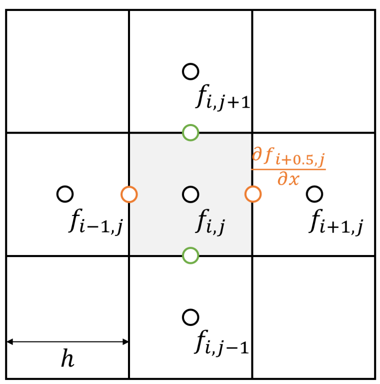
Grid-Based Fluid

Xuan Wang

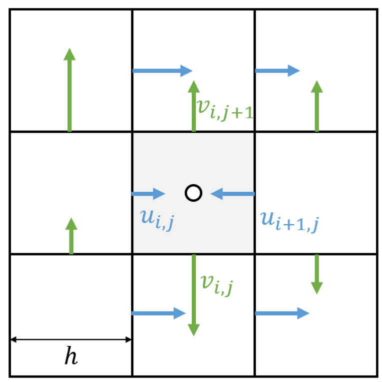
March 2023

1 Grid Representation

In an Eulerian grid representation, we store physical quantities over a grid. The grid is friendly with central difference:



(a) scalar quantity



(b) velocity

Figure 1: Staggered Grid

$$\nabla^2 f_{i,j} = \frac{\partial^2 f_{i,j}}{\partial x^2} + \frac{\partial^2 f_{i,j}}{\partial y^2} \approx \frac{f_{i-1,j} + f_{i+1,j} + f_{i,j-1} + f_{i,j+1} - 4f_{i,j}}{h^2} \quad (1)$$

We can also easily introduce boundary conditions:

$$\text{Dirichlet} \quad f_{i-1,j} = C \quad (2)$$

$$\text{Neumann} \quad f_{i-1,j} = f_{i,j} \quad (3)$$

Note that in fluid simulation, we define velocities on faces (see Figure 1). Intuitively, they represent the flow speed between two cells. For incompressible fluid, this can be formally written as a divergence-free velocity field:

$$u_{i+1,j} + v_{i,j+1} - u_{i,j} - v_{i,j} = 0 \Leftrightarrow \nabla \cdot \mathbf{u}_{ij} \approx \frac{u_{i+1,j} - u_{i,j}}{h} + \frac{v_{i,j+1} - v_{i,j}}{h} = 0 \quad (4)$$

2 Navier-Stokes Equation

Newton's second law tells

$$m\frac{d\mathbf{v}}{dt} = \mathbf{f} \tag{5}$$

For the target fluid blob, we integrate over the entire volume to get m:

$$m = \int_\Omega \rho d\Omega \tag{6}$$

For the right-hand side, we first focus on a small surface patch of the fluid blob:

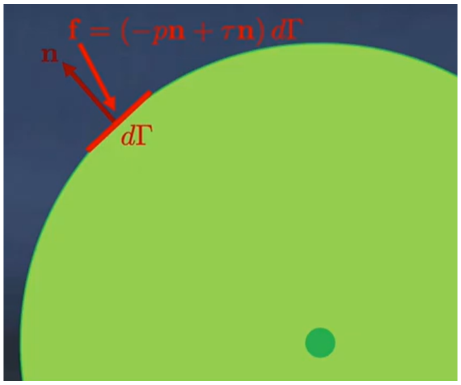


Figure 2: Caption

The force can be divided into 2 parts. The first is pressure, which acts normal to the surface. The negative sign indicates that positive pressure implies the blob is being squeezed. The rest force is stuffed into a remainder term τ.

And now we change the momentum change into

$$\int_\Omega \rho\frac{d\mathbf{v}}{dt}d\Omega = \int_\Gamma (p\mathbf{n} + \tau\mathbf{n})d\Gamma \tag{7}$$

With the divergence theorem, we can change the surface integral into a volume integral:

$$\int_\Omega \rho\frac{d\mathbf{v}}{dt}d\Omega = \int_\Omega (-\nabla p + \nabla\cdot\tau)d\Omega \tag{8}$$

$$\rho\frac{d\mathbf{v}}{dt} = -\nabla p + \nabla\cdot\tau \tag{9}$$

Note that the velocity at any point in space can instantaneously change in time due to some acceleration, but also due to the motion of the fluid blob itself, i.e. $\mathbf{v} = \mathbf{v}(\mathbf{x}(t), t)$. So the left side of Equation 9 is a **material derivative** by applying the chain rule:

$$\frac{d\mathbf{v}}{dt} = \frac{\partial\mathbf{v}}{\partial t} + \nabla\mathbf{v}\cdot\mathbf{v} \tag{10}$$

Finally, we arrive at the famous Navier-Stokes equation, which describes the motion of any fluid:

$$\rho \frac{\partial \mathbf{v}}{\partial t} + \nabla \mathbf{v} \cdot \mathbf{v} = -\nabla p + \nabla \cdot \tau + \rho \mathbf{g} \tag{11}$$

where we add the force of gravity on the right. So points in the fluid react both to gravity and surface tractions.

For incompressible fluids, we add a divergence-free constraint:

$$\nabla \cdot \mathbf{v} = 0 \tag{12}$$

Things like air and water can be modeled as inviscid fluids ($\nabla \cdot \tau = 0$). The resulting N-S equations without viscosity are called the Euler's equations:

$$\begin{aligned} \rho \frac{\partial \mathbf{v}}{\partial t} + \rho \nabla \mathbf{v} \cdot \mathbf{v} &= -\nabla p + \rho \mathbf{g} \\ \nabla \cdot \mathbf{v} &= 0 \end{aligned} \tag{13}$$

3 Fluid Simulation via Splitting

Note that these Euler equations in fact deal with three distinct effects. $\rho \frac{\partial \mathbf{v}}{\partial t} + \rho \nabla \mathbf{v} \cdot \mathbf{v}$ is the change in velocity due to the movement of the fluid, known as advection. $\rho \mathbf{g}$ is the instantaneous acceleration due to external forces like gravity. $-\nabla p$ is the act of pressure attempting to keep our fluid from compressing. This observation allows us to solve Equation 13 by handling these effects individually.

mathematically why?

Input: $\mathbf{v}^t$

Output: $\mathbf{v}^{t+1}$

1 Advection: $\frac{\partial \mathbf{v}}{\partial t} + \nabla \mathbf{v} \cdot \mathbf{v}$

2 External Forces: $\rho \frac{\partial \mathbf{v}}{\partial t} = \rho \mathbf{g}$

3 Pressure Projection: $\rho \frac{\partial \mathbf{v}}{\partial t} = -\nabla p$ subject to $\nabla \cdot \mathbf{v} = 0$

3.1 Advection

Intuitively, advection step evolves the current velocity field. When we talk about advection, we are really talkng about dealing with $\frac{d\mathbf{v}}{dt} = \frac{\partial \mathbf{v}}{\partial t} + \nabla \mathbf{v} \cdot \mathbf{v}$. And since we handle all the forces at later stages, we can assume that acceleration is zero in this step, i.e. $\frac{d\mathbf{v}}{dt} = 0$. So our velocity should only change with the movement of the particles in $\mathbf{x}$, rather than with the changes in time t:

$$\begin{aligned} \mathbf{v}(\mathbf{x}(t + \Delta t), t + \Delta t) &= \mathbf{v}(\mathbf{x}(t), t) \\ \mathbf{x}(t + \Delta t) &\approx \mathbf{x}(t) + \Delta t \mathbf{v}(t) \end{aligned} \tag{14}$$

The solution is to o back-trace a virtual particle from the current position to interpolate a velocity value which will flow into this position in the next time step, i.e. Semi-Lagrangian Method:

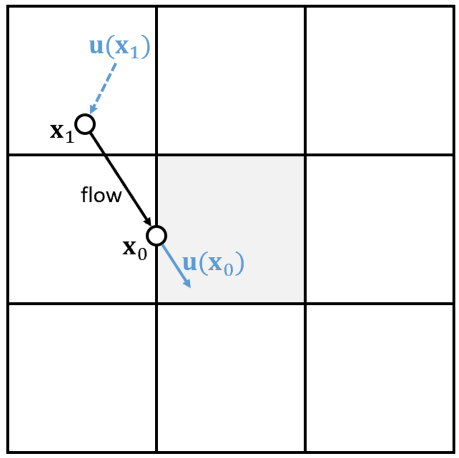


Figure 3: Semi-Lagrangian Method

For better numerical accuracy, we choose the velocity in the location by back-tracing with a half time step.

$$\begin{aligned} \mathbf{v}^{t+\frac{1}{2}}(\mathbf{x}) &= Interpolate(\mathbf{v}^t, \mathbf{v}(\mathbf{x} - \mathbf{v}^t \frac{\Delta t}{2})) \\ \mathbf{v}^*(\mathbf{x}) &= Interpolate(\mathbf{v}^t, \mathbf{v}(\mathbf{x} - \mathbf{v}^{t+\frac{1}{2}} \Delta t)) \end{aligned} \tag{15}$$

3.2 External Forces

External forces change the instantaneous velocity of a fluid blob. So this step is concerned with changing the time variable without moving the particles:

$$\mathbf{v}(\mathbf{x}(t), t + \Delta t) \approx \mathbf{v}(\mathbf{x}(t), t) + \Delta t \mathbf{g} \tag{16}$$

In order to preserve the vorticity details in the flow field, we can introduce an an artificial vorticity confinement force on each grid as:

$$\mathbf{f}_{vc} = \epsilon \frac{1}{\rho} \Delta x (\mathbf{N} \times w) \tag{17}$$

where Δx is the size of the grid, $w = \nabla \times \mathbf{v}$ is the curl of the velocity field and $\mathbf{N} = \frac{\nabla |w|}{\|\nabla |w|\|}$ is the direction of the local rotation. The force aims to enhance the local rotational motion of the fluid. Then the update step is:

$$\mathbf{v}(\mathbf{x}(t), t + \Delta t) \approx \mathbf{v}(\mathbf{x}(t), t) + \Delta t(\mathbf{g} + \mathbf{f}_{vc}) \tag{18}$$

3.3 Pressure Projection

The projection step guarantees that the obtained new velocity field is divergence free. We first try to synthesize a more computationally useful relationship between the partial differential equation and the constraint. The equation can be discretized in the temporal domain as:

$$\begin{aligned}\rho\frac{\mathbf{v}^{t+1}-\mathbf{v}^{t}}{\Delta t} &= -\nabla p \\ \mathbf{v}^{t+1} &= \mathbf{v}^{t} - \frac{\Delta t}{\rho}\nabla p\end{aligned} \tag{19}$$

And we know from the constraint that $\nabla \cdot \mathbf{v}^{t+1} = 0$, which lead to a Poisson equation with the pressure as the unknowns:

$$\nabla \cdot \nabla p = \frac{\rho}{\Delta t}\nabla \cdot \mathbf{v}^{t} \tag{20}$$

In fluid simulation, we solve this Poisson problem with finite differences which discretize the PDE itself. The staggered grid mentioned in the first section makes this process straightforward.

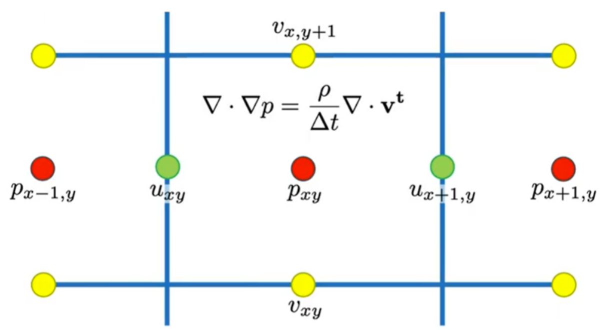


Figure 4: Pressure Projection

$$\begin{aligned}\nabla \cdot \mathbf{v}^{t} &= \begin{pmatrix} -\frac{1}{\Delta x} & \frac{1}{\Delta x} & -\frac{1}{\Delta y} & \frac{1}{\Delta y} \end{pmatrix} \begin{pmatrix} u_{x,y} \\ u_{x+1,y} \\ v_{x,y} \\ v_{x,y+1} \end{pmatrix} \\ \nabla \cdot \mathbf{v}^{t} &= B\mathbf{q}_j\end{aligned} \tag{21}$$

$$\begin{aligned}\nabla p &= \begin{pmatrix} -\frac{1}{\Delta x} & 0 & \frac{1}{\Delta x} & 0 & 0 \\ 0 & \frac{1}{\Delta x} & -\frac{1}{\Delta x} & 0 & 0 \\ 0 & 0 & \frac{1}{\Delta y} & -\frac{1}{\Delta y} & 0 \\ 0 & 0 & -\frac{1}{\Delta y} & 0\frac{1}{\Delta y} & \end{pmatrix} \begin{pmatrix} p_{x-1,y} \\ p_{x+1,y} \\ p_{x,y} \\ p_{x,y-1} \\ p_{x,y+1} \end{pmatrix} \\ \nabla p &= D\mathbf{p}_j\end{aligned} \tag{22}$$

Finally, we can express the Poisson equation in an elegant matrix form:

$$BD\mathbf{p}_j = \frac{\rho}{\Delta t}B\mathbf{q}_j \tag{23}$$

Note that along the boundary, we should constrain the appropriate velocity sample to be zero. We can introduce the standard fixed point projection matrix $P^T P$, which is a diagonal matrix with 0 elements corresponding to the velocity samples we wish to zero out.

$$B(P^T P)D\mathbf{p}_j = \frac{\rho}{\Delta t} B(P^T P)\mathbf{q}_j \tag{24}$$