LAB REPORT: LAB 6

TNM079, MODELING AND ANIMATION

Algot Sandahl
algsa119@student.liu.se

Wednesday 16th June, 2021

Abstract

This lab report describes how fluids can be simulated using the stable fluid method to solve the Euler equations for inviscid flow. The fluid interface is visualized using a level set and the boundary conditions are briefly described. The method produced useful although not entirely realistic results.

1 Background

Simulating fluids is a notoriously hard problem, that requires careful strategies to solve. One such strategy is the stable fluids method [1] used in this lab.

An incompressible fluid described by the velocity field, $\mathbf{V}$, is described by the Navier–Stokes equations

$$\begin{aligned}\frac{\partial \mathbf{V}}{\partial t} = \mathbf{F} + \nu\nabla^2\mathbf{V} - (\mathbf{V}\cdot\nabla)\mathbf{V} - \frac{\nabla p}{\rho},\\ \nabla\cdot\mathbf{V} = 0\end{aligned} \tag{1}$$

where $\mathbf{F}$ is the external force term, p is the pressure field, and ρ is the constant density.

To solve the equations a technique called operator splitting is used. It allows the different terms to be solved individually one after the other in the following manner

$$\mathbf{V}_0 \xrightarrow{(\mathbf{V}\cdot\nabla)\mathbf{V}} \mathbf{V}_1 \xrightarrow{\mathbf{F}} \mathbf{V}_2 \xrightarrow{\nu\nabla^2\mathbf{V}} \mathbf{V}_3 \xrightarrow{\frac{\nabla p}{\rho},\ \nabla\cdot\mathbf{V}} \mathbf{V}_{\Delta t} \tag{2}$$

where $\mathbf{V}_0$ is the initial velocity field, and $\mathbf{V}_{\Delta t}$ is the velocity field one time step, Δt, later.

To reduce the complexity further, the diffusion term, $\nu\nabla^2\mathbf{V}$, is omitted in this lab. The diffusion term represent the viscosity of the fluid, and water has low viscosity. With this simplification, the procedure becomes

$$\mathbf{V}_0 \xrightarrow{(\mathbf{V}\cdot\nabla)\mathbf{V}} \mathbf{V}_1 \xrightarrow{\mathbf{F}} \mathbf{V}_2 \xrightarrow{\frac{\nabla p}{\rho},\ \nabla\cdot\mathbf{V}} \mathbf{V}_{\Delta t} \tag{3}$$

The vector field is solved on a uniform grid with the grid spacing Δx. The treatment of the different terms are addressed one by one in the following paragraphs.

The first term that is solved is the self-advection term; it represents non-linear phenomenons, for example vortices, and can be interpreted as the movement of the fluid by itself. The differential equation to solve is

$$\frac{\partial \mathbf{V}_1}{\partial t} = -(\mathbf{V}_0\cdot\nabla)\mathbf{V}_0 \tag{4}$$

There are different ways to approximate this term, but many of these might be unstable, especially for time steps that are not very small. The method used here is based on the method of characteristics, and is unconditionally stable. It works by tracing particles backwards in time using the information in the velocity field, and assigns the value of the velocity field at that point to the new velocity field:

$$\mathbf{V}_1(\mathbf{x}) = \mathbf{V}_0(\mathbf{T}(\mathbf{x}, -\Delta t)) \tag{5}$$

where $\mathbf{T}$ is the particle trace operator. While this scheme is not strictly correct if the field changes over time, it provides a good estimate. Since the positions returned by the particle trace operator are usually not located on the grid, the values of $\mathbf{V}_0$ have to interpolated, in this case by trilinear interpolation.

The most straight-forward term is the external force term, $\mathbf{F}$. In this lab it is constant and consists only of gravity. It is defined as a vector field, so the equation to solve is

$$\frac{\partial \mathbf{V}_2}{\partial t} = \mathbf{F} \tag{6}$$

which can be solved by Euler integration:

$$\mathbf{V}_2 = \mathbf{V}_1 + \Delta t \mathbf{F}. \tag{7}$$

The last step of the procedure is by far the most complicated one, and will fill the rest of the section.

According to the Helmholtz–Hodge decomposition any vector field can be decomposed into two parts

$$\mathbf{V}_2 = \mathbf{V}_{\text{df}} + \mathbf{V}_{\text{cf}} \tag{8}$$

that fulfill

$$\nabla \cdot \mathbf{V}_{\text{df}} = 0 \tag{9}$$
$$\nabla \times \mathbf{V}_{\text{cf}} = 0 \tag{10}$$

Since the solution, $\mathbf{V}_{\Delta t}$, should be incompressible, it can be seen as the divergence-free component, $\mathbf{V}_{\text{df}}$. Furthermore, gradients never have curl which can be used to rewrite the decomposition (8) as

$$\mathbf{V}_2 = \mathbf{V}_{\Delta t} + \nabla q \quad \Leftrightarrow \quad \mathbf{V}_{\Delta t} = \mathbf{V}_2 - \nabla q \tag{11}$$

where q is a scalar field. To calculate $\mathbf{V}_{\Delta t}$, q has to be known. It can be found by first applying the divergence operator to (11), giving the Poisson equation

$$\nabla \cdot \mathbf{V}_2 = \nabla^2 q \tag{12}$$

since $\mathbf{V}_{\Delta t}$ is divergence-free.

The discrete divergence operator based on central differencing for the point (i, j, k) in the grid is calculated as

$$\begin{aligned}\nabla \cdot \mathbf{V}_{i,j,k} \approx & \frac{u_{i+1,j,k} - u_{i-1,j,k}}{2\Delta x} \\ & + \frac{v_{i,j+1,k} - v_{i,j-1,k}}{2\Delta y} \\ & + \frac{w_{i,j,k+1} - w_{i,j,k-1}}{2\Delta z}\end{aligned} \tag{13}$$

where u, v, and w are the components of the vectors of the vector field. The discretized Laplace operator, $\nabla^2 q_{i,j,k}$ can be written in vector form:

$$\frac{1}{\Delta x^2} \begin{bmatrix} 1 & 1 & 1 & -6 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} q_{i+1,j,k} \\ q_{i-1,j,k} \\ q_{i,j+1,k} \\ q_{i,j,k} \\ q_{i,j-1,k} \\ q_{i,j,k+1} \\ q_{i,j,k-1} \end{bmatrix} \tag{14}$$

Using these discretizations, an equation system

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{15}$$

can be formed from (12) by vectorizing the divergence values for the grid into $\mathbf{b}$. The matrix $\mathbf{A}$ is formed by the discrete Laplace operator. Solving for $\mathbf{x}$ gives the values of q vectorized in the same way as the values in $\mathbf{b}$.

The $\mathbf{A}$ matrix is typically huge, but luckily it is sparse—at most seven elements per row are non-zero—so the memory usage can be kept under control. The conjugate gradient method is used to solve (15) to a predetermined tolerance. Since q is then known, $\mathbf{V}_{\Delta t}$ can be calculated according to (11).

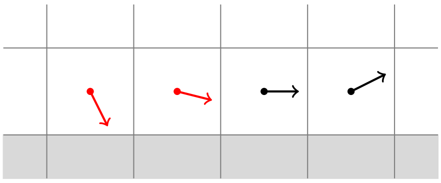
Further complicating matters, the boundaries need to be taken into account. This lab uses two boundary conditions

$$\text{Dirichlet: } \mathbf{V} \cdot \mathbf{n} = 0 \tag{16}$$
$$\text{Neumann: } (\partial \mathbf{V})/(\partial \mathbf{n}) = 0 \tag{17}$$

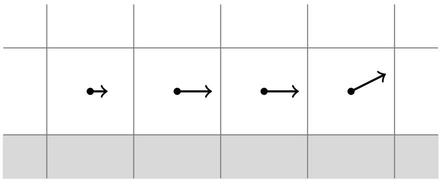
which specify that there should be no flow in or out of the boundary, and that the flow should not change along the boundary normal, $\mathbf{n}$. Information about which voxels are solid, or contain air or fluid is generated from the information in the scene to allow them to be treated differently.

The Dirichlet condition is enforced by projecting vectors that point into the boundary onto the tangent plane as demonstrated by Figure 1; this is done before and after the projection ($\mathbf{V}_2 \rightarrow \mathbf{V}_{\Delta t}$).

The Neumann condition is enforced, for each voxel, by forcing the difference between



(a) Before



(b) After

Figure 1: An example of the Dirichlet boundary condition. Gray voxels are solid and white voxels are fluid. Red velocity vectors defy the boundary conditions.

solid neighboring voxels, and the current voxel to zero. This is done by modifying the Laplace matrix, **A**, by setting the coefficient corresponding to the solid neighbors to 0 instead of $1/\Delta x^2$ and then adding the number of solid neghbors divided by Δx^2 to the coefficient for the current voxel.

To visualize the fluid, a level set is used. The velocity field is extended to cover the narrow band of the level set so it can be used to change the level set.

2 Results

The results of the fluid simulation is shown in Figure 3 without the self-advection term, and in Figure 4 with the self-advection term (the full Euler equations).

It is clear that the self-advection term is very important to the behavior of a water-like liquid; it can be seen how the liquid moves with itself in a way that it does not without the self-advection term.

Since only the Euler equations were solved as opposed to the Navier–Stokes equations, the resulting fluid should be inviscid. Nonetheless the result looks somewhat viscous. This can probably be traced back to the

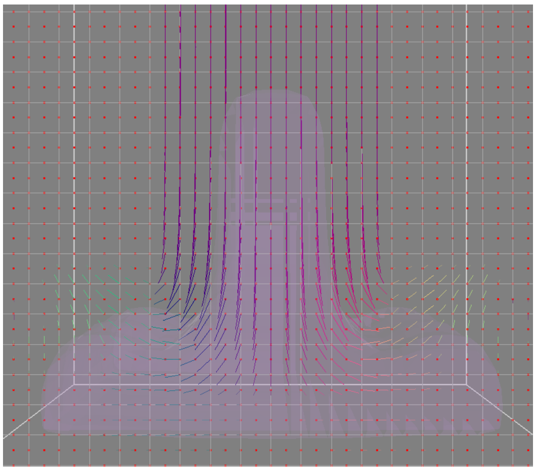


Figure 2: The velocity field visualized directly in a plane.

discretization schemes, where for example values in the self-advection term are linearly interpolated since the grid does not have infinite resolution.

Figure 2 shows the velocity field directly. It clearly shows the flow of the fluid that is used to move the interface.

Another thing to note is that the volume decreases during the simulation, likely caused by numerical diffusion.

3 Conclusion

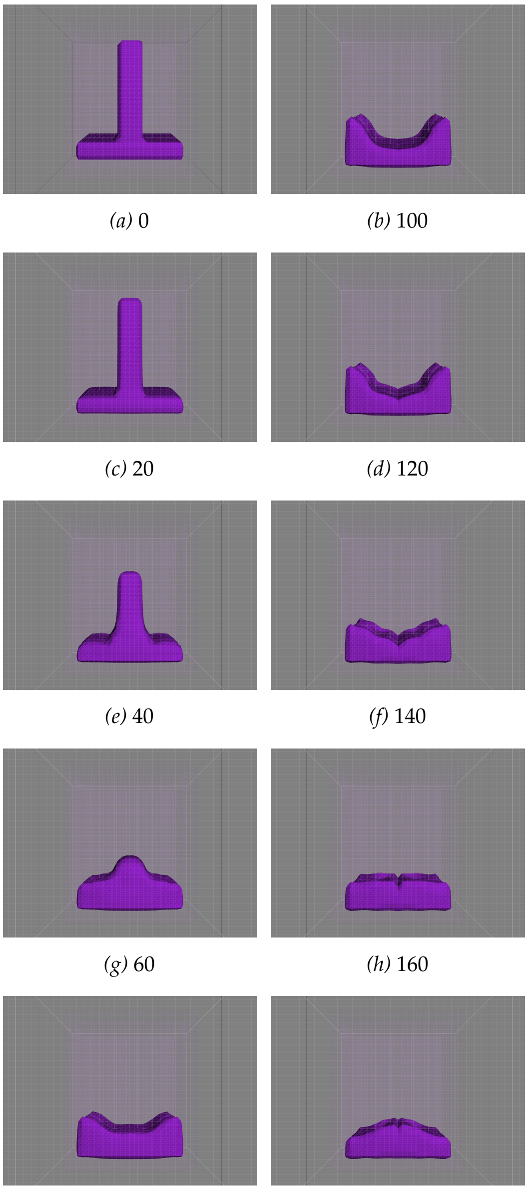
The fluid simulation effectively simulates a water-like fluid, although some aspects are not completely realistic, for example the significant loss of volume.

Lab Partner and Grade

The lab was done together with Viktor Sjögren. All lab tasks marked 3 or 4 were finished, and the report aims for grade 4.

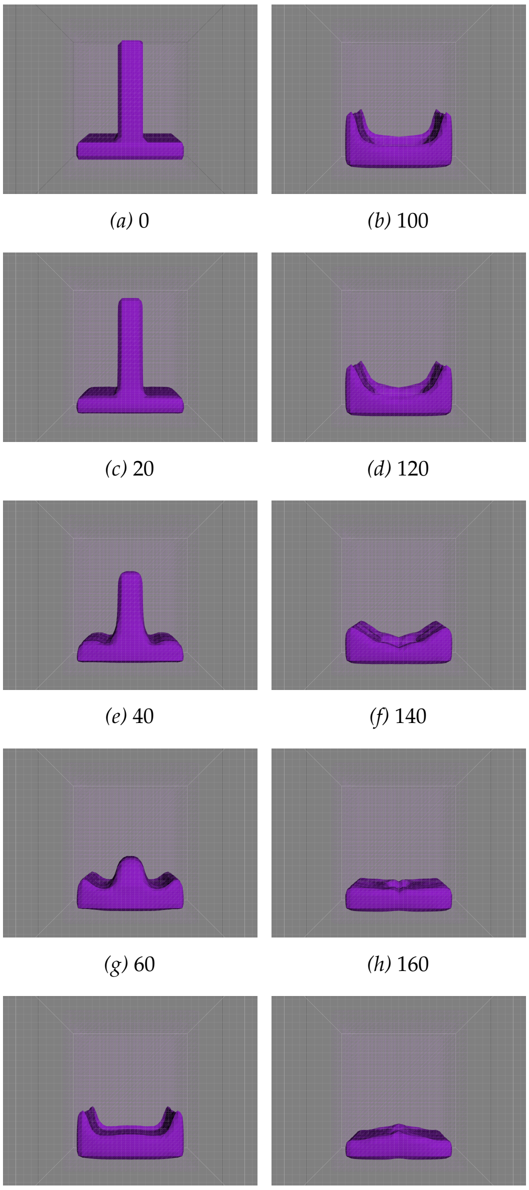
References

[1] J. Stam, "Stable fluids," *ACM SIGGRAPH 99*, vol. 1999, 11 2001.



(a) 0 *(b)* 100 *(c)* 20 *(d)* 120 *(e)* 40 *(f)* 140 *(g)* 60 *(h)* 160 *(i)* 80 *(j)* 180

Figure 3: Fluid simulation without the self-advection term shown for different number of iterations.



(a) 0 *(b)* 100 *(c)* 20 *(d)* 120 *(e)* 40 *(f)* 140 *(g)* 60 *(h)* 160 *(i)* 80 *(j)* 180

Figure 4: Fluid simulation with the self-advection term (full Euler equations) shown for different number of iterations.