KOÇ UNIVERSITY

College of Engineering

DEPARTMENT OF MECHANICAL ENGINEERING

SPRING 2021

MECH 534 PROJECT #1

FINAL REPORT

Berke Ataseven

54326

Instructor: Çağatay Başdoğan

Introduction and Abstract

In this project, we were asked to physically simulate a 3D flexible rope. We assumed the rope to be consisting particles each of which is connected to 2 other particles with a spring and a damper. The rope hangs from a ceiling as shown in the figure below.

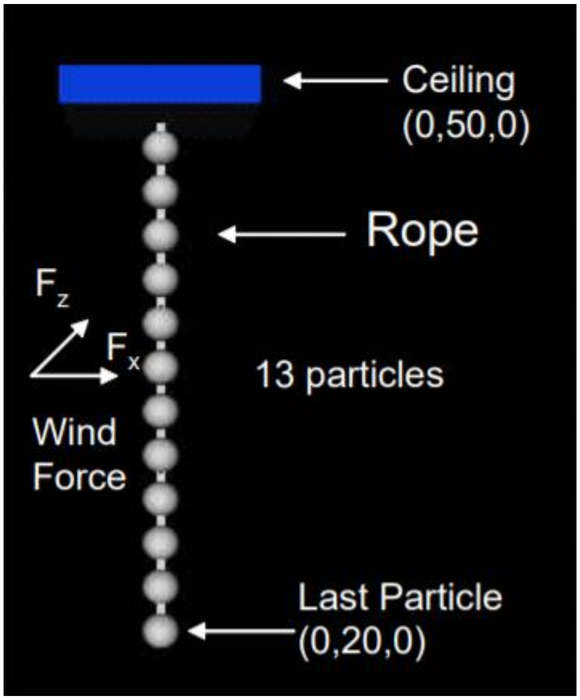

Figure 1 The simulated rope

Methodology

To simulate the rope, one must know the position of each particle on the rope. And that means solving a second-order differential equation for each particle. The equation of motions (EoM's) can be grouped into three.

- EoM of the first particle on the rope
- EoM of the last particle on the rope
- EoM of the other particles in between

The reason for this classification is that particles in the middle have similar EoM's (i.e., They are connected to another particle from the top and the bottom), whereas the top-most and the bottom-most particles have slightly different EoMs from the rest. Free body diagrams EoM's of these particles are shown in figure 2.

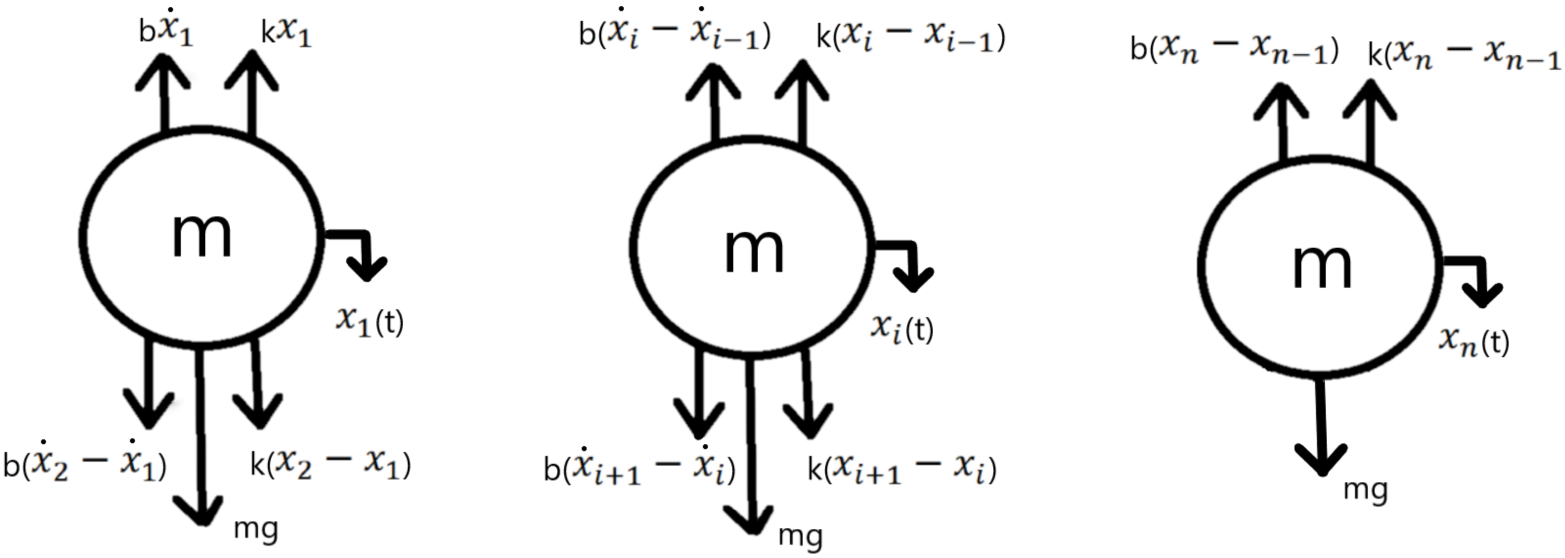

Figure 2 Equations of motions of particles. First particle, particles in between, last particle, respectively

Where, n is the number of particles (13 in our case) and i = 2,3,4, 5……, n. Additionally, each of the particles are under the influence of drag force which is in the opposite direction to their velocity. One can write the equations of motion as follows:

$$m\ddot{x}_1 = mg + b(\dot{x}_2 - \dot{x}_1) + k(x_2 - x_1) - b\dot{x}_1 - kx_1$$

$$m\ddot{x}_i = mg + b(\dot{x}_{i+1} - \dot{x}_i) + k(x_{i+1} - x_i) - b(\dot{x}_i - \dot{x}_{i-1}) - k(x_i - x_{i-1})$$

$$m\ddot{x}_n = mg - b(\dot{x}_n - \dot{x}_{n-1}) - k(x_n - x_{n-1})$$

In every time step dt, force applied to each particle is calculated. From the force, the acceleration is determined (F = ma). Velocity is calculated by integrating the acceleration. Finally, the position is updated in each time step by integrating the velocity. This basically is a simple implementation of Euler's method for solving ordinary differential equations.

Drag force is calculated using the following formula:

$$F_{drag} = C_d * V^2$$

Where C_d is the drag coefficient and V is the velocity. In an ideal drag force formula, there would be other constants such as the reference area of the object and the density of the fluid. In my implementation I have made the assumption that the product of these constants is defined as a single variable C_d.

Changes since Part 1

Wind forces from both Z and X directions can now be activated by pressing their respective buttons (z and x keys). Wind forces are exerted as long the keys are held pressed. Forces are removed when the key is released. Only one particle can be under the influence of the wind force. That particle is highlighted with red. Magnitude of the force exerted on particles are indicated by a cylinder on the right which transitions from a green color to a red color as the wind strength increases. Particle selection and the wind strength adjustment can be made using arrow keys. Simulation can be stopped and continued using the spacebar.

I have fixed the direction of the gravity. It was pointing in the reverse direction. Even though it did not interfere with the dynamics of the system, you had to rotate the camera by 180 degrees around the x-axis to see the simulation as intended.

Variables at the top of the code can be manipulated to change the properties of the simulated rope.

I have commented my code as much as possible for a better readability.