CS205b/CME306

Lecture 6

1 Springs

1.1 1D Mass Spring System

We begin with a 1D mass spring system consisting of a line of n points with mass m and $n-1$ identical springs connecting them. For a particular spring s_i, let x_i, x_{i+1}, v_i, and v_{i+1} be the position and velocity for the left and right ends of the spring. Let $\Delta x = x_{i+1} - x_i$ and $\Delta v = v_{i+1} - v_i$. As with the simple spring with a fixed endpoint, the force expected will be

$$F = -k_s\left(\frac{\Delta x}{\ell_0} - 1\right) - k_d \Delta v.$$

Note that translating the spring and examining it at a different point in space do not affect the force that it applies, so we may observe the spring from the position x_i of the left endpoint moving at its velocity v_i. From this vantage point, the spring looks *similar* to the simple spring with its non-fixed endpoint at location Δx with velocity Δv. (The system is different, though, in that neither endpoint is fixed. In particular, it will respond differently to the force applied by the spring. This does not affect the force the spring exerts, though.) Note that this is the force the spring applies to the right endpoint. The force applied to the left endpoint is $-F$ as required by Newton's third law. The frequency of the system is λ in Hertz (s^{-1}), and the sound speed is $c = \ell_0 \lambda$, in ms^{-1}. The sound speed looks like

$$c = \ell_0 \sqrt{\frac{k_s}{m\ell_0}} = \sqrt{\frac{k_s \ell_0}{m}}.$$

To get more accuracy, we may want to refine this discretization. We would like to double the number of points to $2n$, which increases the number of springs to $2n-1 \sim 2(n-1)$. To keep the total mass constant, we must replace the n nodes with mass m with $2n$ nodes with mass $\hat{m} = m/2$. To keep the length constant, we must replace the $n-1$ springs of length ℓ_0 with $2n-1$ springs of length $\hat{\ell_0} = \frac{n-1}{2n-1}\ell_0 \sim \ell_0/2$. The Young's modulus k_s is a characteristic of the material and does

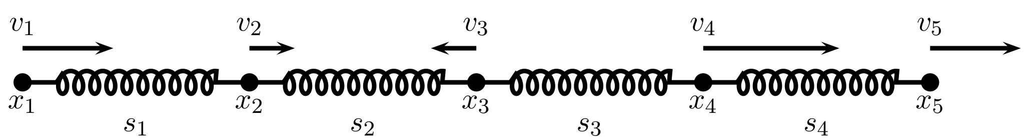


Figure 1: Row of springs.

not change. Ignoring the fencepost problem, the sound speed $\hat{c} = \sqrt{k_s \ell_0 / m} = c$ then remains the same, and the frequency λ doubles to

$$\hat{\lambda} = 2\lambda \sim 2\sqrt{\frac{k_s}{m\ell_0}}.$$

Note that halving the spring length but keeping the same sound speed forces the frequency to double, since information must travel through twice as many springs in the same amount of time. Since $\Delta t \lambda \approx 1$, the stable time step size Δt is halved to $\hat{\Delta t} = \Delta t/2$ due to the refinement. The fraction x/ℓ_0 that the material compresses or stretches does not change under the refinement, since both x and ℓ_0 are both halved. The resulting elastic force and acceleration for a refined spring are

$$\hat{F} = -k_s\left(\frac{x/2}{\ell_0/2} - 1\right) = F \qquad \hat{a} = \frac{F}{m/2} = 2a.$$

In particular, the force is unchanged, but the acceleration of the individual particles doubles. Finally k_{d_0} is a property of the material and should not change, so that

$$\hat{k_d} = k_{d_0}\sqrt{\frac{k_s m/2}{\ell_0/2}} = k_d.$$

In summary, if the resolution of the discretization is doubled, the various parameters of the system change as follows:

$$\hat{m} = \frac{m}{2} \qquad \hat{\ell_0} = \frac{\ell_0}{2} \qquad \hat{k_s} = k_s \qquad \hat{k_d} = k_d$$

$$\hat{c} = c \qquad \hat{\lambda} = 2\lambda \qquad \hat{\Delta t} = \frac{\Delta t}{2} \qquad \hat{F} = F \qquad \hat{a} = 2a$$

1.2 Zero Length Spring

Simply setting $\ell_0 = 0$ in the simple spring model causes problems. However, when modeling materials, this value is reached through the limit $n \to \infty$, $\ell_0 \to 0$, and $m \to 0$.

Another potential situation where a zero-length spring might be used is to connect two objects. One solution to this problem is to place the connection points an arbitrary distance $\ell_0/2$ inside each object. Letting s be the amount the joint is separated, the distance between the ends of the spring is $x = \ell_0 + s$, and its derivative is where $v = x' = s'$. The spring force is

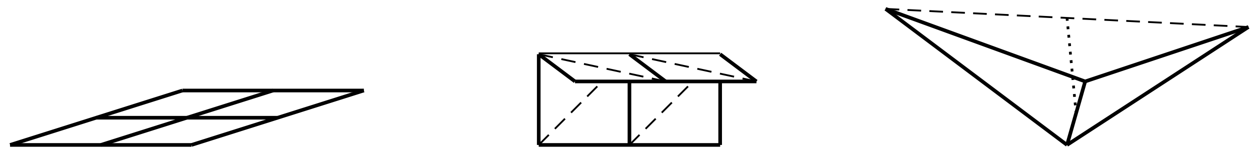
$$F = -k_s\left(\frac{x}{\ell_0} - 1\right) - k_d v = -k_s\left(\frac{\ell_0 + s}{\ell_0} - 1\right) - k_d v = -\frac{k_s}{\ell_0}s - k_d v.$$

The resulting system is then the same as before, except that it lacks the inhomogeneous term, and is

$$\begin{pmatrix} s \\ v \end{pmatrix}' = \begin{pmatrix} 0 & 1 \\ -\frac{k_s}{m\ell_0} & -\frac{k_d}{m} \end{pmatrix}\begin{pmatrix} s \\ v \end{pmatrix}.$$

We can now choose k_s and ℓ_0, or we could just factor out the arbitrary parameter ℓ_0 into the spring coefficient with

$$\hat{k_s} = \frac{k_s}{\ell_0} \qquad F = -\hat{k_s}x - k_d v \qquad k_d = k_{d_0}\sqrt{mk_s}.$$



(a) Edge springs do not prevent shear.

(b) Edge and shear springs do not prevent bending.

(c) Bending springs between two triangles.

Figure 2: Mass spring model for cloth.

1.3 2D Mass Spring System in 3D

Cloth may be modeled as a 2D surface that lives in 3D. The cloth surface might be discretized as a Cartesian grid of masses and springs, with masses at the corners connected along the edges by springs. The springs effectively restrict stretching and compression along the axes.

This cloth discretization lacks forces that are able to combat shear in the cloth. If a piece of cloth were modeled as a single square, one could fold the cloth into a line that is twice the edge length. Shearing the entire Cartesian grid of cloth has the same effect, illustrated in Figure 2(a). One way of solving this problem is to add springs along the diagonals of the squares in the cloth grid. One might choose to place springs on all of the left diagonals, all of the right diagonals, some mixture of left and right diagonals, or both left and right diagonals. Shear forces are different from stretching forces. Cloth tends to be very resistant to stretch but shears relatively easily, so shear springs are typically much weaker than edge springs. An alternative approach would be to use finite elements for the cloth, but for now we shall stick to springs.

This cloth model still has a major problem. This panel of cloth has no resistance to bending along the lines of springs that run along the axial directions, as in Figure 2(b). This may be fixed with additional springs that correct bending, such as the version obtained here by connecting opposing vertices of adjacent triangles (the shear springs create triangles). Bending forces in cloth are also typically very weak.

These bending springs have a couple potential limitations. If the desired angle between two triangles is nonzero, the bending spring will be unable to apply a force that would tend to correct the bend if the two triangles are coplanar or are bent in the wrong direction. Another issue with this model is that bending forces become very weak when the triangles are nearly coplanar. These two issues can be alleviated by connecting the shared edge between the triangles to the bending spring with yet another spring. This configuration is shown in Figure 2(c).

This setup works well in practice but may fail to apply forces in the right direction for some configurations of the cloth triangles. Note that the two triangles (formed by edge and shear springs) along with the original bending spring form a tetrahedron. achieving the desired lengths of these six springs is equivalent to achieving the desired shape for this tetrahedron. The tetrahedron may become inverted from its desired shape or degenerate. The second bending spring applies forces between a pair of opposing edges. In addition to this spring, there are two other edge-edge connections possible as well as four point-face possibilities (altitude springs). Between these seven possible springs locations, there will always be at least one choice (except in highly degenerate collinear configurations) that is capable of applying a suitable restoring force to the tetrahedron.