Animation & Simulation

He Wang (王鹤)

Fluids: Liquids and Gases

- Computational Fluid Dynamics (CFD, no more cheating)
  - Particle-based approach (Lagrangian)
    - Use particles to approximate mass of a fluid
    - Metaballs, simplistic
    - Smoothed particle hydrodynamics (SPH)
      - Trace particles (positions, velocities)
      - Each particle has mass, density and influence an nearby area

Fluids: Liquids and Gases

- Computational Fluid Dynamics (CFD, no more cheating)
  - Particle-based approach (Lagrangian)
    - Particle-based Fluid Simulation for Interactive Applications (SCA 2003)

Fluids: Liquids and Gases

- Computational Fluid Dynamics (CFD, no more cheating)
  - Particle-based approach (Lagrangian) Matthias Müller, David Charypar and Markus Gross
    - Particle-based Fluid Simulation for Interactive Applications (SCA 2003)

scalar $A_S(\mathbf{r}) = \sum_j m_j \frac{A_j}{\rho_j} W(\mathbf{r} - \mathbf{r}_j, h),$

where j iterates over all particles, m_j is the mass of particle j, $\mathbf{r}_j$ its position, ρ_j the density and A_j the field quantity at $\mathbf{r}_j$.

$W(\mathbf{r}, h)$ is called the smoothing kernel $\int W(\mathbf{r}) d\mathbf{r} = 1$

If we are interested in density, then

$$\rho_S(\mathbf{r}) = \sum_j m_j \frac{\rho_j}{\rho_j} W(\mathbf{r} - \mathbf{r}_j, h) = \sum_j m_j W(\mathbf{r} - \mathbf{r}_j, h)$$

If we want derivatives, then

$$\nabla A_S(\mathbf{r}) = \sum_j m_j \frac{A_j}{\rho_j} \nabla W(\mathbf{r} - \mathbf{r}_j, h)$$

$$\nabla^2 A_S(\mathbf{r}) = \sum_j m_j \frac{A_j}{\rho_j} \nabla^2 W(\mathbf{r} - \mathbf{r}_j, h)$$

Fluids: Liquids and Gases

- Computational Fluid Dynamics (CFD, no more cheating)
  - Particle-based approach (Lagrangian)
    - Particle-based Fluid Simulation for Interactive Applications (SCA 2003)

scalar $A_S(\mathbf{r}) = \sum_j m_j \frac{A_j}{\rho_j} W(\mathbf{r}-\mathbf{r}_j, h),$

Not needed anymore, due to the derivative of the velocity field is essentially the derivative of the particle velocity, Dv/Dt

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0$$

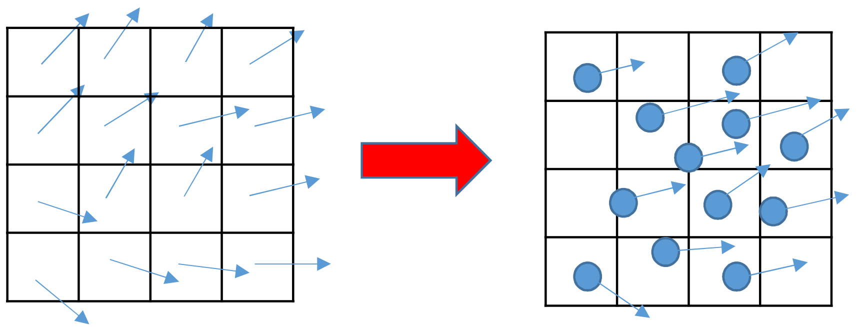
Mass conservation

Not needed anymore!

$$\rho \left(\boxed{\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v}} \right) = \boxed{-\nabla p + \rho \mathbf{g} + \mu \nabla^2 \mathbf{v}}$$

Three forces, pressure, external force and viscosity

Momentum conservation



Fluids: Liquids and Gases

- Computational Fluid Dynamics (CFD, no more cheating)
  - Particle-based approach (Lagrangian)
    - Particle-based Fluid Simulation for Interactive Applications (SCA 2003)

scalar $A_S(\mathbf{r}) = \sum_j m_j \frac{A_j}{\rho_j} W(\mathbf{r}-\mathbf{r}_j, h),$

Not needed anymore, due to the derivative of the velocity field is essentially the derivative of the particle velocity, Dv/Dt

$$\rho\left(\boxed{\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v}\cdot\nabla\mathbf{v}}\right) = \boxed{-\nabla p + \rho\mathbf{g} + \mu\nabla^2\mathbf{v}}$$

Three forces, pressure, external force and viscosity

- Pressure

$$\mathbf{f}_i^{\text{pressure}} = -\nabla p(\mathbf{r}_i) = -\sum_j m_j \frac{p_j}{\rho_j}\nabla W(\mathbf{r}_i-\mathbf{r}_j, h)$$

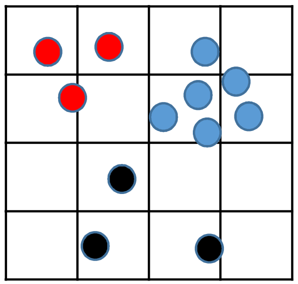
So $$\mathbf{f}_i^{\text{pressure}} = -\sum_j m_j \frac{p_i + p_j}{2\rho_j}\nabla W(\mathbf{r}_i-\mathbf{r}_j, h)$$

Not symmetric

Normal Density

Low Density

High Density



Where pressure can be computed as $p = k(\rho - \rho_0)$ k is a gas constant depending on the temperature

Rest density, manually defined

Fluids: Liquids and Gases

- Computational Fluid Dynamics (CFD, no more cheating)
  - Particle-based approach (Lagrangian)
    - Particle-based Fluid Simulation for Interactive Applications (SCA 2003)

scalar $A_S(\mathbf{r}) = \sum_j m_j \frac{A_j}{\rho_j} W(\mathbf{r} - \mathbf{r}_j, h),$

Not needed anymore, due to the derivative of the velocity field is essentially the derivative of the particle velocity. Dv/Dt

$$\rho \left(\boxed{\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v}} \right) = \boxed{-\nabla p + \rho \mathbf{g} + \mu \nabla^2 \mathbf{v}}$$

Three forces, pressure, external force and viscosity

    - Viscosity

$$\mathbf{f}_i^{\text{viscosity}} = \mu \nabla^2 \mathbf{v}(\mathbf{r}_a) = \mu \sum_j m_j \frac{\mathbf{v}_j}{\rho_j} \nabla^2 W(\mathbf{r}_i - \mathbf{r}_j, h)$$

Since viscosity only depends on the velocity differences

$$\mathbf{f}_i^{\text{viscosity}} = \mu \sum_j m_j \frac{\mathbf{v}_j - \mathbf{v}_i}{\rho_j} \nabla^2 W(\mathbf{r}_i - \mathbf{r}_j, h)$$

Fluids: Liquids and Gases

- Computational Fluid Dynamics (CFD, no more cheating)
    - Particle-based approach (Lagrangian)
        - Particle-based Fluid Simulation for Interactive Applications (SCA 2003)

scalar $A_S(\mathbf{r}) = \sum_j m_j \frac{A_j}{\rho_j} W(\mathbf{r} - \mathbf{r}_j, h),$

Not needed anymore, due to the derivative of the velocity field is essentially the derivative of the particle velocity, Dv/Dt

$$\rho \left(\boxed{\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v}} \right) = \boxed{-\nabla p + \rho \mathbf{g} + \mu \nabla^2 \mathbf{v}}$$

Three forces, pressure, external force and viscosity

        - Surface tension, (not in the equation, why?)
            - The equation is for internal where attributes are equal in all directions

Fluids: Liquids and Gases

- Computational Fluid Dynamics (CFD, no more cheating)
  - Particle-based approach (Lagrangian)
    - Particle-based Fluid Simulation for Interactive Applications (SCA 2003)

scalar $A_S(\mathbf{r}) = \sum_j m_j \frac{A_j}{\rho_j} W(\mathbf{r} - \mathbf{r}_j, h),$

Not needed anymore, due to the derivative of the velocity field is essentially the derivative of the particle velocity. Dv/Dt

$$\rho \left(\boxed{\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v}} \right) = \boxed{-\nabla p + \rho \mathbf{g} + \mu \nabla^2 \mathbf{v}}$$

Three forces, pressure, external force and viscosity

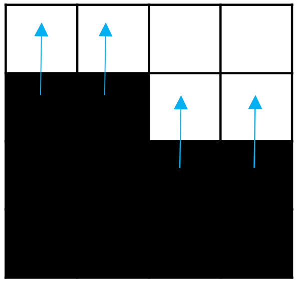
- Surface tension
  - A field with 1 wherever there is fluid and 0 otherwise, colour field c

$c_S(\mathbf{r}) = \sum_j m_j \frac{1}{\rho_j} W(\mathbf{r} - \mathbf{r}_j, h)$ then its gradient $\mathbf{n} = \nabla c_s$ yields the surface normal pointing into the grid
its divergence gives curvature

$$\kappa = \frac{-\nabla^2 c_s}{|\mathbf{n}|}$$

$$\mathbf{f}^{\text{surface}} = \sigma \kappa \mathbf{n} = -\sigma \nabla^2 c_S \frac{\mathbf{n}}{|\mathbf{n}|}$$

The minus is necessary to get
positive curvature for convex fluid volumes



Black cells: 1, White cells: 0

Fluids: Liquids and Gases

- Computational Fluid Dynamics (CFD, no more cheating)
  - Particle-based approach (Lagrangian)
    - Particle-based Fluid Simulation for Interactive Applications (SCA 2003)

scalar $A_S(\mathbf{r}) = \sum_j m_j \frac{A_j}{\rho_j} W(\mathbf{r} - \mathbf{r}_j, h),$

Not needed anymore, due to the derivative of the velocity field is essentially the derivative of the particle velocity. Dv/Dt

$$\rho \left(\boxed{\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v}} \right) = \boxed{-\nabla p + \rho \mathbf{g} + \mu \nabla^2 \mathbf{v}}$$

Three forces, pressure, external force and viscosity

    - External forces
      - Constant, e.g. gravity
      - Changing, user interaction

Fluids: Liquids and Gases

- Computational Fluid Dynamics (CFD, no more cheating)
  - Particle-based approach (Lagrangian)
    - Particle-based Fluid Simulation for Interactive Applications (SCA 2003)

scalar $A_S(\mathbf{r}) = \sum_j m_j \frac{A_j}{\rho_j} W(\mathbf{r} - \mathbf{r}_j, h),$

Not needed anymore, due to the derivative of the velocity field is essentially the derivative of the particle velocity. Dv/Dt

$$\rho\left(\boxed{\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v}\cdot\nabla\mathbf{v}}\right) = \boxed{-\nabla p + \rho\mathbf{g} + \mu\nabla^2\mathbf{v}}$$

Three forces, pressure, external force and viscosity

- Kernel functions W(r, h)-> h is radius, manually set.
  - Stability, accuracy and speed highly dependent on the choice of W
  - Smooth functions are good
  - Vanishing derivatives at the boundary tends to more stable

Fluids: Liquids and Gases

- Computational Fluid Dynamics (CFD, no more cheating)
  - Particle-based approach (Lagrangian)
    - Particle-based Fluid Simulation for Interactive Applications (SCA 2003)

scalar $A_S(\mathbf{r}) = \sum_j m_j \frac{A_j}{\rho_j} W(\mathbf{r} - \mathbf{r}_j, h),$

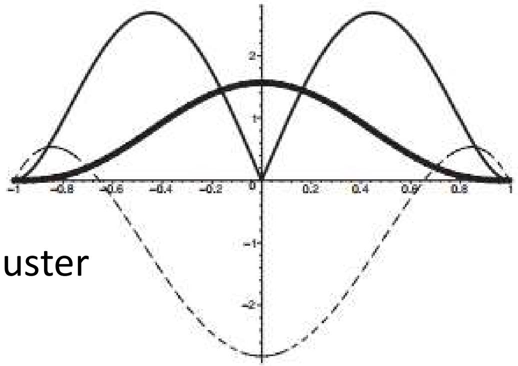
Not needed anymore, due to the derivative of the velocity field is essentially the derivative of the particle velocity, Dv/Dt

$$\rho\left(\boxed{\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v}}\right) = \boxed{-\nabla p + \rho \mathbf{g} + \mu \nabla^2 \mathbf{v}}$$

Three forces, pressure, external force and viscosity

- Kernel functions W(r, h), general purpose

$$W_{\text{poly6}}(\mathbf{r}, h) = \frac{315}{64\pi h^9} \begin{cases} (h^2 - r^2)^3 & 0 \le r \le h \\ 0 & \text{otherwise} \end{cases}$$



Not for computing pressure force as particles can cluster
Why? (hint: gradient near zero)

$$\mathbf{f}_i^{\text{pressure}} = -\sum_j m_j \frac{p_i + p_j}{2\rho_j} \nabla W(\mathbf{r}_i - \mathbf{r}_j, h)$$

Thick line: function, thin line: gradient, dashed: Laplacian

Fluids: Liquids and Gases

- Computational Fluid Dynamics (CFD, no more cheating)
  - Particle-based approach (Lagrangian)
    - Particle-based Fluid Simulation for Interactive Applications (SCA 2003)

scalar $A_S(\mathbf{r}) = \sum_j m_j \frac{A_j}{\rho_j} W(\mathbf{r} - \mathbf{r}_j, h),$

Not needed anymore, due to the derivative of the velocity field is essentially the derivative of the particle velocity, Dv/Dt

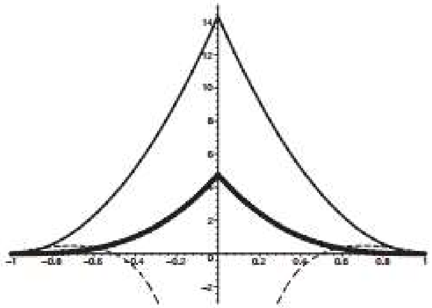
$$\rho \left(\boxed{\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v}} \right) = \boxed{-\nabla p + \rho \mathbf{g} + \mu \nabla^2 \mathbf{v}}$$

Three forces, pressure, external force and viscosity

    - Kernel functions W(r, h), for pressure

For pressure force

$$W_{\text{spiky}}(\mathbf{r}, h) = \frac{15}{\pi h^6} \begin{cases} (h-r)^3 & 0 \le r \le h \\ 0 & \text{otherwise,} \end{cases}$$



First, Laplacian vanishes at boundaries

However, do not work for viscosity, why? (hint: Laplacian near zero)

$$\mathbf{f}_i^{\text{viscosity}} = \mu \sum_j m_j \frac{\mathbf{v}_j - \mathbf{v}_i}{\rho_j} \nabla^2 W(\mathbf{r}_i - \mathbf{r}_j, h)$$

Thick line: function, thin line: gradient, dashed: Laplacian

Fluids: Liquids and Gases

- Computational Fluid Dynamics (CFD, no more cheating)
  - Particle-based approach (Lagrangian)
    - Particle-based Fluid Simulation for Interactive Applications (SCA 2003)

scalar $A_S(\mathbf{r}) = \sum_j m_j \frac{A_j}{\rho_j} W(\mathbf{r}-\mathbf{r}_j, h),$

Not needed anymore, due to the derivative of the velocity field is essentially the derivative of the particle velocity, Dv/Dt

$$\rho \left(\boxed{\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v}} \right) = \boxed{-\nabla p + \rho \mathbf{g} + \mu \nabla^2 \mathbf{v}}$$

Three forces, pressure, external force and viscosity

- Kernel functions W(r, h), for viscosity

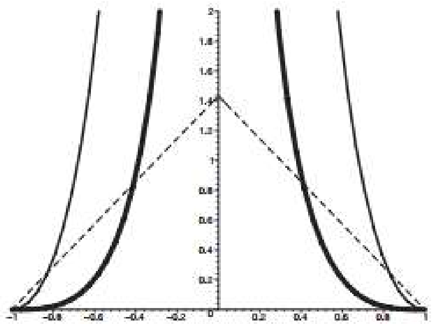
  Viscosity, needs Laplacian to be positive to reduce relative speeds

$$W_{\text{viscosity}}(\mathbf{r}, h) = \frac{15}{2\pi h^3} \begin{cases} -\frac{r^3}{2h^3} + \frac{r^2}{h^2} + \frac{h}{2r} - 1 & 0 \le r \le h \\ 0 & \text{otherwise.} \end{cases}$$

$$\nabla^2 W(\mathbf{r}, h) = \frac{45}{\pi h^6}(h - r)$$

$$W(|\mathbf{r}| = h, h) = 0$$

$$\nabla W(|\mathbf{r}| = h, h) = \mathbf{0}$$



Thick line: function, thin line: gradient, dashed: Laplacian

Fluids: Liquids and Gases

- Computational Fluid Dynamics (CFD, no more cheating)
  - Particle-based approach (Lagrangian)
    - Integration (leapfrog)

$$\begin{aligned} \boldsymbol{v}_i^{n+1/2} &= \boldsymbol{v}_i^n + \boldsymbol{a}_i^n \frac{\Delta t}{2}, \\ \boldsymbol{r}_i^{n+1} &= \boldsymbol{r}_i^n + \boldsymbol{v}_i^{i+1/2} \Delta t, \\ \boldsymbol{v}_i^{n+1} &= \boldsymbol{v}_i^{n+1/2} + \boldsymbol{a}_i^{i+1} \frac{\Delta t}{2}. \end{aligned}$$

    - Implementation tricks
      - Use a grid with cell of size h, to limit search within one cell and its neighbours.
      - Use the colour field and its gradient to identify surface particles, with threshold l

        $|\mathbf{n}(\mathbf{r}_i)| > l.$ $-\mathbf{n}(\mathbf{r}_i)$ Is the direction of the surface normal at particle i
      - Visualising the surface (you don't have to do this in the coursework)
        - Point splatting (Zwicker et al., In Proceedings of the 28th annual conference on Computer graphics and interactive techniques)
        - Marching cubes

Fluids: Liquids and Gases

- Computational Fluid Dynamics (CFD, no more cheating)
  - Particle-based approach (Lagrangian)
    - Particle-based Fluid Simulation for Interactive Applications (SCA 2003)

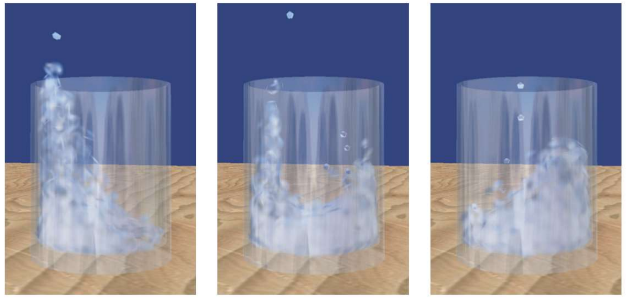


Figure 4: *The user interacts with the fluid causing it to splash.*

Figure 5: *Pouring water into a glass at 5 frames per second.*

Fluids: Liquids and Gases

- Computational Fluid Dynamics (CFD, no more cheating)
  - Particle-based approach (Lagrangian)
    - Fast
    - Less accurate, more difficult to extract surface geometries (e.g Marching Cubes)