LAB REPORT: LAB 6

TNM079, MODELING AND ANIMATION

Anna Wästling
annwa917@student.liu.se

Thursday 26th May, 2022

Abstract

This report describes the theory and implementation of simulating fluids using the Navier Stokes equations. The main equation that describes the flow of the fluid is split up into several parts where two are explicitly implemented in this report. The result of the implementation is presented and discussed.

1 Introduction

Although this report only focuses on the simulation of water, the Navier-Stokes equations can be used for several effects such as fire and wind making the Navier-Stokes equations very well known.

2 Background

The Navier-Stokes equations describes how the flow of a fluid changes over time. A vector field V is used to describe the flow where each vector in V is the velocity of the flow. Because of this V can also be referred to as the velocity field. The equations for incompressible flow is defined as

$$\frac{\delta V}{\delta t} = F + v\nabla^2 V - (V \cdot \nabla)V - \frac{\nabla p}{\rho} \tag{1}$$

$$\nabla \cdot V = 0 \tag{2}$$

This equation can be divided into several parts where F is the external force term, $v\nabla^2 V$ is the viscosity that control the thickness of the fluid, $(V \cdot \nabla)V$ is the self advection term that moves fluid with itself, p is the pressure field and ρ is the constant density. In this report the fluid water is simulated which has a viscosity close to zero meaning we can ignore this part without any big losses. The Navier-Stokes equations without the viscosity term is called the Euler equations and are not i the same order as in equation 2:

$$V_0 \xrightarrow{(V\cdot\nabla)V} V_1 \xrightarrow{F} V_2 \xrightarrow{\frac{\nabla p}{\rho}, \nabla\cdot V} V_{\Delta t} \tag{3}$$

$(V \cdot \nabla)V$ is as mentioned the self advection and represent non linear phenomenon. Since V is time dependent the following differential equation can be used

$$\frac{\delta V_1}{\delta t} = -(V_0 \cdot \nabla)V_0 \tag{4}$$

This method calculates V_1 by backwards tracing in time. This means that it is very dependent on which interpolation that is used.

The external force term can be solved in a simple way and the only external force in this report is gravity. The force field F is defined and creates V_2 from V_1 by solving:

$$\frac{\delta V_2}{\delta t} = F \tag{5}$$

This can be done by first order Euler time integration:

$$\frac{V_2 - V_1}{\Delta t} = F \Rightarrow \tag{6}$$

$$V_2 = V_1 + \Delta t \cdot F \tag{7}$$

The only step remaining is to enforce incompressibility by using $\frac{\nabla p}{\rho}, \nabla \cdot V$. A divergence free vector field is volume conserving and therefore incompressible. This is the same as $\nabla \cdot V = 0$. According to Helmholtz-Hodge decomposition it is always possible to split a vector field into a divergence free part V_{df} and a curl free part V_{cf} as $V_2 = V_{df} + V_{cf}$. Since $V_\Delta t$ should be divergence free we can refer to it instead of V_{df} and since a gradient field is curl free ∇q can be referred to instead of V_{cf} giving:

$$V_{\Delta t} = V_2 - \nabla q \tag{8}$$

To calculate $V_{\Delta t}$, q has to be known. The divergence operator can be applied to all terms in equation 8 and since $V_{\Delta t}$ is divergence free it can be canceled giving by some rewriting:

$$\nabla \cdot V_2 = \nabla^2 q \tag{9}$$

The discrete divergence operator for a point in the grid using central differencing is:

$$\begin{aligned}\nabla \cdot V_{i,j,k} = &\frac{u_{i+1,j,k} - u_{i-1,j,k}}{2\Delta x} + \\ &\frac{v_{i,j+1,k} - v_{i,j-1,k}}{2\Delta y} + \\ &\frac{w_{i,j,k+1} - w_{i,j,k-1}}{2\Delta z}\end{aligned} \tag{10}$$

where u,v and w are the x,y and components of every vector in V. The divergence of the gradient of q can be defined as the Laplacian applied to q. The discrete Laplacian in vector notation is:

$$\nabla^2_{q i,j,k} = \frac{1}{\Delta x^2} \begin{bmatrix} 1 & 1 & 1 & -6 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} q_{i+1,j,k} \\ q_{i-1,j,k} \\ q_{i,j+1,k} \\ q_{i,j,k} \\ q_{i,j-1,k} \\ q_{i,j,k+1} \\ q_{i,j,k-1} \end{bmatrix} \tag{11}$$

Since a uniform grid is used in this report $\Delta x = \Delta y = \Delta z$. By using the discrete Laplacian and the discrete divergence operator defined in equation 10, equation 9 can be defined as $Ax = b$. Where $A = \nabla^2$, x = q and

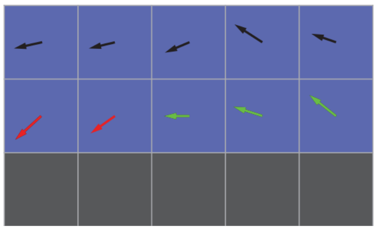
Figure 1: Application of the dirichlet boundary condition where the red arrows need to be redirected since they flow into the solid (grey blocks). From [1]

$b = \nabla \cdot V_2$. The A matrix is very big since it contains one row for each voxel that contains fluid. But A is also very sparse since every row at most will contain seven elements that are non-zero. These seven are the voxel itself and its six neighbours. The neighbours of the voxel(who is always fluid because of conditions) can be classified as either fluid, solid or empty. The following classifications are used in the report

$$\begin{cases} fluid, & \phi_{fluid} \leq \Delta x/2 \\ solid, & \phi_{solid} \leq 0 \\ empty, & \text{otherwise} \end{cases} \tag{12}$$

Where ϕ_{fluid} and ϕ_{solid} are level set distance function describing the fluid and solid. Two boundary condition are important to consider when solving equation 9. The first condition is the Dirichlet boundary that states that there can be no flow, in or out, of the boundary surface to which n is normal. Meaning it forbids fluid to flow into solid objects. It is defined mathematically as $V \cdot n = 0$ and can be visualized as in figure 1.

The second condition is the Neumann boundary condition:

$$\frac{\delta V}{\delta n} = 0 \tag{13}$$

Which forbids any flow along the normal direction of a solid surface and can be visualised in figure 2.

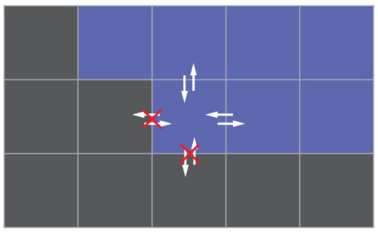

Figure 2: Application of the Neumann boundary condition where the arrows flowing into the solid (grey blocks) is not allowed. From [1]

3 Tasks

Implementation of the basic functionality for the fluid solver is explained in this section.

3.1 External forces

The external force in this lab was a previously stated only the gravity and was implemented by using equation 7 to add the forces to the velocity field.

3.2 Dirichlet boundary conditions

The Dirichlet boundary conditions was enforced by setting any velocity vector that was pointing towards a solid object to zero along the given dimension.

3.3 Projection

The projection step was implemented by using the conjugate gradient method, which is an iterative approach. First the divergence of the velocity field where computed by using equation 10. Then the A matrix was computed. To maintain the Neumann boundary condition the center voxels neighbour where checked to see if they where classified as solid. If they where the constant in front of that neighbour-voxel became 0 and the constant in front of the center voxel was increased by 1 for each solid neighbour. If the neighbour was classified as a solid or empty, fluid is allowed to flow and this is represented by a constant of 1 in front of the voxel. Lastly the divergence where subtracted from the velocity field to preserve volume by once again using the equation 10 to get the divergence.

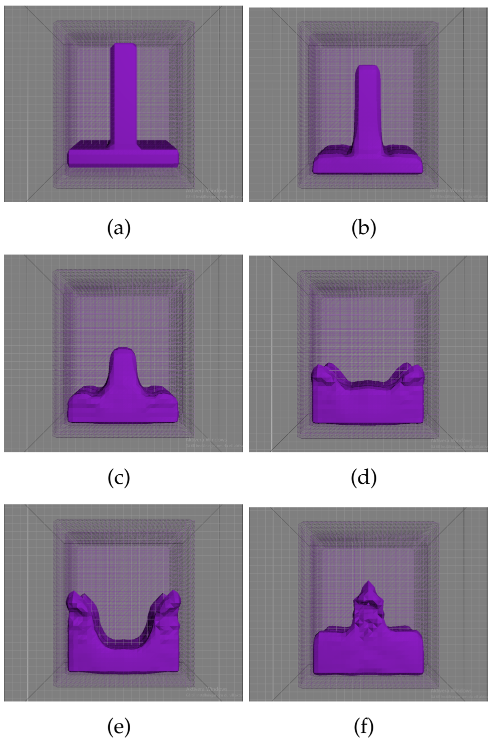

Figure 3: Result from task one showing from (a) to (f) screenshots of the implemented simulation of water

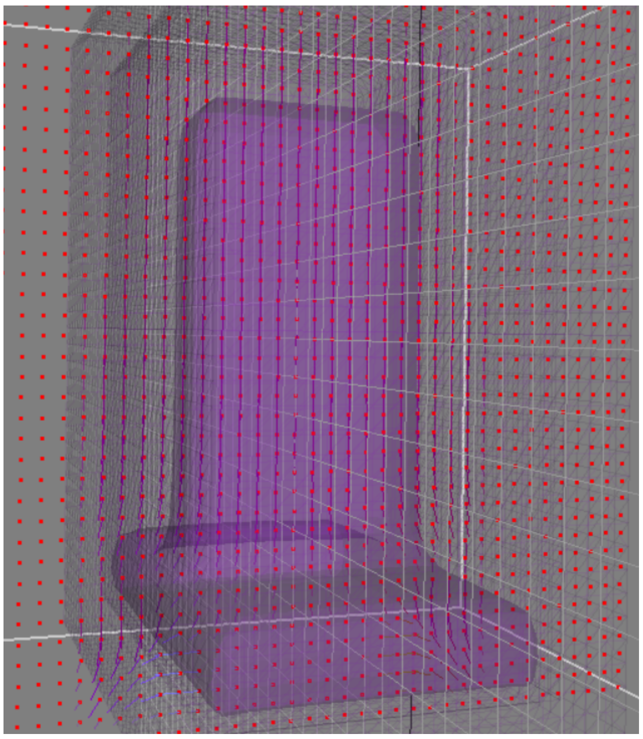

Figure 4: The velocity field

4 Results

The results of the simulation is shown in figure 3. It contains a source of error making it spill outside the box a bit. Otherwise it simulates according to the theory. In spite of the projection step the volume is not preserved. In figure 4 the velocity field can be inspected. The velocity vectors are pointed downwards since the fluid is affected by the gravity.

5 Conclusion

The resulting simulation does not do a good job of simulating water since it spills outside the box creating instability. Why this is the case is unknown but might be connected with the boundary conditions since neither the Dirichlet or Neumann boundary condition preventing the fluid to flow into the solid is not entirely maintained. These are however implemented correctly according to the theory and no mistakes could be found during the lab. As mentioned the volume was not entirely preserved and was mainly caused by numerical diffusion. Overall the Navier-Stokes equations is apprehensible and a appropriate way of simulation fluid.

6 Lab partner and grade

The lab was done together with Tim Olsson and aims for grade 3.

References

[1] Robin Skånberg Mark E. Dieckmann and Emma Broman. “fluid simulation”. 2021.