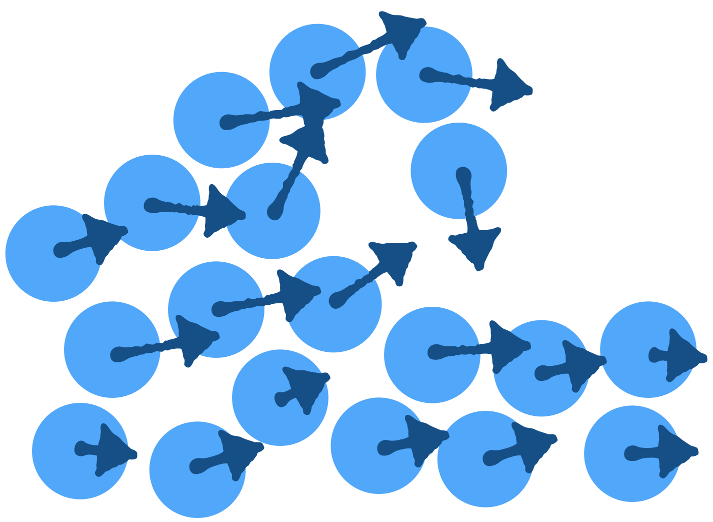
COL865: Special Topics in Computer Applications

# Physics-Based Animation

## 15 — Fluid simulation with particles

# Particles



Throw away the grid, do everything with only particles

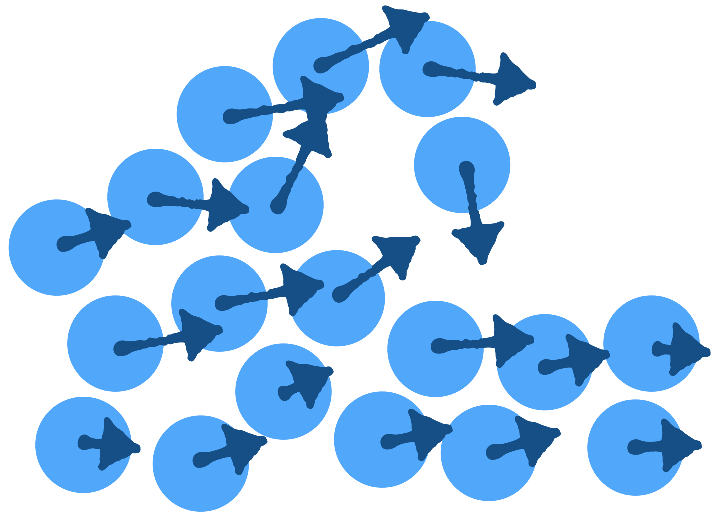
Particle data: mass $m_i$, position $\mathbf{x}_i$, velocity $\mathbf{v}_i$

How to solve Navier-Stokes equations?

$$\rho\, D\mathbf{v}/Dt = -\nabla p + \mu\, \nabla^2\mathbf{v} + \mathbf{f}^{ext}$$

# Particles

$$\rho \, D\mathbf{v}/Dt = -\nabla p + \mu \, \nabla^2 \mathbf{v} + \mathbf{f}^{ext}$$



- ***Advection*** is trivial: Just move particles
- For now, treat fluid as compressible: ***pressure*** is function of density, e.g. $p = k((\rho - \rho_0)^7 - 1)$

$$d\mathbf{x}_i/dt = \mathbf{v}_i$$

$$m_i \, d\mathbf{v}_i/dt = \mathbf{F}_i$$
$$= V_i \, (-\nabla p(\mathbf{x}_i) + \mu \, \nabla^2 \mathbf{v}(\mathbf{x}_i) + \mathbf{f}_i^{ext})$$

How to compute spatial derivatives $\nabla p$, $\nabla^2 \mathbf{v}$?
Standard strategy: reconstruct smooth function, differentiate

# Smoothed particle hydrodynamics

***Readings***:

- Müller et al., “Particle-Based Fluid Simulation for Interactive Applications”, 2003
- Becker & Teschner, “Weakly Compressible SPH for Free Surface Flows”, 2007

For further reference:

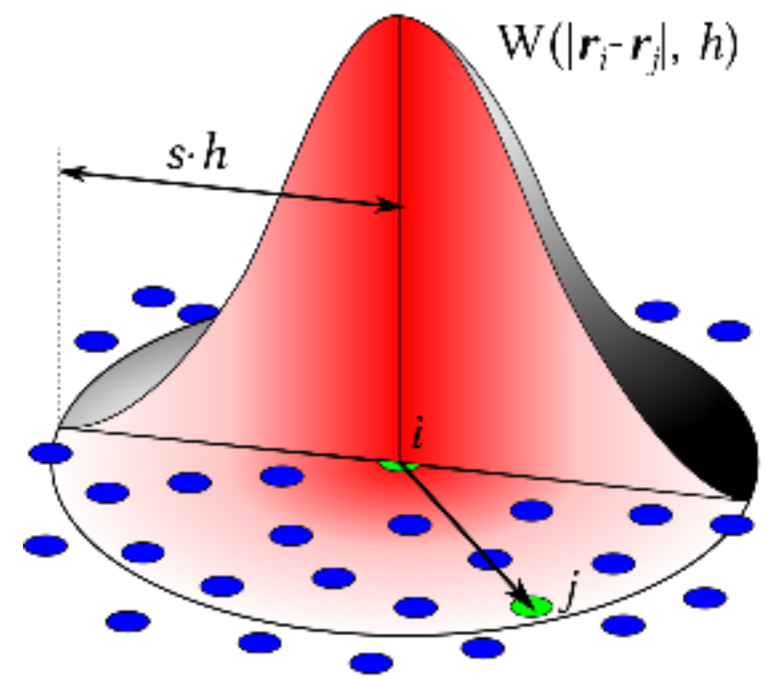
- ***Theoretical background***: Monaghan, “Smoothed Particle Hydrodynamics”, 2005
- ***State-of-the-art survey in graphics***: Ihmsen et al., “SPH Fluids in Computer Graphics”, 2014

# Interpolation via smoothing

Given values $A_i$ at particles $\mathbf{x}_i$, estimate $A(\mathbf{x})$

Define ***kernel function*** $W(\mathbf{x}, h)$

Then what?




- ***Weighted sum***: $A(\mathbf{x}) \stackrel{?}{=} \sum A_i\, W(\mathbf{x}-\mathbf{x}_i, h)$

  Doesn't work if particles are too close / too far

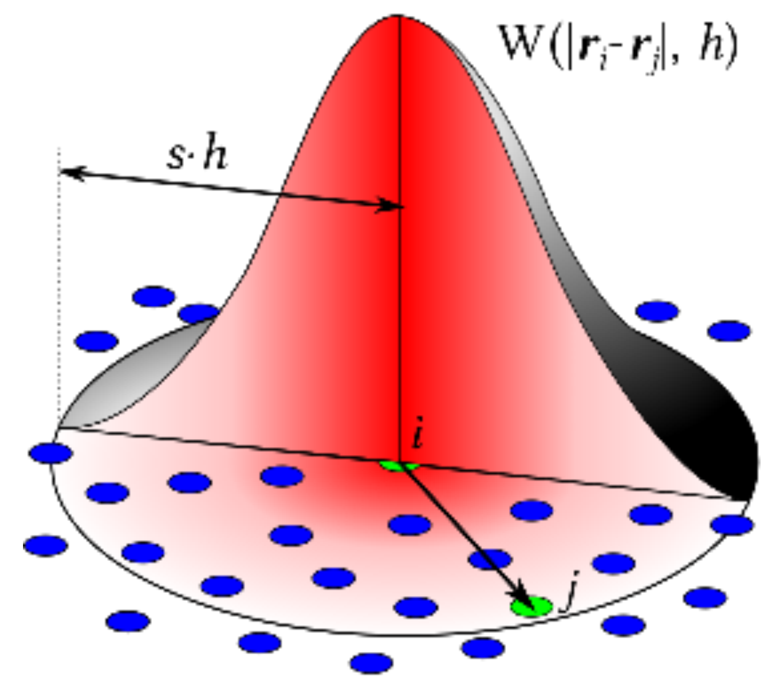
- ***Weighted average***: $A(\mathbf{x}) \stackrel{?}{=} (\sum A_i\, W(\mathbf{x}-\mathbf{x}_i, h)) \,/\, (\sum W(\mathbf{x}-\mathbf{x}_i, h))$

  Works, but denominator makes differentiation complicated

# SPH interpolation

We can use $W$ to smooth a function:

$$A_{\text{smooth}}(\mathbf{x}) = \iiint A(\mathbf{x}')\, W(\mathbf{x}-\mathbf{x}', h)\, \mathrm{d}V'$$




Discretize volume integral over particles:

$$A_s(\mathbf{x}) = \sum A_i\, W(\mathbf{x}-\mathbf{x}_i, h)\, V_i$$

What is volume $V_i$ for a particle?
Suppose(!) we know density $\rho_i = \rho(\mathbf{x}_i)$. Then $V_i = m_i/\rho_i$

Plug it in:

$$\rho(\mathbf{x}) = \sum \rho_i\, W(\mathbf{x}-\mathbf{x}_i, h)\, m_i/\rho_i$$
$$= \sum m_i\, W(\mathbf{x}-\mathbf{x}_i, h)$$

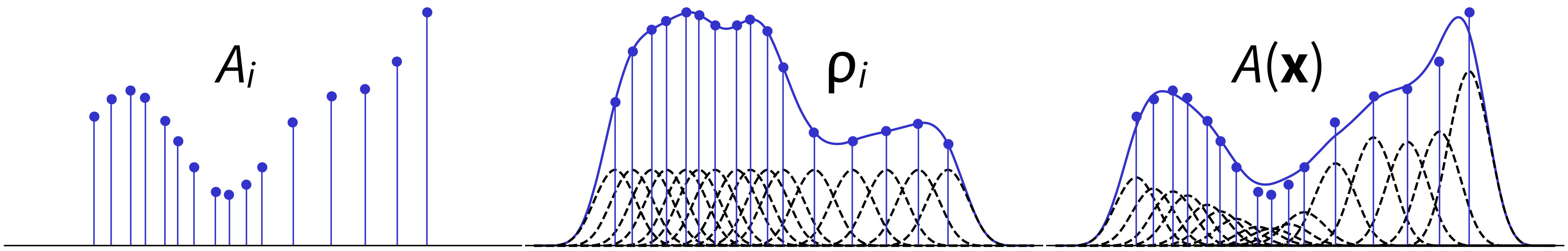
# SPH interpolation

1. Compute density at each particle

$$\rho_i = \sum m_j \, W(\mathbf{x}_i - \mathbf{x}_j, h)$$

2. Interpolate other quantities

$$A(\mathbf{x}) = \sum (m_i/\rho_i) \, A_i \, W(\mathbf{x} - \mathbf{x}_i, h)$$



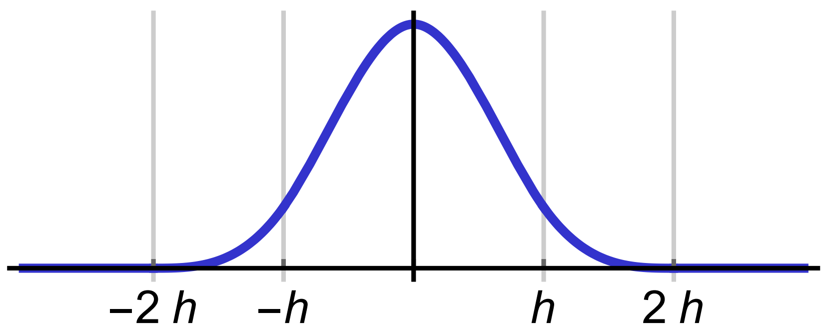

Smoothing effect: $A(\mathbf{x}_i) \neq A_i$

# Choice of kernel

Desirable properties of $W(\mathbf{x}, h)$:

- Decreasing with $\|\mathbf{x}\|$
- $\iiint W(\mathbf{x}, h)\, \mathrm{d}V = 1$
- Finite support: $W(\mathbf{x}, h) = 0$ when $\|\mathbf{x}\|/h$ is large
    - Sum only over nearby nearby particles,
      use uniform grid to accelerate neighbour search

Typical choice: cubic spline [Ihmsen et al. 2014], though others also used




Choice of $h$: usually close to particle spacing

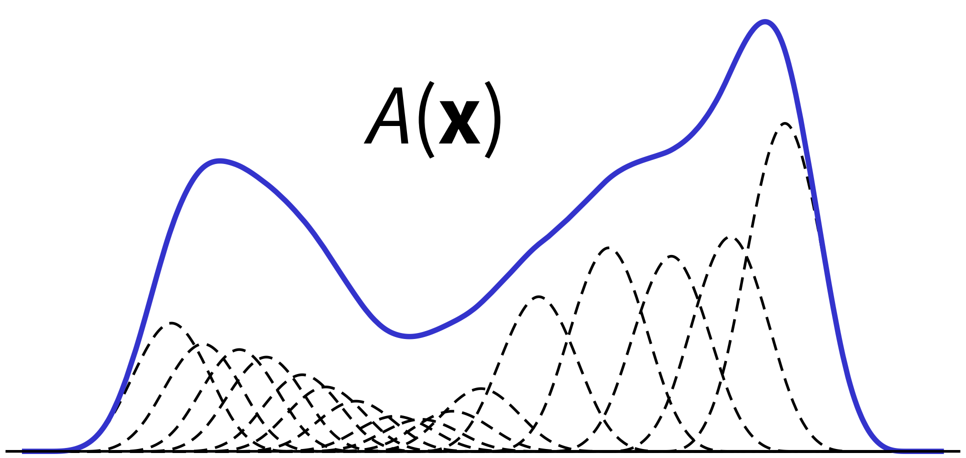
# Derivatives

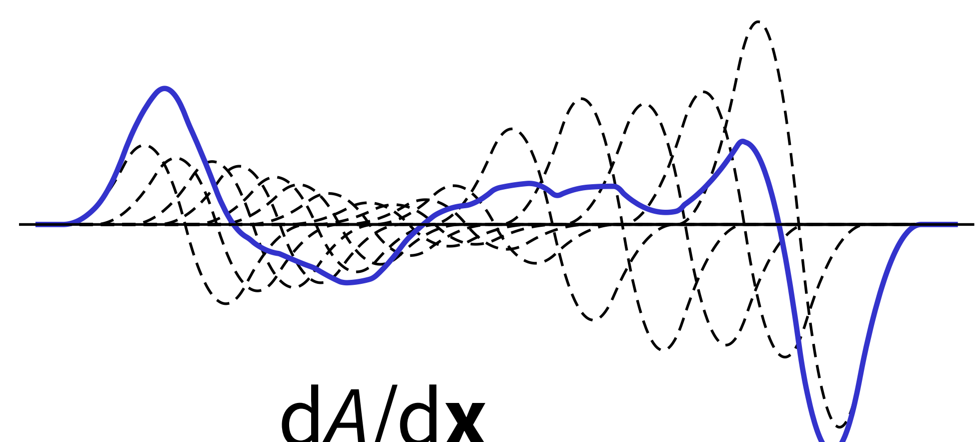
Derivatives just require differentiating the kernel:

$$A(\mathbf{x}) = \sum (m_i/\rho_i)\, A_i\, W(\mathbf{x}-\mathbf{x}_i, h)$$

$$\nabla A(\mathbf{x}) = \sum (m_i/\rho_i)\, A_i\, \nabla W(\mathbf{x}-\mathbf{x}_i, h)$$

$$\nabla^2 A(\mathbf{x}) = \sum (m_i/\rho_i)\, A_i\, \nabla^2 W(\mathbf{x}-\mathbf{x}_i, h)$$



# SPH forces

Now that we have $\rho_i$, we can set particle volume $V_i = m_i/\rho_i$, and particle pressure $p_i = p(\rho_i)$

$$m_i \, \mathrm{d}\mathbf{v}_i/\mathrm{d}t = V_i \left(-\nabla p(\mathbf{x}_i) + \mu \nabla^2 \mathbf{v}(\mathbf{x}_i) + \mathbf{f}_i^{\mathrm{ext}}\right)$$

Forces on particle $i$:

- ***Pressure***: $\mathbf{F}_i^{\mathrm{p}} = -m_i/\rho_i \, \nabla p(\mathbf{x}_i)$
- ***Viscosity***: $\mathbf{F}_i^{\mathrm{v}} = m_i \, \nu \, \nabla^2 \mathbf{v}(\mathbf{x}_i)$
- ***Body force***: $\mathbf{F}_i^{\mathrm{ext}}$

Just apply SPH derivatives?

# SPH forces

$\mathbf{F}_i^{\text{p}} = -m_i/\rho_i \, \nabla p(\mathbf{x}_i)$

$\quad = -m_i/\rho_i \sum (m_j/\rho_j) \, p_j \, \nabla_i W_{ij}$

***Problem***: force on *i* due to *j* ≠ −force on *j* due to i

Various solutions:

- Just replace $p_j$ with $(p_i + p_j)/2$ [Müller et al. 2003]
- Compute $\nabla p/\rho$ as $\nabla(p/\rho) + p/\rho^2 \, \nabla\rho$ [Monaghan 2005]:

$$\left(\frac{\nabla p}{\rho}\right)_i = \sum_j m_j \left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2}\right) \nabla_i W_{ij}$$

# SPH forces

$$\mathbf{F}_i^{\nu} = m_i\,\nu\,\nabla^2\mathbf{v}(\mathbf{x}_i)$$
$$= m_i\,\nu \sum (m_j/\rho_j)\,\mathbf{v}_j\,\nabla_i^2 W_{ij}$$

***Problems***:

- Force is nonzero for rigid motion, or even if all $\mathbf{v}_i$ are equal!
- Second derivative causes numerical difficulties

Physically consistent formula [Monaghan 2005]:

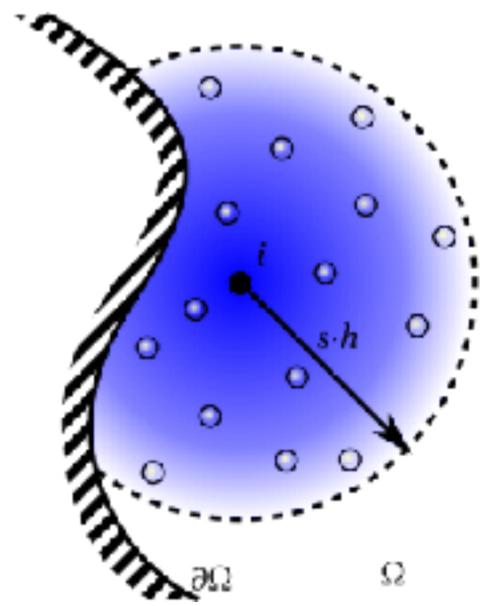
$$(\nabla^2\mathbf{v})_i = 2(d+2)\sum_j \frac{m_j}{\rho_j}\frac{\mathbf{v}_{ij}\cdot\mathbf{x}_{ij}}{\|\mathbf{x}_{ij}\|^2 + 0.01h^2}\nabla_i W_{ij}$$


[Weiler et al. 2018]

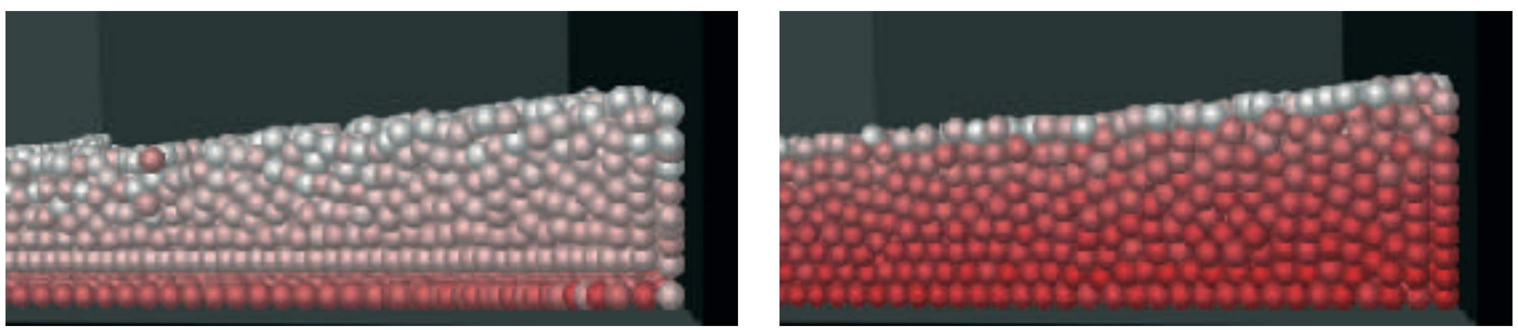
Cheap hack: Just update $\mathbf{v}_i \mathrel{+}= \varepsilon \sum (m_j/\rho_j)\,(\mathbf{v}_j - \mathbf{v}_i)\,W_{ij}$

# Boundary issues

SPH interpolation doesn't know about boundaries



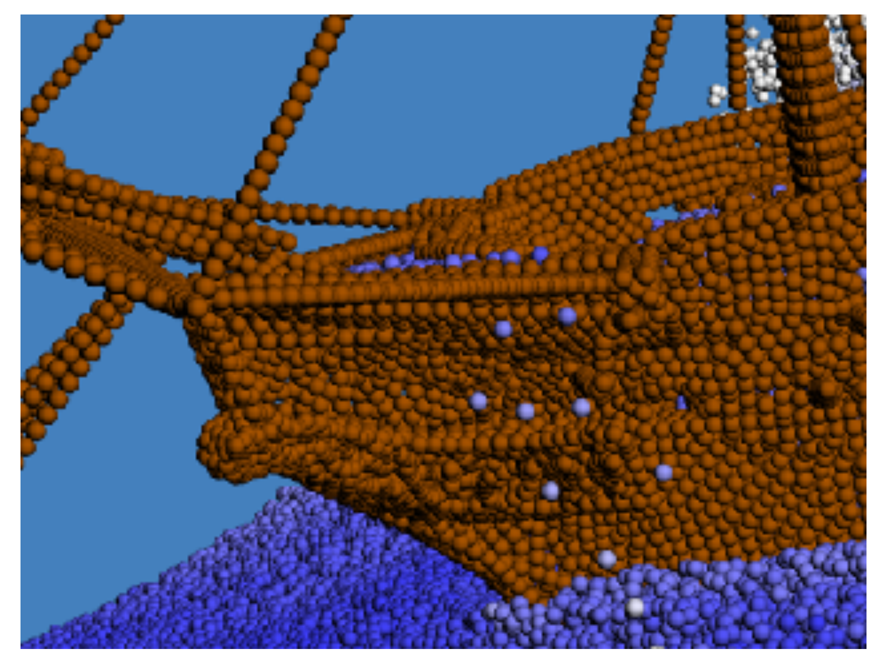

***Particle deficiency*** → density underestimated → clumping, sticking



[Ihmsen et al. 2010]

***Solution***: sample solids with particles too, include in ρ, $p$ computation

See Ihmsen et al. survey, Ch 4



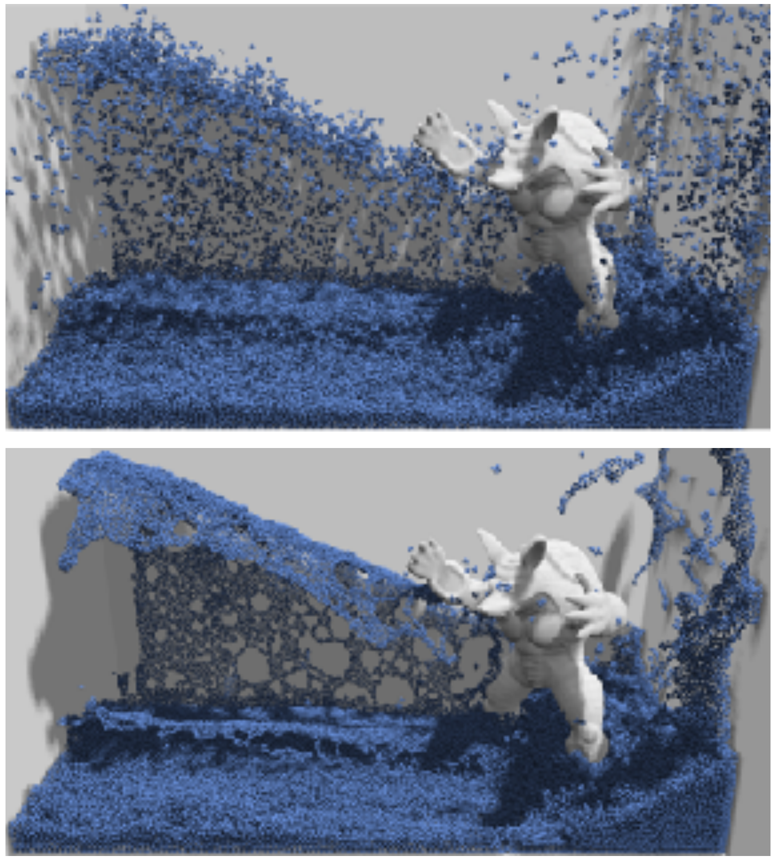
[Akinci et al. 2012]

# Boundary issues

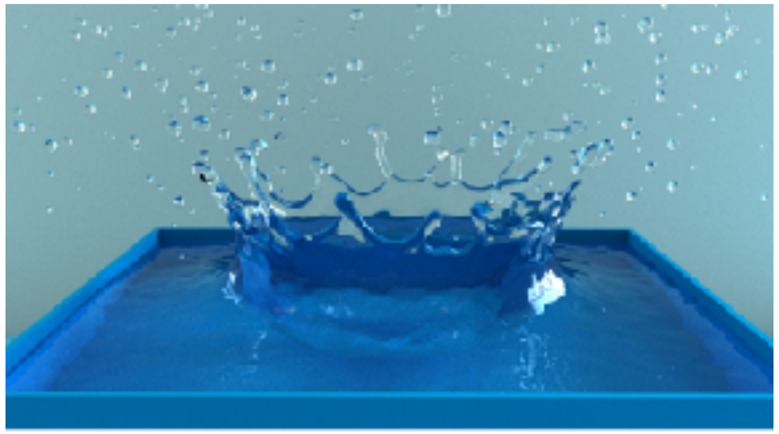
Same problem at free surfaces:
***tensile instability***

- Create ghost particles [Schechter & Bridson 2012]
- Or add a repulsion term to prevent clumping [Macklin & Müller 2012, Akinci et al. 2013]

Surface tension can be added in various ways, see Ihmsen et al. survey Ch 6.3



[Macklin & Müller 2012]

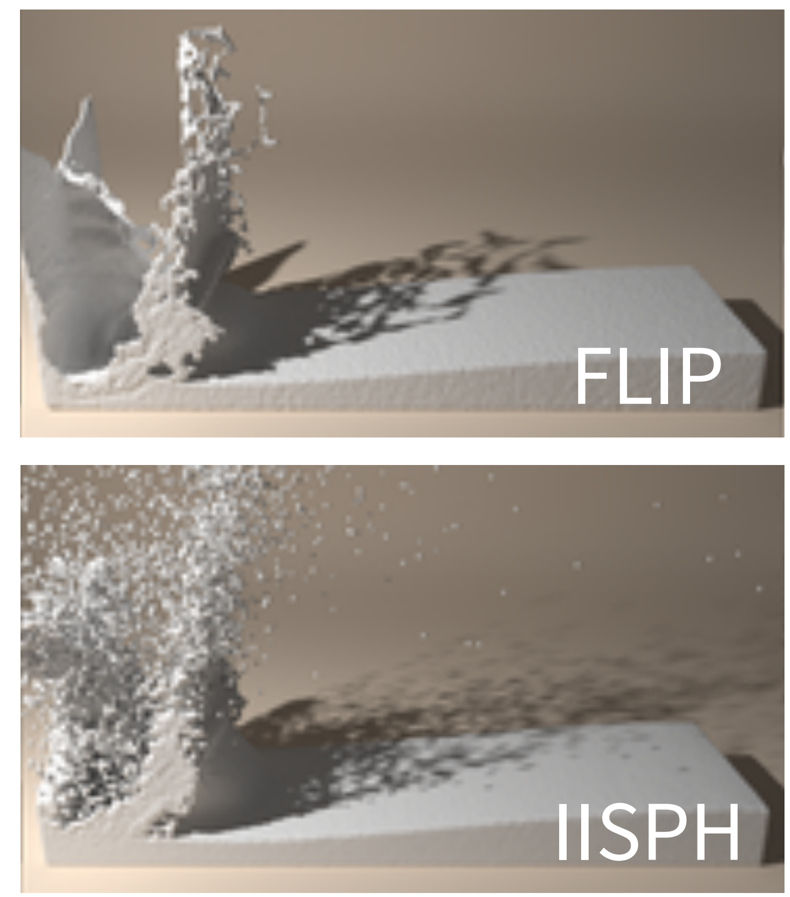


[Akinci et al. 2013]

# Summary

Core idea of SPH is simple, but consistent handling of forces, boundaries is more complicated

- Stability: small time steps, artificial viscosity
- Incompressible fluids with SPH:
  PCISPH [Solenthaler & Pajarola 2009],
  IISPH [Ihmsen et al. 2013],
  DFSPH [Bender & Koschier 2015]
- Parameter tuning usually needed
- Better splashes than grid fluids for same resolution (but more compute)




[Um et al. 2017]