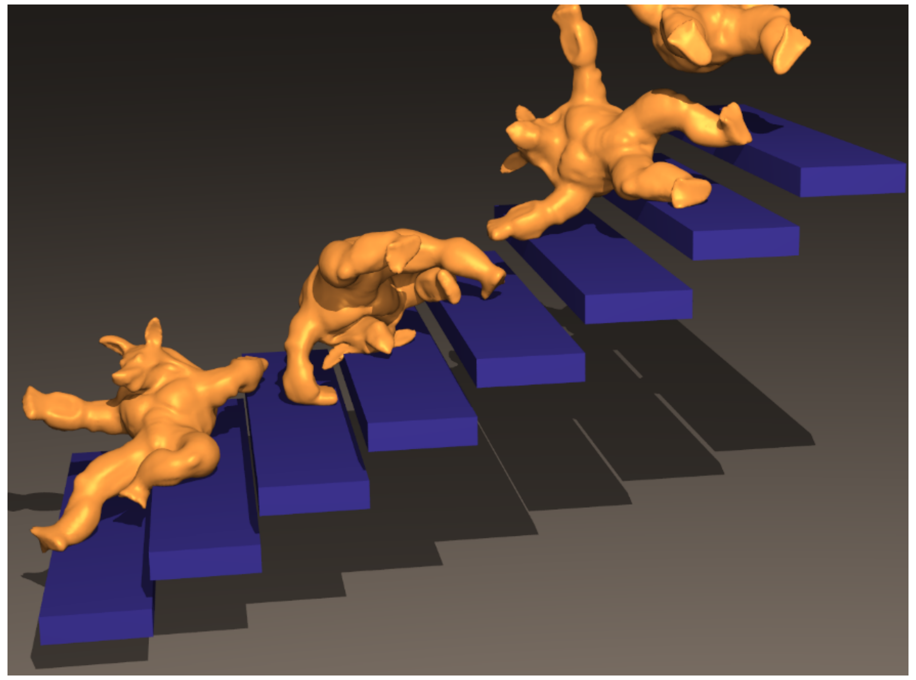
Fundamental models

1- Particles
2- Rigid bodies
3- Continuum material

Deformation of a continuous shape



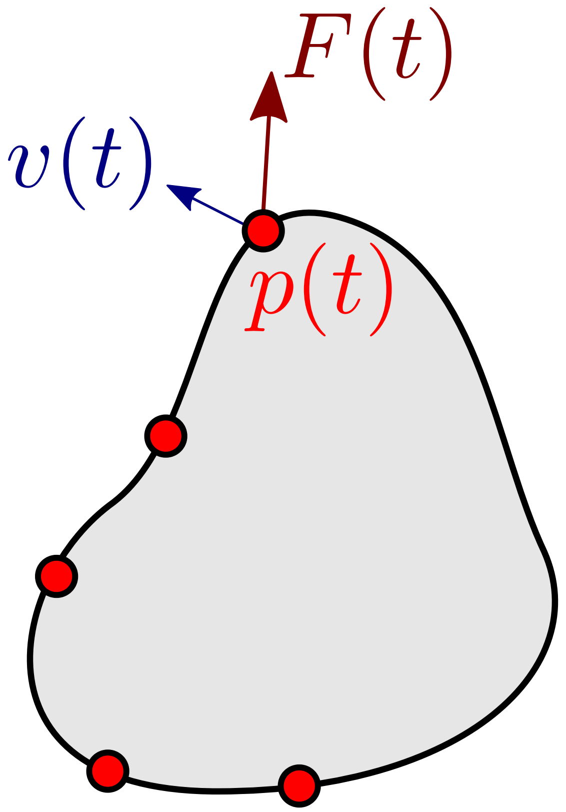
Every part of the shape can be deformed

ex. Describing elastic shapes, visco-elastic shapes, fluids, etc.

Two ways to describe the deforming object

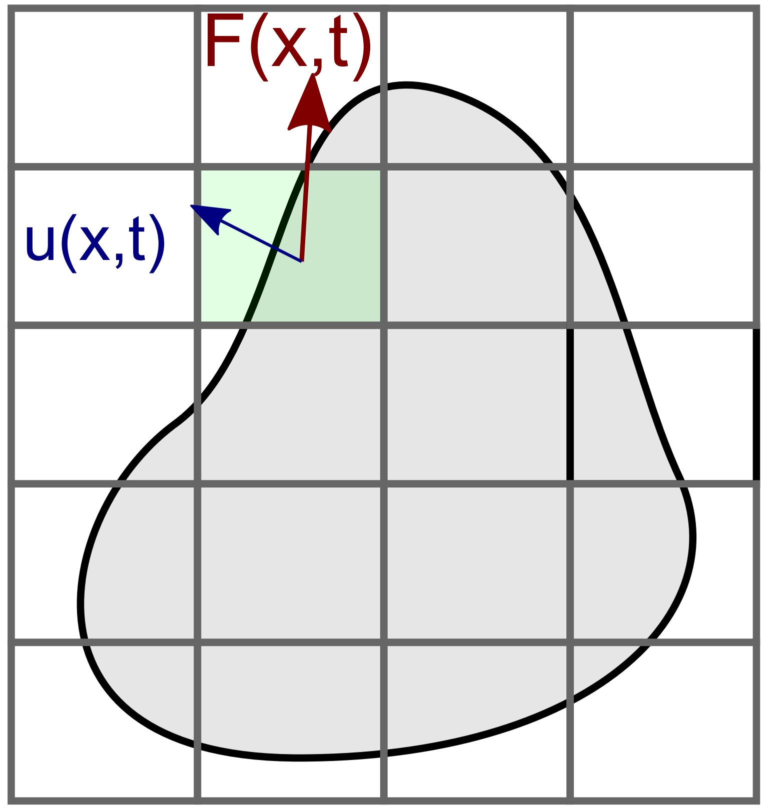
1. Lagrange representation

Positions follow the object deformation



2. Euler representation

Positions are fixed in 3D space

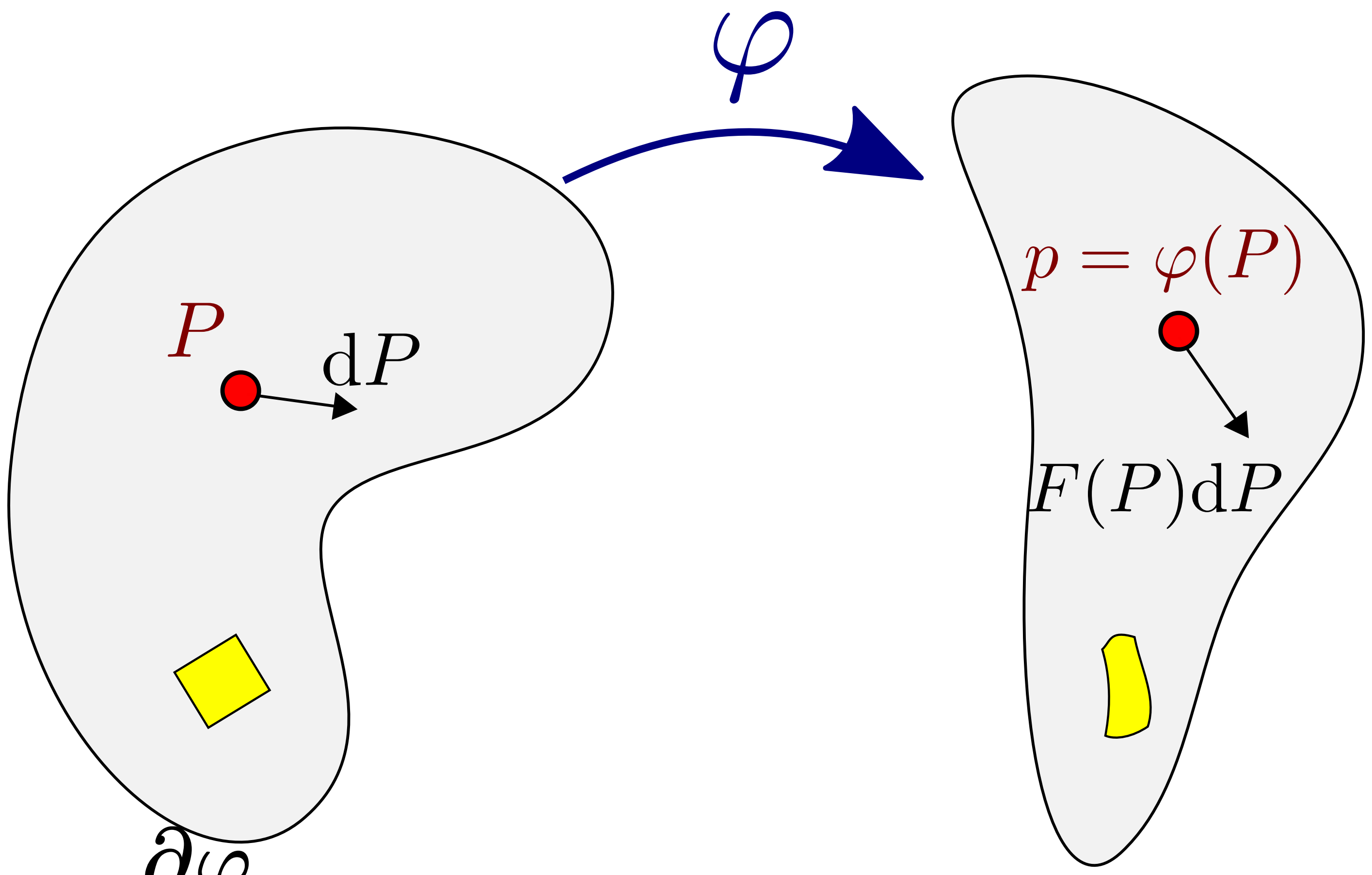


Deformation the Lagrangian description

Deformation map $\varphi : \mathbb{R}^3 \to \mathbb{R}^3$ such that $p = \varphi(P)$

P position in the reference undeformed shape

p position in the deformed configuration.



Deformation Gradient F

- $F(P) = \dfrac{\partial \varphi}{\partial P}(P) = \dfrac{\partial p}{\partial P} \in \mathbb{R}^{3\times 3}$
- Characterizes the local deformation associated to φ

Position $P + \mathrm{d}P$ is mapped into $\varphi(P + \mathrm{d}P) \simeq p + \dfrac{\partial \varphi}{\partial P}\,\mathrm{d}P$

- $F(P) = \begin{pmatrix} \frac{\partial \varphi_x}{\partial X} & \frac{\partial \varphi_x}{\partial Y} & \frac{\partial \varphi_x}{\partial Z} \\ \frac{\partial \varphi_y}{\partial X} & \frac{\partial \varphi_y}{\partial Y} & \frac{\partial \varphi_y}{\partial Z} \\ \frac{\partial \varphi_z}{\partial X} & \frac{\partial \varphi_z}{\partial Y} & \frac{\partial \varphi_z}{\partial Z} \end{pmatrix}$

Strain

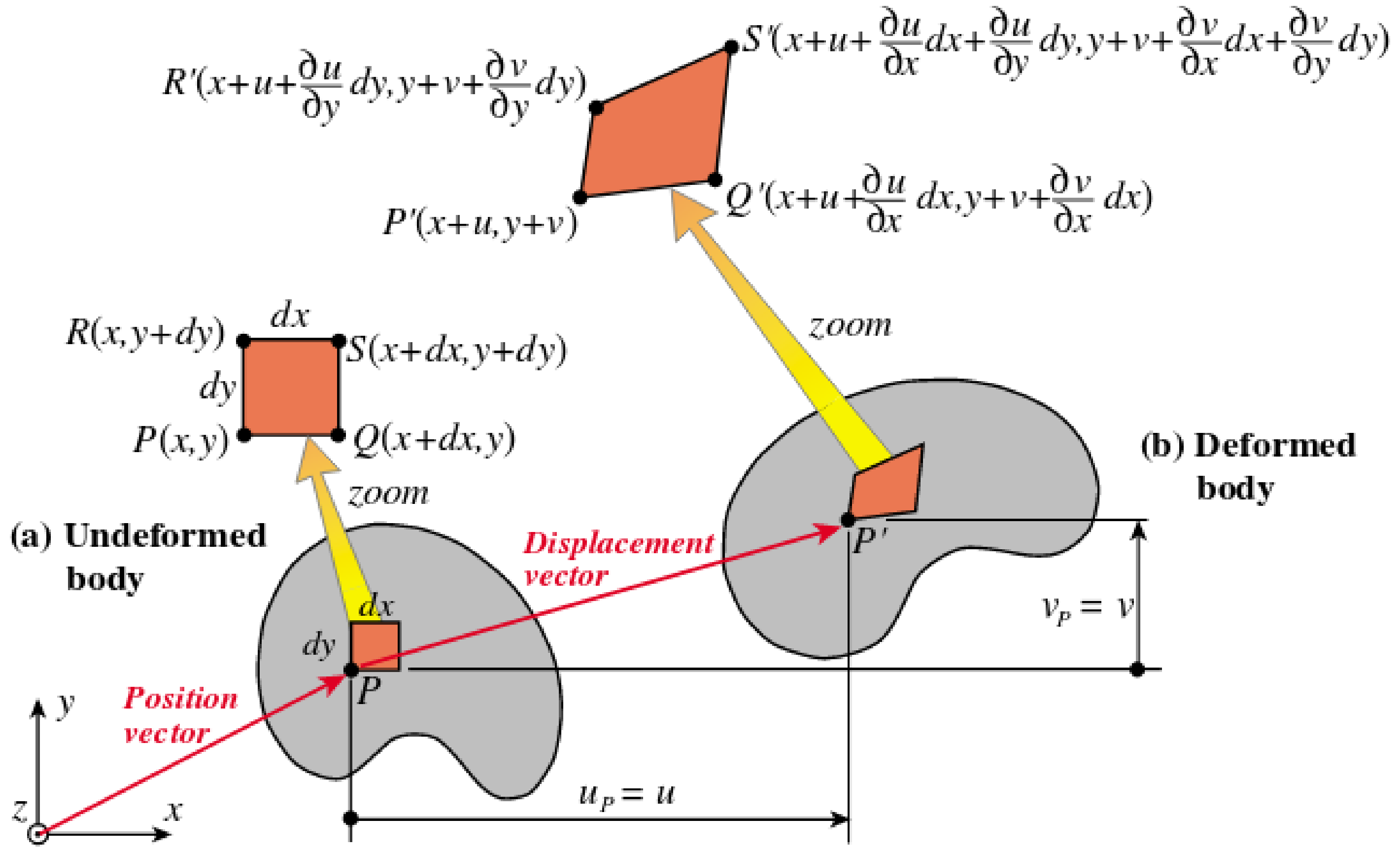
Deformation gradient F describe both

- Rigid transformation (rotation) - not related to material effort
- Any other deformation inducing local length change - related to material effort

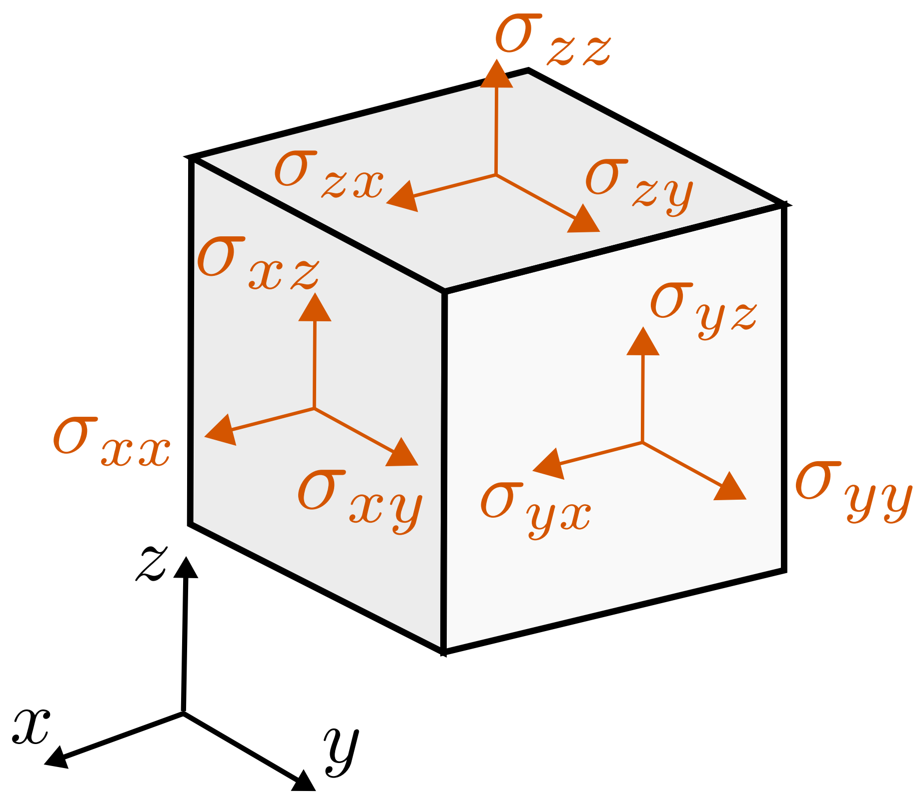
Strain ϵ is a measure of deformation ignoring rigid transformation.

Several possible measure of strain

- Green strain tensor $\epsilon = \frac{1}{2}\left(F\,F^T - \mathrm{Id}\right)$
  - *(+) If φ is a rotation $F = R \Rightarrow \epsilon = 0$*
  - *(-) Non linear in p*
- Linearized Cauchy strain $\epsilon = \frac{1}{2}(F^T + F) - \mathrm{Id}$
  - *Used for small deformations*



Stress

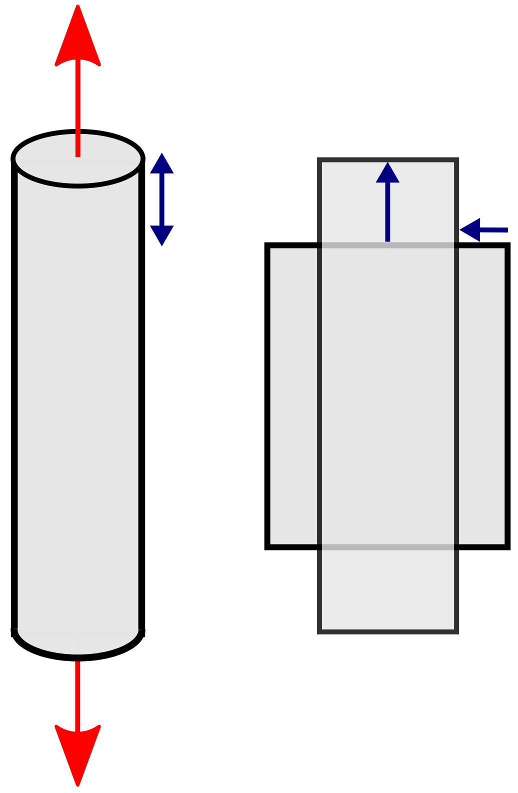


Stress $\sigma \in \mathbb{R}^{3\times 3}$ describes internal forces (per area unit) induced by the local deformation (strain) in any direction
Constitutive Relation: Relation between stress and strain, characterize a type of material.

For linear constitutive relation:

$$\sigma_{ij} = \sum_{k,l} C_{ijkl}\,\epsilon_{kl} \quad , \quad C\text{: stiffness tensor (81 coefficients)}$$

Strain energy/elastic potential energy: $U = \frac{1}{2}\sum_{i,j,k,l}\sigma_{ij}(\epsilon)\epsilon_{kl} = \frac{1}{2}\sum_{i,j,k,l} C_{ijkl}\,\epsilon_{ij}\,\epsilon_{kl}$



For homogeneous isotropic elastic material, constitutive relation simplifies to

$\sigma = 2\mu\,\epsilon + \lambda \mathrm{tr}(\epsilon)\,\mathrm{Id}$, (μ, λ): Lamé parameters

Related to common mechanical modulus : Young' modulus Y and Poisson's ratio ν

$$\mu = \frac{Y}{2(1+\nu)}, \lambda = \frac{Y\nu}{(1+\nu)(1-2\nu)}$$

Evolution equation

Fundamental principle of dynamics in the entire volume Ω

Change of momentum = External forces (in volume) + Traction (stress applied on exterior surface normals)

$$\Rightarrow \quad \underbrace{\int_{\Omega} \rho\, p''(t)\, \mathrm{d}\Omega}_{Change\ of\ momentum} \;=\; \underbrace{\int_{\Omega} F(t)\, \mathrm{d}\Omega}_{External\ forces} \;+\; \underbrace{\int_{\partial\Omega} \sigma\, n\, \mathrm{d}S}_{Traction\ force\ on\ the\ boundary}$$

Using divergence theorem $\displaystyle\int_{\partial\Omega} \sigma\, n\, \mathrm{d}S = \int_{\Omega} div(\sigma)\, \mathrm{d}\Omega$

Equation in volume satisfied at each position $p \in \Omega$

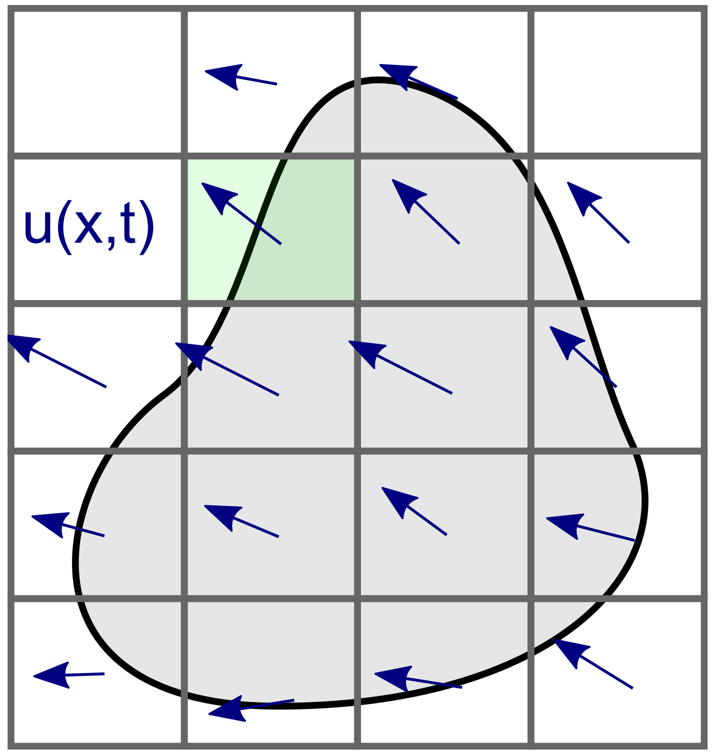
$$\boxed{\rho\, p''(t) = F(t) + div(\sigma(t))} \quad \sigma = \begin{pmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{pmatrix} \quad div(\sigma) = \begin{pmatrix} \frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{yx}}{\partial y} + \frac{\partial \sigma_{zx}}{\partial z} \\ \frac{\partial \sigma_{xy}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{zy}}{\partial z} \\ \frac{\partial \sigma_{xz}}{\partial x} + \frac{\partial \sigma_{yz}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z} \end{pmatrix}$$

Euler formulation

In Euler formulation quantities are expressed at fixed position in 3D space.

Deformation described by velocity $u(p,t)$ at a given 3D fixed point $p = (x, y, z)$ at time t.

- Do not require anymore a reference shape
- Usefull for heavily deforming shapes (ex. fluids, gaz).



- Change of speed during $\mathrm{d}t$

$$\frac{\mathrm{d}u}{\mathrm{d}t}(p,t) = \frac{\partial u}{\partial t} + \sum_i \frac{\partial u}{\partial p_i} \underbrace{\frac{\mathrm{d}p_i}{\mathrm{d}t}}_{u_i} = \frac{\partial u}{\partial t} + (u \cdot \nabla) u$$

Called *material derivative*.

- Similarily to Lagrangian derivation:
  - Strain-rate tensor ϵ (rate of change of deformation in a neighborhood of a point) expressed with respect to u: $\epsilon = \frac{1}{2}\left(\nabla u + \nabla u^T\right)$
  - Stress-rate tensor σ (rate of change of direction force per area in a neighborhood of a point).

Equation of motion for a fluid

- Fundamental principle of dynamics on linear momentum

$$\rho \frac{\mathrm{d}u}{\mathrm{d}t} = F + \mathrm{div}(\sigma)$$

$$\Rightarrow \rho \frac{\partial u}{\partial t} = F + \mathrm{div}(\sigma) - \rho\,(u \cdot \nabla)u.$$

The term $(u \cdot \nabla)u$ is called *advection*.

- External force: weight $F = \rho\, g$
- Stress decomposed into

$$\sigma = \sigma_{viscous} + \sigma_{pressure}$$

$\sigma_{pressure} = -p\,\mathrm{Id}$ *(pressure acts along normal of surface elements)*

$$\rho \frac{\partial u}{\partial t} = \rho\, g - \rho\, u \cdot \nabla u + \mathrm{div}\left(\sigma_{viscous} - p\,\mathrm{Id}\right)$$

$$\Rightarrow \rho \frac{\partial u}{\partial t} = \rho\, g - \rho\, u \cdot \nabla u - \nabla p + \mathrm{div}\left(\sigma_{viscous}\right)$$

Navier-Stokes equation

- Isotropic Newtonian fluid $\Rightarrow$ Linear (scalar) relation between strain-rate ϵ and stress-rate $\sigma_{viscous}$
  - $\sigma_{viscous} = 2\mu\,\epsilon = \mu\left(\nabla u + \nabla u^T\right)$, μ constant viscosity parameter
- Incompressible fluid $\Rightarrow \mathrm{div}(u) = 0$

Equation of motion

$$\Rightarrow \rho\,\frac{\partial u}{\partial t} = \rho\,g - \rho\,u \cdot \nabla u - \nabla p + \mathrm{div}\left(\mu\left(\nabla u + \nabla u^T\right)\right)$$

- Noting that $\mathrm{div}(\nabla u^T) = \nabla\,\mathrm{div}(u) = 0$
- And $\mathrm{div}(\nabla u) = \triangle u$
- Set $\nu = \mu/\rho$

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

Navier-Stokes equation for incompressible Newtonian fluid.

Animating fluids (I)

Stable Fluid

Solving Navier-Stokes on grid

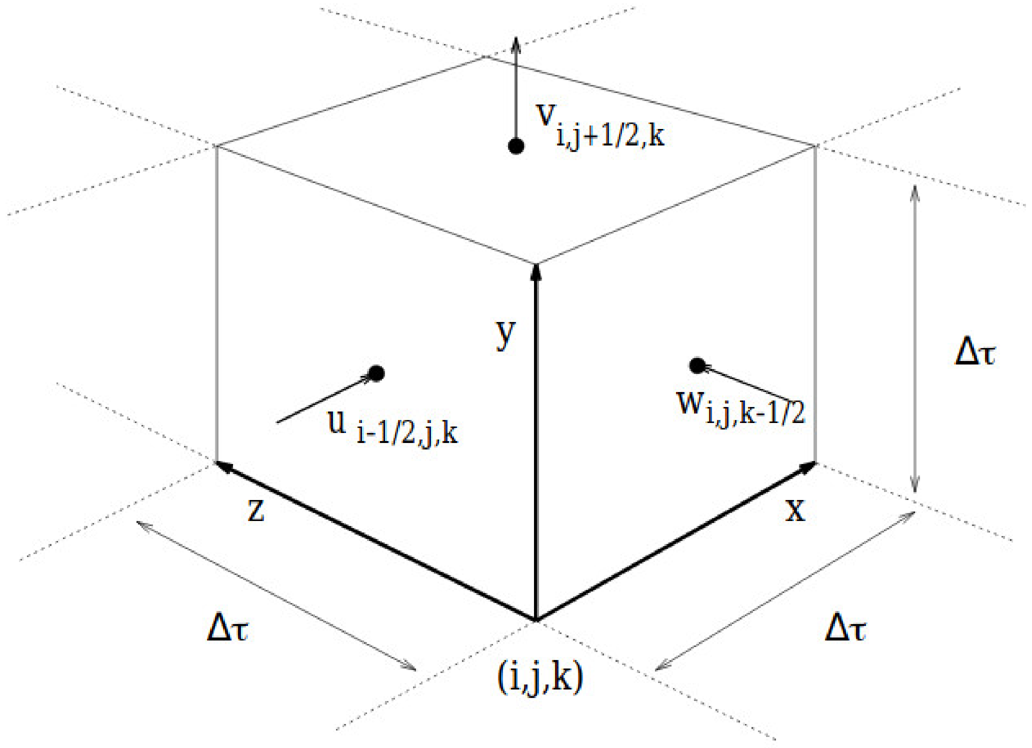
"Brute force" approach

- Rectangular grid filled with fluid
- Use finite differences on the grid for Navier-Stokes equation

$$\frac{\partial u}{\partial t} = -\frac{1}{\rho}\nabla \mathrm{p} + f - (u \cdot \nabla)u + \nu \triangle u$$
$$div(u) = 0$$

(-) Stability conditions

(-) Loose advection details on the grid



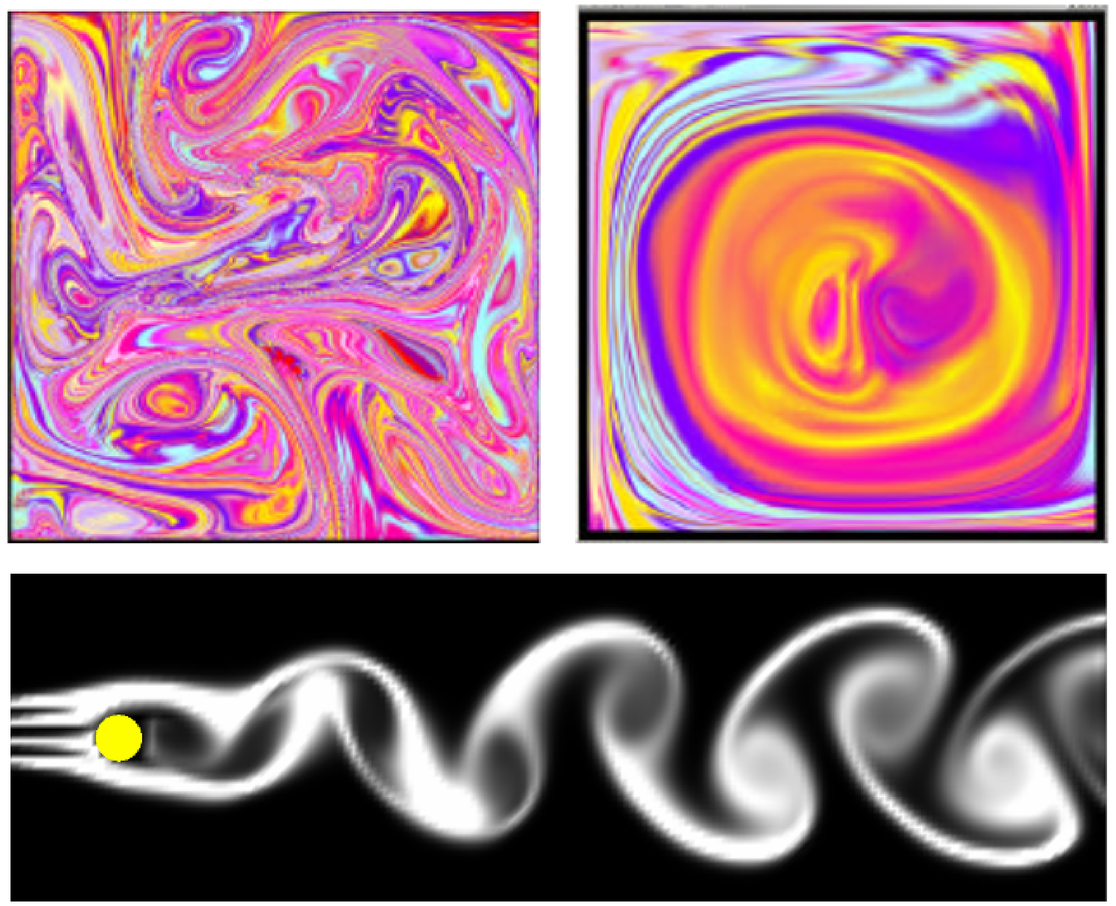
[Modeling the Motion of a Hot, Turbulent Gas. N Foster and D. Metaxas. SIGGRAPH 1997]

Stable Fluids - Idea

Well known improvement: **Jos Stam, Stable Fluids, ACM SIGGRAPH 1999**

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

- $1/\rho\ \nabla \mathrm{p}$: Pressure term only used to ensure divergence free
- Similar to Lagrange multipler for constraints



[Stable Fluids. J. Stam. SIGGRAPH 1999]
[Real Time Fluid Dynamics for Games. J. Stam. Game Dev. Conf. 2003]

1st Idea

Remove pressure term

Replace by explicit projection on divergence free vector field P

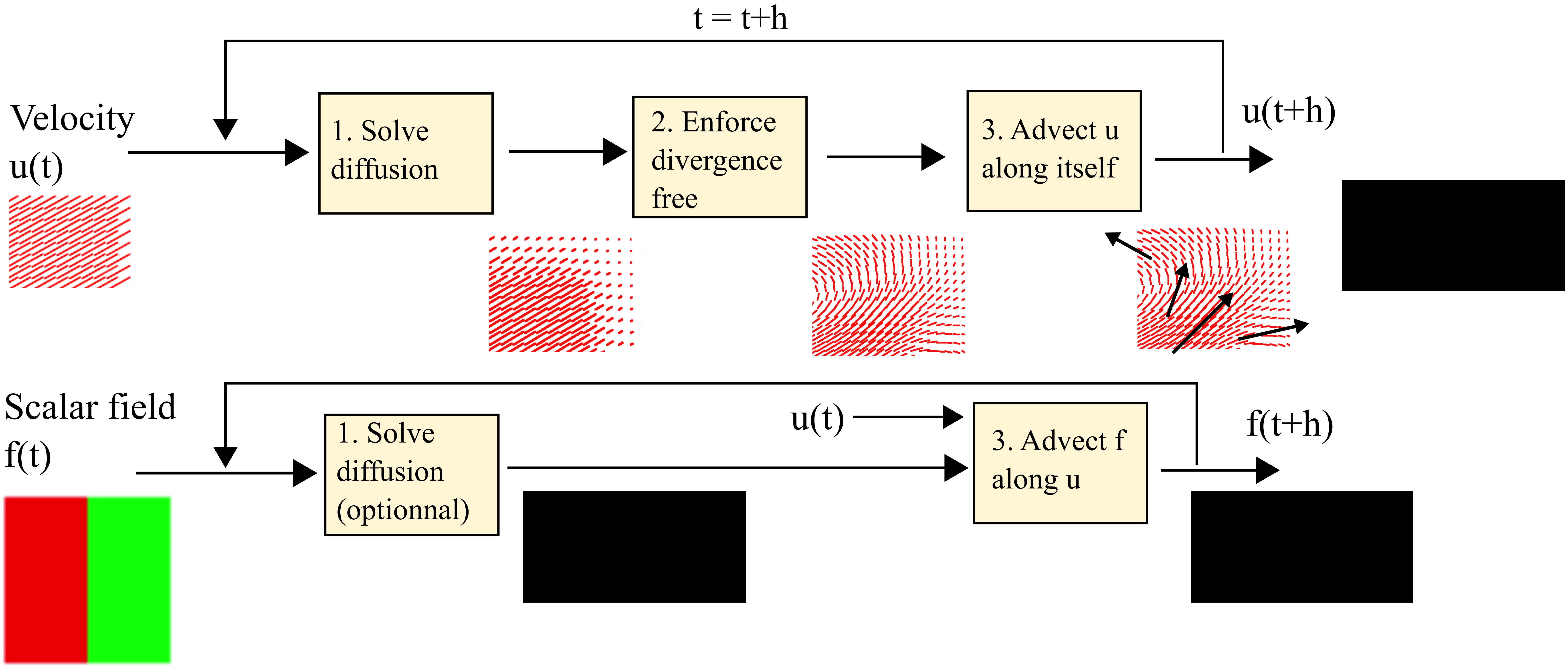
$$\Rightarrow \frac{\partial u}{\partial t} = \mathrm{P}(f - (u \cdot \nabla)u + \nu \triangle u)$$

2nd Idea

Splitting: Compute each terms one after the other

$$u^k \underbrace{\rightarrow}_{\substack{addforces \\ f}} u_1^k \underbrace{\rightarrow}_{\substack{diffuse \\ \nu\triangle u}} u_2^k \underbrace{\rightarrow}_{\substack{project \\ \mathrm{P}}} u_3^k \underbrace{\rightarrow}_{\substack{advect \\ (u\cdot\nabla)u}} u^{k+1}$$

Stable Fluids - General Algorithm



1 - Diffusion

Use finite difference on $\frac{\partial f}{\partial t} = \nu \triangle f$

Notation: $f^t_{x,y} = f(k_x \, \Delta x, \; k_y \, \Delta y, \; k_t \, \Delta t)$

Explicit schemes may oscillates/diverge for large time steps

$\Rightarrow$ Use implicit scheme for unconditional stability

$$\frac{f^{k+1}_{x,y} - f^{k}_{x,y}}{\Delta t} = \nu \left(\frac{f^{k+1}_{x+1,y} - 2\, f^{k+1}_{x,y} + f^{k+1}_{x-1,y}}{(\Delta x)^2} + \frac{f^{k+1}_{x,y+1} - 2\, f^{k+1}_{x,y} + f^{k+1}_{x,y+1}}{(\Delta y)^2} \right)$$

Assuming $\Delta x = \Delta y = 1$

$$(1 + 4\, \nu \Delta t)\, f^{k+1}_{x,y} - \nu\, \Delta t (f^{k+1}_{x+1,y} + f^{k+1}_{x-1,y} + f^{k+1}_{x,y+1} + f^{k+1}_{x,y-1}) = f^{k}_{x,y}$$

Use Gauss-Seidel iterative method to solve the sparse linear system

Initialize $f^{k+1} = f^k$

for $i = 1..N_{\text{max}}$

$$f^{k+1}_{x,y} = \tfrac{1}{1+4\,a} (f^{k}_{x,y} + a(f^{k+1}_{x-1,y} + f^{k+1}_{x+1,y} + f^{k+1}_{x,y-1} + f^{k+1}_{x,y+1}))\ , \quad a = \nu\, \Delta t$$

2 - Advection

Advection = move some function along given velocity u.

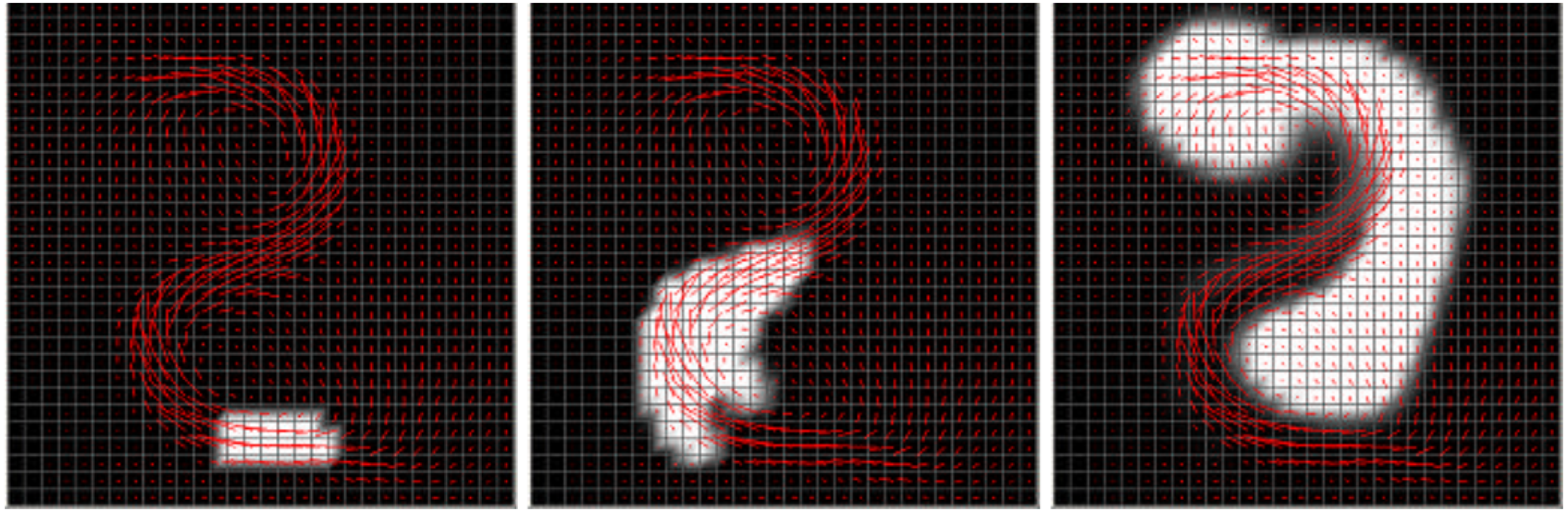
- Advecting a scalar field f along u

$$\frac{\partial f}{\partial t}(p,t) + u(p,t) \cdot \nabla f = 0$$

- Advecting a vector field f along u

$$\frac{\partial f}{\partial t}(p,t) + \left(u(p,t) \cdot \nabla\right) f = 0$$

- In Navier-Stokes advect the velocity itself $f = u$
- Can also advect density, color, texture coordinates, etc. to visualize the motion.



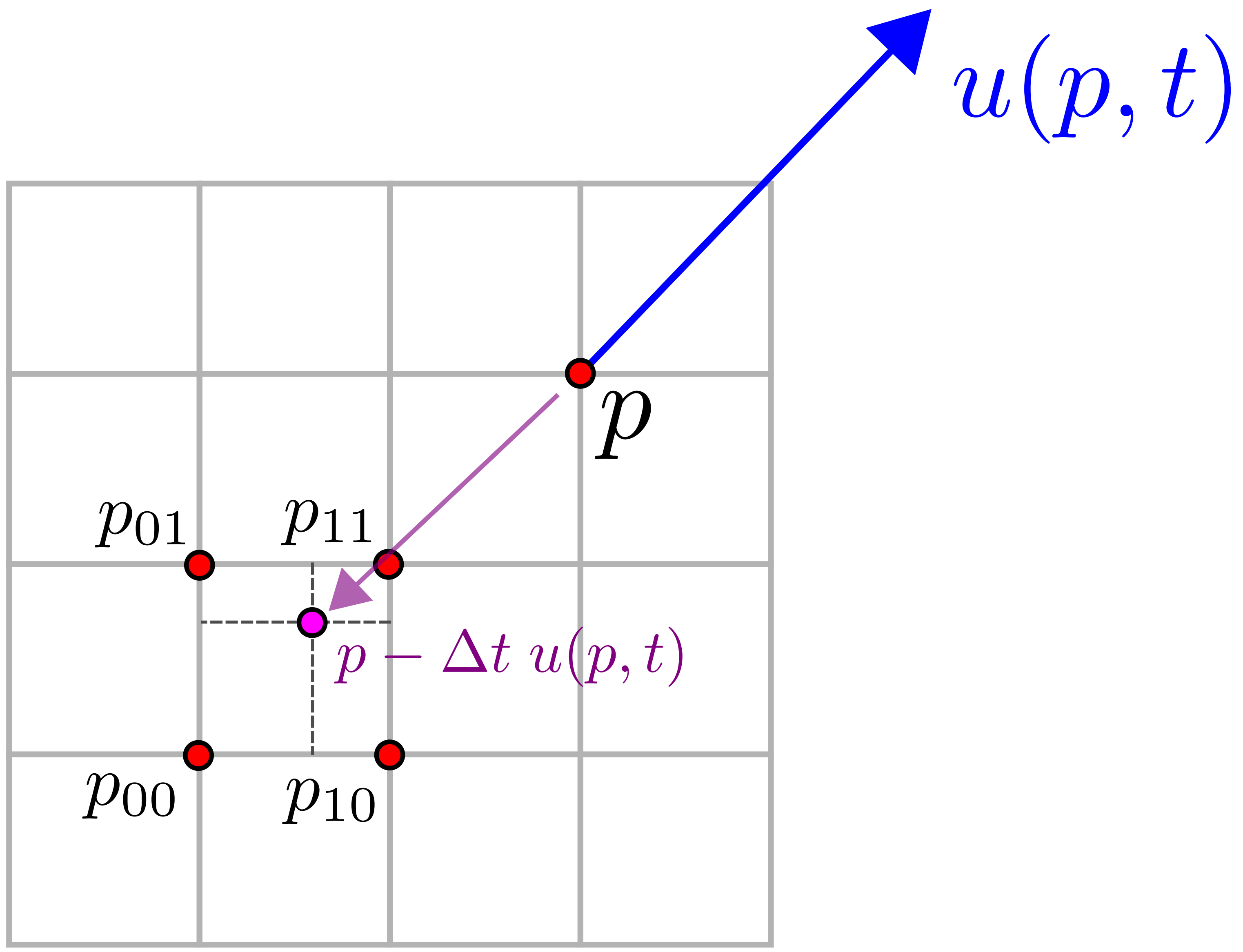
2 - Computing advection

Advecting generic value f along u

Idea Compute value of f at time t at fixed position grid p in moving back at $t - \Delta t$.

Value of f advected at point p at time t was at position $p_{prev} = p - \Delta t \; v(p,t)$ at time $t - \Delta t$.

$\Rightarrow f(p,t) = f(p_{prev}, t - \Delta t)$

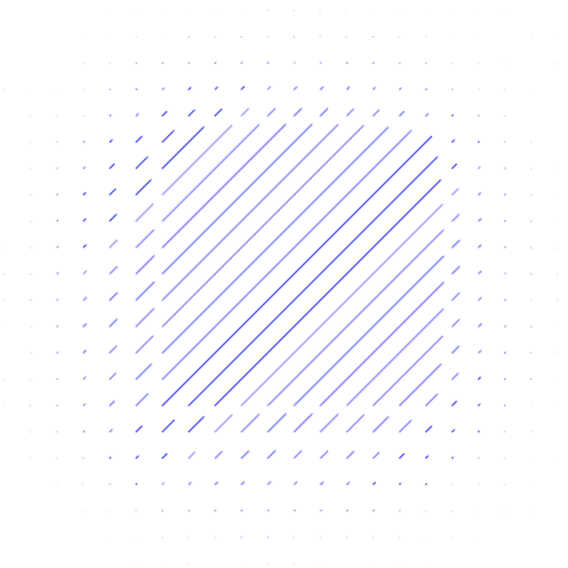


p_{prev} is not a grid point coordinates: Use interpolation
Can use Bilinear interpolation

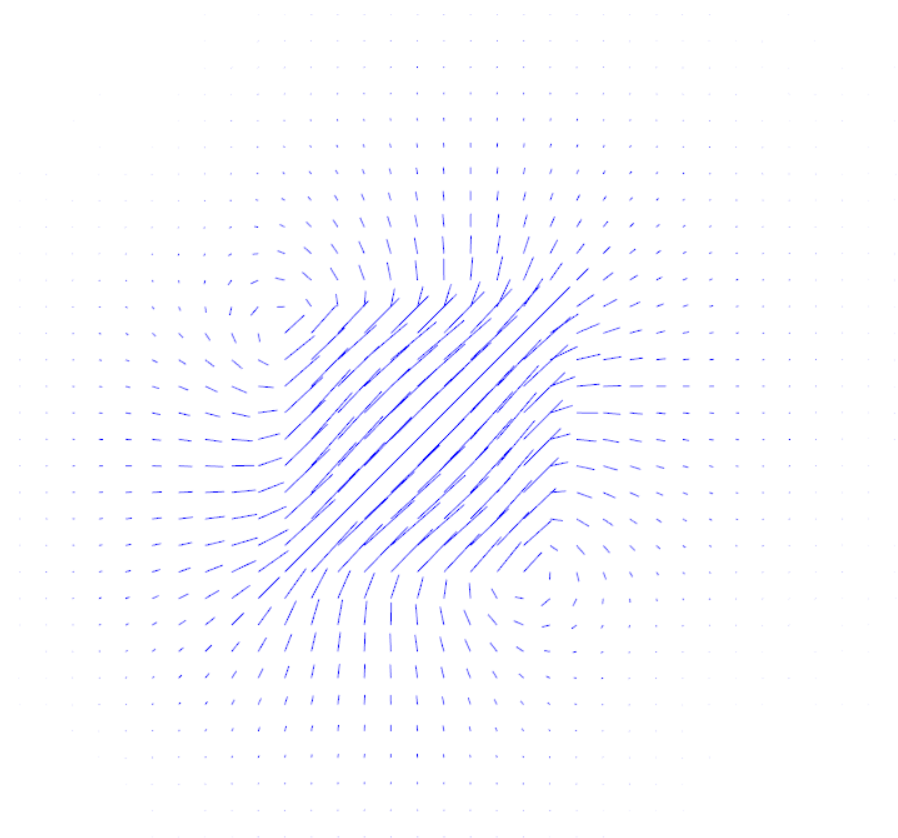
$$f(p_{prev}) = (1-\alpha)\,(1-\beta)\;p_{00} + (1-\alpha)\,\beta\,p_{01} + \alpha\,(1-\beta)\;p_{10} + \alpha\,\beta\;p_{11}$$

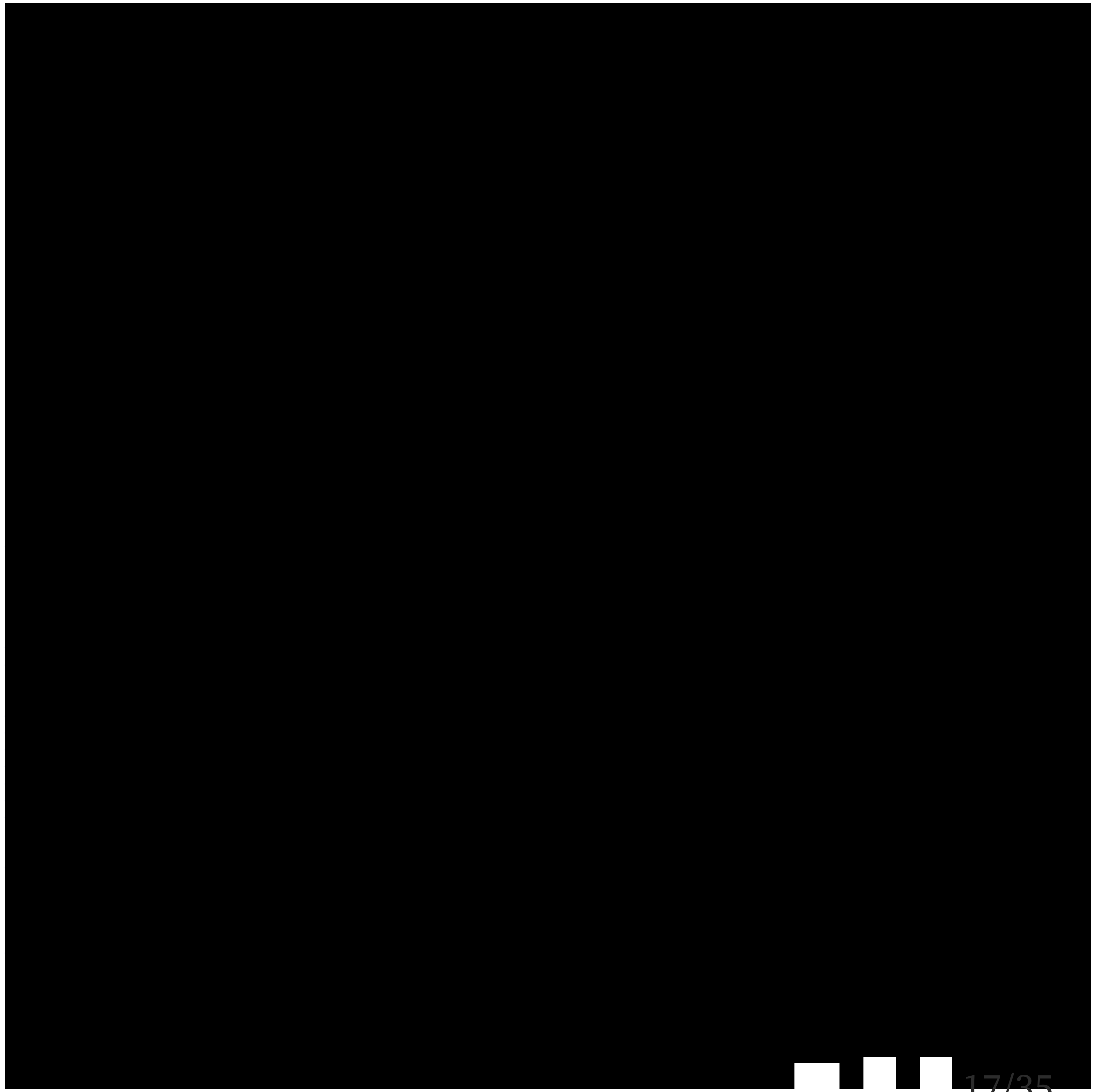
3 - Divergence Free Vector Field

Before projection:



After projection:





3 - Projection to divergence free vector field

Consider a general vector field w

Helmoltz decomposition: $w = u + v$

- u: Divergence free vector field such that $div(u) = 0$
- v: Gradient field $v = \nabla q$, q scalar field.

q satisfies a Poisson equation

$$div(w) = \underbrace{div(u)}_{=0} + div(v) \Rightarrow div(w) = \underbrace{div(\nabla q)}_{\triangle q}$$

Method- Given an input field w

1. Compute q as solution of $\triangle q = div(w)$
2. Compute $u = w - \nabla q$

3 - Projection to divergence free vector field (Algo)

Input vector field $w = (w^x, w^y)$

Note: we assume in the following $\Delta x = \Delta y = 1$

1 - Compute $d = \text{div}(w)$

$$d_{x,y} = (w^x_{x+1,y} - w^x_{x-1,y} + w^y_{x,y+1} - w^y_{x,y-1})/2$$

2 - Compute q in solving $\triangle q = d$

$$(q_{x+1,y} + q_{x-1,y} - 2\,q_{x,y}) + (q_{x,y+1} + q_{x,y-1} - 2\,q_{x,y}) = d_{x,y}$$
$$\Rightarrow 4q_{x,y} = q_{x+1,y} + q_{x-1,y} + q_{x,y+1} + q_{x,y-1} - d_{x,y}$$

ex. Numerical iterations using Gauss Seidel

Initialize $q = 0$
For $i = [1..N_{\max}]$
$$q_{x,y} = 1/4\,(q_{x+1,y} + q_{x-1,y} + q_{x,y+1} + q_{x,y-1} - d_{x,y})$$

3 - Compute $u = w - \nabla q$

$$u_{x,y} = w_{x,y} - (q_{x+1,y} - q_{x-1,y}\,,\; q_{x,y+1} - q_{x,y-1})\,/2$$

Handling boundaries

Boundaries $x = 0, x = N_x - 1, y = 0, y = N_y - 1$ need special care

- For density

  Assume value C^0 continuity on the boundary

  Row/Column $f_{x,0} = f_{x,1}$, $f_{0,y} = f_{1,y}$ etc.

- For velocity: $f = (f^x, f^y)$

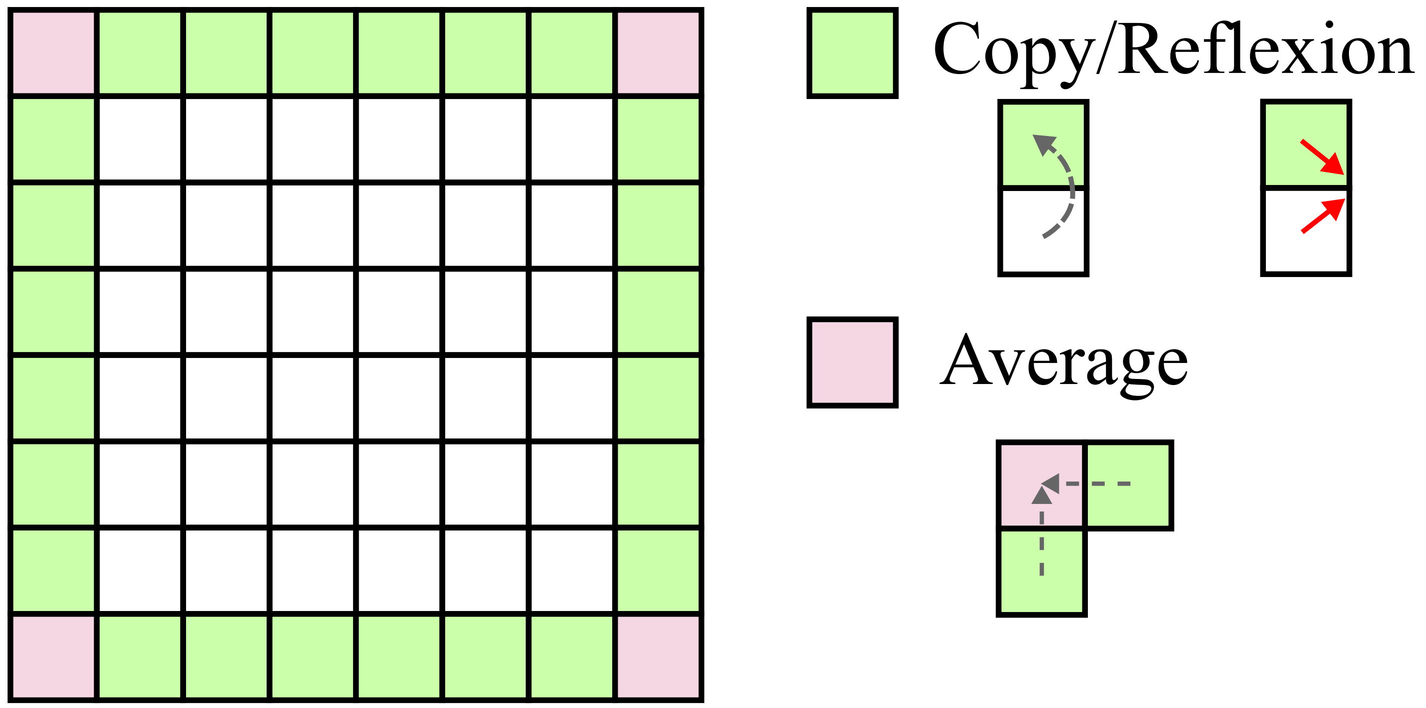
  Assume reflexion on walls

  Row: $f_{x,0} = (f^x_{x,1}, -f^y_{x,1})$

  Column: $f_{0,y} = (-f^x_{1,y}, f^y_{1,y})$

- In all cases: Average value for corners

  $f_{0,0} = (f_{1,0} + f_{0,1})/2$, etc.



Stable fluids example

Animating fluids

SPH

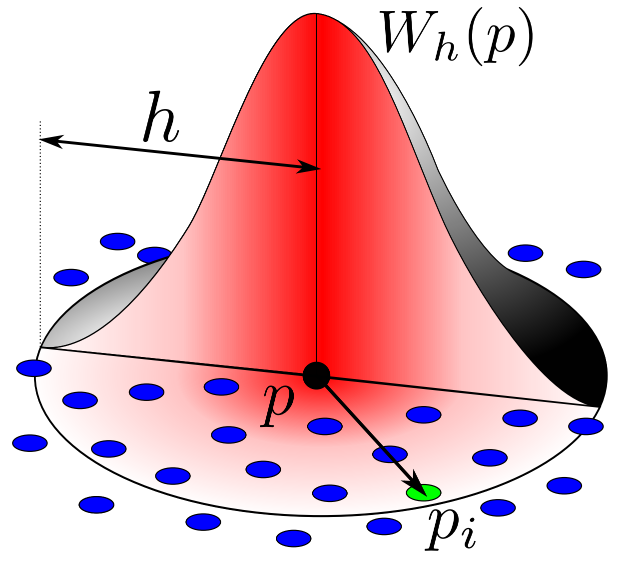
SPH - Smoothed Particle Hydrodynamics

Pure Lagrangian approach.

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

  Use some local weighting kernel W

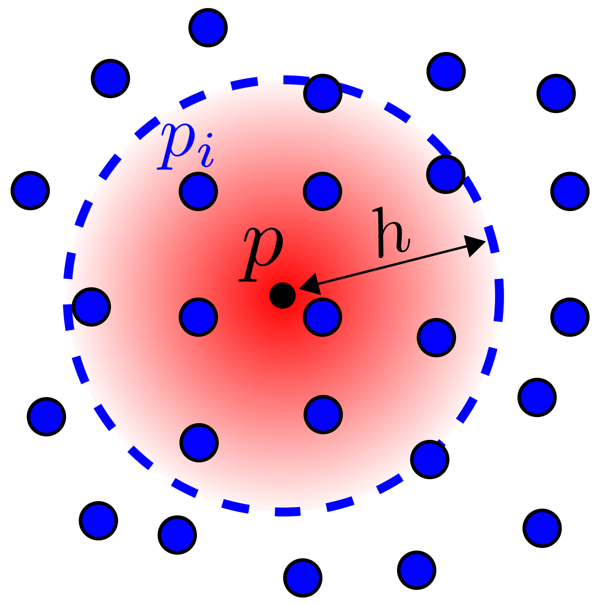
- Express derivatives/Navier-Stokes on the continuous field



Advantages

(+) Particle based - can interact with other models

(+) Scalable

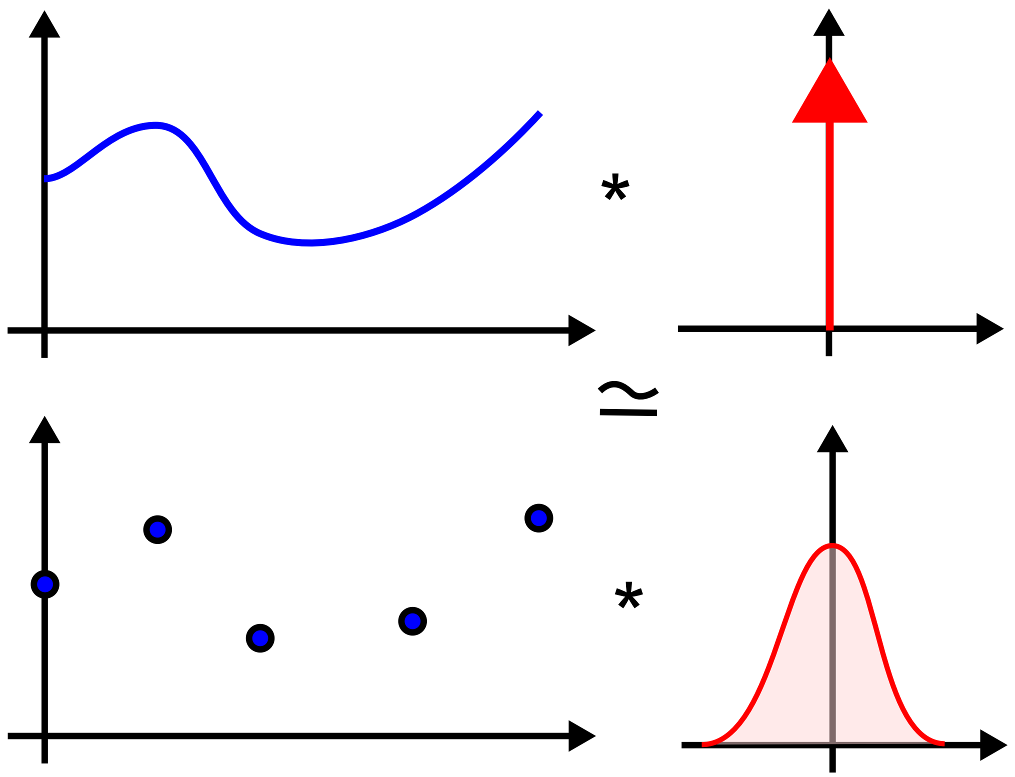


Initial proposed in Astronomy field

[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 $\int_\Omega W_h(p)\, \mathrm{d}p = 1$

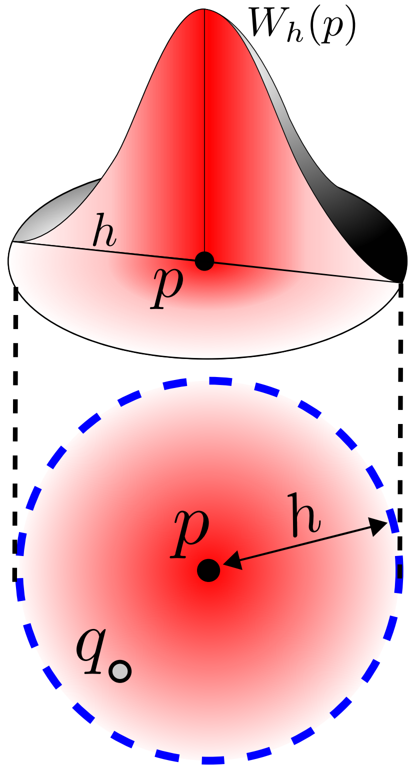
   $A(p) \simeq (A \star W_h)(p) = \int_\Omega A(q)\, W_h(p - q)\, \mathrm{d}q$

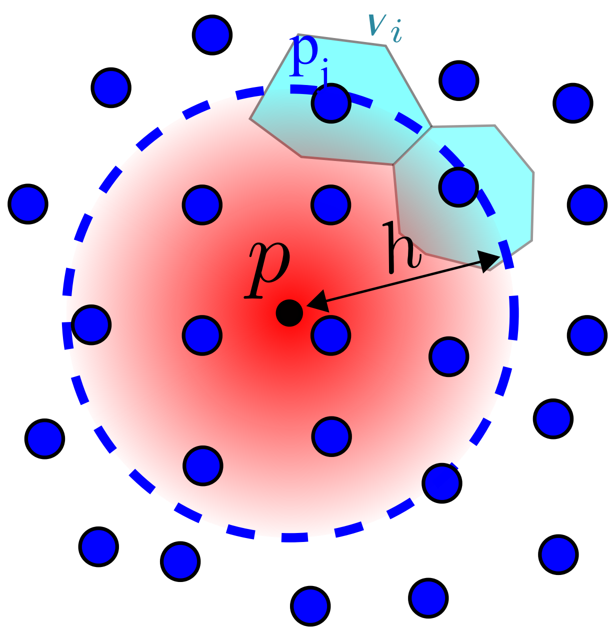
   Low pass filter applied to A

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

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

Objective:

1. Express ρ_i, $\nabla \mathrm{p_r}_i$, $\triangle v_i$ 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: $\rho_i V_i = m_i \quad \Rightarrow \quad A(p_i) = \sum_j A(p_j)\, m_j/\rho_j\, W_h(p_i - p_j)$

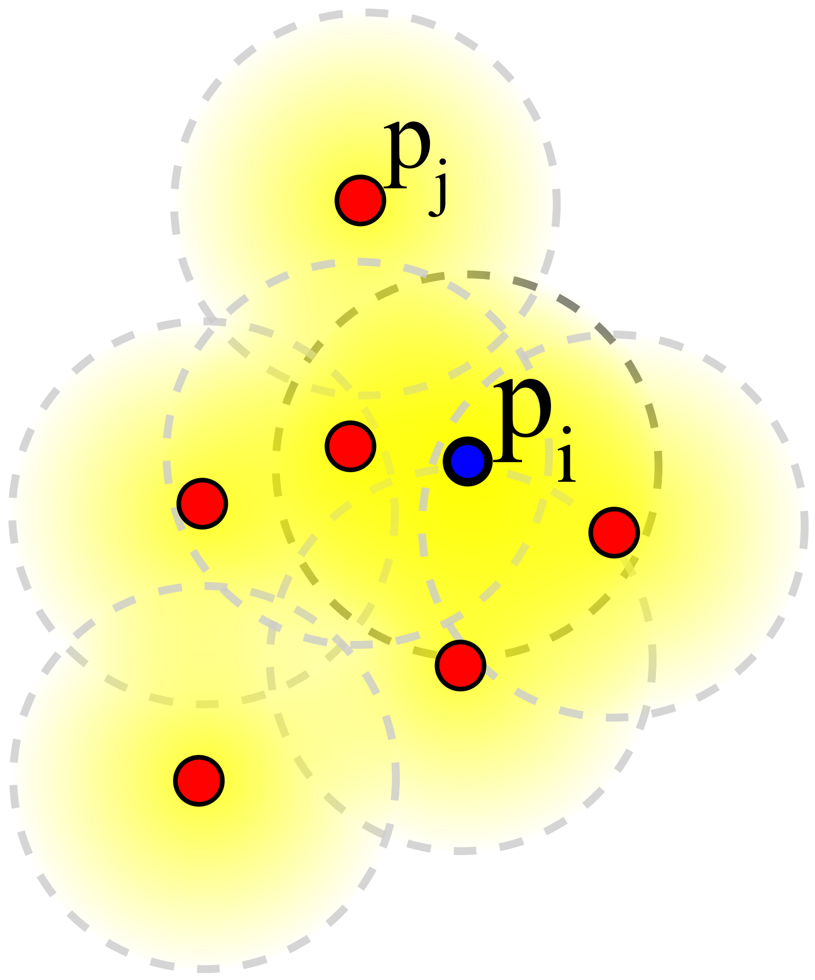
Usually W_h are distance function : $$A(p_i) = \sum_j A(p_j)\, m_j/\rho_j\, W_h(\|p_i - p_j\|)$$

Density

ρ_i: Replace A(p) as ρ

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

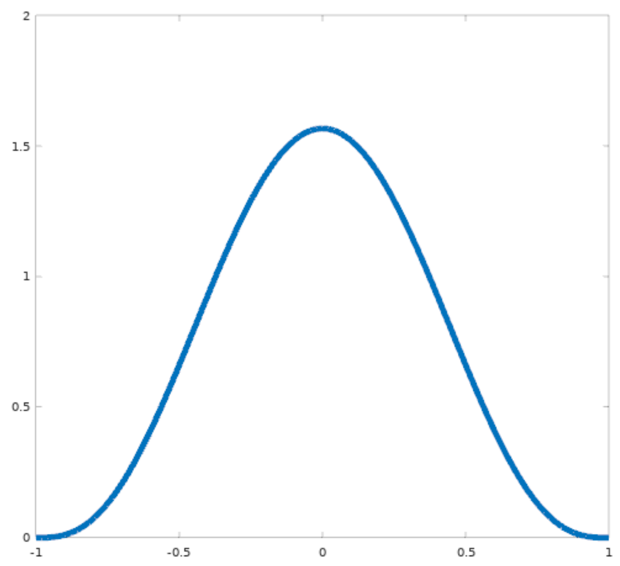
$$\Rightarrow \boxed{\rho_i = \sum_{j=0}^{N-1} m_j\, W_h(\|p_i - p_j\|)}$$



Choice of weight functions

Use a smooth polynomial:

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



Pressure

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

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

$$F_{pressure} = -\frac{m_i}{\rho_i} \sum_{\substack{j=0 \\ j \neq i}}^{N-1} m_j \frac{\mathrm{p_r}_j + \mathrm{p_r}_i}{2\,\rho_j} \nabla W_h(\|p_i - p_j\|)$$

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

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

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

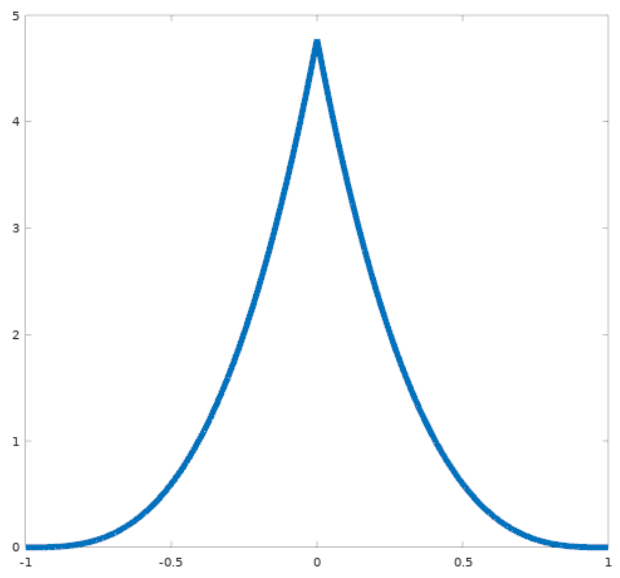
3. Weight function

   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*

   $$F_{viscosity} = m_i \, \nu \sum_{\substack{j=0 \\ j \neq i}}^{N-1} m_j \frac{(\mathrm{v}_j - \mathrm{v}_i)}{\rho_j} \triangle W_h(\|p_i - p_j\|)$$

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

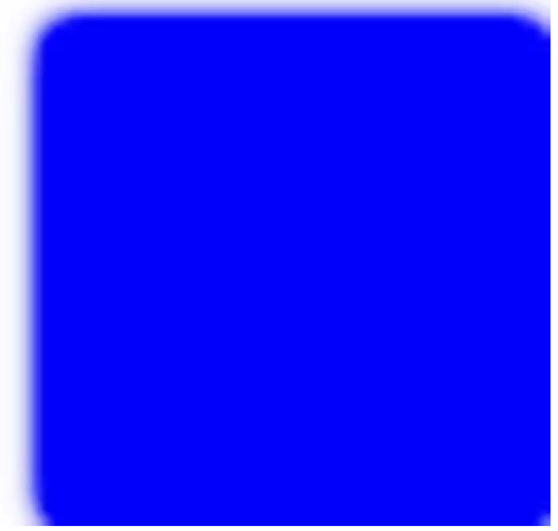
Set initial conditions v_i

Compute values

- Density: $\rho_i = \sum_{j=0}^{N-1} m_j \, W_h^{poly6}(\|p_i - p_j\|)$
- Pressure: $\mathrm{p}_i = s\,(\rho_i - \rho_0)$

Compute forces

- $F_{weight} = m_i \; g$
- $F_{pressure} = -\dfrac{m_i}{\rho_i} \sum_{\substack{j=0 \\ j \neq i}}^{N-1} m_j \, \dfrac{\mathrm{p_r}_j + \mathrm{p_r}_i}{2\,\rho_j} \; \nabla W_h^{spiky}(\|p_i - p_j\|)$
- $F_{viscosity} = m_i \, \nu \sum_{\substack{j=0 \\ j \neq i}}^{N} m_j \, \dfrac{(\mathrm{v}_j - \mathrm{v}_i)}{\rho_j} \; \triangle W_h^{spiky}(\|p_i - p_j\|)$

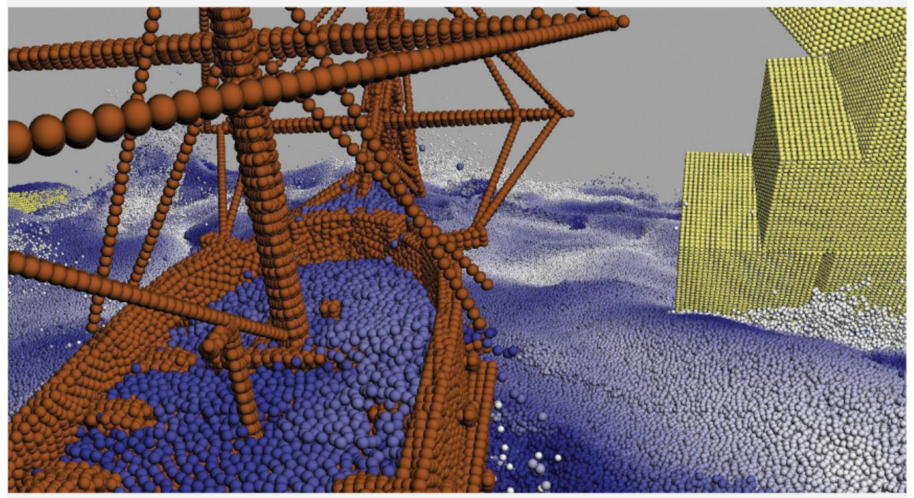


Time integration: $v_i^{k+1} = v_i^k + \Delta t \;\; (F_{weight} + F_{pressure} + F_{viscosity})\;/m_i$

SPH examples

Muller 2003

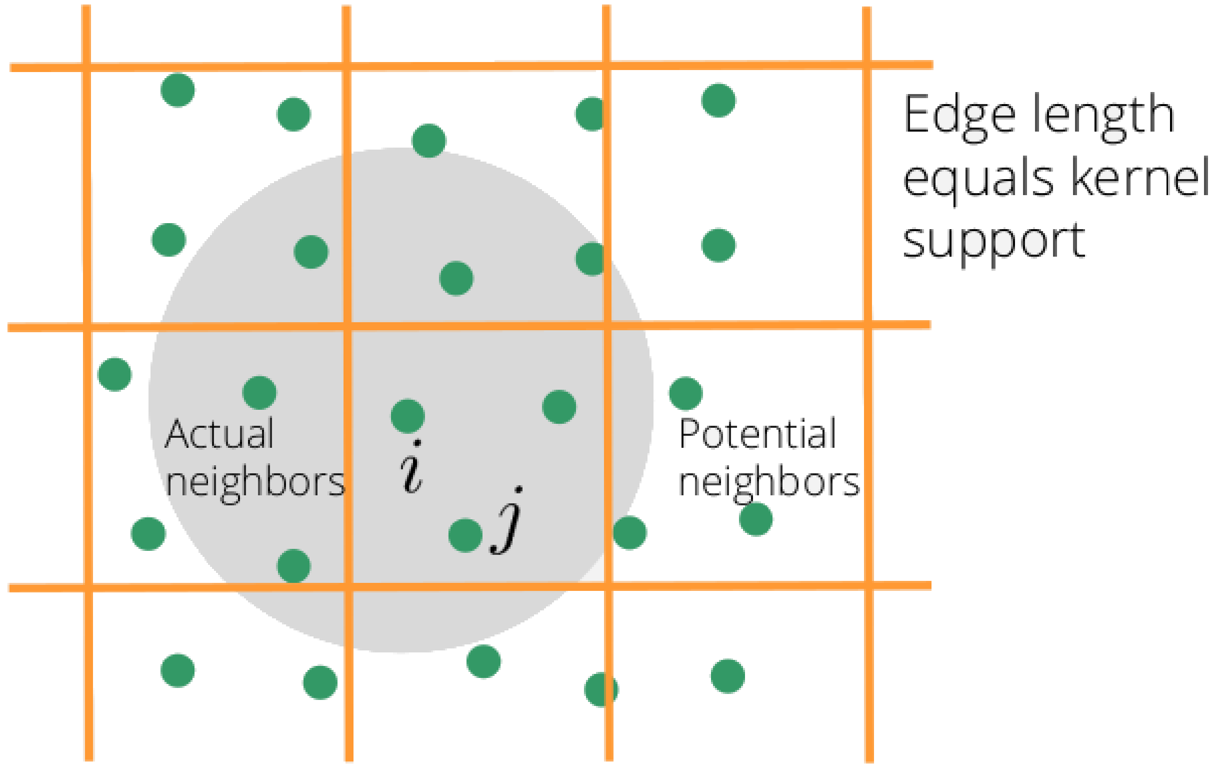
M. Teschner 2012 - 20M particles

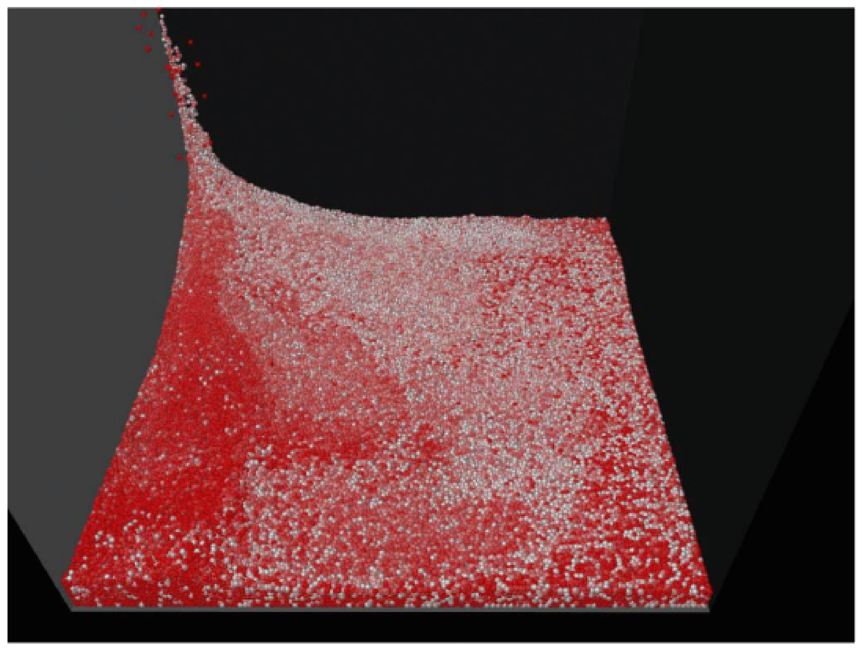


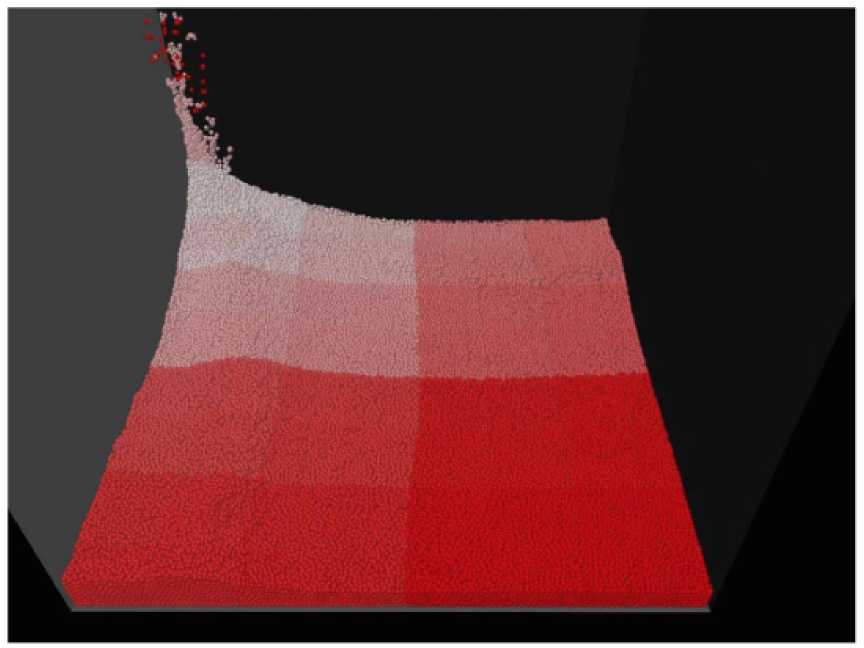
Acceleration structure

SPH based on pair-wise interation ⇒ 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

[redacted]

Bruno Levy

(+) 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]

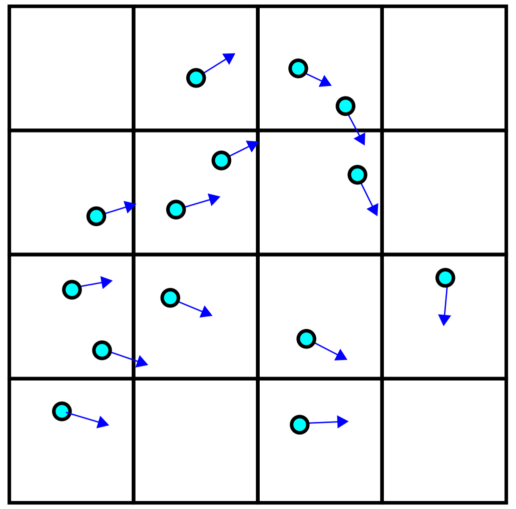
[redacted]

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

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

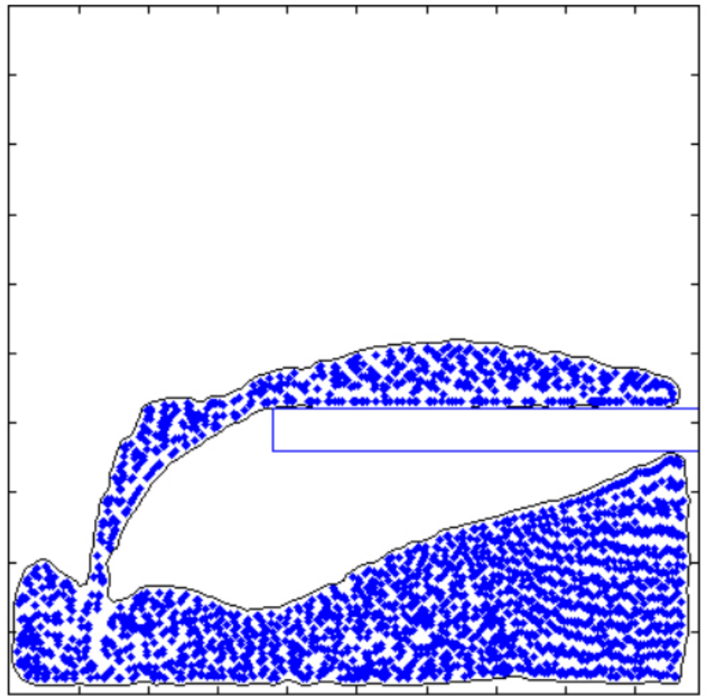
$$u_p^{k+1} = interp(u_g^{k+1}, p^{k+1})$$

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

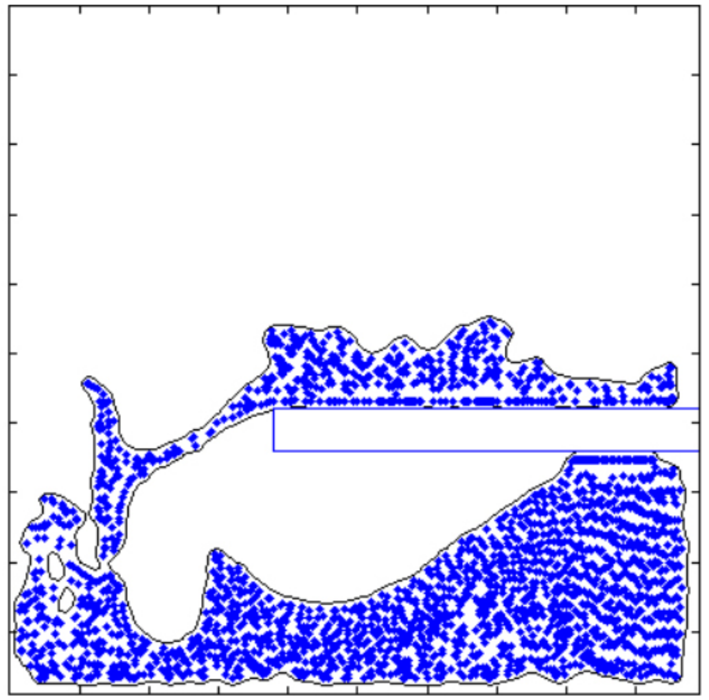
$$u_p^{k+1} = v_p^k + (interp(u_g^{k+1}, p^{k+1}) - interp(u_g^k, p^k))$$

- **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

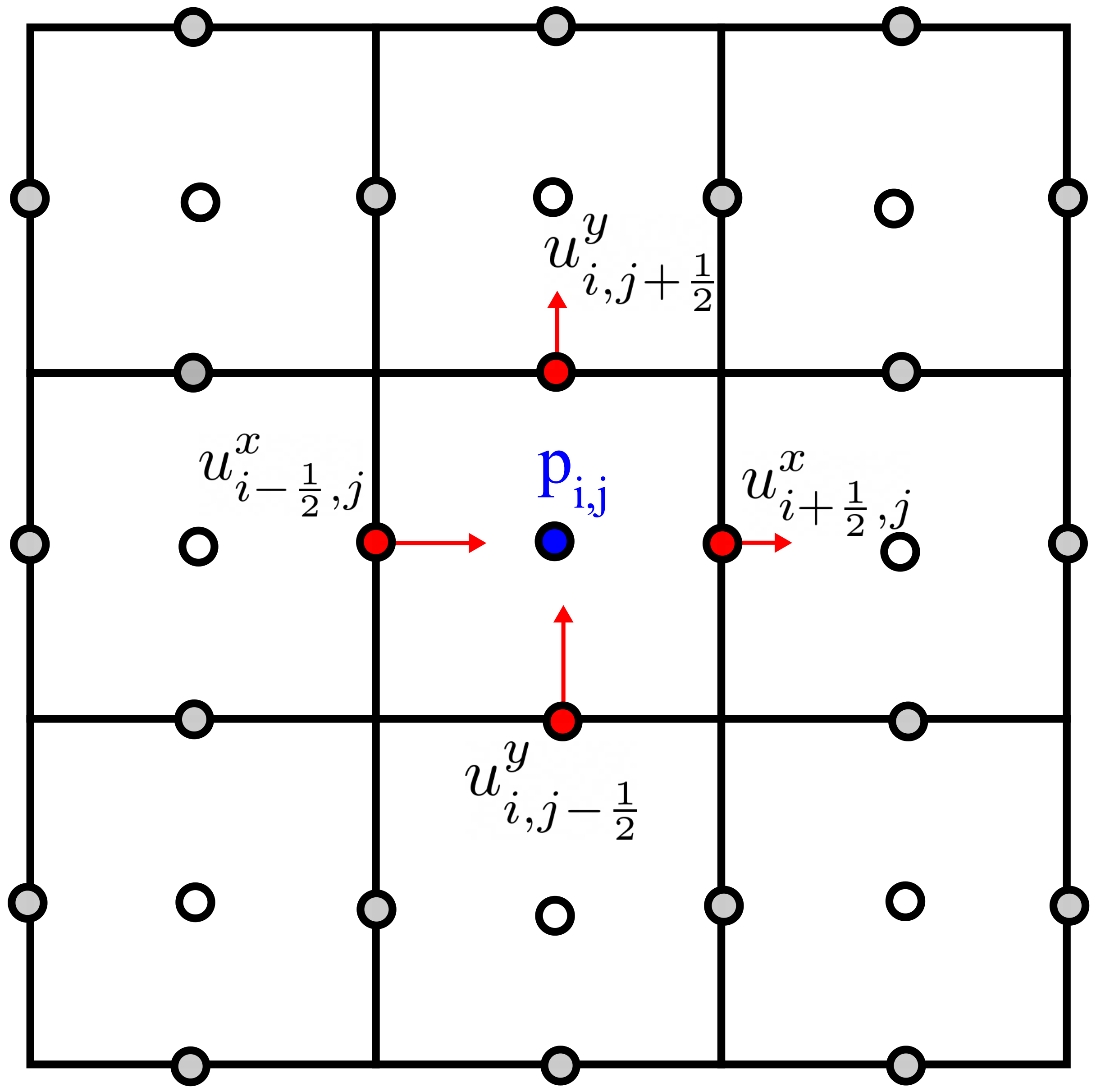
MAC = Marker And Cell

Staggered grid b/w scalar and velocity

Widely used grid storage to handle velocity and scalar values.

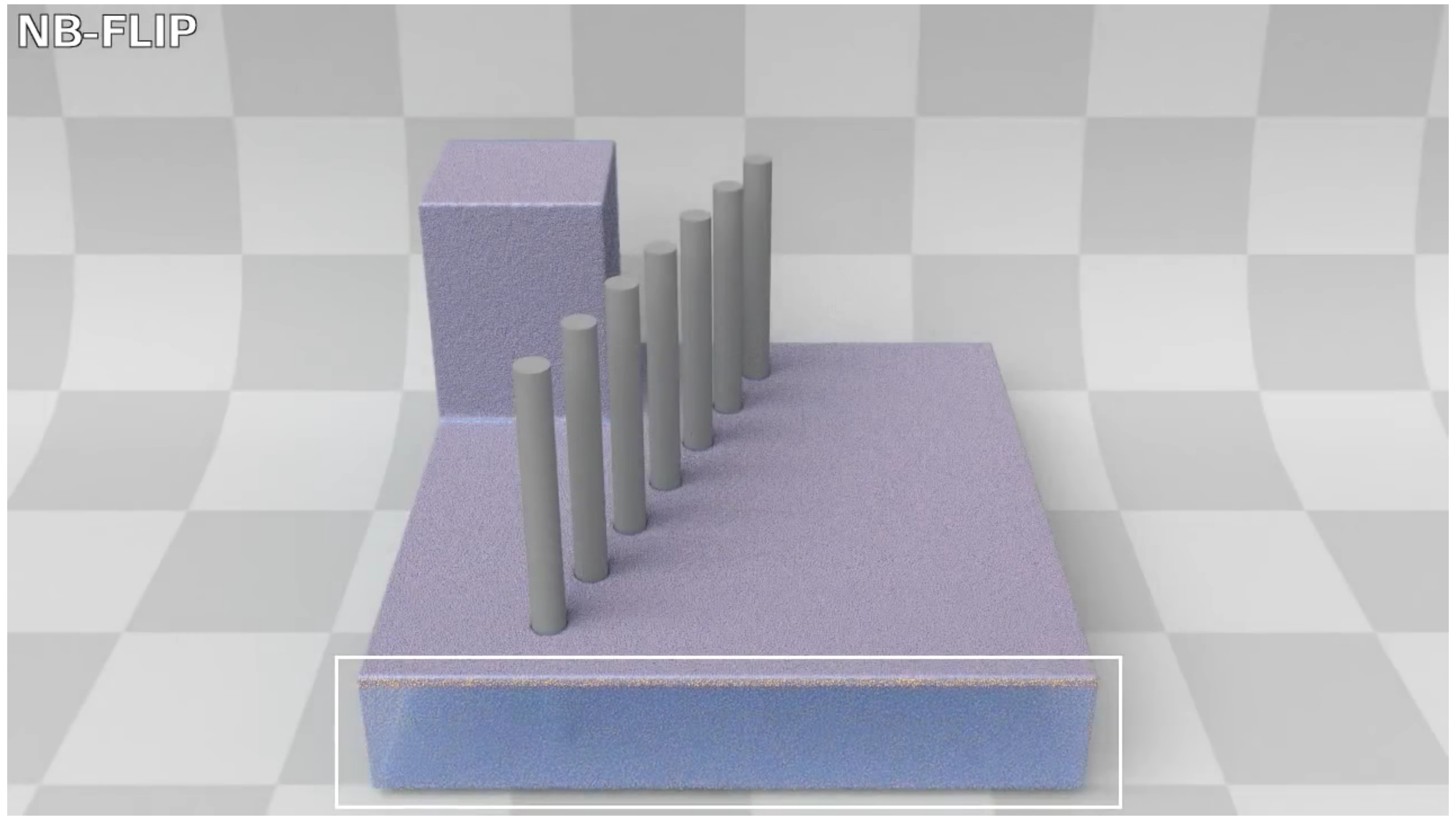
- Store scalar (pressure, density), in the center of the cell
- Store velocity components (u^x, u^y) on the cell edges

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]