Amittai Joel Wekesa

COSC 89.18: Physical Computing

Assignment 2: Mass-Spring System

For the first part of the assignment, I implemented a function to calculate the spring force for any given spring index on the connected particles.

```
Vector3 Spring_Force(const int spring_index)
{
  //// This is an auxiliary function to compute the spring force f=fS+fD for the spring with spring_i
  //// You may want to call this function in Apply_Spring_Force

  /* Your implementation start */
  auto& spring = this->springs[spring_index];
  auto& i = spring[0];
  auto& j = spring[1];

  auto& xI = this->particles.X(i);
  auto& xJ = this->particles.X(j);
  auto xDiff = xJ - xI;
  auto distance = xDiff.norm();
  auto direction = xDiff.normalized();

  auto& vI = this->particles.V(i);
  auto& vJ = this->particles.V(j);
  auto vDiff = vJ - vI;

  auto& ksIJ = this->ks[spring_index];
  auto& kdIJ = this->kd[spring_index];

  auto& restLength = this->rest_length[spring_index];

  Vector3 fS = ksIJ * (distance - restLength) * direction;

  /* 0.05 scaling -- for some reason, this is necessary or else the motion is over-damped
      and dies super-fast */

  Vector3 fD = 0.05 * kdIJ * (vDiff - (vDiff.dot(direction) * direction));

  return fS + fD;
  /* Your implementation end */
}
```

I also implemented the force accumulation and time-step updates to velocity and position.

For the second part of the assignment, I implemented a model for a helical hair structure.

The hair I modelled closely resembles type 4*c* hair.

See video 4 for results.

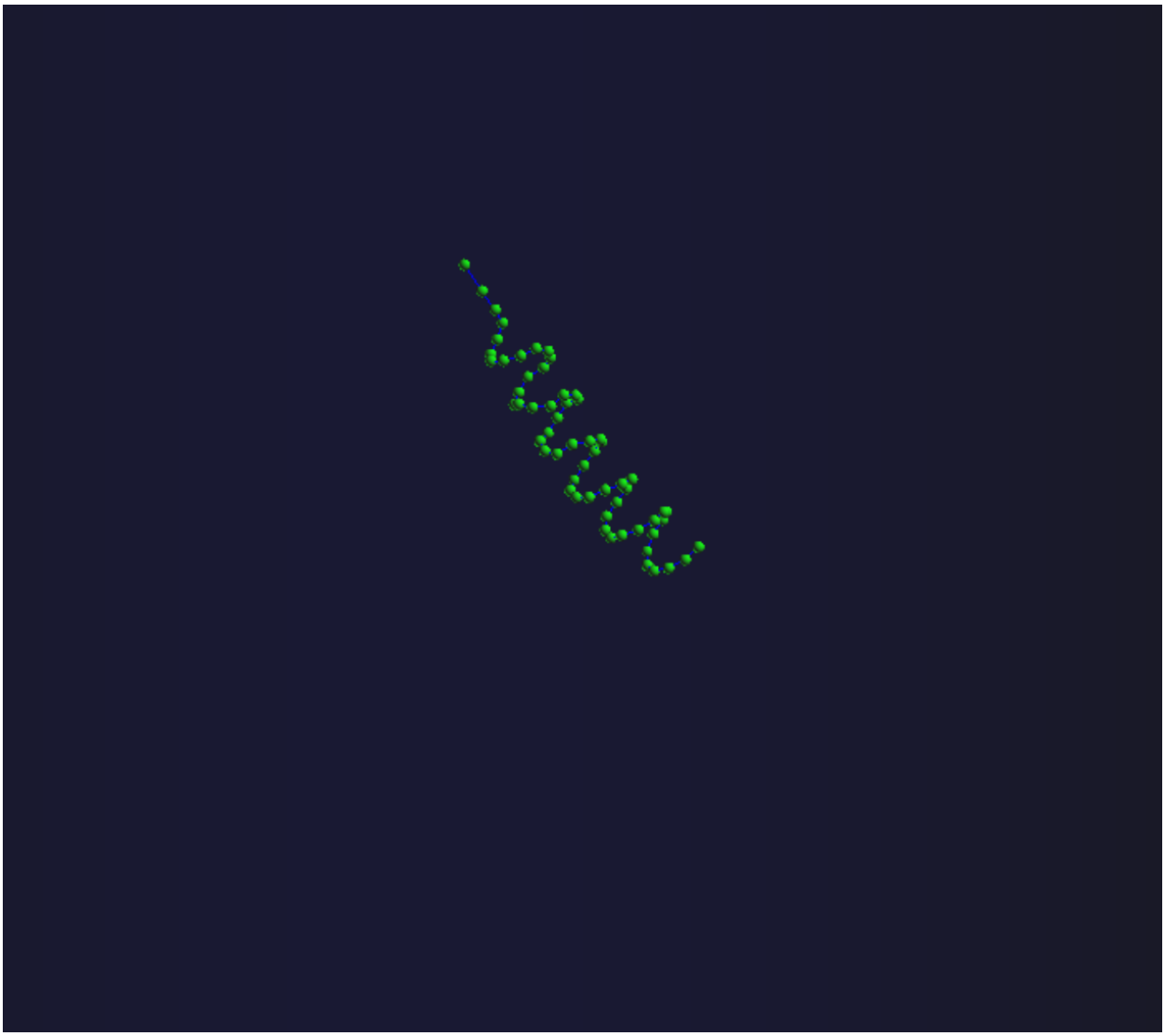

Figure 1: Screenshot of a Helical hair simulation