ONTARIO TECH UNIVERSITY

Faculty of Science, Computer Science

Simulation of a Flag Using the Verlet Mass-Spring System

by
Alireza Shahbazpour
Student ID: 100812651

Date: April 6, 2023

Abstract

This report presents the simulation of a cloth using the Verlet mass spring system. Simulating a flag is only a matter of lining anchors vertically and adding wind.

1. Introduction

Verlet integration is a numerical method used to solve differential equations. It is especially useful in computing such equations that exist in simulations like cloth, fluid, or any mass system with particles that feed information of and to each other.

2. Implementation

Verlet integration computes the future position of a moving particle. The advantage it has over other approaches and why I chose this is because it's fast and numerically stable. It is fast because we don't have to save velocity for the previous steps and only use the previous position to estimate the future position which also makes it numerically stable making it perfect for any mass particle system.

Springs for the particles are very simple. Calculate the difference in position for two particles and change their position based on the stiffness of the spring and how far away the particles are. Connecting the springs and particles in a grid manner would give us a cloth system.

Cloth system itself needs to be anchored somewhere which I implemented by not applying updates to a particle that has been tagged as an anchor. After making the anchor I realized I need the cloth to move in order to show it so I implemented dragging. which is as simple as disabling physics on a point and has been selected by the mouse and following the position of the mouse cursor.

To fully simulate a flag we need wind which at first I implemented by adding a constant to x and y position of every particle on the screen which looked very unnatural. To make it look more wavy I added a small variance to the wind force that changes over time from -C to C which creates the wavy form of the wind.

UI is simple pygame stuff with very basic button and main menu implementation.

Cloth tearing functionality is just see where the cursor is if it's between a line in between two points find and remove that constraint based on its position. This continues to remove constraints as long as the respective button is held down.

Color of constraints change depending on how far away its two particles are. The rest length is green.

Some optimization techniques I used: the constraints update twice per frame to make the simulation more stable at the cost of performance. I have adjusted the numbers for the most realistic looking cloth I can get for 60 frames per second. I added drag to the particles so they stop floating more than they should removed as much of the redundant code that I was able to do to squeeze more performance.

Further Improvements

Further improvements we can do: Using a better data structure than list for saving objects, deleting particles we don't need, better collision system allows for a full particle simulation (like sand simulation for example), UI and UX need improvement, The code isn't as scalable or modular as it should be, using a faster language would improve the performance drastically.

Conclusion

Verlet integration is a very fast, very simple solver that can handle many tasks and even a whole system very efficiently. Overall the resulting flag looks realistic enough given the scale of the project. I would continue to improve it afterward on a separate instance and honestly, I had fun doing the project.

References

Pekár, M. (2022, May 10). Verlet Integration and Cloth Physics Simulation. Pikuma. https://pikuma.com/blog/verlet-integration-2d-cloth-physics-simulation

My Github code:

https://github.com/Dantedanette0/Flag_test/blob/main/Object_creator.py

(I wont make any changes until I get the marks.)