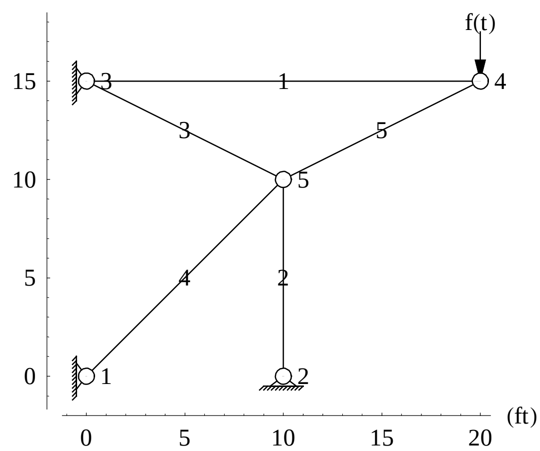
Computer Implementation 8.4 ***(Matlab)*** *Transient analysis of a plane truss (p. 573)*

Consider solution of the plane truss structure All members have the same cross-sectional area and are of the same material, $\rho = 490\,\mathrm{lb}/\mathrm{ft}^3$, $E = 30 \times 10^6\,\mathrm{lb}/\mathrm{in}^2$ and $A = 1.25\,\mathrm{in}^2$. The truss supports a rotating machine weighing 2000 lbs at its tip. Due to unbalance in the machine a harmonic force $P = 100\cos(7\pi t)\,\mathrm{lb}$ is exerted on the truss.



The weight of the machine is an added mass at node 4. This is defined as ma in the following line and is added directly to the global mass matrix corresponding to degrees of freedom 7 and 8 associated with node 4. In defining the global load vector, a load of 1 unit is applied at degree of freedom 8. This global load vector is multiplied by $f(t) = 100\cos(7\pi t)$ before solving the differential equations of motion. The other procedure for generating the global mass, stiffness, and the load vector is exactly the same as that for the static analysis used in the previous chapters. Using in-lb units the calculations are as follows.

MatlabFiles\Chap8\TransientTrussEx.m

```
% Transient analysis of a plane truss
global Mf Kf Rf
g = 386.4; e = 30*10^6; A = 1.25;
rho = (490/(12^3))/g; ma = 2000/g;
nodes = 12 * [0, 0; 10, 0; 0, 15; 20, 15; 10, 10];
conn = [3, 4; 2, 5; 5, 3; 1, 5; 5, 4];
elems = size(conn,1);
lmm=[];
for i=1:elems
    lmm = [lmm; [2*conn(i,1)-1, 2*conn(i,1),2*conn(i,2)-1, 2*conn(i,2)]];
```

```
end
debc = [1:6]; ebcVals=zeros(length(debc),1);
dof=2*size(nodes,1);
M=zeros(dof); K=zeros(dof);
R = zeros(dof,1); R(8)=1;
% Add nodal masses to the global M matrix.

M(7,7)=ma; M(8,8) = ma;
% Generate equations for each element and assemble them.
for i=1:elems
    con = conn(i,:);
    lm = lmm(i,:);
    [m, k] = TransientPlaneTrussElement(e, A, rho, nodes(con,:));
    M(lm, lm) = M(lm, lm) + m;
    K(lm, lm) = K(lm, lm) + k;
end

% Adjust for essential boundary conditions
dof = length(R);
df = setdiff(1:dof, debc);
Mf = M(df, df);
Kf = K(df, df);
Rf = R(df) - K(df, debc)*ebcVals;

% Setup and solve the resulting first order differential equations
u0 = zeros(length(Mf),1);
v0 = zeros(length(Mf),1);
[t,d] = ode23('TrussODE',[0,1],[u0; v0]);
plot(t,d(:,2)); xlabel('time'); ylabel('Disp');
title('Vertical displacement at node 4');
```

MatlabFiles\Chap8\TrussODE.m

```
function ddot = TrussODE(t, d)
% ddot = TrussODE(t, d)
% function to set up equations for a transient truss problem
global Mf Kf Rf
ft = 100*cos(7*pi*t);
n=length(d);
u = d(1:n/2);
v = d(n/2+1:n);
vdot = inv(Mf)*(Rf*ft - Kf*u);
udot = v;
% format short g
% soln=[t, ft, u(1), udot(1), vdot(1)]
ddot = [udot; vdot];
```

Executing the script file TransientTrussEx a plot of the vertical displacement at node 4. The script file can be modified to see other quantities of interest.

```
>> TransientTrussEx
```