CS**5643**

14 Position-based Fluids

Steve Marschner
Cornell University
Spring 2025

Ideas feeding into PBF

Smoothed Particle Hydrodynamics: Lagrangian fluid simulation

- developed originally in astrophysics in the 1970s-80s.

Incompressibility as a constraint

- standard in Eulerian simulation of incompressible fluids
- predictive-corrective SPH did a similar constraint solve for SPH

Position-based dynamics as the constraint solver

- Typical applications use PBD to solve permanent constraints between specific particles
- PBF adapts the same solver to the incompressibility constraint with dynamic neighborhoods

Vorticity confinement as a way to counteract damping

- Introduced earlier to graphics

Sources

Things to read for more details

- Monaghan, J. J. (1992). SMOOTHED PARTICLE HYDRODYNAMICS. Annu. Rev. Astron. Astrophys, 543, 74.
- Müller, M., Charypar, D., & Gross, M. (2003, July). Particle-based fluid simulation for interactive applications. In Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation (pp. 154-159).
- Solenthaler, B., & Pajarola, R. (2009). Predictive-corrective incompressible SPH. In ACM SIGGRApH 2009 papers (pp. 1-6).
- Macklin, M., & Müller, M. (2013). Position based fluids. ACM Transactions on Graphics (TOG), 32(4), 1-12.

SPH review

Fluid dynamics is about the evolution of fields like velocity, pressure, etc.

The idea of SPH is to parameterize these fields with point samples

- quantities of interest stored at particles (at least mass and velocity)
- continuous fields defined by simply blurring the particles with a convolution filter W
- this means the value of any quantity A at any point $\mathbf{r}$ is a weighted sum of nearby particles

$$A_S(\mathbf{x}) = \sum_i m_i \frac{A_i}{\rho_i} W(\mathbf{x} - \mathbf{x}_i)$$

- special case for when the quantity of interest is the density

$$\rho(\mathbf{x}) = \sum_i m_i W(\mathbf{x} - \mathbf{x}_i)$$

SPH review

Continuous fluid equation says

- $\rho \frac{D\mathbf{u}(\mathbf{x})}{Dt} = -\nabla p(\mathbf{x}) + \rho \mathbf{g} + \mu \nabla^2 \mathbf{v}$

But the derivative on the left is exactly the particle velocity

- so this is a recipe for the particle acceleration
- we just need to be able to compute the terms on the right, at the position of each particle

Derivatives of SPH interpolant

- differentiating the interpolant only requires differentiating the smoothing kernel

$$A_S(\mathbf{x}) = \sum_i m_i \frac{A_i}{\rho_i} W(\mathbf{x} - \mathbf{x}_i) \qquad \nabla A_S(\mathbf{x}) = \sum_i m_i \frac{A_i}{\rho_i} \nabla W(\mathbf{x} - \mathbf{x}_i)$$

Computing the pressure force

Pressure exerts a force on each particle

- the pressure is a function of the nearby particles
- given pressure p_i for each particle the interpolant is
  - $p(\mathbf{x}) = \sum_i \frac{m_i}{\rho_i} p_i W(\mathbf{x} - \mathbf{x}_i)$
- and its gradient is
  - $\nabla p(\mathbf{x}) = \sum_i \frac{m_i}{\rho_i} p_i \nabla W(\mathbf{x} - \mathbf{x}_i)$
- resulting in the force due to pressure on particle i
  - $\mathbf{f}_i^{\mathrm{p}} = -\nabla p(\mathbf{x}_i) = -\sum_j \frac{m_j}{\rho_j} p_j \nabla W(\mathbf{x}_i - \mathbf{x}_j)$

Computing the pressure force

Symmetrized force is typically used

- $$\mathbf{f}_i^{\mathrm{p}} = -\nabla p(\mathbf{x}_i) = -\sum_j \frac{m_j}{\rho_j}\frac{p_i + p_j}{2}\nabla W(\mathbf{x}_i - \mathbf{x}_j)$$

How to define pressures for the particles?

- original SPH is for compressible fluids and defines pressure as a function of density
- but this is numerically stiff for nearly incompressible fluids
- same problem exists in Eulerian fluids, solved by using pressure solve to enforce incompressibility
- PBF and the earlier PCSPH apply this idea in the setting of SPH

Position based velocity projection

PBD pattern starts with a set of constraints

In this case the constraints are asking for constant density

- this is a constraint on a continuous field
- approximated by individual constraints at each particle position
- density at each particle depends on neighbors

$$C_i(\mathbf{p}_1, ..., \mathbf{p}_n) = \frac{\rho_i}{\rho_0} - 1,$$

$$\rho_i = \sum_j m_j W(\mathbf{p}_i - \mathbf{p}_j, h).$$

Position based velocity projection

Rather than solve for pressures or velocities, we'll move particles directly

- following PBD strategy

$$\Delta \mathbf{p} \approx \nabla C(\mathbf{p})\lambda$$

$$\begin{aligned} C(\mathbf{p}+\Delta\mathbf{p}) &\approx C(\mathbf{p}) + \nabla C^T \Delta\mathbf{p} = 0 \\ &\approx C(\mathbf{p}) + \nabla C^T \nabla C \lambda = 0. \end{aligned}$$

- thus if we can compute C and ∇C we can update particle positions

$$\nabla_{\mathbf{p}_k} C_i = \frac{1}{\rho_0} \sum_j \nabla_{\mathbf{p}_k} W(\mathbf{p}_i - \mathbf{p}_j, h)$$

$$\lambda_i = -\frac{C_i(\mathbf{p}_1, \ldots, \mathbf{p}_n)}{\sum_k \left|\nabla_{\mathbf{p}_k} C_i\right|^2}$$

$$\nabla_{\mathbf{p}_k} C_i = \frac{1}{\rho_0} \begin{cases} \sum_j \nabla_{\mathbf{p}_k} W(\mathbf{p}_i - \mathbf{p}_j, h) & \text{if } k = i \\ -\nabla_{\mathbf{p}_k} W(\mathbf{p}_i - \mathbf{p}_j, h) & \text{if } k = j \end{cases}$$

$$\Delta \mathbf{p}_i = \frac{1}{\rho_0} \sum_j (\lambda_i + \lambda_j) \nabla W(\mathbf{p}_i - \mathbf{p}_j, h).$$

Results