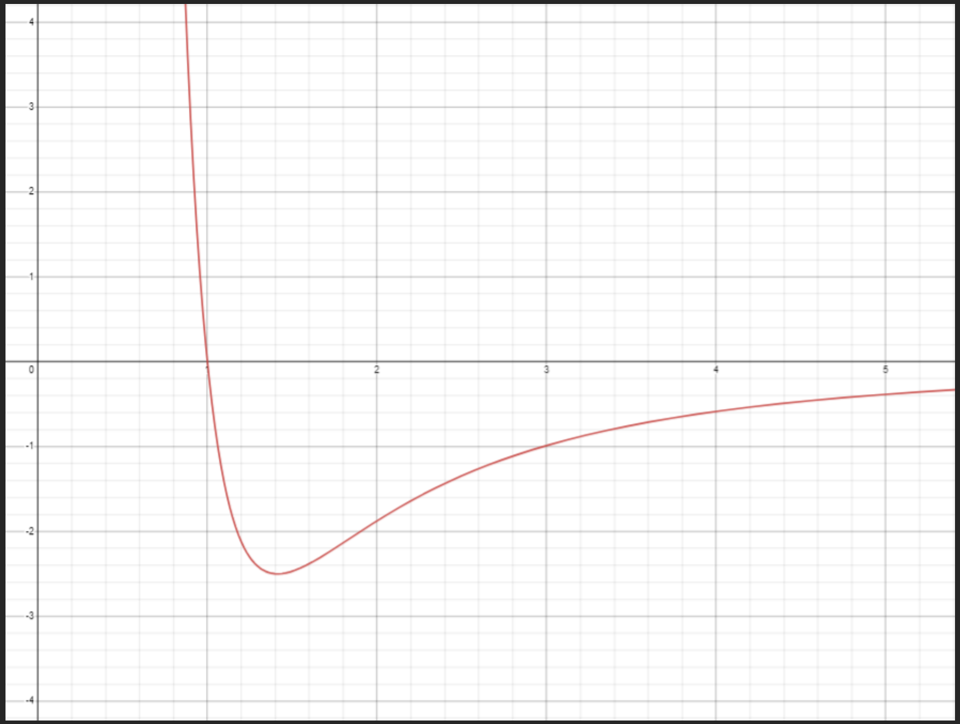
Particles Simulation

- The fluid is represented by n particles $\{\mathcal{P}_0, \dots, \mathcal{P}_{n-1}\}$.
- Each particle $\mathcal{P}_i$ is defined by:
  - Mass: m_i
  - Position vector: $\boldsymbol{p}_i$
  - Velocity vector: $\boldsymbol{u}_i$
  - Total external force: $\boldsymbol{f}_i$
- **Newton's equations of motion** for moving particles:
  - $\frac{d\boldsymbol{p}_i}{dt} = \boldsymbol{u}_i$ (3 equations in 3D space)
  - $\frac{d\boldsymbol{u}_i}{dt} = \frac{\boldsymbol{f}_i}{m_i}$ (3 equations in 3D space)

External Forces

- The attribute $\boldsymbol{f}_i$ of a particle $\mathcal{P}_i$ is a **sum** of all forces acting on the particle.
- We usually want Earth's **gravity** to act on particles:
  - Force of a homogenous field: $m_i \boldsymbol{g}$
  - Typically: $\boldsymbol{g} = (0,0,-10)^\mathsf{T}$

- Interaction between particles $\mathcal{P}_i$ and $\mathcal{P}_j$ via **Lennard-Jones** force:
  - Let $d_{i,j} = |\boldsymbol{p}_i - \boldsymbol{p}_j|$ and $\boldsymbol{d}_{i,j} = \frac{\boldsymbol{p}_i - \boldsymbol{p}_j}{d_{i,j}}$.
  - $\boldsymbol{F}_{i,j} = \left(\frac{k_1}{d_{i,j}^m} - \frac{k_2}{d_{i,j}^n} \right) \boldsymbol{d}_{i,j}, \qquad \boldsymbol{F}_{j,i} = -\boldsymbol{F}_{i,j}$
  - where typically $k_1 = k_2$, $m = 4$ and $n = 2$.



$|\boldsymbol{F}_{i,j}|$ for $k_1 = k_2 = 10, m = 4, n = 2$.

Plotted at https://www.desmos.com/calculator

Lagrange approach

DEMO!

Smoothed Particle Hydrodynamics

Smoothed Particle Hydrodynamics

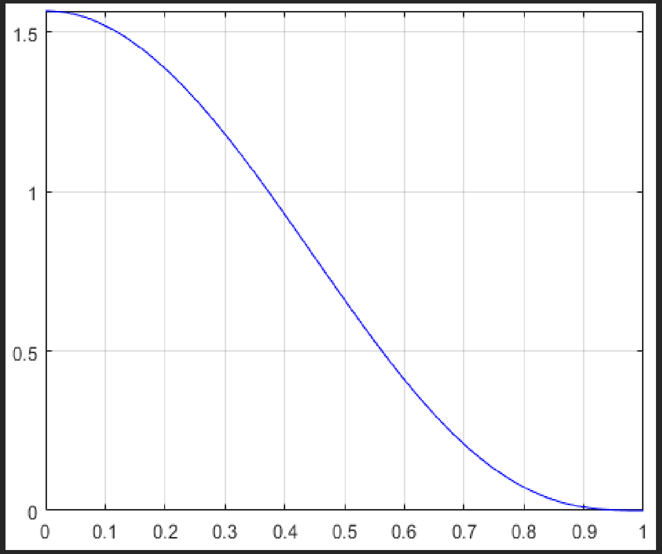
- Simulate fluid using a set of n particles, i.e., **Lagrange approach.**
- Compute forces acting on the particles by **Euler approach.** How?
- Smooth properties of particles into continuous fields.
  - Use a **smoothing kernel** $W(x)$, e.g., **poly6**:

$$W(x) = \frac{315}{64\pi d^9}\begin{cases}(d^2 - x^2)^3 & \text{if } 0 \le x \le d \\ 0 & \text{otherwise}\end{cases}$$

  - Let A be a property of particle. Then continuous field $A(\boldsymbol{x})$ is:

$$A(\boldsymbol{x}) = \sum_{j=0}^{n-1} m_j \frac{A_j}{\rho_j} W(|\boldsymbol{x}_j - \boldsymbol{x}|).$$

    - Example: $\rho(\boldsymbol{x}) = \sum_{j=0}^{n-1} m_j \frac{\rho_j}{\rho_j} W(|\boldsymbol{x}_j - \boldsymbol{x}|) = \sum_{j=0}^{n-1} m_j W(|\boldsymbol{x}_j - \boldsymbol{x}|)$.



$W(x)$ for $d = 1$.

Smoothed Particle Hydrodynamics

- With the fields defined we can use **momentum** and **incompressibility** equations:

$$\frac{\partial \boldsymbol{u}}{\partial t} = -(\boldsymbol{u} \cdot \nabla)\boldsymbol{u} - \frac{1}{\rho}\nabla p + \nu \nabla^2 \boldsymbol{u} + \boldsymbol{g}, \qquad \nabla \cdot \boldsymbol{u} = 0.$$

- We simulate particles => mass is conserved => $\nabla \cdot \boldsymbol{u} = 0$ is **not** needed.
- Particles automatically move with the fluid => $-(\boldsymbol{u} \cdot \nabla)\boldsymbol{u}$ is **not** needed.
- So, we only solve: $\frac{\partial \boldsymbol{u}}{\partial t} = -\frac{1}{\rho}\nabla p + \nu \nabla^2 \boldsymbol{u} + \boldsymbol{g}$.
- Recall second Newton's equation of motion: $\frac{d\boldsymbol{u}_i}{dt} = \frac{\boldsymbol{f}_i}{m_i}$.
- Therefore, $\frac{\boldsymbol{f}_i}{m_i} = -\frac{1}{\rho(\boldsymbol{x}_i)}\nabla p(\boldsymbol{x}_i) + \nu \nabla^2 \boldsymbol{u}(\boldsymbol{x}_i) + \boldsymbol{g}$.

Smoothed Particle Hydrodynamics

- The pressure field p can be obtained from density field ρ by law of ideal gas:
  - $p(\boldsymbol{x}) = k(\rho(\boldsymbol{x}) - \rho_0)$, where k is a gas constant and ρ_0 is the environment pressure.
- Derivatives of any field $A(\boldsymbol{x})$:

$$\nabla A(\boldsymbol{x}) = \sum_{j=0}^{n-1} m_j \frac{A_j}{\rho_j} \nabla W(|\boldsymbol{x}_j - \boldsymbol{x}|), \qquad \nabla^2 A(\boldsymbol{x}) = \sum_{j=0}^{n-1} m_j \frac{A_j}{\rho_j} \nabla^2 W(|\boldsymbol{x}_j - \boldsymbol{x}|)$$

where $\nabla W(|\boldsymbol{x}_j - \boldsymbol{x}|) = W'(|\boldsymbol{x}_j - \boldsymbol{x}|)\frac{\boldsymbol{x}_j - \boldsymbol{x}}{|\boldsymbol{x}_j - \boldsymbol{x}|}$, $\nabla^2 W(|\boldsymbol{x}_j - \boldsymbol{x}|) = W''(|\boldsymbol{x}_j - \boldsymbol{x}|) + \frac{2W'(|\boldsymbol{x}_j - \boldsymbol{x}|)}{|\boldsymbol{x}_j - \boldsymbol{x}|}$.

- Forces between two particles generated by fields $\nabla p, \nabla^2 \boldsymbol{u}$ should be **symmetric** => we usually modify their computation:

$$\nabla p(\boldsymbol{x}_i) = \sum_{j=0}^{n-1} m_j \frac{p_i + p_j}{2\rho_j} \nabla W(|\boldsymbol{x}_j - \boldsymbol{x}_i|), \qquad \nabla^2 \boldsymbol{u}(\boldsymbol{x}_i) = \sum_{j=0}^{n-1} m_j \frac{\boldsymbol{u}_j - \boldsymbol{u}_i}{\rho_j} \nabla^2 W(|\boldsymbol{x}_j - \boldsymbol{x}_i|).$$

Height-field surface approximation

Fluid Surface Model

- We model a fluid surface by a function $h(x, y, t)$.
  - At a point (x, y) in the XY plane and in time t the function defines fluid height $\mathrm{z} = h(x, y, t)$.
- Change of h in time is given by:

$$\frac{\partial^2 h}{\partial t^2} = v^2 \nabla^2 h$$

where v is the speed of waves in the fluid.

- How to solve the equation?
  - Introduce an auxiliary function $q = \frac{\partial h}{\partial t}$.
  - Rewrite the equation into this system:

$$\frac{\partial q}{\partial t} = v^2 \nabla^2 h, \qquad \frac{\partial h}{\partial t} = q.$$

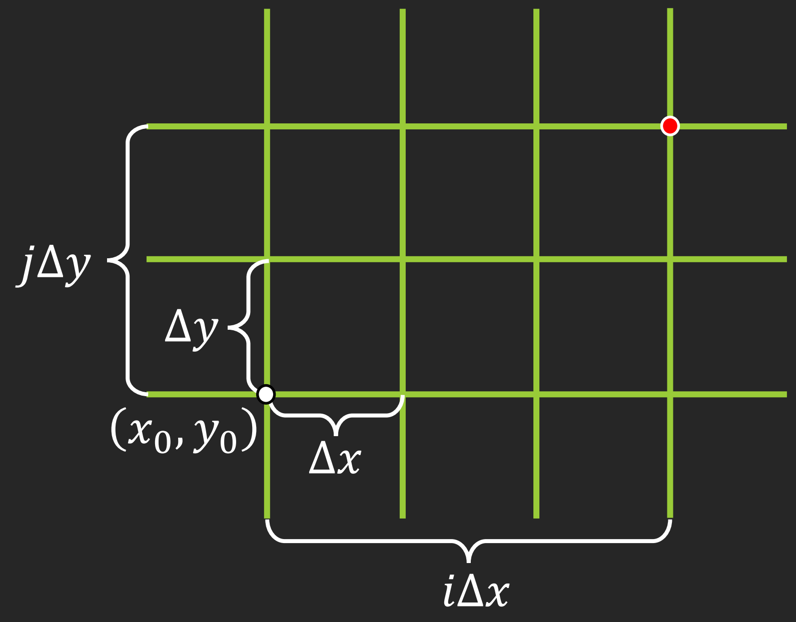
  - Discretize (next slide).

Discretize Model

- We discretize the functions h, q by 2D arrays:

$$h(x_0 + i\Delta x, y_0 + j\Delta y, t_0 + k\Delta t) \Rightarrow h_{i,j}^k$$
$$q(x_0 + i\Delta x, y_0 + j\Delta y, t_0 + k\Delta t) \Rightarrow q_{i,j}^k$$

  where
  - i, j are indices to the arrays.
  - $\Delta x, \Delta y$ are distances between grid cells in X,Y axes.
  - k simulation step number.
  - Δt simulation time step.
  - NOTE: Usually, $x_0 = y_0 = t_0 = 0$.



- We solve the discretized system numerically, e.g., using forward Euler method:

$$q_{i,j}^{k+1} = q_{i,j}^k + \Delta t v^2 \left(\frac{h_{i+1,j}^k - 2h_{i,j}^k + h_{i-1,j}^k}{\Delta x^2} + \frac{h_{i,j+1}^k - 2h_{i,j}^k + h_{i,j-1}^k}{\Delta y^2} \right),$$
$$h_{i,j}^{k+1} = h_{i,j}^k + \Delta t q_{i,j}^{k+1}.$$

Hight-field surface approximation

DEMO!

References

[1] W.J. Laan, S. Green, M. Sainz; Screen Space Fluid Rendering with Curvature Flow; I3D 2009.

[2] S. Green; Screen Space Fluid Rendering for Games; GDC 2010.

[3] Jos Stam; Stable Fluids; ACM Transactions on Graphics, 2001.

[4] R.Bridson, M.Müller; Fluid simulation; SIGGRAPH 2007 course notes.

[5] GPU Gems 3; Chapter 30: Real-Time Simulation and Rendering of 3D Fluids.

[6] GPU Gems; Chapter 38: Fast Fluid Dynamics Simulation on the GPU; https://developer.download.nvidia.com/books/HTML/gpugems/gpugems_ch38.html

[7] C. Johanson; Real-time water rendering; Master thesis, Lund University,2004.