An introduction to the finite element method using MATLAB

Donald W. Mueller, Jr
Indiana University–Purdue University Fort Wayne, Fort Wayne, IN 46805-1499, USA
E-mail: mueller@engr.ipfw.edu

Abstract This paper outlines an efficient approach to introducing the finite element method to undergraduate mechanical engineering students. This approach requires that the students have prior experience with MATLAB and a fundamental understanding of solid mechanics. Only two-dimensional beam element problems are considered, to simplify the development. The approach emphasizes an orderly solution procedure and involves important finite element concepts, such as the stiffness matrix, element and global coordinates, force equilibrium, and constraints. Two important and challenging engineering problems – a statically indeterminate beam structure and a stepped shaft – are analyzed with the systematic solution procedure and a MATLAB program. The ability of MATLAB to manipulate matrices and solve matrix equations makes the computer solution concise and easy to follow. The flexibility associated with the computer implementation allows example problems to be easily modified into design projects.

Keywords finite element method; beam elements; MATLAB

Introduction

The finite element method (FEM) is a powerful and versatile tool for practicing engineers that can be used to solve a wide variety of important engineering problems. Clearly, exposure to the FEM is beneficial to undergraduate engineering students, and engineering educators have an obligation to introduce students to modern engineering tools. What is not clear is the best specific approach to integrating the FEM into the undergraduate curriculum. The purpose of this paper is to present an efficient and effective approach to introducing the FEM to undergraduate mechanical engineering students.

Traditionally, the FEM has been introduced in a senior- or graduate-level technical elective that relies heavily on the mathematical development. Recently, different approaches to introducing students to the FEM have been proposed. The method can be introduced earlier in the curriculum and integrated throughout the student's education [1, 2]. Suggested approaches make use of commercial finite element packages such as ANSYS [1, 3], ALGOR [4, 5], COSMOS [6] and PRO/MECHANICA [7], while others suggest the use of a students-written FORTRAN code [8], internet learning modules [9] or a combination of commercial packages and student-written software [10]. Knight [11] has written a FEM primer geared for undergraduate design students that contains the software program FEPC, which solves truss, beam, and two-dimensional planar problems.

This paper outlines an efficient approach to introducing the FEM to second-semester sophomore or junior mechanical engineering students. This approach requires that the students have prior experience with MATLAB and a fundamental understanding of solid mechanics. It stresses an orderly solution procedure that involves the introduction of the element stiffness matrix, the transformation to global coordinates, and the construction of the global stiffness matrix. Only simple two-dimensional beam element problems are considered. Because of the specific type of problem considered, the required theoretical development is limited and can effectively be presented in two or three lecture periods.

The systematic solution procedure is illustrated with a statically indeterminate problem that is solved both by hand and with a MATLAB program that is provided to the students. The ability of MATLAB to manipulate matrices and solve matrix equations makes the code concise and easy to follow. Then, an important, but challenging, mechanical engineering problem – the deflection of a stepped shaft – is analyzed. The flexibility associated with the computer implementation allows the example problems to be easily modified into design projects. The approach outlined in this paper efficiently introduces the FEM, reinforces solid mechanics concepts, provides a useful analysis tool, and motivates students.

Background

An approach similar to that outlined in this paper has been used in a machine component design class at the University of Missouri–Rolla (UMR) and in 'Intermediate numerical methods for mechanical engineers' at Indiana University–Purdue University Fort Wayne (IPFW). Both UMR and IPFW offer a one-semester technical elective in the FEM that introduces students to a commercial computer package. UMR also offers a three-course sequence in the FEM at the technical elective/graduate level. The purpose of the approach outlined in this paper is to introduce the FEM to students who are unable to take these courses and to motivate interested students to study further.

The approach presented in this paper might be referred to as a matrix method to solve structural mechanics problems and is similar to approaches that are presented in a number of textbooks [e.g. 11–14]. However, these textbooks often present the development with much more detail and consider many different types of elements and cases. While this detail is useful for a full semester course, it is difficult to extract the relevant information for inclusion within an existing undergraduate course.

The software of choice for this study is MATLAB. MATLAB is a popular mathematical package that conveniently manipulates matrices and solves matrix equations. At IPFW students are introduced to MATLAB in a freshman course, and throughout this development students are assumed to be familar with the programming capabilities of MATLAB. MATLAB has also been integrated into solid mechanics textbooks [15, 16].

Overview of method

A brief overview of the approach is provided in this section. In class, this material is proceeded by a general overview of the FEM, which includes a description of nodes, elements, and the concept of domain discretization. For brevity, this material will not be included here; however, it should be noted that students can be motivated by a colorful presentation of 'real-world' problems analyzed with the FEM.

The approach outlined in this paper is to decompose a structure into a finite number of elements. Once each element is completely described with appropriate geometric and material properties, the elements are properly combined or 'assembled' to describe the entire structure. The method as outlined is similar to the approach taken by Knight [11].

Each element is modeled as an elastic member for which the displacement is linearly related to the applied loading. That is:

$$\{f\} = [k]\{u\} \tag{1}$$

where $\{f\}$ is the element load vector, $[k]$ is the element stiffness matrix, and $\{u\}$ is the element displacement vector. To indicate the specific element under consideration, a superscript will be used on the quantities in equation 1. The element or member loads can be a force, in which case the displacement is a linear translation, or a moment, in which case the displacement is an angular rotation.

In this paper, only two-dimensional beam elements connected end to end are considered. Two-dimensional beam elements are valid to model 'beam-like' structures in which the loading is in the same plane as the structure. The end-to-end connection simplifies the 'book-keeping' associated with the assembly process. These are significant limitations that should be noted. However, despite these limitations a variety of problems of interest to mechanical engineers can be considered.

To model the structure, the individual beam elements are properly combined so that the entire structure is described by:

$$\{F\} = [K]\{U\} \tag{2}$$

where $\{F\}$ is the nodal load vector, $[K]$ is the global stiffness matrix, and $\{U\}$ is the nodal displacement vector. The objective of the type of problems considered in this paper to is determine the unknown nodal displacements.

For the purposes of this paper, the FEM can be conveniently divided into five (or six) steps:

(1) construction of the element stiffness matrix in local coordinates,
(2) transformation of the element stiffness matrix into global coorindinates,
(3) assembly to the global stiffness matrix using transformed element stiffness matrices,
(4) application of the constraints to reduce the global stiffness matrix,
(5) determination of unknown nodal displacements.

An additional step that might be implemented is the post-processing of results, i.e. the determination of unknown forces or stresses from the calculated displacements.

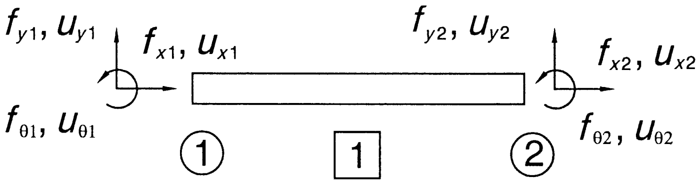

Fig. 1 *Schematic of a beam element.*

Detailed description of the method

The first step in the formulation is to discretize the domain, i.e., to select the number and type of elements. Only two-dimensional beam elements are considered in this study. A schematic of the beam element is shown in Fig. 1. Each beam element has two nodes. Two-dimensional beam elements allow for an axial force f_x, transverse force f_y, and a bending moment f_θ at each node and have three degrees of freedom (dof) per node, i.e. axial displacement u_x, transverse displacement u_y, and angular rotation u_θ. The positive sign convention is shown in Fig. 1.

The stiffness matrix for each of the beam elements is given as:

$$[k]^{(e)} = \begin{bmatrix} \frac{AE}{L} & 0 & 0 & -\frac{AE}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{AE}{L} & 0 & 0 & \frac{AE}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & -\frac{4EI}{L} \end{bmatrix} \tag{3}$$

where A is the cross-sectional area, E is the modulus of elasticity, L is the length, and I is the area moment of inertia. The superscript e is a reminder that the stiffness matrix can be different for each element. The size of the element stiffness matrix is: number of nodes × dof per node. Thus, the stiffness matrix for a two-dimensional beam element is 6 × 6 because of the three degrees of freedom associated with each node. A detailed description of this matrix is given by Logan [12] and Solecki and Conant [14].

For the purposes of this method, it is sufficient to note that this matrix describes the stiffness of an elastic element that relates the member or element load vector $\{f\} = \{f_{x1}\ f_{y1}\ f_{\theta1}\ f_{x2}\ f_{y2}\ f_{\theta2}\}^T$ to the member displacement vector $\{u\} = \{u_{x1}\ u_{y1}\ u_{\theta1}\ u_{x2}\ u_{y2}\ u_{\theta2}\}^T$, i.e.

$$\begin{Bmatrix} f_{x1} \\ f_{y1} \\ f_{\theta 1} \\ f_{x2} \\ f_{y2} \\ f_{\theta 2} \end{Bmatrix} = \begin{bmatrix} \frac{AE}{L} & 0 & 0 & -\frac{AE}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{AE}{L} & 0 & 0 & \frac{AE}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix} \begin{Bmatrix} u_{x1} \\ u_{y1} \\ u_{\theta 1} \\ u_{x2} \\ u_{y2} \\ u_{\theta 2} \end{Bmatrix} \tag{4}$$

Note that the superscript T denotes a transpose, i.e., $\{f\}$ and $\{u\}$ are actually column vectors.

To verify that this form of the stiffness matrix is correct and to illustrate several FEM concepts, consider the situation of a cantilevered beam such that node 1 in Fig. 1, is constrained from movement, i.e., $u_{x1} = u_{y1} = u_{\theta 1} = 0$. Next, consider a downward force applied at node 2, so that $f_{y2} = -F$ and $f_{x2} = f_{\theta 2} = 0$. With these considerations, the system of equations in (4) can be simplified and rewritten as:

$$\begin{Bmatrix} f_{x2} = 0 \\ f_{y2} = -F \\ f_{\theta 2} = 0 \end{Bmatrix} = \begin{bmatrix} \frac{AE}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix} \begin{Bmatrix} u_{x2} \\ u_{y2} \\ u_{\theta 2} \end{Bmatrix} \tag{5}$$

The solution to this system of equations yields $u_{x2} = 0$, $u_{y1} = -FL^3/(3EI)$, and $u_{\theta 2} = -FL^2/(EI)$, which agree with the expressions for the slope and deflection at the tip of a cantilevered beam as given in any mechanics of materials textbook. Further verification can be achieved with consideration of an applied axial load and then an applied moment.

The second step in the development is to transform the element load vector $\{f\}$ and the element displacement vector $\{u\}$ into the global coordinate system so that the individual elements can be combined in the assembly process. Consider a beam element at an arbitrary orientation described by an angle θ, measured counterclockwise from the horizontal, and a global coordinate system indicated with a bar, as shown in Fig. 2. From simple trigonometry, the loads and displacements in the global coordinate system can be related to the loads and displacements in the element coordinate system, i.e.

$$\{f\} = [\lambda]\{\bar{f}\} \quad \text{and} \quad \{u\} = [\lambda]\{\bar{u}\} \tag{6}$$

where the transformation matrix $[\lambda]$ is given by

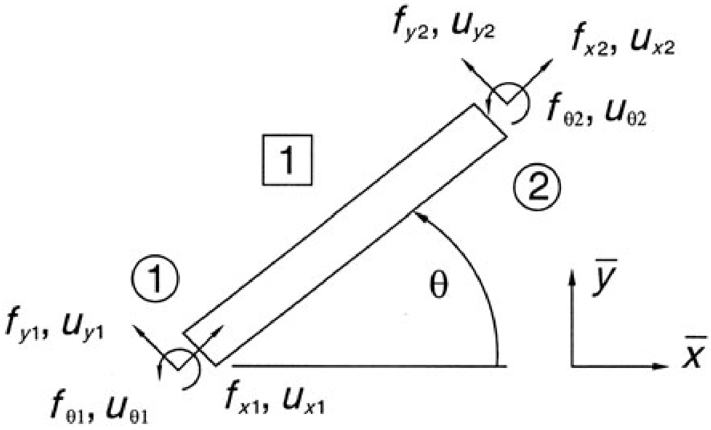

Fig. 2 *A beam element with arbitrary orientation.*

$$[\lambda] = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos\theta & \sin\theta & 0 \\ 0 & 0 & 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{7}$$

The bar in equation 6 indicates that the loads and displacements are in global coordinates.

The coordinate transformation can be incorporated in the element stiffness matrices using properties of matrix algebra. Substitution of equation 6 into equation 1 yields:

$$[\lambda]\{\bar{f}\} = [k][\lambda]\{\bar{u}\} \tag{8}$$

Then, for an orthogonal matrix $[\lambda]^{-1} = [\lambda]^T$, the element equation (equation 1) can be written in global coordinates as:

$$\{\bar{f}\} = [\bar{k}]\{\bar{u}\} \tag{9}$$

where the element stiffness matrix transformed to global coordinates is defined as:

$$[\bar{k}] = [\lambda]^T[k][\lambda] \tag{10}$$

The third step is to properly combine the individual transformed element stiffness matrices to construct the global stiffness matrix that describes the entire structure. In this study, only beam elements that are connected ‘end to end’ are considered. This is significant limitation; however, a wide variety of important problems can be solved. This step is best illustrated with a two-element example.

Consider the case of two beam elements connected ‘end to end’ as shown in Fig. 3. The global stiffness matrix $[K]$ for the two-element case is 9×9 because beam

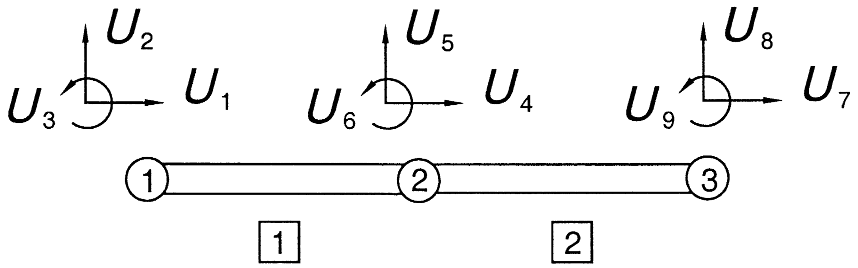

Fig. 3 *Schematic of two beam elements connected 'end to end'.*

elements have three degrees of freedom per node and there are three nodes in the structure. The external loads at the nodes are given by:

$$\{F\} = \left\{F_{x_1}\ F_{y_1}\ F_{\theta_1}\ F_{x_2}\ F_{y_2}\ F_{\theta_2}\ F_{x_3}\ F_{y_3}\ F_{\theta_3}\right\}^T$$
$$= \left\{F_1\ F_2\ F_3\ F_4\ F_5\ F_6\ F_7\ F_8\ F_9\right\}^T \tag{11}$$

which are related to the displacements at the nodes, i.e.

$$\{U\} = \left\{U_{x_1}\ U_{y_1}\ U_{\theta_1}\ U_{x_2}\ U_{y_2}\ U_{\theta_2}\ U_{x_3}\ U_{y_3}\ U_{\theta_3}\right\}^T$$
$$= \left\{U_1\ U_2\ U_3\ U_4\ U_5\ U_6\ U_7\ U_8\ U_9\right\}^T \tag{12}$$

via equation 2. The member loads in the global coordinates for the elements are given as:

$$\left\{\bar{f}\right\}^{(1)} = \left\{\bar{f}_{x1}^{(1)}\ \bar{f}_{y1}^{(1)}\ \bar{f}_{\theta1}^{(1)}\ \bar{f}_{x2}^{(1)}\ \bar{f}_{y2}^{(1)}\ \bar{f}_{\theta2}^{(1)}\right\}^T \quad \text{and}$$
$$\left\{\bar{f}\right\}^{(2)} = \left\{\bar{f}_{x2}^{(2)}\ \bar{f}_{y2}^{(2)}\ \bar{f}_{\theta2}^{(2)}\ \bar{f}_{x3}^{(2)}\ \bar{f}_{y3}^{(2)}\ \bar{f}_{\theta3}^{(2)}\right\}^T \tag{13}$$

while the element displacements in global coordinates are given as:

$$\{\bar{u}\}^{(1)} = \left\{\bar{u}_{x1}^{(1)}\ \bar{u}_{y1}^{(1)}\ \bar{u}_{\theta1}^{(1)}\ \bar{u}_{x2}^{(1)}\ \bar{u}_{y2}^{(1)}\ \bar{u}_{\theta2}^{(1)}\right\}^T \quad \text{and}$$
$$\{\bar{u}\}^{(2)} = \left\{\bar{u}_{x2}^{(2)}\ \bar{u}_{y2}^{(2)}\ \bar{u}_{\theta2}^{(2)}\ \bar{u}_{x3}^{(2)}\ \bar{u}_{y3}^{(2)}\ \bar{u}_{\theta3}^{(2)}\right\}^T \tag{14}$$

Force and moment equilibrium at each node requires that the sum of the element loads must equal the nodal loads and compatibility at each node requires that the element displacements must equal the nodal displacements, i.e.

$$\left\{\begin{array}{c} F_1 = \bar{f}_{x1}^{(1)} \\ F_2 = \bar{f}_{y1}^{(1)} \\ F_3 = \bar{f}_{\theta1}^{(1)} \\ F_4 = \bar{f}_{x2}^{(1)} + \bar{f}_{x2}^{(2)} \\ F_5 = \bar{f}_{y2}^{(1)} + \bar{f}_{y2}^{(2)} \\ F_6 = \bar{f}_{\theta2}^{(1)} + \bar{f}_{\theta2}^{(2)} \\ F_7 = \bar{f}_{x3}^{(2)} \\ F_8 = \bar{f}_{y3}^{(2)} \\ F_9 = \bar{f}_{\theta3}^{(2)} \end{array}\right\} \quad \text{and} \quad \left\{\begin{array}{c} U_1 = \bar{u}_{x1}^{(1)} \\ U_2 = \bar{u}_{y1}^{(1)} \\ U_3 = \bar{u}_{\theta1}^{(1)} \\ U_4 = \bar{u}_{x2}^{(1)} = \bar{u}_{x2}^{(2)} \\ U_5 = \bar{u}_{y2}^{(1)} = \bar{u}_{y2}^{(2)} \\ U_6 = \bar{u}_{\theta2}^{(1)} = \bar{u}_{\theta2}^{(2)} \\ U_7 = \bar{u}_{x3}^{(2)} \\ U_8 = \bar{u}_{y3}^{(2)} \\ U_9 = \bar{u}_{\theta3}^{(2)} \end{array}\right\} \tag{15}$$

Equations 9 and 15 allow the element stiffness matrices, $[\bar{k}]^{(1)}$ and $[\bar{k}]^{(2)}$, to be combined to form the global stiffness matrix $[K]$.

In order to easily combine the element stiffness matrices, each element stiffness matrix is stored in a matrix the size of the global stiffness matrix, with the extra spaces filled with zeros. In this example, the element stiffness matrix for element 1 is stored in the portion of the global stiffness matrix that involves nodes 1 and 2, i.e., the upper 6×6 portion of the matrix. Thus, the expanded stiffness matrix that describes element 1 is given by:

$$[K]^{(1)} = \begin{bmatrix} \bar{k}_{11}^{(1)} & \bar{k}_{12}^{(1)} & \bar{k}_{13}^{(1)} & \bar{k}_{14}^{(1)} & \bar{k}_{15}^{(1)} & \bar{k}_{16}^{(1)} & 0 & 0 & 0 \\ \bar{k}_{21}^{(1)} & \bar{k}_{22}^{(1)} & \bar{k}_{23}^{(1)} & \bar{k}_{24}^{(1)} & \bar{k}_{25}^{(1)} & \bar{k}_{26}^{(1)} & 0 & 0 & 0 \\ \bar{k}_{31}^{(1)} & \bar{k}_{32}^{(1)} & \bar{k}_{33}^{(1)} & \bar{k}_{34}^{(1)} & \bar{k}_{35}^{(1)} & \bar{k}_{36}^{(1)} & 0 & 0 & 0 \\ \bar{k}_{41}^{(1)} & \bar{k}_{42}^{(1)} & \bar{k}_{43}^{(1)} & \bar{k}_{44}^{(1)} & \bar{k}_{45}^{(1)} & \bar{k}_{46}^{(1)} & 0 & 0 & 0 \\ \bar{k}_{51}^{(1)} & \bar{k}_{52}^{(1)} & \bar{k}_{53}^{(1)} & \bar{k}_{54}^{(1)} & \bar{k}_{55}^{(1)} & \bar{k}_{56}^{(1)} & 0 & 0 & 0 \\ \bar{k}_{61}^{(1)} & \bar{k}_{62}^{(1)} & \bar{k}_{63}^{(1)} & \bar{k}_{64}^{(1)} & \bar{k}_{65}^{(1)} & \bar{k}_{66}^{(1)} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{16}$$

The element stiffness matrix for element 2 is stored in the portion of the global stiffness matrix that involves nodes 2 and 3, i.e., the lower 6×6 portion of the matrix. The expanded stiffness matrix that describes element 2 is given by:

$$[K]^{(2)} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \bar{k}_{11}^{(2)} & \bar{k}_{12}^{(2)} & \bar{k}_{13}^{(2)} & \bar{k}_{14}^{(2)} & \bar{k}_{15}^{(2)} & \bar{k}_{16}^{(2)} \\ 0 & 0 & 0 & \bar{k}_{21}^{(2)} & \bar{k}_{22}^{(2)} & \bar{k}_{23}^{(2)} & \bar{k}_{24}^{(2)} & \bar{k}_{25}^{(2)} & \bar{k}_{26}^{(2)} \\ 0 & 0 & 0 & \bar{k}_{31}^{(2)} & \bar{k}_{32}^{(2)} & \bar{k}_{33}^{(2)} & \bar{k}_{34}^{(2)} & \bar{k}_{35}^{(2)} & \bar{k}_{36}^{(2)} \\ 0 & 0 & 0 & \bar{k}_{41}^{(2)} & \bar{k}_{42}^{(2)} & \bar{k}_{43}^{(2)} & \bar{k}_{44}^{(2)} & \bar{k}_{45}^{(2)} & \bar{k}_{46}^{(2)} \\ 0 & 0 & 0 & \bar{k}_{51}^{(2)} & \bar{k}_{52}^{(2)} & \bar{k}_{53}^{(2)} & \bar{k}_{54}^{(2)} & \bar{k}_{55}^{(2)} & \bar{k}_{56}^{(2)} \\ 0 & 0 & 0 & \bar{k}_{61}^{(2)} & \bar{k}_{62}^{(2)} & \bar{k}_{63}^{(2)} & \bar{k}_{64}^{(2)} & \bar{k}_{65}^{(2)} & \bar{k}_{66}^{(2)} \end{bmatrix} \tag{17}$$

If the structure contains additional elements, the stiffness matrices are stored in the proper portion of the matrix, i.e. each matrix is shifted by three rows and three columns. That fact that only end-to-end connections are allowed simplifies the storage of these matrices and eliminates the complicated book-keeping often associated with the FEM.

The expanded individual stiffness matrices in equations 16 and 17 can now be added together so that the global stiffness matrix for the two-element structure is given as:

$$[K]=\begin{bmatrix} \bar{k}_{11}^{(1)} & \bar{k}_{12}^{(1)} & \bar{k}_{13}^{(1)} & \bar{k}_{14}^{(1)} & \bar{k}_{15}^{(1)} & \bar{k}_{16}^{(1)} & 0 & 0 & 0 \\ \bar{k}_{21}^{(1)} & \bar{k}_{22}^{(1)} & \bar{k}_{23}^{(1)} & \bar{k}_{24}^{(1)} & \bar{k}_{25}^{(1)} & \bar{k}_{26}^{(1)} & 0 & 0 & 0 \\ \bar{k}_{31}^{(1)} & \bar{k}_{32}^{(1)} & \bar{k}_{33}^{(1)} & \bar{k}_{34}^{(1)} & \bar{k}_{35}^{(1)} & \bar{k}_{36}^{(1)} & 0 & 0 & 0 \\ \bar{k}_{41}^{(1)} & \bar{k}_{42}^{(1)} & \bar{k}_{43}^{(1)} & \bar{k}_{44}^{(1)}+\bar{k}_{11}^{(2)} & \bar{k}_{45}^{(1)}+\bar{k}_{12}^{(2)} & \bar{k}_{46}^{(1)}+\bar{k}_{13}^{(2)} & \bar{k}_{14}^{(2)} & \bar{k}_{15}^{(2)} & \bar{k}_{16}^{(2)} \\ \bar{k}_{51}^{(1)} & \bar{k}_{52}^{(1)} & \bar{k}_{53}^{(1)} & \bar{k}_{54}^{(1)}+\bar{k}_{21}^{(2)} & \bar{k}_{55}^{(1)}+\bar{k}_{22}^{(2)} & \bar{k}_{56}^{(1)}+\bar{k}_{23}^{(3)} & \bar{k}_{24}^{(2)} & \bar{k}_{25}^{(2)} & \bar{k}_{26}^{(2)} \\ \bar{k}_{61}^{(1)} & \bar{k}_{62}^{(1)} & \bar{k}_{63}^{(1)} & \bar{k}_{64}^{(1)}+\bar{k}_{31}^{(2)} & \bar{k}_{65}^{(1)}+\bar{k}_{32}^{(2)} & \bar{k}_{66}^{(1)}+\bar{k}_{33}^{(2)} & \bar{k}_{34}^{(2)} & \bar{k}_{35}^{(2)} & \bar{k}_{36}^{(2)} \\ 0 & 0 & 0 & \bar{k}_{41}^{(2)} & \bar{k}_{42}^{(2)} & \bar{k}_{43}^{(2)} & \bar{k}_{44}^{(2)} & \bar{k}_{45}^{(2)} & \bar{k}_{46}^{(2)} \\ 0 & 0 & 0 & \bar{k}_{51}^{(2)} & \bar{k}_{52}^{(2)} & \bar{k}_{53}^{(2)} & \bar{k}_{54}^{(2)} & \bar{k}_{55}^{(2)} & \bar{k}_{56}^{(2)} \\ 0 & 0 & 0 & \bar{k}_{61}^{(2)} & \bar{k}_{62}^{(2)} & \bar{k}_{63}^{(2)} & \bar{k}_{64}^{(2)} & \bar{k}_{65}^{(2)} & \bar{k}_{66}^{(2)} \end{bmatrix} \tag{18}$$

This matrix is of block-diagonal form and symmetric with respect to the diagonal. Extension to structures with more elements is performed in a similar manner and the matrix has a similar block-diagonal form. At this point in the development, the global stiffness matrix $[K]$ is singular, i.e., the inverse does not exist and there is no unique solution to the problem. The physical reason why the problem has no solution at this point is that any applied loading will cause rigid body motion – the structure must be constrained so that the applied loading causes the structure to deform.

The fourth step is to apply the constraints and reduce the global stiffness matrix so that the specific problem of interest can be solved. At the point of constraint, the displacement of the structure is known. Because these displacements are known (and in this paper assumed to be zero), matrix algebra allows the removal of the corresponding rows and columns, i.e. the global stiffness matrix can be reduced. The resulting system of equations can be written as:

$$\{F\}_r=[K]_r\{U\}_r \tag{19}$$

$[K]_r$ is the reduced global stiffness matrix that contains information about the structure and the boundary conditions. With MATLAB, rows and columns can be easily deleted, and a shift of the remaining elements in the matrix is performed automatically. This step will be illustrated in the example problem below.

The final step is simply to solve the reduced system of equations for the unknown displacements. Formally, the solution can be written in terms of an inverse:

$$\{U\}_r=[K]_r^{-1}\{F\}_r \tag{20}$$

although in practice the inverse is seldom computed. MATLAB efficiently solves a system of equations with the backslash command. The backslash command uses Cholesky factorization and Gaussian elimination to solve equation 19.

Example problem 1: two beam elements

Problem statement

A simple structure made of two beams is shown in Fig. 4. The following parameters are given: L_1 = 12 in, L_2 = 18 in, and $E = 30 \times 10^6$ psi. The cross-section of the

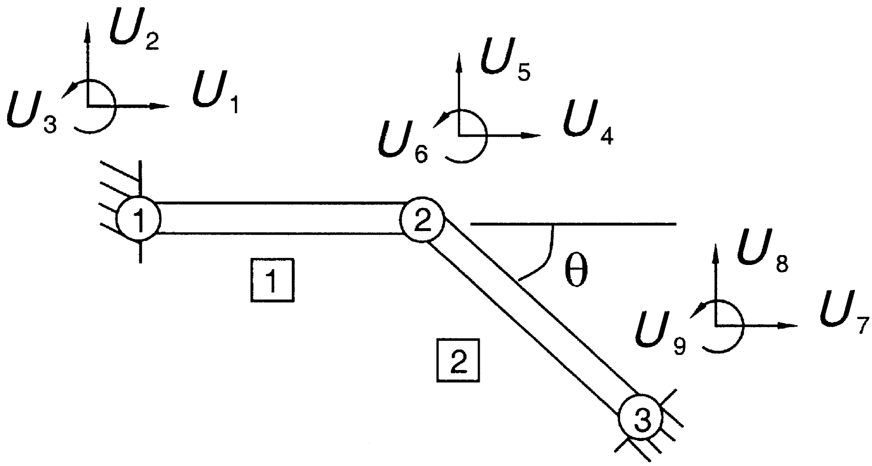

Fig. 4 *Schematic of the two-beam problem.*

beams is 0.5 in × 0.5 in and $\theta = 45°$. If a downward vertical force of 1000 lb is applied at node 2, find the displacement at node 2.

Solution

This problem is solved with the MATLAB code provided in Appendix A. The geometric parameters, material properties, and applied forces are input.

The first step in the solution procedure is to discretize the domain, i.e. select the number of elements. As a first approximation, the problem is modeled with two beam elements. The element stiffness matrix for each of the beam elements is of the form given in equation 3.

The second step is to transform each element stiffness matrix in local coordinates to the global coordinate system. This transformation is accomplished by equation 10. The transformation angle for element 1 is $\theta = 0$, and, thus, the transformation matrix is simply the identity matrix. Because the rotation angle is measured counterclockwise, the transformation angle for element 2 is $\theta = -45°$.

The third step is to assemble the global stiffness matrix that describes the entire structure by properly combining the individual element stiffness matrices. For the case of two elements, as shown in Fig. 3, the global stiffness matrix is of the form given in equation 10.

The fourth step is to apply the constraints and reduce the global stiffness matrix so that the specific problem of interest can be solved. For this problem, the displacements at node 1 and node 3 are known, i.e. $U_1 = U_2 = U_3 = U_7 = U_8 = U_9 = 0$, and the loads are applied to node 2 are specified, i.e. $F_4 = 0$, $F_5 = -1000\,\text{lb}$, and $F_6 = 0$. Thus, the system of equations can be written as:

$$\begin{Bmatrix} F_1=? \\ F_2=? \\ F_3=? \\ F_4=0 \\ F_5=-1000 \\ F_6=0 \\ F_7=? \\ F_8=? \\ F_9=? \end{Bmatrix} = \begin{bmatrix} \begin{bmatrix} \bar{k}_{11}^{(1)} & \bar{k}_{12}^{(1)} & \bar{k}_{13}^{(1)} & \bar{k}_{14}^{(1)} & \bar{k}_{15}^{(1)} & \bar{k}_{16}^{(1)} & 0 & 0 & 0 \\ \bar{k}_{21}^{(1)} & \bar{k}_{22}^{(1)} & \bar{k}_{23}^{(1)} & \bar{k}_{24}^{(1)} & \bar{k}_{25}^{(1)} & \bar{k}_{26}^{(1)} & 0 & 0 & 0 \\ \bar{k}_{31}^{(1)} & \bar{k}_{32}^{(1)} & \bar{k}_{33}^{(1)} & \bar{k}_{34}^{(1)} & \bar{k}_{35}^{(1)} & \bar{k}_{36}^{(1)} & 0 & 0 & 0 \\ \bar{k}_{41}^{(1)} & \bar{k}_{42}^{(1)} & \bar{k}_{43}^{(1)} & \bar{k}_{44}^{(1)}+\bar{k}_{11}^{(2)} & \bar{k}_{45}^{(1)}+\bar{k}_{12}^{(2)} & \bar{k}_{46}^{(1)}+\bar{k}_{13}^{(2)} & \bar{k}_{14}^{(2)} & \bar{k}_{15}^{(2)} & \bar{k}_{16}^{(2)} \\ \bar{k}_{51}^{(1)} & \bar{k}_{52}^{(1)} & \bar{k}_{53}^{(1)} & \bar{k}_{54}^{(1)}+\bar{k}_{21}^{(2)} & \bar{k}_{55}^{(1)}+\bar{k}_{22}^{(2)} & \bar{k}_{56}^{(1)}+\bar{k}_{23}^{(2)} & \bar{k}_{24}^{(2)} & \bar{k}_{25}^{(2)} & \bar{k}_{26}^{(2)} \\ \bar{k}_{61}^{(1)} & \bar{k}_{62}^{(1)} & \bar{k}_{63}^{(1)} & \bar{k}_{64}^{(1)}+\bar{k}_{31}^{(2)} & \bar{k}_{65}^{(1)}+\bar{k}_{32}^{(2)} & \bar{k}_{66}^{(1)}+\bar{k}_{33}^{(2)} & \bar{k}_{34}^{(2)} & \bar{k}_{35}^{(2)} & \bar{k}_{36}^{(2)} \\ 0 & 0 & 0 & \bar{k}_{41}^{(2)} & \bar{k}_{42}^{(2)} & \bar{k}_{43}^{(2)} & \bar{k}_{44}^{(2)} & \bar{k}_{45}^{(2)} & \bar{k}_{46}^{(2)} \\ 0 & 0 & 0 & \bar{k}_{51}^{(2)} & \bar{k}_{52}^{(2)} & \bar{k}_{53}^{(2)} & \bar{k}_{54}^{(2)} & \bar{k}_{55}^{(2)} & \bar{k}_{56}^{(2)} \\ 0 & 0 & 0 & \bar{k}_{61}^{(2)} & \bar{k}_{62}^{(2)} & \bar{k}_{63}^{(2)} & \bar{k}_{64}^{(2)} & \bar{k}_{65}^{(2)} & \bar{k}_{66}^{(2)} \end{bmatrix} \end{bmatrix} \begin{Bmatrix} U_1=0 \\ U_2=0 \\ U_3=0 \\ U_4=? \\ U_5=? \\ U_6=? \\ U_7=0 \\ U_8=0 \\ U_9=0 \end{Bmatrix} \tag{21}$$

This problem has three unknowns, U_4, U_5, and U_6, and, thus, requires three independent equations. Matrix algebra allows three independent equations to be constructed by the removal of the rows and columns that correspond to the known displacements, i.e. the global stiffness matrix reduces to:

$$[K]_r = \begin{bmatrix} \bar{k}_{44}^{(1)}+\bar{k}_{11}^{(2)} & \bar{k}_{45}^{(1)}+\bar{k}_{12}^{(2)} & \bar{k}_{46}^{(1)}+\bar{k}_{13}^{(2)} \\ \bar{k}_{54}^{(1)}+\bar{k}_{21}^{(2)} & \bar{k}_{55}^{(1)}+\bar{k}_{22}^{(2)} & \bar{k}_{56}^{(1)}+\bar{k}_{23}^{(2)} \\ \bar{k}_{64}^{(1)}+\bar{k}_{31}^{(2)} & \bar{k}_{65}^{(1)}+\bar{k}_{32}^{(2)} & \bar{k}_{66}^{(1)}+\bar{k}_{33}^{(2)} \end{bmatrix} \tag{22}$$

This is the form of the reduced global stiffness matrix that contains information about the structure and the boundary conditions. The numerical values from the code in Appendix A are given below:

$$[K]_r = 1\times 10^5 \begin{bmatrix} 8.3349 & -2.0817 & 0.0205 \\ -2.0817 & 2.0958 & -0.0446 \\ 0.0205 & -0.0446 & 0.8681 \end{bmatrix} \tag{23}$$

Like the element stiffness matrix and the global stiffness matrix, the reduced stiffness matrix is also symmetric with respect to the diagonal. Numerical values from the code will be printed to the screen when the semicolon at the end of the line is removed.

The final step is simply to solve the reduced system of equations for the unknown displacements. The numerical solution from the MATLAB code in Appendix A is $U_4 = -0.0016$ in, $U_5 = -0.0064$ in, and $U_6 = -0.0003$ radians. Once all the displacements are known, the reactions at the wall can be found with $\{F\} = [K]\{U\}$.

Discussion

This problem is statically indeterminate, and the solution is not trivial. Additional information can be found with the addition of nodes. For example, the deflection at the midpoint of each element can be found if two additional nodes are added. Of course, then a 9 × 9 system of equations must be solved. Note that the addition of nodes at the midpoints does not change the solution at node 2.

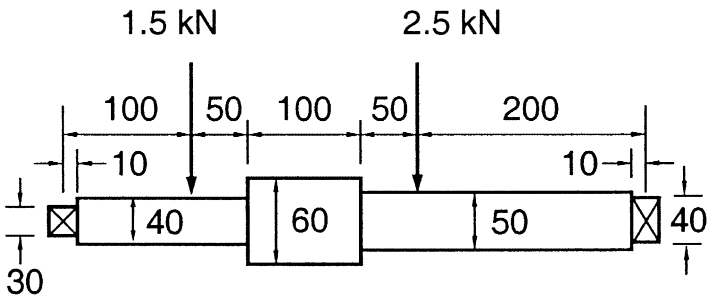

Fig. 5 *Schematic of the shaft (dimensions in mm).*

Example problem 2: stepped shaft

Problem statement

A simply supported stepped shaft is shown in Fig. 5. The shaft is made of steel with $E = 207 \times 10^9$ Pa. Find the vertical displacement along the shaft.

Solution

This problem is solved with the MATLAB code provided in Appendix B. The geometric parameters, material properties, and applied forces are input.

The first step in the solution procedure is to discretize the domain, i.e. select the number of elements. As a first approximation, this shaft might be modeled with seven beam elements. A node is required at all locations where 'something is happening'. In this case, nodes are required where the cross-section changes or a force is applied. The element stiffness matrix for each of the beam elements is given in equation 3.

The second step is to transform each element stiffness matrix in local coordinates to the global coordinate system with the use of equation 10. In this problem, each element is horizontal, i.e. $\theta = 0$, and the transformation matrix is simply the identity matrix.

The third step is to assemble the global stiffness matrix that describes the entire structure by properly combining the individual element stiffness matrices. A block-diagonal matrix similar in form to that given in equation 18 results. However, for the case of seven elements (eight nodes), the global stiffness matrix is 24×24.

The fourth step is to apply the constraints and reduce the global stiffness matrix so that the specific problem of interest can be solved. For this problem, simple supports are located at each end. Thus, the vertical displacement at each end is zero. It is also necessary to constrain the shaft in the horizontal direction. This constraint can be applied at either end. Thus, for this problem the constraints are $U_2 = U_{22} = U_{23} = 0$, and the reduced global stiffness matrix is 21×21. Vertical forces are applied

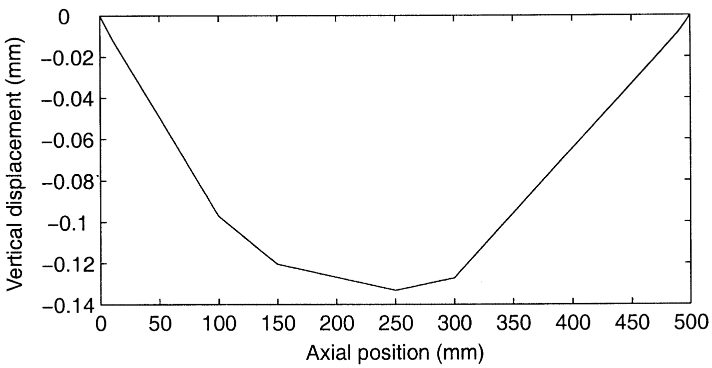

Fig. 6 *Deflection of the shaft.*

to the structure at nodes 3 and 6. Following the notation in the paper, $F_8 = -1.5\,\text{kN}$ and $F_{17} = -2.5\,\text{kN}$, while all other applied loads are zero.

The reduced system of equations is solved using MATLAB. The deflection of the shaft can be shown graphically, as in Fig. 6, with the addition of the following to the end of the code in Appendix B:

```
x = [0 10 100 150 250 300 490 500];
y = 1000*1000*[0 u(4) u(7) u(10) u(13) u(16) u(19)
    0];
plot(x, y)
```

The reaction forces F_2 and F_{23} are unknown but can be found with the FEM once all the nodal displacements are known. (In this problem, the reaction forces can also be found with the use of statics.)

Discussion

This problem is similar to sample problem 5.3 in Juvinall and Marshek [17]. The solution procedure presented in most machine design textbooks [e.g. 17] is the double integration method, which is tedious to carry out by hand. With the FEM and the code in Appendix B, this example can easily be extended to a design problem in which the students are required to limit the deflection of the shaft to a specified value by changing the location of the forces and/or the diameter of shaft. Another quantity of interest is the slope of the shaft at the bearing supports, i.e. U_3 and U_{24}.

Closing comments

An efficient, effective approach to introducing the FEM has been outlined. The approach involves an orderly solution procedure based on stiffness matrices and matrix algebra. The method makes use of a MATLAB code that must be modified

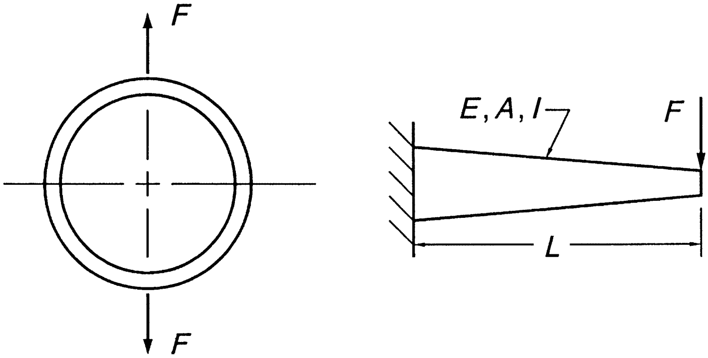

Fig. 7 *Schematics of other projects.*

by the student. At this stage, no attempt has been made to optimize the code with respect to speed or storage.

Although this approach is limited to beam elements connected end to end, many other interesting problems can be analyzed. Examples of projects that have been used to study domain discretization and convergence are shown in Fig. 7. The approach outlined is useful to introduce students to FEM terminology, provide students with a means to solve important problems in solid mechanics, and to motivate students for future study.

Appendix A. MATLAB code for two-beam problem

```
function beam2
%
% Step 0: Input constants
%
m = 2;                        % number of elements
b = [0.5 0.5];                % width (in)
h = [0.5 0.5];                % height (in)
a = b.*h;                     % cross-sectional area
I = b.*h.^3./12;              % moment of inertia
l = [12 18];                  % length of each element (in)
e = [30*10^6 30*10^6];        % modulus of elasticity (psi)
theta = [0 -45*pi/180];       % orientation angle
f = -1000;                    % force (lbs)
%
% Step 1: Construct element stiffness matrix
%
k = zeros(6,6,m);
for n = 1:m
 k11 = a(n)*e(n)/l(n); k22 = 12*e(n)*I(n)/l(n)^3;
```

```
 k23 = 6*e(n)*I(n)/l(n)^2; k33= 4*e(n)*I(n)/l(n);
 k36 = 2*e(n)*I(n)/l(n);
 k(:,:,n)   =     [k11  0    0     -k11    0     0;
      0     k22   k23   0    -k22   k23;
      0     k23   k33   0    -k23   k36;
      -k11  0     0     k11  0      0;
      0     -k22  -k23  0    k22    -k23;
      0     k23   k36   0    -k23   k33];
end
%
% Step 2: Transform element stiffness matrices to global
coordinates
%
shift = 0;
kbar = zeros(6,6,m); Ke = zeros(9,9,m);
for n = 1:m
 c = cos(theta(n)); s = sin(theta(n));
 lamda = [c s 0 0 0 0; -s c 0 0 0 0; 0 0 1 0 0 0;
  0 0 0 c s 0; 0 0 0 -s c 0; 0 0 0 0 0 1];
 kbar (:,:,n) = lamda'*k(:,:,n)*lamda;
%
% Step 3: Combine element stiffness matrices to form
global stiffness matrix
%
 for i = 1:6
  for j = 1:6
  Ke(i+shift,j+shift,n) = kbar(i,j,n);
  end
 end
 shift = shift + 3;
end
K = sum(Ke,3); Kr = K;
%
% Step 4: Reduce global stiffness matrix with constraints
%
Kr(:,9) = []; Kr(9,:) = []; Kr(:,8) = []; Kr(8,:) = [];
Kr(:,7) = []; Kr(7,:) = [];
Kr(:,3) = []; Kr(3,:) = []; Kr(:,2) = []; Kr(2,:) = [];
Kr(:,1) = []; Kr(1,:) = []
%
% Step 5: Solve for unknown displacements
%
Fr = [0; f; 0];
Ur = Kr\Fr
%
```

```
% Step 6: Solve for forces
%
U = [0;0;0; Ur; 0;0;0];
F = K*U
```

Appendix B. MATLAB code for stepped-shaft problem

```
function shaft
%
% Step 0: Input constants
%
m = 7;                              % number of elements
nodes = m + 1;                      % number of nodes
dof = 3*nodes;                      % degrees of freedom
d = [30 40 40 60 50 50 40];         % diameter (mm)
a = pi*d.^2/4;                      % cross-sectional area
I = pi*d.^4./64;                    % moment of inertia
l = [10 90 50 100 50 190 10];       % length of each element
                                      (mm)
e = 207e9*ones(m);                  % modulus of elasticity
                                      (Pa)
theta = [0 0 0 0 0 0 0];            % orientation angle
f1 = -1.5e3; f2 = -2.5e3;           % forces (N)
%
% Step 1: Construct element stiffness matrix
%
k = zeros(6,6,m);
for n = 1:m
 k11 = a(n)*e(n)/l(n); k22 = 12*e(n)*I(n)/l(n)^3;
 k23 = 6*e(n)*I(n)/l(n)^2; k33= 4*e(n)*I(n)/l(n);
 k36 = 2*e(n)*I(n)/l(n);
 k(:,:,n)   =     [k11  0     0      -k11   0    0;
       0     k22    k23   0     -k22    k23;
       0     k23    k33   0     -k23    k36;
       -k11  0      0     k11   0       0;
       0     -k22   -k23  0     k22     -k23;
       0     k23    k36   0     -k23    k33];
end
%
% Step 2: Transform element stiffness matrices to global
coordinates
%
shift = 0;
kbar = zeros(6,6,m); Ke = zeros(dof,dof,m);
for n = 1:m
```

```
 c = cos(theta(n)); s = sin(theta(n));
 lamda = [c s 0 0 0 0; -s c 0 0 0 0; 0 0 1 0 0 0;
  0 0 0 c s 0; 0 0 0 -s c 0; 0 0 0 0 0 1];
 kbar(:,:,n) = lamda'*k(:,:,n)*lamda;
%
% Step 3: Combine element stiffness matrices to form
global stiffness matrix
%
 for i = 1:6
  for j = 1:6
   Ke(i+shift,j+shift,n) = kbar(i,j,n);
  end
 end
 shift = shift + 3;
end
K = sum(Ke,3); Kr = K;
%
% Step 4: Reduce global stiffness matrix with constraints
%
Kr(:,dof-1) = []; Kr(dof-1,:) = [];
Kr(:,dof-2) = []; Kr(dof-2,:) = [];
Kr(:,2) = []; Kr(2,:) = [];
%
% Step 5: Solve for unknown displacements
%
Fr = [0;0; 0;0;0; 0;f1;0; 0;0;0; 0;0;0; 0;f2;0 0;0;0;
0];
Ur = Kr\Fr
%
% Step 6: Solve for forces
%
U = [Ur(1); 0; Ur(2:20); 0; 0; Ur(21)];
F = K*U
```

References

[1] R. D. Earley, 'Use of FEA in an introductory strength of materials course', *ASEE Annual Conference Proceedings*, Session **3648** (1998).

[2] M. F. Coyle and C. G. Keel, 'Teaching finite element analysis to second year students', *ASEE Annual Conference Proceedings*, Session **2793** (2001).

[3] J. R. Backer, V. R. Capece and R. J. Lee, 'Integration of finite element software in undergraduate engineering courses', *ASEE Annual Conference Proceedings*, Session **1520** (2001).

[4] W. E. Howard, J. C. Musto and V. Prantil, 'Finite element analysis in a mechanics course sequence', *ASEE Annual Conference Proceedings*, Session **2793** (2001).

[5] L. J. Logue and K. A. Hall, 'Introducing finite element analysis in an MET strength of material course', *ASEE Annual Conference Proceedings*, Session **3248** (2001).

[6] M. Pike, 'Introducing finite element analysis in statics', *ASEE Annual Conference Proceedings*, Session **2268** (2001).
[7] C. J. Lissenden, G. S. Wagle and N. J. Salamon, 'Applications of finite element analysis for undergraduates', *ASEE Annual Conference Proceedings*, Session **3568** (2002).
[8] P. J. Cronin, 'A plane stress FEA problem for which students can write a modest computer program', *ASEE Annual Conference Proceedings*, Session **1447** (2002).
[9] J. Rencis, *et al*., 'Learning modules for finite element method on the world wide web', *ASEE Annual Conference Proceedings*, Session **2520** (1999).
[10] D. Jensen and E. Pramondo, 'Method for teaching finite elements which combines the advantages of commercial pre- and post-processing with student-written software', *Computer Applications in Engineering Education*, **6** (1998), 105–114.
[11] C. E. Knight, *A Finite Element Method Primer for Mechanical Design* (PWS Publishing, Boston, 1994).
[12] D. L. Logan, *A First Course in the Finite Element Method* (PWS-Kent, Boston, 1986).
[13] L. P. Felton and R. B. Nelson, *Matrix Structural Analysis* (Wiley, New York, 1997).
[14] R. Solecki and R. J. Conant, *Advanced Mechanics of Materials* (Oxford University Press, Oxford, 2003).
[15] Y. W. Kwon and H. Y. Bang, *The Finite Element Method Using MATLAB* (CRC Press, Boca Raton, FL, 2000).
[16] H. Wilson, L. H. Turcotte and D. Halpern, *Advanced Mathematics and Mechanics Applications Using MATLAB* (3rd edn) (CRC Press, Boca Raton, FL, 2002).
[17] R. C. Juvinall and K. M. Marshek, *Fundamentals of Machine Component Design* (Wiley, New York, 2000).