IMPORTANT NOTE: Please use these appendices as a companion to my technical paper published in the MMVR proceedings. If you follow the MMVR paper and use these notes in tandem, it will be fairly easy to construct the finite element model that I used in my simulations.

Ref: Basdogan, C., 2001, "Real-time Simulation of Dynamically Deformable Finite Element Models Using Modal Analysis and Spectral Lanczos Decomposition Methods", to appear in the *Proceedings of the Medicine Meets Virtual Reality (MMVR'2001) Conference, Jan 24-27, Irvine, CA.*

Cagatay Basdogan, Ph.D.
Virtual Environments Laboratory (http://vel.jpl.nasa.gov)
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive, M/S: 126-205A
Pasadena, CA, 91109
e-mail: Cagatay.Basdogan@jpl.nasa.gov
http://eis.jpl.nasa.gov/~basdogan
tel: (818)-354-8892
fax: (818)-393-3003

Appendix A: Finite-Element-Modeling

1. Construction of local element stiffness matrix:

If the thickness of the triangular element is assumed to be constant, the inplane stiffness matrix ($\left[k^e{}_m\right]$) can be separated into two parts; one the matrix due to normal stresses ($\left[k^e{}_n\right]$), and the other due to shear stresses ($\left[k^e{}_s\right]$) (Rao, 1989).

$$\left[k^e{}_m\right] = \left[k^e{}_n\right] + \left[k^e{}_s\right]$$

The components of these matrices can be given in the explicit form as

$$\left[k^e{}_n\right] = \frac{Et}{4A(1-v^2)}\begin{bmatrix} y_{32}^2 & & & & & \\ -vy_{32}x_{32} & x_{32}^2 & & \mathit{Symmetric} & & \\ -y_{32}y_{31} & vx_{32}y_{31} & y_{31}^2 & & & \\ vy_{32}x_{31} & -x_{32}x_{31} & -vy_{31}x_{31} & x_{31}^2 & & \\ y_{32}y_{21} & -vx_{32}y_{21} & -y_{31}y_{21} & vx_{31}y_{21} & y_{21}^2 & \\ -vy_{32}x_{21} & x_{32}x_{21} & vy_{31}x_{21} & -x_{31}x_{21} & -vy_{21}x_{21} & x_{21}^2 \end{bmatrix}$$

$$\left[k^e{}_s\right] = \frac{Et}{8A(1+v)}\begin{bmatrix} x_{32}^2 & & & & & \\ -x_{32}y_{32} & y_{32}^2 & & \mathit{Symmetric} & & \\ -x_{32}x_{31} & y_{32}x_{31} & x_{31}^2 & & & \\ x_{32}y_{31} & -y_{32}y_{31} & -x_{31}y_{31} & y_{31}^2 & & \\ x_{32}x_{21} & -y_{32}x_{21} & -x_{31}x_{21} & y_{31}x_{21} & x_{21}^2 & \\ -x_{32}y_{21} & y_{32}y_{21} & x_{31}y_{21} & -y_{31}y_{21} & -x_{21}y_{21} & y_{21}^2 \end{bmatrix}$$

where,

$$x_{ij} = x_i - x_j$$
$$y_{ij} = y_i - y_j$$

and t is the thickness of the element, E is the Young's modulus, A is the area of the triangle and, v is the Poisson's ratio. Young's modulus determines the flexibility of the object, whereas the Poisson's ratio defines the relation between lateral strain and strain along the direction of loading (Hence, smaller values of Poisson's ratio indicates that the object will not stretch much under tension.

Similarly, the bending stiffness matrix ($[k^e{}_b]$) can be expressed as the multiplication of two matrices (Rao, 1989);

$$[k^e{}_b] = [N^{-1}]^T [A][N^{-1}]$$

where,

$$[N] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & y_2 & 0 & 0 & y_2^2 & 0 & 0 & y_2^3 \\ 0 & 0 & 1 & 0 & 0 & 2y_2 & 0 & 0 & 3y_2^2 \\ 0 & -1 & 0 & 0 & -y_2 & 0 & 0 & -y_2^2 & 0 \\ 1 & x_3 & y_3 & x_3^2 & x_3y_3 & y_3^2 & x_3^3 & (x_3^2y_3 + x_3y_3^2) & y_3^3 \\ 0 & 0 & 1 & 0 & x_3 & 2y_3 & 0 & (2x_3y_3 + x_3^2) & 3y_3^2 \\ 0 & -1 & 0 & -2x_3 & -y_3 & 0 & -3x_3^2 & -(y_3^2 + 2x_3y_3) & 0 \end{bmatrix}$$

and A is a symmetric matrix ($A[i][j] = A[j][i], \quad i = 1,...,9;\ j = 1,...,9$) with elements described as

$$A[1][1] = 0$$
$$A[2][1] = A[2][2] = 0$$
$$A[3][1] = A[3][2] = A[3][3] = 0$$
$$A[4][1] = A[4][2] = A[4][3] = 0$$
$$A[4][4] = 2x_3y_2$$
$$A[5][1] = A[5][2] = A[5][3] = A[5][4] = 0$$
$$A[5][5] = x_3y_2(1-v)$$
$$A[6][1] = A[6][2] = A[6][3] = A[6][5] = 0$$
$$A[6][4] = 2vx_3y_2$$
$$A[6][6] = 2x_3y_2$$
$$A[7][1] = A[7][2] = A[7][3] = A[7][5] = 0$$
$$A[7][4] = 2x_3^2y_2$$
$$A[7][6] = 2vx_3^2y_2$$
$$A[7][7] = 3x_3^3y_2$$
$$A[8][1] = A[8][2] = A[8][3] = 0$$
$$A[8][4] = \frac{2}{3}x_3y_2(vx_3 + (y_2 + y_3))$$
$$A[8][5] = \frac{2}{3}(1-v)x_3y_2(x_3y_2 + (y_2 + y_3))$$
$$A[8][6] = \frac{2}{3}x_3y_2(x_3y_2 + v(y_2 + y_3))$$

$$A[8][7] = \frac{1}{2} x_3^{\,2} y_2 \left\{ 2vx_3 y_2 + (y_2 + 2y_3) \right\}$$

$$A[8][8] = \frac{1}{3} x_3 y_2 \left\{ (3-2v)(x_3^{\,2} y_2) + (2-v)(x_3 y_2)(y_2 + 2y_3) + (3-2v)(y_2^{\,2} + y_2 y_3 + y_3^{\,2}) \right\}$$

$$A[9][1] = A[9][2] = A[9][3] = A[9][5] = 0$$

$$A[9][4] = 2vx_3 y_2 (y_2 + y_3)$$

$$A[9][6] = 2x_3 y_2 (y_2 + y_3)$$

$$A[9][7] = \frac{3}{2} (vx_3^{\,2} y_2)(y_2 + 2y_3)$$

$$A[9][8] = \frac{1}{2} (x_3^{\,2} y_2)(y_2 + 2y_3) + vx_3 y_2 (y_2^{\,2} + y_2 y_3 + y_3^{\,2})$$

$$A[9][9] = 3x_3 y_2 (y_2^{\,2} + y_2 y_3 + y_3^{\,2})$$

2. Computer Implementation of Assembly of Overall Stiffness Matrix (Adapted from Rao, 1989)

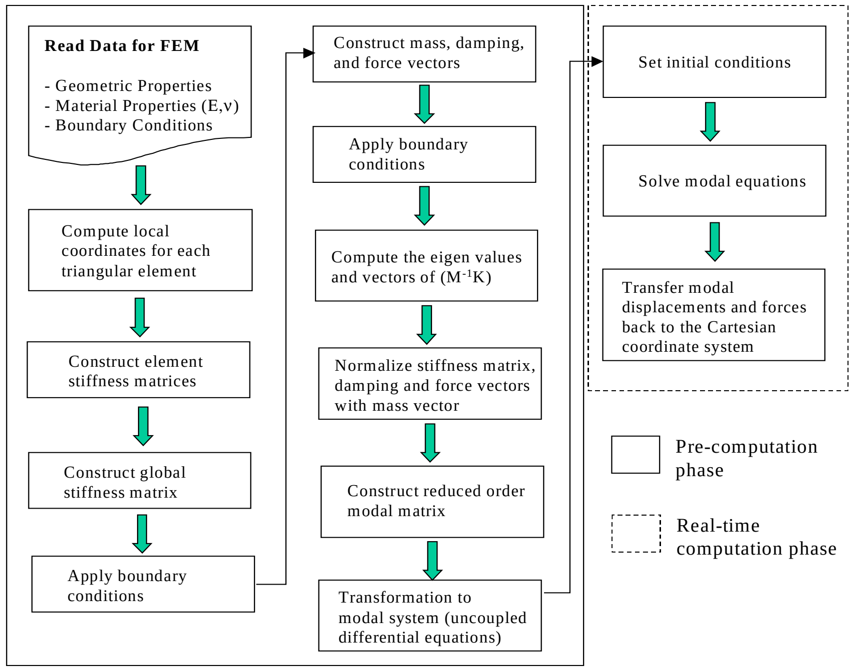
If K is stored as a symmetric square matrix

```
initialize K (set the elements of the K matrix to zero)
for i = 1: number of triangles
        for j = 1:3
        ij = the index of the j-th node of the i-th triangle
            for k = 1:3
            ik = the index of the k-th node of the i-th triangle
            K[ij][ik] = K[ij][ik] + K^i [j][k]
            end
        end
end
```

K can also be stored as a banded matrix to save from the memory space. If K is going to be stored as a banded matrix, we need to know the band-width of the matrix to construct it. The band-width is equal to the maximum difference between the numbered degrees of freedom of the structure plus one.

```
initialize K (set the elements of the K matrix to zero)
for i = 1: number of triangles
        for j = 1:3
        ij = the index of the j-th node of the i-th triangle
        for k = 1:3
                ik = the index of the k-th node of the i-th triangle
                ikm = ik – ij + 1
                if (ikm < 1)
                        continue;
                else
                        K[ij][ikm] = K[ij][ikm] + K^i [j][k]
           end
        end
    end
```

3. Flow chart of the finite element computations



APPENDIX B: Numerical Integration Using Newmark Procedure

Note that

- the damping matrix assumed to be linearly proportional with mass matrix in the original equations, $B = \alpha M = diag(\alpha m_i)$
- the steps for "Direct Numerical Integration of Original System" (DNIOS) and the "Numerical Integration of Reduced Modal System" (NIRMS) are slightly different from each other. These differences were clearly marked in the algorithm below.

1. Pre-computation phase:

a) Construct stiffness and mass matrices: K and M= $diag(m_i)$

b) Define the initial conditions:

$U_0, \dot{U}_0, \ddot{U}_0$ (DNIOS)

$X_0 = \Phi U_0, X = \Phi \dot{U}_0, X = \Phi \ddot{U}_0$ (NIRMS)

c) Calculate the constants of "Newmark" integration

$$a_0 = \frac{4 + 2\alpha \Delta t}{\Delta t^2} \qquad a1 = a0 \qquad a2 = \frac{4}{\Delta t} + \alpha \qquad a_3 = 1$$

$$a_4 = \frac{4}{\Delta t^2} \qquad a_5 = -a_4 \qquad a_6 = a_5(\Delta t) \qquad a_7 = -1$$

$$a_8 = \frac{\Delta t}{2} \qquad a_9 = \frac{\Delta t^2}{4} \qquad a_{10} = a_9$$

d) Construct the modified stiffness matrix:

$\hat{K} = K + a_0 M$ (DNIOS)

$\hat{K} = K + a_0 I = diag(\omega_i^2 + a_0)$ (NIRMS)

e) Decompose the modified stiffness matrix using Cholesky decomposition (see Numerical Recipies, Press et al.) (Only for DNIOS)

f) Compute the modal force vector (Only for NIRMS)

$^{t+\Delta t} f^{R} = (\Phi^{R})^{T} \ ^{t+\Delta t}F$

2. Real-time simulation phase (Computational steps for direct integration of equilibrium equations)

Step 1: Construct modified force vector

$\hat{F} = {}^{t+\Delta t}F + M(a_1 {}^{t}U + a_2 {}^{t}\dot{U} + a_3 {}^{t}\ddot{U})$ (DNIOS)

$\hat{f}^{R} = {}^{t+\Delta t}f^{R} + a_1 {}^{t}X^{R} + a_2 {}^{t}\dot{X}^{R} + a_3 {}^{t}\ddot{X}^{R}$ (NIRMS)

Step 2: Compute modified displacements using

$\hat{K}\hat{U} = \hat{F}$ "Cholesky backsubstitution" (see Press et al, 1993) (DNIOS)

$\hat{X}^{R} = \hat{f}^{R} / \hat{K}$ (NIRMS)

Step 3: Compute accelerations, velocities, and displacements at time $t + \Delta t$

$$
\begin{aligned}
{}^{t+\Delta t}\ddot{U} &= a_4 \hat{U} + a_5 {}^{t}U + a_6 {}^{t}\dot{U} + a_7 {}^{t}\ddot{U} \\
{}^{t+\Delta t}\dot{U} &= {}^{t}\dot{U} + a_8 ({}^{t}\ddot{U} + {}^{t+\Delta t}\ddot{U}) \\
{}^{t+\Delta t}U &= {}^{t}U + \Delta t \ {}^{t}\dot{U} + a_9 {}^{t}\ddot{U} + a_{10} {}^{t+\Delta t}\ddot{U}
\end{aligned}
\quad \text{(DNIOS)}
$$

$$
\begin{aligned}
{}^{t+\Delta t}X^{R} &= a_4 \hat{X}^{R} + a_5 {}^{t}X^{R} + a_6 {}^{t}\dot{X}^{R} + a_7 {}^{t}\ddot{X}^{R} \\
{}^{t+\Delta t}\dot{X}^{R} &= {}^{t}\dot{X}^{R} + a_8 ({}^{t}\ddot{X}^{R} + {}^{t+\Delta t}\ddot{X}^{R}) \\
{}^{t+\Delta t}\ddot{X}^{R} &= {}^{t}X^{R} + \Delta t \ {}^{t}\dot{X}^{R} + a_9 {}^{t}\ddot{X}^{R} + a_{10} {}^{t+\Delta t}\ddot{X}^{R} \\
{}^{t+\Delta t}U &= (\Phi^{R}) \ {}^{t+\Delta t}X^{R}
\end{aligned}
\quad \text{(NIRMS)}
$$

Step 4: Compute the forces that will be reflected to the user

${}^{t+\Delta t}F = M \ {}^{t+\Delta t}\ddot{U} + B \ {}^{t+\Delta t}\dot{U} + K \ {}^{t+\Delta t}U$ (DNIOS)

$$
\begin{cases}
{}^{t+\Delta t}f^{R} = {}^{t+\Delta t}\ddot{X}^{R} + \alpha \ {}^{t+\Delta t}\dot{X}^{R} + \Omega^2 \ {}^{t+\Delta t}X^{R} \\
{}^{t+\Delta t}F = (\Phi^{R}) \ {}^{t+\Delta t}f^{R}
\end{cases}
\quad \text{where, } \Omega^2 = diag(\omega_i^2) \quad \text{(NIRMS)}
$$

Step 5: Goto Step 1.