Stiffness Matrices of Carbon Nanotube Structures

by

Kirk J. Samaroo

Submitted to the Department of Mechanical Engineering in Partial Fulfillment of the Requirements for the Degree of

Bachelor of Science

at the

Massachusetts Institute of Technology

June 2005

Signature of Author _____
Department of Mechanical Engineering
May 13, 2005

Certified by _____
David M. Parks
Professor of Mechanical Engineering
Thesis Supervisor

Accepted by _____
Ernest G. Cravalho
Chairman, Undergraduate Thesis Committee

Stiffness Matrices of Carbon Nanotube Structures

by

Kirk J. Samaroo

Submitted to the Department of Mechanical
Engineering on May 6, 2005 in Partial Fulfillment of
the Requirements for the Degree of Bachelor of Science
in Mechanical Engineering

ABSTRACT

An analytical modeling study was done to determine the stiffness matrices of the lattice structure of graphene, the planar building block of carbon nanotubes. Through continuum linear elastic analysis and a displacement-based finite element method, the global in-plane stiffness matrix for an arbitrary carbon atom of the lattice was found.

The matrix provides the atomic level forces induced on members of the lattice structures due to local atomic displacements.

Thesis Supervisor: David M. Parks

Title: Professor of Mechanical Engineering

Introduction

Carbon nanotubes are composed entirely of covalently bonded carbon atoms. These atoms are arranged in identical hexagonal carbon rings which are bonded to each other forming a lattice structure. A single layer of this lattice structure, as shown in Figure 1, is known as a graphene sheet and can be thought of as an unraveled single wall carbon nanotube. When these sheets are stacked on other such sheets, the graphite structure of carbon is formed. The discrete elements of the graphene structure provide convenience for analysis by a finite element method.

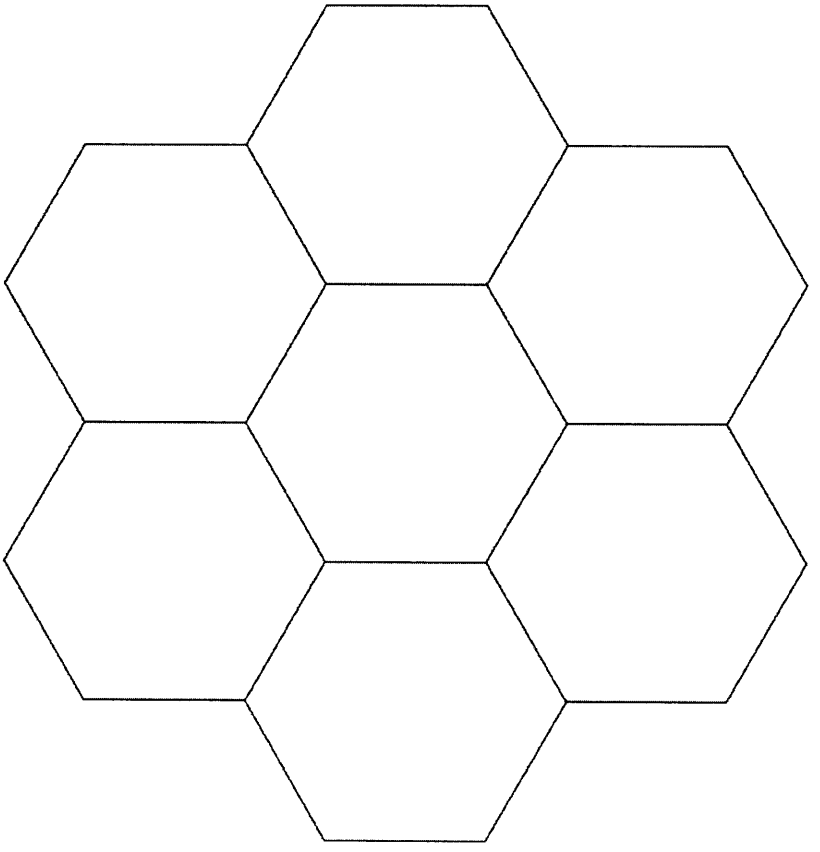

Figure 1: Above is a model of an unraveled partial lattice structure that forms the carbon nanotube. The equilateral hexagons represent the covalently bonded carbon rings, with each vertex being the location of a carbon atom.

Conventionally, a continuous system is broken down into discrete elements that are sufficiently small to simplify calculations. When the resulting calculations from the discretized elements are superimposed, the system approaches the continuum from which it was based. In the case of the carbon lattice as described, the system is essentially a finite system with discrete nodes, yet its mechanical behavior has been found to be consistent with continuum theory. This counterintuitive behavior should be noted, and may possibly be explained through mechanical analysis of the respective structures. Experimentation on carbon nanotubes is difficult due to their size- on the order of nanometers in diameter and microns in length. For more general infotmation and detail on the structure of the nanotubes, see Harris's text listed in the references.

In this paper, a stiffness matrix of a single layered flat sheet of the carbon nanotube structure will be found through finite element based continuum analysis. The stiffness matrix of any structure can be found through two fundamental matrices: an elasticity matrix that describes the material properties of a material point and its stress-strain relationship, and a strain-displacement matrix that describes the induced strain in an element from distortions in its shape.

The Elasticity Matrix

The elasticity matrix is defined by material properties of the substance it describes. Through continuum shell theory analysis and a finite element approach confirmed by experimentation, the relevant material properties of the carbon nanotubes have been found. For further detail into how this was accomplished, see the works of Pantano et. al. listed in the references. These analyses provide the modulus of elasticity, Poisson's ratio, and effective thickness of the nanotube sheet and will be visited upon later in the text.

The elasticity matrix **D** is directly derived from the stress-strain relationship shown in Equation 1:

$$\sigma = \mathbf{D}\varepsilon . \qquad (1)$$

The generalized Hooke's law which for isotropic elasticity further describes this relationship is given in Equations 2 and 3:

$$\varepsilon_{ij} = \frac{1}{E}\left[(1+\nu)\sigma_{ij} - \nu\delta_{ij}\sum_{k}\sigma_{kk}\right]; \qquad (2)$$

$$\sigma_{ij} = \frac{E}{1+\nu}\left[\varepsilon_{ij} + \frac{\nu}{1-2\nu}\delta_{ij}\sum_{k}\varepsilon_{kk}\right]; \qquad (3)$$

$$\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}.$$

Young's modulus of elasticity is denoted by E, Poisson's ratio by v and δ_{ij} is the Kronecker delta function.

In the case of the graphene sheet considered, Equations 2 and 3 are specified to plane stress conditions. Plane stress is observed in this case because the characteristic in-plane dimensions of the sheet are sufficiently large compared to the graphene sheet's effective thickness. The thin walled structure also means that the normal stresses through the thickness are small compared to the in plane stresses, and can be ignored. The shear stresses out of the plane are also negligible since they act through weak van der Waals bonds, compared to the in-plane covalent bonding of the carbon atoms. Applying the plane stress restrictions provides the two-dimensional stress matrix shown in Equation 4, and its strain counterpart in Equation 5.

$$\sigma = \begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \tau_{xy} \end{bmatrix} \qquad (4)$$

$$\varepsilon = \begin{bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{bmatrix} \qquad (5)$$

The relationship between shear stress and shear strain is shown in Equation 6, where G is the shear modulus.

$$\tau_{xy} = G\gamma_{xy} = \frac{E}{2(1+\nu)}\gamma_{xy} \qquad (6)$$

Manipulating Equations 1 through 6 provides the plane stress elasticity matrix in Equation 7:

$$\mathbf{D} = \frac{E}{1-\nu^2}\begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1}{2}(1-\nu) \end{bmatrix}. \qquad (7)$$

Linear Analysis: Displacement-Based Finite Element Method

Finite element analysis involves segmenting a structure to analyze parts that make the whole. The element chosen is part of a continuum much smaller than the structure in question. The carbon nanotube case is interesting because the dimension of each ring is of the same order as the diameter of the tube: the distance between each carbon atom of a sheet is approximately 0.14 nm, whereas the diameter of a tube is approximately 1 nm. Finding the stiffness matrix of the carbon sheets that make these tubes will present some clarity and insight into why finite element analysis and continuum theory both support this unorthodox case. The theory of this analysis can be found in Chapters 4 and 5 of the text written by Bathe listed among the references.

Identifying and Mapping the Element

Each carbon sheet is made up of identical hexagonal carbon rings as shown below in Figure 2. These rings are composed of six carbon atoms covalently bonded to each other. Using finite element analysis on these rings isolates each atom as a node.

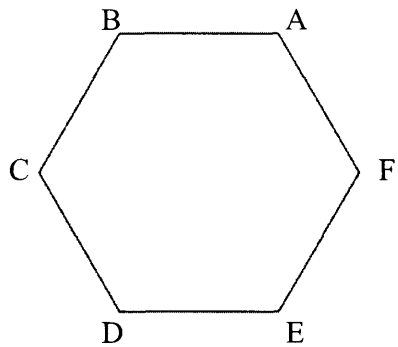

Figure 2: Above is a carbon ring modeled as an equilateral hexagon with each of the carbon atoms represented by lettered nodes.

The area covered by carbon rings can be broken down into isosceles trapezoidal elements as shown in Figure 3, allowing for the structure to be mapped to axes r and s by Equations 8 and 9. This makes it easier to analyze the stiffness elements among the four nodal (atomic) components of the trapezoid.

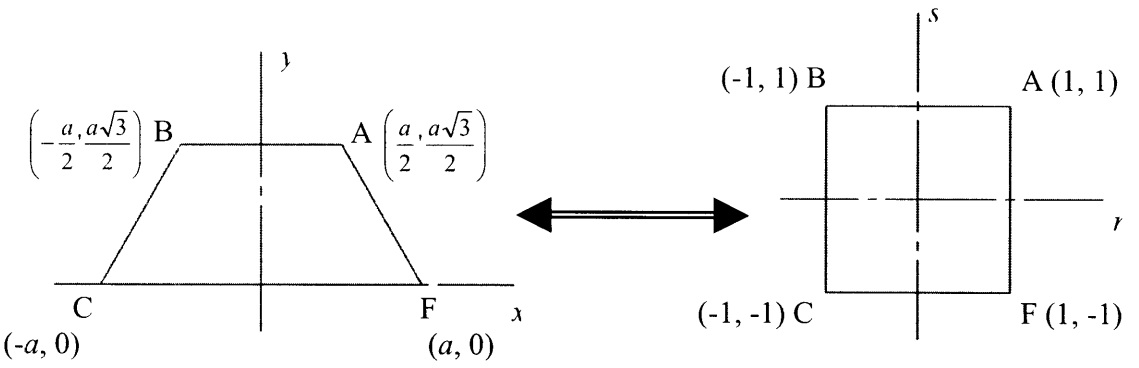

Figure 3: Above is an isosceles trapezoid from the hexagonal carbon ring plotted in *x*-*y* space, along with its corresponding mapped structure in an alternate *r*-*s* space. The corresponding coordinates are listed next to the lettered nodes of the quadrilaterals.

The mapping of Figure 3 is mathematically defined by Equations 8 and 9:

$$x=\frac{1}{4}(1+r)(1+s)x_A+\frac{1}{4}(1-r)(1+s)x_B+\frac{1}{4}(1-r)(1-s)x_C+\frac{1}{4}(1+r)(1-s)x_F; \qquad (8)$$

$$y=\frac{1}{4}(1+r)(1+s)y_A+\frac{1}{4}(1-r)(1+s)y_B+\frac{1}{4}(1-r)(1-s)y_C+\frac{1}{4}(1+r)(1-s)y_F. \qquad (9)$$

Equations 8 and 9 describe the change of coordinates where x_i and y_i are the x and y components of node i, respectively. The values of each node's components can be found through simple geometry based on the isosceles trapezoidal element:

$$\left(x_A = \frac{a}{2},\ y_A = \frac{a\sqrt{3}}{2}\right)$$

$$\left(x_B = -\frac{a}{2},\ y_B = \frac{a\sqrt{3}}{2}\right)$$

$$\left(x_C = -a,\ y_C = 0\right)$$

$$\left(x_F = a,\ y_C = 0\right).$$

All of these nodal coordinates are dependent on the distance between each node (the length of a side of the given hexagon), defined as length a. These nodes are mapped to their r_i and s_i counterparts given below:

$$\left(r_A = 1,\ s_A = 1\right)$$

$$\left(r_B = -1,\ s_B = 1\right)$$

$$\left(r_C = -1,\ s_C = -1\right)$$

$$\left(r_F = 1,\ s_F = -1\right).$$

The relationship between the gradient operators in the two planar spaces is described by Equation 10 below, which can be found using Equations 8 and 9 along with the nodal positions provided.

$$\begin{bmatrix} \frac{\partial}{\partial r} \\ \frac{\partial}{\partial s} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial y}{\partial r} \\ \frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} \qquad (10)$$

In matrix notation, this system becomes Equation 11:

$$\frac{\partial}{\partial \mathbf{r}} = \mathbf{J}\frac{\partial}{\partial \mathbf{x}} \qquad (11)$$

where the Jacobian operator **J** is given by

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial y}{\partial r} \\ \frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} \end{bmatrix} = \frac{a}{4}\begin{bmatrix} 3-s & 0 \\ -r & \sqrt{3} \end{bmatrix}. \qquad (12)$$

The Jacobian matrix will help convert systems of equations in *x-y* space to *r-s* space Conversely, to convert equations in *r-s* space to *x-y* space the inverse Jacobian, $\mathbf{J}^{-1}$, is used:

$$\frac{\partial}{\partial \mathbf{x}} = \mathbf{J}^{-1}\frac{\partial}{\partial \mathbf{r}} \qquad (13)$$

$$\mathbf{J}^{-1} = \begin{bmatrix} \frac{\partial r}{\partial x} & \frac{\partial s}{\partial x} \\ \frac{\partial r}{\partial y} & \frac{\partial s}{\partial y} \end{bmatrix} = \frac{4}{a\sqrt{3}(3-s)}\begin{bmatrix} \sqrt{3} & 0 \\ r & 3-s \end{bmatrix}. \qquad (14)$$

The Strain-Displacement Matrix

The strain-displacement matrix is needed to calculate the stiffness of the element. This relationship describes how individual nodal displacements affect the continuum strain at any point of the structure. This matrix is derived from appropriately differentiating and combining the elements of the displacement interpolation matrix, which describes how a structure's displacement field is affected by the local displacement of each node. Applying Equations 8 and 9 to interpolate the displacement components of each node provides Equations 15 and 16, where u denotes x-displacement and v is y-displacement:

$$u = \frac{1}{4}(1+r)(1+s)u_A + \frac{1}{4}(1-r)(1+s)u_B + \frac{1}{4}(1-r)(1-s)u_C + \frac{1}{4}(1+r)(1-s)u_F; \qquad (15)$$

$$v = \frac{1}{4}(1+r)(1+s)v_A + \frac{1}{4}(1-r)(1+s)v_B + \frac{1}{4}(1-r)(1-s)v_C + \frac{1}{4}(1+r)(1-s)v_F. \qquad (16)$$

Collectively this system of equations can be represented in matrix form by Equation 17:

$$\mathbf{U} = \mathbf{N}\hat{\mathbf{u}}. \qquad (17)$$

Here, $\mathbf{U}$ is a matrix containing the continuum-interpolated displacement components in the structure, and $\hat{\mathbf{u}}$ is the local displacement matrix of each node in the structure. The matrix $\mathbf{N}$ is the displacement interpolation matrix, and can be obtained from Equations 15 and 16.

The strain-displacement relationship is given in Equation 18, where ε again is the strain matrix and **B** as the strain-displacement matrix:

$$\varepsilon = \mathbf{B}\hat{\mathbf{u}}\,. \qquad (18)$$

The matrix **B** is a composite of the derivatives of the matrix **N** as described in Equation 19:

$$\varepsilon = \begin{bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial u}{\partial x} \\ \dfrac{\partial v}{\partial y} \\ \dfrac{\partial u}{\partial y} + \dfrac{\partial v}{\partial x} \end{bmatrix}. \qquad (19)$$

Equations 21 and 22 combine to form the strain matrix and the system in Equation 18 from which **B** can be obtained, where the $\hat{\mathbf{u}}$ matrix of the trapezoid *ABCF* is given in Equation 20.

$$\hat{\mathbf{u}}^{T}_{ABCF} = \begin{bmatrix} u_A & v_A & u_B & v_B & u_C & v_C & u_F & v_F \end{bmatrix} \qquad (20)$$

$$\begin{bmatrix} \dfrac{\partial u}{\partial x} \\ \dfrac{\partial u}{\partial y} \end{bmatrix} = \frac{\partial}{\partial \mathbf{x}}\begin{bmatrix} 1 & 0 \end{bmatrix}\mathbf{U} = \mathbf{J}^{-1}\begin{bmatrix} 1 & 0 \end{bmatrix}\frac{\partial}{\partial \mathbf{r}}\mathbf{N}\hat{\mathbf{u}}_{ABCF}$$

$$\begin{bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial u}{\partial y} \end{bmatrix} = \frac{1}{4}\mathbf{J}^{-1}\begin{bmatrix} 1+s & 0 & -(1+s) & 0 & -(1-s) & 0 & 1-s & 0 \\ 1+r & 0 & 1-r & 0 & -(1-r) & 0 & -(1+r) & 0 \end{bmatrix}\hat{\mathbf{u}}_{ABCF} \qquad (21)$$

$$\begin{bmatrix} \frac{\partial v}{\partial x} \\ \frac{\partial v}{\partial y} \end{bmatrix} = \frac{\partial}{\partial \mathbf{x}}[0 \quad 1]\mathbf{U} = \mathbf{J}^{-1}[0 \quad 1]\frac{\partial}{\partial \mathbf{r}}\mathbf{N}\hat{\mathbf{u}}_{ABCF}$$

$$\begin{bmatrix} \frac{\partial v}{\partial x} \\ \frac{\partial v}{\partial y} \end{bmatrix} = \frac{1}{4}\mathbf{J}^{-1}\begin{bmatrix} 0 & 1+s & 0 & -(1+s) & 0 & -(1-s) & 0 & 1-s \\ 0 & 1+r & 0 & 1-r & 0 & -(1-r) & 0 & -(1+r) \end{bmatrix}\hat{\mathbf{u}}_{ABCF} \qquad (22)$$

The transpose of the strain-displacement matrix is given in Equation 22:

$$\mathbf{B}^T = \frac{1}{a\sqrt{3}(3-s)}\begin{bmatrix} \sqrt{3}(1+s) & 0 & 4r+3-s \\ 0 & 4r+3-s & \sqrt{3}(1+s) \\ -\sqrt{3}(1+s) & 0 & -(4r+3-s) \\ 0 & -(4r+3-s) & -\sqrt{3}(1+s) \\ -\sqrt{3}(1-s) & 0 & 2r-3+s \\ 0 & 2r-3+s & -\sqrt{3}(1-s) \\ \sqrt{3}(1-s) & 0 & -(2r-3+s) \\ 0 & -(2r-3+s) & \sqrt{3}(1-s) \end{bmatrix}. \qquad (22)$$

The Stiffness Matrix for a Trapezoidal Element

Using the strain-displacement matrix and the continuum elasticity matrix, the stiffness of trapezoid *ABCF* can be determined.

$$\mathbf{F}_{ABCF} = \mathbf{K}_{ABCF}\hat{\mathbf{u}}_{ABCF}$$

$$\mathbf{F}^T_{ABCF} = \begin{bmatrix} f_A^x & f_A^y & f_B^x & f_B^y & f_C^x & f_C^y & f_F^x & f_F^y \end{bmatrix}$$

$$\mathbf{K}_{ABCF} = \int_{V_{ABCF}} \mathbf{B}^T\mathbf{D}\mathbf{B}\,dV_{ABCF} = \int_{A_{ABCF}} \mathbf{B}^T\mathbf{D}\mathbf{B}\,t dA_{ABCF} = \iint \mathbf{B}^T\mathbf{D}\mathbf{B}\,t dx dy \qquad (23)$$

Here, $\mathbf{F}_{ABCF}$ contains the nodal forces associated with nodal displacements $\hat{\mathbf{u}}_{ABCF}$, acting over the area *ABCF*. Notice that the volume of the element *ABCF* is equivalent to the effective thickness *t* times the area of the trapezoid. The volume integral of Equation 23 can be converted to the isometric *r-s* space by multiplying the integrand by the determinant of **J**.

$$\mathbf{K}_{ABCF} = \int_{-1}^{+1}\int_{-1}^{+1} \mathbf{B}^T\mathbf{D}\mathbf{B}\det\mathbf{J}\,t dr ds \qquad (24)$$

Evaluating the integrals provides the eight by eight matrix below. Notice that the stiffness matrix is symmetric. Analyses done by Pantano et. al (see references), provide a Poisson's ratio of 0.19 used in the matrix simplifications. The same analyses provide a *t* of 0.075 nm and an *E* of 4.84 TPa. These analyses were done using a numeric quadrature

to approximate the integral Equation 24. A numeric stiffness matrix and its corresponding eigenvalues were calculated and compared to the analytic result below in Appendix E. Equation 25 displays the stiffness matrix with two significant figures; in Appendices A and B, the matrix is given with four digits per entry.

$$\mathbf{K}_{ABCF} = Et \begin{bmatrix} 0.54 & 0.15 & -0.17 & -0.056 & -0.23 & -0.15 & -0.14 & 0.056 \\ 0.15 & 0.83 & 0.056 & 0.073 & -0.15 & -0.35 & -0.056 & -0.55 \\ -0.17 & 0.056 & 0.54 & -0.15 & -0.14 & -0.056 & -0.23 & 0.15 \\ -0.056 & 0.073 & -0.15 & 0.83 & 0.056 & -0.55 & 0.15 & -0.35 \\ -0.23 & -0.15 & -0.14 & 0.056 & 0.38 & 0.15 & -0.020 & -0.056 \\ -0.15 & -0.35 & -0.056 & -0.55 & 0.15 & 0.59 & 0.056 & 0.31 \\ -0.14 & -0.056 & -0.23 & 0.15 & -0.020 & 0.056 & 0.38 & -0.15 \\ 0.056 & -0.55 & 0.15 & -0.35 & -0.056 & 0.31 & -0.15 & 0.59 \end{bmatrix} \quad (25)$$

The Stiffness Matrix for a Carbon Ring

Similar procedures of linear analysis can determine the stiffness matrix of the other isosceles trapezoids that make up the hexagon, but notice that the trapezoids are of all the same shape, and any such trapezoid formed within the hexagon *ABCDEF* is a mere rotation of another. Figure 4 shows that an arbitrary trapezoid of the hexagon can be transformed to x'- y' axes rotated by a specific angle θ to mimic the trapezoid *ABCF* defined on x-y axes. The theory described in this section can be referenced in Section 2.4 of the text written by Bathe.

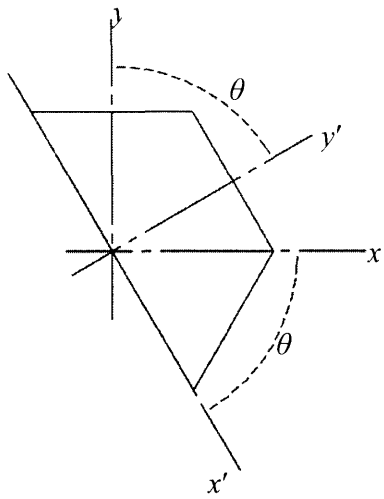

Figure 4: A trapezoidal element of the original hexagon is of the same geometry as trapezoid *ABCF* rotated in plane by an angle θ. In the trapezoid depicted (*FABE*), the counterclockwise rotation angle θ is -60 degrees.

Changing axes can easily be done through a rotation matrix. The rotation matrix in two-dimensional space for a counterclockwise angle θ is given in Equation 26 by $\mathbf{R}_\theta$.

$$\mathbf{x}' = \mathbf{R}_\theta \mathbf{x}$$

$$\mathbf{R}_\theta = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \qquad (26)$$

Since the local displacement matrix for the trapezoid *ABCF* is arranged in *x*-*y* pairs for all four nodes, each node must be rotated, making its rotation matrix an eight by eight matrix $\mathbf{Q}_\theta$. The two-dimensional rotation matrices $\mathbf{R}_\theta$ along the diagonal rotate each node.

$$\hat{\mathbf{u}}'_{ABCF} = \mathbf{Q}_\theta \hat{\mathbf{u}}_{ABCF}$$

$$\mathbf{Q}_\theta = \begin{bmatrix} \mathbf{R}_\theta & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_\theta & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{R}_\theta & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{R}_\theta \end{bmatrix} \qquad (27)$$

$$\mathbf{0} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

The stiffness matrix of trapezoid *ABCF* is equal to the stiffness matrix of the arbitrary trapezoid transformed by the rotation matrix $\mathbf{Q}_\theta$ and its transpose in the following way, with θ again being the counterclockwise angle by which the geometry of the new trapezoid is shifted.

$$\mathbf{K}_{ABCF} = \mathbf{Q}_\theta \mathbf{K}' \mathbf{Q}_\theta^T \qquad (28)$$

Transforming the stiffness matrix of trapezoid *ABCF* to the stiffness matrix of the other trapezoid can be done by taking the inverse of the relation given in Equation 28. The

inverse relation is simple to derive due to the orthogonality of the rotation matrix- the transpose of the matrix is equal to its inverse:

$$\mathbf{K}' = \mathbf{Q}_\theta^T \mathbf{K}_{ABCF} \mathbf{Q}_\theta . \quad (29)$$

As shown in Figure 5, six possible trapezoids reside in hexagon *ABCDEF*: *ABCF*, *BCDA*, *CDEB*, *DEFC*, *EFAD*, and *FABE*.

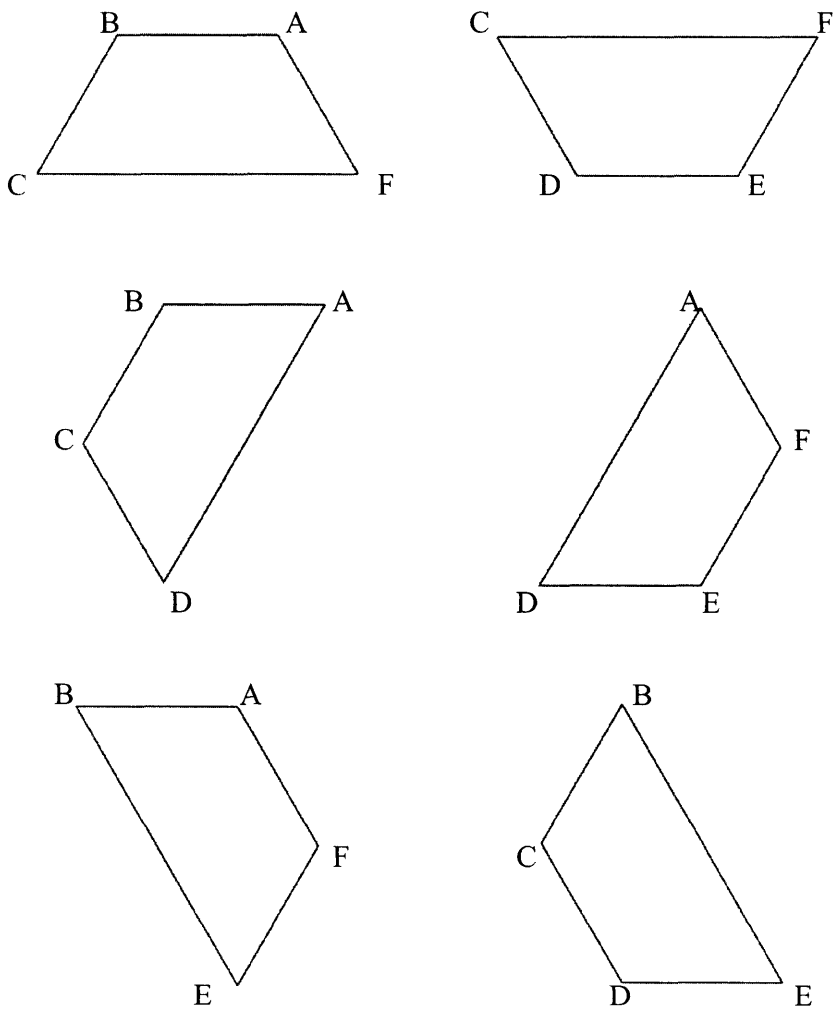

Figure 5: The six possible trapezoids formed within hexagon *ABCDEF* are shown. Each trapezoid is a rotated version of another.

A special case exists when the transformation angle θ is 180 degrees. The rotation matrix becomes minus the identity matrix, and the matrix being transformed equals the original. For example, consider trapezoid *DEFC*, which has a transformation angle of 180 degrees with respect to trapezoid *ABCF*. $\mathbf{Q}_{+180^\circ}$ becomes minus the identity matrix of rank 8. Transforming the stiffness matrix of *ABCF* by 180 degrees simply means that the matrix is multiplied by $-\mathbf{I}^{(8)}$ twice, making the stiffness matrix of *ABCF* equal to that of *DEFC*.

$$\mathbf{K}_{DEFC} = \mathbf{Q}_{+180^\circ}^{T}\mathbf{K}_{ABCF}\mathbf{Q}_{+180^\circ}$$

$$\mathbf{R}_{+180^\circ} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} = -\mathbf{I}^{(2)}$$

$$\mathbf{Q}_{+180^\circ} = -\mathbf{I}^{(8)}$$

$$\mathbf{K}_{DEFC} = \left(-\mathbf{I}^{(8)}\right)\mathbf{K}_{ABCF}\left(-\mathbf{I}^{(8)}\right) = \mathbf{K}_{ABCF} \qquad (30)$$

The same geometries that ensure that the stiffness matrices of *ABCF* and *DEFC* are equal, make those of *BCDA* and *EFAD* equivalent, as well as those of *FABE* and *CDEB*. This reduces the total number of unique eight by eight stiffness matrices to three. One is the original $\mathbf{K}_{ABCF}$ found through the displacement-based finite element analysis

described earlier. The other two, $\mathbf{K}_{FABE}$ and $\mathbf{K}_{BCDA}$, are transformations of $\mathbf{K}_{ABCF}$ with counterclockwise transformation angles of -60 degrees and +60 degrees respectively.

$$\mathbf{K}_{FABE} = \mathbf{Q}^{T}_{-60^\circ}\mathbf{K}_{ABCF}\mathbf{Q}_{-60^\circ} \qquad (31)$$

$$\mathbf{K}_{BCDA} = \mathbf{Q}^{T}_{+60^\circ}\mathbf{K}_{ABCF}\mathbf{Q}_{+60^\circ} \qquad (32)$$

With all six stiffness matrices known, the task now lies in overlaying them to find the stiffness matrix for the hexagon *ABCDEF*. This matrix will be a twelve by twelve square matrix since the force and local displacement vectors each have twelve components. Each of the eight by eight stiffness matrices then needed to be expanded and ordered to twelve by twelve matrices before they are superimposed. The expansion of a matrix is simply accomplished by placing the corresponding entries in the corresponding spots, for example $\mathbf{K}(\mathbf{F}_i, \hat{\mathbf{u}}_j)$ is the entry of a $\mathbf{K}$ matrix corresponding to $\mathbf{F}_i$ and $\hat{\mathbf{u}}_j$ and thus is mapped the ith columns and jth rows of matrix $\mathbf{K}$. It should be noted that the vectors of $\mathbf{F}_i$ and $\hat{\mathbf{u}}_j$ are listed in *x*-*y* pairs, and the ith columns and jth rows of matrix $\mathbf{K}$ refer to the corresponding two rows and two columns. A detailed mapping is shown below by the matrix $\mathbf{K}_{ABCF}$ and its twelve by twelve expansion.

$$\mathbf{F}^{T}_{ABCDEF} = \begin{bmatrix} f^x_A & f^y_A & f^x_B & f^y_B & f^x_C & f^y_C & f^x_D & f^y_D & f^x_E & f^y_E & f^x_F & f^x_F \end{bmatrix}$$

$$\mathbf{F}_{ABCDEF} = \mathbf{K}_{ABCDEF}\hat{\mathbf{u}}_{ABCDEF}$$

$$\mathbf{K}_{ABCF} \rightarrow \mathbf{K}_{ABCF}^{(12)}$$

$$\mathbf{K}_{ABCF} = Et\begin{bmatrix} 0.54 & 0.15 & -0.17 & -0.06 & -0.23 & -0.15 & -0.14 & 0.06 \\ 0.15 & 0.83 & 0.06 & 0.07 & -0.15 & -0.35 & -0.06 & -0.55 \\ -0.17 & 0.06 & 0.54 & -0.15 & -0.14 & -0.06 & -0.23 & 0.15 \\ -0.06 & 0.07 & -0.15 & 0.83 & 0.06 & -0.55 & 0.15 & -0.35 \\ -0.23 & -0.15 & -0.14 & 0.06 & 0.38 & 0.15 & -0.02 & -0.06 \\ -0.15 & -0.35 & -0.06 & -0.55 & 0.15 & 0.59 & 0.06 & 0.31 \\ -0.14 & -0.06 & -0.23 & 0.15 & -0.02 & 0.06 & 0.38 & -0.15 \\ 0.06 & -0.55 & 0.15 & -0.35 & -0.06 & 0.31 & -0.15 & 0.59 \end{bmatrix}$$

$$\mathbf{K}_{ABCF}^{(12)} = Et\begin{bmatrix} 0.54 & 0.15 & -0.17 & -0.06 & -0.23 & -0.15 & 0 & 0 & 0 & 0 & -0.14 & 0.06 \\ 0.15 & 0.83 & 0.06 & 0.07 & -0.15 & -0.35 & 0 & 0 & 0 & 0 & -0.06 & -0.55 \\ -0.17 & 0.06 & 0.54 & -0.15 & -0.14 & -0.06 & 0 & 0 & 0 & 0 & -0.23 & 0.15 \\ -0.06 & 0.07 & -0.15 & 0.83 & 0.06 & -0.55 & 0 & 0 & 0 & 0 & 0.15 & -0.35 \\ -0.23 & -0.15 & -0.14 & 0.06 & 0.38 & 0.15 & 0 & 0 & 0 & 0 & -0.02 & -0.06 \\ -0.15 & -0.35 & -0.06 & -0.55 & 0.15 & 0.59 & 0 & 0 & 0 & 0 & 0.06 & 0.31 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.14 & -0.06 & -0.23 & 0.15 & -0.02 & 0.06 & 0 & 0 & 0 & 0 & 0.38 & -0.15 \\ 0.06 & -0.55 & 0.15 & -0.35 & -0.06 & 0.31 & 0 & 0 & 0 & 0 & -0.15 & 0.59 \end{bmatrix}$$

As seen in Equation 23, the stiffness matrix of an element is obtained by integrating certain properties through the volume of the element, or in the two-dimensional case, its area. The six distinct trapezoids of the carbon ring cover the hexagon's area three times over. Thus, the stiffness matrix of the hexagon *ABCDEF* is one-third of the sum of the six twelve by twelve stiffness matrices of the trapezoidal components. Notice that the hexagonal stiffness matrix is symmetric.

$$\mathbf{K}_{ABCDEF} = \frac{1}{3}\left(\mathbf{K}^{(12)}_{ABCF} + \mathbf{K}^{(12)}_{BCDA} + \mathbf{K}^{(12)}_{CDEB} + \mathbf{K}^{(12)}_{DEFC} + \mathbf{K}^{(12)}_{EFAD} + \mathbf{K}^{(12)}_{FABE}\right) \tag{33}$$

$$\mathbf{K}_{4BCDEF} = Et \begin{bmatrix} 0.74 & 0.06 & -0.35 & -0.06 & -0.23 & -0.06 & 0.15 & -0.10 & -0.13 & 0 & -0.19 & 0.15 \\ 0.06 & 0.81 & 0.06 & -0.13 & -0.06 & -0.16 & -0.10 & 0.04 & 0 & -0.26 & 0.04 & -0.30 \\ -0.35 & 0.06 & 0.74 & -0.06 & -0.19 & -0.15 & -0.13 & 0 & 0.15 & 0.10 & -0.23 & -0.06 \\ -0.06 & -0.13 & -0.06 & 0.81 & -0.04 & -0.30 & 0 & -0.26 & 0.10 & 0.04 & 0.06 & -0.16 \\ -0.23 & -0.06 & -0.19 & -0.04 & 0.85 & 0 & -0.19 & 0.04 & -0.23 & 0.06 & -0.01 & 0 \\ -0.06 & -0.16 & -0.15 & -0.30 & 0 & 0.71 & 0.15 & -0.30 & 0.06 & -0.16 & 0 & 0.21 \\ 0.15 & -0.10 & -0.13 & 0 & -0.19 & 0.15 & 0.74 & 0.06 & -0.35 & -0.06 & -0.23 & -0.06 \\ -0.10 & 0.04 & 0 & -0.26 & 0.04 & -0.30 & 0.06 & 0.81 & 0.06 & -0.13 & -0.06 & -0.16 \\ -0.13 & 0 & 0.15 & 0.10 & -0.23 & 0.06 & -0.35 & 0.06 & -0.13 & -0.06 & -0.19 & -0.15 \\ 0 & -0.26 & 0.10 & 0.04 & 0.06 & -0.16 & -0.06 & -0.13 & -0.06 & 0.81 & -0.04 & -0.30 \\ -0.19 & 0.04 & -0.23 & 0.06 & -0.01 & 0 & -0.23 & -0.06 & -0.19 & -0.04 & 0.85 & 0 \\ 0.15 & -0.30 & 0.06 & -0.16 & 0 & 0.21 & -0.06 & -0.16 & -0.15 & -0.30 & 0 & 0.71 \end{bmatrix}$$

Two trapezoids can cover the area of the hexagon, but in doing so the universality of the matrix on all hexagons is lost. For example, the sum of the twelve by twelve stiffness matrices of trapezoids *ABCF* and *DEFC* does not equal that of *FABE* and *CDEB*.

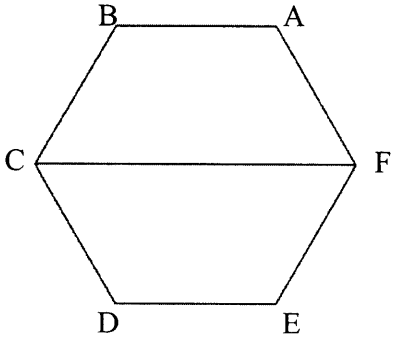

Figure 6: Above is the hexagon *ABCDEF* compartmentalized into trapezoids *ABCF* and *DEFC*. Special consideration is given in this orientation to the introduced segment *CF*.

The stiffness matrices trapezoids *ABCF* and *DEFC* give special consideration to the relationship between nodes C and F, but neglects the equivalent links between A and D, and B and E. By taking a third of the sum of all six trapezoids, each such link is equally weighted, and the resulting matrix is universal for any hexagonal ring of a carbon nanotube sheet of the given geometry. With some reordering, it holds for any rotation angle that is a multiple of 60 degrees. This behavior is investigated further in Appendix D.

The Global Stiffness Matrix for a Carbon Atom

Each atom of a carbon nanotube sheet belongs to three hexagonal rings. The global stiffness matrix for each node would then involve three hexagons. The global structure of node A is given in Figure 7.

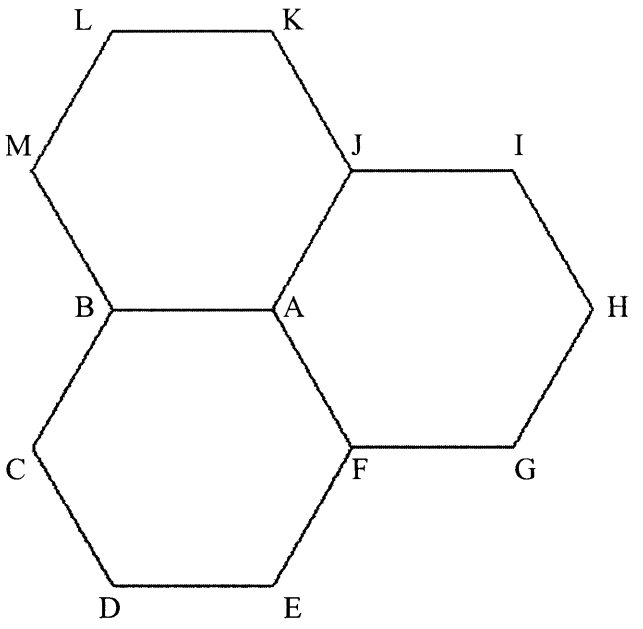

Figure 7: The global structure of node A is given by the three hexagonal rings that it belongs to.

The stiffness matrix of each ring is equivalent to the other two given their geometries. Reordering and expanding each of the three stiffness matrices and then superimposing them forms a twenty-six by twenty-six stiffness matrix of the three ring structure.

$$\mathbf{K}_{ABCDEF} = \mathbf{K}_{IJAFGH} = \mathbf{K}_{KLMBAJ} \qquad (34)$$

$$\mathbf{K}_{ABCDEF} \rightarrow \mathbf{K}^{(26)}_{ABCDEF}$$

$$\mathbf{K} = \mathbf{K}^{(26)}_{ABCDEF} + \mathbf{K}^{(26)}_{IJAFGH} + \mathbf{K}^{(26)}_{KLMBAJ} \qquad (35)$$

The node A columns, and by symmetry the transpose of the node A rows, are given below. These columns represent the force on the members of the global three ring

structure required to keep all nodes still with displacements of node A. The vectors **F** and $\hat{\mathbf{u}}$ are defined for all thirteen nodes ordered alphabetically from A through M. The full twenty-six by twenty-six global stiffness matrix is given in Appendix B.

$$\mathbf{K}(\mathbf{F},\hat{\mathbf{u}}_A)=(\mathbf{K}(\mathbf{F}_A,\hat{\mathbf{u}}))^T = Et\begin{bmatrix} 2.34 & 0 \\ 0 & 2.34 \\ -0.71 & 0 \\ 0 & -0.27 \\ -0.23 & -0.06 \\ -0.06 & -0.16 \\ 0.15 & -0.10 \\ -0.10 & 0.04 \\ -0.13 & 0 \\ 0 & -0.26 \\ -0.38 & 0.19 \\ 0.19 & -0.60 \\ -0.23 & 0.06 \\ 0.06 & -0.16 \\ -0.01 & 0 \\ 0 & 0.21 \\ -0.23 & -0.06 \\ -0.06 & -0.16 \\ -0.38 & -0.19 \\ -0.19 & -0.60 \\ -0.13 & 0 \\ 0 & -0.26 \\ 0.15 & 0.10 \\ 0.10 & 0.04 \\ -0.23 & 0.06 \\ 0.06 & -0.16 \end{bmatrix}$$

Conclusion

With atomic-level structures such as the one with the graphene sheet, methods involving atomic potentials are commonly used to derive stiffness matrices of the structure. To compare the results from the finite element based stiffness calculated in this paper to the methods of two and three atom potentials, the nodes on opposite sides of a hex are examined:

$$\frac{1}{Et}\mathbf{K}(\mathbf{F}_A, \hat{\mathbf{u}}_A) = \begin{bmatrix} 2.34 & 0 \\ 0 & 2.34 \end{bmatrix} \qquad \frac{1}{Et}\mathbf{K}(\mathbf{F}_A, \hat{\mathbf{u}}_H) = \begin{bmatrix} -0.01 & 0 \\ 0 & 0.21 \end{bmatrix}.$$

Interesting to note, the two by two self stiffness matrix of node A is isotropic as expected: a unit displacement should correspond to a unit force on itself. As seen, a unit displacement on node A of the hexagon results in a relatively large force on node A, yet a negligible force almost two orders of magnitude less on node H. This is consistent with findings of the methods involving two and three atom potentials. For more information on the potential methods and their results consult the works of Tersoff, Brenner, and Odegard listed in the references.

Not mentioned previously is the existence of pi electron orbitals on each carbon that are normal to the hexagon's plane. The finite element approach is a convenient approach and can be used to analyze the shell elements of carbon nanotubes involving the out of plane response, something that was ignored in this analysis of a flat single layer.

References

Bathe, Klaus-Jürgen. Finite Element Procedures. Prentice-Hall. New Jersey, 1996.

Brenner, Donald W. "Empirical Potential for Hydrocarbons for Use in Simulating the Chemical Vapor Deposition of Diamond Films." *Physical Review B*, 42.15 (1990): 9458-9471.

Harris, Peter J. F. Carbon Nanotubes and Related Structures. Cambridge University Press. 1999.

Odegard, Gregory M. et al. "Equivalent-Continuum Modeling of Nano-Structured Materials." *Composites Science and Technology*, 62 (2002): 1869–1880

Pantano, Antonio et al. "Mechanics of Deformation of Single and Multi-Wall Carbon Nanotubes." *Journal of the Mechanics and Physics of Solids*, 52.4 (2004): 789-821.

Pantano, Antonio et al. "Mixed Finite Element-Tight-Binding Electromechanical Analysis of Carbon Nanotubes." *Journal of Applied Physics*, 92.11 (2004): 6756-6760.

Tersoff, J. "New Empirical Approach for the Structure and Energy of Covalent Systems." *Physical Review B*, 37.12 (1988): 6991-7000.

Appendix A

The following is the m-file used to derive the stiffness matrices described in the text.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   CNTstiff.m      written by:      Kirk Samaroo
%   Last Update: 5/6/2005
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

r=sym('r');
s=sym('s');
a=sym('a');

E=sym('E');
%v=sym('v');

%E=4.8*10^12; %Pascals
v=0.19;
                                      %The following refers to pgs. 8-11 of the text
J=a/4*[3-s 0; -r sqrt(3)];                       %The Jacobian matrix

dxdr=1/4*[1+s 0 -(1+s) 0 -(1-s) 0 1-s 0];        %The derivatives of the mapping equations
dxds=1/4*[1+r 0 1-r 0 -(1-r) 0 -(1+r) 0];

dydr=1/4*[0 1+s 0 -(1+s) 0 -(1-s) 0 1-s];
dyds=1/4*[0 1+r 0 1-r 0 -(1-r) 0 -(1+r)];
                                      %The following refers to pgs. 12-14 of the text
du=J^-1*[dxdr; dxds];                       %This is the intermediate step of calculating
dv=J^-1*[dydr; dyds];                       %the strain-displacement matrix B.

B=[du(1,:); dv(2,:); du(2,:)+dv(1,:)];            %The strain-displacement matrix.
                                      %The following refers to pg. 7 of the text
D=(E/(1-v^2))*[1 v 0; v 1 0; 0 0 (1/2)*(1-v)];   %The elasticity matrix.
                                      %The following refers to pgs. 15-16 of the text

F=(B.')*D*B*det(J);

K1=int(int(F, r, -1, 1),s,-1,1);                   %Integrating out r and s to find the
                                                   %stifness matrix for ABCF and DEFC.
                                      %The following refers to pgs. 16-20 of the text
theta2=pi/3;

R2=[cos(theta2) -sin(theta2); sin(theta2) cos(theta2)];
Z=zeros(2,2);
Q2=[R2 Z Z Z; Z R2 Z Z; Z Z R2 Z; Z Z Z R2];       %Rotation matrix for +60 degrees.

K2=Q2*K1*(Q2.');                                   %Stiffness matrix for BCDA and EFAD.

theta3=-pi/3;

R3=[cos(theta3) -sin(theta3); sin(theta3) cos(theta3)];
Z=zeros(2,2);
Q3=[R3 Z Z Z; Z R3 Z Z; Z Z R3 Z; Z Z Z R3];          %Rotation matrix for -60 degrees.


K3=Q3*K1*(Q3.');                                   %Stiffness matrix for FABE and CDEB.
                                      %The following refers to pgs. 20-24 of the text
k1a=[K1(1:6, 1:6) zeros(6,4) K1(1:6, 7:8);                %12x12 expansion of ABCF.
    zeros(4,12);
    K1(7:8, 1:6) zeros(2,4) K1(7:8, 7:8)];

k1b=[zeros(4,12);                                         %12x12 expansion of DEFC.
    zeros(2,4) K1(7:8, 7:8) K1(7:8, 1:6);
    zeros(6,4) K1(1:6, 7:8) K1(1:6, 1:6)];
                                                          %12x12 expansion of BCDA.
```

```
k2a=[K2(7:8, 7:8) K2(7:8, 1:6) zeros(2,4);
    K2(1:6, 7:8) K2(1:6, 1:6) zeros(6,4);
    zeros(4,12)];

k2b=[K2(5:6, 5:6) zeros(2,4) K2(5:6, 7:8) K2(5:6, 1:4);     %12x12 expansion of EFAD.
    zeros(4,12);
    K2(7:8, 5:6) zeros(2,4) K2(7:8, 7:8) K2(7:8, 1:4);
    K2(1:4, 5:6) zeros(4,4) K2(1:4, 7:8) K2(1:4, 1:4)];

k3a=[K3(3:6, 3:6) zeros(4,4) K3(3:6, 7:8) K3(3:6, 1:2);     %12x12 expansion of FABE.
    zeros(4,12);
    K3(7:8, 3:6) zeros(2,4) K3(7:8, 7:8) K3(7:8, 1:2);
    K3(1:2, 3:6) zeros(2,4) K3(1:2, 7:8) K3(1:2, 1:2)];

k3b=[zeros(2,12);                                           %12x12 expansion of CDEB.
    zeros(2,2) K3(7:8, 7:8) K3(7:8, 1:6) zeros(2,2);
    zeros(6,2) K3(1:6, 7:8) K3(1:6, 1:6) zeros(6,2);
    zeros(2,12)];

k=1/3*(k1a+k1b+k2a+k2b+k3a+k3b);            %Composite 12x12 stiffness matrix for ABCDEF.
                                  %The following refers to pgs. 24-26 of the text
ka=[k(1:12, 1:12) zeros(12,14);                             %26x26 expansion for ABCDEF
    zeros(14,26)];
                                                            %26x26 expansion of IJAFGH.
kb=[k(5:6, 5:6) zeros(2,8) k(5:6, 7:12) k(5:6, 1:4) zeros(2,6);
    zeros(8,26);
    k(7:12, 5:6) zeros(6,8) k(7:12, 7:12) k(7:12, 1:4) zeros(6,6);
    k(1:4, 5:6) zeros(4,8) k(1:4, 7:12) k(1:4, 1:4) zeros(4,6);
    zeros(6,26)];
                                                            %26x26 expansion of KLMBAJ.
kc=[k(9:10, 9:10) k(9:10, 7:8) zeros(2,14) k(9:10, 11:12) k(9:10, 1:6);
    k(7:8, 9:10) k(7:8, 7:8) zeros(2,14) k(7:8, 11:12) k(7:8, 1:6);
    zeros(14, 26);
    k(11:12, 9:10) k(11:12, 7:8) zeros(2,14) k(11:12, 11:12) k(11:12, 1:6);
    k(1:6, 9:10) k(1:6, 7:8) zeros(6,14) k(1:6, 11:12) k(1:6, 1:6)];

K=ka+kb+kc;                                  %26x26 Global stiffness matrix for node A.
```

Appendix B

Listings of essential normalized stiffness matrices follow.

$$\frac{1}{Et}\mathbf{K}_{ABCF} =$$

0.5389	0.1543	-0.1751	-0.0558	-0.2280	-0.1543	-0.1358	0.0558
0.1543	0.8252	0.0558	0.0733	-0.1543	-0.3522	-0.0558	-0.5462
-0.1751	0.0558	0.5389	-0.1543	-0.1358	-0.0558	-0.2280	0.1543
-0.0558	0.0733	-0.1543	0.8252	0.0558	-0.5462	0.1543	-0.3522
-0.2280	-0.1543	-0.1358	0.0558	0.3835	0.1543	-0.0196	-0.0558
-0.1543	-0.3522	-0.0558	-0.5462	0.1543	0.5887	0.0558	0.3098
-0.1358	-0.0558	-0.2280	0.1543	-0.0196	0.0558	0.3835	-0.1543
0.0558	-0.5462	0.1543	-0.3522	-0.0558	0.3098	-0.1543	0.5887

$$\frac{1}{Et}\mathbf{K}_{ABCDEF} =$$

Columns 1 through 8

0.7446	0.0592	-0.3541	-0.0558	-0.2276	-0.0592	0.1516	-0.0951
0.0592	0.8130	0.0558	-0.1345	-0.0592	-0.1592	-0.0951	0.0418
-0.3541	0.0558	0.7446	-0.0592	-0.1894	-0.1508	-0.1250	0
-0.0558	-0.1345	-0.0592	0.8130	-0.0393	-0.2992	0	-0.2618
-0.2276	-0.0592	-0.1894	-0.0393	0.8472	0	-0.1894	0.0393
-0.0592	-0.1592	-0.1508	-0.2992	0	0.7104	0.1508	-0.2992
0.1516	-0.0951	-0.1250	0	-0.1894	0.1508	0.7446	0.0592
-0.0951	0.0418	0	-0.2618	0.0393	-0.2992	0.0592	0.8130
-0.1250	0	0.1516	0.0951	-0.2276	0.0592	-0.3541	0.0558
0	-0.2618	0.0951	0.0418	0.0592	-0.1592	-0.0558	-0.1345
-0.1894	0.0393	-0.2276	0.0592	-0.0131	0	-0.2276	-0.0592
0.1508	-0.2992	0.0592	-0.1592	0	0.2065	-0.0592	-0.1592

Columns 9 through 12

-0.1250	0	-0.1894	0.1508
0	-0.2618	0.0393	-0.2992
0.1516	0.0951	-0.2276	0.0592
0.0951	0.0418	0.0592	-0.1592
-0.2276	0.0592	-0.0131	0
0.0592	-0.1592	0	0.2065
-0.3541	-0.0558	-0.2276	-0.0592
0.0558	-0.1345	-0.0592	-0.1592
0.7446	-0.0592	-0.1894	-0.1508
-0.0592	0.8130	-0.0393	-0.2992
-0.1894	-0.0393	0.8472	0
-0.1508	-0.2992	0	0.7104

$$\frac{1}{Et}\mathbf{K}^{(26)} =$$

Columns 1 through 8

2.3364	0	-0.7082	0	-0.2276	-0.0592	0.1516	-0.0951
0	2.3364	0	-0.2691	-0.0592	-0.1592	-0.0951	0.0418
-0.7082	0	1.4892	0	-0.1894	-0.1508	-0.1250	0
0	-0.2691	0	1.6260	-0.0393	-0.2992	0	-0.2618
-0.2276	-0.0592	-0.1894	-0.0393	0.8472	0	-0.1894	0.0393
-0.0592	-0.1592	-0.1508	-0.2992	0	0.7104	0.1508	-0.2992
0.1516	-0.0951	-0.1250	0	-0.1894	0.1508	0.7446	0.0592
-0.0951	0.0418	0	-0.2618	0.0393	-0.2992	0.0592	0.8130
-0.1250	0	0.1516	0.0951	-0.2276	0.0592	-0.3541	0.0558
0	-0.2618	0.0951	0.0418	0.0592	-0.1592	-0.0558	-0.1345
-0.3789	0.1902	-0.2276	0.0592	-0.0131	0	-0.2276	-0.0592
0.1902	-0.5984	0.0592	-0.1592	0	0.2065	-0.0592	-0.1592
-0.2276	0.0592	0	0	0	0	0	0
0.0592	-0.1592	0	0	0	0	0	0
-0.0131	0	0	0	0	0	0	0
0	0.2065	0	0	0	0	0	0
-0.2276	-0.0592	0	0	0	0	0	0

-0.0592	-0.1592	0	0	0	0	0	0
-0.3789	-0.1902	-0.2276	-0.0592	0	0	0	0
-0.1902	-0.5984	-0.0592	-0.1592	0	0	0	0
-0.1250	0	0.1516	-0.0951	0	0	0	0
0	-0.2618	-0.0951	0.0418	0	0	0	0
0.1516	0.0951	-0.1250	0	0	0	0	0
0.0951	0.0418	0	-0.2618	0	0	0	0
-0.2276	0.0592	-0.1894	0.0393	0	0	0	0
0.0592	-0.1592	0.1508	-0.2992	0	0	0	0

Columns 9 through 16

-0.1250	0	-0.3789	0.1902	-0.2276	0.0592	-0.0131	0
0	-0.2618	0.1902	-0.5984	0.0592	-0.1592	0	0.2065
0.1516	0.0951	-0.2276	0.0592	0	0	0	0
0.0951	0.0418	0.0592	-0.1592	0	0	0	0
-0.2276	0.0592	-0.0131	0	0	0	0	0
0.0592	-0.1592	0	0.2065	0	0	0	0
-0.3541	-0.0558	-0.2276	-0.0592	0	0	0	0
0.0558	-0.1345	-0.0592	-0.1592	0	0	0	0
0.7446	-0.0592	-0.1894	-0.1508	0	0	0	0
-0.0592	0.8130	-0.0393	-0.2992	0	0	0	0
-0.1894	-0.0393	1.5918	0.0592	-0.3541	-0.0558	-0.2276	-0.0592
-0.1508	-0.2992	0.0592	1.5234	0.0558	-0.1345	-0.0592	-0.1592
0	0	-0.3541	0.0558	0.7446	-0.0592	-0.1894	-0.1508
0	0	-0.0558	-0.1345	-0.0592	0.8130	-0.0393	-0.2992
0	0	-0.2276	-0.0592	-0.1894	-0.0393	0.8472	0
0	0	-0.0592	-0.1592	-0.1508	-0.2992	0	0.7104
0	0	0.1516	-0.0951	-0.1250	0	-0.1894	0.1508
0	0	-0.0951	0.0418	0	-0.2618	0.0393	-0.2992
0	0	-0.1250	0	0.1516	0.0951	-0.2276	0.0592
0	0	0	-0.2618	0.0951	0.0418	0.0592	-0.1592
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Columns 17 through 24

-0.2276	-0.0592	-0.3789	-0.1902	-0.1250	0	0.1516	0.0951
-0.0592	-0.1592	-0.1902	-0.5984	0	-0.2618	0.0951	0.0418
0	0	-0.2276	-0.0592	0.1516	-0.0951	-0.1250	0
0	0	-0.0592	-0.1592	-0.0951	0.0418	0	-0.2618
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0.1516	-0.0951	-0.1250	0	0	0	0	0
-0.0951	0.0418	0	-0.2618	0	0	0	0
-0.1250	0	0.1516	0.0951	0	0	0	0
0	-0.2618	0.0951	0.0418	0	0	0	0
-0.1894	0.0393	-0.2276	0.0592	0	0	0	0
0.1508	-0.2992	0.0592	-0.1592	0	0	0	0
0.7446	0.0592	-0.3541	-0.0558	0	0	0	0
0.0592	0.8130	0.0558	-0.1345	0	0	0	0
-0.3541	0.0558	1.5918	-0.0592	-0.1894	0.0393	-0.2276	0.0592
-0.0558	-0.1345	-0.0592	1.5234	0.1508	-0.2992	0.0592	-0.1592
0	0	-0.1894	0.1508	0.7446	0.0592	-0.3541	-0.0558
0	0	0.0393	-0.2992	0.0592	0.8130	0.0558	-0.1345
0	0	-0.2276	0.0592	-0.3541	0.0558	0.7446	-0.0592
0	0	0.0592	-0.1592	-0.0558	-0.1345	-0.0592	0.8130
0	0	-0.0131	0	-0.2276	-0.0592	-0.1894	-0.0393
0	0	0	0.2065	-0.0592	-0.1592	-0.1508	-0.2992

Columns 25 through 26

-0.2276	0.0592
0.0592	-0.1592
-0.1894	0.1508
0.0393	-0.2992
0	0
0	0
0	0
0	0
0	0
0	0
0	0
0	0
0	0
0	0
0	0
0	0
0	0
-0.0131	0
0	0.2065
-0.2276	-0.0592
-0.0592	-0.1592
-0.1894	-0.1508
-0.0393	-0.2992
0.8472	0
0	0.7104

Appendix C

Both the $\mathbf{K}_{ABCDEF}$ and the twenty-six by twenty-six global matrix demonstrated some initially unexpected behavior. These unexpected behaviors were ultimately rationalized, but the following m-file was written as a check calculation. It calculates all six of the eight by eight matrices contributing to $\mathbf{K}_{ABCDEF}$ individually using the unique Jacobian operator and strain-displacement matrix in conventional *x-y* coordinates for each of the trapezoidal structures.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   troubleshoot.m      written by      Kirk Samaroo
%   Last Update 5/6/2005
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


r=sym('r');
s=sym('s');
a=sym('a');

E=sym('E');
%v=sym('v');

%E=4.8*10^12; %Terapascals
v=0.19;

D=(E/(1-v^2))*[1 v 0; v 1 0; 0 0 (1/2)*(1-v)];

xa=a/2; xb=-a/2; xc=-a; xd=-a/2; xe=a/2; xf=a;
ya=a/2*3^.5; yb=a/2*3^.5; yc=0; yd=-a/2*3^.5; ye=-a/2*3^.5; yf=0;

dxdr=1/4*[1+s 0 -(1+s) 0 -(1-s) 0 1-s 0];
dxds=1/4*[1+r 0 1-r 0 -(1-r) 0 -(1+r) 0];

dydr=1/4*[0 1+s 0 -(1+s) 0 -(1-s) 0 1-s];
dyds=1/4*[0 1+r 0 1-r 0 -(1-r) 0 -(1+r)];

trap=['abcf'; 'defc'; 'bcda'; 'efad'; 'fabe'; 'cdeb'];

for ii=1:6
    eval(['q=[x' trap(ii,1) '; y' trap(ii,1) '; x' trap(ii,2) '; y' trap(ii,2) '; x'
trap(ii,3) '; y' trap(ii,3) '; x' trap(ii,4) '; y' trap(ii,4) '];'])
    eval(['J' trap(ii,:) '=[[[dxdr; dxds]*q] [[dydr; dyds]*q]];'])

    eval(['du_' trap(ii,:) '=inv(J' trap(ii,:) ')*[dxdr; dxds];'])
    eval(['dv_' trap(ii,:) '=inv(J' trap(ii,:) ')*[dydr; dyds];'])

    eval(['B_' trap(ii,:) '=[du_' trap(ii,:) '(1,:); dv_' trap(ii,:) '(2,:); du_'
trap(ii,:) '(2,:)+dv_' trap(ii,:) '(1,:)];'])

    eval(['F_' trap(ii,:) '=transpose(B_' trap(ii,:) ')*D*B_' trap(ii,:) '*det(J'
trap(ii,:) ');'])
    eval(['k_' trap(ii,:) '=int(int(F_' trap(ii,:) ', r, -1, 1),s,-1,1);'])
end
```

Appendix D

The matrix $\mathbf{K}_{ABCDEF}$ demonstrated the geometric symmetry expected of an equilateral hexagon. Through nodal reordering, any rotation of a multiple of sixty degrees produces an identical matrix. In the orientation of nodes used in this paper a +60 degree rotation is a single leftward shift in the reordering of nodes of the structure and likewise for the matrix columns and rows. For example, hexagon *ABCDEF* rotated by +60 degrees has the same orientation as hexagon *BCDEFA*, and likewise with their respective stiffness matrices. Notice in the two stiffness matrices listed below, the first two columns and first two rows of the first matrix (those that correspond to the *x-y* force and displacement components of node A) are the last two columns and rows of the transformed matrix. A -60 degree rotation corresponds to a rightward shift of the ordering.

$$\frac{1}{Et}\mathbf{K}_{ABCDEF} =$$

Columns 1 through 8

0.7446	0.0592	-0.3541	-0.0558	-0.2276	-0.0592	0.1516	-0.0951
0.0592	0.8130	0.0558	-0.1345	-0.0592	-0.1592	-0.0951	0.0418
-0.3541	0.0558	0.7446	-0.0592	-0.1894	-0.1508	-0.1250	0
-0.0558	-0.1345	-0.0592	0.8130	-0.0393	-0.2992	0	-0.2618
-0.2276	-0.0592	-0.1894	-0.0393	0.8472	0	-0.1894	0.0393
-0.0592	-0.1592	-0.1508	-0.2992	0	0.7104	0.1508	-0.2992
0.1516	-0.0951	-0.1250	0	-0.1894	0.1508	0.7446	0.0592
-0.0951	0.0418	0	-0.2618	0.0393	-0.2992	0.0592	0.8130
-0.1250	0	0.1516	0.0951	-0.2276	0.0592	-0.3541	0.0558
0	-0.2618	0.0951	0.0418	0.0592	-0.1592	-0.0558	-0.1345
-0.1894	0.0393	-0.2276	0.0592	-0.0131	0	-0.2276	-0.0592

0.1508	-0.2992	0.0592	-0.1592	0	0.2065	-0.0592	-0.1592

Columns 9 through 12

-0.1250	0	-0.1894	0.1508
0	-0.2618	0.0393	-0.2992
0.1516	0.0951	-0.2276	0.0592
0.0951	0.0418	0.0592	-0.1592
-0.2276	0.0592	-0.0131	0
0.0592	-0.1592	0	0.2065
-0.3541	-0.0558	-0.2276	-0.0592
0.0558	-0.1345	-0.0592	-0.1592
0.7446	-0.0592	-0.1894	-0.1508
-0.0592	0.8130	-0.0393	-0.2992
-0.1894	-0.0393	0.8472	0
-0.1508	-0.2992	0	0.7104

$$\frac{1}{Et}\mathbf{Q}_{+60^\circ}^{T}\mathbf{K}_{ABCDEF}\mathbf{Q}_{+60^\circ} =$$

Columns 1 through 8

0.7446	-0.0592	-0.1894	-0.1508	-0.1250	-0.0000	0.1516	0.0951
-0.0592	0.8130	-0.0393	-0.2992	-0.0000	-0.2618	0.0951	0.0418
-0.1894	-0.0393	0.8472	0	-0.1894	0.0393	-0.2276	0.0592
-0.1508	-0.2992	-0.0000	0.7104	0.1508	-0.2992	0.0592	-0.1592
-0.1250	-0.0000	-0.1894	0.1508	0.7446	0.0592	-0.3541	-0.0558
-0.0000	-0.2618	0.0393	-0.2992	0.0592	0.8130	0.0558	-0.1345
0.1516	0.0951	-0.2276	0.0592	-0.3541	0.0558	0.7446	-0.0592
0.0951	0.0418	0.0592	-0.1592	-0.0558	-0.1345	-0.0592	0.8130
-0.2276	0.0592	-0.0131	0.0000	-0.2276	-0.0592	-0.1894	-0.0393
0.0592	-0.1592	0.0000	0.2065	-0.0592	-0.1592	-0.1508	-0.2992
-0.3541	-0.0558	-0.2276	-0.0592	0.1516	-0.0951	-0.1250	-0.0000
0.0558	-0.1345	-0.0592	-0.1592	-0.0951	0.0418	-0.0000	-0.2618

```
  Columns 9 through 12

   -0.2276    0.0592   -0.3541    0.0558
    0.0592   -0.1592   -0.0558   -0.1345
   -0.0131    0.0000   -0.2276   -0.0592
    0.0000    0.2065   -0.0592   -0.1592
   -0.2276   -0.0592    0.1516   -0.0951
   -0.0592   -0.1592   -0.0951    0.0418
   -0.1894   -0.1508   -0.1250   -0.0000
   -0.0393   -0.2992   -0.0000   -0.2618
    0.8472         0   -0.1894    0.0393
   -0.0000    0.7104    0.1508   -0.2992
   -0.1894    0.1508    0.7446    0.0592
    0.0393   -0.2992    0.0592    0.8130
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   symmetry.m      written by:      Kirk Samaroo
%   Last Update: 5/6/2005
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

run CNTstiff

Q=zeros(12,12);
kn=eval(simplify(1/E*k));

for t=1:1:6;

    R=[cos(t*pi/3) sin(t*pi/3); -sin(t*pi/3) cos(t*pi/3)];
    Q(1:2,:)=[R zeros(2,10)];
    Q(11:12, :)=[zeros(2,10) R];

    for ii=4:2:12;
        Q((ii-1):ii,:)=[zeros(2, ii-2) R zeros(2,12-ii)];
    end

    P=(Q.');
    eval(['k' num2str(t) '=P*kn*Q;'])
end
```

Appendix E

A normalized two by two Gauss quadrature approximation of the trapezoidal stiffness matrix for *ABCF* was obtained and compared to the normalized analytical matrix of the isosceles trapezoid. Gauss quadrature is a numeric method using weighted sampling points and values to approximate integrals. A two by two quadrature was used because the symmetries of the mapped square *ABCF* in *r-s* space are easily manipulated into two sampling points per axis.

The integral of Equation 24 then becomes:

$$\mathbf{K}_{ABCF} = \int_{-1}^{+1}\int_{-1}^{+1} \mathbf{B}^T\mathbf{D}\mathbf{B}\det\mathbf{J}\,t dr ds \equiv \int_{-1}^{+1}\int_{-1}^{+1} \mathbf{G}(r,s)\,dr ds \cong \int_{-1}^{+1} (\alpha_1\mathbf{G}(r_1,s)+\alpha_2\mathbf{G}(r_2,s))\,ds$$

$$\equiv \int_{-1}^{+1} \mathbf{H}(s)\,ds \cong \beta_1\mathbf{H}(s_1)+\beta_2\mathbf{H}(s_2). \qquad \text{(E1)}$$

Here, α_i and β_i are weights of the approximated functions, and r_i and s_i are sampling points. The sampling points can be derived from the following equations:

$$\int_{-1}^{+1}(r-r_1)(r-r_2)\,dr = 0\,; \quad \int_{-1}^{+1}(r-r_1)(r-r_2)\,r dr = 0\,.$$

By symmetry, replacing r by s everywhere provides the s-direction sampling points. These equations combine to show:

$$r_1 = s_1 = \frac{1}{\sqrt{3}}; \quad r_2 = s_2 = -\frac{1}{\sqrt{3}}. \qquad \text{(E2)}$$

The weights of the approximation are defined below:

$$\alpha_1 = \int_{-1}^{+1} \frac{r - r_2}{r_1 - r_2} dr = 1; \quad \alpha_2 = \int_{-1}^{+1} \frac{r - r_1}{r_2 - r_1} dr = 1. \quad \text{(E3)}$$

Again by symmetry, the weights β_i can be found replacing β by α and r by s everywhere. All weights are thus unity. Plugging the weights and sampling points into Equation E1 provides the stiffness matrix obtained by the two by two Gaussian quadrature. More detail about generalized Gauss quadrature can be found in Chapter 5 of Bathe's text listed in the references.

The difference of the analytical and approximated matrices was normalized with the largest element of the analytical matrix.

$$\Delta \mathbf{K}_{norm} = \frac{\mathbf{K}_{ABCF} - \tilde{\mathbf{K}}_{ABCF}}{(\mathbf{K}_{ABCF})_{\max}} =$$

0.0038	0	-0.0038	0	0.0019	0.0000	-0.0019	-0.0000
0	0.0019	0	-0.0019	0	0.0009	0	-0.0009
-0.0038	0	0.0038	0	-0.0019	0.0000	0.0019	-0.0000
0	-0.0019	0	0.0019	0	-0.0009	0	0.0009
0.0019	0	-0.0019	0	0.0010	0	-0.0010	0
0.0000	0.0009	0.0000	-0.0009	0	0.0005	0	-0.0005

-0.0019	0	0.0019	0	-0.0010	0	0.0010	0
-0.0000	-0.0009	-0.0000	0.0009	0	-0.0005	0	0.0005

The normalized analytical matrix follows.

$$\frac{1}{Et}\mathbf{K}_{ABCF} =$$

0.5389	0.1543	-0.1751	-0.0558	-0.2280	-0.1543	-0.1358	0.0558
0.1543	0.8252	0.0558	0.0733	-0.1543	-0.3522	-0.0558	-0.5462
-0.1751	0.0558	0.5389	-0.1543	-0.1358	-0.0558	-0.2280	0.1543
-0.0558	0.0733	-0.1543	0.8252	0.0558	-0.5462	0.1543	-0.3522
-0.2280	-0.1543	-0.1358	0.0558	0.3835	0.1543	-0.0196	-0.0558
-0.1543	-0.3522	-0.0558	-0.5462	0.1543	0.5887	0.0558	0.3098
-0.1358	-0.0558	-0.2280	0.1543	-0.0196	0.0558	0.3835	-0.1543
0.0558	-0.5462	0.1543	-0.3522	-0.0558	0.3098	-0.1543	0.5887

The eigenvalues of the analytical stiffness matrix are given in the first row below with each corresponding eigenvector in the column directly beneath.

-0.0000	-0.0000	0.0000	0.3708	0.7129	0.7138	1.0449	1.8304
-0.3252	0.4089	0.1862	0.1636	-0.6816	-0.2395	0.3673	-0.0931
-0.2237	-0.1061	-0.5154	0.0361	0.0760	0.4468	0.4727	-0.4929
-0.3252	0.4089	0.1862	-0.1636	0.6816	-0.2395	0.3673	0.0931
-0.5223	-0.2239	-0.0630	0.0361	0.0760	-0.4468	-0.4727	-0.4929
-0.0666	0.5109	-0.2055	0.6860	0.1545	0.2395	-0.3673	0.0741
-0.6717	-0.2827	0.1632	-0.0361	-0.0760	0.4308	-0.0817	0.4929
-0.0666	0.5109	-0.2055	-0.6860	-0.1545	0.2395	-0.3673	-0.0741
-0.0744	-0.0473	-0.7416	-0.0361	-0.0760	-0.4308	0.0817	0.4929

The normalized approximation using two by two Gaussian quadrature follows.

$$\frac{1}{Et}\tilde{\mathbf{K}}_{ABCF} =$$

0.5358	0.1543	-0.1719	-0.0558	-0.2296	-0.1543	-0.1343	0.0558
0.1543	0.8236	0.0558	0.0748	-0.1543	-0.3530	-0.0558	-0.5455
-0.1719	0.0558	0.5358	-0.1543	-0.1343	-0.0558	-0.2296	0.1543
-0.0558	0.0748	-0.1543	0.8236	0.0558	-0.5455	0.1543	-0.3530
-0.2296	-0.1543	-0.1343	0.0558	0.3827	0.1543	-0.0188	-0.0558
-0.1543	-0.3530	-0.0558	-0.5455	0.1543	0.5883	0.0558	0.3101
-0.1343	-0.0558	-0.2296	0.1543	-0.0188	0.0558	0.3827	-0.1543
0.0558	-0.5455	0.1543	-0.3530	-0.0558	0.3101	-0.1543	0.5883

The eigenvalues of the numerical stiffness matrix are given in the first row below with each corresponding eigenvector in the column directly beneath.

-0.0000	-0.0000	0.0000	0.3675	0.7084	0.7110	1.0437	1.8303
0.4261	-0.2955	0.1970	0.1713	-0.6797	-0.2376	0.3686	-0.0928
-0.1410	-0.2919	-0.4710	0.0354	0.0762	0.4493	0.4703	-0.4929
0.4261	-0.2955	0.1970	-0.1713	0.6797	-0.2376	0.3686	0.0928
-0.2207	-0.5269	0.0251	0.0354	0.0762	-0.4493	-0.4703	-0.4929
0.4951	-0.0920	-0.2326	0.6842	0.1623	0.2376	-0.3686	0.0742
-0.2605	-0.6444	0.2732	-0.0354	-0.0762	0.4304	-0.0841	0.4929
0.4951	-0.0920	-0.2326	-0.6842	-0.1623	0.2376	-0.3686	-0.0742
-0.1012	-0.1744	-0.7190	-0.0354	-0.0762	-0.4304	0.0841	0.4929

Note that each of the non-zero eigenvalues of the analytical matrix is greater than the corresponding eigenvalue of the approximation. The following are the inputs used to obtain the approximation of a trapezoidal stiffness matrix using a Gaussian quadriture then comparing that with the analytical results obtained preceded by a description of the inputs themselves.

Normalizing the `F` (integrand of the stiffness matrix of *ABCF*) found in CNTstiff.m (see Appendix A) by dividing it by `E` (`F` is already normalized with the thickness t), then renaming the matrix as `R1` and using the value for `r=3^-(1/2)`. The process was repeated using the name `R2` and the value `r=-3^-(1/2)`. The sum of these two matrices `R1` and `R2` complete the approximation in the r-direction. The resultant matrix `R` is then renamed `S1` with a value `s=3^(1/2)`, then again `S2` with a value `s=-3^(1/2)`. The sum of `S1` and `S2`, `G`, completes the approximation over both axes. These values for `r` and `s` are obtained through the Gaussian quadrature point approximations described in Section 5.5.3 of the text by Bathe listed in the references. The equivalent matrix from CNTstiff.m, `K1`, is compared to `G`. First `K1` is normalized without `E` or a thickness t, then renamed `W`. The difference between the approximation and the analytical result, `dK=W-G`, is normalized with the largest difference and renamed `dKn`.

```
s=sym('s');

r=3^-(1/2);
R1=[                    -25/77112*(443+238*s+227*s^2+432*r^2+648*r-216*r*s)*3^(1/2)/(-3+s),
-25/648*(4*r+3-s)*(1+s)/(-3+s),                                                    25/77112*(-
43+562*s+173*s^2+432*r^2)*3^(1/2)/(-3+s),
25/77112*(476*r+129-43*s)*(1+s)/(-3+s),                    -25/77112*(-443+173*s^2+216*r^2-
162*r+54*r*s+162*s)*3^(1/2)/(-3+s),                                    -25/77112*(-248*r-
357+248*s+400*r*s-43*s^2)/(-3+s),              25/77112*(43+227*s^2+216*r^2+486*r-162*r*s-
162*s)*3^(1/2)/(-3+s),                                 25/77112*(-248*r-129+400*s+400*r*s-
119*s^2)/(-3+s);
                                          -25/648*(4*r+3-s)*(1+s)/(-3+s),                -
25/231336*(3200*r^2+4800*r-1600*r*s+2043-714*s+443*s^2)*3^(1/2)/(-3+s),
25/77112*(476*r-129+43*s)*(1+s)/(-3+s),                                25/231336*(3200*r^2-
```

```
1557+1686*s+43*s^2)*3^(1/2)/(-3+s),                                    -25/77112*(10*r+314*r*s-357-
10*s+43*s^2)/(-3+s),    -25/231336*(1600*r^2-1200*r+400*r*s-2043+1200*s+43*s^2)*3^(1/2)/(-
3+s),                                  25/77112*(10*r+314*r*s+129+314*s-119*s^2)/(-3+s),
25/231336*(1600*r^2+3600*r-1200*r*s+1557-1200*s+443*s^2)*3^(1/2)/(-3+s);
                     25/77112*(-43+562*s+173*s^2+432*r^2)*3^(1/2)/(-3+s),
25/77112*(476*r-129+43*s)*(1+s)/(-3+s),              -25/77112*(443+238*s+227*s^2+432*r^2-
648*r+216*r*s)*3^(1/2)/(-3+s),                                                  -25/648*(4*r-
3+s)*(1+s)/(-3+s),            25/77112*(43+227*s^2+216*r^2-486*r+162*r*s-162*s)*3^(1/2)/(-
3+s),                          25/77112*(-248*r+129-400*s+400*r*s+119*s^2)/(-3+s),       -
25/77112*(-443+173*s^2+216*r^2+162*r-54*r*s+162*s)*3^(1/2)/(-3+s),                      -
25/77112*(-248*r+357-248*s+400*r*s+43*s^2)/(-3+s);
                                 25/77112*(476*r+129-43*s)*(1+s)/(-3+s),
25/231336*(3200*r^2-1557+1686*s+43*s^2)*3^(1/2)/(-3+s),
-25/648*(4*r-3+s)*(1+s)/(-3+s),                  -25/231336*(3200*r^2-4800*r+1600*r*s+2043-
714*s+443*s^2)*3^(1/2)/(-3+s),                                  25/77112*(10*r+314*r*s-129-
314*s+119*s^2)/(-3+s),                           25/231336*(1600*r^2-3600*r+1200*r*s+1557-
1200*s+443*s^2)*3^(1/2)/(-3+s),                                -25/77112*(10*r+314*r*s+357+10*s-
43*s^2)/(-3+s),  -25/231336*(1600*r^2+1200*r-400*r*s-2043+1200*s+43*s^2)*3^(1/2)/(-3+s);
      -25/77112*(-443+173*s^2+216*r^2-162*r+54*r*s+162*s)*3^(1/2)/(-3+s),
-25/77112*(10*r+314*r*s-357-10*s+43*s^2)/(-3+s),             25/77112*(43+227*s^2+216*r^2-
486*r+162*r*s-162*s)*3^(1/2)/(-3+s),                          25/77112*(10*r+314*r*s-129-
314*s+119*s^2)/(-3+s),                              -25/77112*(443-562*s+227*s^2+108*r^2-
324*r+108*r*s)*3^(1/2)/(-3+s),                                                  -25/648*(2*r-
3+s)*(-1+s)/(-3+s),                          25/77112*(-43-238*s+173*s^2+108*r^2)*3^(1/2)/(-
3+s),                                 25/77112*(238*r-129+43*s)*(-1+s)/(-3+s);
                     -25/77112*(-248*r-357+248*s+400*r*s-43*s^2)/(-3+s),                -
25/231336*(1600*r^2-1200*r+400*r*s-2043+1200*s+43*s^2)*3^(1/2)/(-3+s),
25/77112*(-248*r+129-400*s+400*r*s+119*s^2)/(-3+s),                    25/231336*(1600*r^2-
3600*r+1200*r*s+1557-1200*s+443*s^2)*3^(1/2)/(-3+s),
-25/648*(2*r-3+s)*(-1+s)/(-3+s),                    -25/231336*(800*r^2-2400*r+800*r*s+2043-
1686*s+443*s^2)*3^(1/2)/(-3+s),                                  25/77112*(238*r+129-
43*s)*(-1+s)/(-3+s),                        25/231336*(800*r^2-1557+714*s+43*s^2)*3^(1/2)/(-
3+s);
        25/77112*(43+227*s^2+216*r^2+486*r-162*r*s-162*s)*3^(1/2)/(-3+s),
25/77112*(10*r+314*r*s+129+314*s-119*s^2)/(-3+s),                                  -25/77112*(-
443+173*s^2+216*r^2+162*r-54*r*s+162*s)*3^(1/2)/(-3+s),                            -
25/77112*(10*r+314*r*s+357+10*s-43*s^2)/(-3+s),                                25/77112*(-43-
238*s+173*s^2+108*r^2)*3^(1/2)/(-3+s),
25/77112*(238*r+129-43*s)*(-1+s)/(-3+s),       -25/77112*(443-562*s+227*s^2+108*r^2+324*r-
108*r*s)*3^(1/2)/(-3+s),                                                  -25/648*(2*r+3-s)*(-
1+s)/(-3+s);
                     25/77112*(-248*r-129+400*s+400*r*s-119*s^2)/(-3+s),
25/231336*(1600*r^2+3600*r-1200*r*s+1557-1200*s+443*s^2)*3^(1/2)/(-3+s),
-25/77112*(-248*r+357-248*s+400*r*s+43*s^2)/(-3+s),   -25/231336*(1600*r^2+1200*r-400*r*s-
2043+1200*s+43*s^2)*3^(1/2)/(-3+s),                                      25/77112*(238*r-
129+43*s)*(-1+s)/(-3+s),                                               25/231336*(800*r^2-
1557+714*s+43*s^2)*3^(1/2)/(-3+s),                                                      -
25/648*(2*r+3-s)*(-1+s)/(-3+s),                   -25/231336*(800*r^2+2400*r-800*r*s+2043-
1686*s+443*s^2)*3^(1/2)/(-3+s)];

r=-3^-(1/2);
R2=[                  -25/77112*(443+238*s+227*s^2+432*r^2+648*r-216*r*s)*3^(1/2)/(-3+s),
-25/648*(4*r+3-s)*(1+s)/(-3+s),                                                  25/77112*(-
43+562*s+173*s^2+432*r^2)*3^(1/2)/(-3+s),
25/77112*(476*r+129-43*s)*(1+s)/(-3+s),                   -25/77112*(-443+173*s^2+216*r^2-
162*r+54*r*s+162*s)*3^(1/2)/(-3+s),                                  -25/77112*(-248*r-
357+248*s+400*r*s-43*s^2)/(-3+s),             25/77112*(43+227*s^2+216*r^2+486*r-162*r*s-
162*s)*3^(1/2)/(-3+s),                                 25/77112*(-248*r-129+400*s+400*r*s-
119*s^2)/(-3+s);
                                    -25/648*(4*r+3-s)*(1+s)/(-3+s),                -
25/231336*(3200*r^2+4800*r-1600*r*s+2043-714*s+443*s^2)*3^(1/2)/(-3+s),
25/77112*(476*r-129+43*s)*(1+s)/(-3+s),                                  25/231336*(3200*r^2-
1557+1686*s+43*s^2)*3^(1/2)/(-3+s),                          -25/77112*(10*r+314*r*s-357-
10*s+43*s^2)/(-3+s),    -25/231336*(1600*r^2-1200*r+400*r*s-2043+1200*s+43*s^2)*3^(1/2)/(-
3+s),                                  25/77112*(10*r+314*r*s+129+314*s-119*s^2)/(-3+s),
25/231336*(1600*r^2+3600*r-1200*r*s+1557-1200*s+443*s^2)*3^(1/2)/(-3+s);
                     25/77112*(-43+562*s+173*s^2+432*r^2)*3^(1/2)/(-3+s),
25/77112*(476*r-129+43*s)*(1+s)/(-3+s),              -25/77112*(443+238*s+227*s^2+432*r^2-
648*r+216*r*s)*3^(1/2)/(-3+s),                                                  -25/648*(4*r-
3+s)*(1+s)/(-3+s),            25/77112*(43+227*s^2+216*r^2-486*r+162*r*s-162*s)*3^(1/2)/(-
3+s),                          25/77112*(-248*r+129-400*s+400*r*s+119*s^2)/(-3+s),       -
```

```
25/77112*(-443+173*s^2+216*r^2+162*r-54*r*s+162*s)*3^(1/2)/(-3+s),                                -
25/77112*(-248*r+357-248*s+400*r*s+43*s^2)/(-3+s);
                                  25/77112*(476*r+129-43*s)*(1+s)/(-3+s),
25/231336*(3200*r^2-1557+1686*s+43*s^2)*3^(1/2)/(-3+s),
-25/648*(4*r-3+s)*(1+s)/(-3+s),                  -25/231336*(3200*r^2-4800*r+1600*r*s+2043-
714*s+443*s^2)*3^(1/2)/(-3+s),                                    25/77112*(10*r+314*r*s-129-
314*s+119*s^2)/(-3+s),                            25/231336*(1600*r^2-3600*r+1200*r*s+1557-
1200*s+443*s^2)*3^(1/2)/(-3+s),                                -25/77112*(10*r+314*r*s+357+10*s-
43*s^2)/(-3+s),  -25/231336*(1600*r^2+1200*r-400*r*s-2043+1200*s+43*s^2)*3^(1/2)/(-3+s);
      -25/77112*(-443+173*s^2+216*r^2-162*r+54*r*s+162*s)*3^(1/2)/(-3+s),
-25/77112*(10*r+314*r*s-357-10*s+43*s^2)/(-3+s),             25/77112*(43+227*s^2+216*r^2-
486*r+162*r*s-162*s)*3^(1/2)/(-3+s),                           25/77112*(10*r+314*r*s-129-
314*s+119*s^2)/(-3+s),                                  -25/77112*(443-562*s+227*s^2+108*r^2-
324*r+108*r*s)*3^(1/2)/(-3+s),                                                  -25/648*(2*r-
3+s)*(-1+s)/(-3+s),                          25/77112*(-43-238*s+173*s^2+108*r^2)*3^(1/2)/(-
3+s),                                 25/77112*(238*r-129+43*s)*(-1+s)/(-3+s);
                      -25/77112*(-248*r-357+248*s+400*r*s-43*s^2)/(-3+s),                 -
25/231336*(1600*r^2-1200*r+400*r*s-2043+1200*s+43*s^2)*3^(1/2)/(-3+s),
25/77112*(-248*r+129-400*s+400*r*s+119*s^2)/(-3+s),                    25/231336*(1600*r^2-
3600*r+1200*r*s+1557-1200*s+443*s^2)*3^(1/2)/(-3+s),
-25/648*(2*r-3+s)*(-1+s)/(-3+s),                    -25/231336*(800*r^2-2400*r+800*r*s+2043-
1686*s+443*s^2)*3^(1/2)/(-3+s),                                   25/77112*(238*r+129-
43*s)*(-1+s)/(-3+s),                          25/231336*(800*r^2-1557+714*s+43*s^2)*3^(1/2)/(-
3+s);
        25/77112*(43+227*s^2+216*r^2+486*r-162*r*s-162*s)*3^(1/2)/(-3+s),
25/77112*(10*r+314*r*s+129+314*s-119*s^2)/(-3+s),                                  -25/77112*(-
443+173*s^2+216*r^2+162*r-54*r*s+162*s)*3^(1/2)/(-3+s),                            -
25/77112*(10*r+314*r*s+357+10*s-43*s^2)/(-3+s),                                25/77112*(-43-
238*s+173*s^2+108*r^2)*3^(1/2)/(-3+s),
25/77112*(238*r+129-43*s)*(-1+s)/(-3+s),       -25/77112*(443-562*s+227*s^2+108*r^2+324*r-
108*r*s)*3^(1/2)/(-3+s),                                          -25/648*(2*r+3-s)*(-
1+s)/(-3+s);
                    25/77112*(-248*r-129+400*s+400*r*s-119*s^2)/(-3+s),
25/231336*(1600*r^2+3600*r-1200*r*s+1557-1200*s+443*s^2)*3^(1/2)/(-3+s),
-25/77112*(-248*r+357-248*s+400*r*s+43*s^2)/(-3+s),   -25/231336*(1600*r^2+1200*r-400*r*s-
2043+1200*s+43*s^2)*3^(1/2)/(-3+s),                                  25/77112*(238*r-
129+43*s)*(-1+s)/(-3+s),                                          25/231336*(800*r^2-
1557+714*s+43*s^2)*3^(1/2)/(-3+s),                                                -
25/648*(2*r+3-s)*(-1+s)/(-3+s),                  -25/231336*(800*r^2+2400*r-800*r*s+2043-
1686*s+443*s^2)*3^(1/2)/(-3+s)];

R=R1+R2

s=3^-(1/2);
S1=[                                                                                              -
25/38556*3^(1/2)*(587+238*s+227*s^2)/(-3+s),
25/324+25/324*s,
25/38556*(101+562*s+173*s^2)*3^(1/2)/(-3+s),
-1075/38556-1075/38556*s,                    -25/38556*3^(1/2)*(-371+173*s^2+162*s)/(-3+s),
25/2713137300514013184*(25121641671426049-17451448556060672*s+3025855999639552*s^2)/(-
3+s),                                          25/38556*3^(1/2)*(115+227*s^2-162*s)/(-3+s),
-25/2713137300514013184*(9077567998918657-2814749767106560O*s+837388055714201G*s^2)/(-
3+s);

25/324+25/324*s,                                                                                  -
7974004942468769/81129638414606681695789005144064*3^(1/2)*(6838229317014869-
1570102604464128*s+974167302209536*s^2)/(-3+s),
1075/38556+1075/38556*s,
7974004942468769/1106804644422573096960*(-1471+5058*s+129*s^2)*3^(1/2)/(-3+s),
-1075/38556*s-25/324,       -7974004942468769/64903710731685345356631204115251 2*3^(1/2)*(-
26558336865053355+2111062325329920O*s+756463999909888*s^2)/(-3+s),
-25/324*s-1075/38556,
7974004942468769/32451855365842672678315602057625 6*3^(1/2)*(18386766447422121-
10555311626649600*s+3896669208838144*s^2)/(-3+s);

25/38556*(101+562*s+173*s^2)*3^(1/2)/(-3+s),
1075/38556+1075/38556*s,                                                                          -
25/38556*3^(1/2)*(587+238*s+227*s^2)/(-3+s),
-25/324-25/324*s,                              25/38556*3^(1/2)*(115+227*s^2-162*s)/(-3+s),
25/2713137300514013184*(9077567998918657-2814749767106560O*s+837388055714201G*s^2)/(-
3+s),                                        -25/38556*3^(1/2)*(-371+173*s^2+162*s)/(-3+s),
```

```
-25/27131373005140131184*(25121641671426049-17451448556060672*s+3025855999639552*s^2)/(-
3+s);
                                                                                -1075/38556-
1075/38556*s,
7974004942468769/110680464442257309696*(-1471+5058*s+129*s^2)*3^(1/2)/(-3+s),
-25/324-25/324*s,                                                                          -
7974004942468769/81129638414606681695789005144064*3^(1/2)*(6838229317014869-
1570102604464128*s+974167302209536*s^2)/(-3+s),
25/324*s+1075/38556,
7974004942468769/32451855365842672678315602057625 6*3^(1/2)*(18386766447422121-
10555311626649600*s+3896669208838144*s^2)/(-3+s),
1075/38556*s+25/324,        -7974004942468769/64903710731685345356631204115 2512*3^(1/2)*(-
26558336865053355+2111062325329920 0*s+756463999909888*s^2)/(-3+s);
                                                    -25/38556*3^(1/2)*(-
371+173*s^2+162*s)/(-3+s),
-1075/38556*s-25/324,
25/38556*3^(1/2)*(115+227*s^2-162*s)/(-3+s),
25/324*s+1075/38556,                          -25/38556*3^(1/2)*(479-562*s+227*s^2)/(-3+s),
25/324-25/324*s,                                25/38556*(-7-238*s+173*s^2)*3^(1/2)/(-3+s),
-1075/38556+1075/38556*s;
     25/27131373005140131184*(25121641671426049-
17451448556060672*s+3025855999639552*s^2)/(-3+s),                                          -
7974004942468769/64903710731685345356631204115 2512*3^(1/2)*(-
26558336865053355+2111062325329920 0*s+756463999909888*s^2)/(-3+s),
25/27131373005140131184*(9077567998918657-28147497671065600*s+8373880557142016*s^2)/(-
3+s),       7974004942468769/32451855365842672678315602057625 6*3^(1/2)*(18386766447422121-
10555311626649600*s+3896669208838144*s^2)/(-3+s),                                    25/324-
25/324*s,                                                                                  -
7974004942468769/16225927682921336339157801028 8128*3^(1/2)*(10158021425146539-
7415106417721344*s+1948334604419072*s^2)/(-3+s),                                1075/38556-
1075/38556*s,
7974004942468769/110680464442257309696*(-3871+2142*s+129*s^2)*3^(1/2)/(-3+s);
                                                      25/38556*3^(1/2)*(115+227*s^2-
162*s)/(-3+s),
-25/324*s-1075/38556,                                                  -25/38556*3^(1/2)*(-
371+173*s^2+162*s)/(-3+s),
1075/38556*s+25/324,                          25/38556*(-7-238*s+173*s^2)*3^(1/2)/(-3+s),
1075/38556-1075/38556*s,                    -25/38556*3^(1/2)*(479-562*s+227*s^2)/(-3+s),
-25/324+25/324*s;
     -25/27131373005140131184*(9077567998918657-
28147497671065600*s+8373880557142016*s^2)/(-3+s),
7974004942468769/32451855365842672678315602057625 6*3^(1/2)*(18386766447422121-
10555311626649600*s+3896669208838144*s^2)/(-3+s),                                          -
25/27131373005140131184*(25121641671426049-17451448556060672*s+3025855999639552*s^2)/(-
3+s),                        -7974004942468769/64903710731685345356631204115 2512*3^(1/2)*(-
26558336865053355+2111062325329920 0*s+756463999909888*s^2)/(-3+s),
-1075/38556+1075/38556*s,
7974004942468769/110680464442257309696*(-3871+2142*s+129*s^2)*3^(1/2)/(-3+s),
-25/324+25/324*s,                                                                          -
7974004942468769/16225927682921336339157801028 8128*3^(1/2)*(10158021425146539-
7415106417721344*s+1948334604419072*s^2)/(-3+s)];


s=-3^-(1/2);
S2=[                                                                                       -
25/38556*3^(1/2)*(587+238*s+227*s^2)/(-3+s),
25/324+25/324*s,
25/38556*(101+562*s+173*s^2)*3^(1/2)/(-3+s),
-1075/38556-1075/38556*s,                    -25/38556*3^(1/2)*(-371+173*s^2+162*s)/(-3+s),
25/27131373005140131184*(25121641671426049-17451448556060672*s+3025855999639552*s^2)/(-
3+s),                                          25/38556*3^(1/2)*(115+227*s^2-162*s)/(-3+s),
-25/27131373005140131184*(9077567998918657-28147497671065600*s+8373880557142016*s^2)/(-
3+s);

25/324+25/324*s,                                                                           -
7974004942468769/81129638414606681695789005144064*3^(1/2)*(6838229317014869-
1570102604464128*s+974167302209536*s^2)/(-3+s),
1075/38556+1075/38556*s,
7974004942468769/110680464442257309696*(-1471+5058*s+129*s^2)*3^(1/2)/(-3+s),
-1075/38556*s-25/324,        -7974004942468769/64903710731685345356631204115 2512*3^(1/2)*(-
26558336865053355+2111062325329920 0*s+756463999909888*s^2)/(-3+s),
```

```
-25/324*s-1075/38556,
7974004942468769/3245185536584267267831560205762563^(1/2)*(18386766447422121-
10555311626649600*s+3896669208838144*s^2)/(-3+s);

25/38556*(101+562*s+173*s^2)*3^(1/2)/(-3+s),
1075/38556+1075/38556*s,                                                    -
25/38556*3^(1/2)*(587+238*s+227*s^2)/(-3+s),
-25/324-25/324*s,                                25/38556*3^(1/2)*(115+227*s^2-162*s)/(-3+s),
25/2713137300514013184*(9077567998918657-28147497671065600*s+83738805571420163*s^2)/(-
3+s),                                          -25/38556*3^(1/2)*(-371+173*s^2+162*s)/(-3+s),
-25/2713137300514013184*(25121641671426049-17451448556060672*s+3025855999639552*s^2)/(-
3+s);
                                                                                -1075/38556-
1075/38556*s,
7974004942468769/1106804644422573096*(-1471+5058*s+129*s^2)*3^(1/2)/(-3+s),
-25/324-25/324*s,                                                           -
7974004942468769/81129638414606681695789005144064*3^(1/2)*(6838229317014869-
1570102604464128*s+974167302209536*s^2)/(-3+s),
25/324*s+1075/38556,
7974004942468769/3245185536584267267831560205762563^(1/2)*(18386766447422121-
10555311626649600*s+3896669208838144*s^2)/(-3+s),
1075/38556*s+25/324,        -7974004942468769/64903710731685345356631204115251 2*3^(1/2)*(-
26558336865053355+2111062325329920 0*s+756463999909888*s^2)/(-3+s);
                                                    -25/38556*3^(1/2)*(-
371+173*s^2+162*s)/(-3+s),
-1075/38556*s-25/324,
25/38556*3^(1/2)*(115+227*s^2-162*s)/(-3+s),
25/324*s+1075/38556,                          -25/38556*3^(1/2)*(479-562*s+227*s^2)/(-3+s),
25/324-25/324*s,                               25/38556*(-7-238*s+173*s^2)*3^(1/2)/(-3+s),
-1075/38556+1075/38556*s;
     25/2713137300514013184*(25121641671426049-
17451448556060672*s+3025855999639552*s^2)/(-3+s),                           -
7974004942468769/64903710731685345356631204115251 2*3^(1/2)*(-
26558336865053355+2111062325329920 0*s+756463999909888*s^2)/(-3+s),
25/2713137300514013184*(9077567998918657-28147497671065600*s+83738805571420163*s^2)/(-
3+s),       7974004942468769/3245185536584267267831560205762563^(1/2)*(18386766447422121-
10555311626649600*s+3896669208838144*s^2)/(-3+s),                                  25/324-
25/324*s,                                                                   -
7974004942468769/1622592768292133633915780102881 28*3^(1/2)*(10158021425146539-
7415106417721344*s+1948334604419072*s^2)/(-3+s),                          1075/38556-
1075/38556*s,
7974004942468769/1106804644422573096*(-3871+2142*s+129*s^2)*3^(1/2)/(-3+s);
                                                    25/38556*3^(1/2)*(115+227*s^2-
162*s)/(-3+s),
-25/324*s-1075/38556,                                          -25/38556*3^(1/2)*(-
371+173*s^2+162*s)/(-3+s),
1075/38556*s+25/324,                            25/38556*(-7-238*s+173*s^2)*3^(1/2)/(-3+s),
1075/38556-1075/38556*s,                       -25/38556*3^(1/2)*(479-562*s+227*s^2)/(-3+s),
-25/324+25/324*s;
     -25/2713137300514013184*(9077567998918657-
28147497671065600*s+83738805571420163*s^2)/(-3+s),
7974004942468769/3245185536584267267831560205762563^(1/2)*(18386766447422121-
10555311626649600*s+3896669208838144*s^2)/(-3+s),                           -
25/2713137300514013184*(25121641671426049-17451448556060672*s+3025855999639552*s^2)/(-
3+s),                      -7974004942468769/64903710731685345356631204115251 2*3^(1/2)*(-
26558336865053355+2111062325329920 0*s+756463999909888*s^2)/(-3+s),
-1075/38556+1075/38556*s,
7974004942468769/1106804644422573096*(-3871+2142*s+129*s^2)*3^(1/2)/(-3+s),
-25/324+25/324*s,                                                           -
7974004942468769/1622592768292133633915780102881 28*3^(1/2)*(10158021425146539-
7415106417721344*s+1948334604419072*s^2)/(-3+s)];


G=S1+S2;

run CNTstiff

W=eval(simplify(1/E*K1));

dK=W-G;
```

```
dKn=dK/max(max(dK));

[Va,Ea]=eig(W);
[Vg,Eg]=eig(G);
```