Real time fluid dynamics

Toffalini Leonardo

1. Notation

In this report we are going to heavily rely on differential operators, as this project is about solving a partial differential equation. Thus, before delving into the details of the main topic at hand, we shall quickly remind the reader of the notations that the following sections use. For a more complete introduction to the notations, we refer the reader to consult [1].

Let ∇ (formally) represent the $(\partial_1, \partial_2, ..., \partial_n)$ vector, where n is always deduced by the context the operator is used in. With the help of this vector we can define the usual gradient of a field as

$$\nabla u = (\partial_1 u, \partial_2 u, ..., \partial_n u).$$

Moreover, as ∇ is a vector, we can apply vector operations to it to get different operations, such as

$$u \cdot \nabla = \sum_{i=1}^{n} u_i \partial_i$$

or

$$(u \cdot \nabla)u = \sum_{i=1}^{n} u_i \partial_i u.$$

The Δ operator, called the **Laplace operator**, sometimes written as $\nabla \cdot \nabla$ or ∇^2, is defined as

$$\nabla \cdot \nabla u = \Delta u = \sum_{i=1}^{n} (\partial_i u)^2.$$

2. The equations of fluids

The advective form of the **Navier-Stokes equations** for an incompressible fluid with uniform viscosity are as follows:

$$\begin{aligned} \frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} &= \nu \Delta \boldsymbol{u} - \frac{1}{\rho}\nabla p + \frac{1}{\rho}\boldsymbol{f} \\ \nabla \cdot u &= 0 \end{aligned} \tag{1}$$

where

- $\boldsymbol{u}$ is the velocity vector field.
- $\boldsymbol{f}$ is the external forces.
- ρ is the scalar density field.
- p is the pressure field.
- ν is the kinematic viscosity.

We will not go into the details of deriving the above equations, instead we will just take it as granted that they truly formulate the evolving velocity of a viscous incompressible fluid.

For us to later simulate Equation 1 we need to understand what each part is responsible for.

$$\frac{\partial \boldsymbol{u}}{\partial t} + \overbrace{(\boldsymbol{u} \cdot \nabla)\boldsymbol{u}}^{\text{Advection}} = \underbrace{\nu \Delta \boldsymbol{u}}_{\text{Diffusion}} \quad \overbrace{-\frac{1}{\rho}\nabla p}^{\text{Internal source}} \quad + \quad \underbrace{\frac{1}{\rho}\boldsymbol{f}}_{\text{External source}} \quad .$$

In broad strokes the parts can be described as follows:

1. Advection – How the velocity moves.
2. Diffusion – How the velocity spreads out.
3. Internal source – How the velocity points towards parts of lesser pressure.
4. External source – How the velocity is changed subject to external intervention, like a fan blowing air.

The intrigued reader may find satisfaction in exploring the derivation of the above equations in [2].

3. Equations for fluid simulation

It is best to mention here, that the described method will not be exact. This, however, will not be to our detriment, as our aim here is to show interesting and realistic visuals as opposed to precise measurements useful for engineering efforts.

For our purposes we will rearrange Equation 1 such that we have only the time derivate of the velocity on the left, and omit the internal force part, as we will recover it later. So we arrive at the equation which we will solve:

$$\frac{\partial \boldsymbol{u}}{\partial t} = -(\boldsymbol{u} \cdot \nabla)\boldsymbol{u} + \nu \Delta \boldsymbol{u} + \frac{1}{\rho}\boldsymbol{f}.$$

The evolving of the velocity field in and of itself is not that interesting to see, because in real life we do not see the velocity field, we only see it's effect. To provide a more stimulating visual experience we must let the velocity field act upon a density field and in term visualize said density field. We present the equation for the density field

$$\frac{\partial \rho}{\partial t} = -(\boldsymbol{u} \cdot \nabla)\rho + \kappa \Delta \rho + S,$$

where κ is the diffusion coefficient of the fluid and S is a scalar field of external sources.

The astute reader might find this equation quite similar to that of the velocity field, which is by no means a coincidence, as the derivation of the density equation follows from the same physical laws, that resulted in the velocity equation. The only part that we must be mindful of, is that the density field's advection is dependent on the velocity field.

The previous system of partial differential equations by itself does not result in a unique solution for us to find, for this we must also specify a boundary condition. In our endeavor to simulate a fluid, we will work with the homogenous Neumann boundary condition, which prescribes that the normal derivatives on the boundary must vanish. We can formulate the previous statement as follows:

$$\partial_\nu u|_{\partial\Omega} = 0$$

where Ω represents the domain on which we are searching for the solution. Simply meaning that the fluid must not escape the domain.

4. Simulating fluids

The equations presented in the previous sections hold on the entire 2 or 3-dimensional space in which the fluid resides. However, when numerically solving a partial differential equation one often discretizes the space into a grid of small rectangles and only bothers to calculate the solution on these rectangles. For our purposes we will confine ourselves to the 2-dimensional plane, but everything mentioned hereafter can be easily extended to the third dimension.

We will make another simplification, which is to only consider a square domain. This makes the calculations easier to handle as matrix operations, however, one is free to extend the domain to more complex shapes by exercising proper caution when handling the boundary. Figure 1 shows a schematic diagram of how one must imagine a grid on a square domain. Notice, how we introduced a 0th and an $(N+1)$th row and column to handle the boundary.

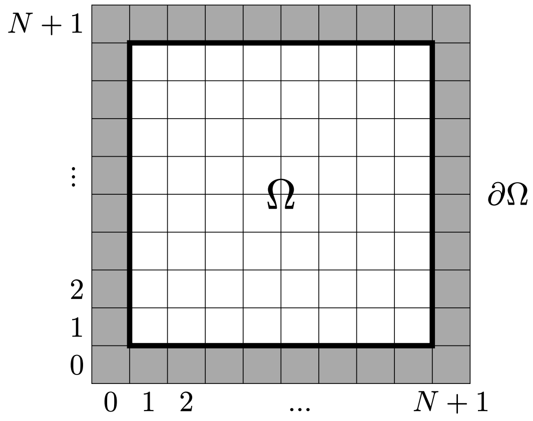


Figure 1: Discretization of the rectangular domain.

The equations, which we will be solving in our simulation are all of the form, where the time dependent partial derivative of the field is on the left hand side. To represent this time derivative of a field we will always keep two fields for ρ and $\boldsymbol{u}$, them being $\rho_{\text{prev}}, \rho_{\text{next}}$ and $\boldsymbol{u}_{\text{prev}}, \boldsymbol{u}_{\text{next}}$. Then we will approximate the time derivative with a simple forward difference scheme as

$$\frac{\partial \rho}{\partial t} = \frac{\rho_{\text{next}} - \rho_{\text{prev}}}{\Delta t},$$

where Δt is the time between the two frames of the simulation, known as the delta time. After each frame of the simulation we will swap the **prev** and **next** fields for ρ and $\boldsymbol{u}$.

5. Moving densities

In our simulation, we will solve the parts of the simplified Navier-Stokes equations for the density field one by one. Starting with the simplest, adding additional density, then diffusion, and lastly advection. The simulation steps can be seen in Figure 2, where one most imagine applying the steps in order from left to right, and repeating the last three steps, while the simulation is running.

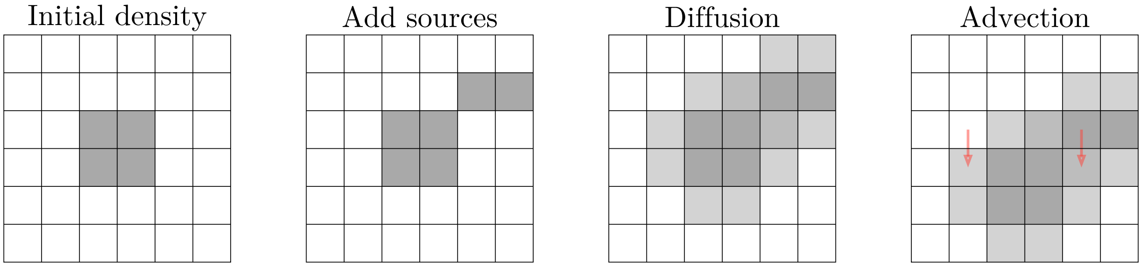


Figure 2: Outline of the density solver.

5.1. Adding sources

Possibly the simplest step of the simulation is to add new sources to the density field. For a rectangular grid this step can be simplified to a simple matrix addition $\rho_h + S_h$, where ρ_h is the density field on the discretized domain, and S_h is the matrix containing the sources at each grid cell.

5.2. Diffusion

During the diffusion step of the simulation we must solve the following equation

$$\frac{\partial \rho}{\partial t} = \kappa \Delta \rho.$$

As mentioned before, instead of $\partial \rho / \partial t$ we will use the forward difference scheme, changing the problem as follows

$$\frac{\rho_{\text{next}} - \rho_{\text{prev}}}{\Delta t} = \kappa \Delta \rho_{\text{prev}}$$

$$\rho_{\text{next}} = \rho_{\text{prev}} + (\Delta t) \kappa \Delta \rho_{\text{prev}}. \tag{2}$$

To solve this, we are going to employ the most intuitive method, known as the finite difference method, where we think of the density moving outwards from each cell to each of it's four neighbors, and density flowing in to it from it's neighbors. Figure 3 aids in visualizing the density exchange between the neighboring cells.

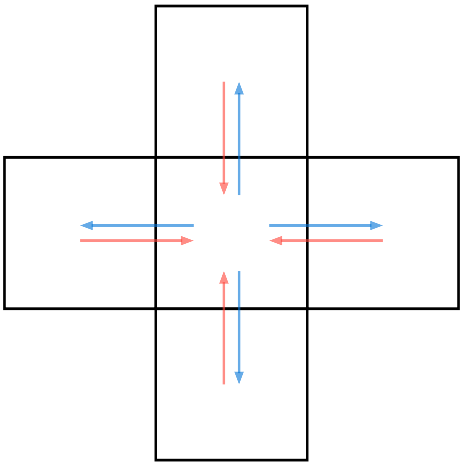


Figure 3: Five point density exchange intuition.

In a more formal sense, what this method does is it approximates the Laplacian $\Delta \rho$ with two second order finite difference schemes as

$$(\Delta_h \rho_h)_{i,j} = \frac{\rho_{i+1,j} + \rho_{i-1,j} + \rho_{i,j+1} + \rho_{i,j-1} - 4\rho_{i,j}}{h^2},$$

where h is the mesh fineness, that is $h = \frac{1}{N}$. We present an illustration of the Δ_h discrete Laplacian in Figure 4.

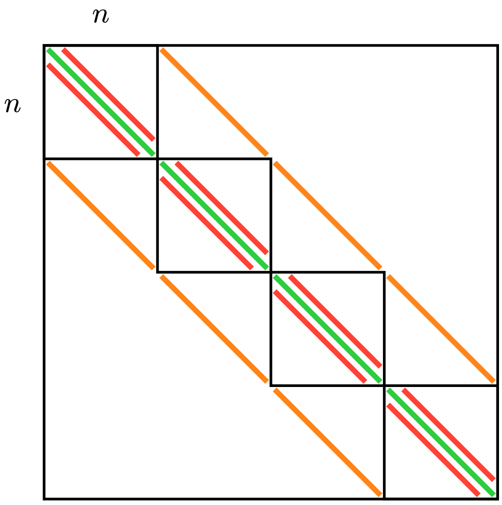


Figure 4: Discretization matrix of the Laplacian, where
green $= 4/h^2$, **red** $= -1/h^2$, **orange** $= -1/h^2$.

The previous discretization reduces Equation 2 to a linear system of equations of the form $A_h \rho_h = f_h$.

There are many ways to solve linear systems of this form, each of which has it's advantages and disadvantages. Since our goal is to simulate realistic fluid behavior in real time, at the expense of exactness, we will opt to use an iterative method, like the Gauss–Seidel method, which is fast but not exact.

Note, that we only mentioned the matrix of the discrete Laplacian. However, must not forget about the linear part of Equation 2, but this can be easily taken care of with a slight modification of the discretization matrix A_h.

For a more extensive treatment of the subject, the reader is advised to consult section 2.2 of [3] for deeper theoretical background, or https://github.com/leonardo-toffalini/fishy for implementation details.

5.3. Advection

For the advection step, we must solve the following equation

$$\frac{\partial \rho}{\partial t} = -(\boldsymbol{u} \cdot \nabla)\rho.$$

The tricky part with solving the advection step is that it is dependent on the velocity field, thus one must think of something clever to handle this difficulty.

The following novel idea that [4] and [5] present, is to think about fluid particles moving along the velocity field. We must think of our density grid as point masses centered at the middle of each cell, then tracing said point masses along the velocity field. The problem with this method, is that it will be unstable for certain parameters. However, we can fix this issue by instead of tracing the particles forwards along the velocity, we trace them back through time. This simply means that we trace back the origin of each particle that ended up in the center of a grid cell. Figure 5 provides visual understanding for the backwards path tracing.

After tracing back the possible locations where fluid particles could have come from we might get a particle that came from not the exact center of a cell. Remember, that we established that we shall think of the fluid as point masses centered at the middle of the grid cells. If a particle came from not the exact center then we must somehow give meaning to it too. In this case we will take the linear interpolation of the four closest neighbors of where the particle came from.

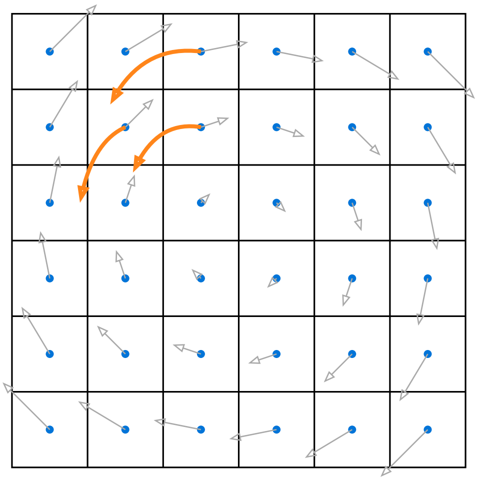


Figure 5: Tracing back the particle path along the velocity field.

6. Evolving velocities

Recall, that the velocity equation is almost the same as the density equation, thus, the method is almost complete as we can apply the previous diffusion and advection steps to the velocity field too. However, we need to recover the **internal source** part of the velocity equation, which we omitted in the second section.

This is where the second novel idea comes into play. The part we left out made sure that the velocity field was divergence free, that is $\nabla \cdot u = 0$, meaning that the velocity field was mass conserving. This is intuitive for incompressible fluids, that if some fluid flows in to a point, then an equal amount must flow out from that point.

Since we did not take care to hold the divergence free property during the diffusion and advection steps we quite possibly ended up with a velocity field which has non zero divergence. To combat this we rely on a result called the Helmholtz–Hodge decomposition from vector calculus, which states that a vector field can be decomposed as a sum of a field with no divergence and one which is the gradient of a scalar potential.

Formally, the Helmholtz–Hodge decomposition states that any vector field $\boldsymbol{w}$ can be uniquely decomposed into the sum of a divergence field ∇q and a rotation field $\boldsymbol{u}$, more concisely

$$\boldsymbol{w} = \boldsymbol{u} + \nabla q,$$

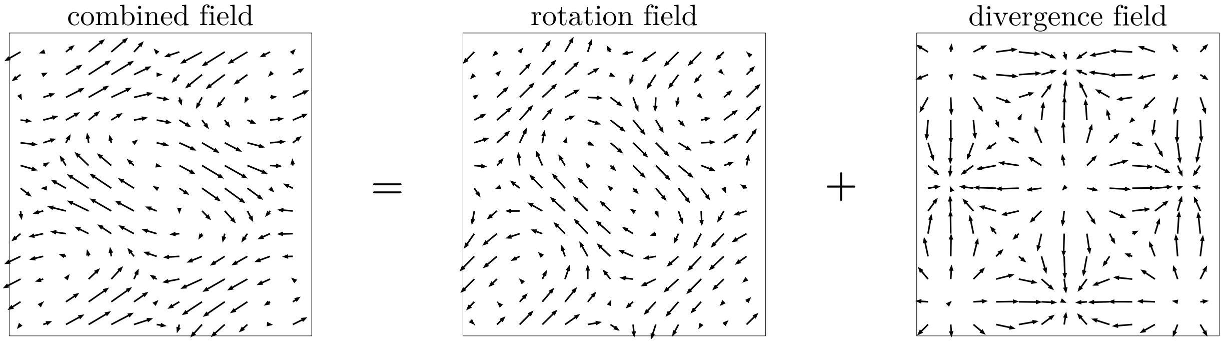
where $\nabla \cdot \boldsymbol{u} = 0$, and q is a scalar field. Figure 6 illustrates the idea of the decomposition.

Finding such a decomposition is almost as simple as stating the result, we only need to take the dot product with the ∇ operator of both sides, to get

$$\begin{aligned}\nabla \cdot \boldsymbol{w} &= \nabla \cdot \boldsymbol{u} + \nabla \cdot \nabla q\\ \nabla \cdot \boldsymbol{w} &= 0 + \nabla \cdot \nabla q\\ \nabla \cdot \boldsymbol{w} &= \Delta q.\end{aligned}$$

Since $\boldsymbol{w}$ is known to us, we can calculate it's divergence, making the left hand side some fixed value. Thus, the relation between $\boldsymbol{w}$ and q we just derived is a simple Poisson equation for q, which can be solved with the finite difference method we outlined in the diffusion section. After solving for q, we can extract $\boldsymbol{u}$ as

$$\boldsymbol{u} = \boldsymbol{w} - \nabla q.$$


combined field

rotation field

divergence field

Figure 6: Helmholtz–Hodge decomposition.

With this result in our hands we can finally resolve the mass conserving property of the velocity of the simulated fluid by decomposing the resulting field after the last step into a divergence free field.

In summary, one can then imagine the simulation steps for the velocity field as follows:

$$u_1 \xrightarrow{\text{add source}} u_2 \xrightarrow{\text{diffusion}} u_3 \xrightarrow{\text{advection}} u_4 \xrightarrow{\text{projection}} u_5$$

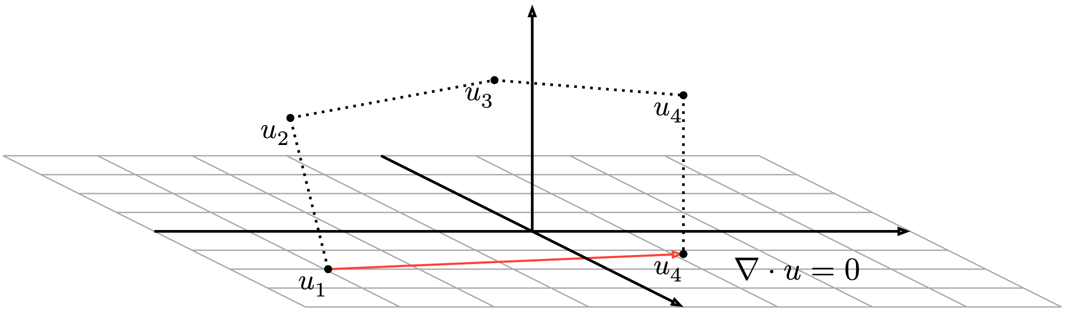
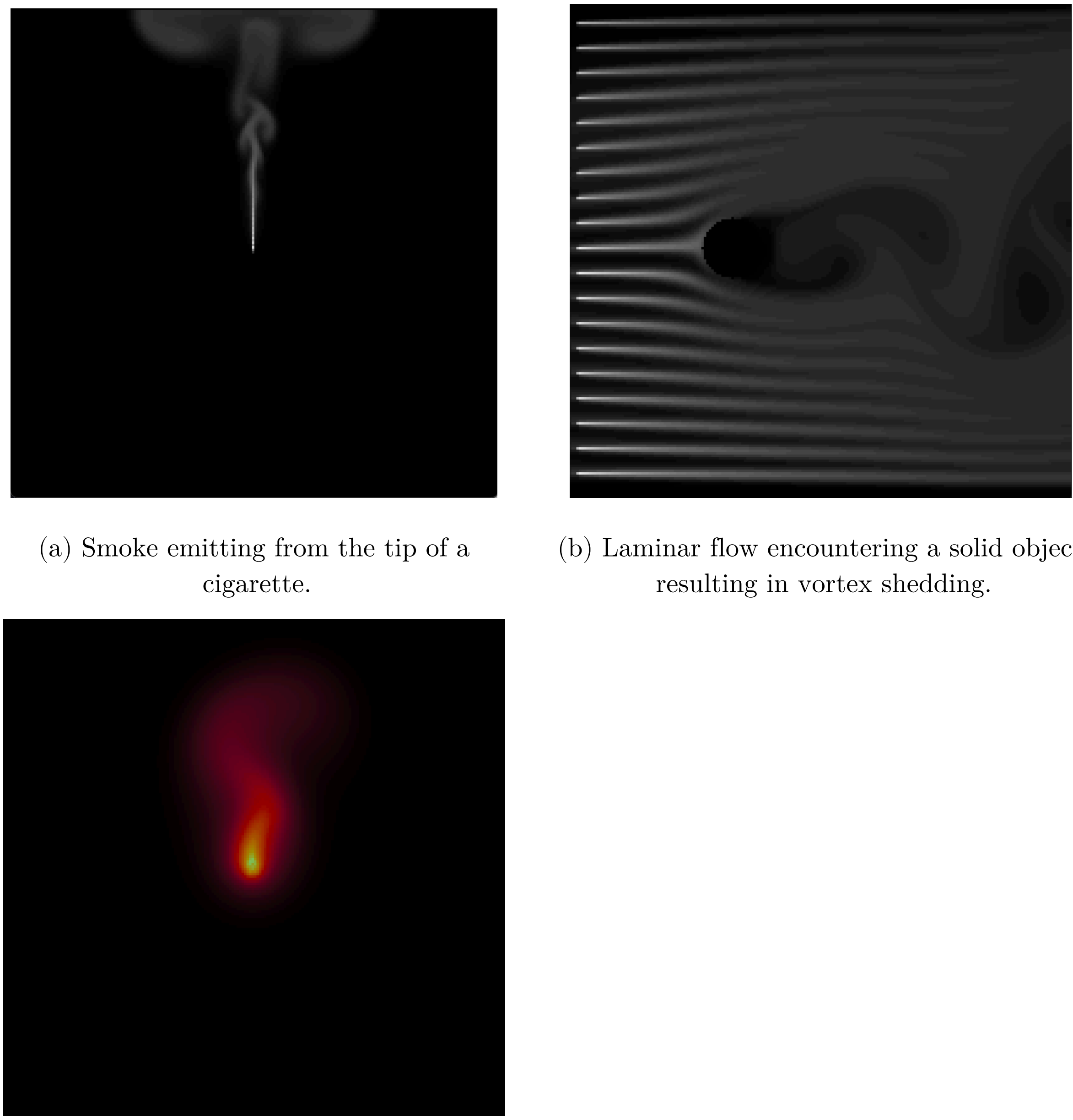


Figure 7: Illustrative steps of the method for the velocity field.

7. Appendix

All the code can be found at https://github.com/leonardo-toffalini/viscous.



(a) Smoke emitting from the tip of a cigarette.

(b) Laminar flow encountering a solid object, resulting in vortex shedding.

(c) Flickering fire.

Figure 8: Screenshots from the fluid simulation.

Bibliography

[1] Á. Besenyei, V. Komornik, and L. Simon, Parciális differenciálegyenletek. Budapest, 2013.

[2] A. J. Chorin, J. E. Marsden, and J. E. Marsden, A mathematical introduction to fluid mechanics, vol. 168. Springer, 1990.

[3] R. H. János Karátson, Numerical Methods for Elliptic Partial Differential Equations. [Online]. Available: https://kajkaat.web.elte.hu/pdnmell-ang-2023-I.pdf

[4] J. Stam, "Stable fluids," Seminal Graphics Papers: Pushing the Boundaries, Volume 2. pp. 779–786, 2023.

[5] J. Stam, "Real-time fluid dynamics for games," in Proceedings of the game developer conference, 2003.