Leap-Frogging Vortices

Weston Pace

September 2021

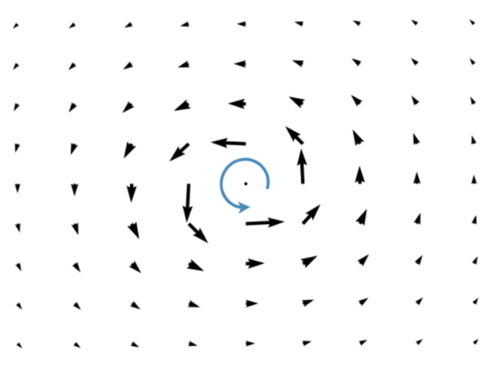
1 Introduction

A Vortex has the amazing ability to propel itself forward through its toroidal flow of a fluid. By itself and in a non-turbulent environment, a single vortex follows a relatively basic path of travel: moving in a straight line and holding its form. However, when a second vortex is introduced into the system, this ordinary action shifts into a more complex relationship. Instead of moving independently of one another, these vorticies move in a 'leap-frogging' manner, passing through one another sequentially.

The basic effect that a vortex has on itself and the other vortex can be modeled by the equation:

$$V_\theta = \frac{\Gamma}{2\pi r} \tag{1}$$

This equation leads to a model that can be visualized as such:



2 Methods

For this assignment, the top and bottom of each vortex were the points of interest being used to model the movement of the vortices. The high and low point of an individual vortex was intially spaced one meter apart, and the distance between the vortices was the same value.

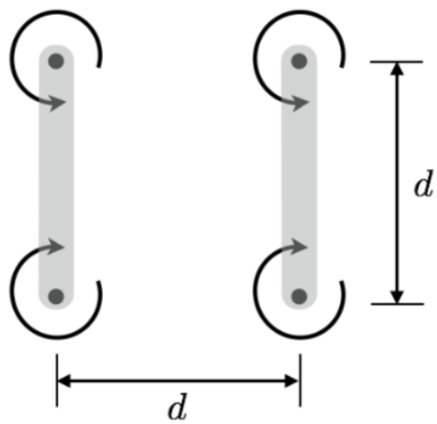


Figure 1: This visual represents the initial formation of the vortices

Although the movement of these vortices is difficult to track due to the constantly changing velocities, they are possible to mimic through the use of a program. For the model created in this report, Julia was the chosen language. By creating loops that calculated the velocities at each interval and multiplied these velocities by $\Delta t = 0.01$ seconds, a matrix was able to be created containing the position of the top and bottom of each vortex.

3 Results

Lines $y1$ and $y2$ represent the vortex that started in the back while lines $y3$ and $y4$ show the other. Though these vortices follow a complex movement, their paths are surprisingly simple when graphed. As shown below, the motion is very symmetrical, and the vorticies switch back and forth between two different patterns of movement (seen below).

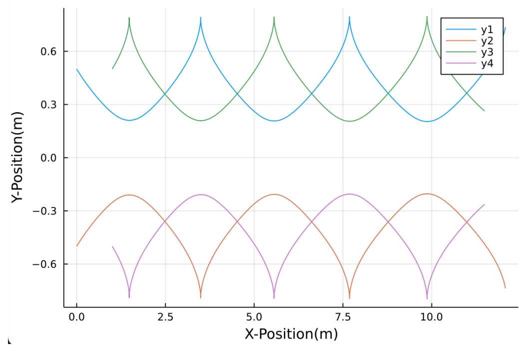


Figure 2: The parametric lines created from each point's matrix

By switching certain parameters of the original simulation, new and interesting results can be obtained. Some of these variations are shown below:

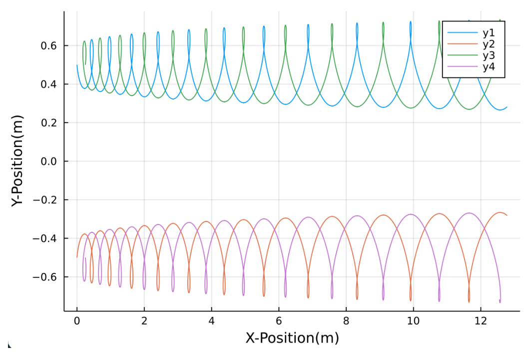


Figure 3: Vortex movement when the initial spacing between the vortices was 0.25 meters

As seen in figure 3, the two vortices have quick and somewhat abrupt 'leap-frogging' motions. However, as the simulation continues to run, their spacing increases, and their movements begin to resemble the original simulation.

In the case of figure 4 (below), the movement of the vortices is much slower and obtuse, and the time it takes for a single pass through to occur is much longer. Additionally, it took 32,000 interations in this simulation compared to the original 4,000 because the vortices had to travel much farther before a clear visual was created.

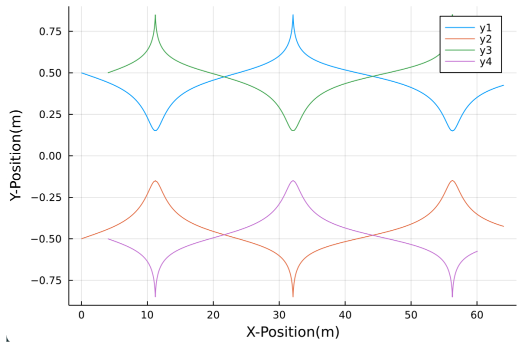


Figure 4: Vortex movement when the initial spacing between the vortices was 4 meters

What is learned from these three different cases is that the initial parameters of the simulation play a huge role in whether the effect of 'leap-frogging' is obtained or not; slight adjustments ultimately bring drastically different results. If considered in a real-world setting, this shows how difficult it is for this phenomenon to exist perfectly in nature. In the simulation, no turbulance was present; however, nature doesn't give conditions like this. An effect like this would completely disrupt the system. Additionally, as stated before, perfect distances are not going to always be present.

4 Conclusion

Working through this problem was extremely beneficial for me. Not only did it introduce me to an interesting phenomenon but it taught me how to use Julia and how to work through complex problems. On the surface, this project seemed pretty straight forward; however, as I continued to dive deeper into it, I realized that I completely underestimated the work required. Initially, I only had a two equations to model the movement of these vortices, but by the end, my code required nine in order to represent the simulation correctly. It was a battle of trial and error, and I was constantly learning new things as problems arose. Funnily enough, my final issue with this project was that I forgot that Julia worked in radians. I had most of my equations in terms of degrees, which was a huge detriment to the result, so when I changed these values to radians, I was extremely surprised and happy. Although this project was only an onboarding assignment, I still feel extremely proud of myself for accomplishing it since the coding and problem solving were harder than anything that I've ever done before. I hope that the skills learned through this assignment will be a foundation for the rest of my time in the lab, and I can't wait to keep obtaining more information through projects like this.