Animating fluids (II)

- Procedural
- Level Set
- PIC/FLIP
- SPH

Procedural models for free surface water

Often based on **Shallow water equation**

neglect depth velocity component

[A. Fournier, W. T. Reeves. A simple Model of Ocean Waves, ACM SIGGRAPH 1986]

[D. Hinsiger et al. Interactive Animation of Ocean Waves. SCA 2002]

[Jerry Tessendorf, Simulating Ocean Water, ACM SIGGRAPH Course Notes 2004]

ex. *Trochoid/Gerstner models*

Particles following circular trajectories, waves propagates

$$\mathbf{x} - \mathbf{x}_0 = \sum_i a_i \mathbf{k}_i / \|\mathbf{k}_i\| \ \sin(\mathbf{k}_i \cdot \mathbf{x}_0 - \omega_i \, t + \phi_i)$$

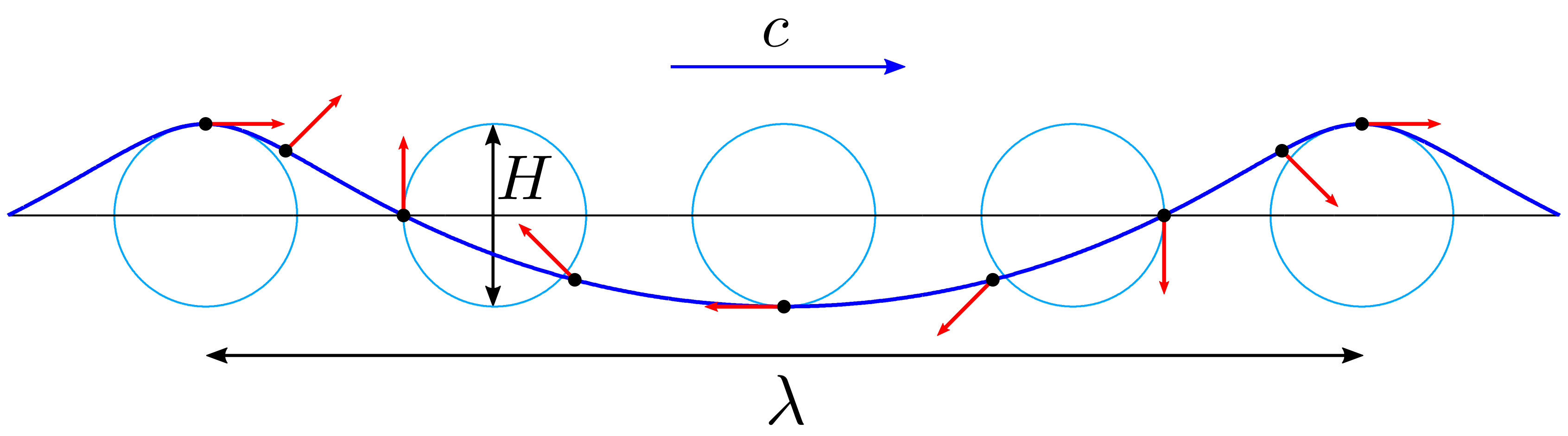
$$z - z_0 = \sum_i a_i \cos(\mathbf{k}_i \cdot \mathbf{x}_0 - \omega_i \, t + \phi_i)$$

$\omega_i^2 = g \, k_i$ - *dispersion*

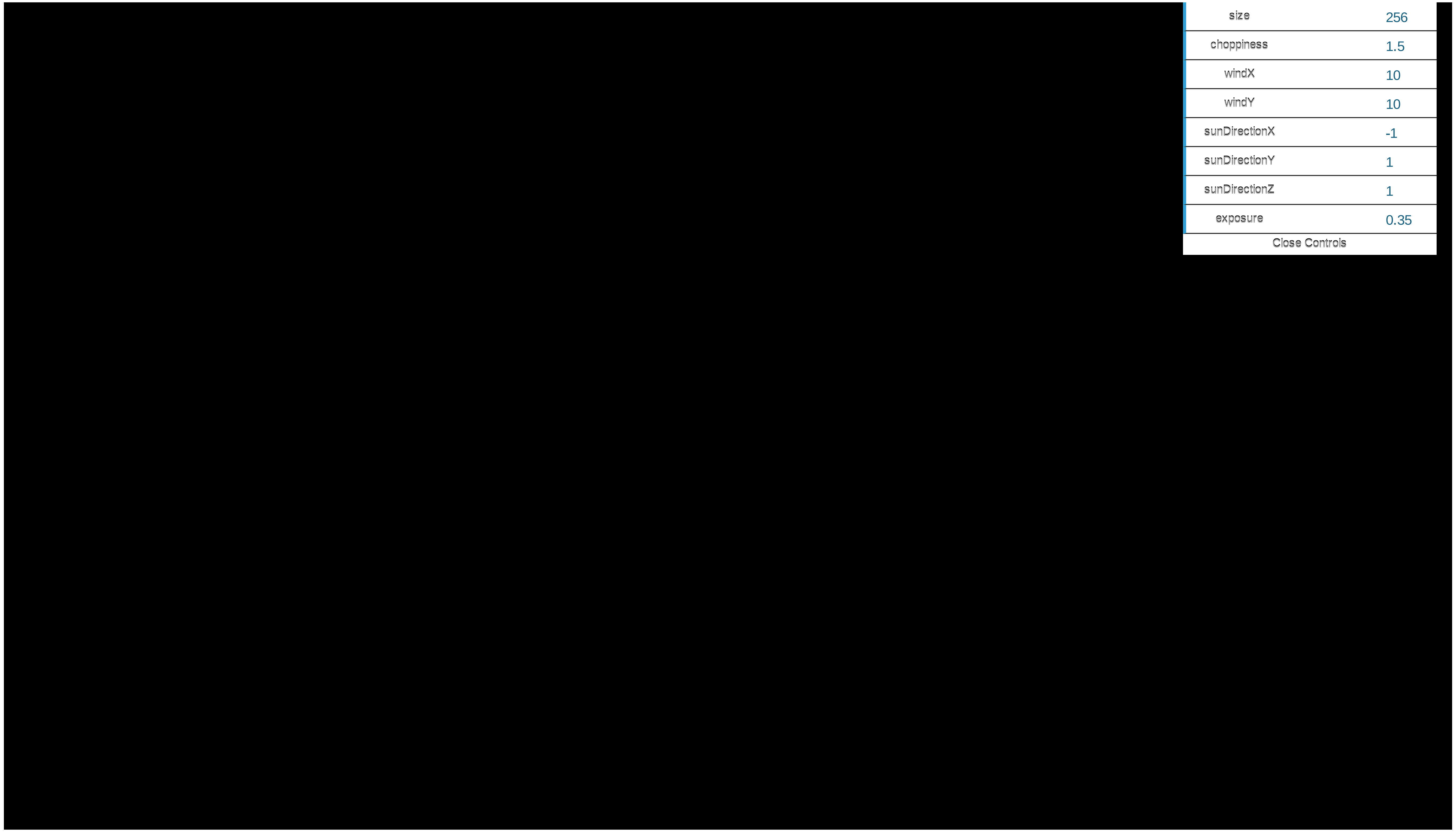
Used for procedural ocean modeling

(+) Simple and scalable

(-) Interaction with other objects



Free surface water - Example



Link Three.js

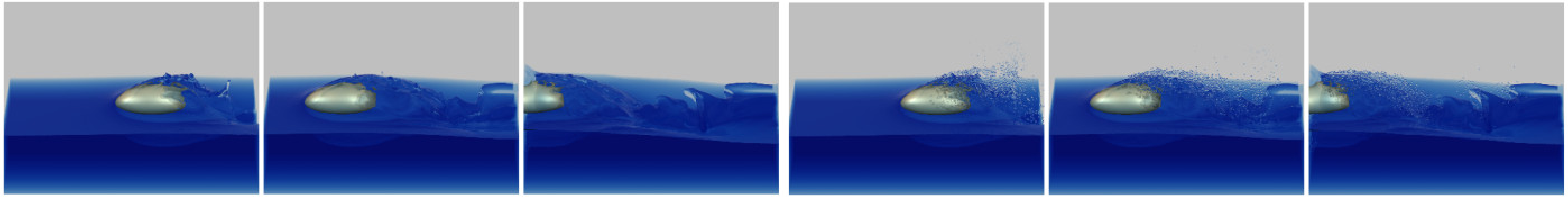
Level Set methods introduction

Eulerian models (ex. stable fluids) do not handle natively free boundaries such as fluid/air.

Idea: Track surface boundary using implicit surfaces

- Encode fluid volume by $\Omega = \{p \in \mathbb{R}^3, \varphi(p) = 0\}$, φ stored within a 3D grid.
- Solve Navier-Stokes equation within Ω
- Deform fluid volume using implicit surface deformation $\dfrac{\partial \varphi}{\partial t} + u \cdot \nabla \varphi = 0$

Introduced by Ronald Fedkiw, Nick Foster, Stanley Osher (ex. [Practical Animation of Liquids, ACM SIGGRAPH 2001])



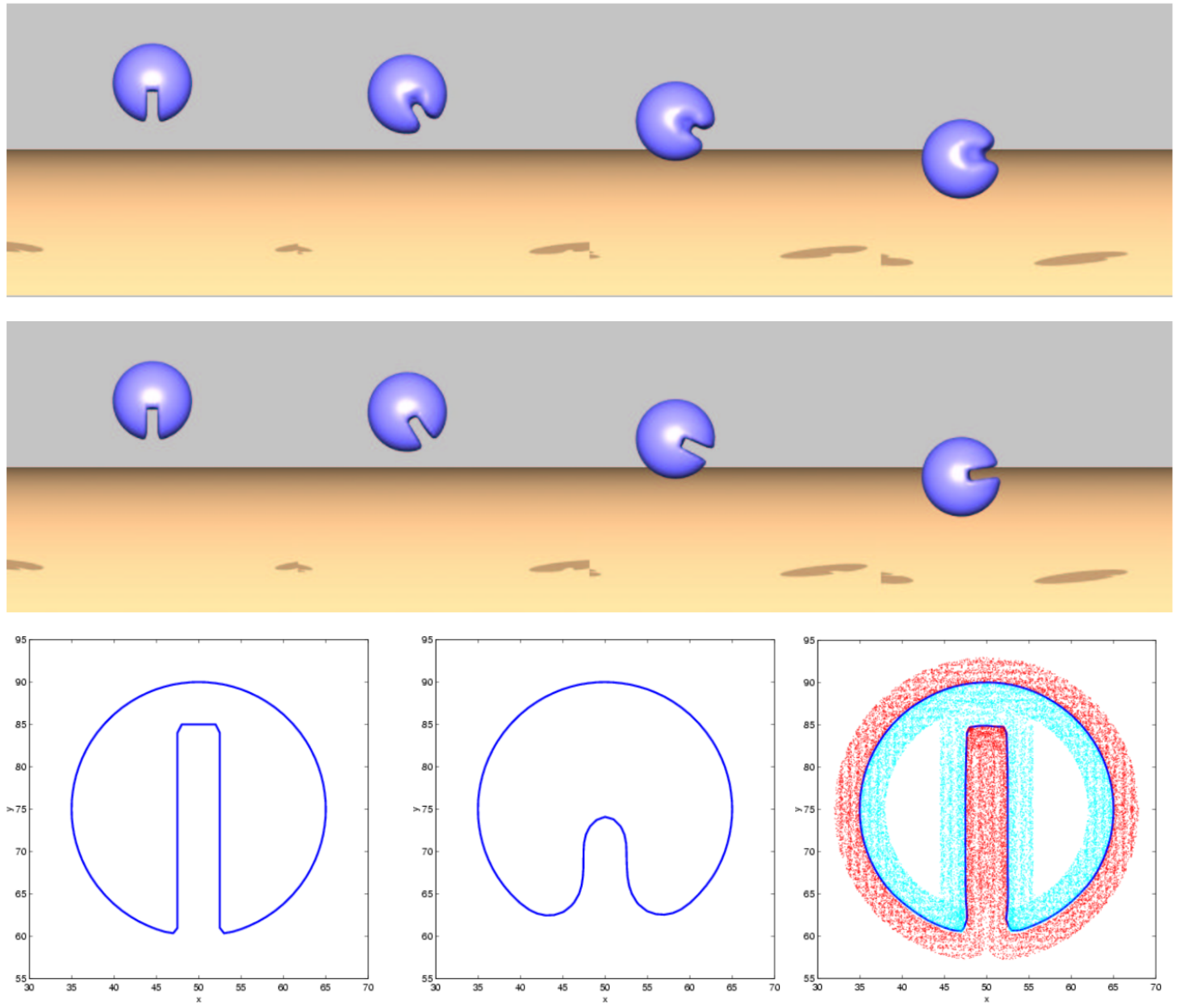
Particles Level Set

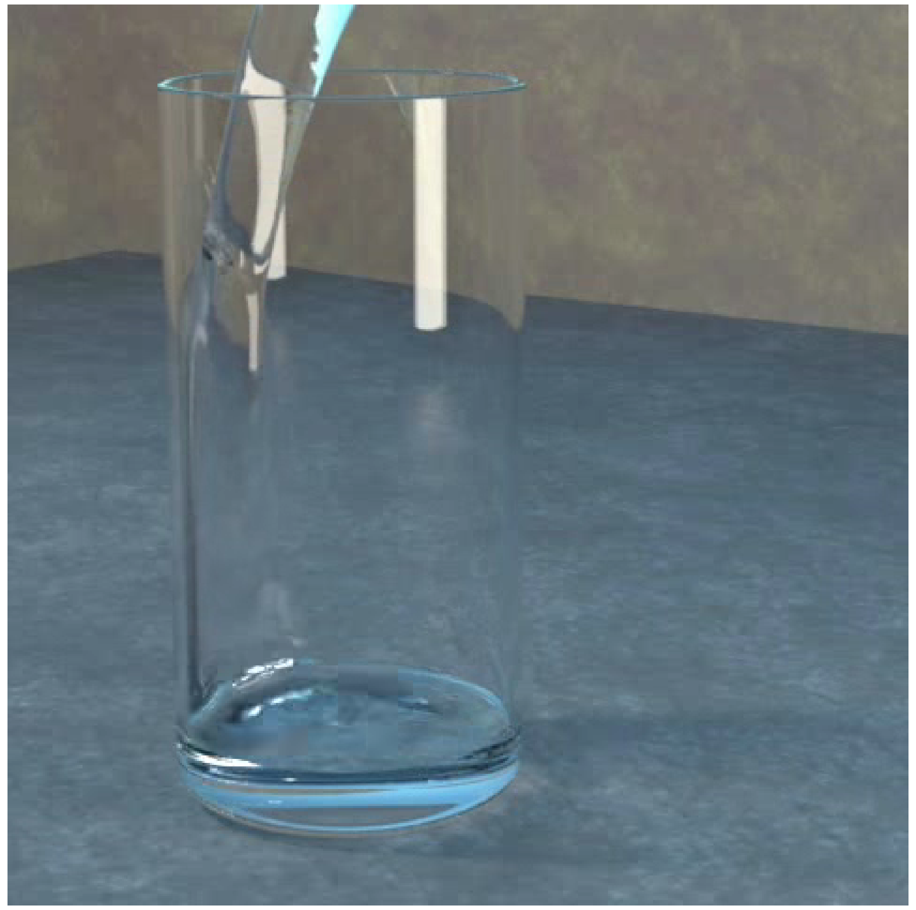
Limitation of level sets: smoothing and loss of volume from grid interpolation

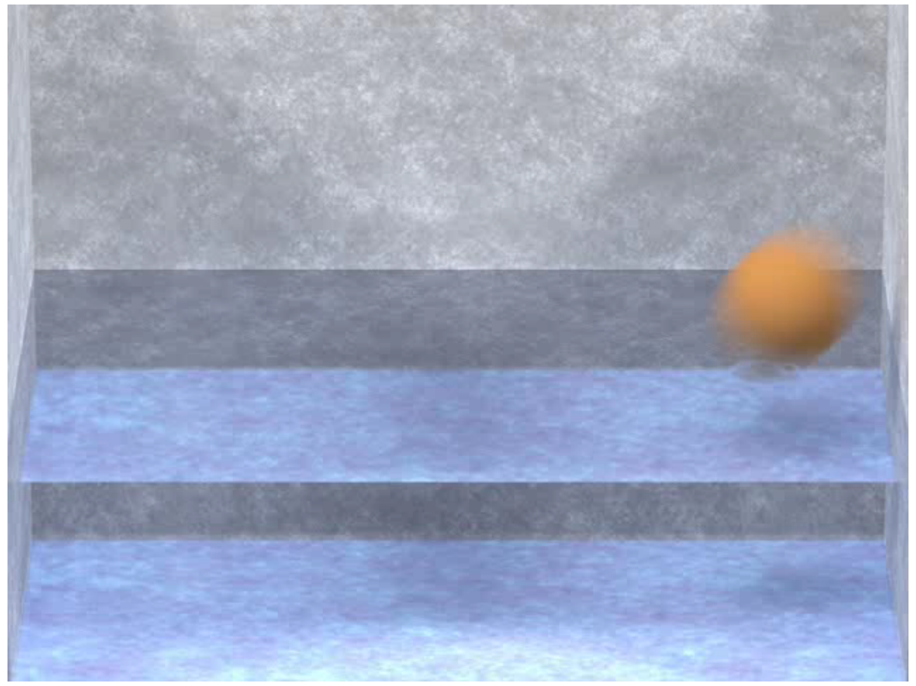
⇒ Use of **Particle Level Set Method**

[D. Enright et al. Hybrid Particle Level Set Method for Improved Interface Capturing. J. Comp. Physics 2001]

[D. Enright et al. Animation and Rendering of Complex Water Surfaces, ACM SIGGRAPH 2002]





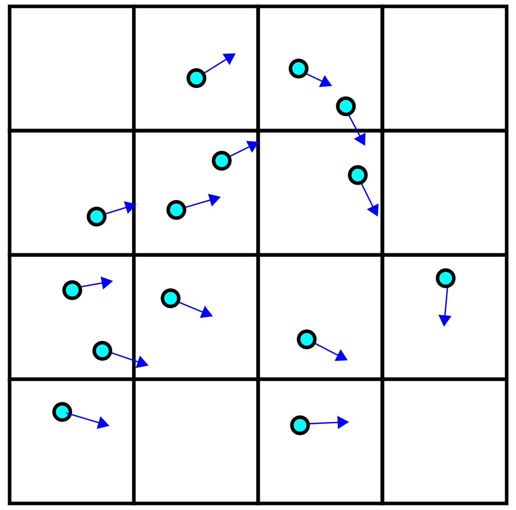


D. Enright et al.

PIC/FLIP (Material Point Method)

Mix between particles and grid based approach.

- Particles: good for advection
- Grid: forces, pressure, viscosity



u_p: velocity on particle

u_g: velocity on grid

- **PIC** approach - Transfert velocity from grid to particles

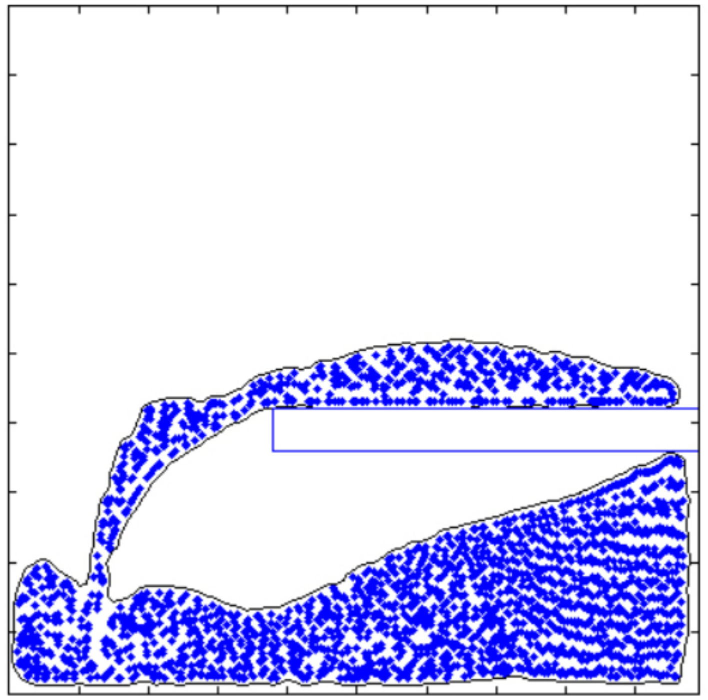
$$u_p^{k+1} = interp(u_g^{k+1}, p^{k+1})$$

- **FLIP** approach - Add velocity difference from grid to particles.

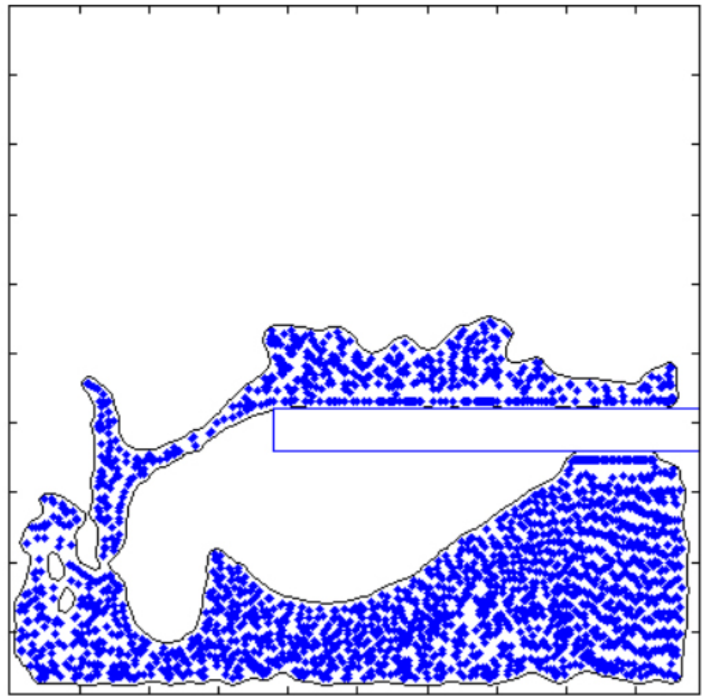
$$u_p^{k+1} = v_p^k + (interp(u_g^{k+1}, p^{k+1}) - interp(u_g^k, p^k))$$

- **PIC/FLIP** : blending b/w two approaches

[Y. Zhu and R. Bridson, Animating Sand as a Fluid, ACM SIGGRAPH 2005]



PIC: Stable, smoothed-out



FLIP: Details, few dissipation

MAC grid

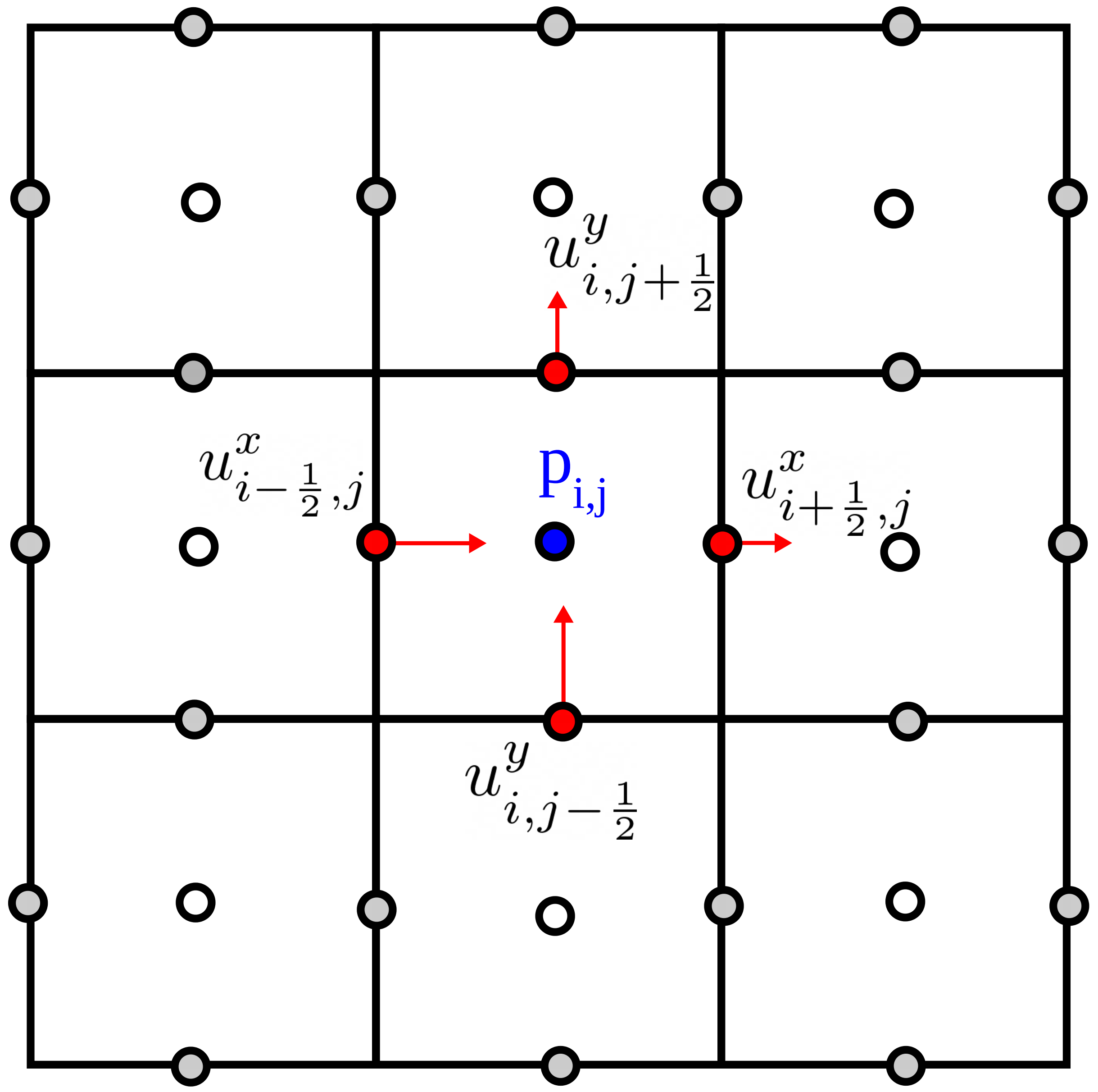
MAC = Marker And Cell

Staggered grid b/w scalar and velocity

Widely used grid storage to handle velocity and scalar values.

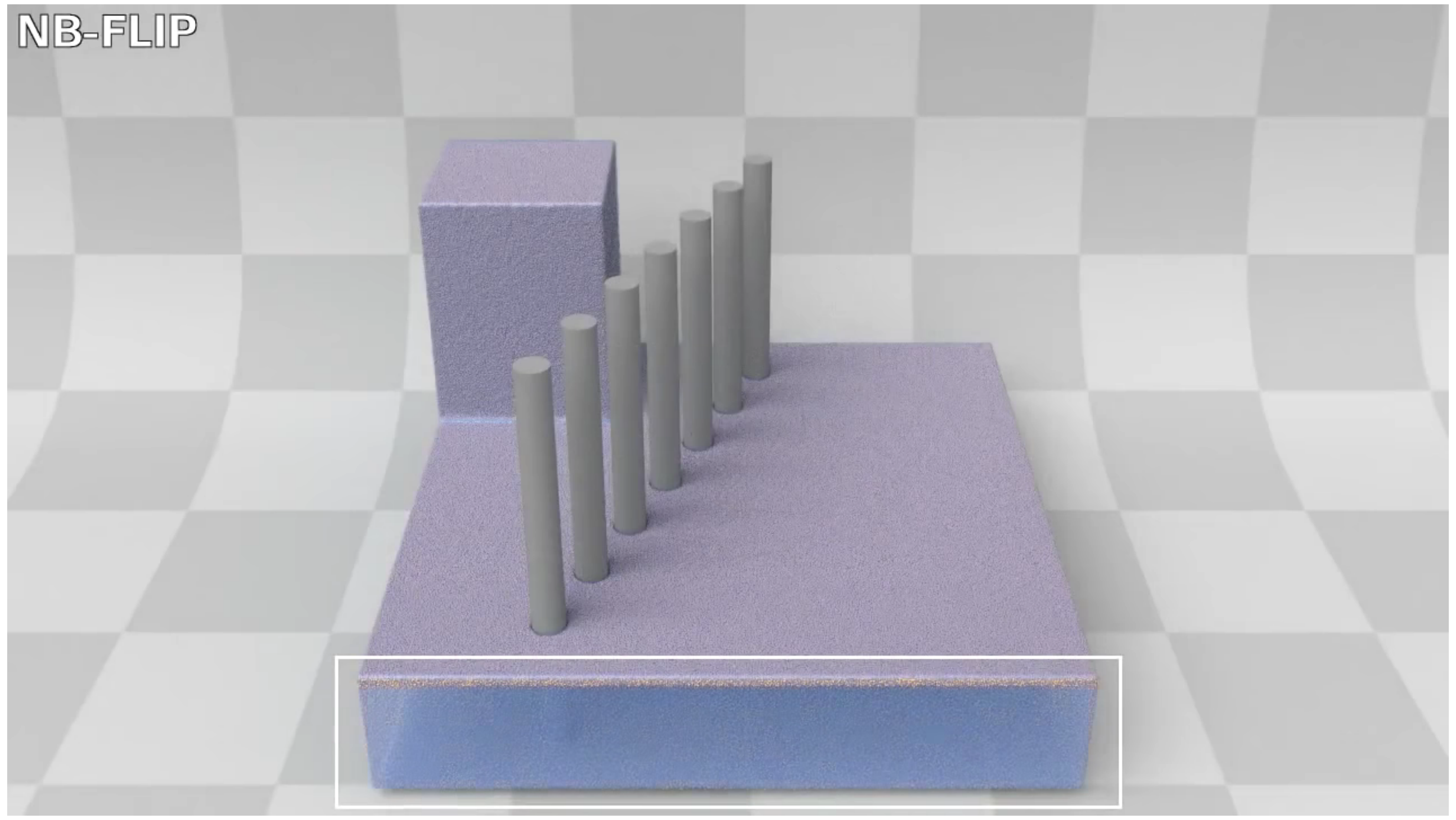
- Store scalar (pressure, density), in the center of the cell
- Store velocity components (u^x, u^y) on the cell edges

Improves accuracy and stability



PIC/FLIP Method

- Transfert particle velocity to the MAC grid (Store velocity u^k on grid)
- Evolve velocity on grid (pressure, forces, viscosity) excepted advection to u^{k+1}
- Add velocity difference $\Delta u = u^{k+1} - u^k$ to particles using interpolation (FLIP approach)
- Blend particle velocity with interpolated grid velocity (PIC/FLIP)
- Advect particles along their new velocity



[F. Ferstl et al., EUROGRAPHICS 2016]

Animating fluids

SPH

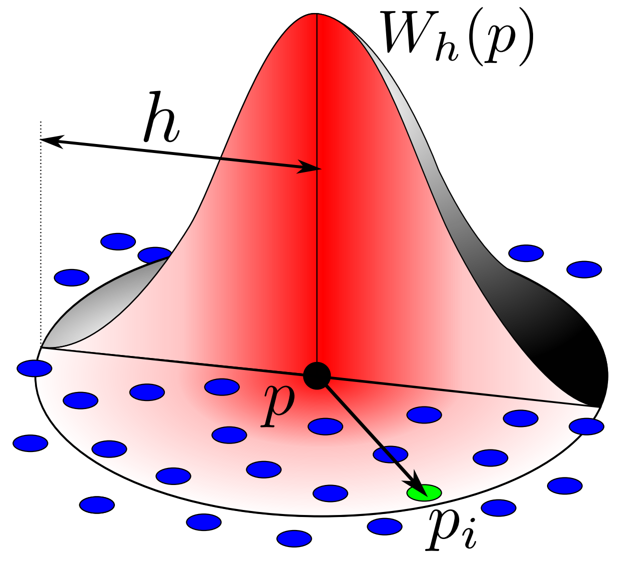
SPH - Smoothed Particle Hydrodynamics

Pure Lagrangian approach.

- Sample the fluid volume with particles
- Build a continuous field from local averaging around samples

  Use some local weighting kernel W

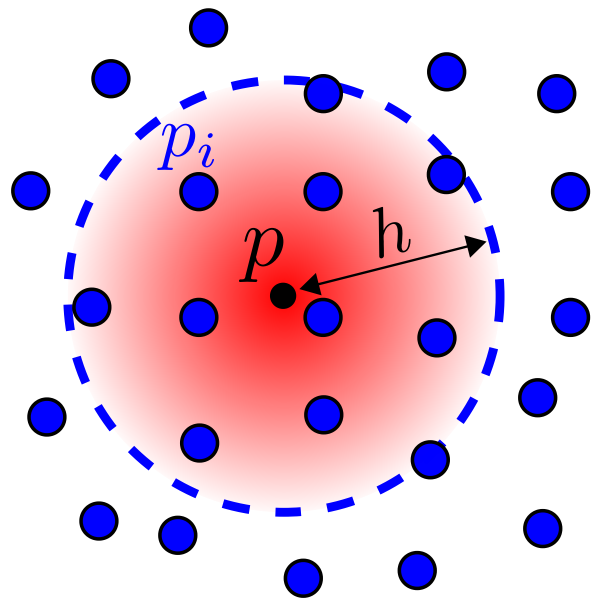
- Express derivatives/Navier-Stokes on the continuous field



Advantages

(+) Particle based - can interact with other models

(+) Scalable

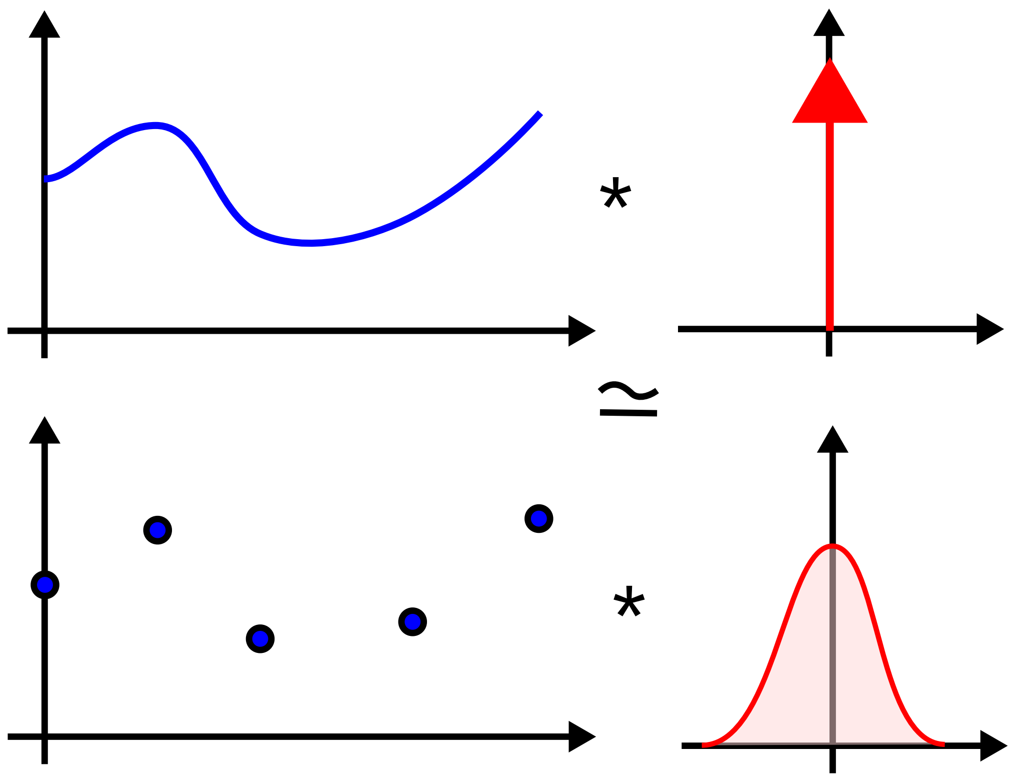


Initial proposed in Astronomy field

[L. Lucy, A numerical approach to the testing of the fission hypothesis. The Astronomical Journal, 1977.]

Sampling and density

How-to build a continuous field from arbitrary sampled particles ?



Consider arbitrary continuous field $A(p)$

Def. of convolution: $A(p) = (A \star \delta)(p) = \int_\Omega A(q)\,\delta(p-q)\,\mathrm{d}q$

1. Consider W_h a smooth kernel with $\int_\Omega W_h(p)\,\mathrm{d}p = 1$

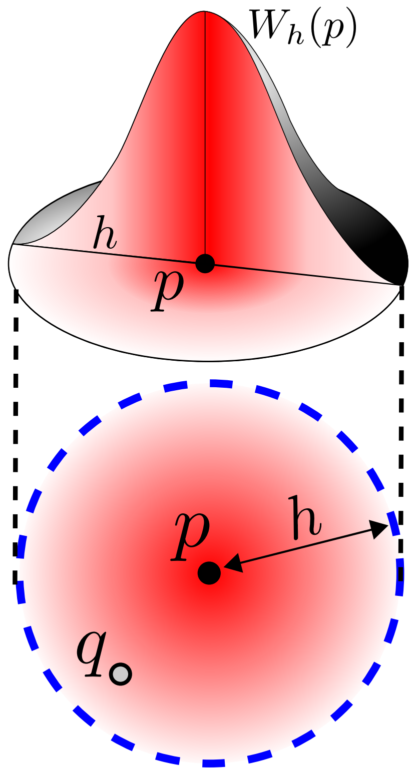
   $A(p) \simeq (A \star W_h)(p) = \int_\Omega A(q)\,W_h(p-q)\,\mathrm{d}q$

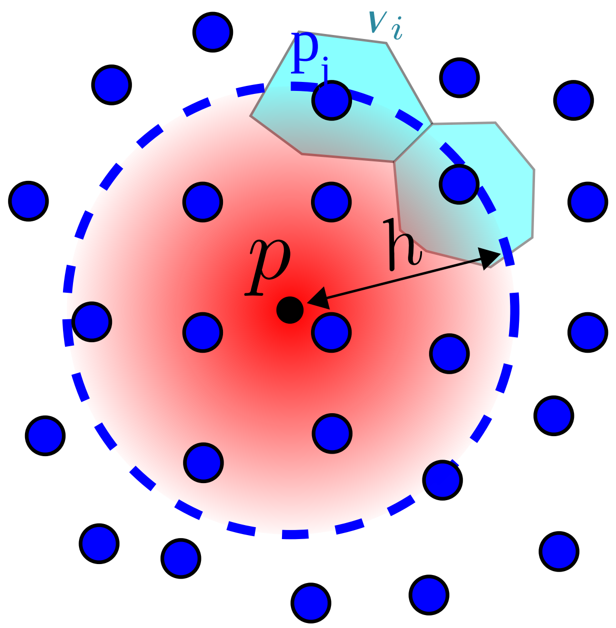
   Low pass filter applied to A

2. Discrete sampling on p_j

   $A(p) = \sum_j A(p_j) W_h(p - p_j) V_j$

   V_j: *small volume associated to* p_j





SPH for Navier Stokes

[Desbrun and Cani, Smoothed Particles: A new paradigm for animating highly deformable bodies, EGCAS 1996]
[M. Muller et al., Particle-Based Fluid Simulation for Interactive Applications, SCA 2003]
[M. Ihmsen et al., SPH Fluids in Computer Graphics, EG STAR 2014]

Lagrangian representation on particle i

$$m_i \frac{\mathrm{d}v_i}{\mathrm{d}t} = \underbrace{m_i\, g}_{F_{weight}} - \underbrace{\frac{m_i}{\rho_i}\, \nabla \mathrm{p}_i}_{F_{pressure}} + \underbrace{m_i\, \nu \triangle v_i}_{F_{viscosity}}$$

Objective:

1. Express ρ_i, $\nabla \mathrm{p}_i$, $\triangle v_i$ using SPH formulation
2. Then integrate: ex. $v_i^{k+1} = v_i^k + \Delta t \left(F_{weight} + F_{pressure} + F_{viscosity}\right)/m_i$

Generic SPH representation:

Arbitrary field A at position p_i: $A(p_i) = \sum_j A(p_j)\, W_h(p_i - p_j)\, V_j$

For a particle of total mass m_i in the volume V_i: $\rho_i V_i = m_i \quad \Rightarrow \; A(p_i) = \sum_j A(p_j)\, m_j/\rho_j\, W_h(p_i - p_j)$

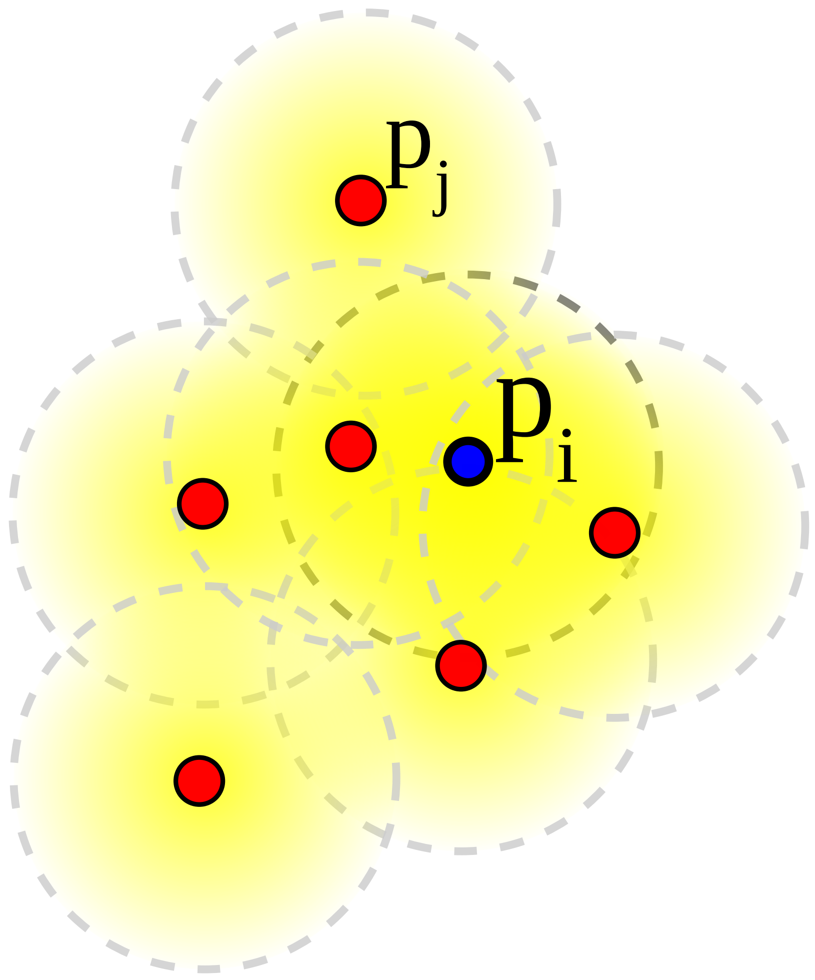
Usually W_h are distance function : $$A(p_i) = \sum_j A(p_j)\, m_j/\rho_j\, W_h(\|p_i - p_j\|)$$

Density

ρ_i: Replace A(p) as ρ

$$\rho(p_i) = \sum_j \rho(p_j)\, m_j / \rho_j \; W_h(\|p_i - p_j\|)$$

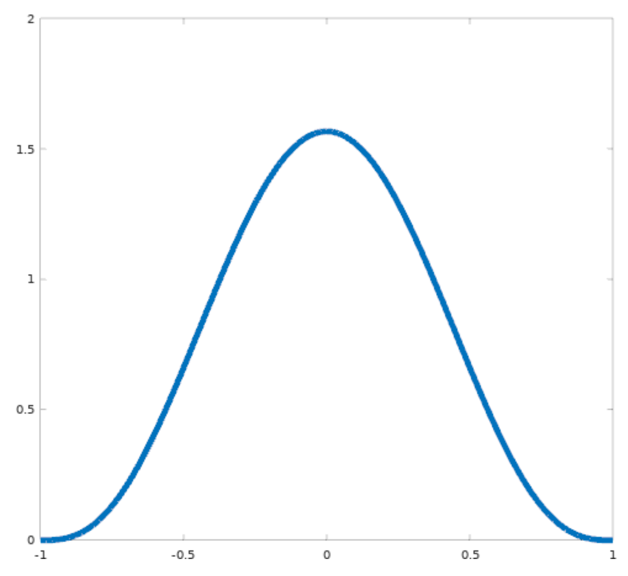
$$\Rightarrow \boxed{\rho_i = \sum_j m_j\, W_h(\|p_i - p_j\|)}$$



Choice of weight functions

Use a smooth polynomial:

ex. $W_h^{poly6}(d) = \frac{315}{64\,\pi\,h^9}\,(h^2 - d^2)^3 \quad 0 \leq d \leq h$



Pressure

$$F_{pressure} = -\frac{m_i}{\rho_i} \nabla \mathrm{p}_i$$

1. Use symetric gradient b/w (i,j) $F_{pressure} = -\frac{m_i}{\rho_i} \nabla(\mathrm{p}_i + \mathrm{p}_j)/2$

$$F_{pressure} = -\frac{m_i}{\rho_i} \sum_j m_j \frac{\mathrm{p}_i + \mathrm{p}_j}{2\,\rho_j} \nabla W_h(\|p_i - p_j\|)$$

2. Express the pressure as a function of the density ρ

   Simple approximation: $\mathrm{p}_i = s\,(\rho_i - \rho_0)$

   - s: Stiffness property
   - ρ_0: Rest density of the fluid

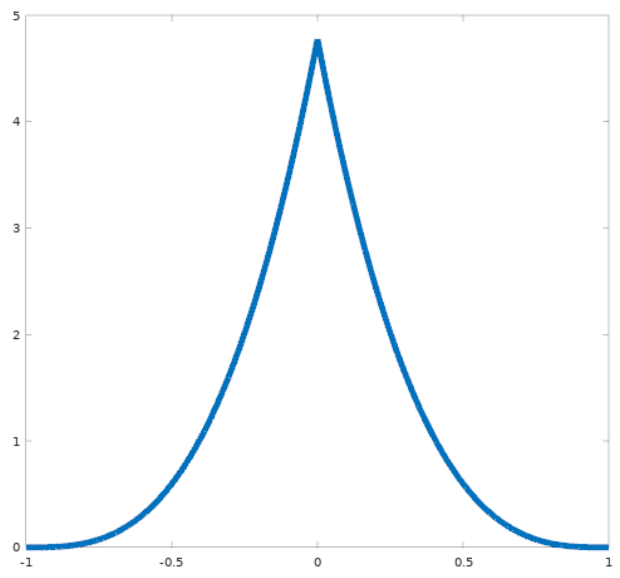
3. Weight function

   Pressure is used to avoid particles to group together

   Avoid local maxima $\Rightarrow$ non smooth "spiky" function at 0

$$W_h^{spiky}(d) = \frac{15}{\pi\, h^6}(h-d)^3 \quad 0 \leq d \leq h$$

$$\nabla W_h^{spiky}(p_i - p_j) = -\frac{45}{\pi\, h^6}(h - \|p_i - p_j\|)^2 \frac{p_i - p_j}{\|p_i - p_j\|} \quad 0 \leq \|p_i - p_j\| \leq h$$



Viscosity

$$F_{viscosity} = m_i \, \nu \, \triangle v_i$$

1. Use symetric laplacian b/w (i,j)

   $F_{viscosity} = m_i \, \nu \, \triangle(\mathrm{v}_j - \mathrm{v}_i)$ - *viscosity depends on velocity differences*

   $F_{viscosity} = m_i \, \nu \, \sum_j m_j \frac{(\mathrm{v}_j - \mathrm{v}_i)}{\rho_j} \, \triangle W_h(\|p_i - p_j\|)$

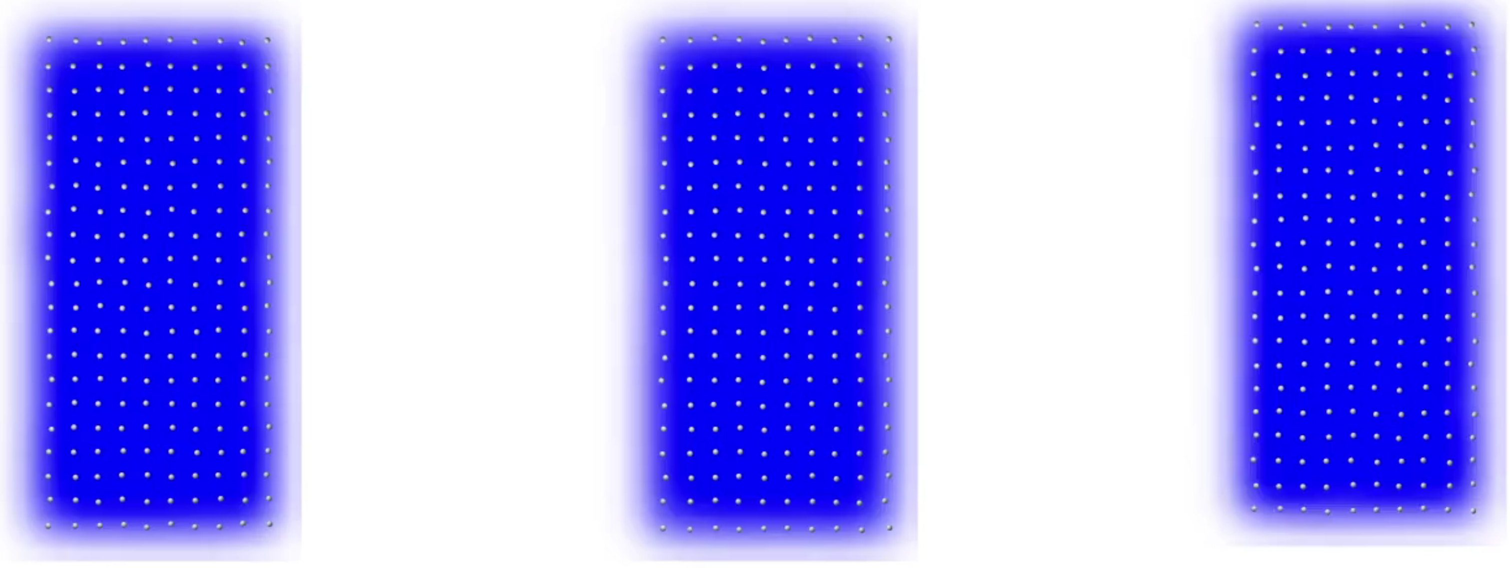
2. Weight function

   Second derivative should remain positive

   Can use the spiky kernel

   $W_h^{spiky}(d) = \frac{15}{2\pi h^6}(h-d)^3 \quad 0 \leq d \leq h$

   $\triangle W_h^{spiky}(d) = \frac{45}{\pi h^6}(h-d) \quad 0 \leq d \leq h$



Increasing viscosity ν

SPH Summary

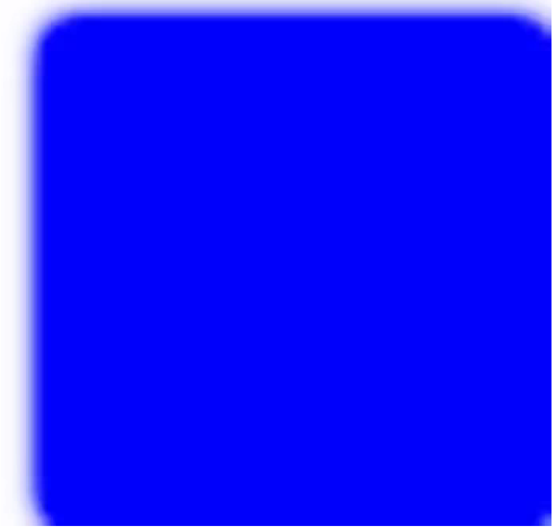
Set initial conditions v_i

Compute values

- Density: $\rho_i = \sum_j m_j \, W_h^{poly6}(\|p_i - p_j\|)$
- Pressure: $\mathrm{p}_i = s\,(\rho_i - \rho_0)$

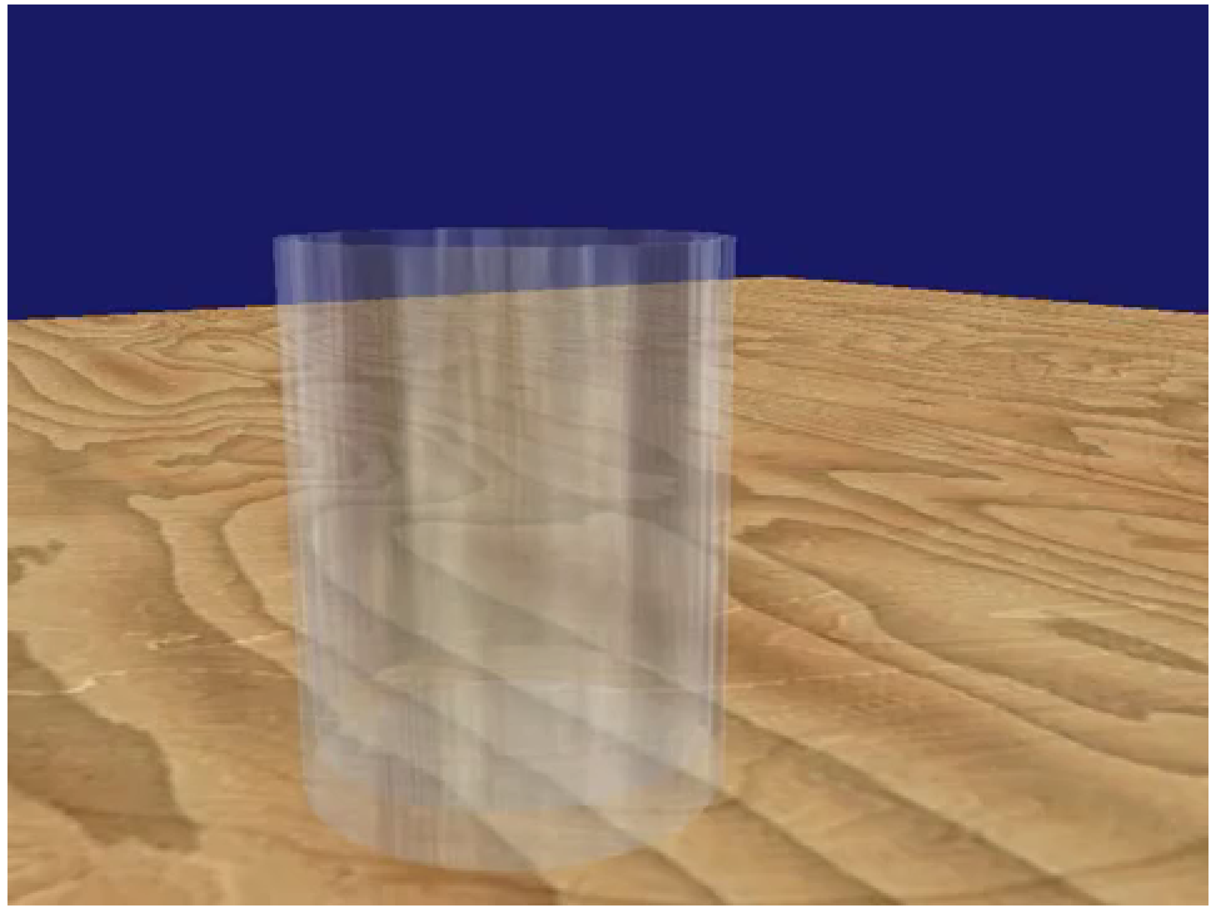
Compute forces

- $F_{weight} = m_i \; g$
- $F_{pressure} = -\frac{m_i}{\rho_i} \; \sum_j^{j\neq i} m_j \, \frac{\mathrm{p}_i + \mathrm{p}_j}{2\,\rho_j} \; \nabla W_h^{spiky}(\|p_i - p_j\|)$
- $F_{viscosity} = m_i \, \nu \, \sum_j^{j\neq i} m_j \, \frac{(\mathrm{v}_j - \mathrm{v}_i)}{\rho_j} \; \triangle W_h^{spiky}(\|p_i - p_j\|)$



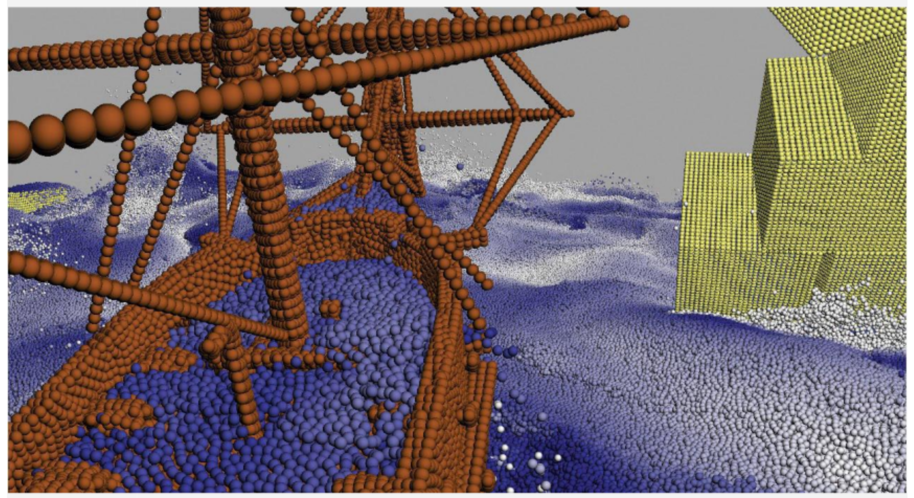
Time integration: $v_i^{k+1} = v_i^k + \Delta t \; \left(F_{weight} + F_{pressure} + F_{viscosity}\right)/m_i$

SPH examples



Muller 2003

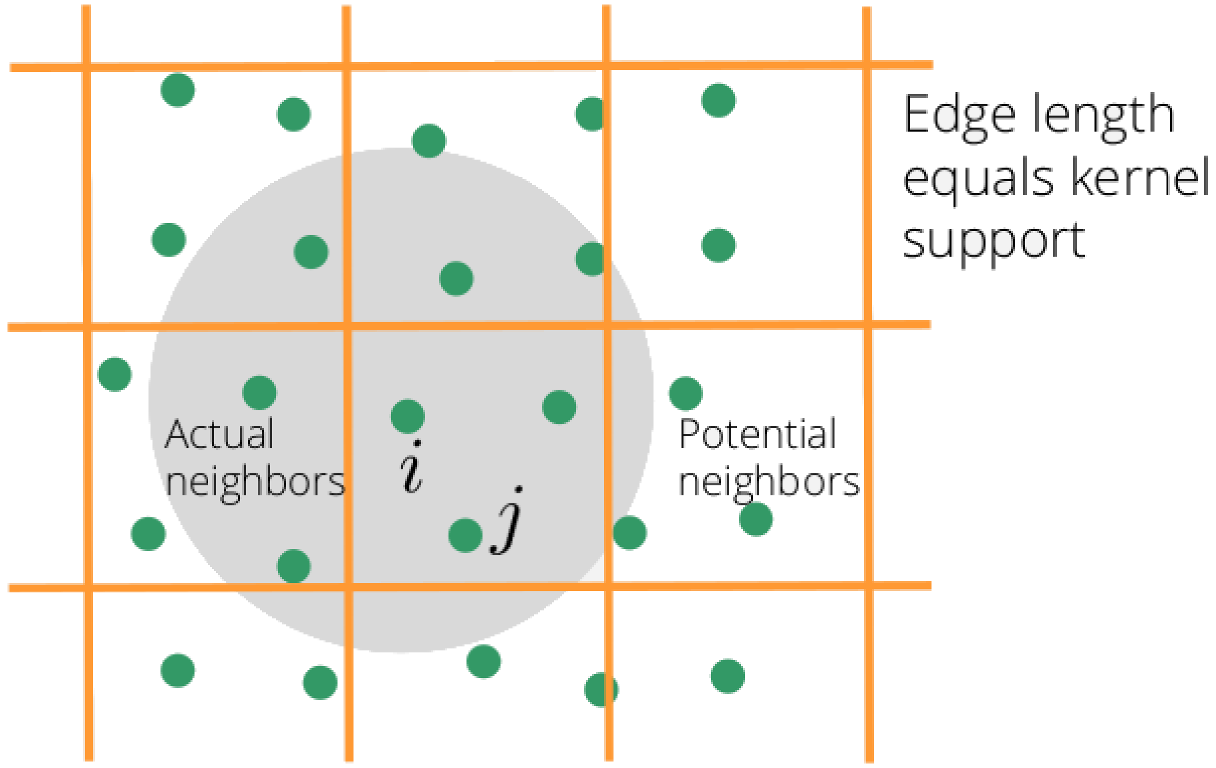
M. Teschner 2012 - 20M particles

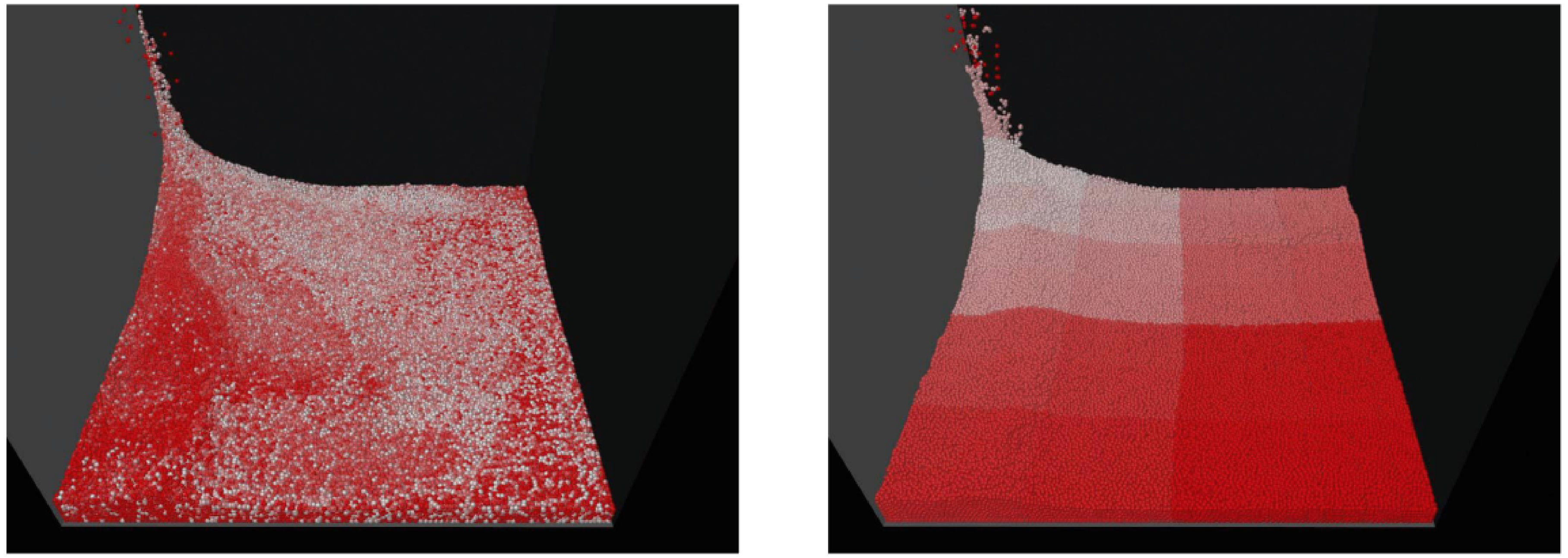


Acceleration structure

SPH based on pair-wise interation $\Rightarrow$ spatial sorting acceleration structure

- Uniform grid: simple and efficient.
- Verlet lists (wider neighborhood, updated every n steps only)
- List of vertices per cell, hash table for cell storage
- Spatial sorting for cache efficiency





M. Teschner

SPH extensions

(+) Very versatile (interaction between any deforming shapes)

Not only fluids

(-) Not well understood accuracy

(-) Compressible

[Solenthaler et al., Predictive-Corrective Incompressible SPH, ACM SIGGRAPH 2009]

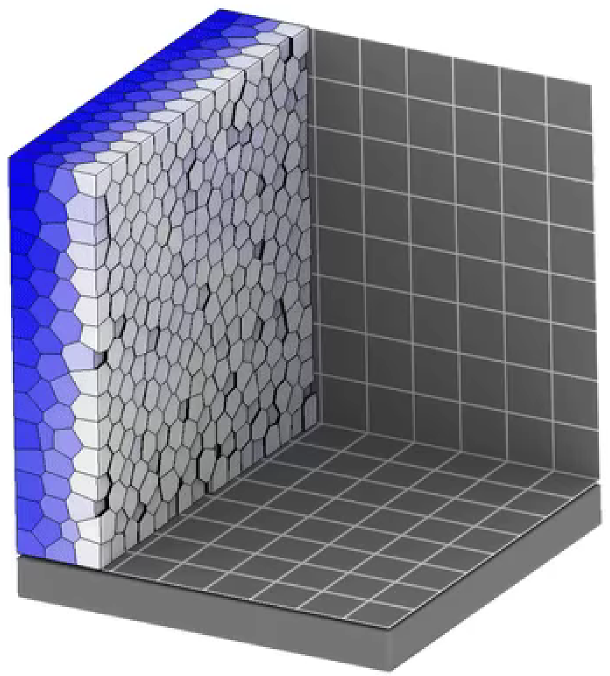
[Ihmsen et al, Implicit Incompressible SPH, IEEE TVCG 2013]

(-) Limited time step

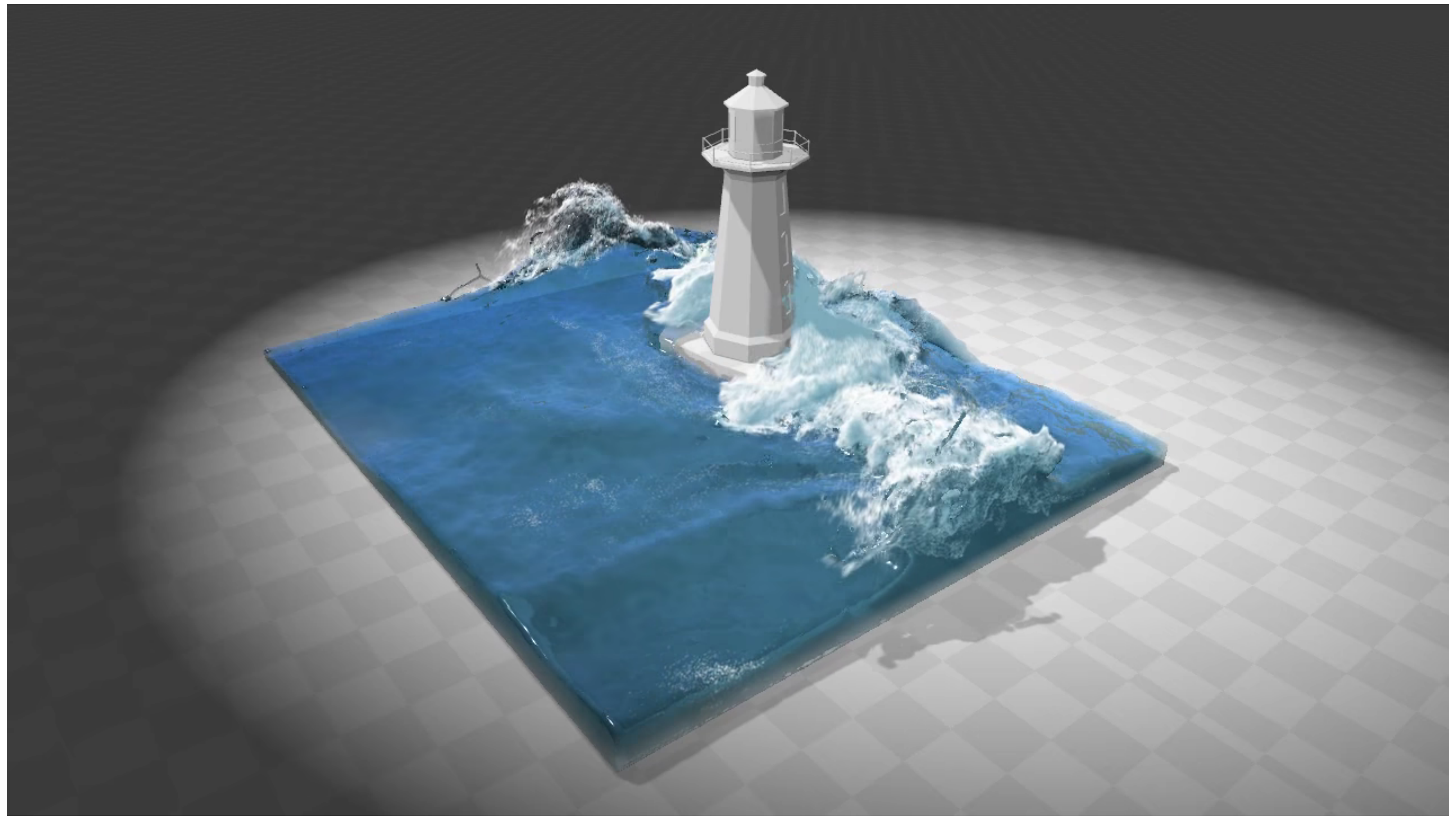
[Macklin and Muller, Position based Fluids, ACM SIGGRAPH 2013]

(-) Boundaries are hard to handle

[Brand et al., Pressure Boundaries for Implicit Incompressible SPH, ACM TOG 2018]



Bruno Levy



[Macklin and Muller 2013] , *[Yu and Turk 2009]*