Differential Equations, Spring 2017

Computer Project 4, Due 11:30 am, Friday, May 5

Submit the following file on Canvas:

`string2.m`

Make sure the file has exactly this name.

The goal of this project is to model the motion of a vibrating string. Imagine a string, say a guitar string, which has length L meters and both ends are fixed in place. We will assume the string vibrates up and down only. Then the position of the string can be described by a function of two variables $u(t, x)$ representing the vertical displacement at time t and position x. (See the picture.) For example, the initial position of the string is the graph of the function $u(0, x)$. The initial velocity is $\frac{\partial u}{\partial t}(0, x)$. The condition that both ends are fixed is $u(t, 0) = u(t, L) = 0$ for all time t. The motion of the string is governed by the partial differential equation (PDE)

$$\frac{\partial^2 u}{\partial t^2} - c^2 \frac{\partial^2 u}{\partial x^2} = 0.$$

This equation is called the wave equation. c is a constant (related to the tension, mass and length of the string) which represents the speed of propagation of waves in the string.

We will use a system of ODEs to approximate the solution to the wave equation. The main idea is the following: instead of thinking of the string as a continuous object of uniform density, we will think of it as many small masses attached to each other by massless springs. The masses can vibrate up and down (horizontal vibration is negligible), and this corresponds to the string vibrating up and down. (See the picture.) We will use $N = 1,000$ small masses. Let $u_1(t), \dots, u_N(t)$ denote the vertical displacements of the masses (we will use u_i's as the dependent variables instead of the usual x_i's). Say the mass of the string is M kg, then each of the small masses has mass M/N. Let T Newtons be the tension in the string, then each small spring has spring constant $k = TN/L$ Newtowns per meter (this gives a total tension of T). Then $F = ma$ gives (see the picture)

$$\begin{aligned}
\frac{M}{N} x_1''(t) &= -k x_1 + k(x_2 - x_1), \\
\frac{M}{N} x_i''(t) &= k(x_{i-1} - x_i) + k(x_{i+1} - x_i), \quad (1 < i < N) \\
\frac{M}{N} x_N''(t) &= -k x_N + k(x_{N-1} - x_N).
\end{aligned}$$

(The first and last equations are different because the endpoints are held at displacement 0.) Using $k = TN/L$ gives the system

$$\begin{aligned}
x_1''(t) &= -\frac{TN^2}{ML} x_1 + \frac{TN^2}{ML}(x_2 - x_1), \\
x_i''(t) &= \frac{TN^2}{ML}(x_{i-1} - x_i) + \frac{TN^2}{ML}(x_{i+1} - x_i), \quad (1 < i < N) \\
x_N''(t) &= -\frac{TN^2}{ML} x_N + \frac{TN^2}{ML}(x_{N-1} - x_N),
\end{aligned}$$

This is a 2nd order, 1000 dimensional system of ODEs! It can be thought of as an approximation of the wave equation with $c^2 = TL/M$ (notice this has units of velocity squared). For the G string on a guitar, typical values of the constants are $M = .00107$ kg, $T = 66.61$ Newtons, $L = 1.0388$ meters. Use these as the values for M, T, L. (This gives $c = 254.3$ meters per second as the speed of waves in the string.)

Now we want to rewrite as a 1st order, 2000 dimensional system. Let $x_{N+1}, \ldots, x_{2N}$ be extra variables with

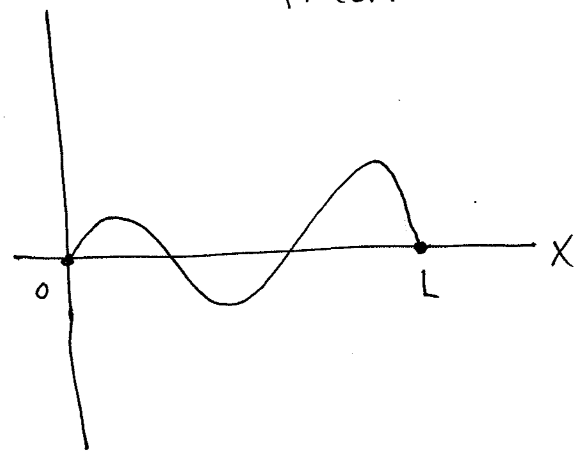
$$x_{N+1} = x_1', x_{N+2} = x_2', \ldots, x_{2N} = x_N'.$$

Rewrite as a first order system in the variables $x_1, \ldots, x_{2N}$. Then program the right hand side of the ODE into a Matlab function `string2.m`:

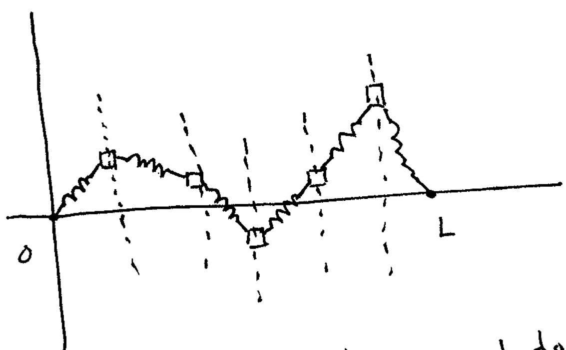
```
function val=string2(t,x)

end
```

You can test that it works by running the script `stringAnimation.m` (under computer projects section on the class website). You will also need to download `RungeKuttaMethod.m` (you can think of this as an improved Improved Euler Method).

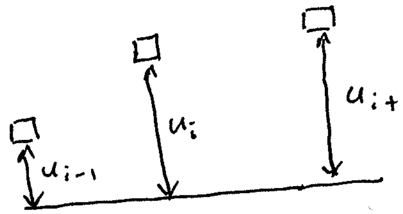
u = vertical displacement



approx. using springs



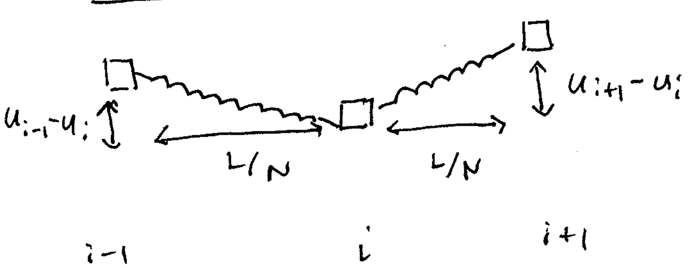
masses oscillate up and down
(negligible horizontal oscillation)



u_i = vertical displacement of ith spring

$u_i = u_i(t)$

Force on i^{th} mass $(1 < i < N)$



equil. length of spring $= L/N$

displacement from equilibrium $\approx \Delta u$
(if angle is small)

Force on i^{th} mass $= K(u_{i+1}-u_i) + K(u_{i-1}-u_i)$

For $i=1$, $i=N$