太极图形课

第10讲 Fluid Simulation 01: The Particle-based Methods

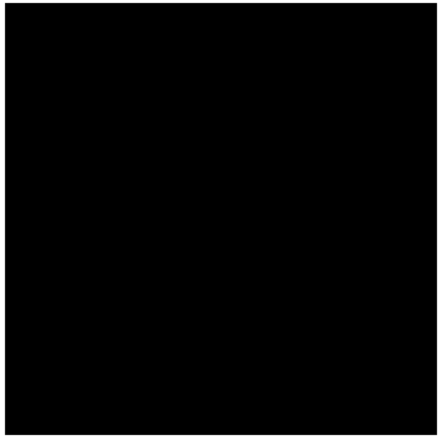
taichi

Where are we?

Procedural Animation

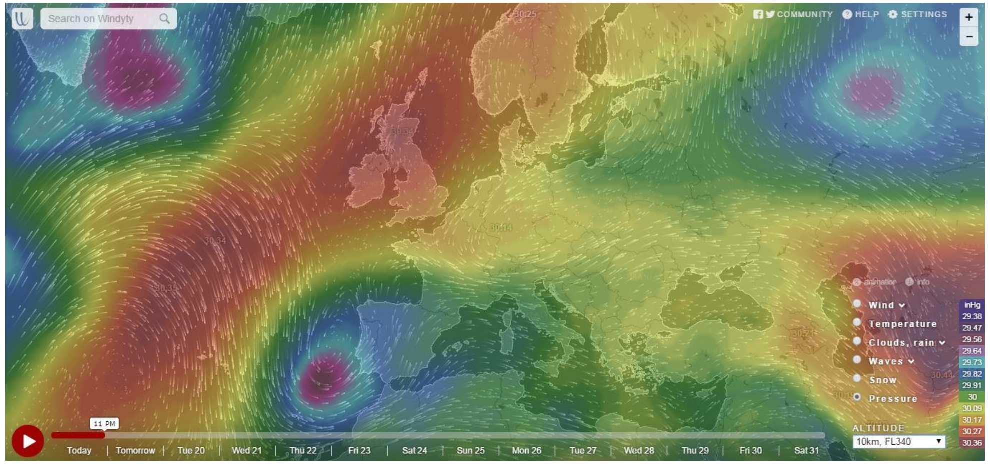
Rendering

Deformable Simulation

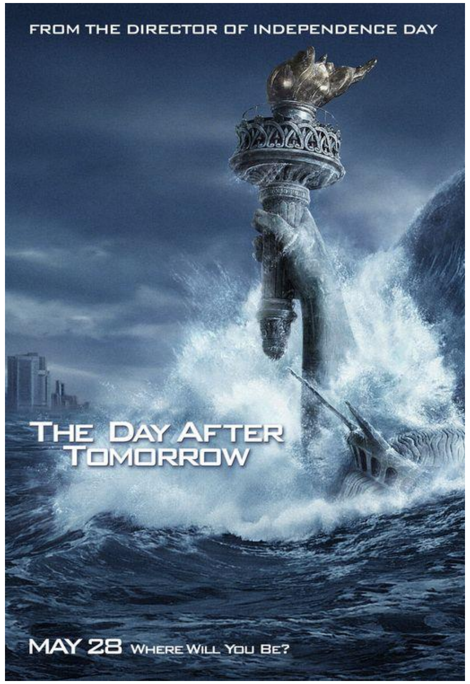


Fluid Simulation

Fluid simulation

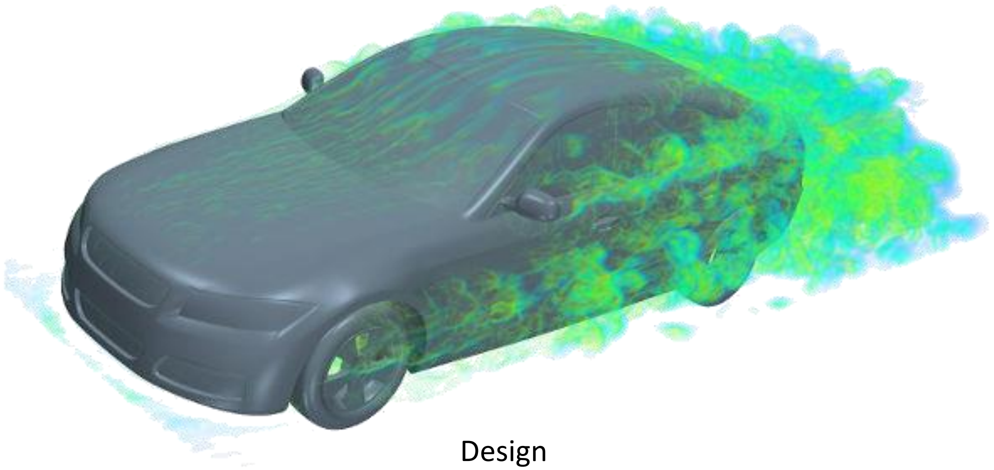


Forecast



VFX

Game

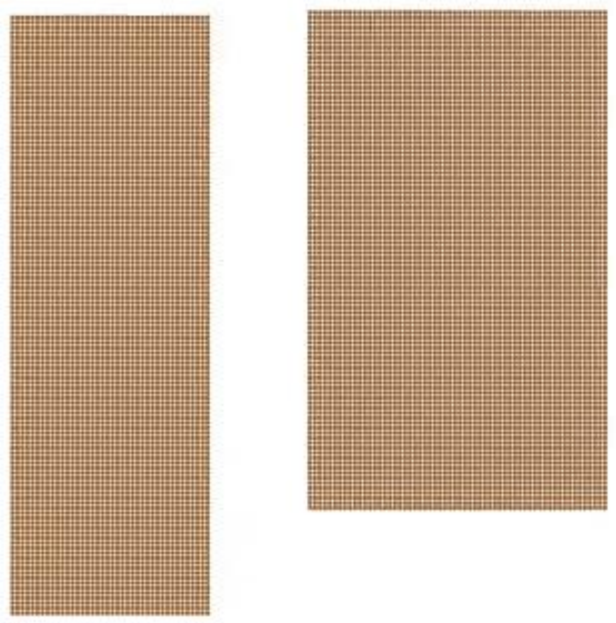


Design

Animation

Code of the day

- Code: https://github.com/taichiCourse01/taichi_sph
- Code courtesy of @erizmr

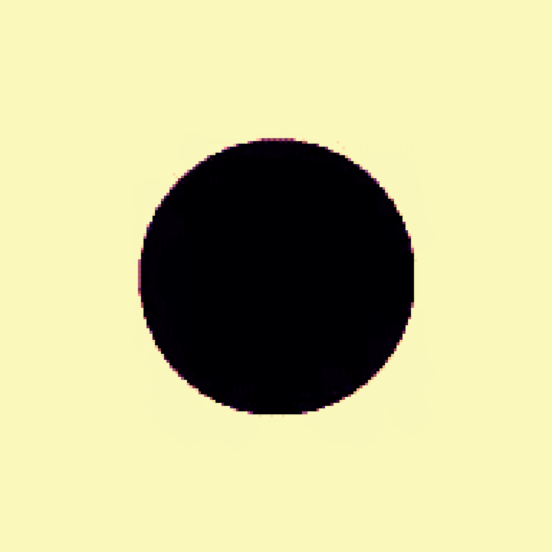


Outline today

- Incompressible fluid dynamics
  - Incompressible Navier–Stokes equations
- Time discretization
  - Operator splitting
  - Integration with the weakly compressible (WC) assumption
- Spatial discretization
  - Smoothed particle hydrodynamics (SPH)
- Implementation details (WCSPH)
  - Simulation Pipeline
  - Boundary conditions
  - Neighbor search

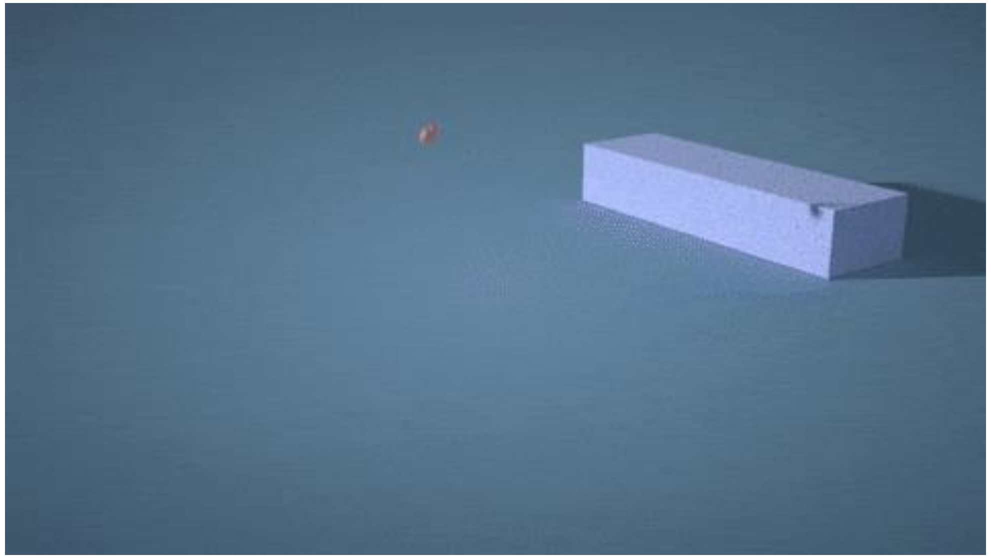
Incompressible fluid dynamics

Compressible and incompressible fluids



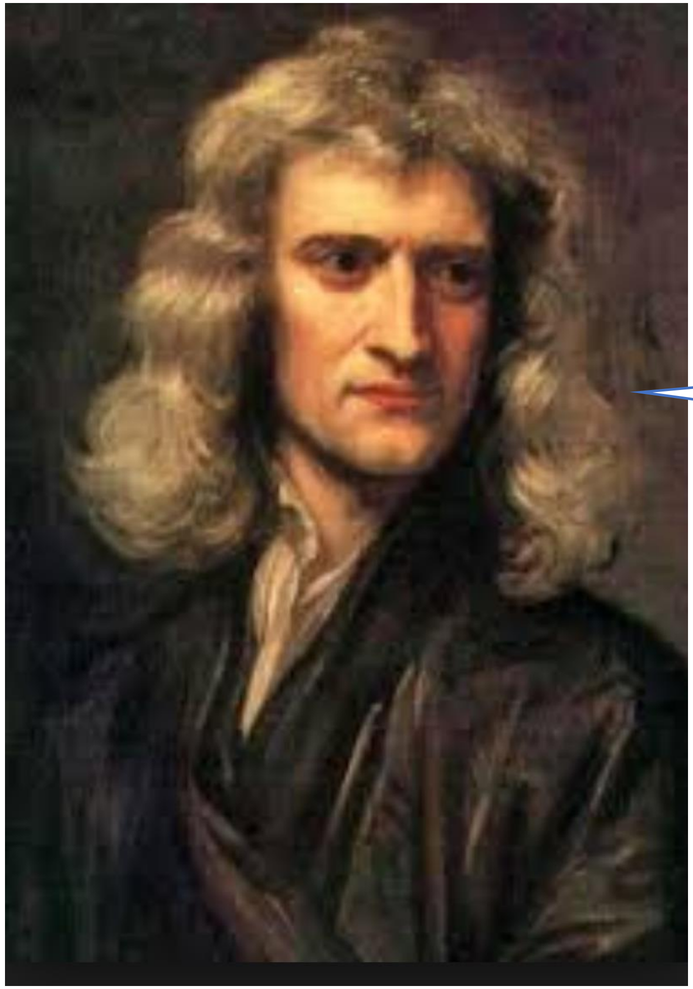
Compressible
shock wave

Incompressible
smoke



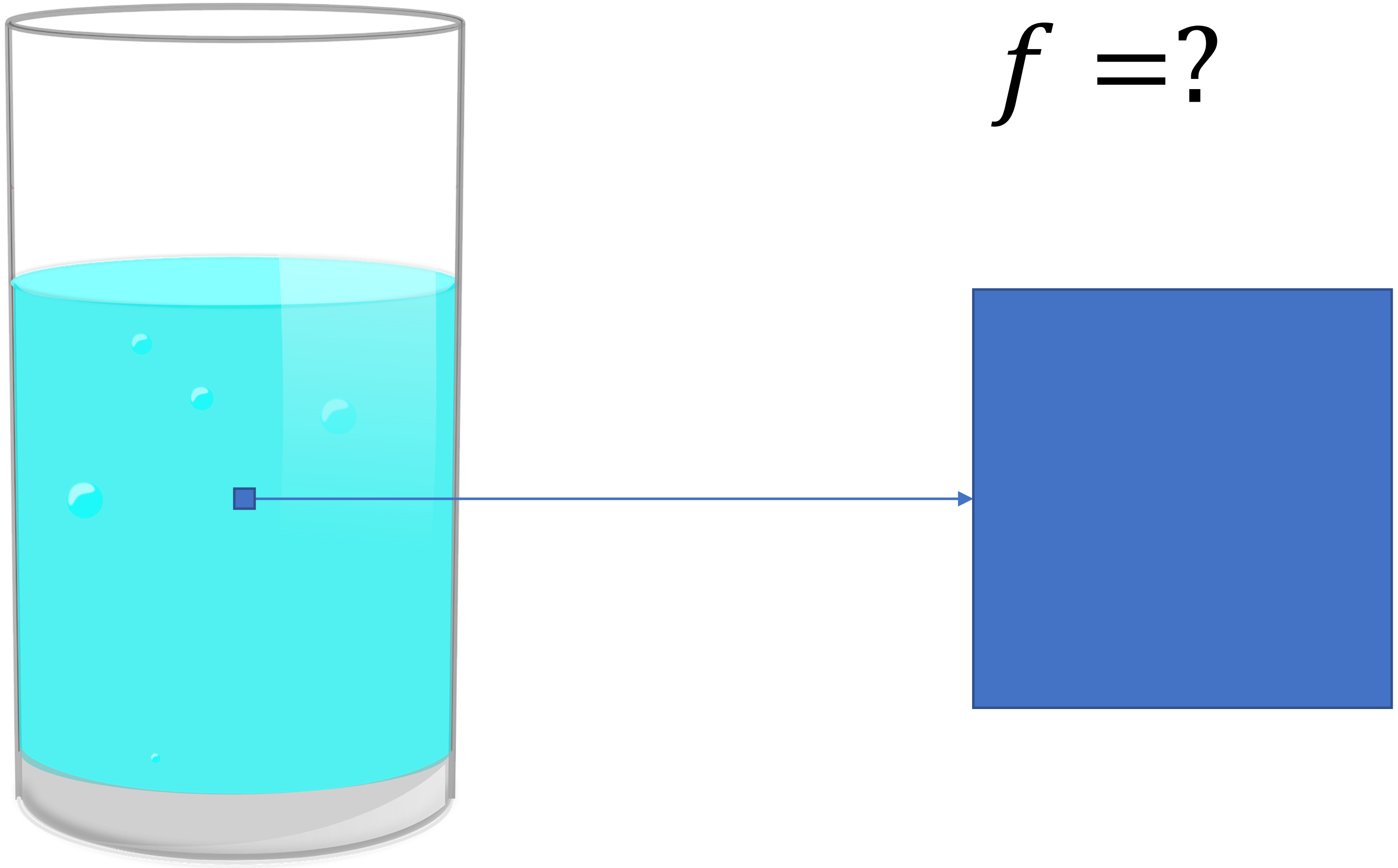
Incompressible
liquid

Laws of physics for incompressible fluids



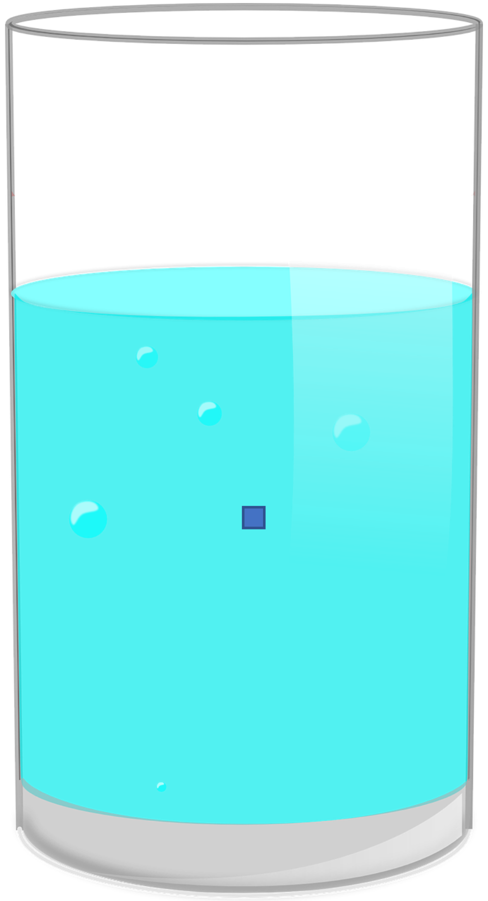
$$\boldsymbol{f} = \mathbf{M}\boldsymbol{a}$$

Forces for incompressible fluids



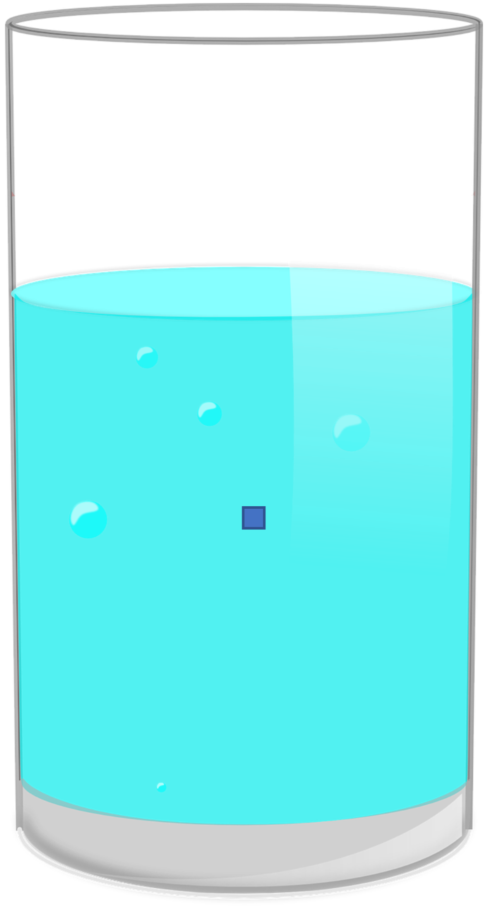
$$f = ?$$

Forces for incompressible fluids

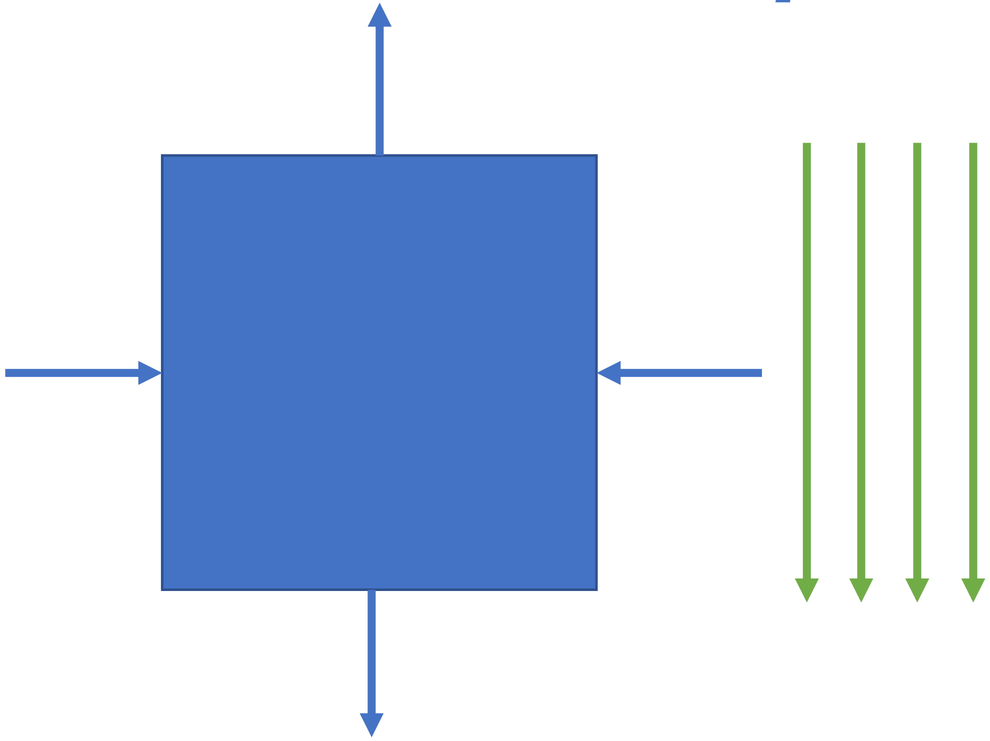


$$f = f_{ext}$$

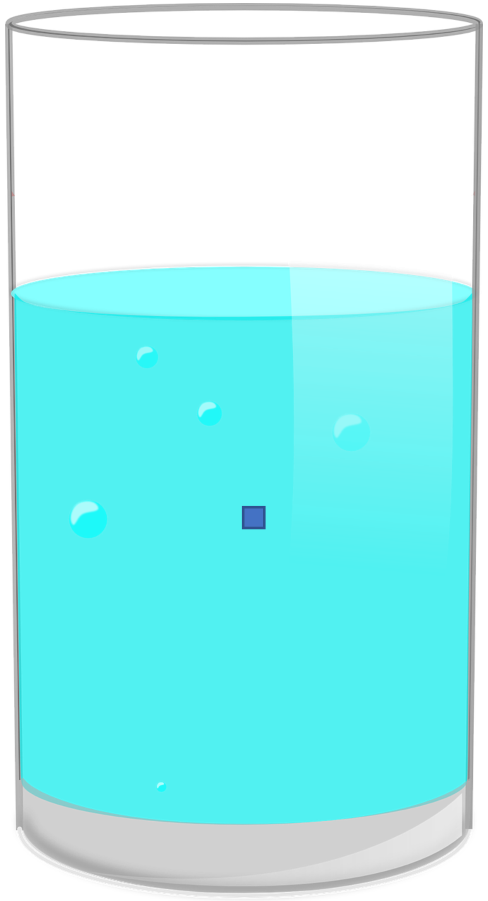
Forces for incompressible fluids



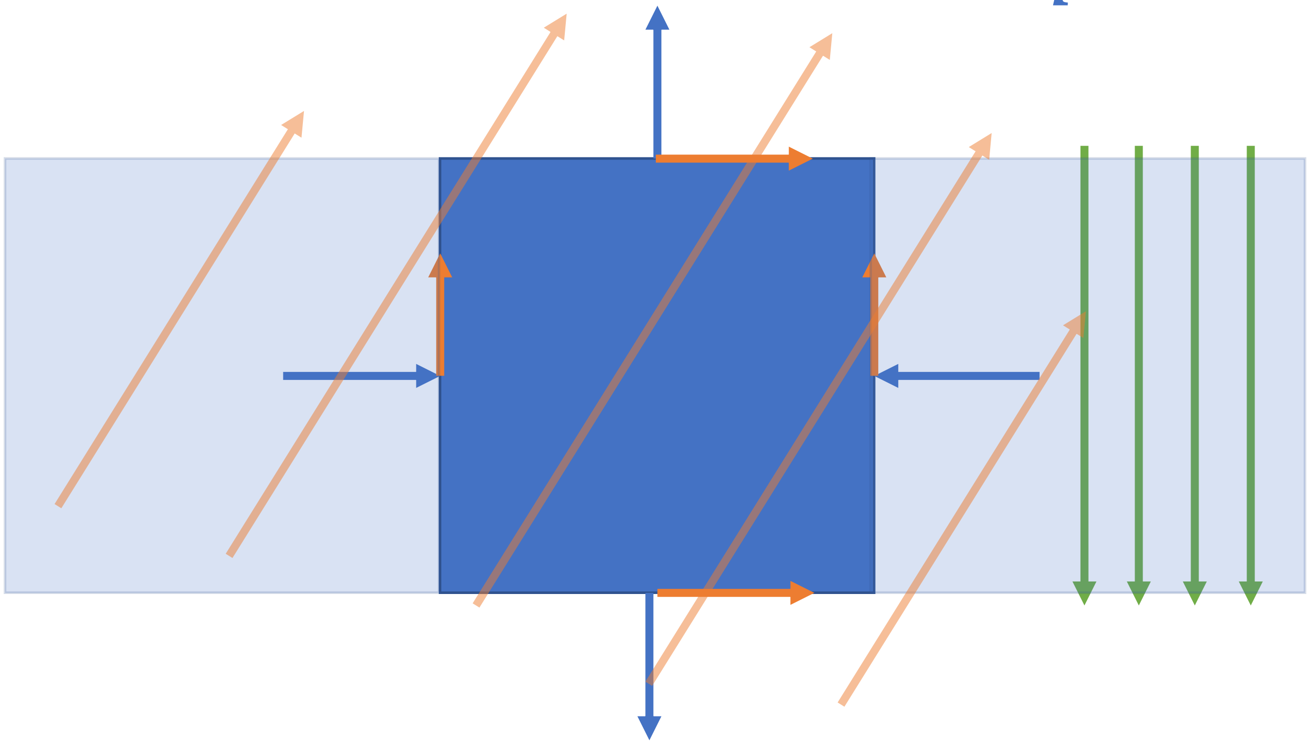
$$f = f_{ext} + f_{pres}$$



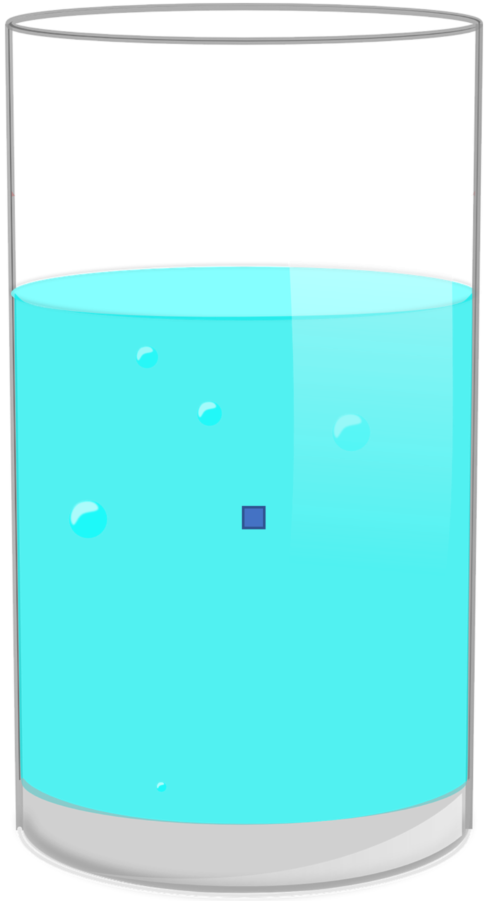
Forces for incompressible fluids



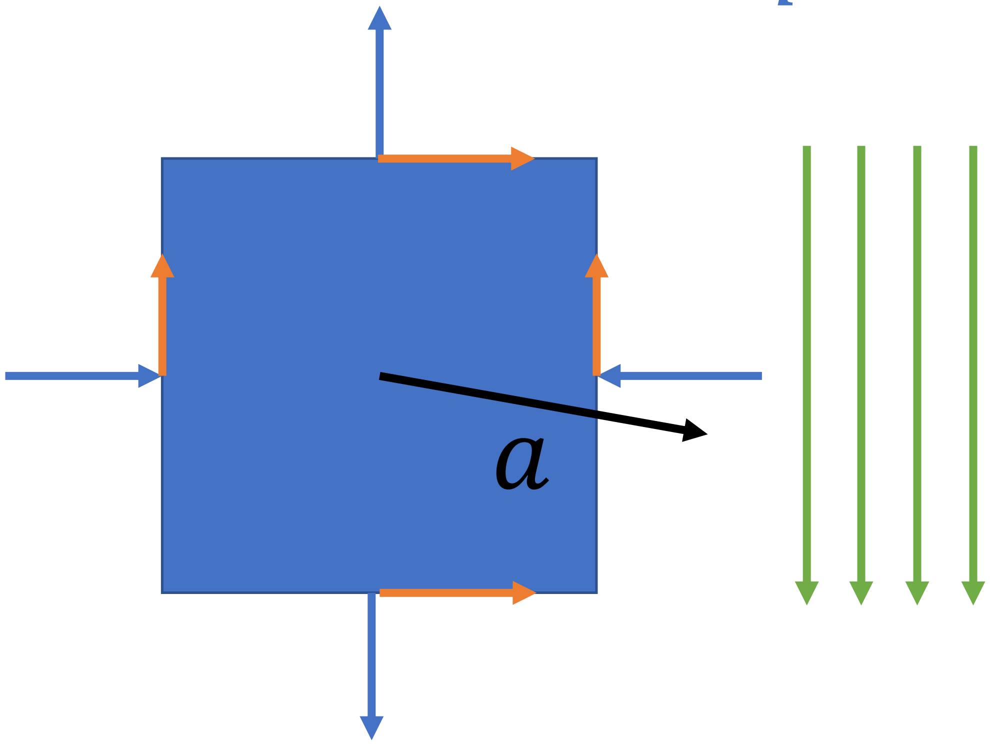
$$f = f_{ext} + f_{pres} + f_{visc}$$



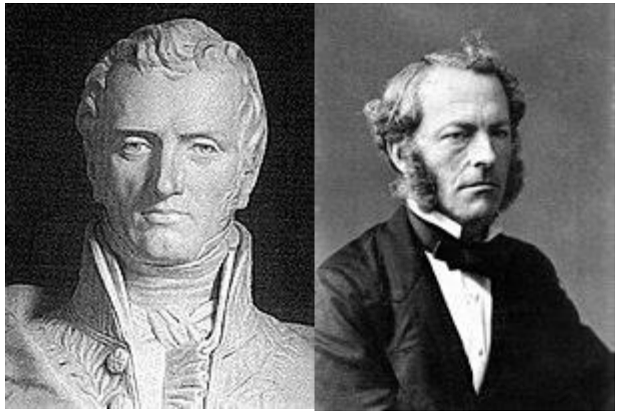
Laws of motion for incompressible fluids



$$ma = f_{ext} + f_{pres} + f_{visc}$$



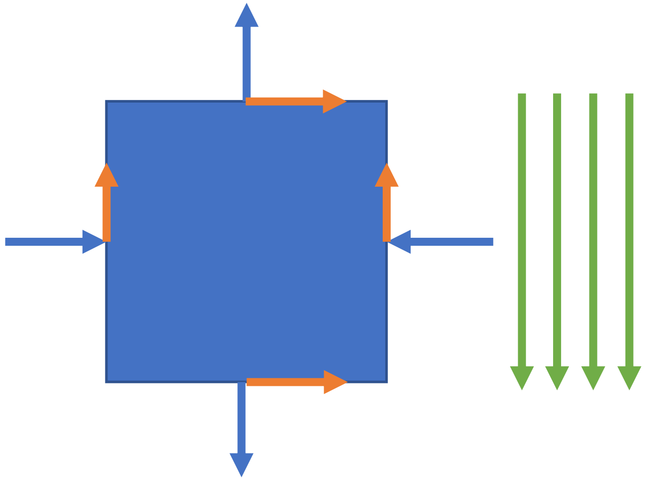
Incompressible Navier-Stokes equation



$$ma = f_{ext} + f_{pres} + f_{visc}$$

$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

$$\nabla \cdot v = 0$$



The spatial derivative operators in 3D

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

- Gradient $\nabla: \mathbb{R}^1 \to \mathbb{R}^3$
  - $grad\ s = \nabla s = \left[\frac{\partial s}{\partial x}, \frac{\partial s}{\partial y}, \frac{\partial s}{\partial z}\right]^T$
- Divergence $\nabla\ \cdot: \mathbb{R}^3 \to \mathbb{R}^1$
  - $div\ v = \nabla \cdot v = \frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z}$
- Curl $\nabla\ \times: \mathbb{R}^3 \to \mathbb{R}^3$
  - $curl\ v = \nabla \times v = \left[\frac{\partial v_z}{\partial y} - \frac{\partial v_y}{\partial z}, \frac{\partial v_z}{\partial x} - \frac{\partial v_x}{\partial z}, \frac{\partial v_y}{\partial x} - \frac{\partial v_x}{\partial y}\right]^T$
- Laplace $\Delta = \nabla^2 = \nabla \cdot \nabla: \mathbb{R}^n \to \mathbb{R}^n$
  - $laplace\ s = div\ (grad\ s) = \nabla^2 s = \frac{\partial^2 s}{\partial x^2} + \frac{\partial^2 s}{\partial y^2} + \frac{\partial^2 s}{\partial z^2}$

Incompressible Navier-Stokes equation

$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

$$\nabla \cdot v = 0$$

ρ: mass-density

$\frac{D(\cdot)}{Dt}$: material derivative

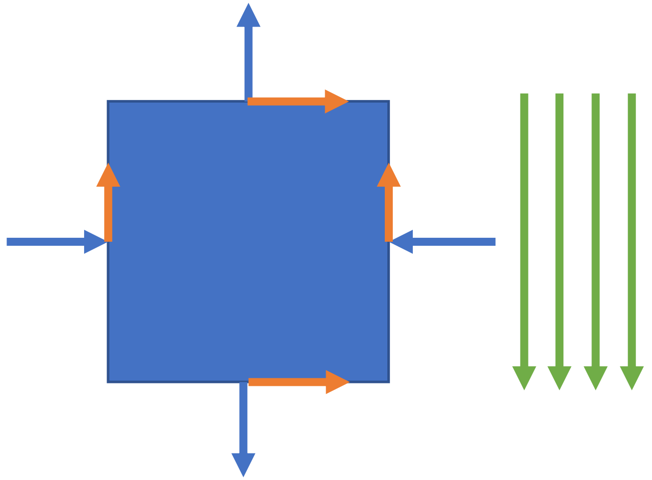
g: gravity ($g = [0, -9.8, 0]^T$ in 3D)

p: pressure, $p = k(\rho - \rho_0)$

μ: shear modulus (dynamics visc.), $\nu = \frac{\mu}{\rho_0}$: kinematic viscosity

∇: vector differential operator ($\nabla = \left[\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}\right]^T$ in 3D)

$\nabla^2 = \Delta$: Laplace operator, $\Delta f = \nabla \cdot \nabla f$ ($\Delta f = \frac{\partial^2}{\partial x^2} f + \frac{\partial^2}{\partial y^2} f + \frac{\partial^2}{\partial z^2} f$ in 3D)



The ma in $f = ma$

$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

$$\nabla \cdot v = 0$$

This is simply "mass" times "acceleration" divided by "volume"

External force term

$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

$$\nabla \cdot v = 0$$

Gravitational force divided by “volume”

Viscosity term

$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

$$\nabla \cdot v = 0$$

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

$\nabla^2 = \Delta$: Laplace operator (or diffusion operator)

∇^2: takes a scalar/vector, returns a scalar/vector

$\nabla^2 f = \frac{\partial^2}{\partial x^2} f + \frac{\partial^2}{\partial y^2} f + \frac{\partial^2}{\partial z^2} f$ in 3D

Still remember the diffusion problem?

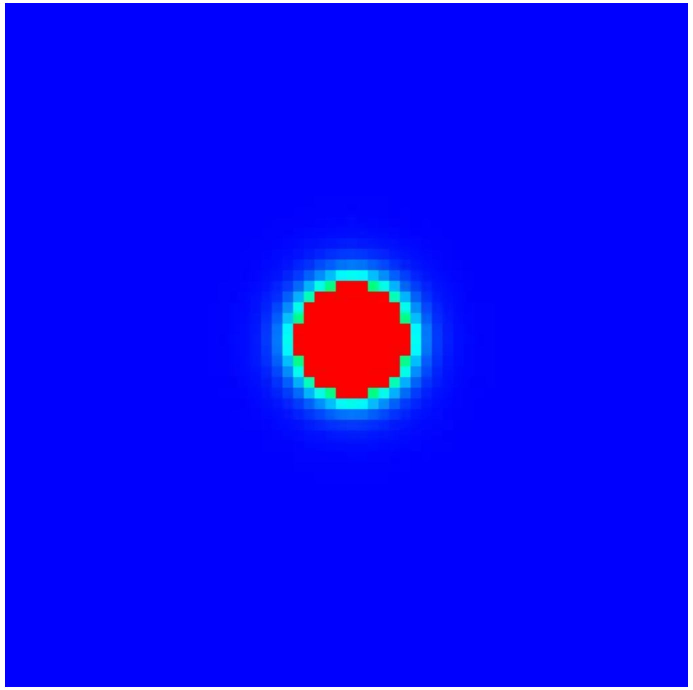
- $\frac{\partial T}{\partial t} = \kappa \nabla^2 T$

	0	
0	100	0
	0	

$T \downarrow$

	100	
100	0	100
	100	

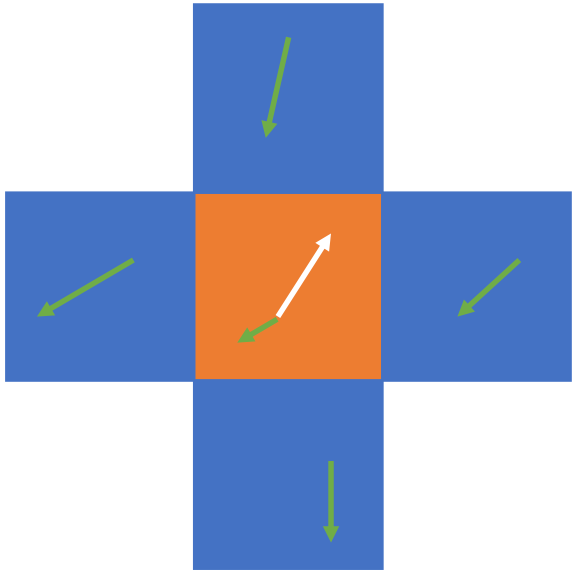
$T \uparrow$



Viscosity term

$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

$$\nabla \cdot v = 0$$



Viscosity term: how fluids want to move together

Viscosity term

$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

$$\nabla \cdot v = 0$$

μ:  >  > 

μ: some fluids are more viscous than others

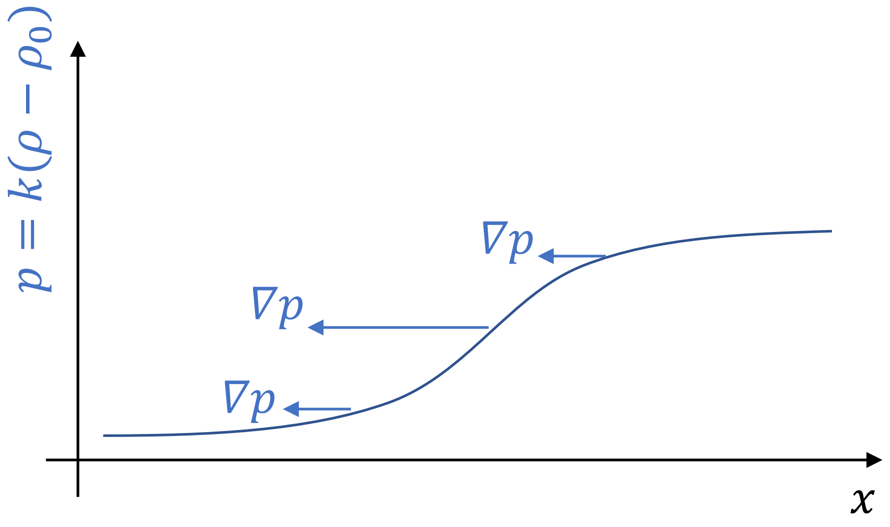
Pressure term

$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

$$\nabla \cdot v = 0$$

∇: gradient operator
∇: takes a scalar, returns a vector

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



Pressure term: fluids do not want to change volume

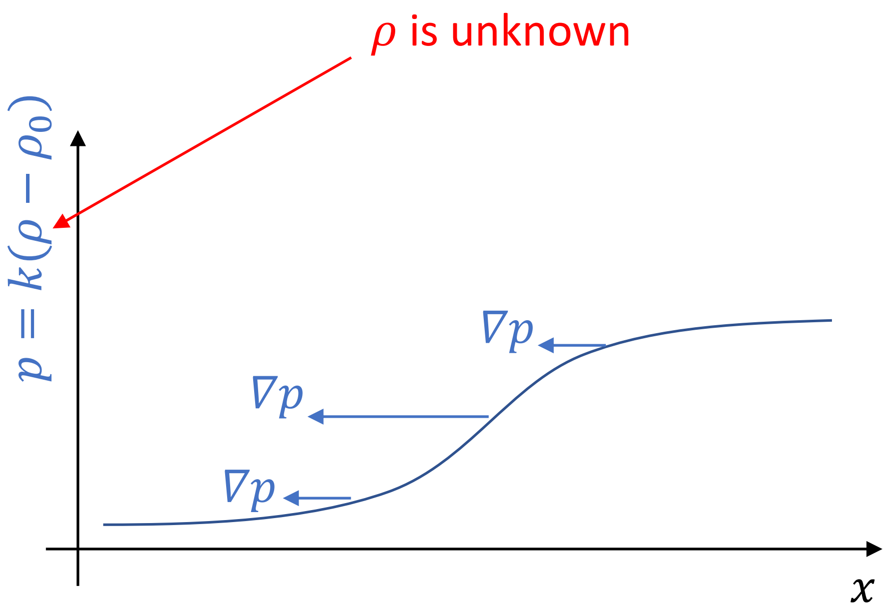
Pressure term

$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

$$\nabla \cdot v = 0$$

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

∇: gradient operator
∇: takes a scalar, returns a vector



Pressure term: fluids do not want to change volume

Divergence free condition

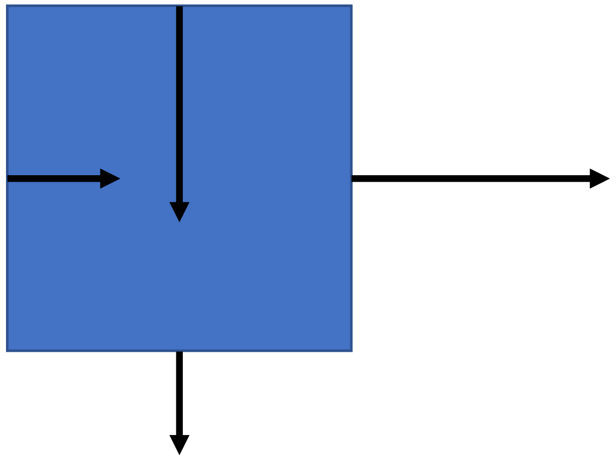
$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

$$\nabla \cdot v = 0$$

$$\nabla \cdot v = \frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z} = 0$$

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

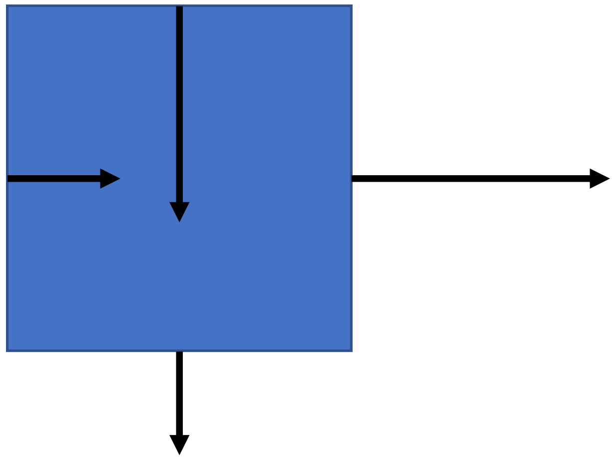
$\nabla \cdot$: divergence operator
$\nabla \cdot$: takes a vector, returns a scalar



Divergence free condition

$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

$$\nabla \cdot v = 0 \Leftrightarrow \frac{D\rho}{Dt} = (v \cdot \nabla)\rho = 0$$



Divergence ($\nabla \cdot v$) free: outbound flow equals to inbound flow
The mass conserving condition

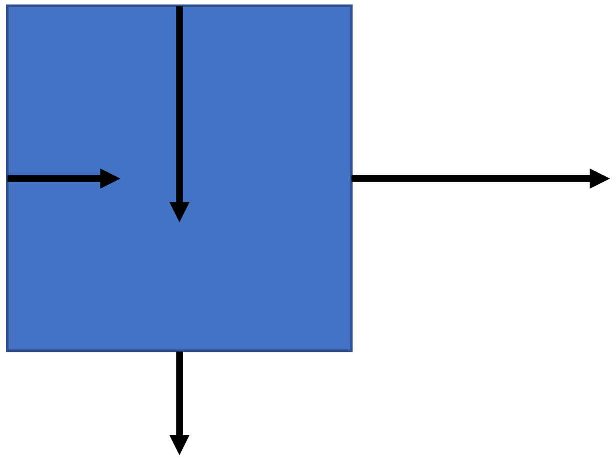
Divergence free condition

$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

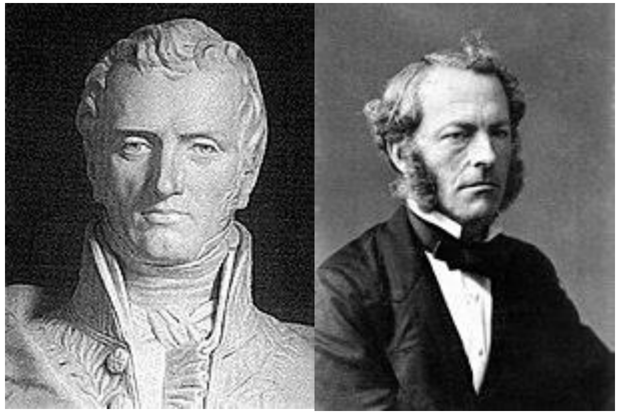
$$\nabla \cdot v = 0$$

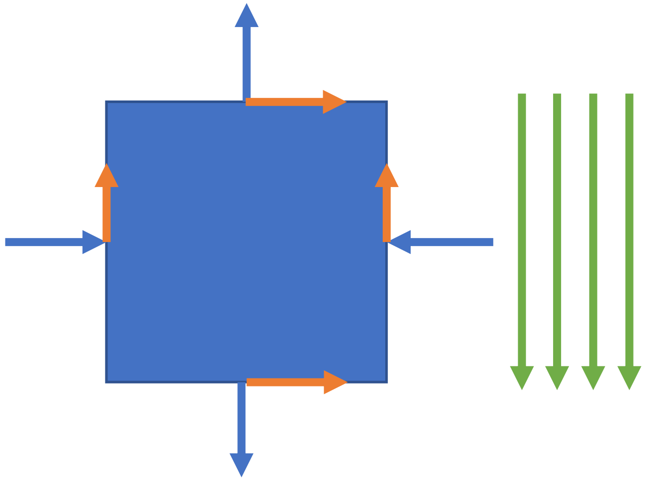
← The incompressibility assumption goes here!

Divergence ($\nabla \cdot v$) free: outbound flow equals to inbound flow
The mass conserving condition



Incompressible Navier-Stokes equation





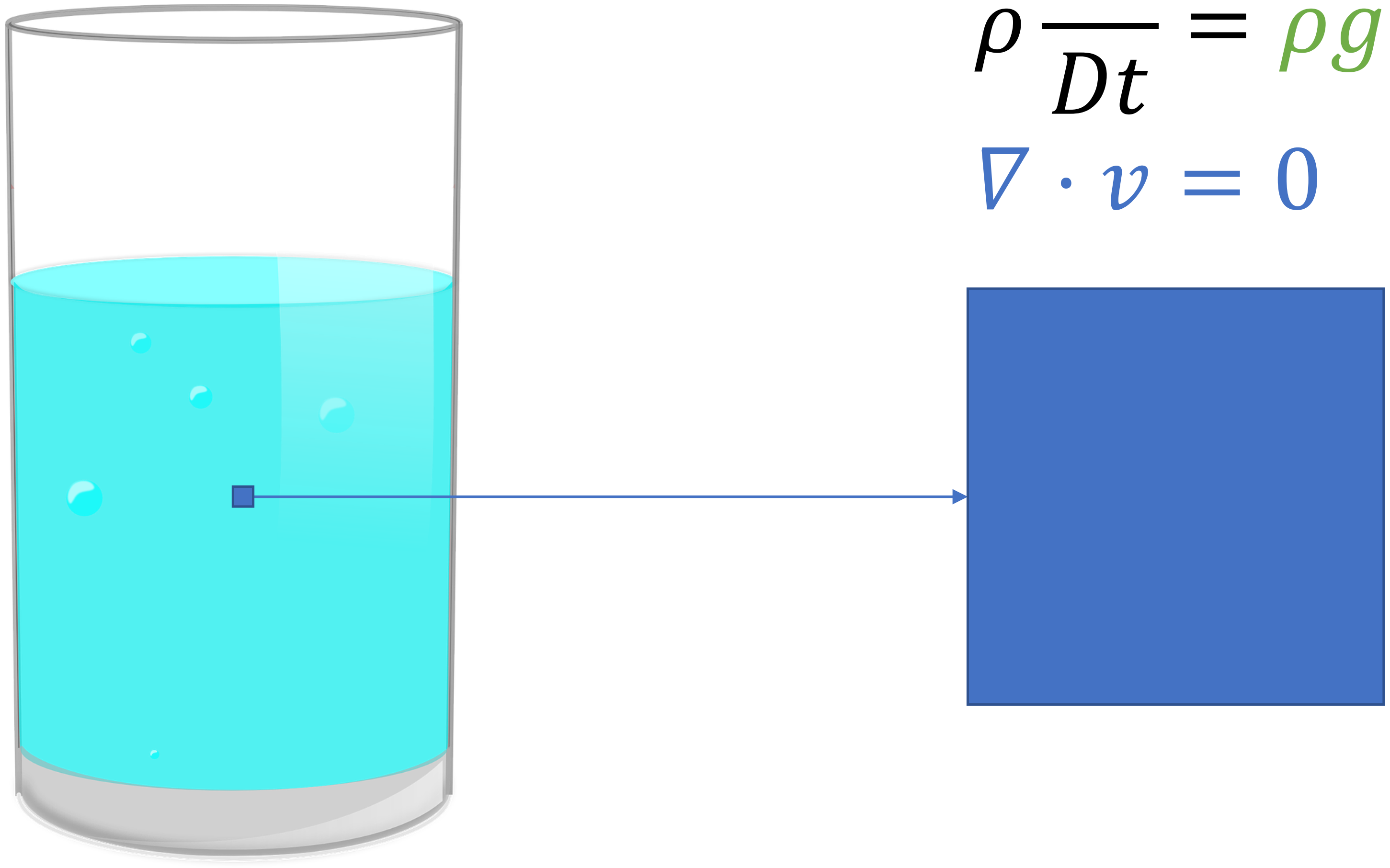
$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$ ←The momentum equation

$$\nabla \cdot v = 0$$ ←The mass conserving condition

Integrate incompressible Navier-Stokes equation?

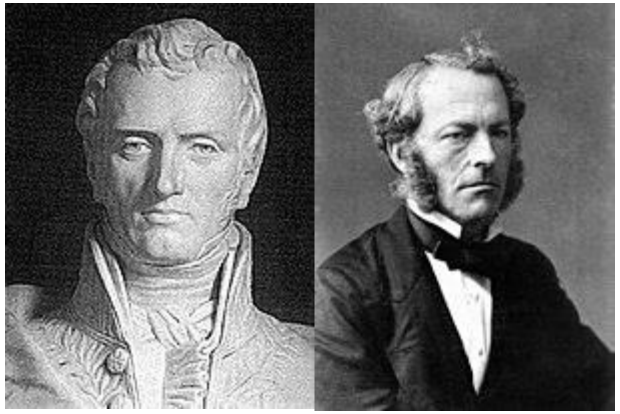
$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

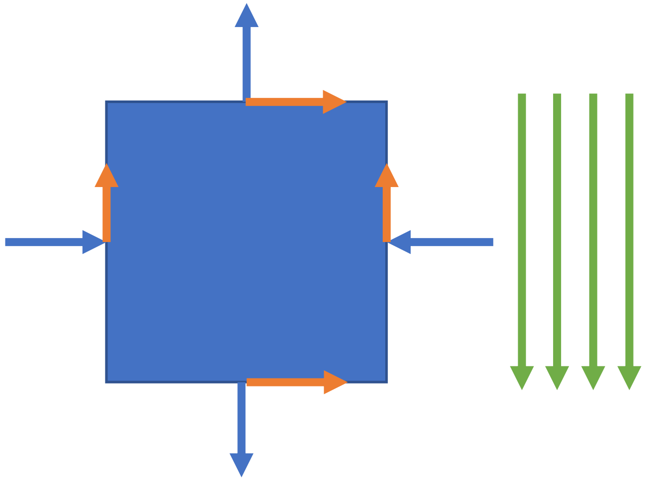
$$\nabla \cdot v = 0$$



Temporal discretization

Incompressible Navier-Stokes equation





The ρ in $p = k(\rho - \rho_0)$ is unknown!

$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

←The momentum equation

$$\nabla \cdot v = 0$$

←The mass conserving condition

$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$
$$\nabla \cdot v = 0$$

- The divergence free condition $\nabla \cdot v = 0$ comes to the rescue:
  - $\nabla \cdot v = 0 \Leftrightarrow \frac{D\rho}{Dt} = 0$
- Integrate the incompressible Navier-Stokes equation in steps:
  - Step 1: input v_n, output $v_{n+0.5}$
    - $\rho \frac{Dv}{Dt} = \rho g + \mu \nabla^2 v$
  - Step 2: input $v_{n+0.5}$, output v_{n+1}
    - $\rho \frac{Dv}{Dt} = -\nabla p$
    - $\nabla \cdot v = 0$

This integration method is sometime referred as "*Operator splitting*" or "*Advection-Projection*" in different contexts

Full time integration

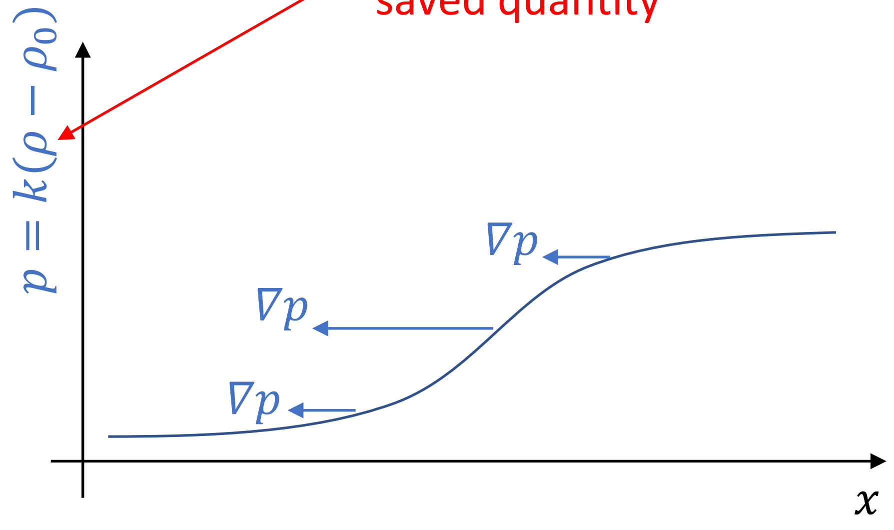
- Given x_n, v_n:
  - Step 1: Advection / external and viscosity force integration
    - Solve: $dv = g + \nu \nabla^2 v_n$
    - Update: $v_{n+0.5} = v_n + \Delta t\ dv$
  - Step 2: Projection / pressure solver
    - Solve: $dv = -\frac{1}{\rho} \nabla\left(k(\rho - \rho_0)\right)$ *and* $\frac{D\rho}{Dt} = \nabla \cdot (v_{n+0.5} + dv) = 0$
    - Update: $v_{n+1} = v_{n+0.5} + \Delta t\ dv$
  - Step 3: Update position
    - Update: $x_{n+1} = x_n + \Delta t\ v_{n+1}$
  - Return x_{n+1}, v_{n+1}

The weakly compressible assumption

- Storing the density ρ as an individual variable that advect with the velocity field:

$$\frac{Dv}{Dt} = g - \frac{1}{\rho}\nabla p + \nu\nabla^2 v$$

$$\cancel{\nabla \cdot v = 0}$$



Integrate with the weakly compressible assumption

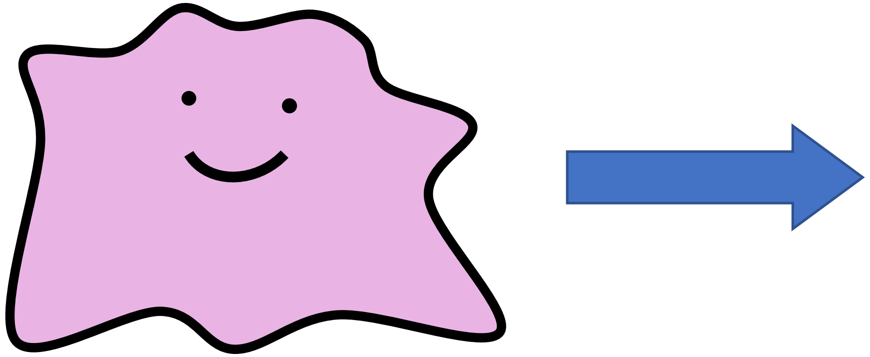
$$\frac{Dv}{Dt} = g - \frac{1}{\rho}\nabla p + \nu \nabla^2 v$$

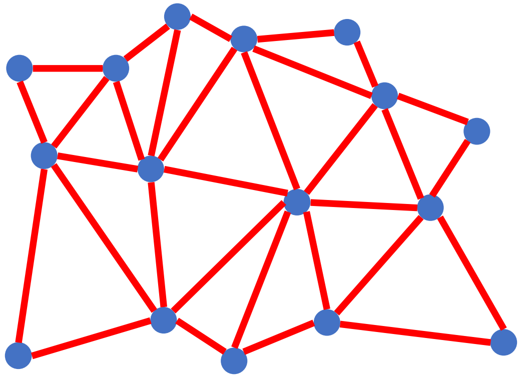
- Given x_n, v_n:
  - Step 1: Advection / external and viscosity force integration
    - Solve: $dv = g + \nu \nabla^2 v_n$
    - Update: $v_{n+0.5} = v_n + \Delta t \; dv$
  - Step 2: Projection / pressure solver
    - Solve: $dv = -\frac{1}{\rho}\nabla\big(k(\rho - \rho_0)\big)$
    - Update: $v_{n+1} = v_{n+0.5} + \Delta t \; dv$
  - Step 3: Update position
    - Update: $x_{n+1} = x_n + \Delta t \; v_{n+1}$
  - Return x_{n+1}, v_{n+1}

This is nothing but *Symplectic Euler* integration.

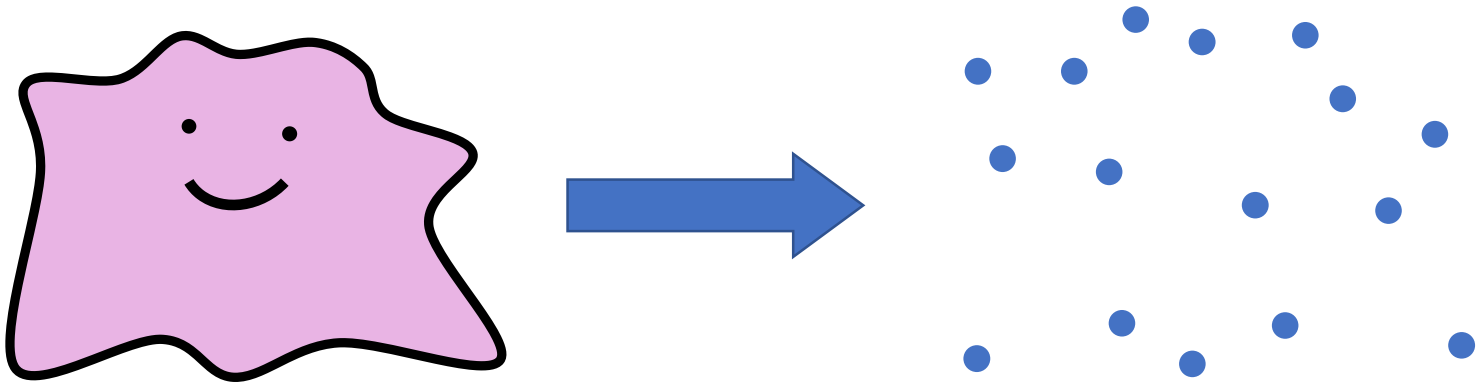
Spatial discretization (Lagrangian view)

Previously in this course (mesh-based simulation)





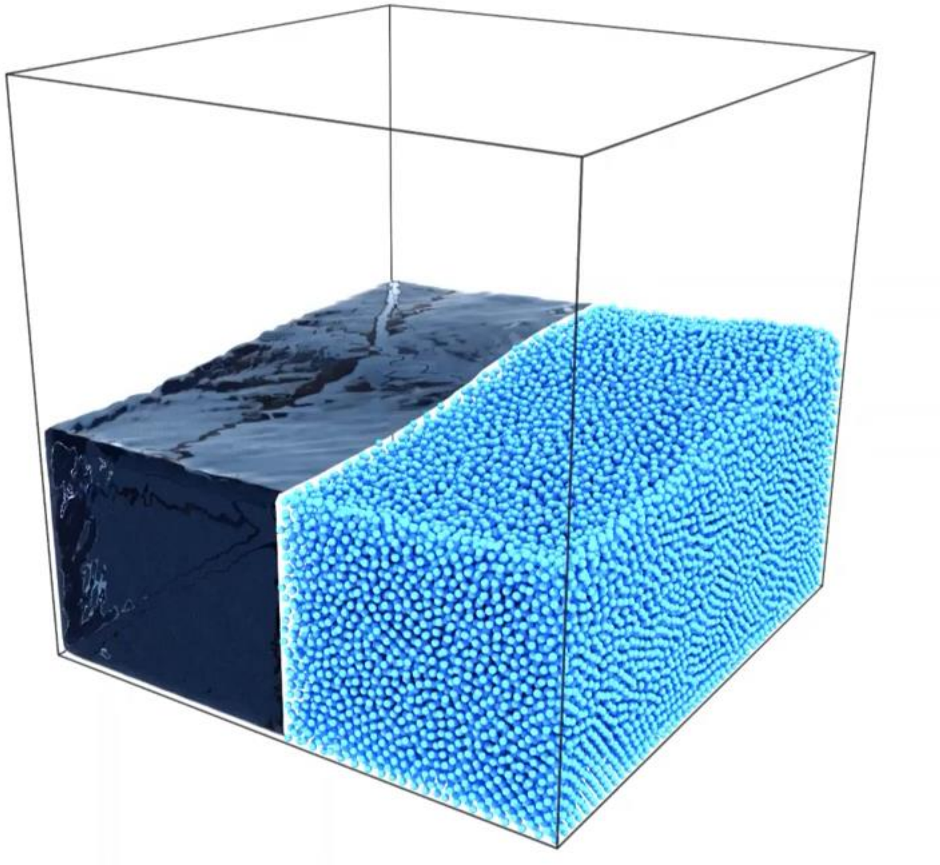
Today in this course (mesh-free simulation)



Fluid can be discretized using particles as well

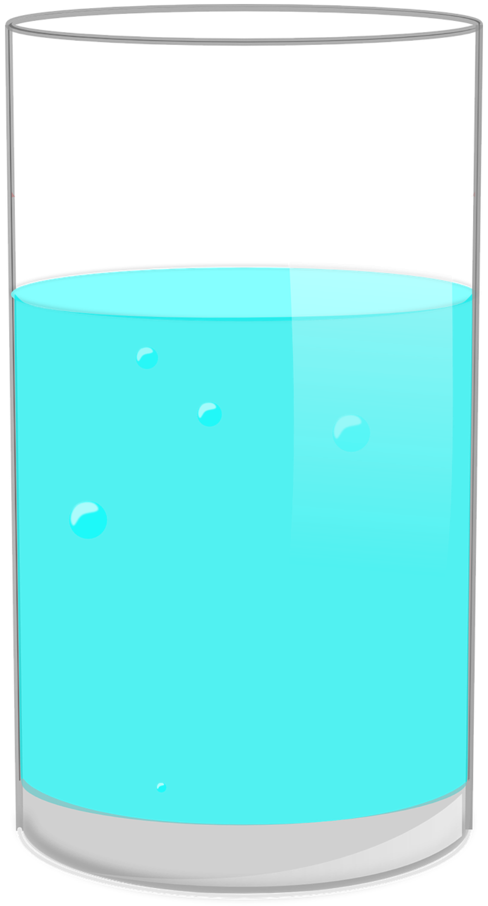
[The Big Bang Theory]

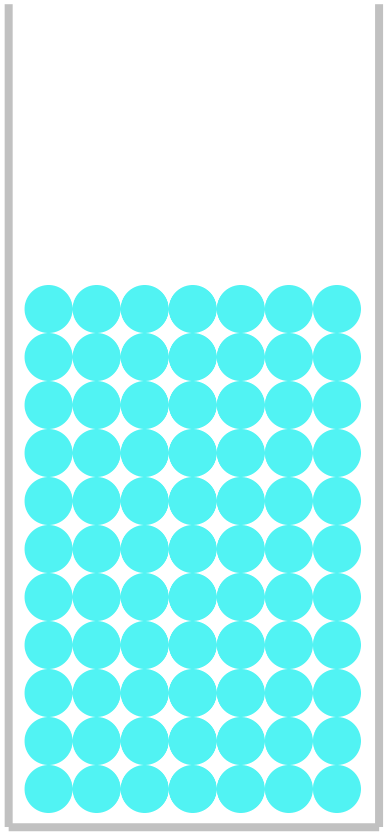
Fluid can be discretized using particles as well



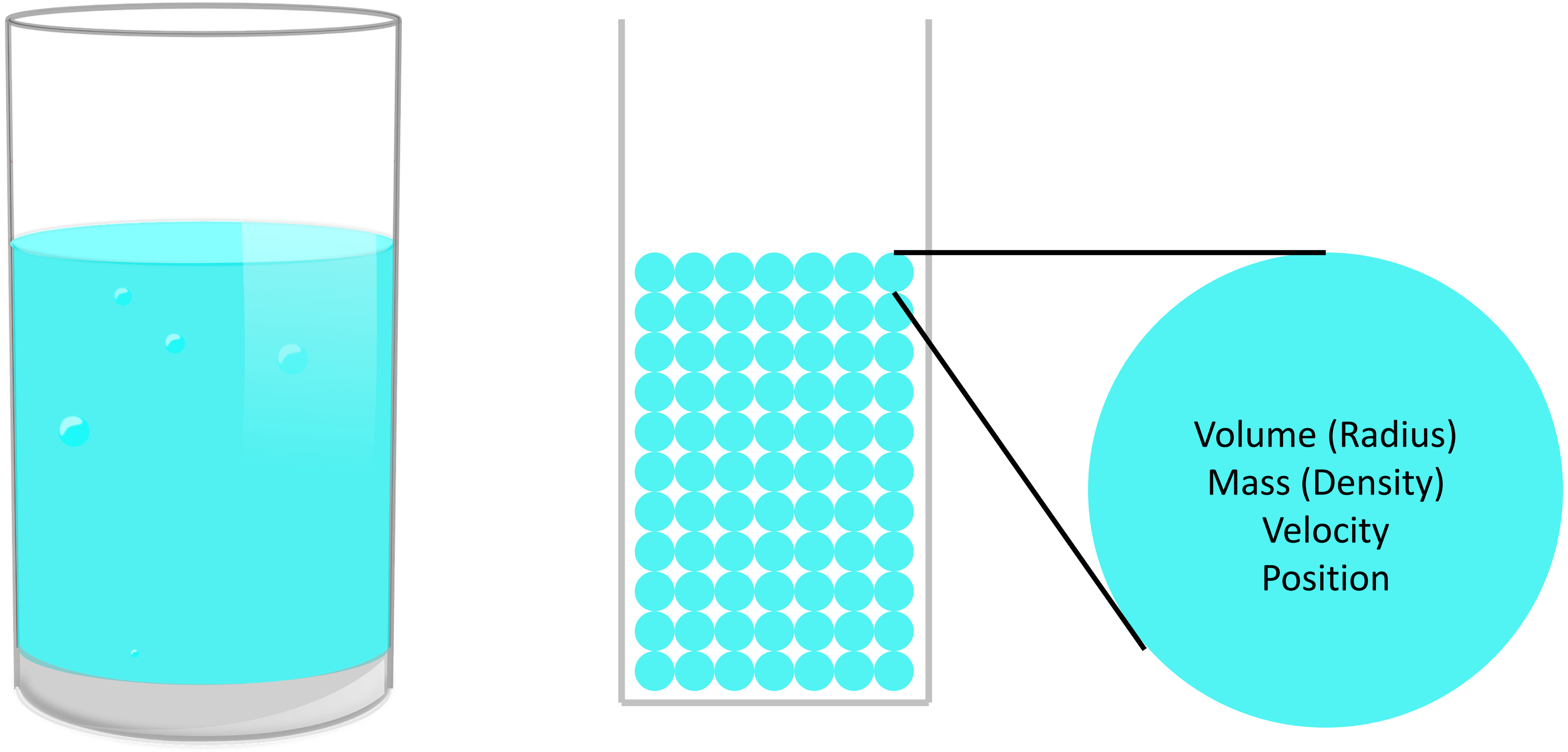
[www.dive-solutions.de]

Fluid -> Particles

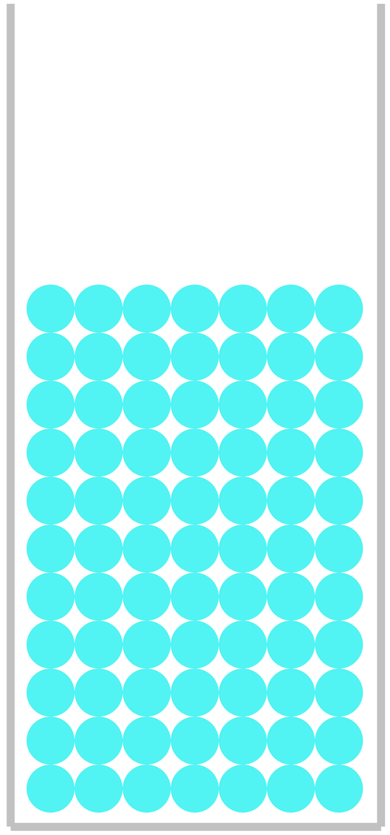




What do we store in a particle?



Fluid dynamics with particles (weakly compressible)



Continuous view:

$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

$$\cancel{\nabla \cdot v = 0}$$

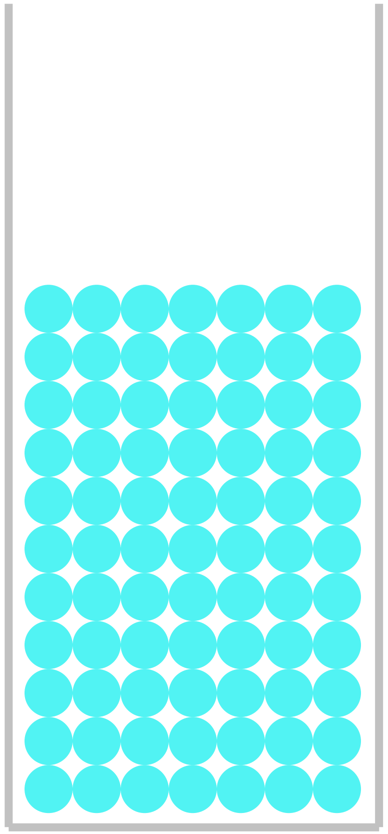
Discrete view (using particles):

$$\frac{dv_i}{dt} = \underbrace{g - \frac{1}{\rho}\nabla p(x_i) + \nu \nabla^2 v(x_i)}_{a_i} \quad , \mathit{where}\ \nu = \frac{\mu}{\rho_0}$$

Time integration (Symplectic Euler):

- for i in particles:
  - $v_i = v_i + \Delta t\, a_i$
- for i in particles:
  - $x_i = x_i + \Delta t\, v_i$

Fluid dynamics with particles (weakly compressible)



Continuous view:

$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

$$\cancel{\nabla \cdot v = 0}$$

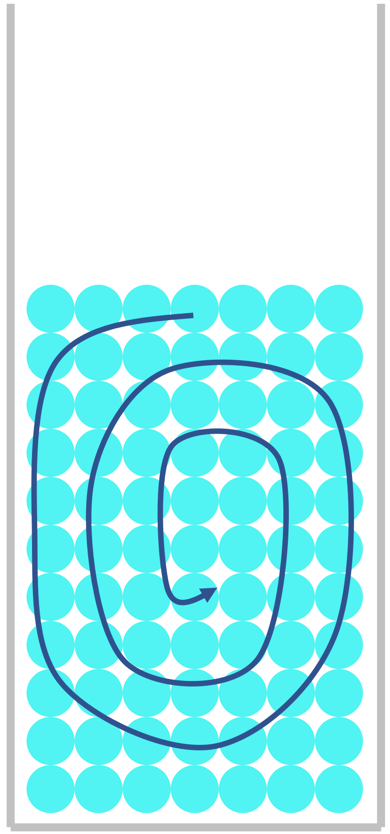
Discrete view (using particles):

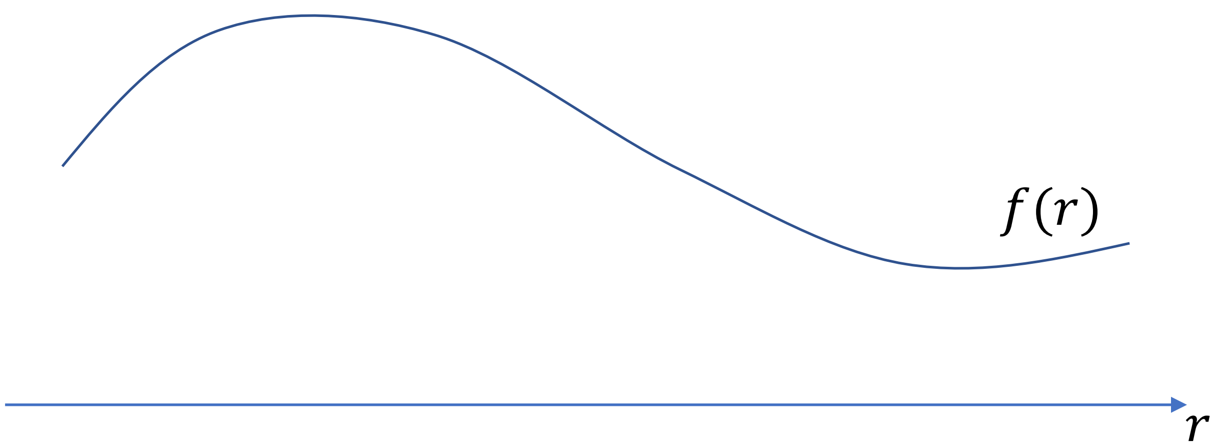
$$\frac{dv_i}{dt} = \underbrace{g - \frac{1}{\rho}\boxed{\nabla p(x_i)} + \nu \boxed{\nabla^2 v(x_i)}}_{a_i} \quad , where\ \nu = \frac{\mu}{\rho_0}$$

Time integration (Symplectic Euler):

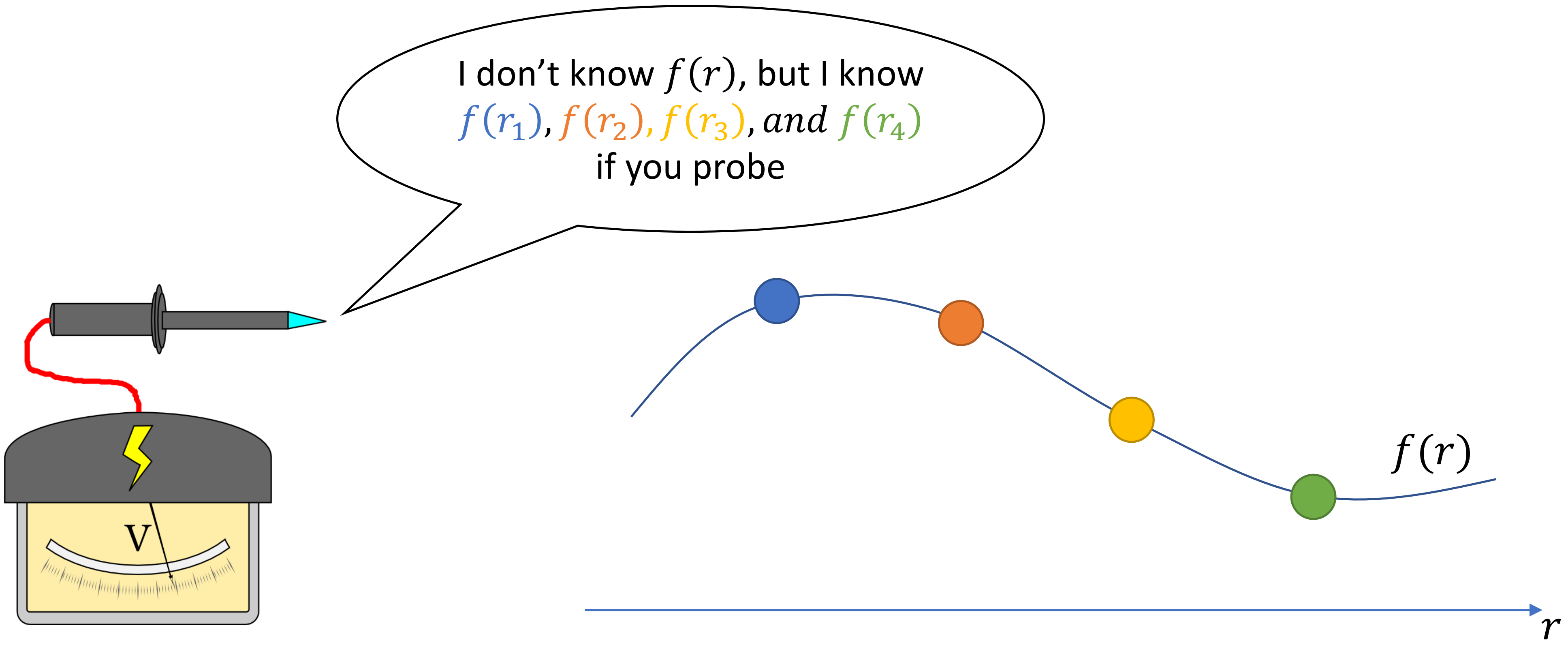
- for i in particles:
  - $v_i = v_i + \Delta t\ a_i$
- for i in particles:
  - $x_i = x_i + \Delta t\ v_i$

How to evaluate a function? $\rho(x_i)$ $\nabla p(x_i)$ $\nabla^2 v(x_i)$





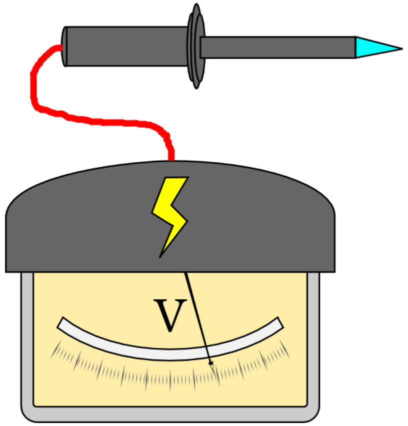
Imaging a magic probe…

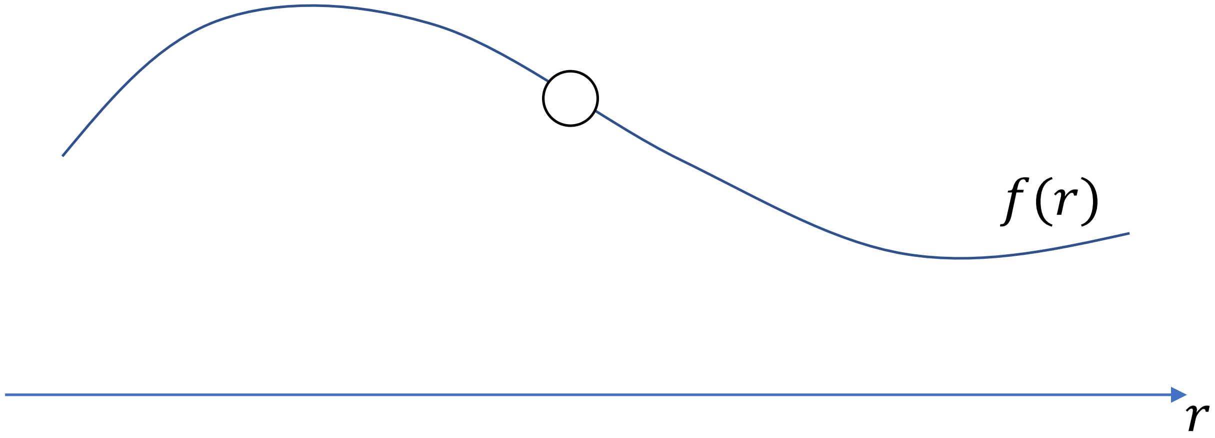


A trivial identity with Dirac delta

$$f(r) = \int_{-\infty}^{\infty} f(r')\delta(r - r')dr'$$

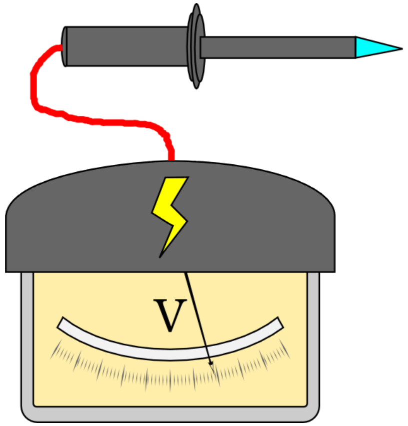
$$\delta(r) = \begin{cases} +\infty, if\ r = 0 \\ 0, otherwise \end{cases}$$ and $\int_{-\infty}^{\infty} \delta(r)dr = 1$

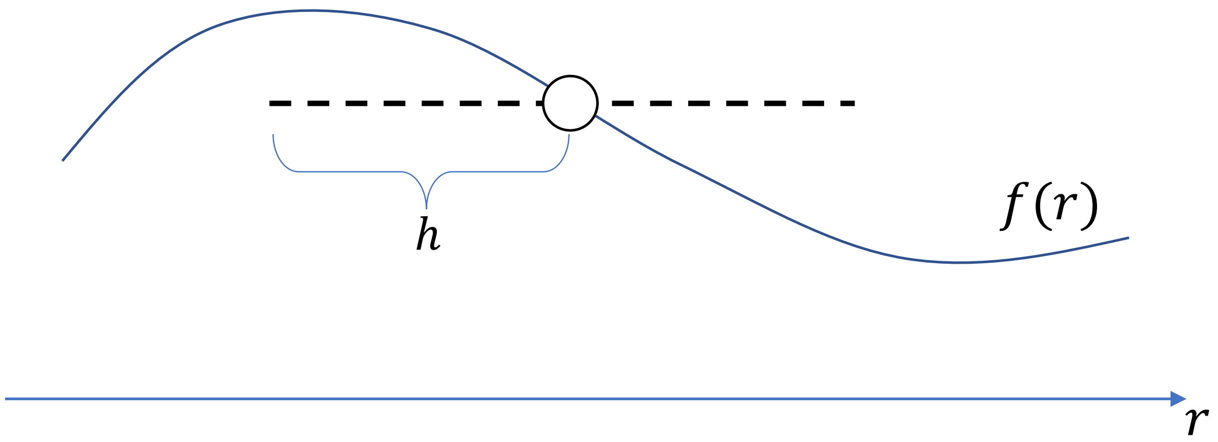




Let us widen the Dirac delta

$$f(r) \approx \int f(r')W(r-r',h)dr' \text{ , where } \lim_{h\to 0} W(r,h) = \delta(r)$$



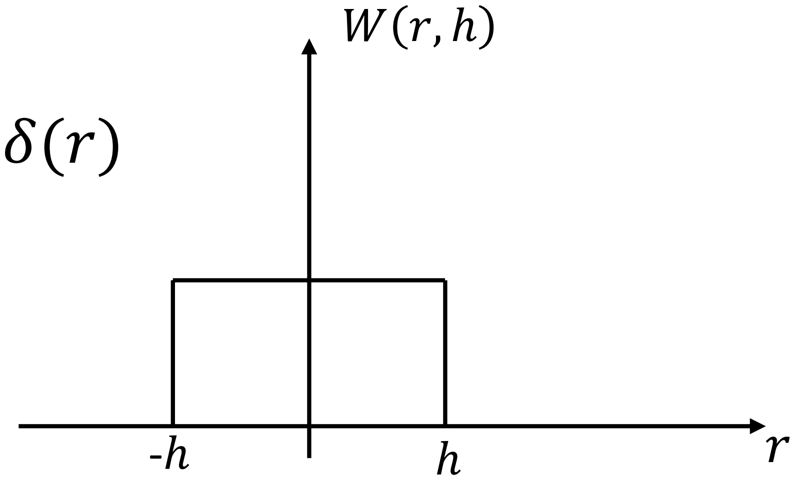


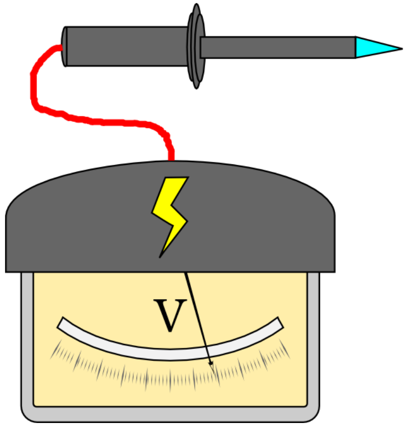
$W(r,h)$: kernel function

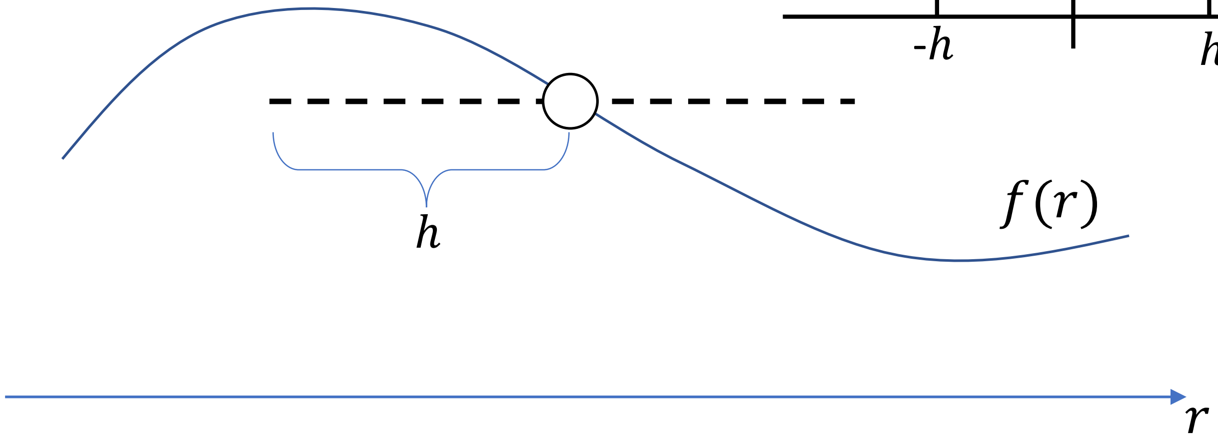
- Sum to unity: $\int W(r,h)dr = 1$
- Symmetric: $W(r,h) = W(-r,h)$
- Compact support: $W(r,h) = 0$ if $|r| > h$

A trivial kernel function

$$f(r) \approx \int f(r')W(r-r',h)dr' \text{ , where } \lim_{h\to 0} W(r,h) = \delta(r)$$



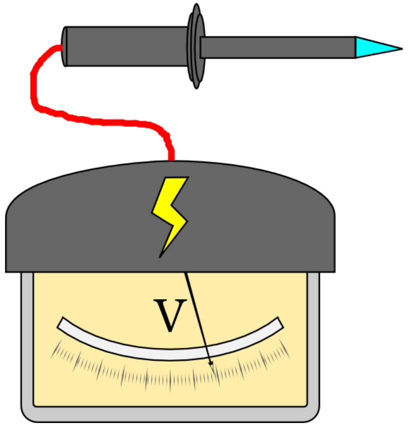


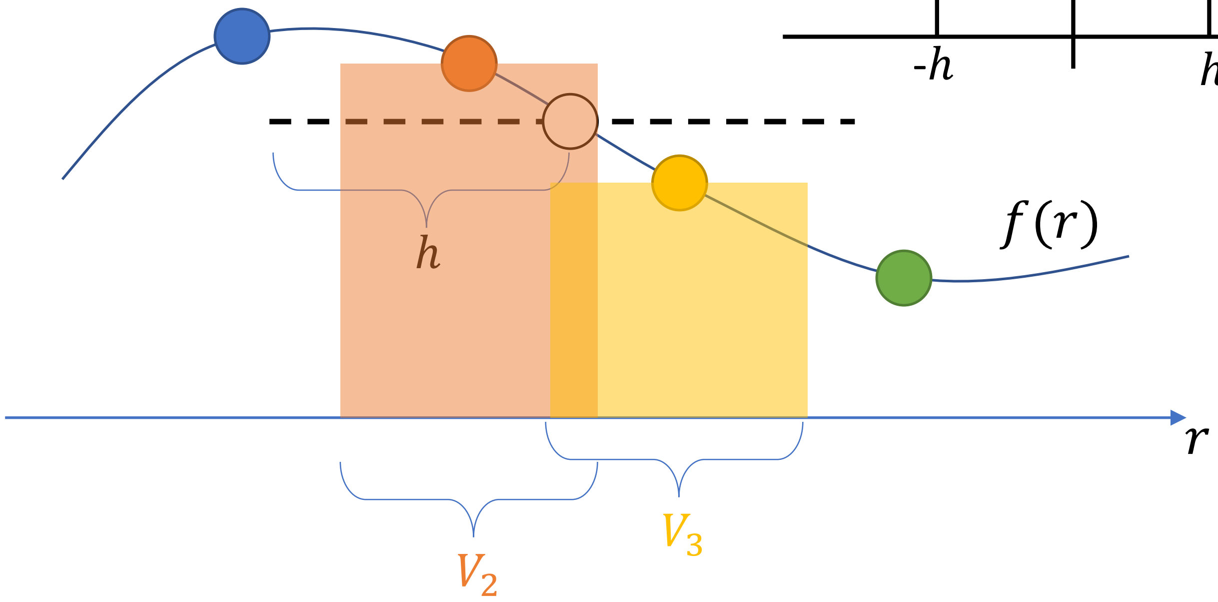


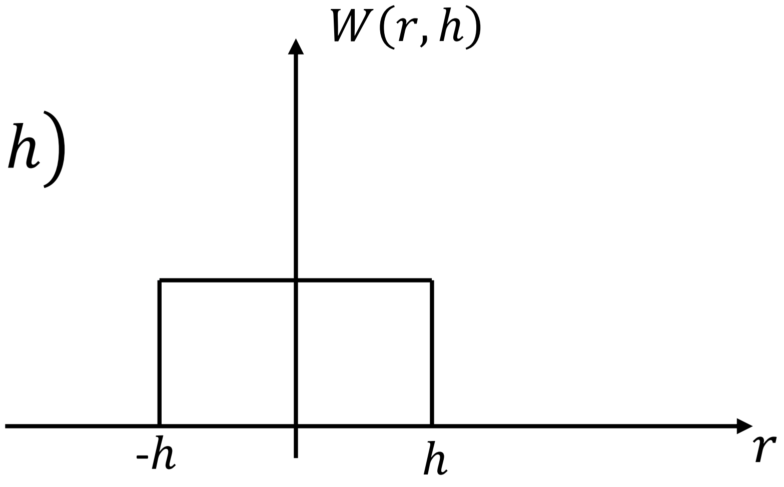
$$W(r,h) = \begin{cases} \dfrac{1}{2h}, if\ |r| < h \\ 0, otherwise \end{cases}$$

Finite probes: from integration to summation

$$f(r) \approx \int f(r')W(r-r',h)dr' \approx \sum_j V_j f(r_j)W(r-r_j,h)$$

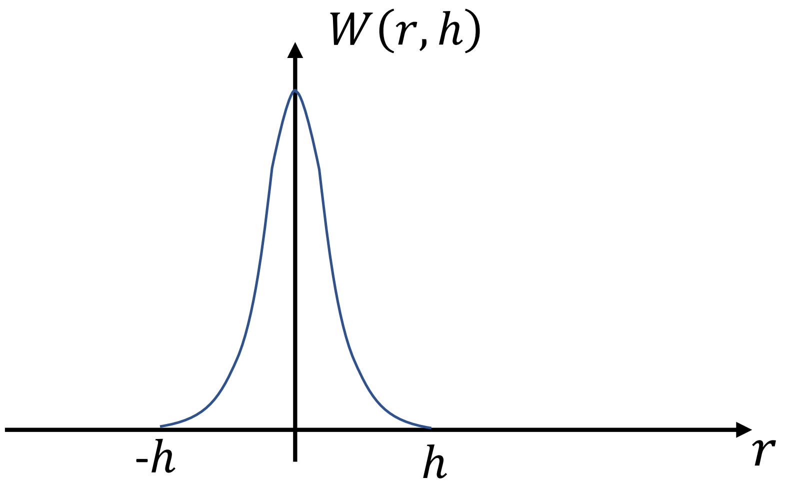


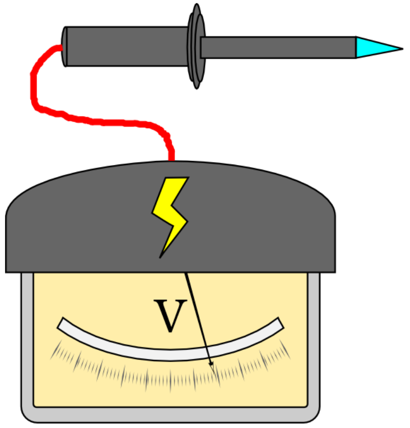


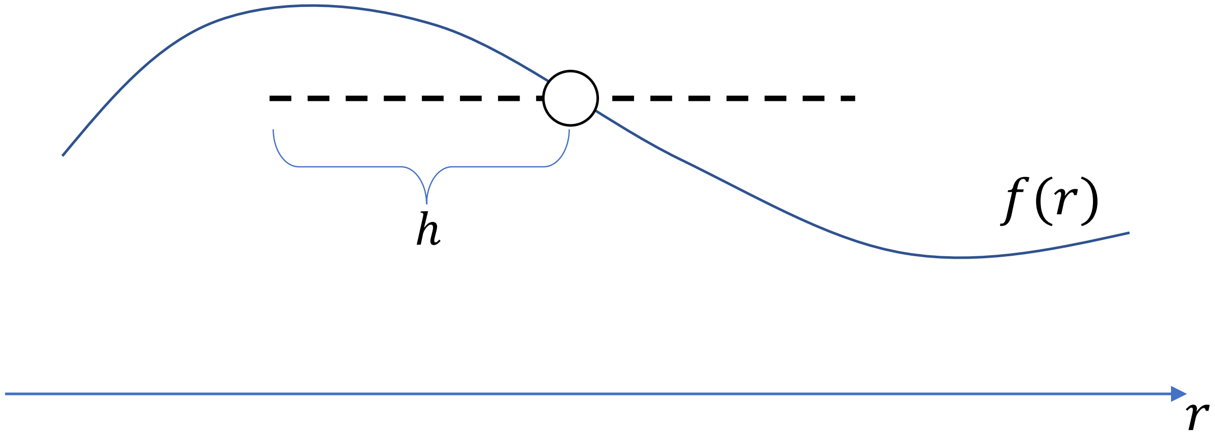


A smoother kernel function

$$f(r) \approx \sum_j V_j f(r_j) W(r - r_j, h)$$

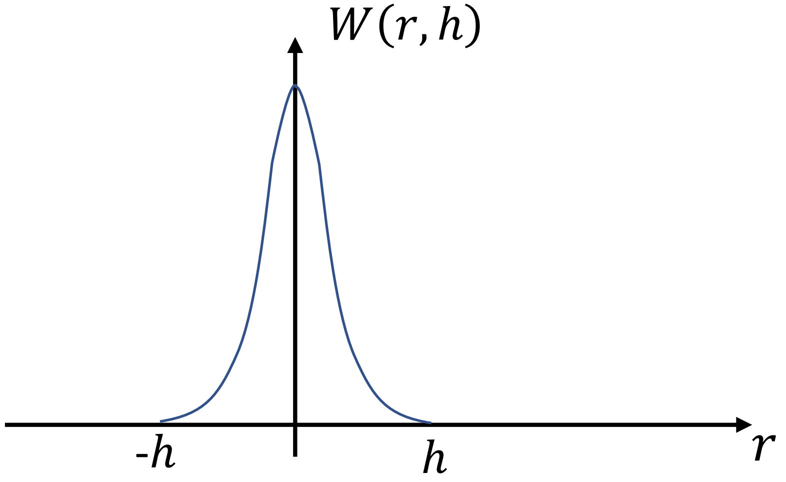






Smooth $W(r, h)$: “we trust the closer probes better”

A smoother kernel function



$$f(r) \approx \sum_j V_j f(r_j) W(r - r_j, h)$$

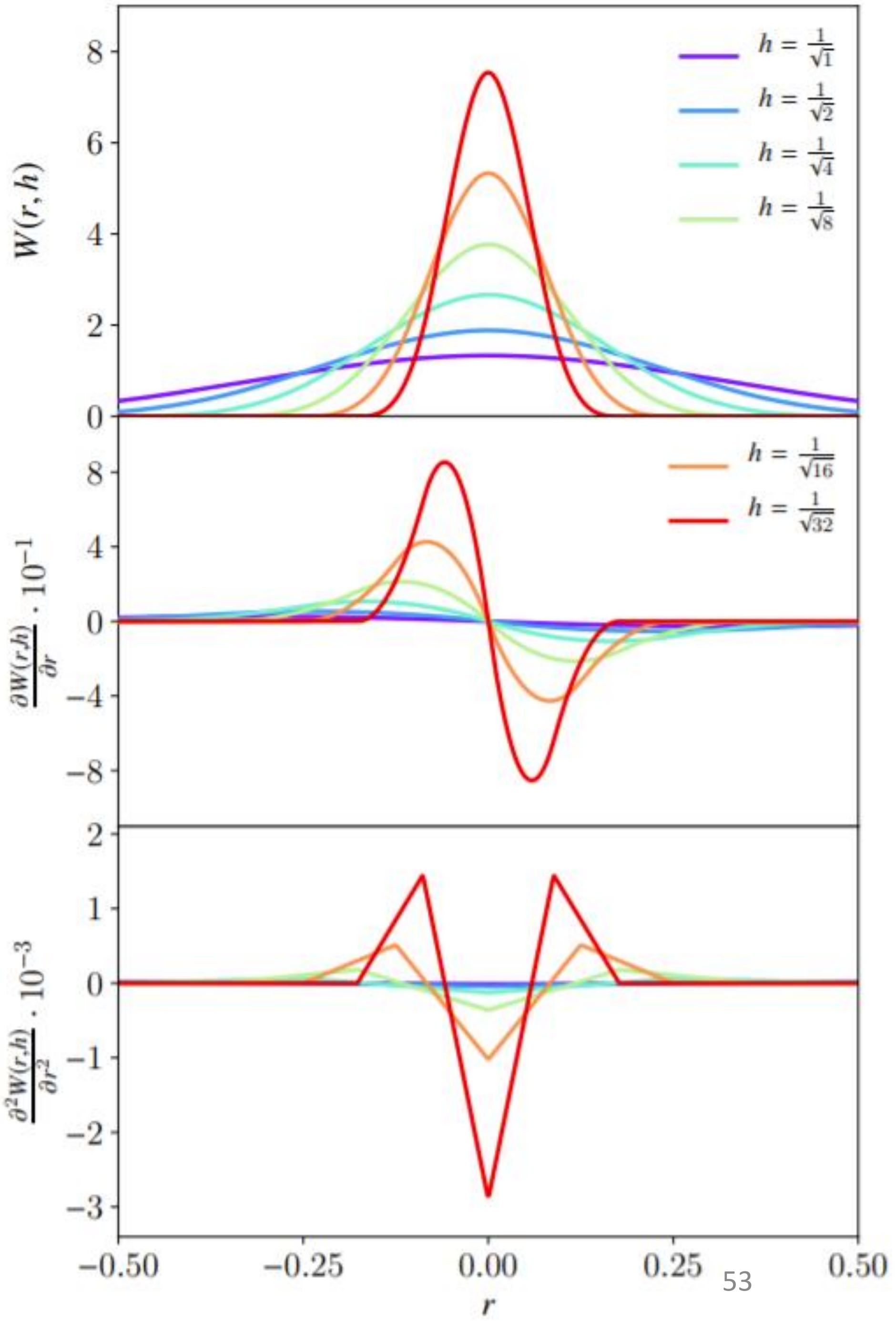
$$W(r,h) = \sigma_d \begin{cases} 6(q^3 - q^2) + 1 & for\ 0 \le q \le \frac{1}{2} \\ 2(1-q)^3 & for\ \frac{1}{2} \le q \le 1 \\ 0 & otherwise \end{cases}$$

with $q = \frac{1}{h}\|r\|, \sigma_1 = \frac{4}{3h}, \sigma_2 = \frac{40}{7\pi h^2}, \sigma_3 = \frac{8}{\pi h^3}$

A smoother kernel function

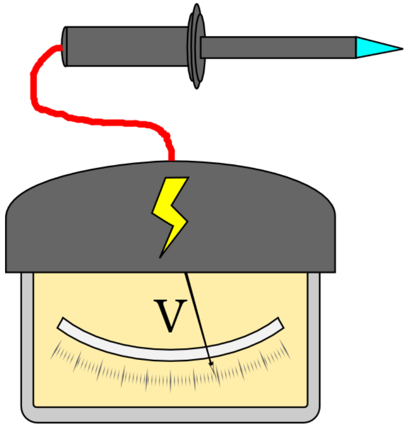
$$W(r,h) = \sigma_d \begin{cases} 6(q^3 - q^2) + 1 & for\ 0 \le q \le \frac{1}{2} \\ 2(1-q)^3 & for\ \frac{1}{2} \le q \le 1 \\ 0 & otherwise \end{cases}$$

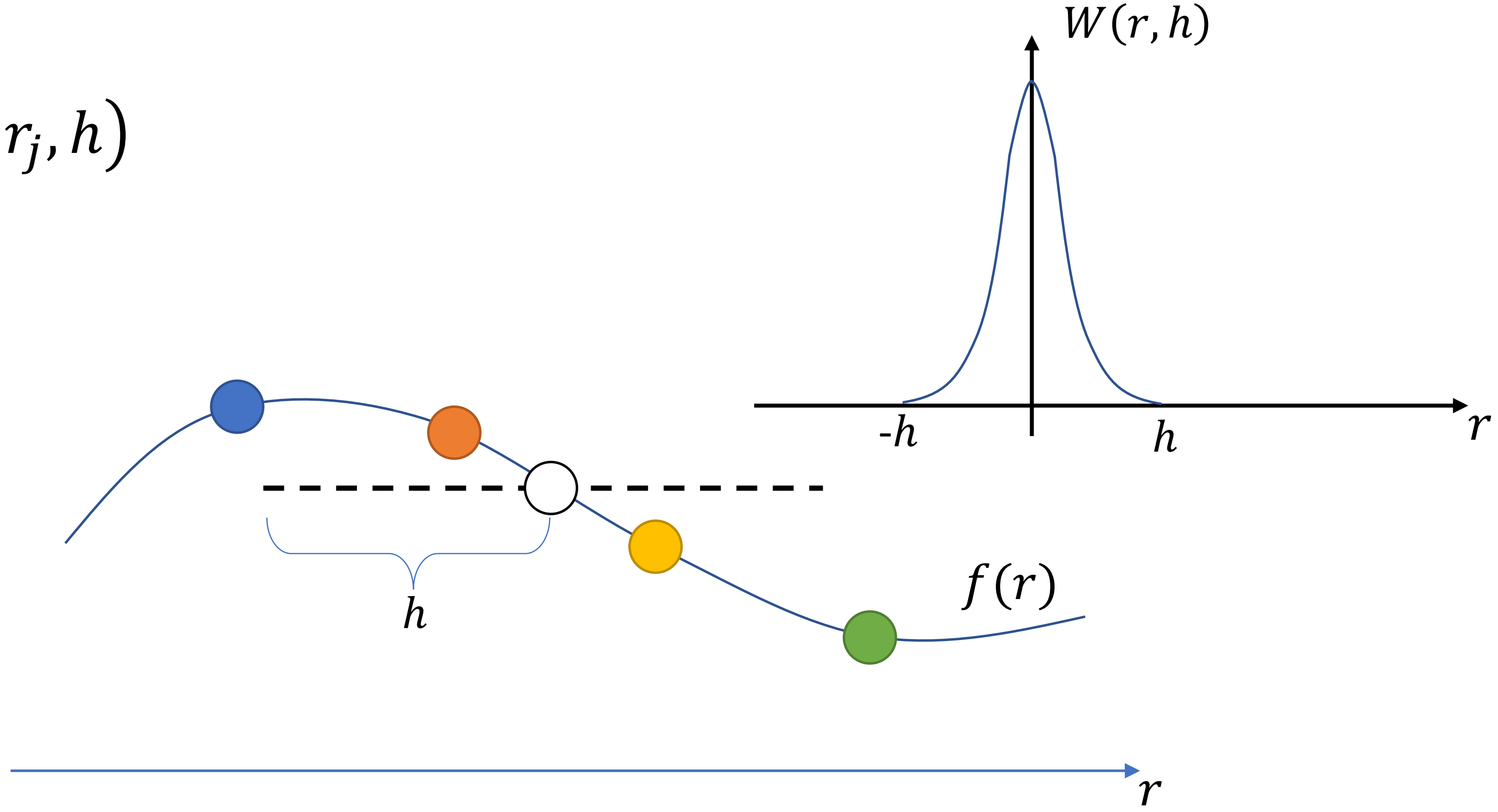
with $q = \frac{1}{h}\|r\|, \sigma_1 = \frac{4}{3h}, \sigma_2 = \frac{40}{7\pi h^2}, \sigma_3 = \frac{8}{\pi h^3}$



A Smoothed particle

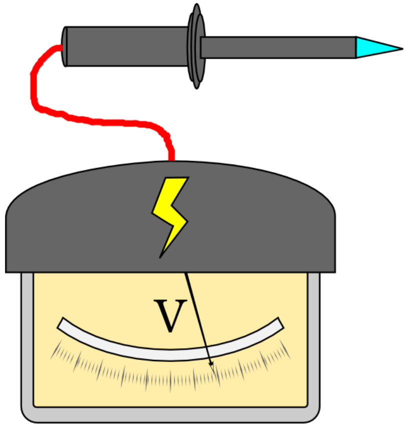
$$f(r) \approx \sum_j V_j f(r_j) W(r - r_j, h)$$

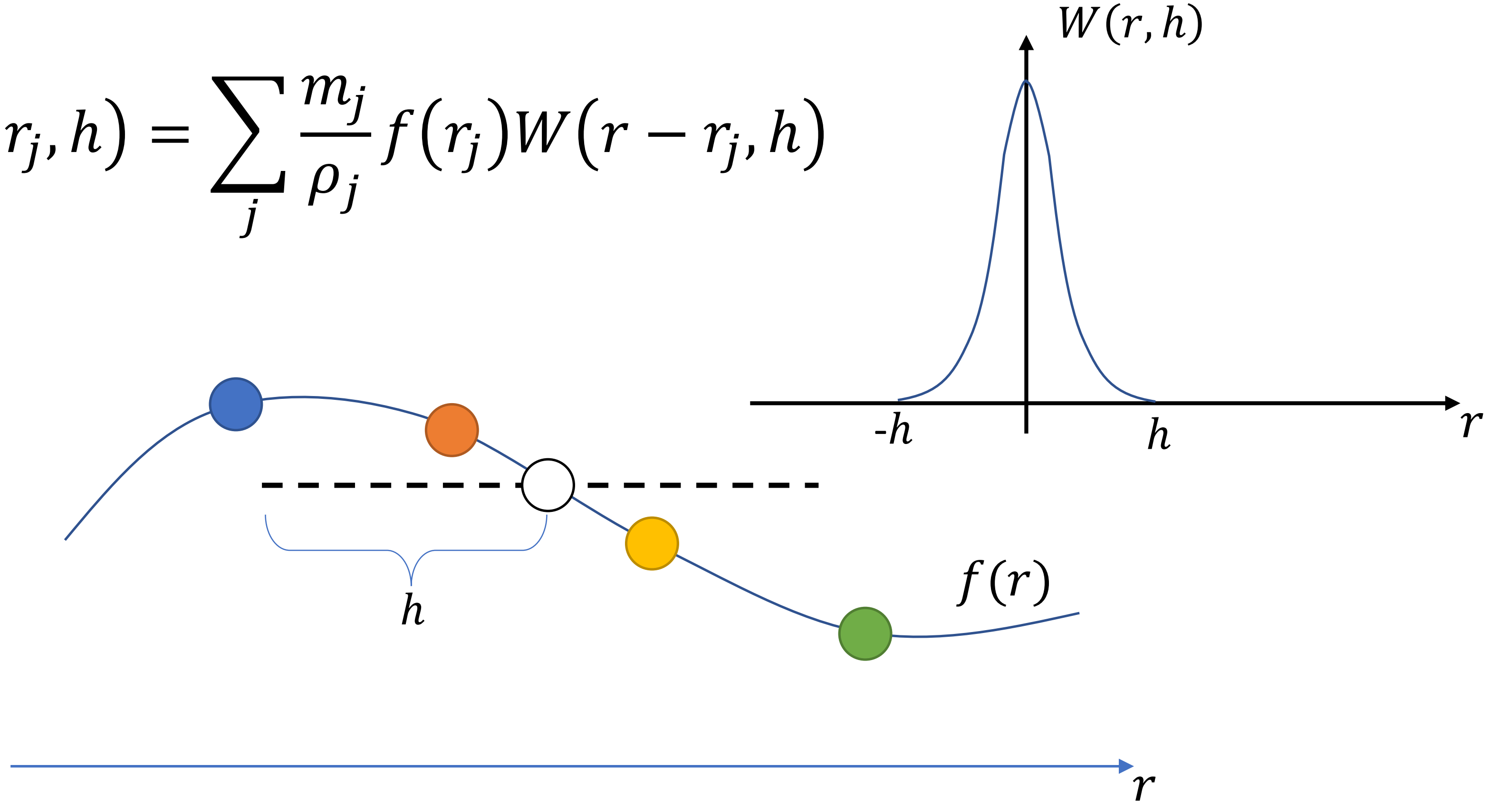




Smoothed particle hydrodynamics (SPH)

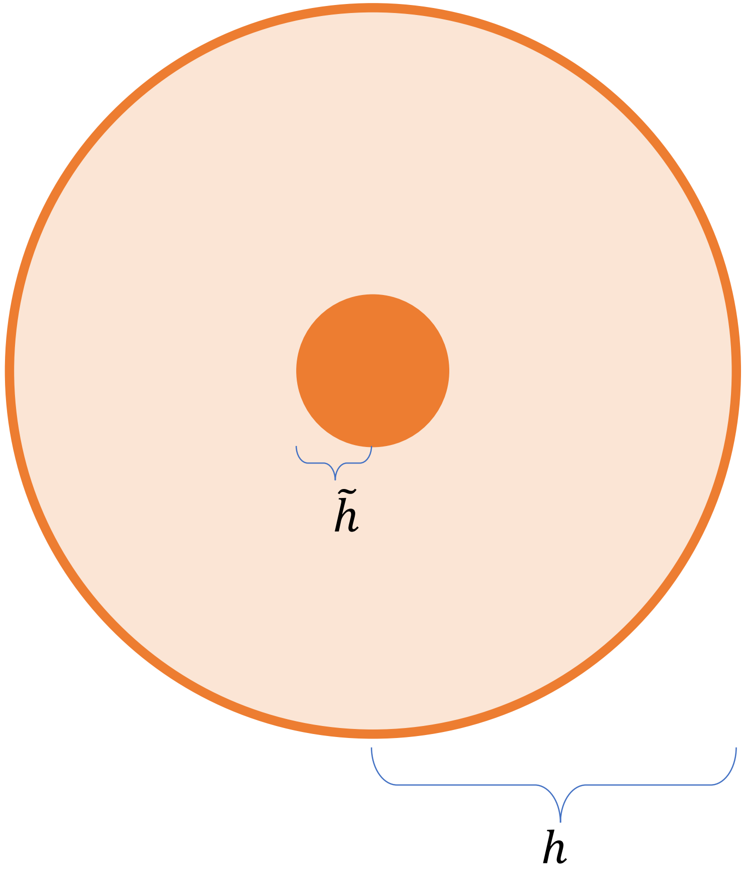
$$f(r) \approx \sum_j V_j f(r_j) W(r - r_j, h) = \sum_j \frac{m_j}{\rho_j} f(r_j) W(r - r_j, h)$$





Smoothed particle hydrodynamics: theory and application to non-spherical stars
[Gingold and Monaghan 1977][Link]

A smoothed particle in 2D



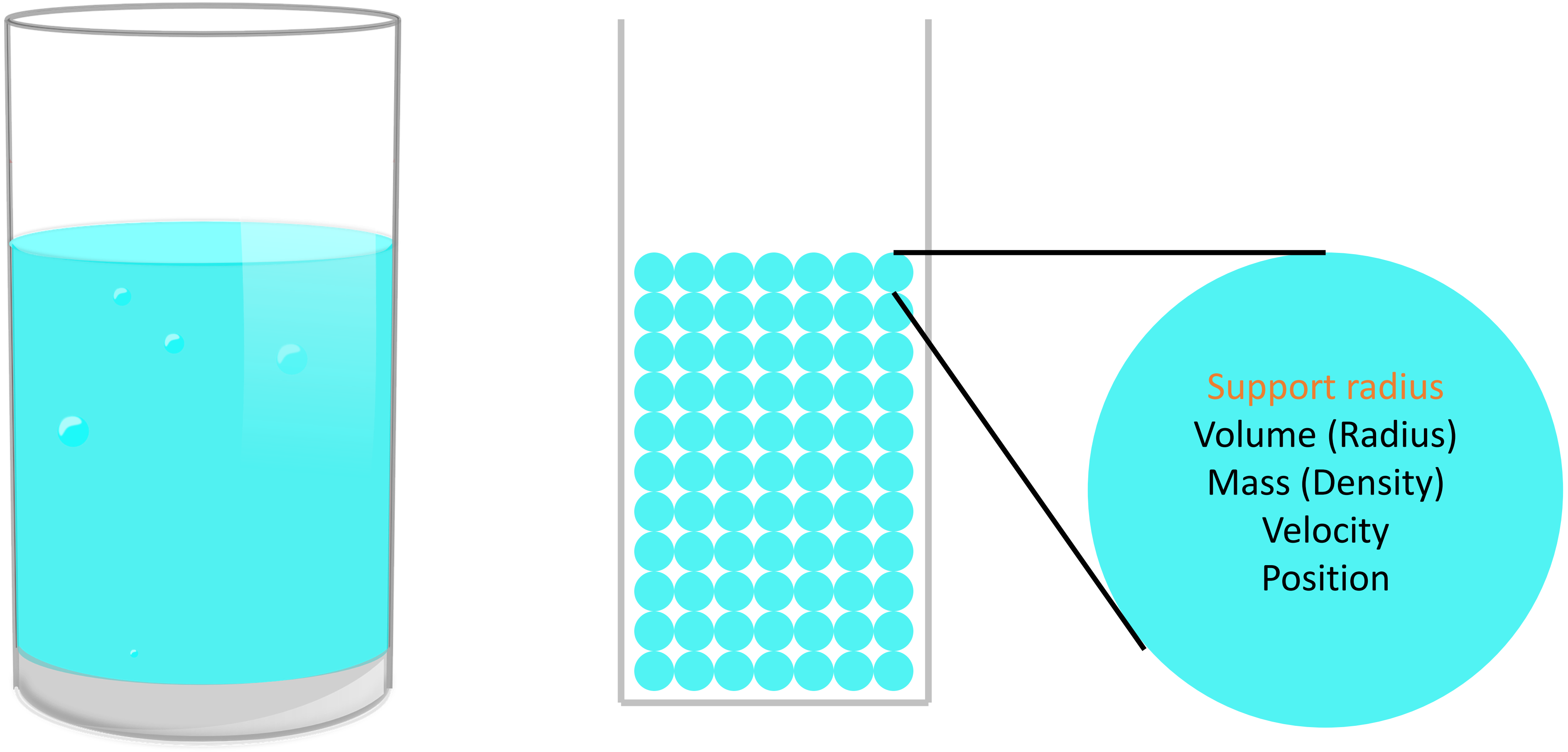
Intrinsic quantities:

- h: support radius
- $\tilde{h}$: particle radius -> V: particle volume

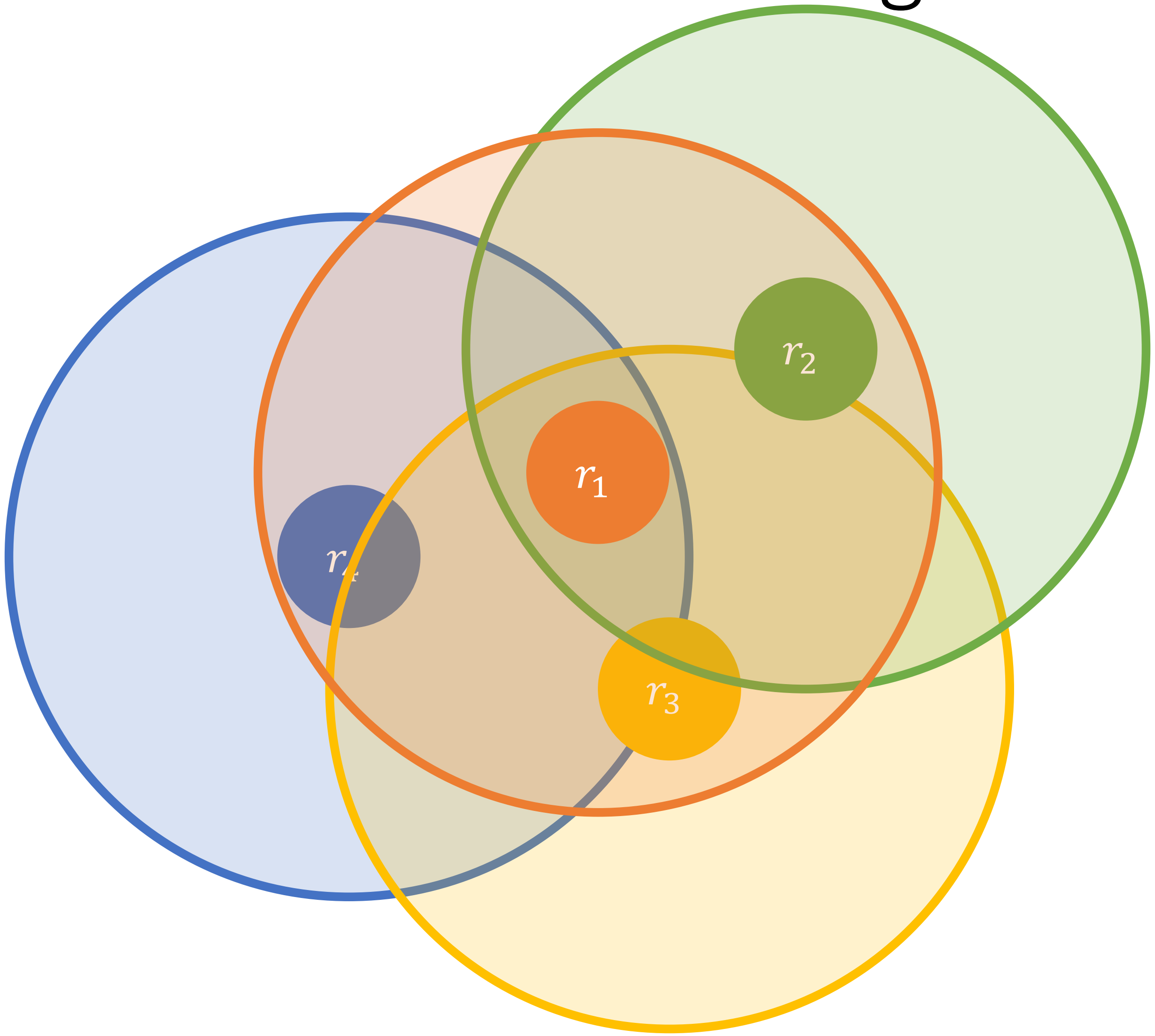
Time varying quantities:

- ρ: density
- v: velocity
- x: position

What do we store in a smoothed particle?



Evaluate 2D fields using the smoothed particles

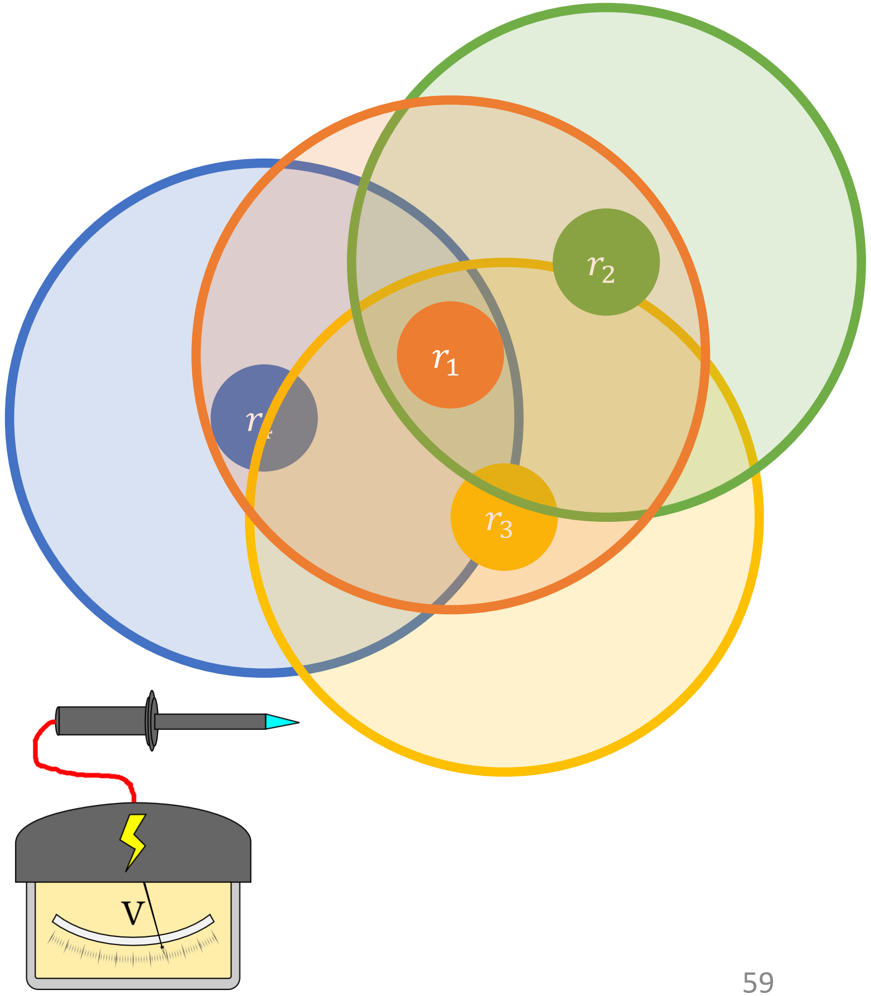


$$f(r) \approx \sum_j \frac{m_j}{\rho_j} f(r_j) W(r - r_j, h)$$

$$\begin{aligned} f(r_1) \\ &\approx \frac{m_2}{\rho_2} f(r_2) W(r_1 - r_2, h) \\ &+ \frac{m_3}{\rho_3} f(r_3) W(r_1 - r_3, h) \\ &+ \frac{m_4}{\rho_4} f(r_4) W(r_1 - r_4, h) \end{aligned}$$

Smoothed particle hydrodynamics (SPH)

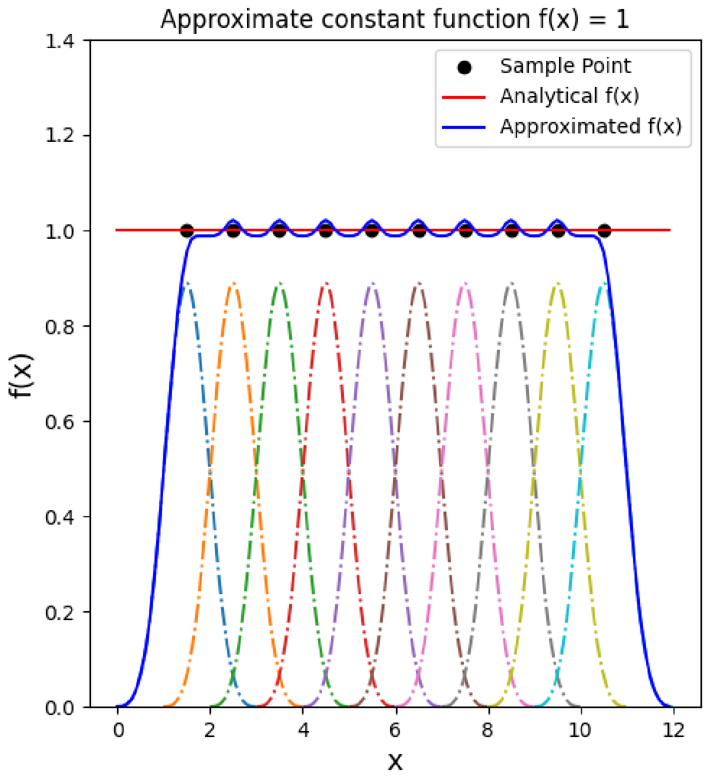
- SPH discretization:
  - $f(r) \approx \sum_j \frac{m_j}{\rho_j} f(r_j) W(r - r_j, h)$
- SPH spatial derivatives:
  - $\nabla f(r) \approx \sum_j \frac{m_j}{\rho_j} f(r_j) \nabla W(r - r_j, h)$
  - $\nabla \cdot \boldsymbol{F}(r) \approx \sum_j \frac{m_j}{\rho_j} \boldsymbol{F}(r_j) \cdot \nabla W(r - r_j, h)$
  - $\nabla \times \boldsymbol{F}(r) \approx -\sum_j \frac{m_j}{\rho_j} f(r_j) \times \nabla W(r - r_j, h)$
  - $\nabla^2 f(r) \approx \sum_j \frac{m_j}{\rho_j} f(r_j) \nabla^2 W(r - r_j, h)$



Improving approximations for spatial derivatives

- $f(r) \approx \sum_j \frac{m_j}{\rho_j} f(r_j) W(r - r_j, h)$
- $\nabla f(r) \approx \sum_j \frac{m_j}{\rho_j} f(r_j) \nabla W(r - r_j, h)$

- Let $f(r) \equiv 1$, we have:
  - $1 \approx \sum_j \frac{m_j}{\rho_j} W(r - r_j, h)$
  - $0 \approx \sum_j \frac{m_j}{\rho_j} \nabla W(r - r_j, h)$



Improving approximations for spatial derivatives

- Since $f(r) \equiv f(r) * 1$, we have:
  - $\nabla f(r) = \nabla f(r) * 1 + f(r) * \nabla 1$
  - Or equivalently: $\nabla f(r) = \nabla f(r) - f(r) * \nabla 1$

Improving approximations for spatial derivatives

- Since $f(r) \equiv f(r) * 1$, we have:
  - $\nabla f(r) = \nabla f(r) * 1 + f(r) * \nabla 1$
  - Or equivalently: $\nabla f(r) = \nabla f(r) - f(r) * \nabla 1$

$$\nabla f(r) \approx \sum_j \frac{m_j}{\rho_j} f(r_j) \nabla W(r - r_j, h)$$

Improving approximations for spatial derivatives

- Since $f(r) \equiv f(r) * 1$, we have:
  - $\nabla f(r) = \nabla f(r) * 1 + f(r) * \nabla 1$
  - Or equivalently: $\nabla f(r) = \nabla f(r) - f(r) * \nabla 1$

- $\nabla f(r) \approx \sum_j \frac{m_j}{\rho_j} f(r_j) \nabla W(r - r_j, h) - f(r) \sum_j \frac{m_j}{\rho_j} \nabla W(r - r_j, h)$

Improving approximations for spatial derivatives

- Since $f(r) \equiv f(r) * 1$, we have:
  - $\nabla f(r) = \nabla f(r) * 1 + f(r) * \nabla 1$
  - Or equivalently: $\nabla f(r) = \nabla f(r) - f(r) * \nabla 1$

- $\nabla f(r) \approx \sum_j \frac{m_j}{\rho_j} f(r_j) \nabla W(r - r_j, h) - f(r) \sum_j \frac{m_j}{\rho_j} \nabla W(r - r_j, h)$
- $\boxed{\nabla f(r) \approx \sum_j m_j \frac{f(r_j) - f(r)}{\rho_j} \nabla W(r - r_j, h)}$

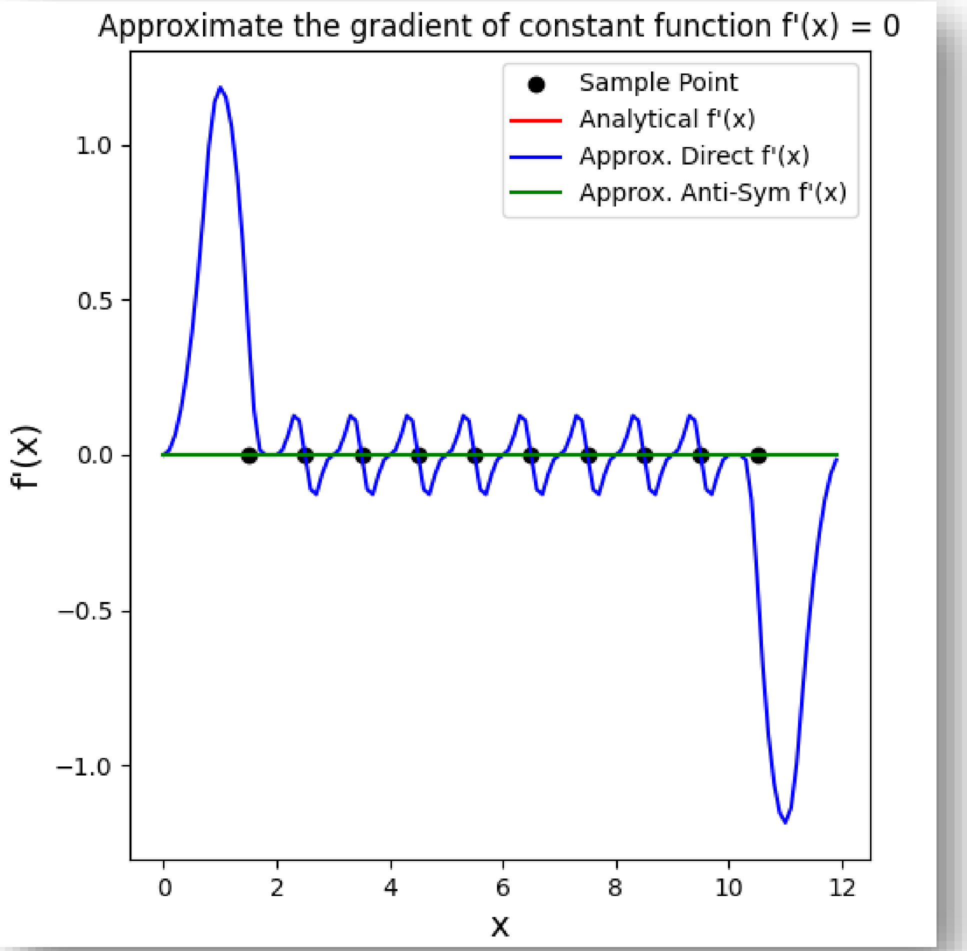
The anti-symmetric form

Improving approximations for spatial derivatives

- Since $f(r) \equiv f(r) * 1$, we have:
  - $\nabla f(r) = \nabla f(r) * 1 + f(r) * \nabla 1$
  - Or equivalently: $\nabla f(r) = \nabla f(r) - f(r) * \nabla 1$

- $\nabla f(r) \approx \sum_j \frac{m_j}{\rho_j} f(r_j) \nabla W(r - r_j, h) - f(r) \sum$
- $\boxed{\nabla f(r) \approx \sum_j m_j \frac{f(r_j) - f(r)}{\rho_j} \nabla W(r - r_j, h)}$

The anti-symmetric form



Improving approximations for spatial derivatives

- A more general case:
  - $\nabla(f(r)\rho^n) = \nabla f(r) * \rho^n + f(r) * n\rho^{n-1}\nabla\rho$
  - $\Rightarrow \nabla f(r) = \frac{1}{\rho^n}(\nabla(f(r) * \rho^n) - f(r) * n * \rho^{n-1}\nabla\rho)$

Improving approximations for spatial derivatives

- A more general case:
  - $\nabla(f(r)\rho^n) = \nabla f(r) * \rho^n + f(r) * n\rho^{n-1}\nabla\rho$
  - $\Rightarrow \nabla f(r) = \frac{1}{\rho^n}(\nabla(f(r) * \rho^n) - f(r) * n * \rho^{n-1}\nabla\rho)$

$$\nabla f(r) \approx \sum_j \frac{m_j}{\rho_j} f(r_j) \nabla W(r - r_j, h)$$

Improving approximations for spatial derivatives

- A more general case:
  - $\nabla(f(r)\rho^n) = \nabla f(r) * \rho^n + f(r) * n\rho^{n-1}\nabla\rho$
  - $\Rightarrow \nabla f(r) = \frac{1}{\rho^n}(\nabla(f(r) * \rho^n) - f(r) * n * \rho^{n-1}\nabla\rho)$

  - $\nabla f(r) \approx \frac{1}{\rho^n}\left(\sum_j \frac{m_j}{\rho_j} f(r_j)\rho_j^n \nabla W(r - r_j, h) - n\rho^{n-1} f(r) \sum_j \frac{m_j}{\rho_j} \rho_j \nabla W(r - r_j, h)\right)$

Improving approximations for spatial derivatives

- A more general case:
  - $\nabla(f(r)\rho^n) = \nabla f(r) * \rho^n + f(r) * n\rho^{n-1}\nabla\rho$
  - $\Rightarrow \nabla f(r) = \frac{1}{\rho^n}(\nabla(f(r) * \rho^n) - f(r) * n * \rho^{n-1}\nabla\rho)$

  - $\nabla f(r) \approx \frac{1}{\rho^n}\left(\sum_j \frac{m_j}{\rho_j} f(r_j)\rho_j^n \nabla W(r - r_j, h) - n\rho^{n-1} f(r) \sum_j \frac{m_j}{\rho_j}\rho_j \nabla W(r - r_j, h)\right)$
  - $\nabla f(r) \approx \sum_j m_j \left(\frac{f(r_j)\rho_j^{n-1}}{\rho^n} - \frac{nf(r)}{\rho}\right)\nabla W(r - r_j, h)$

Improving approximations for spatial derivatives

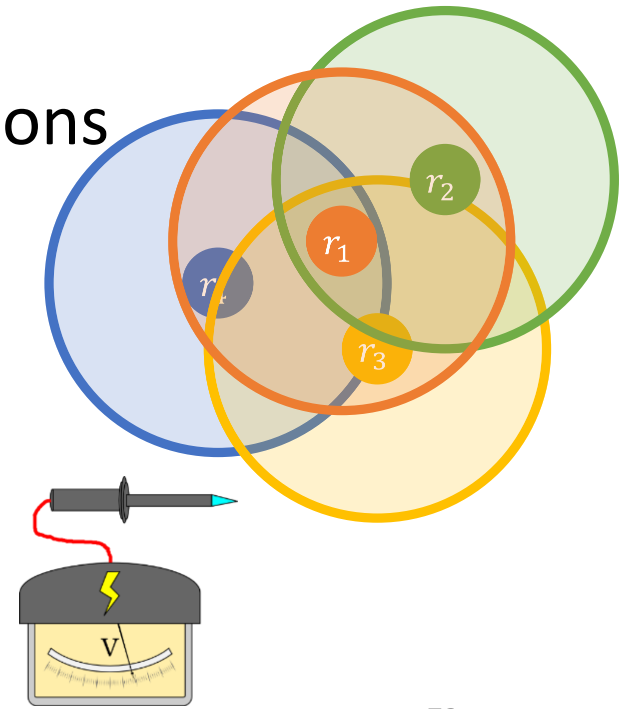
- $\nabla f(r) \approx \sum_j m_j \left(\frac{f(r_j)\rho_j^{n-1}}{\rho^n} - \frac{nf(r)}{\rho} \right) \nabla W(r - r_j, h)$
- When $n = -1$
  - Type equation here.
  - $\boxed{\nabla f(r) \approx \rho \sum_j m_j \left(\frac{f(r_j)}{\rho_j^2} + \frac{f(r)}{\rho^2} \right) \nabla W(r - r_j, h)}$

The symmetric form

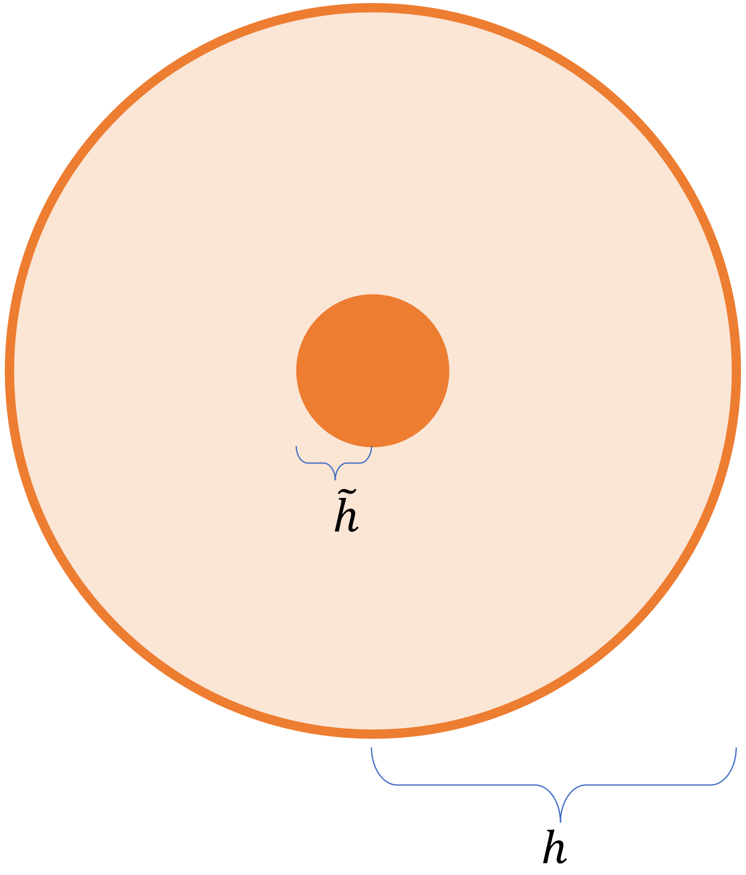
Smoothed particle hydrodynamics (SPH)

$$f(r) \approx \sum_j \frac{m_j}{\rho_j} f(r_j) W(r - r_j, h)$$

- Approximate a function $f(r)$ using finite probes $f(r_j)$
- The degree of freedom (r) goes inside the kernel functions
  - Anti-sym: $\nabla f(r) \approx \sum_j m_j \frac{f(r_j) - f(r)}{\rho_j} \nabla W(r - r_j, h)$
  - Sym: $\nabla f(r) \approx \rho \sum_j m_j \left(\frac{f(r_j)}{\rho_j^2} + \frac{f(r)}{\rho^2} \right) \nabla W(r - r_j, h)$



Quiz: which one is the smoothed particle in SPH?



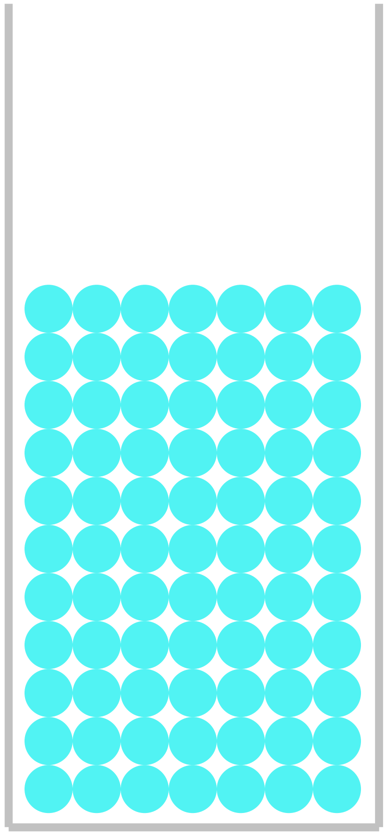
The bigger circle?
or
The smaller circle?

Intrinsic quantities:
- h: support radius
- $\tilde{h}$: particle radius -> V: particle volume

Time varying quantities:
- ρ: density
- v: velocity
- x: position

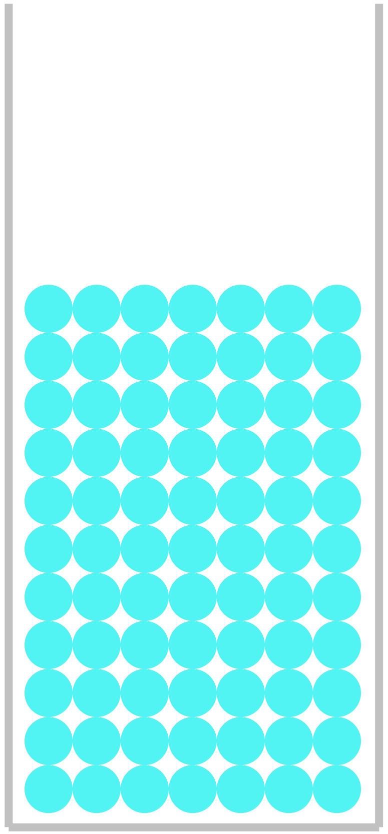
Implementation details (WCSPH)

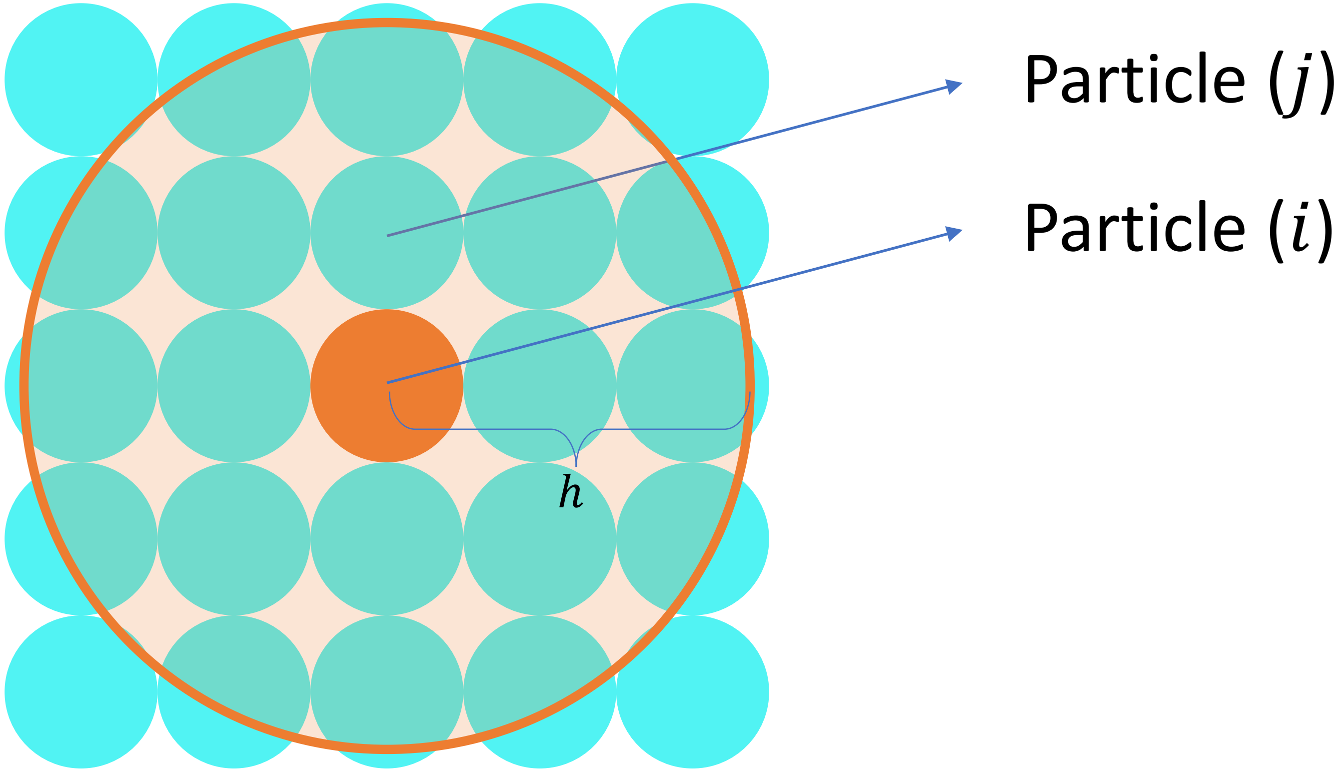
Simulation pipeline



- for i in particles:
  - Search for neighbors j
- for i in particles:
  - Sample the velocity/density field using SPH
  - Compute force/acceleration using Navier-Stokes equation
- for i in particles:
  - Update velocity using acceleration
  - Update position using velocity

Find a particle of interest (i) and its neighbors (j) within its support radius h





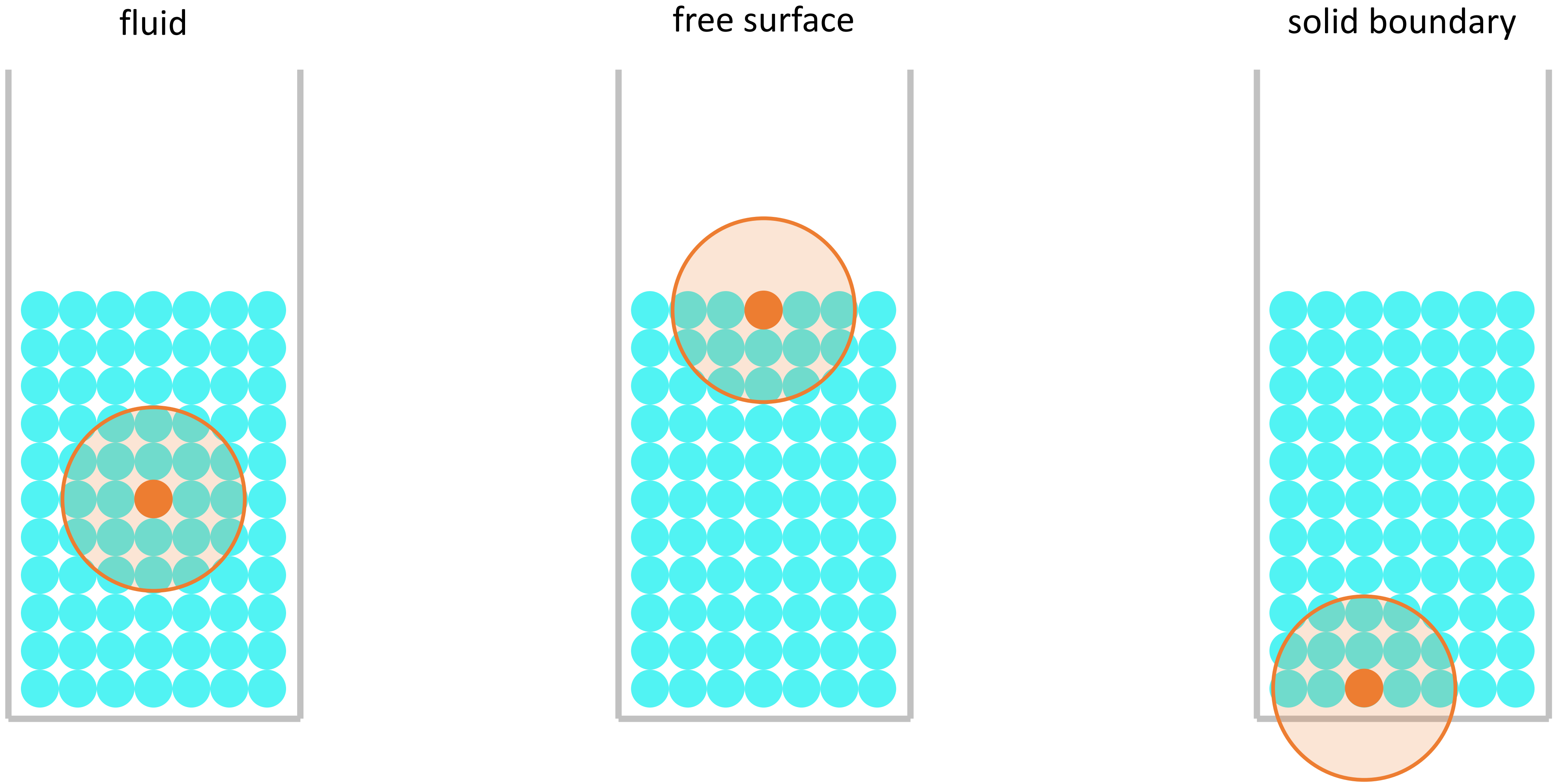
Compute the acceleration for particle ($\boldsymbol{i}$)

- for i in particles:
  - Step 1: Evaluate density
    - $\rho_i = \sum_j \frac{m_j}{\rho_j} \rho_j W(r_i - r_j, h) = \sum_j m_j W_{ij}$
  - Step 2: Evaluate viscosity
    - $\nu \nabla^2 v_i = \nu \sum_j m_j \frac{v_j - v_i}{\rho_j} \nabla^2 W_{ij}$
  - Step 3: Evaluate pressure gradient
    - $-\frac{1}{\rho_i} \nabla p_i = -\frac{\rho_i}{\rho_i} \sum_j m_j \left(\frac{p_j}{\rho_j^2} + \frac{p_i}{\rho_i^2} \right) \nabla W_{ij}, where\ p = k(\rho_j - \rho_0)$
  - $\frac{dv_i}{dt} = g - \frac{1}{\rho_i} \nabla p_i + \nu \nabla^2 v_i$

Time integration (Symplectic Euler)

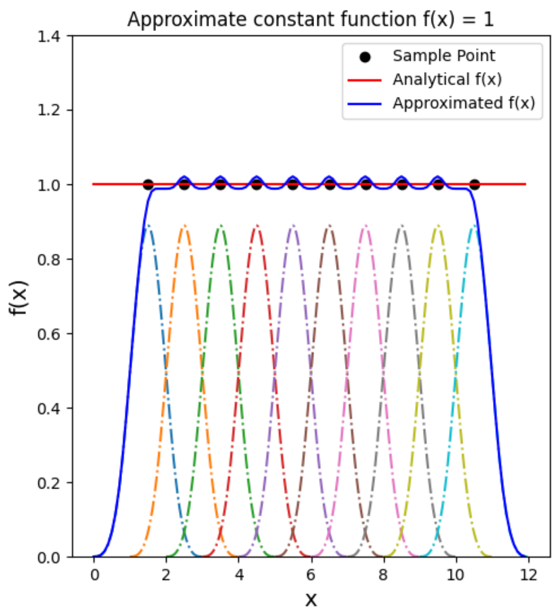
- for i in particles:
  - $v_i = v_i + \Delta t * \frac{dv_i}{dt}$
  - $x_i = x_i + \Delta t * v_i$

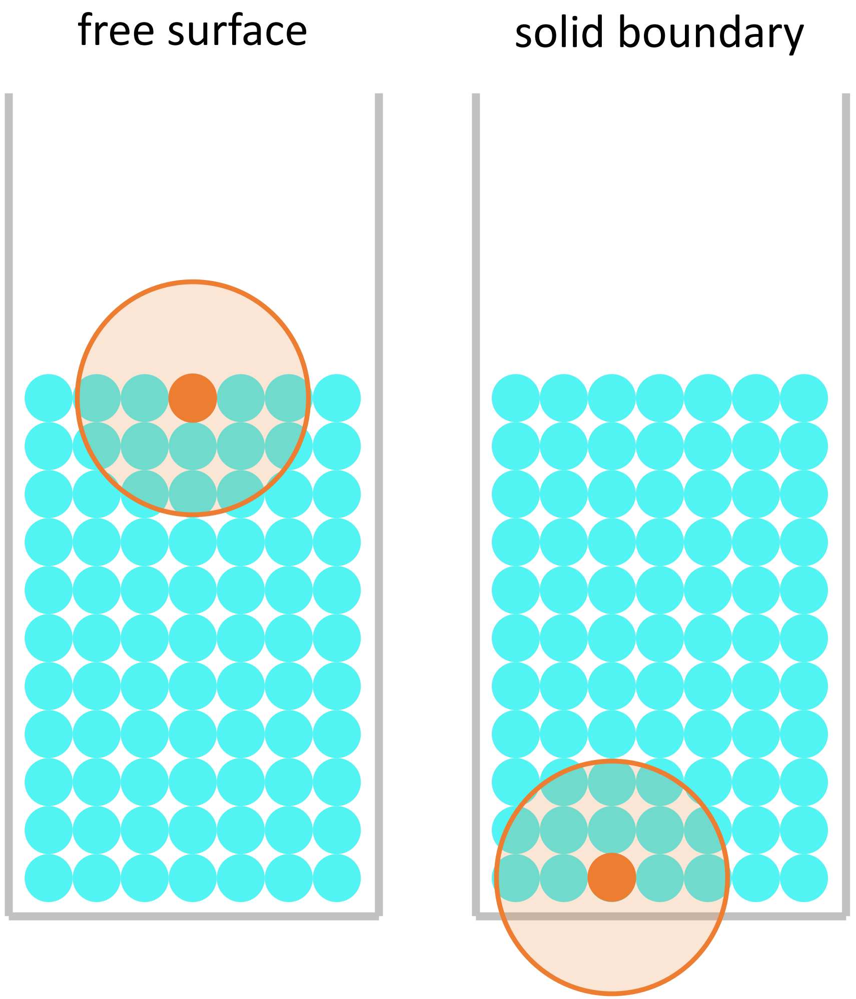
Boundary conditions



Problems of boundaries

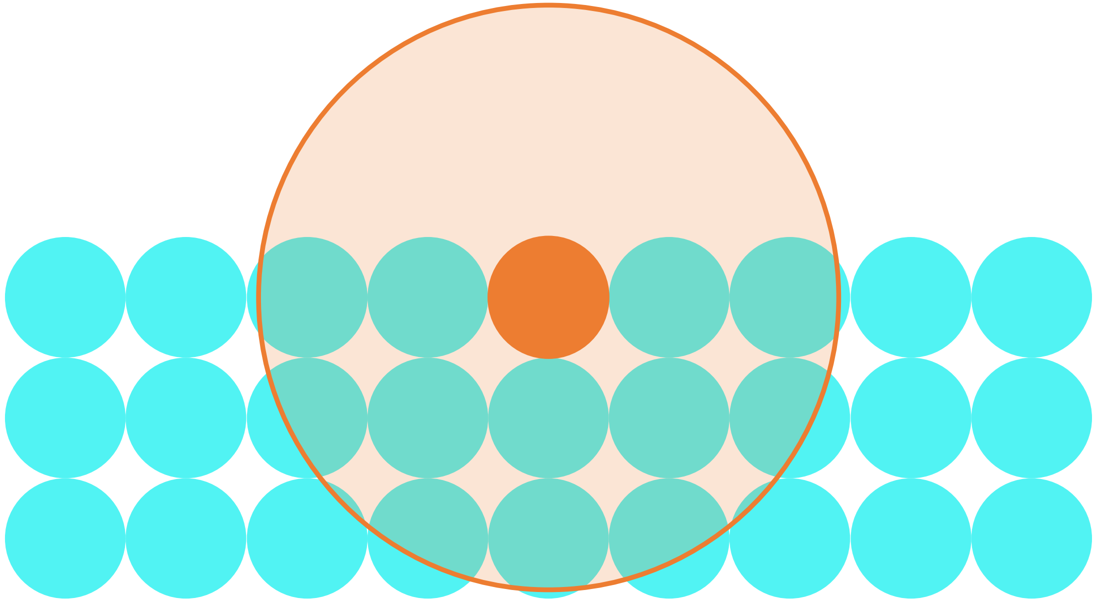
- Not enough samples within the supporting radius
  - Density: ↓
    - $\rho_i = \sum_j m_j W_{ij}$





Free surface

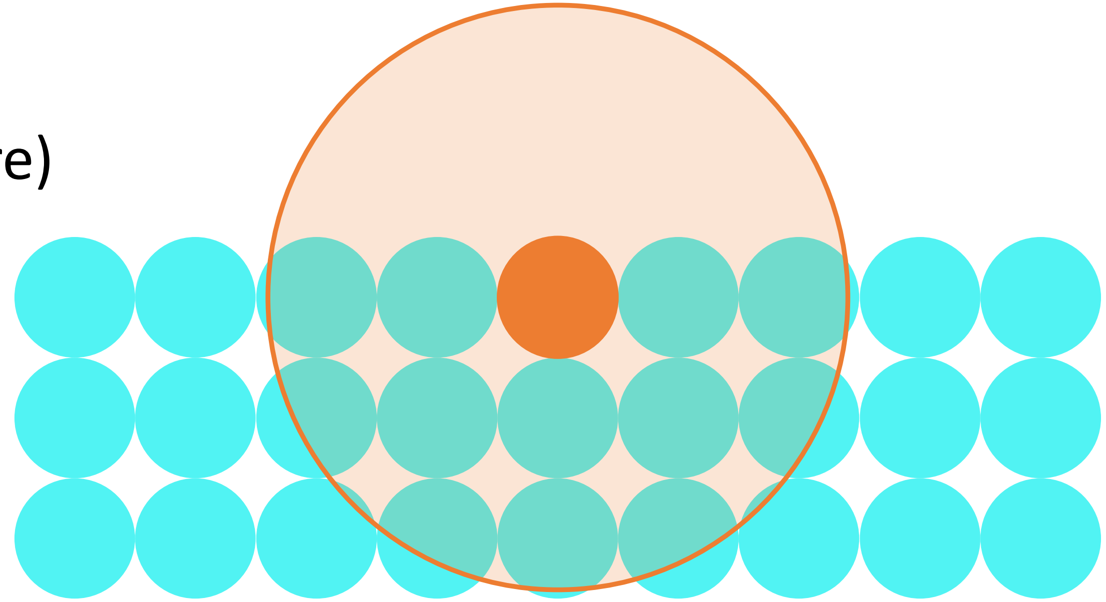
- Problem:
  - Density ↓ Pressure ↓
  - Generate outward pressure



Free surface

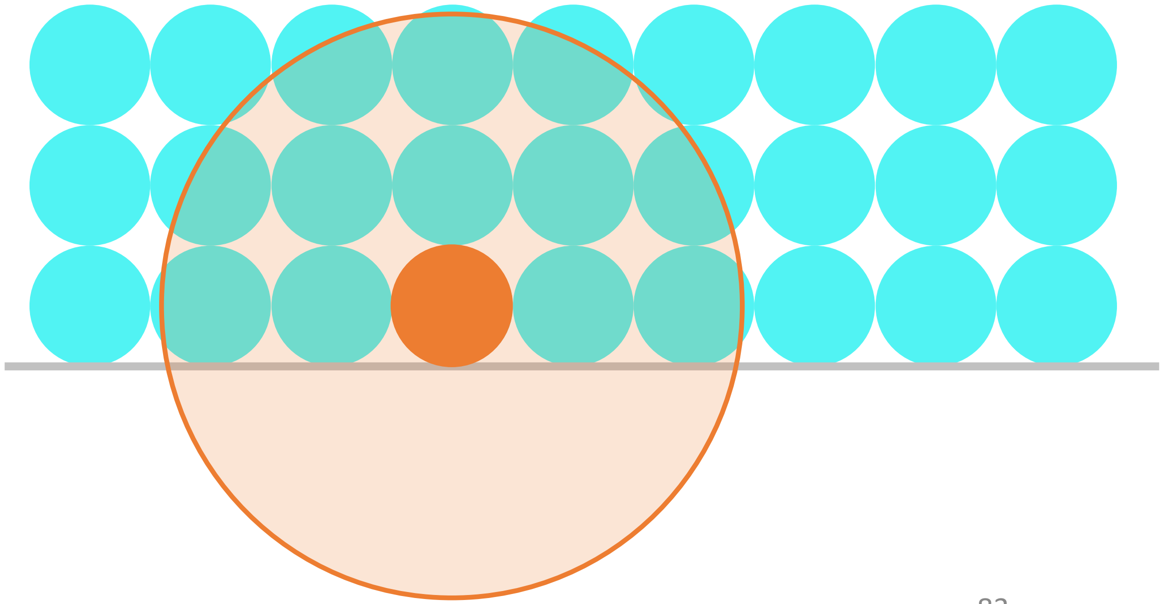
- Problem:
  - Density ↓ Pressure ↓
  - Generate outward pressure

- Solution:
  - Clamp the negative pressure (everywhere)
  - $p = \max\big(0, k(\rho - \rho_0)\big)$



Solid boundary

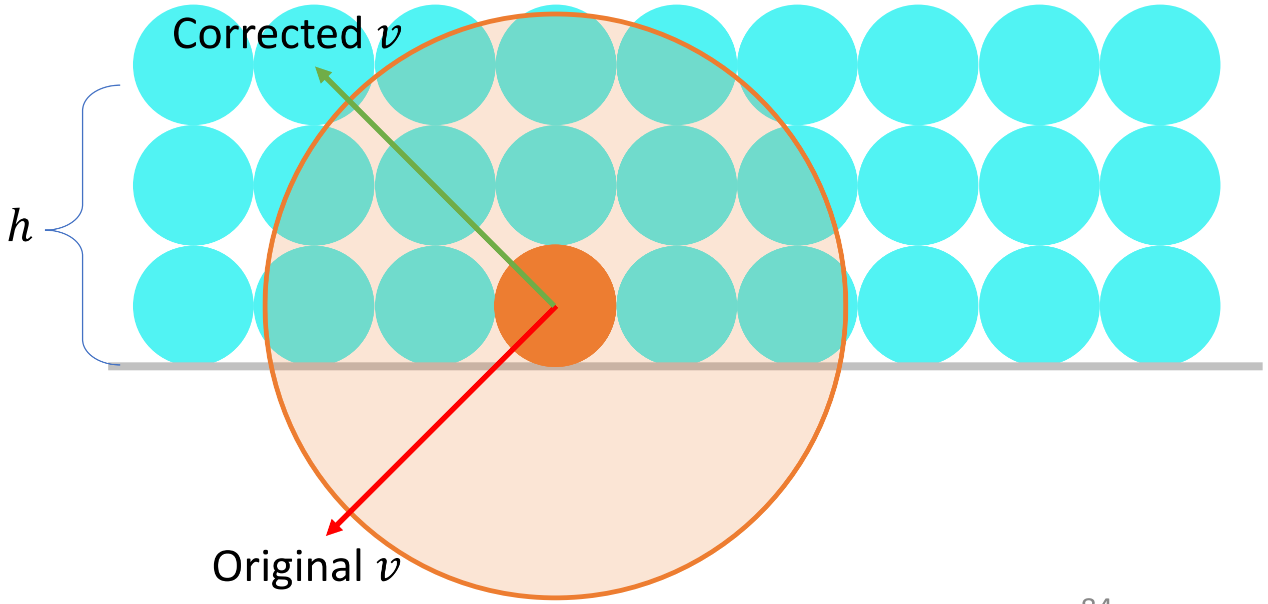
- Problem:
  - Density ↓ Pressure ↓
  - Fluid leakage (due to outbound velocity)



Solid boundary

- Problem:
  - Density ↓ Pressure ↓
  - Fluid leakage (due to outbound velocity)

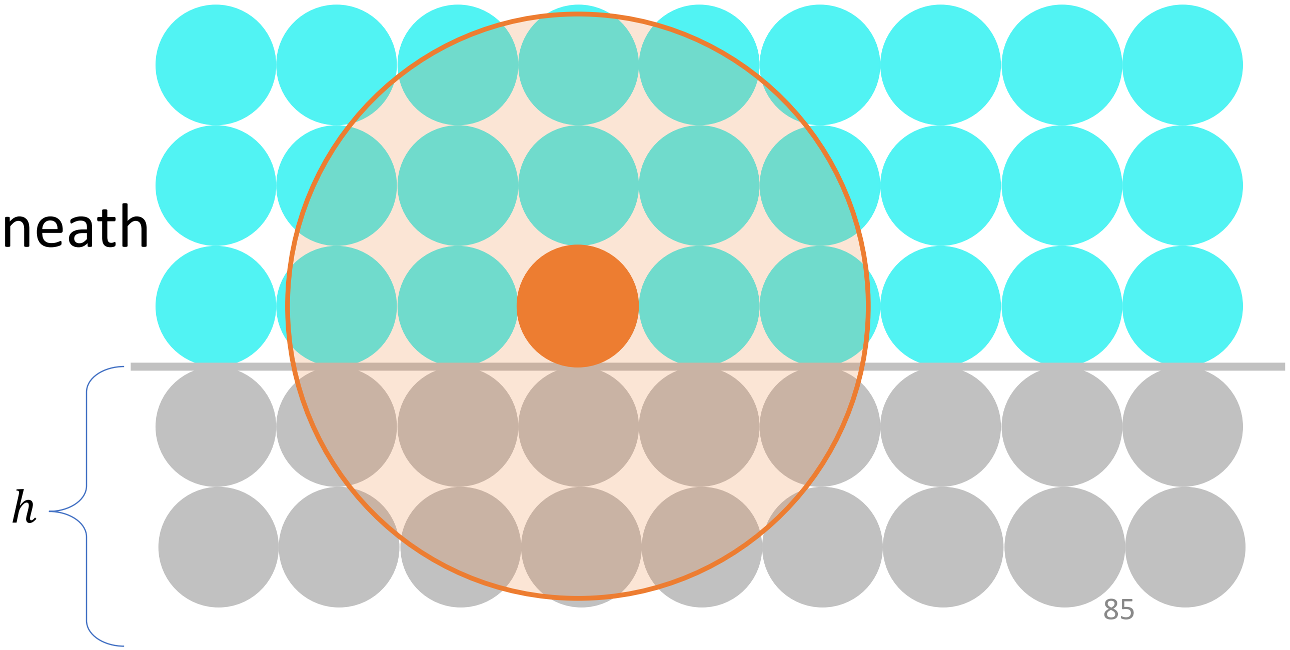
- $p = \max\big(0, k(\rho - \rho_0)\big)$
- Solution 1 for leakage:
  - Reflect the outbound velocity when close to boundary



Solid boundary

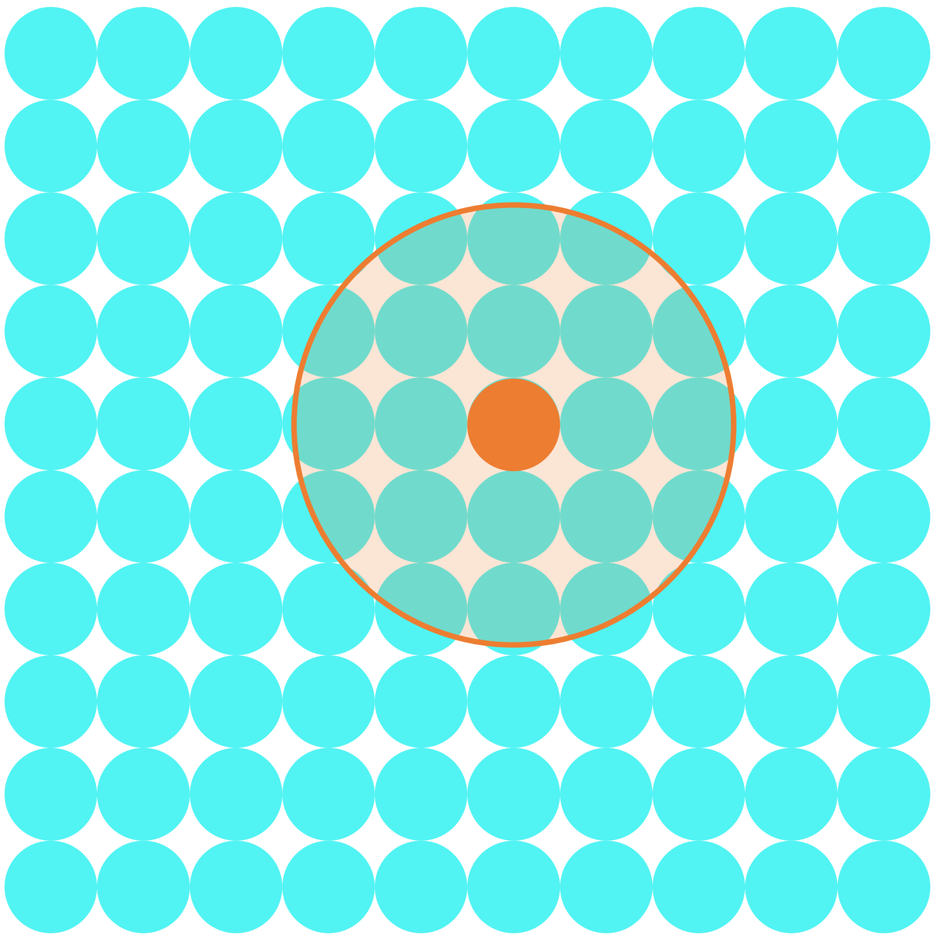
- Problem:
  - Density ↓ Pressure ↓
  - Fluid leakage (due to outbound velocity)

- $p = \max\left(0, k(\rho - \rho_0)\right)$
- Solution 2 for leakage:
  - Pad a layer of solid particles underneath the boundaries
    - $\rho_{solid} = \rho_0$
    - $v_{solid} = 0$



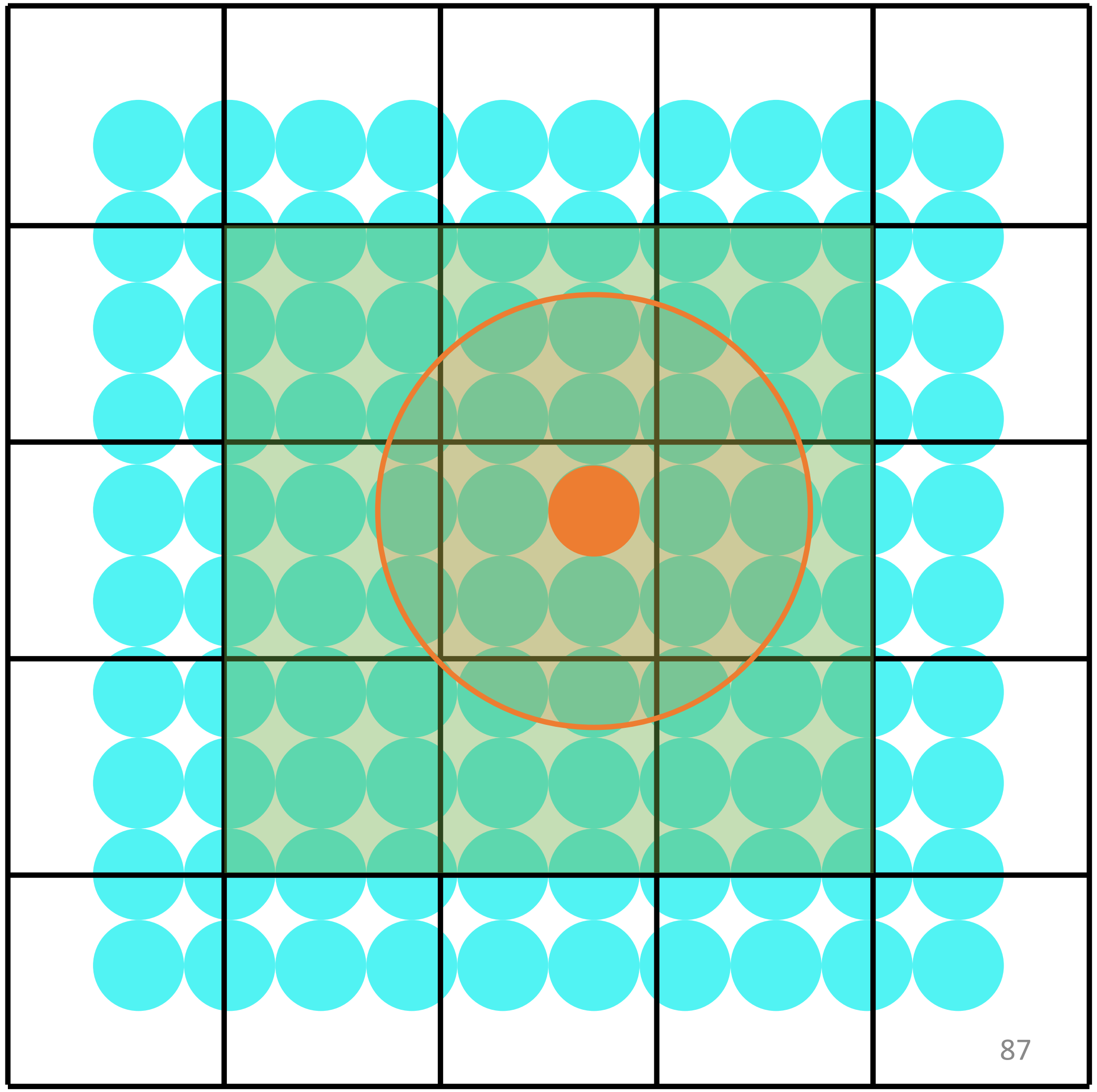
Neighbor search

- Naïve search methods takes $\mathcal{O}(n^2)$ time



Neighbor search

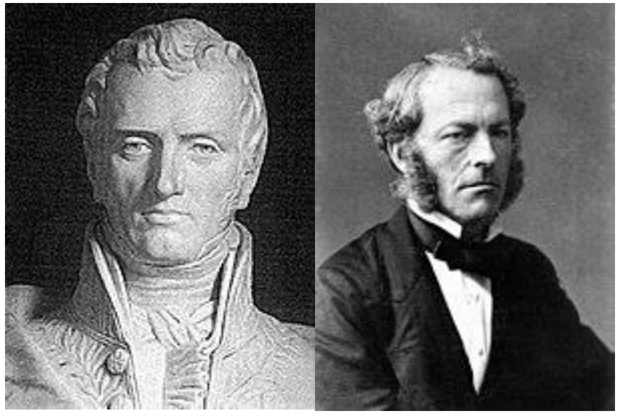
- Naïve search methods takes $\mathcal{O}(n^2)$ time
- A background grid can help
  - Common grid size = h (the support radius in SPH)
  - Each neighbor search takes 9 grids in 2D and 27 grids in 3D



Remark

Remark

- Incompressible fluid dynamics
  - Incompressible Navier–Stokes equations
- Time discretization
  - Operator splitting
  - Integration with the weakly compressible assumption
- Spatial discretization
  - Smoothed particle hydrodynamics (SPH)
- Implementation details (WCSPH)
  - Simulation Pipeline
  - Boundary conditions
  - Neighbor search



$$\rho \frac{Dv}{Dt} = \rho g - \nabla p + \mu \nabla^2 v$$

$$\nabla \cdot v = 0$$

Remark

- Incompressible fluid dynamics
  - Incompressible Navier–Stokes equations
- Time discretization
  - Operator splitting
  - Integration with the weakly compressible assumption
- Spatial discretization
  - Smoothed particle hydrodynamics (SPH)
- Implementation details (WCSPH)
  - Simulation Pipeline
  - Boundary conditions
  - Neighbor search

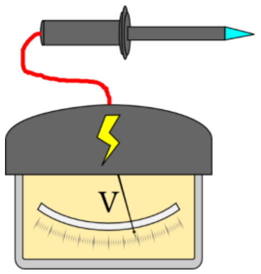
$$\rho \frac{Dv}{Dt} = \rho g + \mu \nabla^2 v$$

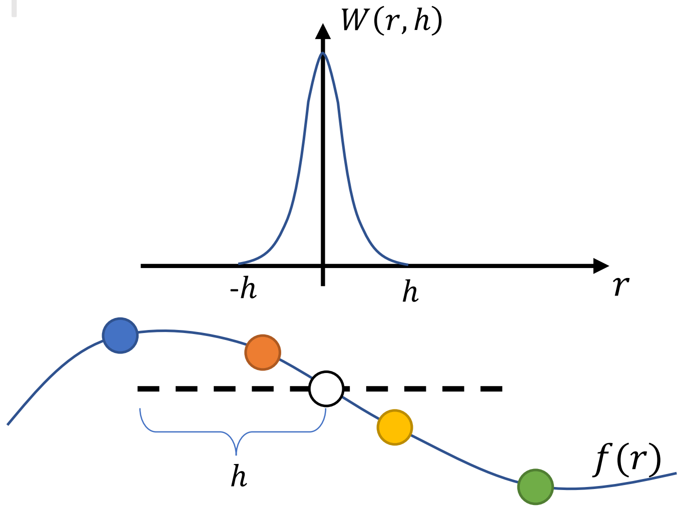
$$\rho \frac{Dv}{Dt} = -\nabla p$$
$$\nabla \cdot v = 0$$

Remark

- Incompressible fluid dynamics
  - Incompressible Navier–Stokes equations
- Time discretization
  - Operator splitting
  - Integration with the weakly compressible assumption
- Spatial discretization
  - Smoothed particle hydrodynamics (SPH)
- Implementation details (WCSPH)
  - Simulation Pipeline
  - Boundary conditions
  - Neighbor search

$$f(r) \approx \sum_j \frac{m_j}{\rho_j} f(r_j) W(r - r_j, h)$$



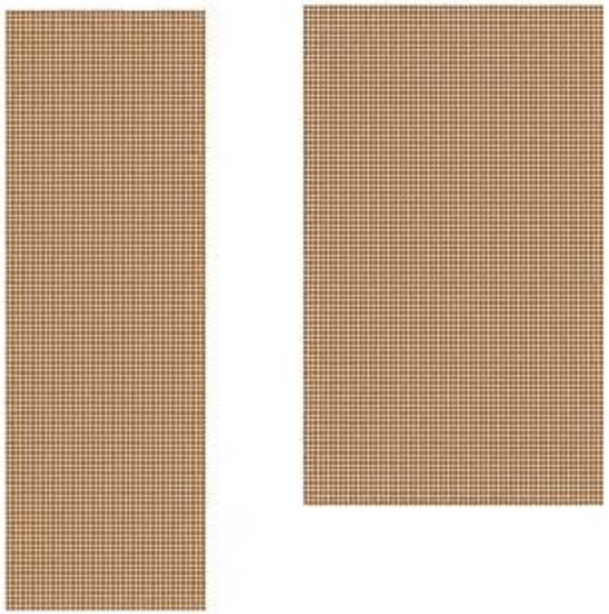


Remark

- Incompressible fluid dynamics
  - Incompressible Navier–Stokes equations
- Time discretization
  - Operator splitting
  - Integration with the weakly compressible assumption
- Spatial discretization
  - Smoothed particle hydrodynamics (SPH)
- Implementation details (WCSPH)
  - Simulation Pipeline
  - Boundary conditions
  - Neighbor search

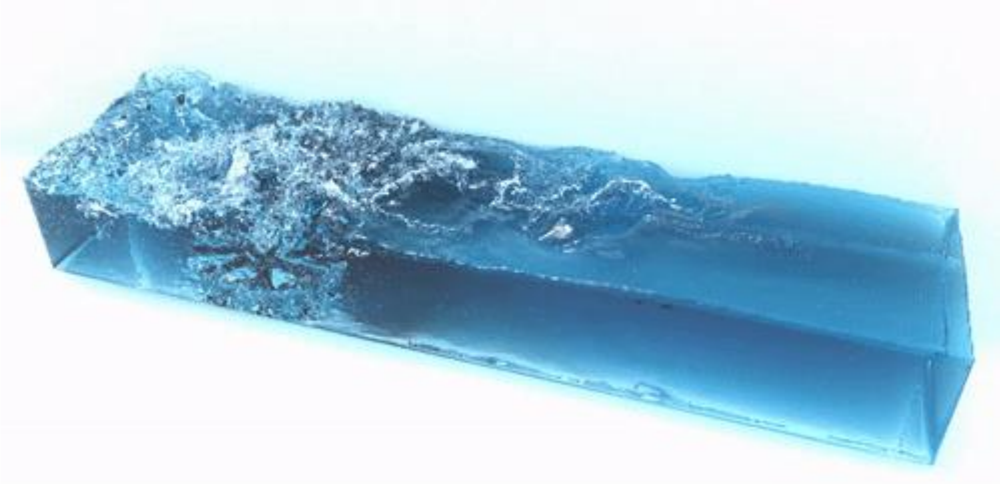
Remark

- Incompressible fluid dynamics
  - Incompressible Navier–Stokes equations
- Time discretization
  - Operator splitting
  - Integration with the weakly compressible assumption
- Spatial discretization
  - Smoothed particle hydrodynamics (SPH)
- Implementation details (WCSPH)
  - Simulation Pipeline
  - Boundary conditions
  - Neighbor search



Further readings

- *Smoothed Particle Hydrodynamics* [Monaghan 2005][Link]
- *Smoothed particle hydrodynamics and magnetohydrodynamics* [Price 2012][Link][Preprint]
- *Smoothed Particle Hydrodynamics Techniques for the Physics Based Simulation of Fluids and Solids* [Eurographics Tutorial, 2019][Link][Code]



Next week

Homework

Homework Today

- Download the repo (taichi_sph):
  - https://github.com/taichiCourse01/taichi_sph

- Try:
  - Designing your own scene
  - Implementing a particle based boundary handling
  - Changing the dense grid in the codebase to a sparse grid [03讲]

Candidate projects for your final

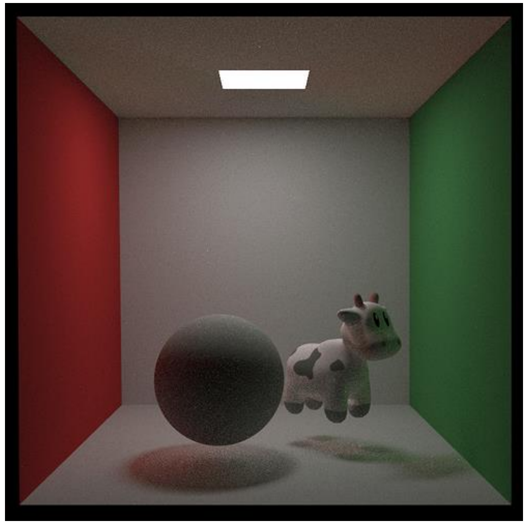
- Candidate topics:
  - Try advanced pressure (Poisson) solvers: IISPH/PCISPH/DFSPH [Chapter 4]
  - Put a statue/kinematically-controlled fan into your pond (One way coupling)
  - Throw a rubber duck into your pond (Two way coupling)
  - Render your fluid using your own path tracer (You may want a marching cube/square to construct the water surface)

- Both 2D and 3D projects are great!
  - As long as your pictures look great ☺

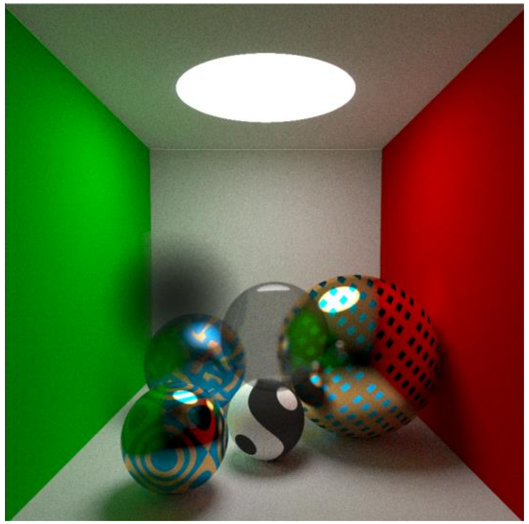
Final project

- 死线：2022年1月3日
- 要求：
  - 使用[大作业模板]
  - 需要有设计文档，如果有参照代码也需要标明
- 题材：
  - 任何使用Taichi完成的内容（图形学更佳）
  - 可以参考每节图形课后给出的大作业选题灵感 [参考第07,09,10,11讲]
  - 鼓励实现任意图形学论文/图形学课程内容
  - 可以在小作业的基础上完成大作业 (Homework Promotion!)
- 形式：
  - 使用 GitHub/Gitee提交并邀请tgc01@taichi.graphics加入你的代码仓
  - 允许三人以下合作，记得管理多人合作的git commits
- 奖励：
  - 太极图形课第一季结业证书一份+神秘Taichi小礼物一份

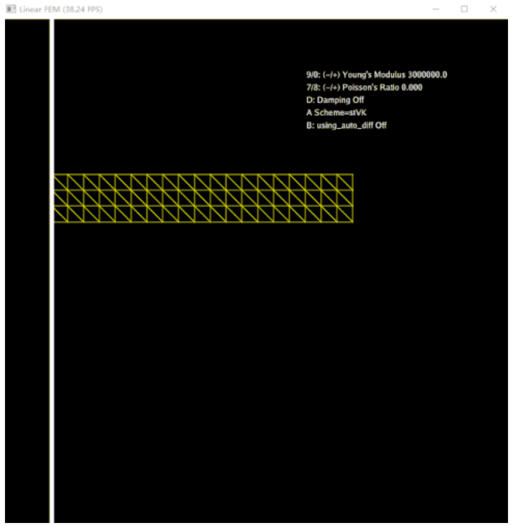
Excellent homework assignments



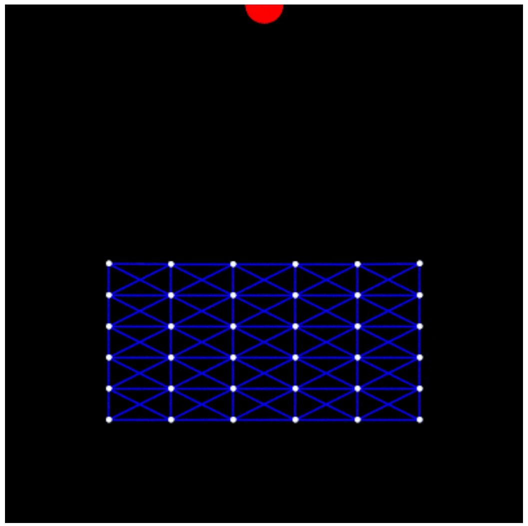
[@moxunbai]



[@runck]



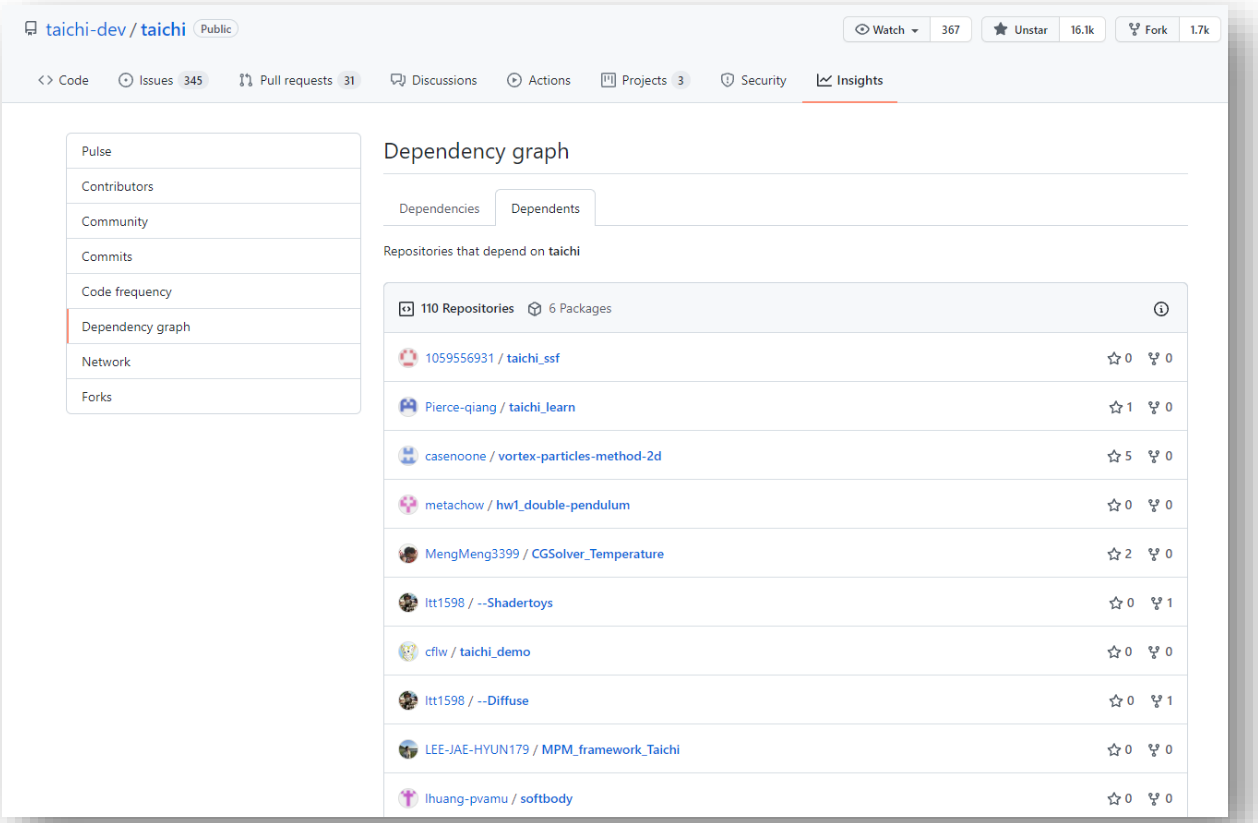
[@MengMeng3399]

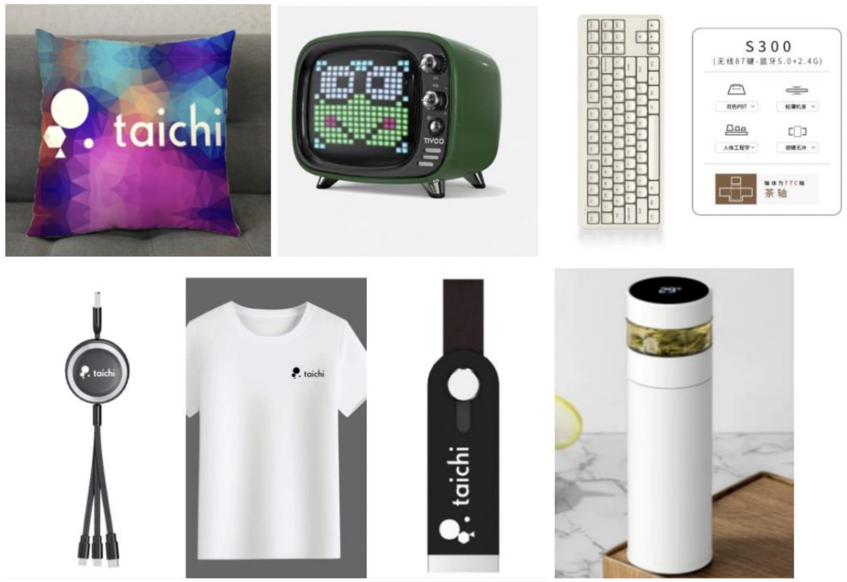


[@runck]

Gifts for the gifted

- Use Template for your homework
- Next check Dec. 14, 2021





Questions?

本次答疑：12/02 🡐作业分享也在这里

下次直播：12/07

直播回放：Bilibili 搜索「太极图形」

主页&课件：https://github.com/taichiCourse01

主页&课件(backup)：https://docs.taichi.graphics/tgc01