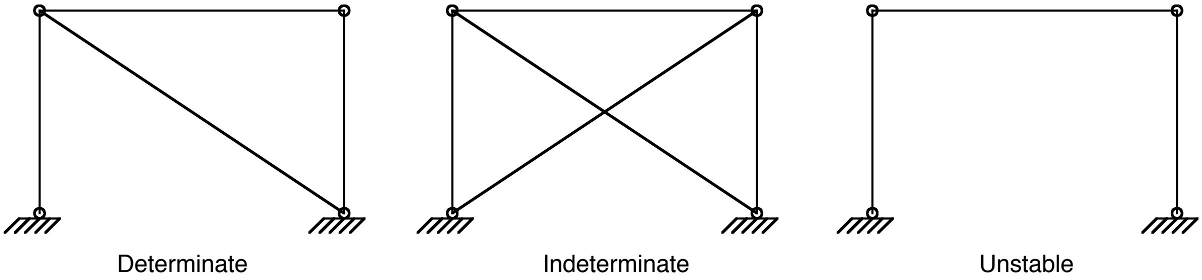
Project 1
Computer Analysis of Simple Structures
September 19, 2007

1 Trusses

This project focuses on developing a MATLAB program to provide a static analysis of a simple structure, called a truss. A truss is an idealized structure that is built from elastic bars. Each bar resists a change in its length, and that resisting force permits the structure to withstand a load. The bars are connected at points called nodes with pin joints, which can only transmit forces *along* the bars. The pin joints allow the bars to point in any direction. and each bar as a whole can turn. Bars are simpler than beams, plates, and shells. Unlike beams, they do not bend; unlike plates, they are one-dimensional; and unlike shells, they are simple and straight.

Figure 1 shows three trusses. All three have 4 nodes. The first has 4 bars; the second has 5 bars; and the third has only 3 bars. The diagram indicates that the two nodes on the bottom of each truss are fixed. That is, they are assumed to be immovable. This means that the two nodes at the top of each truss can each move in the x and y direction, which means that each of these trusses has two nodes each of which can move in two directions for a total of 4 degrees of freedom in terms of displacements. In addition each truss has an internal force acting along each bar. The first truss is called *Determinate* because it has 4 degrees of freedom in terms of displacements and exactly 4 internal forces. Determinate trusses are relatively easy to analyze by hand. The second truss is called *Indeterminate* because while it has exactly 4 degrees of freedom in terms of displacements, it has more than 4 internal forces. Completing its analysis means that we must determine the forces and displacements that minimize the potential energy of the system. The third truss is unstable. Fortunately, the linear algebra approach that we are going to explore will determine the unique set of displacements and internal forces for both determinate and indeterminate trusses and will also indicate when we have an unstable truss.

Figure 1: Three Trusses



Determinate

Indeterminate

Unstable

1.1 Specifying and Drawing the Truss

At this point we want to get things started by considering how to concisely describe the nodes and bars of a truss. Then we want to develop a computer program to read in this description and draw the corresponding diagram. To describe the nodes we need a list of the x-y coordinates for each node. In MATLAB the most direct approach is to simply state the number of nodes in the list and then list the x-y coordinates for each node on a separate line. If we assume that the trusses in Figure 1 are 3 units wide and 2 units high and that the bottom left node is at the origin, then the list of nodes would look as follows

```
4
0.0  0.0
3.0  0.0
0.0  2.0
3.0  2.0
```

A straightforward MATLAB code to read in this list from a file is the following:

```
name = input('Enter the name of the file defining the truss: ');
fid = fopen(name);
% Read in the number of nodes
nn = fscanf(fid,'%d',1)
% Read in the list of nodes - one x-y coordinate for each node
nodes = fscanf(fid,'%f %f',[2 nn])
```

Assuming that we have entered the description into a file called `truss1`, we will get the following results from running the preceding code.

```
Enter the name of the file defining the graph: 'truss01'

nn =
     4

nodes =
     0     3     0     3
     0     0     2     2
```

Note that the list of nodes is stored in a 2×4 matrix called `nodes` with the coordinates of each node stored in a column. We now need to indicate which nodes each bar is connected to. Keeping in mind the order in which we listed the nodes all we need to do is list the bars and for each bar indicate the two nodes that it connects. For the first truss in Figure 1, which has four bars, we would get the following list.

```
4
1 3
2 3
2 4
3 4
```

We can read in this description of the bars using code that is almost identical to the code that we used in reading in the nodes.

```
% Read in the number of bars
nb = fscanf(fid,'%d',1)
% Read in the list of bars - each bar connects two nodes
bars = fscanf(fid,'%d %d',[2 nb])
```

When we append the description of the bars to the file with the description of the nodes, and when we append this code to the previous code we get the following additional output.

```
nb =
     4


bars =
     1     2     2     3
     3     3     4     4
```

At this point, we have all the information we need to draw a diagram of the truss. First we plot the nodes. Since the first row of the nodes matrix contains the x coordinates and the second row contains the y coordinates we can plot all the nodes as small black circles with the following code

```
% Plot the nodes
xn = nodes(1,:);
yn = nodes(2,:);
plot(xn,yn,'ok')
```

Then we can add the file name as the title of the plot and we can use the `axis` command to provide a window with a margin around the diagram. We also add the `hold on` command so that we can plot the bars.

```
title(name)
% Set the view
xmin = min(xn)-0.5;
xmax = max(xn)+0.5;
ymin = min(yn)-0.5;
ymax = max(yn)+0.5;
axis([xmin xmax ymin ymax]);
hold on
```

Now we simply walk through the list of bars, and for each bar we draw a straight line from the first node of the bar to the second node of the bar. The x and y coordinates of these nodes are given in the corresponding columns of the `nodes` matrix. The following code implements these steps. However, since we had already copied the first and second rows of the `nodes` matrix into `xn` and `yn` arrays, using these arrays keeps the code a little shorter and a little clearer.

```
% Plot the bars
for k=1:nb
    % Get the index for the first node of bar k
    s = bars(1,k);
    % Get the index for the second node of bar k
    t = bars(2,k);
    plot([xn(s) xn(t)], [yn(s) yn(t)])
end
```

1.2 A Short Mathematical Detour

Before we delve into engineering questions there are a few things we can learn from the information we already have in hand. First, the diagram can also represent a mathematical object called a *graph*. This is not the same object as the graph of a function. This graph simply represents a set of points and connections between pairs of points. The points are usually called nodes and the connection between a pair of nodes is called an edge. The nodes of the graph correspond to the nodes of the truss, and the edges of the graph correspond to the bars in the truss. When we think of the diagram as representing a graph, then a line connecting two nodes represents an edge in the graph. On the other hand, when we think of the diagram as representing a truss, then a line connecting two nodes represents a bar of the truss.

We can construct a matrix, called the edge-node matrix, that represents the structure of the graph. This structural matrix contains one row for each edge and one column for each node. The row corresponding to a given edge will have all zero entries except in the columns corresponding to the two nodes. The entry for the first node will be -1 and the entry for the second node will be $+1$. If we consider the first diagram in Figure 1, then the edge-node matrix is

$$\mathbf{A}_1 = \begin{bmatrix} -1 & 0 & 1 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

If we consider the second diagram in Figure 1, then there is a fifth edge, and the edge-node matrix has an additional row.

$$\mathbf{A_2} = \begin{bmatrix} -1 & 0 & 1 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 1 \\ -1 & 0 & 0 & 1 \end{bmatrix}$$

These two edge-node matrices don't really contain any more information than we already had with the list containing the pair of nodes for each edge. However, this matrix structure leads to some interesting and useful calculations.

If we take the transpose of the edge-node matrix and multiply it times the edge-node matrix itself, we get a square matrix with the same number of rows and columns as the number of nodes. This matrix is called the *Laplacian matrix* of the graph. For the first diagram of Figure 1 we get

$$\mathbf{L_1} = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 2 & -1 & -1 \\ -1 & -1 & 3 & -1 \\ 0 & -1 & -1 & 2 \end{bmatrix}$$

For the second diagram of Figure 1 we get

$$\mathbf{L_2} = \begin{bmatrix} 2 & 0 & -1 & -1 \\ 0 & 2 & -1 & -1 \\ -1 & -1 & 3 & -1 \\ -1 & -1 & -1 & 3 \end{bmatrix}$$

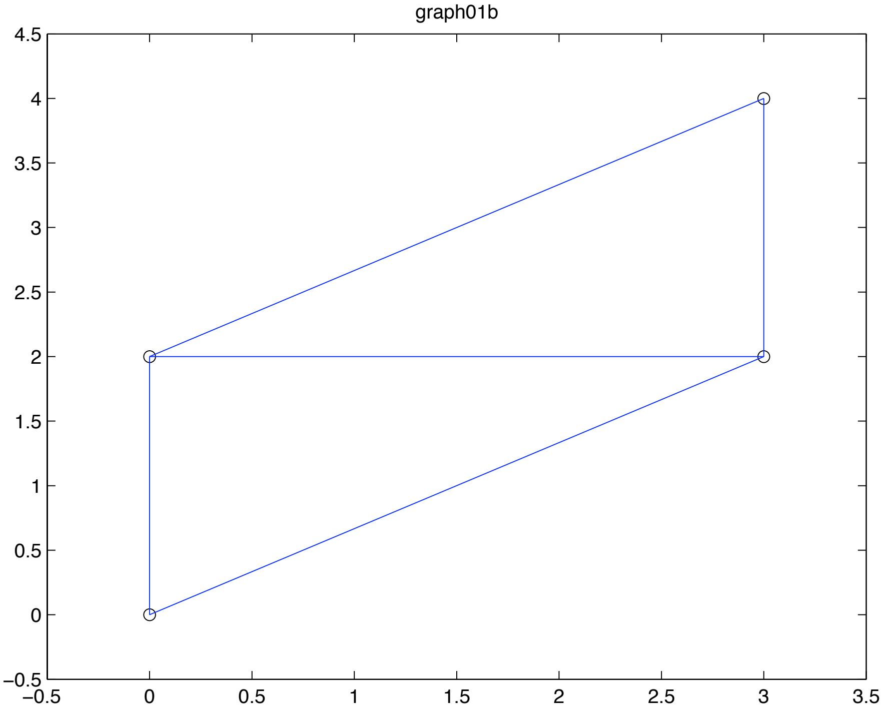
These Laplacian matrices have two particularly interesting properties. First, the entries on the diagonal count how many edges are connected to the corresponding node. For example, in the first diagram node 1 only has one edge connected to it, while in the second diagram it has two edges connected to it. Similarly, in the first diagram node 4 has two edges connected to it, while in the second diagram it has three edge connected to it. On the other hand, nodes 2 and 3 have the same number of edges connected to them in both diagrams. That is, two edges and three edges, respectively.

The second interesting property is that all the off-diagonal elements are either 0 or -1, and there is an edge connecting node i and node j if and only if the i-j entry is not zero. Because of these two properties, it is often a good idea to build the structural edge-node matrix and then compute the Laplacian matrix.

A final piece of easily computed information is the determinant of the Laplacian matrix after we have excluded the last row and column. This number will always be an integer that represents something both surprising and useful. A spanning tree of a graph is a set of edges such that it is possible to find a path from any node to any other node, but not have a path that loops back to where you started. This means that if you include any other edge of the graph you will have a loop. It is fairly easy to show that for a graph with n nodes a spanning tree will always have exactly $n-1$ edges. The surprising and useful result is called the Matrix-Tree Theorem. It states that the determinant of the Laplacian matrix with the last row and column removed is exactly the number of possible spanning trees for the graph. In other words it is the number of ways that you can pick $n-1$ edges from the graph so that it is possible to find a path from any node to any other node, but not have a path that contains any loops.

The first diagram has 3 spanning trees. This is easy to see, since there is only one loop, namely the loop from node 2 to node 3 to node 4 and back to node 2. Removing any one of the three edges from this loop will break the loop and the remaining edges (including the edge from node 1 to node 3) form a spanning tree. The second diagram has one additional edge, but it has 8 spanning trees. You may not find it easy to find all 8 given the diagram. However from the perspective of a graph the x-y coordinates only provide a way to draw the diagram. They do not play any role in the definition or the properties of the graph. If we change the y coordinate of the second node from 0 to 4 we get the diagram of the graph shown in Figure 2, and it becomes fairly easy to see how to identify the 8 different spanning trees.

Figure 2: Graph of Truss 1b



1.3 Back to Trusses

From the perspective of trusses, every spanning tree is an unstable truss. What's important from an engineering perspective are the additional bars that provide the strength and stability. It is also the case that the x-y coordinates of each node in a truss are extremely important in determining the strength and stability of the truss, so we need to develop a structural matrix that takes the geometry and the physics into account.

Suppose that we have a bar connecting nodes 1 and 2 with node 1 at (x_1, y_1) and node 2 at (x_2, y_2). We assume that the bar is very strong, but that under a load it gets displaced slightly so that nodes 1 and 2 are now at $(x_1 + \Delta x_1, y_1 + \Delta y_1)$ and $(x_2 + \Delta x_2, y_2 + \Delta y_2)$. Since the internal force of the bar is determined by how much the bar is stretched or compressed, we must determine a formula for the new length of the bar. The length of the bar before it is displaced is simply

$$L = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

The length of the bar after it is displaced is

$$\begin{aligned}
\bar{L} &= \sqrt{[(x_2 + \Delta x_2) - (x_1 + \Delta x_1)]^2 + [(y_2 + \Delta y_2) - (y_1 + \Delta y_1)]^2} \\
&= \sqrt{[(x_2 - x_1) + (\Delta x_2 - \Delta x_1)]^2 + [(y_2 - y_1) + (\Delta y_2 - \Delta y_1)]^2}
\end{aligned}$$

For a moment let's consider what happens if Δx_1 is nonzero and the other three displacements are zero. Then we have

$$\begin{aligned}
\bar{L} &= \sqrt{[(x_2 - x_1) + (-\Delta x_1)]^2 + (y_2 - y_1)^2} \\
&= \sqrt{(x_2 - x_1)^2 - 2(x_2 - x_1)\Delta x_1 + \Delta x_1^2 + (y_2 - y_1)^2} \\
&= \sqrt{L^2 - 2(x_2 - x_1)\Delta x_1 + \Delta x_1^2} \\
&= \sqrt{L^2 - 2L\cos(\theta)\Delta x_1 + \Delta x_1^2}
\end{aligned}$$

where θ is the angle the bar makes with the x-axis and $\cos(\theta) = (x_2 - x_1)/L$.

We can approximate this function of the displacement, Δx_1, with its Taylor polynomial centered at $\Delta x_1 = 0$.

$$\begin{aligned}
\bar{L} &= L - \frac{1}{2L}(2L\cos(\theta))\Delta x_1 + R \\
&= L - \cos(\theta)\Delta x_1 + R
\end{aligned}$$

Since we are assuming that the displacements are very small, the terms involving Δx_1^2 and higher powers can be neglected and we can approximate $\bar{L}$ by its tangent line (the Taylor polynomial of degree 1).

$$\bar{L} \approx L - \cos(\theta)\Delta x_1$$

If we repeat this process for the displacements Δx_2, Δy_1, and Δy_2, we can conclude that

$$\bar{L} \approx L - \cos(\theta)\Delta x_1 + \cos(\theta)\Delta x_2 - \sin(\theta)\Delta y_1 + \sin(\theta)\Delta y_2$$

Let e denote the elongation of the bar connecting nodes 1 and 2. Then we have

$$e = \bar{L} - L \approx -\cos(\theta)\Delta x_1 + \cos(\theta)\Delta x_2 - \sin(\theta)\Delta y_1 + \sin(\theta)\Delta y_2$$

Suppose that we have n nodes and m bars in our truss, then we can construct the following structural matrix, $\mathbf{A}_0$. This matrix will have m rows, one for each bar, and it will have $2n$ columns, two for each node, the first column in the pair corresponding to the horizontal displacement and the second corresponding to the vertical displacement. For each bar we will compute $\cos(\theta_k)$ and $\sin(\theta_k)$, where θ_k is the angle bar k makes with the x-axis. Let $\mathbf{u}_0$

denote the vector of displacements with $u_1 = \Delta x_1$ and $u_2 = \Delta y_1$, etc. Similarly, let $\mathbf{e}$ denote the vector of elongations. Then for small displacements the vector of elongations can be well approximated by the equation

$$\mathbf{e} = \mathbf{A}_0 \mathbf{u}_0$$

For the first truss in Figure 1 we get the following structural matrix

$$\mathbf{A}_0 = \begin{bmatrix} 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\cos\theta & -\sin\theta & \cos\theta & \sin\theta & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \end{bmatrix}$$

where $\tan(\theta) = -2/3$. For the second truss in Figure 1 we get the following structural matrix, which is the previous matrix with one additional row to account for the bar connecting node 1 with node 4.

$$\mathbf{A}_0 = \begin{bmatrix} 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\cos\theta & -\sin\theta & \cos\theta & \sin\theta & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ -\cos\phi & -\sin\phi & 0 & 0 & 0 & 0 & \cos\phi & \sin\phi \end{bmatrix}$$

where $\tan(\theta) = -2/3$ and $\tan(\phi) = +2/3$.

In Figure 1, nodes 1 and 2 are fixed, which means that their displacements can only be zero. This means that the first four columns of these two matrices can not contribute to the calculation of $\mathbf{e}$, the vector of elongations. Consequently, we can write the matrix equations as

$$\mathbf{e} = \mathbf{A}\mathbf{u} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \cos\theta & \sin\theta & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 1 & 0 \end{bmatrix} \mathbf{u}$$

and

$$\mathbf{e} = \mathbf{A}\mathbf{u} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \cos\theta & \sin\theta & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & \cos\phi & \sin\phi \end{bmatrix} \mathbf{u}$$

where $\mathbf{u}$ is the vector that only contains the displacements corresponding to nodes 3 and 4, which are the only possible nonzero displacements. One important thing to notice is that for the determinate truss, the first truss in Figure 1, the matrix $\mathbf{A}$ is square with 4 rows and 4 columns. However, for the second truss, the matrix $\mathbf{A}$ has 5 rows and 4 columns. This reflects the fact that this truss distributes its load over 5 bars. If the loaded truss is in equilibrium, then the displacements of the nodes will be such that the potential energy of the truss is minimized.

1.4 Hooke's Law and Potential Energy

Consider an undamped spring-mass system consisting of a single mass, m, and a single spring with spring constant, c. Newton's Second Law states that force equals mass times acceleration. If $x(t)$ denotes the distance that the mass has been moved from its equilibrium point at time, t, then Newton's Second Law yields the differential equation

$$m\frac{d^2x}{dt^2} = -cx \tag{1}$$

Since the velocity of the mass is $v(t) = dx/dt$, we can write Equation 1 as

$$m\frac{dv}{dt} = -cx \tag{2}$$

If we multiply both sides by v, using v on the left and dx/dt on the right, we get

$$mv\frac{dv}{dt} = -cx\frac{dx}{dt}$$

And integrating from 0 to T with respect to t, we get

$$m\int_0^T v\frac{dv}{dt}\,dt = -c\int_0^T x\frac{dx}{dt}\,dt$$

and after applying the u-substitution rule we have

$$m\int_{v(0)}^{V(T)} v\,dv = -c\int_{x(0)}^{x(T)} x\,dx$$

Since the two integrals are equal, we have obtained the conservation of energy property for the spring-mass system.

$$\frac{1}{2}mv^2(T) + \frac{1}{2}cx^2(T) = \frac{1}{2}mv^2(0) + \frac{1}{2}cx^2(0) \tag{3}$$

The first term is the kinetic energy of the moving mass and the second term is the potential energy (or work) stored in the elongated spring. The equation states the total energy at time, $t = T$, remains the same as the total energy at time $t = 0$. The important point in the context of a spring in equilibrium is that the potential energy in the spring is $cx^2/2$ where x is the amount that the spring has been elongated (stretched or compressed).

Now consider three suspended masses (m_1, m_2, and m_3) and three springs with spring constants (c_1, c_2, and c_3). We start by assuming that the masses are suspended one below the other with the first spring connecting the first mass to the ceiling, the second spring connecting the second mass to the first mass, and the third spring connecting the third mass to the second mass. Let x_1, x_2, and x_3 denote the distance each mass has moved under the force of gravity, that is the displacement of each mass. For this problem it is convenient to treat moving down as a positive displacement. For convenience we can let $x_0 = 0$ denote the displacement of the ceiling. Then e_1, the elongation of the first spring, is given by $e_1 = x_1 - x_0$. Similarly, $e_2 = x_2 - x_1$ and $e_3 = x_3 - x_2$. This can be written as the matrix equation

$$\mathbf{e} = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

Since $x_0 = 0$ (the ceiling doesn't move) we can delete the column involving x_0 to get

$$\mathbf{e} = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \mathbf{Ax}$$

The force acting on each spring is proportional to the spring constant times the elongation, so we can let $\mathbf{y}$ denote the internal force on each spring and write

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} c_1 & 0 & 0 \\ 0 & c_2 & 0 \\ 0 & 0 & c_3 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \mathbf{Ce}$$

The external forces, $\mathbf{f}$, on the masses are their weights, so

$$\mathbf{f} = \begin{bmatrix} gm_1 \\ gm_2 \\ gm_3 \end{bmatrix}$$

Since this spring-mass system is in equilibrium, the forces acting on each node must balance.

$$\begin{aligned} f_1 &= y_1 - y_2 \\ f_2 &= y_2 - y_3 \\ f_3 &= y_3 \end{aligned}$$

Writing this condition as a matrix equation we get

$$\mathbf{f} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \mathbf{A}^T\mathbf{y}$$

It is important to notice that the structural matrix, $\mathbf{A}$, relates the displacements, $\mathbf{x}$, to the elongations of each bar, $\mathbf{e}$, while its transpose, $\mathbf{A}^T$ relates the external forces at the nodes to the internal forces on the springs.

The spring mass system can be summarized by the three equations

$$\begin{aligned} \mathbf{e} &= \mathbf{Ax} && (4) \\ \mathbf{y} &= \mathbf{Ce} && (5) \\ \mathbf{f} &= \mathbf{A}^T\mathbf{y} && (6) \end{aligned}$$

It is an easy matter to see that for the problem with three springs and three masses we can solve Equation 3 for $\mathbf{y}$, then solve Equation 2 for $\mathbf{e}$, and then finally solve Equation 1 for $\mathbf{x}$.

Suppose that we add a fourth spring connecting the third mass to the floor. Then, assuming that the floor doesn't move, $x_4 = 0$, we get the following matrix equations

$$\mathbf{e} = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

Since $x_0 = x_4 = 0$ (the ceiling and floor don't move) we can delete the columns involving x_0 and x_4 to get the elongations for the four springs

$$\mathbf{e} = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \mathbf{Ax}$$

The force acting on each spring is proportional to the spring constant times the elongation, so the internal forces on the springs are given by

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} c_1 & 0 & 0 & 0 \\ 0 & c_2 & 0 & 0 \\ 0 & 0 & c_3 & 0 \\ 0 & 0 & 0 & c_4 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{bmatrix} = \mathbf{Ce}$$

Since this spring-mass system is in equilibrium, the forces acting on each node must balance.

$$\mathbf{f} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \mathbf{A}^T\mathbf{y}$$

For this case the structural matrix, $\mathbf{A}$, is 4×3, since it relates the three displacements to the internal forces acting on the four springs. We can determine the four elongations given the three displacements, but we can't be guaranteed to find a consistent set of displacements given four arbitrary elongations. The first three elongations would determine the x values and the fourth elongation would either be redundant or inconsistent.

Even though the structural matrix, $\mathbf{A}$, is not square it is still the case that the matrix which relates $\mathbf{f}$, the external forces at the nodes, to $\mathbf{y}$, the internal forces on the springs, is the transpose of $\mathbf{A}$, the matrix which relates $\mathbf{x}$, the displacements of the nodes, to $\mathbf{e}$, the elongations of the springs.

In order to determine the effect of the external forces on the four spring model, we have to determine the potential energy of the system. There are two components. The work needed to restore the masses to their original positions is

$$-(x_1 f_1 + x_2 f_2 + x_3 f_3) = -\mathbf{x}^T \mathbf{f}$$

and, as we saw earlier, the potential energy stored in the springs is

$$\begin{aligned} \frac{1}{2}c_1 e_1^2 + \frac{1}{2}c_2 e_2^2 + \frac{1}{2}c_3 e_3^2 + \frac{1}{2}c_4 e_4^2 &= \frac{1}{2}\mathbf{e}^T \mathbf{C}\mathbf{e} \\ &= \frac{1}{2}\mathbf{x}^T \mathbf{A}^T \mathbf{C}\mathbf{A}\mathbf{x} \end{aligned}$$

Consequently the total potential energy of the system, as a function of the displacements, is

$$P(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{A}^T \mathbf{C}\mathbf{A}\mathbf{x} - \mathbf{x}^T \mathbf{f} \tag{7}$$

Note that if there were only one possible displacement, then this would be a simple quadratic equation of the form

$$P(x) = \frac{1}{2}x\alpha x - xf = \frac{\alpha}{2}x^2 - fx$$

and we can find the minimum by determining where the derivative is zero. That is, solve the equation

$$P'(x) = \alpha x - f = 0$$

As it turns out, the analogous thing is true for Equation 7. If $\mathbf{x}$ satisfies the equation

$$\mathbf{A}^T \mathbf{C}\mathbf{A}\mathbf{x} = \mathbf{f}$$

then $P(\mathbf{x})$ is minimized.

The argument is very straightforward. Suppose that $\mathbf{u}$ is any vector, and let $\mathbf{v} = \mathbf{u} - \mathbf{x}$ then

$$\mathbf{u} = \mathbf{v} + \mathbf{x}$$

and

$$\begin{aligned} P(\mathbf{u}) &= \frac{1}{2}\mathbf{u}^T \mathbf{A}^T \mathbf{C}\mathbf{A}\mathbf{u} - \mathbf{u}^T \mathbf{f} \\ &= \frac{1}{2}(\mathbf{v}+\mathbf{x})^T \mathbf{A}^T \mathbf{C}\mathbf{A}(\mathbf{v}+\mathbf{x}) - (\mathbf{v}+\mathbf{x})^T \mathbf{f} \\ &= \frac{1}{2}\mathbf{v}^T \mathbf{A}^T \mathbf{C}\mathbf{A}\mathbf{v} + \frac{1}{2}\mathbf{v}^T \mathbf{A}^T \mathbf{C}\mathbf{A}\mathbf{x} + \frac{1}{2}\mathbf{x}^T \mathbf{A}^T \mathbf{C}\mathbf{A}\mathbf{v} + \frac{1}{2}\mathbf{x}^T \mathbf{A}^T \mathbf{C}\mathbf{A}\mathbf{x} - \mathbf{v}^T \mathbf{f} - \mathbf{x}^T \mathbf{f} \end{aligned}$$

Since $\mathbf{v}^T \mathbf{A}^T \mathbf{C}\mathbf{A}\mathbf{v}$ is a 1×1 matrix, in other words, an ordinary number, we have

$$\mathbf{v}^T \mathbf{A}^T \mathbf{C}\mathbf{A}\mathbf{v} = \left(\mathbf{v}^T \mathbf{A}^T \mathbf{C}\mathbf{A}\mathbf{v}\right)^T = \mathbf{x}^T \mathbf{A}^T \mathbf{C}\mathbf{A}\mathbf{v}$$

and consequently

$$\begin{aligned} P(\mathbf{u}) &= \frac{1}{2}\mathbf{v}^T\mathbf{A}^T\mathbf{C}\mathbf{A}\mathbf{v} + \mathbf{v}^T\mathbf{A}^T\mathbf{C}\mathbf{A}\mathbf{x} - \mathbf{v}^T\mathbf{f} + \frac{1}{2}\mathbf{x}^T\mathbf{A}^T\mathbf{C}\mathbf{A}\mathbf{x} - \mathbf{x}^T\mathbf{f} \\ &= \frac{1}{2}\mathbf{v}^T\mathbf{A}^T\mathbf{C}\mathbf{A}\mathbf{v} + \mathbf{v}^T\left(\mathbf{A}^T\mathbf{C}\mathbf{A}\mathbf{x} - \mathbf{f}\right) + P(\mathbf{x}) \end{aligned}$$

Since $\mathbf{x}$ satisfies the equation $\mathbf{A}^T\mathbf{C}\mathbf{A}\mathbf{x} - \mathbf{f} = 0$, we get

$$P(\mathbf{u}) = \frac{1}{2}\mathbf{v}^T\mathbf{A}^T\mathbf{C}\mathbf{A}\mathbf{v} + P(\mathbf{x})$$

If we let $\mathbf{w} = \mathbf{A}\mathbf{v}$, then

$$\mathbf{v}^T\mathbf{A}^T\mathbf{C}\mathbf{A}\mathbf{v} = \mathbf{w}^T\mathbf{C}\mathbf{w} = c_1w_1^2 + c_2w_2^2 + c_3w_3^2 + c_4w_4^2$$

Since all the c_i's are positive this is a positive number, and we can conclude that $P(\mathbf{x})$ must be the minimum. In other words, solving the matrix equation

$$\mathbf{A}^T\mathbf{C}\mathbf{A}\mathbf{x} = \mathbf{f}$$

is the same as finding the displacements of the springs which minimize the potential energy of the system.

The code to solve this problem with 3 masses, each with a mass of 2 kg, and 4 springs, each with a spring constant of 10 N/m, is simply the following

```
% The 4 spring 3 mass sample problem
% Set the gravitational acceleration (m/s^2)
g = 9.81
% Set the three masses (kg)
m = [2 2 2]'
% Define the structural matrix
% including the floor and ceiling
A0 = [-1 1 0 0 0; 0 -1 1 0 0; 0 0 -1 1 0; 0 0 0 -1 1]

% Build the structural matrix for the actual displacements
A = A0(:,[2 3 4])
% Set the spring constants (N/m)
C = diag([10 10 10 10])
% Define the external forces
f = g*m

% Build the stiffness matrix
K = A'*C*A
% Find the displacements that minimize the potential energy
x = K\f
```

```
% Calculate the elongations
e = A*x

% Calculate the internal forces
y = C*e
% Calculate the external forces
A'*y
% Compare the previous result with f
f - A'*y
```

1.5 Wrapping up Trusses

The previous section summarized Hooke's Law and its application to determining the potential energy of a spring mass system. The same analysis applies to trusses. The only difference is that the structural matrix, $\mathbf{A}$, is more complicated because it also reflects the two dimensional geometry of a truss. However, $\mathbf{u}$ is the vector of possible displacements and $\mathbf{e}$ is the vector of elongations. These two quantities are related by the structural matrix

$$\mathbf{e} = \mathbf{A}\mathbf{u}$$

The internal forces acting on the bars are then given by

$$\mathbf{y} = \mathbf{C}\mathbf{e}$$

where $\mathbf{C}$ is the diagonal matrix of material constants for the bars. Finally, the external forces, $\mathbf{f}$, (the loads on the nodes) must balance the internal forces so we have

$$\mathbf{f} = \mathbf{A}^T\mathbf{y}$$

The work to restore the bars to their original positions is $-\mathbf{u}^T\mathbf{f}$ and the potential energy stored in the elongated bars is $(1/2)\mathbf{e}^T\mathbf{C}\mathbf{e}$. So in equilibrium, the truss will minimize the total potential energy

$$P(\mathbf{u}) = \frac{1}{2}\mathbf{u}^T\mathbf{A}^T\mathbf{C}\mathbf{A}\mathbf{u} - \mathbf{u}^T\mathbf{f} \tag{8}$$

and as we saw before, this function is minimized when $\mathbf{u}$ satisfies the matrix equation

$$\mathbf{A}^T\mathbf{C}\mathbf{A}\mathbf{u} = \mathbf{f} \tag{9}$$

1.6 Project Assignment

1. Sketch a planar truss diagram for the near side of the Truss Bridge in the picture.
2. Determine appropriate dimensions for the bridge and use them to define the coordinates for your truss.
3. Determine appropriate materials for the bars of your truss and determine their approximate material constants.
4. Prepare an input file that describes the truss.
5. Modify the buildTruss program so that it solves for the displacements given the applied load, calculates and prints out the internal force on each bar, and redraws the truss diagram so that the bars are color coded. The bars which are under compression should be drawn in blue, the bars which are under tension should be drawn in red, and any bar for which the internal force is less than 0.001 of the total applied load is drawn in green.
6. Consider an object crossing the bridge. First decide on whether you want to consider a human, a horse, or a car, and pick an appropriate mass. Then decide on three positions for the mass and determine appropriate applied loads. Finally analyze each of these three situations using your program.
7. Turn in a report which includes discussions of all the decisions made above, the color coded diagrams, the results of your analysis, and any relevant conclusions.