

---

```

% Minjun Park, CAAM 210, 10/26/15
%
% Description: This code graphs undeformed and deformed bridges with
given
% number of fibers and nodes. It also uses Spy command to track the
pattern
% of the matrix
%
% Usage: input the number the Ea, W, nos (see desription below), and
then
% it will give the two graphs(one: loaded bridge, two: spy command)
%
function bridgedriver
close all %close everything before it begins.
Ea=1; %Set thickness (product of Youngs modulus and cross sectional
W=0.05; %Weight of the car
nos=5; %number of stages
bridge(Ea,W,nos) %call bridge
end
%{this function takes inputs and returns}

function bridge(Ea,W,nos)
nos = nos-2; % initial stages -2
n= 2*nos + 2; %number of nodes
m= 5*nos + 5; %number of fibers
s = 1/sqrt(2); % using diagonal rule

%initializing sizes
A = zeros(m, 2*n);
xc = zeros(m, 2);
yc = zeros(m, 2);
length = ones(m, 1);
%Creating the undeformed bridge

%first two fibers
xc(1, :) = [0 1];
yc(1, :) = [0 0];
xc(2, :) = [0 1];
yc(2, :) = [0 1];
A(1, 1) = 1;
A(2, [3 4]) = [s s];
length(2) = 1/s;

%middle fibers
for i = 0:nos-1 %start from 0 to nos -1
r = 5*i;
c = 4*i; % for convenience
xc(3+r,:) = [1+i 1+i]; % 3rd fibers and the rest
yc(3+r,:) = [0 1];
xc(4+r,:) = [1+i 2+i];
yc(4+r,:) = [1 0];
xc(5+r,:) = [1+i 2+i];

```

---

---

```

yc(5+r,:) = [1 1];
xc(6+r,:) = [1+i 2+i];
yc(6+r,:) = [0 1];
xc(7+r,:) = [1+i 2+i];
yc(7+r,:) = [0 0];
A(3+r, [2+c 4+c]) = [-1 1]; % these gives A structure
A(4+r, [3+c 4+c 5+c 6+c]) = [-s s s -s];
A(5+r, [3+c 7+c]) = [-1 1];
A(6+r, [1+c 2+c 7+c 8+c]) = [-s -s s s];
A(7+r, [1+c 5+c]) = [-1 1];
length(4+r) = 1/s;
length(6+r) = 1/s;
end
%right side of bridge
r = 5*nos + 3;
c = 4*nos;
xc(r,:) = [1+nos 1+nos];
yc(r,:) = [0 1];
xc(1+r,:) = [1+nos 2+nos];
yc(1+r,:) = [1 0];
xc(2+r,:) = [1+nos 2+nos];
yc(2+r,:) = [0 0];
A(r, [2+c 4+c]) = [-1 1];
A(1+r, [3+c 4+c]) = [-s s];
A(2+r, 1+c) = -1;
length(1+r) = 1/s;
hold on
line(xc',yc') %initial undeformed bridge
force = zeros(size(A,2),1); % set force

%move on to deformed bridge
for i = 0:nos
force(2+4*i) = -1*W; % for related to the weight
K = diag(Ea./length); % Hook's constant
stiffness = A'*K*A; % calculating stiffness
displacements = stiffness\force; % calculates displacement related to
force
X = displacements(1:2:end); % x direction displacements
Y = displacements(2:2:end); % y direction displacements
%Preallocates the deformed coordinates.
dx = zeros(size(xc));
dy = zeros(size(yc));
%Deform the left side of the bridge
dx(1,:) = xc(1,:) + [0 X(1)];
dx(2,:) = xc(2,:) + [0 X(2)];
dy(1,:) = yc(1,:) + [0 Y(1)];
dy(2,:) = yc(2,:) + [0 Y(2)];

% middle fibers
for k = 0:nos-1
r = 5*k;
dx(3+r,:) = xc(3+r,:) + [X(2*k+1) X(2*k+2)];
dy(3+r,:) = yc(3+r,:) + [Y(2*k+1) Y(2*k+2)];
dx(4+r,:) = xc(4+r,:) + [X(2*k+2) X(2*k+3)];

```

---

---

```

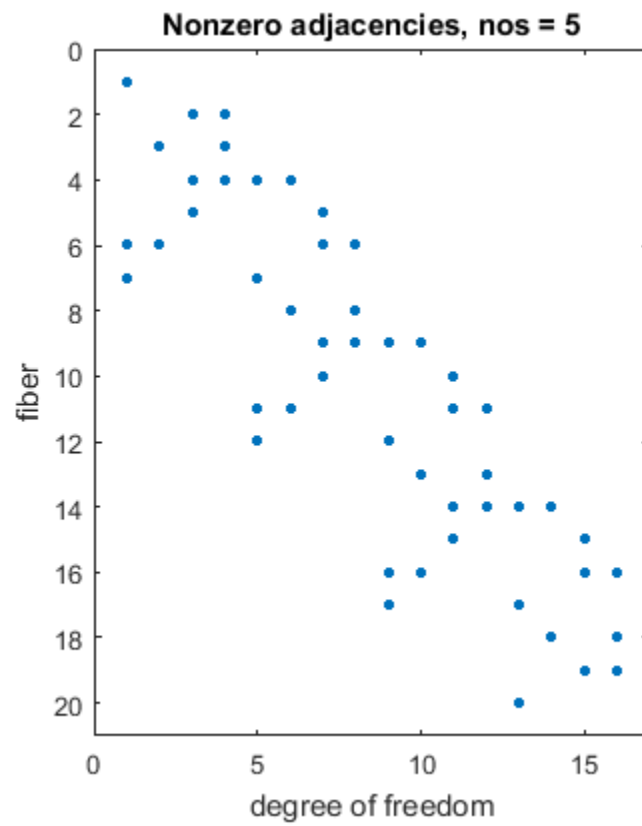
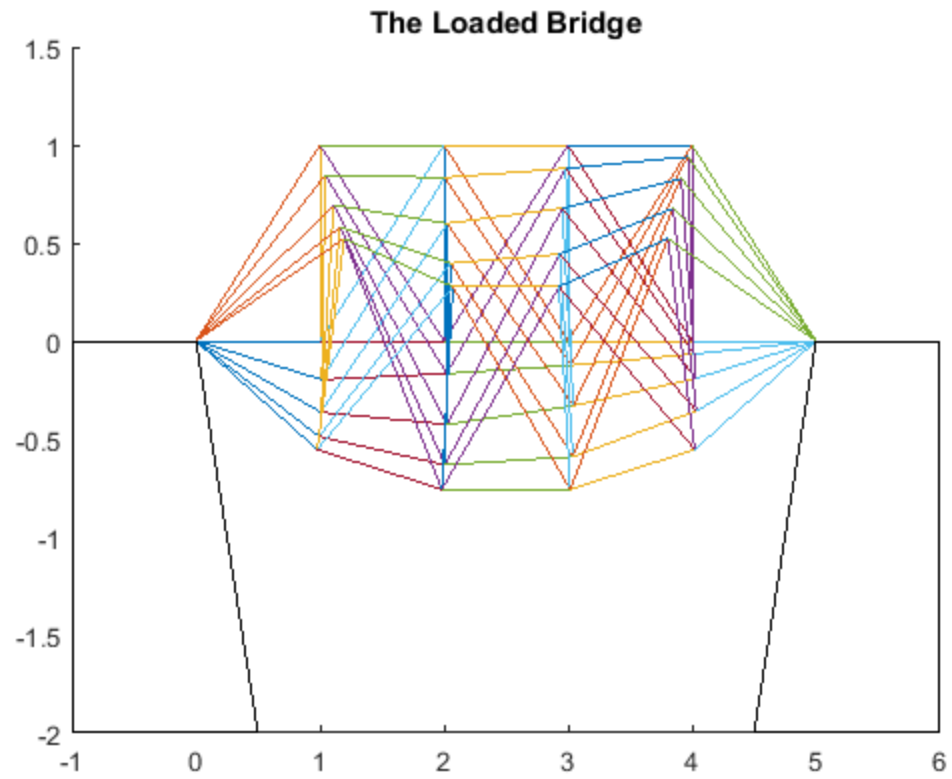
dy(4+r,:) = yc(4+r,:) + [Y(2*k+2) Y(2*k+3)];
dx(5+r,:) = xc(5+r,:) + [X(2*k+2) X(2*k+4)];
dy(5+r,:) = yc(5+r,:) + [Y(2*k+2) Y(2*k+4)];
dx(6+r,:) = xc(6+r,:) + [X(2*k+1) X(2*k+4)];
dy(6+r,:) = yc(6+r,:) + [Y(2*k+1) Y(2*k+4)];
dx(7+r,:) = xc(7+r,:) + [X(2*k+1) X(2*k+3)];
dy(7+r,:) = yc(7+r,:) + [Y(2*k+1) Y(2*k+3)];
end

% last three fibers
dx(m-2,:) = xc(m-2,:) + [X(2*nos+1) X(2*nos+2)];
dx(m-1,:) = xc(m-1,:) + [X(2*nos+2) 0];
dx(m,:) = xc(m,:) + [X(2*nos+1) 0];
dy(m-2,:) = yc(m-2,:) + [Y(2*nos+1) Y(2*nos+2)];
dy(m-1,:) = yc(m-1,:) + [Y(2*nos+2) 0];
dy(m,:) = yc(m,:) + [Y(2*nos+1) 0];
line(dx',dy'); %Graph the edges with line.

hold on
Xl = [-1 -1 0 .5];
Yl = [-2 0 0 -2];
N = nos + 2;
Xr = [N N+1 N+1 N-.5];
Yr = [0 0 -2 -2];
fill(Xl, Yl, 'w')
fill(Xr, Yr, 'w')
axis([-1 N+1 -2 1.5])
title('The Loaded Bridge')
hold off
end
hold off
figure %creating new figure
spy(A); %plot the spy graph.
title(strcat('Nonzero adjacencies, nos =', {' '}, num2str(nos
+2))); %title
xlabel('degree of freedom'); % x label
ylabel('fiber'); % y label
end

```

---



---

*Published with MATLAB® R2015a*