20

Implementation of One-Dimensional Elements

TABLE OF CONTENTS

This Chapter begins Part III of the course. This Part deals with the computer implementation of the Finite Element Method for static analysis. It is organized in "bottom up" fashion. It begins with simple topics, such as programming of bar and beam elements, and gradually builds up toward more complex models and calculations.

Specific examples of this Chapter illustrate the programming of one-dimensional elements: bars and beams, using *Mathematica* as implementation language.

§20.1. The Plane Bar Element

The two-node, prismatic, two-dimensional bar element was studied in Chapters 2-3 for modeling plane trusses. It is reproduced in Figure 20.1 for conveniency. It has two nodes and four degrees of freedom. The element node displacements and conjugate forces are

$$\mathbf{u}^e = \begin{bmatrix} u_{x1} \\ u_{y1} \\ u_{x2} \\ u_{y2} \end{bmatrix}, \quad \mathbf{f}^e = \begin{bmatrix} f_{x1} \\ f_{y1} \\ f_{x2} \\ f_{y2} \end{bmatrix}. \quad (20.1)$$
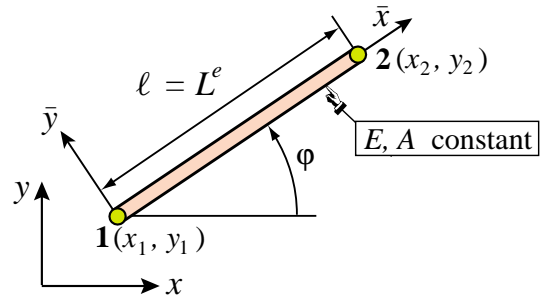

FIGURE 20.1. Plane bar element.

The element geometry is described by the coordinates $\{x_i, y_i\}$, $i = 1, 2$ of the two end nodes. For stiffness computations, the only material and fabrication properties required are the modulus of elasticity $E = E^e$ and the cross section area $A = A^e$, respectively. Both are taken to be constant over the element.

§20.1.1. Stiffness Matrix

The element stiffness matrix in global $\{x, y\}$ coordinates is given by the explicit expression derived in §3.1:

$$\mathbf{K}^e = \frac{EA}{\ell} \begin{bmatrix} c^2 & sc & -c^2 & -sc \\ sc & s^2 & -sc & -s^2 \\ -c^2 & -sc & c^2 & sc \\ -sc & -s^2 & sc & s^2 \end{bmatrix} = \frac{EA}{\ell^3} \begin{bmatrix} x_{21}x_{21} & x_{21}y_{21} & -x_{21}x_{21} & -x_{21}y_{21} \\ x_{21}y_{21} & y_{21}y_{21} & -x_{21}y_{21} & -y_{21}y_{21} \\ -x_{21}x_{21} & -x_{21}y_{21} & x_{21}x_{21} & x_{21}y_{21} \\ -x_{21}y_{21} & -y_{21}y_{21} & x_{21}y_{21} & y_{21}y_{21} \end{bmatrix}. \quad (20.2)$$

Here $c = \cos\varphi = x_{21}/\ell$, $s = \sin\varphi = y_{21}/\ell$, in which $x_{21} = x_2 - x_1$, $y_{21} = y_2 - y_1$, $\ell = \sqrt{x_{21}^2 + y_{21}^2}$, and φ is the angle formed by $\bar{x}$ and x, measured from x positive counterclockwise (see Figure 20.1). The second expression in (20.2) is preferable in a computer algebra system because it enhances simplification possibilities when doing symbolic work, and is the one actually implemented in the module described below.

§20.1.2. Stiffness Module

The computation of the stiffness matrix $\mathbf{K}^e$ of the two-node, prismatic plane bar element is done by *Mathematica* module PlaneBar2Stiffness. This is listed in Figure 20.2. The module is invoked as

            Ke = PlaneBar2Stiffness[ncoor, Em, A, options]   (20.3)

```
PlaneBar2Stiffness[ncoor_,Em_,A_,options_]:= Module[
 {x1,x2,y1,y2,x21,y21,EA,numer,L,LL,LLL,Ke},
  {{x1,y1},{x2,y2}}=ncoor; {x21,y21}={x2-x1,y2-y1};
  EA=Em*A; {numer}=options; LL=x21^2+y21^2; L=Sqrt[LL];
  If [numer,{x21,y21,EA,LL,L}=N[{x21,y21,EA,LL,L}]];
  If [!numer, L=PowerExpand[L]]; LLL=Simplify[LL*L];
  Ke=(Em*A/LLL)*{{ x21*x21, x21*y21,-x21*x21,-x21*y21},
                { y21*x21, y21*y21,-y21*x21,-y21*y21},
                {-x21*x21,-x21*y21, x21*x21, x21*y21},
                {-y21*x21,-y21*y21, y21*x21, y21*y21}};
  Return[Ke]];
```

FIGURE 20.2. *Mathematica* stiffness module for a two-node, prismatic plane bar element.

The arguments are

ncoor	Node coordinates of element arranged as { { x1,y1 },{ x2,y2 } }.
Em	Elastic modulus.
A	Cross section area.
options	A list of processing options. For this element is has only one entry: { numer }. This is a logical flag with the value True or False. If True the computations are carried out in floating-point arithmetic. If False symbolic processing is assumed.

The module returns the 4×4 element stiffness matrix as function value.

```
ClearAll[A,Em,L];
ncoor={{0,0},{30,40}}; Em=1000; A=5;
Ke= PlaneBar2Stiffness[ncoor,Em,A,{True}];
Print["Numerical Elem Stiff Matrix: "];
Print[Ke//MatrixForm];
Print["Eigenvalues of Ke=",Chop[Eigenvalues[N[Ke]]]];
Print["Symmetry check=",Simplify[Chop[Transpose[Ke]-Ke]]];
```

Numerical Elem Stiff Matrix:

$$\begin{pmatrix} 36. & 48. & -36. & -48. \\ 48. & 64. & -48. & -64. \\ -36. & -48. & 36. & 48. \\ -48. & -64. & 48. & 64. \end{pmatrix}$$

Eigenvalues of Ke = {200., 0, 0, 0}

Symmetry check={{0, 0, 0, 0}, {0, 0, 0, 0}, {0, 0, 0, 0}, {0, 0, 0, 0}}

FIGURE 20.3. Test of plane bar stiffness module with numerical inputs.

§20.1.3. Testing the Plane Bar Module

The modules are tested by the scripts listed in Figures 20.3 and 20.4. The script shown on the top of Figure 20.3 tests a numerically defined element with end nodes located at $(0, 0)$ and $(30, 40)$, with $E = 1000$, $A = 5$, and numer set to True. Executing the script produces the results listed in the bottom of that figure.
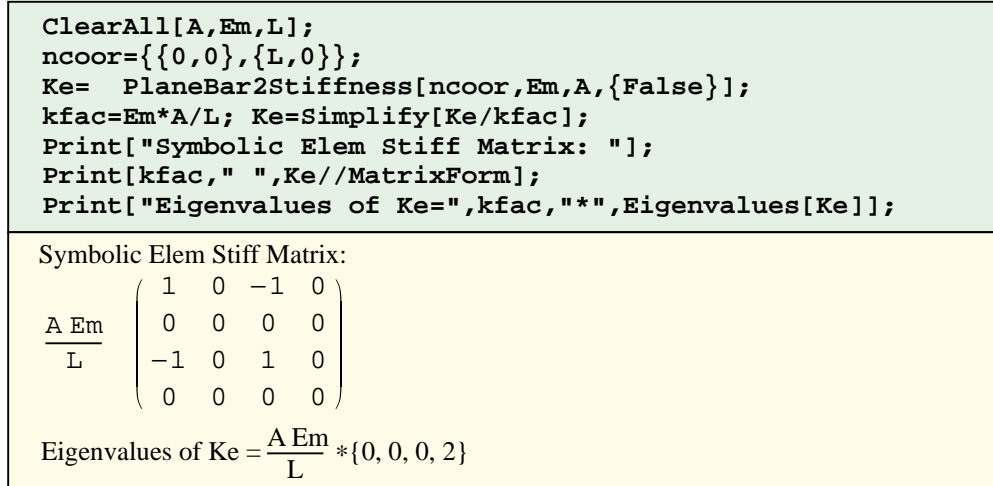
```
ClearAll[A,Em,L];
ncoor={{0,0},{L,0}};
Ke=  PlaneBar2Stiffness[ncoor,Em,A,{False}];
kfac=Em*A/L; Ke=Simplify[Ke/kfac];
Print["Symbolic Elem Stiff Matrix: "];
Print[kfac," ",Ke//MatrixForm];
Print["Eigenvalues of Ke=",kfac,"*",Eigenvalues[Ke]];
```

Symbolic Elem Stiff Matrix:

$$\frac{A\,Em}{L} \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Eigenvalues of $Ke = \dfrac{A\,Em}{L} * \{0, 0, 0, 2\}$

FIGURE 20.4. Test of plane bar stiffness module with symbolic inputs.

On return from PlaneBar2Stiffness, the stiffness matrix returned in Ke is printed. Its four eigenvalues are computed and printed. As expected three eigenvalues, which correspond to the three independent rigid body motions of the element, are zero. The remaining eigenvalue is positive and equal to EA/ℓ. The symmetry of Ke is checked by printing $(\mathbf{K}^e)^T - \mathbf{K}^e$ upon simplification and chopping.

The script of Figure 20.4 tests a symbolically defined bar element with end nodes located at $(0, 0)$ and $(L, 0)$, which is aligned with the x axis. Properties E and A are kept symbolic. Executing the script shown in the top of Figure 20.4 produces the results shown in the bottom of that figure. One thing to be noticed is the use of the stiffness scaling factor EA/ℓ, called kfac in the script. This is a symbolic quantity that can be extracted as factor of matrix $\mathbf{K}^e$. The effect is to clean up matrix and vector output, as can be observed in the printed results.

§20.2. The Space Bar Element

To show how the previous implementation extends easily to three dimensions, this section describes the implementation of the space bar element.

The two-node, prismatic, space bar element is pictured in Figure 20.5. The element has two nodes and six degrees of freedom. The element node displacements and conjugate forces are arranged as

$$\mathbf{u}^e = \begin{bmatrix} u_{x1} \\ u_{y1} \\ u_{z1} \\ u_{x2} \\ u_{y2} \\ u_{z2} \end{bmatrix}, \quad \mathbf{f}^e = \begin{bmatrix} f_{x1} \\ f_{y1} \\ f_{z1} \\ f_{x2} \\ f_{y2} \\ f_{z2} \end{bmatrix}. \tag{20.4}$$
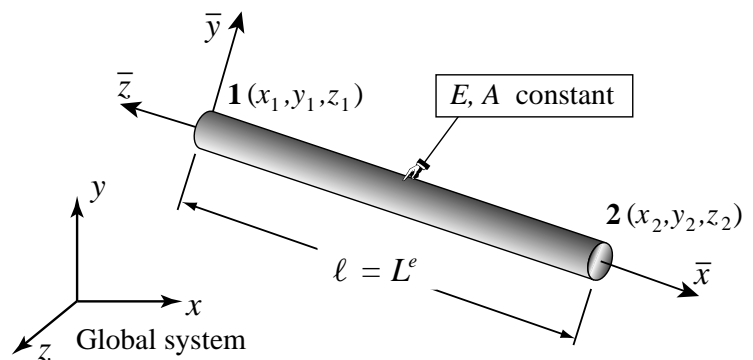
FIGURE 20.5. The space (3D) bar element.

```
SpaceBar2Stiffness[ncoor_,Em_,A_,options_]:=Module[
 {x1,x2,y1,y2,z1,z2,x21,y21,z21,EA,numer,L,LL,LLL,Ke},
  {{x1,y1,z1},{x2,y2,z2}}=ncoor;{x21,y21,z21}={x2-x1,y2-y1,z2-z1};
  EA=Em*A; {numer}=options;  LL=x21^2+y21^2+z21^2; L=Sqrt[LL];
  If [numer,{x21,y21,z21,EA,LL,L}=N[{x21,y21,z21,EA,LL,L}]];
  If [!numer, L=PowerExpand[L]]; LLL=Simplify[LL*L];
  Ke=(Em*A/LLL)*
     {{ x21*x21, x21*y21, x21*z21,-x21*x21,-x21*y21,-x21*z21},
      { y21*x21, y21*y21, y21*z21,-y21*x21,-y21*y21,-y21*z21},
      { z21*x21, z21*y21, z21*z21,-z21*x21,-z21*y21,-z21*z21},
      {-x21*x21,-x21*y21,-x21*z21, x21*x21, x21*y21, x21*z21},
      {-y21*x21,-y21*y21,-y21*z21, y21*x21, y21*y21, y21*z21},
      {-z21*x21,-z21*y21,-z21*z21, z21*x21, z21*y21, z21*z21}};
   Return[Ke];
];
```

FIGURE 20.6. Module to form the stiffness of the space (3D) bar element.

The element geometry is described by the coordinates $\{x_i, y_i, z_i\}$, $i = 1, 2$ of the two end nodes. As in the case of the plane bar, the two properties required for the stiffness computations are the modulus of elasticity E and the cross section area A. Both are assumed to be constant over the element.

§20.2.1. Stiffness Matrix

For the space bar element, introduce the notation $x_{21} = x_2 - x_1$, $y_{21} = y_2 - y_1$, $z_{21} = z_2 - z_1$ and $\ell = \sqrt{x_{21}^2 + y_{21}^2 + z_{21}^2}$. It can be shown[1] that the element stiffness matrix in global coordinates is given by

$$\mathbf{K}^e = \frac{E^e A^e}{\ell^3} \begin{bmatrix} x_{21}x_{21} & x_{21}y_{21} & x_{21}z_{21} & -x_{21}x_{21} & -x_{21}y_{21} & -x_{21}z_{21} \\ x_{21}y_{21} & y_{21}y_{21} & x_{21}z_{21} & -x_{21}y_{21} & -y_{21}y_{21} & -y_{21}z_{21} \\ x_{21}z_{21} & y_{21}z_{21} & z_{21}z_{21} & -x_{21}z_{21} & -y_{21}z_{21} & -z_{21}z_{21} \\ -x_{21}x_{21} & -x_{21}y_{21} & -x_{21}z_{21} & x_{21}x_{21} & x_{21}y_{21} & x_{21}z_{21} \\ -x_{21}y_{21} & -y_{21}y_{21} & -x_{21}z_{21} & x_{21}y_{21} & y_{21}y_{21} & y_{21}z_{21} \\ -x_{21}z_{21} & -y_{21}z_{21} & -z_{21}z_{21} & x_{21}z_{21} & y_{21}z_{21} & z_{21}z_{21} \end{bmatrix}. \quad (20.5)$$

This matrix expression in terms of coordinate differences is useful in symbolic work, because it enhances simplification possibilities.

§20.2.2. Stiffness Module

The computation of the stiffness matrix $\mathbf{K}^e$ of the two-node, prismatic space bar element, is done by *Mathematica* module SpaceBar2Stiffness. This is listed in Figure 20.6. The module is invoked as

$$Ke = SpaceBar2Stiffness[ncoor, Em, A, options] \quad (20.6)$$

The arguments are

ncoor   Node coordinates of element arranged as {{x1,y1,z1},{x2,y2,z2}}.

[1] The derivation was the subject of Exercise 6.10.

```
ClearAll[A,Em];
ncoor={{0,0,0},{2,3,6}}; Em=343; A=10;
Ke= SpaceBar2Stiffness[ncoor,Em,A,{True}];
Print["Numerical Elem Stiff Matrix: "];
Print[Ke//MatrixForm];
Print["Eigenvalues of Ke=",Chop[Eigenvalues[Ke]]];
```

Numerical Elem Stiff Matrix:

$$\begin{pmatrix} 40. & 60. & 120. & -40. & -60. & -120. \\ 60. & 90. & 180. & -60. & -90. & -180. \\ 120. & 180. & 360. & -120. & -180. & -360. \\ -40. & -60. & -120. & 40. & 60. & 120. \\ -60. & -90. & -180. & 60. & 90. & 180. \\ -120. & -180. & -360. & 120. & 180. & 360. \end{pmatrix}$$

Eigenvalues of Ke = {980., 0, 0, 0, 0, 0}

FIGURE 20.7. Testing the space bar stiffness module with numerical inputs.

Em   Elastic modulus.
A   Cross section area.
options   A list of processing options. For this element is has only one entry: { numer }. This
            is a logical flag with the value True or False. If True the computations are carried
            out in floating-point arithmetic. If False symbolic processing is assumed.

The module returns the 6×6 element stiffness matrix as function value.

```
ClearAll[A,Em,L];
ncoor={{0,0,0},{L,2*L,2*L}/3};
Ke=  SpaceBar2Stiffness[ncoor,Em,A,{False}];
kfac=Em*A/(9*L); Ke=Simplify[Ke/kfac];
Print["Symbolic Elem Stiff Matrix: "];
Print[kfac," ",Ke//MatrixForm];
Print["Eigenvalues of Ke=",kfac,"*",Eigenvalues[Ke]];
```

Symbolic Elem Stiff Matrix:

$$\frac{A\,Em}{9\,L} \begin{pmatrix} 1 & 2 & 2 & -1 & -2 & -2 \\ 2 & 4 & 4 & -2 & -4 & -4 \\ 2 & 4 & 4 & -2 & -4 & -4 \\ -1 & -2 & -2 & 1 & 2 & 2 \\ -2 & -4 & -4 & 2 & 4 & 4 \\ -2 & -4 & -4 & 2 & 4 & 4 \end{pmatrix}$$

Eigenvalues of Ke = $\dfrac{A\,Em}{9\,L} * \{0, 0, 0, 0, 0, 18\}$

FIGURE 20.8. Testing the space bar stiffness module with symbolic inputs.

§20.2.3. Testing the Space Bar Module

The modules are tested by the scripts listed in Figures 20.7 and 20.8. As these are similar to previous tests done on the plane bar they need not be described in detail.
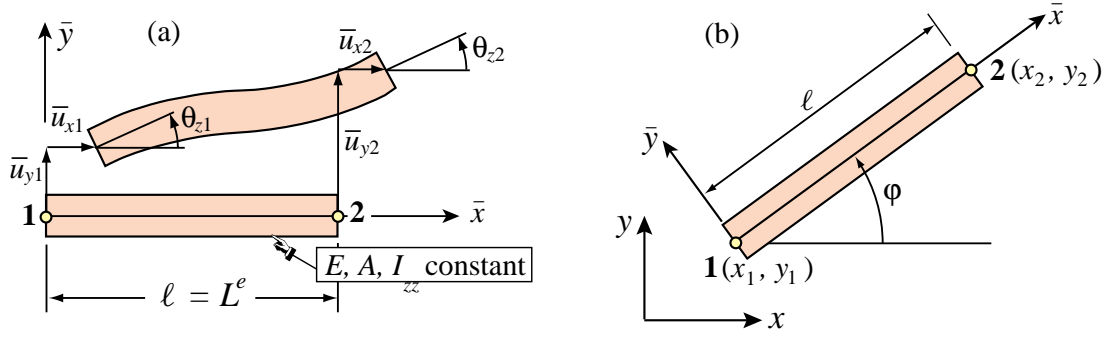
FIGURE 20.9. Plane beam-column element: (a) in its local system; (b) in the global system.

The script of Figure 20.7 tests a numerically defined space bar with end nodes located at $(0, 0, 0)$ and $(30, 40, 0)$, with $E = 1000$, $A = 5$ and `numer` set to `True`. Executing the script produces the results listed in the bottom of that Figure.

The script of Figure 20.8 tests a symbolically defined bar element with end nodes located at $(0, 0, 0)$ and $(L, 2L, 2L)/3$, which has length L and is not aligned with the x axis. The element properties E and A are kept symbolic. Executing the script produces the results shown in the bottom of that Figure. Note the use of a stiffness factor `kfac` of $EA/(9\ell)$ to get cleaner printouts.

§20.3. The Plane Beam-Column Element

Beam-column elements model structural members that resist both axial and bending actions. This is the case in skeletal structures such as frameworks which are common in steel and reinforced-concrete building construction. A plane beam-column element is a combination of a plane bar (such as that considered in §20.1), and a plane beam.

We consider a beam-column element in its local system $(\bar{x}, \bar{y})$ as shown in Figure 20.9(a), and then in the global system (x, y) as shown in Figure 20.9(b). The six degrees of freedom and conjugate node forces of the elements are:

$$\bar{\mathbf{u}}^e = \begin{bmatrix} \bar{u}_{x1} \\ \bar{u}_{y1} \\ \theta_{z1} \\ \bar{u}_{x2} \\ \bar{u}_{y2} \\ \theta_{z2} \end{bmatrix}, \quad \bar{\mathbf{f}}^e = \begin{bmatrix} \bar{f}_{x1} \\ \bar{f}_{y1} \\ m_{z1} \\ \bar{f}_{x2} \\ \bar{f}_{y2} \\ m_{z2} \end{bmatrix}, \quad \mathbf{u}^e = \begin{bmatrix} u_{x1} \\ u_{y1} \\ \theta_{z1} \\ u_{x2} \\ u_{y2} \\ \theta_{z2} \end{bmatrix}, \quad \mathbf{f}^e = \begin{bmatrix} f_{x1} \\ f_{y1} \\ m_{z1} \\ f_{x2} \\ f_{y2} \\ m_{z2} \end{bmatrix}. \tag{20.7}$$

The rotation angles θ and the nodal moments m are the same in the local and the global systems because they are about the z axis, which does not change in passing from local to global.

The element geometry is described by the coordinates $\{x_i, y_i\}$, $i = 1, 2$ of the two end nodes. The element length is $\ell = L^e$. Properties involved in the stiffness calculations are: the modulus of elasticity E, the cross section area A and the moment of inertia $I = I_{zz}$ about the neutral axis. All properties are taken to be constant over the element.

§20.3.1. Stiffness Matrix

To obtain the plane beam-column stiffness in the local system we simply add the stiffness matrices derived in Chapters 11 and 12, respectively, to get

$$
\bar{\mathbf{K}}^e = \frac{EA}{\ell}
\begin{bmatrix}
1 & 0 & 0 & -1 & 0 & 0 \\
 & 0 & 0 & 0 & 0 & 0 \\
 & & 0 & 0 & 0 & 0 \\
 & & & 1 & 0 & 0 \\
 & & & & 0 & 0 \\
\text{symm} & & & & & 0
\end{bmatrix}
+ \frac{EI}{\ell^3}
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
 & 12 & 6\ell & 0 & -12 & 6\ell \\
 & & 4\ell^2 & 0 & -6\ell & 2\ell^2 \\
 & & & 0 & 0 & 0 \\
 & & & & 12 & -6\ell \\
\text{symm} & & & & & 4\ell^2
\end{bmatrix}
\tag{20.8}
$$

The two matrices on the right of (20.8) come from the bar stiffness (12.22) and the Bernoulli-Euler bending stiffness (13.20), respectively. Before adding them, rows and columns have been rearranged in accordance with the nodal freedoms (20.7).

The displacement transformation matrix between local and global systems is

$$
\bar{\mathbf{u}}^e =
\begin{bmatrix}
\bar{u}_{x1} \\
\bar{u}_{y1} \\
\theta_{z1} \\
u_{x2} \\
\bar{u}_{y2} \\
\theta_{z2}
\end{bmatrix}
=
\begin{bmatrix}
c & s & 0 & 0 & 0 & 0 \\
-s & c & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & c & s & 0 \\
0 & 0 & 0 & -s & c & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
u_{x1} \\
u_{y1} \\
\theta_{z1} \\
u_{x2} \\
u_{y2} \\
\theta_{z2}
\end{bmatrix}
= \mathbf{T}\,\mathbf{u}^e,
\tag{20.9}
$$

where $c = \cos\varphi = (x_2 - x_1)/\ell$, $s = \sin\varphi = (y_2 - y_1)/\ell$, and φ is the angle between $\bar{x}$ and x, measured positive-counterclockwise from x; see Figure 20.9. The stiffness matrix in the global system is obtained through the congruent transformation

$$
\mathbf{K}^e = \mathbf{T}^T\,\bar{\mathbf{K}}^e\,\mathbf{T}.
\tag{20.10}
$$

Explicit expressions of the entries of $\mathbf{K}^e$ are messy. Unlike the bar, it is better to let the program do the transformation.

§20.3.2. Stiffness Module

The computation of the stiffness matrix $\mathbf{K}^e$ of the two-node, prismatic plane beam-column element is done by *Mathematica* module `PlaneBeamColumn2Stiffness`. This is listed in Figure 20.10. The module is invoked as

```
Ke = PlaneBeamColumn2Stiffness[ncoor, Em, {A,Izz}, options]      (20.11)
```

The arguments are

ncoor   Node coordinates of element arranged as {{x1,y1},{x2,y2}}.
Em   Elastic modulus.
A   Cross section area.
Izz   Moment of inertia of cross section area respect to axis z.
options   A list of processing options. For this element is has only one entry: {numer}. This is a logical flag with the value True or False. If True the computations are carried out in floating-point arithmetic. If False symbolic processing is assumed.

The module returns the 6×6 element stiffness matrix as function value.

```
PlaneBeamColumn2Stiffness[ncoor_,Em_,{A_,Izz_},options_]:=Module[
 {x1,x2,y1,y2,x21,y21,EA,EI,numer,L,LL,LLL,Te,Kebar,Ke},
  {{x1,y1},{x2,y2}}=ncoor; {x21,y21}={x2-x1,y2-y1};
  EA=Em*A; EI=Em*Izz; {numer}=options;
  LL=Simplify[x21^2+y21^2]; L=Sqrt[LL];
  If [numer,{x21,y21,EA,EI,LL,L}=N[{x21,y21,EA,EI,LL,L}]];
  If [!numer, L=PowerExpand[L]]; LLL=Simplify[LL*L];
  Kebar= (EA/L)*{
 { 1,0,0,-1,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},
 {-1,0,0, 1,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0}} +
         (2*EI/LLL)*{
 { 0,0,0,0,0,0},{0, 6, 3*L,0,-6, 3*L},{0,3*L,2*LL,0,-3*L,  LL},
 { 0,0,0,0,0,0},{0,-6,-3*L,0, 6,-3*L},{0,3*L,  LL,0,-3*L,2*LL}};
  Te={{x21,y21,0,0,0,0}/L,{-y21,x21,0,0,0,0}/L,{0,0,1,0,0,0},
      {0,0,0,x21,y21,0}/L,{0,0,0,-y21,x21,0}/L,{0,0,0,0,0,1}};
  Ke=Transpose[Te].Kebar.Te;
  Return[Ke] ];
```

FIGURE 20.10. *Mathematica* module to form the stiffness matrix of a two-node,
prismatic plane beam-column element.

```
ClearAll[L,Em,A,Izz];
ncoor={{0,0},{3,4}}; Em=100; A=125; Izz=250;
Ke= PlaneBeamColumn2Stiffness[ncoor,Em,{A,Izz},{True}];
Print["Numerical Elem Stiff Matrix: "];
Print[Ke//MatrixForm];
Print["Eigenvalues of Ke=",Chop[Eigenvalues[Ke]]];
```

Numerical Elem Stiff Matrix:

$$
\begin{pmatrix}
2436. & 48. & -4800. & -2436. & -48. & -4800. \\
48. & 2464. & 3600. & -48. & -2464. & 3600. \\
-4800. & 3600. & 20000. & 4800. & -3600. & 10000. \\
-2436. & -48. & 4800. & 2436. & 48. & 4800. \\
-48. & -2464. & -3600. & 48. & 2464. & -3600. \\
-4800. & 3600. & 10000. & 4800. & -3600. & 20000.
\end{pmatrix}
$$

Eigenvalues of Ke = {34800., 10000., 5000., 0, 0, 0}

FIGURE 20.11. Test of two-node plane beam-column element with numeric inputs.

§20.3.3. Testing the Plane Beam-Column Module

The beam-column stiffness are tested by the scripts shown in Figures 20.11 and 20.12.

The script at the top of Figure 20.11 tests a numerically defined element of length $\ell = 5$ with end nodes located at $(0, 0)$ and $(3, 4)$, respectively, with $E = 100$, $A = 125$ and $I_{zz} = 250$. The output is shown at the bottom of that figure. The stiffness matrix returned in Ke is printed. Its six eigenvalues are computed and printed. As expected three eigenvalues, which correspond to the three independent rigid body motions of the element, are zero. The remaining three eigenvalues are positive.

The script at the top of Figure 20.12 tests a plane beam-column of length L with end nodes at $(0, 0)$ and $(3L/5, 4L/5)$. The properties E, A and I_{zz} are kept in symbolic form. The output is shown at the bottom of that figure. The printed matrix looks complicated because bar and beam coupling

occurs when the element is not aligned with the global axes. The eigenvalues are obtained in closed symbolic form, and their simplicity provides a good check that the transformation matrix (20.9) is orthogonal. Three eigenvalues are exactly zero; one is associated with the axial (bar) stiffness and two with the flexural (beam) stiffness.
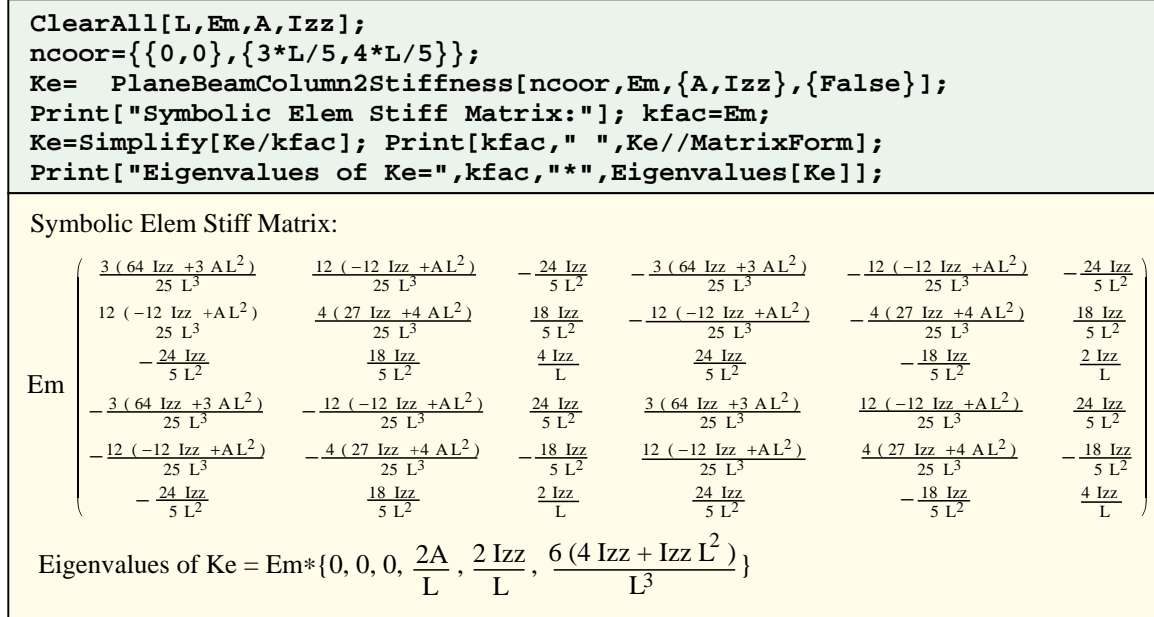
```
ClearAll[L,Em,A,Izz];
ncoor={{0,0},{3*L/5,4*L/5}};
Ke=  PlaneBeamColumn2Stiffness[ncoor,Em,{A,Izz},{False}];
Print["Symbolic Elem Stiff Matrix:"]; kfac=Em;
Ke=Simplify[Ke/kfac]; Print[kfac," ",Ke//MatrixForm];
Print["Eigenvalues of Ke=",kfac,"*",Eigenvalues[Ke]];
```

Symbolic Elem Stiff Matrix:

$$Em \begin{pmatrix} \frac{3(64\ Izz\ +3\ AL^2)}{25\ L^3} & \frac{12(-12\ Izz\ +AL^2)}{25\ L^3} & -\frac{24\ Izz}{5\ L^2} & -\frac{3(64\ Izz\ +3\ AL^2)}{25\ L^3} & -\frac{12(-12\ Izz\ +AL^2)}{25\ L^3} & -\frac{24\ Izz}{5\ L^2} \\ \frac{12(-12\ Izz\ +AL^2)}{25\ L^3} & \frac{4(27\ Izz\ +4\ AL^2)}{25\ L^3} & \frac{18\ Izz}{5\ L^2} & -\frac{12(-12\ Izz\ +AL^2)}{25\ L^3} & -\frac{4(27\ Izz\ +4\ AL^2)}{25\ L^3} & \frac{18\ Izz}{5\ L^2} \\ -\frac{24\ Izz}{5\ L^2} & \frac{18\ Izz}{5\ L^2} & \frac{4\ Izz}{L} & \frac{24\ Izz}{5\ L^2} & -\frac{18\ Izz}{5\ L^2} & \frac{2\ Izz}{L} \\ -\frac{3(64\ Izz\ +3\ AL^2)}{25\ L^3} & -\frac{12(-12\ Izz\ +AL^2)}{25\ L^3} & \frac{24\ Izz}{5\ L^2} & \frac{3(64\ Izz\ +3\ AL^2)}{25\ L^3} & \frac{12(-12\ Izz\ +AL^2)}{25\ L^3} & \frac{24\ Izz}{5\ L^2} \\ -\frac{12(-12\ Izz\ +AL^2)}{25\ L^3} & -\frac{4(27\ Izz\ +4\ AL^2)}{25\ L^3} & -\frac{18\ Izz}{5\ L^2} & \frac{12(-12\ Izz\ +AL^2)}{25\ L^3} & \frac{4(27\ Izz\ +4\ AL^2)}{25\ L^3} & -\frac{18\ Izz}{5\ L^2} \\ -\frac{24\ Izz}{5\ L^2} & \frac{18\ Izz}{5\ L^2} & \frac{2\ Izz}{L} & \frac{24\ Izz}{5\ L^2} & -\frac{18\ Izz}{5\ L^2} & \frac{4\ Izz}{L} \end{pmatrix}$$

Eigenvalues of Ke = Em$*\{0, 0, 0, \frac{2A}{L}, \frac{2\ Izz}{L}, \frac{6(4\ Izz + Izz\ L^2)}{L^3}\}$

FIGURE 20.12. Test of two-node plane beam-column element with symbolic inputs.

§20.4. *Plane Beam With Offset Nodes

§20.4.1. Plate Reinforced With Edge Beams

Consider a plate reinforced with edge beams, as shown in Figure 20.13(a). The conventional placement of the nodes is at the plate midsurface and beam longitudinal (centroidal) axis. But those *element centered* locations do not coincide. To assemble the structure it is necessary to refer both the plate and beam stiffness equations to common locations, because such equations are only written at nodes. We assume that those *connection nodes*, or simply *connectors*, will be placed at the plate midsurface, as sketched in Figure 20.13(b). With that choice there is no need to change the plate equations. The beam connectors have been moved, however, from their original centroidal positions. For the beam these connectors are also known as *offset nodes*.

Structural configurations such as that of Figure 20.13(a) are common in aerospace, civil and mechanical engineering when shells or plates are reinforced with eccentric stiffeners.

The process of moving the beam stiffness equation to the offset nodes is called *offsetting*. It relies on setting up multifreedom constraints (MFC) between centered and offset node freedoms, and applying the master-slave congruential transformation introduced in Chapter 8. For simplicity we discuss only this process assuming that the beam of Figure 20.13(b) is a plane beam whose freedoms are to be moved upwards by a distance d, which is positive if going upward from beam centroid. Freedoms at connection and centroidal nodes are declared to be master and slaves, respectively. They are labeled as shown in Figure 20.13(c,d). The original stiffness equations referred to centroidal (slave) freedoms are
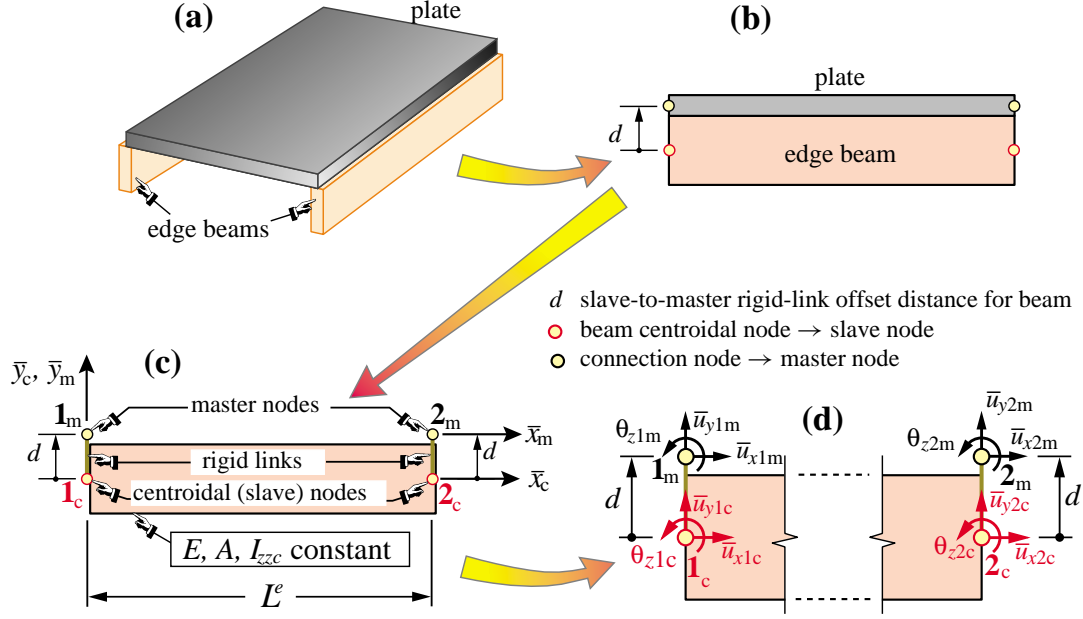
FIGURE 20.13. Plane beam with nodes offset for a rigid-link connection to plate.

$$
\frac{E}{L^e}
\begin{bmatrix}
A & 0 & 0 & -A & 0 & 0 \\
 & \dfrac{12I_{zzc}}{(L^e)^2} & \dfrac{6I_{zzc}}{(L^e)^2} & 0 & -\dfrac{12I_{zzc}}{(L^e)^2} & -\dfrac{6I_{zzc}}{(L^e)^2} \\
 & & \dfrac{4I_{zzc}}{L^e} & 0 & -\dfrac{6I_{zzc}}{(L^e)^2} & \dfrac{2I_{zzc}}{L^e} \\
 & & & A & 0 & 0 \\
 & & & & \dfrac{12I_{zzc}}{(L^e)^2} & -\dfrac{6I_{zzc}}{(L^e)^2} \\
 & \text{symm} & & & & \dfrac{4I_{zzc}}{L^e}
\end{bmatrix}
\begin{bmatrix}
\bar{u}_{x1s} \\ \bar{u}_{y1s} \\ \theta_{z1s} \\ \bar{u}_{x2s} \\ \bar{u}_{y2s} \\ \theta_{z2s}
\end{bmatrix}
=
\begin{bmatrix}
\bar{f}_{x1s} \\ \bar{f}_{y1s} \\ m_{z1s} \\ \bar{f}_{x2s} \\ \bar{f}_{y2s} \\ m_{z2s}
\end{bmatrix},
\quad \text{or} \quad \mathbf{K}_c^e \mathbf{u}_c^e = \mathbf{f}_c^e, \qquad (20.12)
$$

in which A is the beam cross section area while I_{zzc} denotes the section moment of inertia with respect to the centroidal axis $\bar{z}_c$.

§20.4.2. Rigid Link Transformation

Kinematic constraints between master and centroidal (slave) freedoms are obtained assuming that they are connected by *rigid links* as pictured in Figure 20.13(c,d). This gives the centroidal(slave)-to-master transformation

$$
\begin{bmatrix}
\bar{u}_{x1c} \\ \bar{u}_{y1c} \\ \theta_{z1c} \\ \bar{u}_{x2c} \\ \bar{u}_{y2c} \\ \theta_{z2c}
\end{bmatrix}
=
\begin{bmatrix}
1 & 0 & d & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & d \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
\bar{u}_{x1m} \\ \bar{u}_{y1m} \\ \theta_{z1m} \\ \bar{u}_{x2m} \\ \bar{u}_{y2m} \\ \theta_{z2m}
\end{bmatrix}
\quad \text{or} \quad \mathbf{u}_s^e = \mathbf{T}_{sm}\, \mathbf{u}_m^e. \qquad (20.13)
$$

The inverse transformation: $\mathbf{T}_{mc} = \mathbf{T}_{cm}^{-1}$ is obtained on replacing d with $-d$, as is physically obvious. The modified stiffness equations are obtained by the congruential transformation: $\mathbf{T}_{cm}^T \mathbf{K}_c^e \mathbf{T}_{cm} = \mathbf{T}_{cm}^T \mathbf{f}_c^e = \mathbf{f}_m^e$, which yields

$$\frac{E}{L^e}\begin{bmatrix} A & 0 & d & -A & 0 & -d \\ 0 & \frac{12\,I_{zzc}}{(L^e)^2} & \frac{6\,I_{zzc}}{L^e} & 0 & -\frac{12\,I_{zzc}}{(L^e)^2} & \frac{6\,I_{zzc}}{L^e} \\ d & \frac{6\,I_{zzc}}{L^e} & 4\,I_{zzc}+A\,d^2 & -d & -\frac{6\,I_{zzc}}{L^e} & 2\,I_{zzc}-A\,d^2 \\ -A & 0 & -d & A & 0 & d \\ 0 & -\frac{12\,I_{zzc}}{(L^e)^2} & -\frac{6\,I_{zzc}}{L^e} & 0 & \frac{12\,I_{zzc}}{(L^e)^2} & -\frac{6\,I_{zzc}}{L^e} \\ -d & \frac{6\,I_{zzc}}{L^e} & 2\,I_{zzc}-A\,d^2 & d & -\frac{6\,I_{zzc}}{L^e} & 4\,I_{zzc}+A\,d^2 \end{bmatrix}\begin{bmatrix} \bar{u}_{x1m} \\ \bar{u}_{y1m} \\ \theta_{z1m} \\ \bar{u}_{x2m} \\ \bar{u}_{y2m} \\ \theta_{z2m} \end{bmatrix} = \begin{bmatrix} \bar{f}_{x1m} \\ \bar{f}_{y1m} \\ m_{z1m} \\ \bar{f}_{x2m} \\ \bar{f}_{y2m} \\ m_{z2m} \end{bmatrix} \qquad (20.14)$$

Note that the modified equations are still referred to the local system $\{\bar{x}_m, \bar{y}_m\}$ pictured in Figure 20.13(c). Prior to assembly they should be transformed to the global system $\{x, y\}$ prior to assembly.

The foregoing transformation procedure has a flaw: for standard plate elements it will introduce compatibility errors at the interface between plate and beam. This may cause the beam stiffness to be significantly underestimated. See the textbook by Cook et. al. [72] for an explanation, and references therein. The following subsections describes a different scheme that builds $\mathbf{K}^e_m$ directly and cures that incompatibility.

§20.4.3. Direct Fabrication of Offset Beam Stiffness

This approach directly interpolates displacements and strains from master nodes placed at distance d from the beam longitudinal (centroidal) axis, as pictured in Figure 20.14. As usual the isoparametric coordinate ξ along the beam element varies from -1 at node 1 through $+1$ at node 2. The following cross section geometric properties are defined for use below:

$$A = \int_{A^e} dA, \quad S_z = \int_{A^e} \bar{y}\, dA = A\, d, \quad I_{zzc} = \int_{A^e} \bar{y}_c^2\, dA, \quad I_{zzm} = \int_{A^e} \bar{y}^2\, dA = I_{zzc} + A\, d^2, \quad (20.15)$$

The inplane displacements are expressed in term of the master freedoms at nodes 1_m and 2_m. Using the Bernoulli-Euler model gives

$$\begin{bmatrix} \bar{u}_{xm} \\ \bar{u}_{ym} \end{bmatrix} = \begin{bmatrix} N_{ux1} & -\bar{y}\frac{\partial N_{uy1}}{\partial \bar{x}} & -\bar{y}\frac{\partial N_{\theta z1}}{\partial \bar{x}} & N_{ux2} & -\bar{y}\frac{\partial N_{uy2}}{\partial \bar{x}} & -\bar{y}\frac{\partial N_{\theta z2}}{\partial \bar{x}} \end{bmatrix}\begin{bmatrix} \bar{u}_{x1m} \\ \bar{u}_{y1m} \\ \theta_{z1m} \\ \bar{u}_{x2m} \\ \bar{u}_{y2m} \\ \theta_{z2m} \end{bmatrix} \qquad (20.16)$$

where $N_{ux1} = \frac{1}{2}(1-\xi)/2$, $N_{ux2} = \frac{1}{2}(1+\xi)/2$, $N_{uy1} = \frac{1}{4}(1-\xi)^2(2+\xi)$, $N_{\theta z1} = \frac{1}{8}\ell(1-\xi)^2(1+\xi)$, $N_{uy2} = \frac{1}{4}(1+\xi)^2(2-\xi)$, $N_{\theta z2} = -\frac{1}{8}\ell(1+\xi)^2(1-\xi)$ are the usual bar and beam shape functions, but here referred to the offset axis $\bar{x}_m$.

The axial strain is $e_{xx} = \partial u_x/\partial \bar{x}$ and the strain energy $U^e = \frac{1}{2}\int_{V^e} E\, e_{xx}^2\, dV$ where $dV = A\, d\bar{x}{=}A$. Carrying
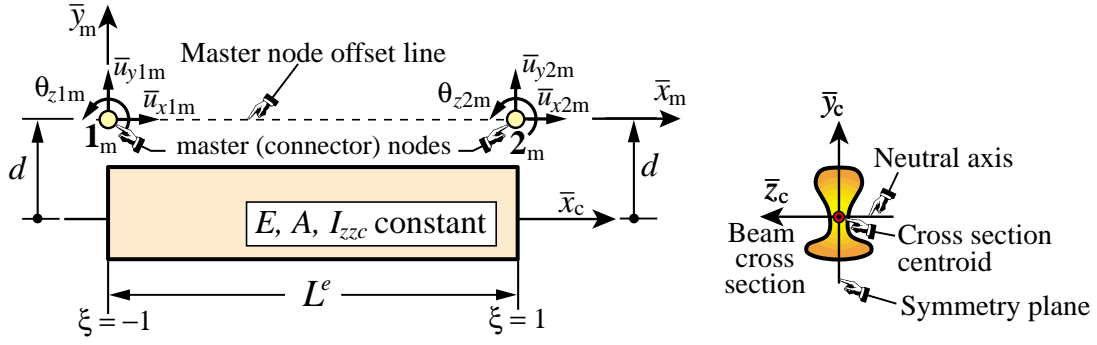
FIGURE 20.14. Plane beam fabricated directly from offset master node freedoms.

out the integral and differentiating twice with respect to the degrees of freedom yields the stiffness matrix

$$
\mathbf{K}^e = \frac{E}{L^e}
\begin{bmatrix}
A & 0 & -A\,d & -A & 0 & A\,d \\
0 & \dfrac{12\,(I_{zzc}+A\,d^2)}{(L^e)^2} & \dfrac{6\,(I_{zzc}+A\,d^2)}{L^e} & 0 & \dfrac{-12\,(I_{zzc}+A\,d^2)}{(L^e)^2} & \dfrac{6\,(I_{zzc}+A\,d^2)}{L^e} \\
-A\,d & \dfrac{6\,(I_{zzc}+A\,d^2)}{L^e} & 4\,(I_{zzc}+A\,d^2) & A\,d & \dfrac{-6\,(I_{zzc}+A\,d^2)}{L^e} & 2\,(I_{zzc}+A\,d^2) \\
-A & 0 & A\,d & A & 0 & -A\,d \\
0 & \dfrac{-12\,(I_{zzc}+A\,d^2)}{(L^e)^2} & \dfrac{-6\,(I_{zzc}+A\,d^2)}{L^e} & 0 & \dfrac{12\,(I_{zzc}+A\,d^2)}{(L^e)^2} & \dfrac{-6\,(I_{zzc}+A\,d^2)}{L^e} \\
A\,d & \dfrac{6\,(I_{zzc}+A\,d^2)}{L^e} & 2\,(I_{zzc}+A\,d^2) & -A\,d & \dfrac{-6\,(I_{zzc}+A\,d^2)}{L^e} & 4\,(I_{zzc}+A\,d^2)
\end{bmatrix}
$$

$$
= \frac{EA}{L^e}
\begin{bmatrix}
1 & 0 & -d & -1 & 0 & d \\
0 & \dfrac{12\,(r_G^2 + d^2)}{(L^e)^2} & \dfrac{6\,(r_G^2 + d^2)}{L^e} & 0 & \dfrac{-12\,(r_G^2 + d^2)}{(L^e)^2} & \dfrac{6\,(r_G^2 + d^2)}{L^e} \\
-d & \dfrac{6\,(r_G^2 + d^2)}{L^e} & 4\,(r_G^2 + d^2) & d & \dfrac{-6\,(r_G^2 + d^2)}{L^e} & 2\,(r_G^2 + d^2) \\
-1 & 0 & d & 1 & 0 & -1\,d \\
0 & \dfrac{-12\,(r_G^2 + d^2)}{(L^e)^2} & \dfrac{-6\,(r_G^2 + d^2)}{L^e} & 0 & \dfrac{12\,(r_G^2 + d^2)}{(L^e)^2} & \dfrac{-6\,(r_G^2 + d^2)}{L^e} \\
d & \dfrac{6\,(r_G^2 + d^2)}{L^e} & 2\,(r_G^2 + d^2) & -d & \dfrac{-6\,(r_G^2 + d^2)}{L^e} & 4\,(r_G^2 + d^2)
\end{bmatrix}
\tag{20.17}
$$

In the second form, $r_G^2 = I_{zzc}/A$ is the squared radius of gyration of the cross section about z.

Comparing the first form of $\mathbf{K}^e$ in (20.17) with (20.14) shows that the bending terms are significantly different if $d \neq 0$. These differences underscore the limitations of the rigid link assumption.

§20.5. Layered Beam Configurations

Another application of rigid link constraints is to modeling of *layered beams*, also called *composite beams* as well as *beam stacks*. These are beam-columns fabricated with beam layers that are linked to operate collectively as a beam member. In this section we study how to form the stiffness equations of beam stacks under the following assumptions:

1. The overall cross section of the beam member is rectangular and prismatic (that is, the cross section is constant along the longitudinal direction).

2. Both the beam stack and each of its constituent layers are prismatic.

3. The layers are of homogeneous isotropic material. Layer material, however, may vary from layer to layer.

The key modeling assumption is: is interlayer slip allosed or not? The two cases are studied next.

§20.5.1. Layered Beam With No Interlayer Slip

The main modeling constraint here is: *if all layers are of the same material, the stiffness equations should reduce to those of a homogenous beam.* To discuss how to meet this requirement it is convenient to introduce a *beam template* that separates the stiffness matrix into basic and higher order components. Consider a homogeneous, isotropic prismatic beam column element with elastic modulus E, cross section area A and cross section second moment of inertia I_{xxc} with respect to its neutral (centroidal) axis. The template form of the stiffness matrix in the local system is

$$\bar{\mathbf{K}}^e = \mathbf{K}_b^e + \mathbf{K}_h^e = \begin{bmatrix} K_{b1} & 0 & 0 & -K_{b1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & K_{b2} & 0 & 0 & -K_{b2} \\ -K_{b1} & 0 & 0 & K_{b1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -K_{b2} & 0 & 0 & K_{b2} \end{bmatrix} + \beta_h \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & K_{h3} & K_{h4} & 0 & -K_{h3} & K_{h4} \\ 0 & K_{h4} & K_{h5} & 0 & -K_{h4} & K_{h5} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -K_{h3} & -K_{h4} & 0 & K_{h3} & -K_{h4} \\ 0 & K_{h4} & K_{h5} & 0 & -K_{h4} & K_{h5} \end{bmatrix} \quad (20.18)$$

in which $K_{b1} = EA/L^e$, $K_{b2} = EI_{zzc}/L^e$, $K_{h3} = 12EI_{zzc}/(L^e)^3$, $K_{h4} = 6EI_{zzc}/(L^e)^2$ and $K_{h5} = 3EI_{zzc}/L^e$. Here β_h is a free parameter that scales the higher order stiffness $\mathbf{K}_h$. If $\beta_h = 1$ we recover the standard beam column stiffness (20.8). For a rectangular cross section of height H and width h, $A = Hh$ and $I_{zzc} = H^3h/12$, and the template (20.18) becomes

$$\bar{\mathbf{K}}^e = \frac{Eh}{L^e} \begin{bmatrix} H & 0 & 0 & -H & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{12}H^3 & 0 & 0 & -\frac{1}{12}H^3 \\ -H & 0 & 0 & H & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{12}H^3 & 0 & 0 & \frac{1}{12}H^3 \end{bmatrix} + \frac{\beta_h \, E \, H^3 h}{4(L^e)^3} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4 & 2L^e & 0 & -4 & 2L^e \\ 0 & 2L^e & (L^e)^2 & 0 & -2L^e & (L^e)^2 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -4 & -2L^e & 0 & 4 & -2L^e \\ 0 & 2L^e & (L^e)^2 & 0 & -2L^e & (L^e)^2 \end{bmatrix} \quad (20.19)$$

Next, cut the foregoing beam into two identical layers of height $H_k = H/2$, where $k = 1, 2$ is used as layer index. See Figure 20.15(b). The layers have area $A_k = H_k h = Hh/2$ and self inertia $I_{xxk} = H_k^3 h/12 = H^3 h/96$. The layer stiffness matrices in template form are

$$\bar{\mathbf{K}}_k^e = \frac{Eh}{L^e} \begin{bmatrix} H_k & 0 & 0 & -H_k & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{12}H_k^3 & 0 & 0 & -\frac{1}{12}H_k^3 \\ -H_k & 0 & 0 & H_k & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{12}H_k^3 & 0 & 0 & \frac{1}{12}H_k^3 \end{bmatrix} + \frac{\beta_{hk} \, E \, H_k^3 h}{4(L^e)^3} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4 & 2L^e & 0 & -4 & 2L^e \\ 0 & 2L^e & (L^e)^2 & 0 & -2L^e & (L^e)^2 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -4 & -2L^e & 0 & 4 & -2L^e \\ 0 & 2L^e & (L^e)^2 & 0 & -2L^e & (L^e)^2 \end{bmatrix}, \quad k = 1, 2. \quad (20.20)$$

Beacuse the layers are identical it is reasonable to assume the same higher order free parameter for both layers, that is, $\beta_{h1} = \beta_{h2} = \beta_h$. The offset distances from each layer to the centroid of the full beam to the centroid of each layer are $d_1 = -H/4$ and $d_2 = H/4$. The rigid-link transformation matrices for (20.20) are the same as those found in the previous Section:

$$\mathbf{T}_1 = \begin{bmatrix} 1 & 0 & -H/4 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -H/4 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{T}_2 = \begin{bmatrix} 1 & 0 & H/4 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & H/4 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (20.21)$$
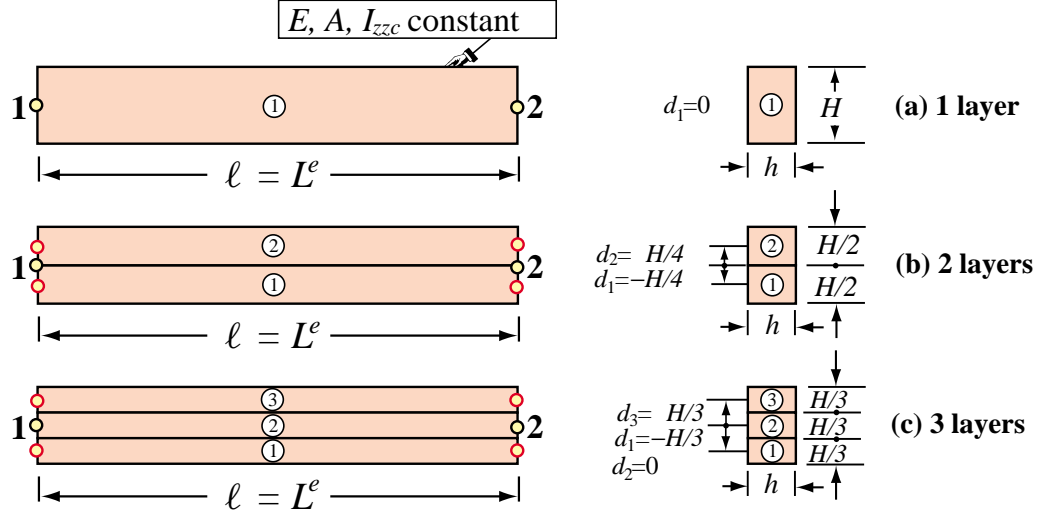
FIGURE 20.15. Plane beam divided into identical "sticking" layers.

Transforming and adding layers contributions as $\bar{\mathbf{K}}^e = \mathbf{T}_1^T \bar{\mathbf{K}}_1^e \mathbf{T}_1 + \mathbf{T}_2^T \bar{\mathbf{K}}_2^e \mathbf{T}_2$ gives

$$\bar{\mathbf{K}}^e = \frac{E\,h}{L^e} \begin{bmatrix} H & 0 & 0 & -H & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{12}H^3 & 0 & 0 & -\frac{1}{12}H^3 \\ -H & 0 & 0 & H & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{12}H^3 & 0 & 0 & \frac{1}{12}H^3 \end{bmatrix} + \frac{\beta_h\,E\,H^3\,h}{16(L^e)^3} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4 & 2L^e & 0 & -4 & 2L^e \\ 0 & 2L^e & (L^e)^2 & 0 & -2L^e & (L^e)^2 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -4 & -2L^e & 0 & 4 & -2L^e \\ 0 & 2L^e & (L^e)^2 & 0 & -2L^e & (L^e)^2 \end{bmatrix}. \quad (20.22)$$

This becomes identical to (20.19) if we set $\beta_h = 4$.

Carrying out the same exercise for three identical layers of height $H/3$, as shown in Figure 20.15(c), yields

$$\bar{\mathbf{K}}^e = \frac{E\,h}{L^e} \begin{bmatrix} H & 0 & 0 & -H & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{12}H^3 & 0 & 0 & -\frac{1}{12}H^3 \\ -H & 0 & 0 & H & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{12}H^3 & 0 & 0 & \frac{1}{12}H^3 \end{bmatrix} + \frac{\beta_h\,E\,H^3\,h}{36(L^e)^3} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4 & 2L^e & 0 & -4 & 2L^e \\ 0 & 2L^e & (L^e)^2 & 0 & -2L^e & (L^e)^2 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -4 & -2L^e & 0 & 4 & -2L^e \\ 0 & 2L^e & (L^e)^2 & 0 & -2L^e & (L^e)^2 \end{bmatrix}, \quad (20.23)$$

which becomes identical to (20.19) if we set $\beta_h = 9$. It is not difficult to show that if the beam is divided into $N \geq 2$ layers of height H/N, the correct beam stiffness is recovered if we take $\beta_h = N^2$.

If is not difficult to prove the following generalization. Suppose that the beam is cut into N layers of heights $H_k = \gamma_k H$, $k = 1, \ldots N$ that satisfy $\sum_1^N H_k = H$ or $\sum_1^N \gamma_k = 1$. To get the correct stiffness of the layered beam take

$$\beta_{hk} = 1/\gamma_k^2. \quad (20.24)$$

For example, suppose that the but is divided into 3 layers of thicknesses $H_1 = H_3 = H/4$ and $H_2 = H/2$. Then pick $\beta_{h1} = \beta_{h3} = 1/(\frac{1}{4})^2 = 16$ and $\beta_{h2} = 1/(\frac{1}{2})^2 = 4$.

What is the interpretation of this boost? A spectral analysis of the combined stiffness shows that taking $\beta_{hk} = 1$ lowers the rigidity associated with the antisymmetric bending mode of the element. But this mode is associated with shear-slippage between layers. Boosting β_{hk} as found above compensates exactly for this rigidity decay.

§20.5.2. Beam Stacks Allowing Interlayer Slip

Elastic modulus of facings $= E_f = 65$ GPa
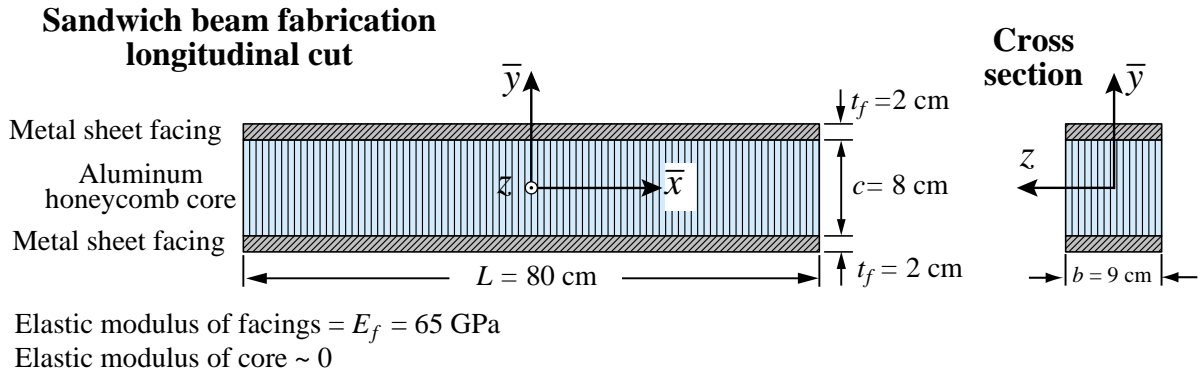Elastic modulus of core ~ 0

FIGURE 20.16. Plane sandwich beam.

There are beam fabrications where layers can slip longitudinally past each other. One important example is the sandwich beam illustrated in Figure 20.16. The beam is divided into three layers: 2 metal sheet facings and a honeycomb core. The core stiffness can be neglected since its effective elastic modulus is very low compared to the facings modulus E_f. In addition, the facings are not longitudinally bonded since they are separated by the weaker core. To form the beam stiffness by the rigid link method it is sufficient to form the faces self-stiffness, which are identical, and the rigid-link transformation matrices:

$$
\mathbf{K}_k^e = \frac{E_f}{L^e}
\begin{bmatrix}
A_f & 0 & 0 & -A_f & 0 & 0 \\
0 & \dfrac{12\,I_{zzf}}{(L^e)^2} & \dfrac{6\,I_{zzf}}{L^e} & 0 & -\dfrac{12\,I_{zzf}}{(L^e)^2} & \dfrac{6\,I_{zzf}}{L^e} \\
0 & \dfrac{6\,I_{zzf}}{L^e} & 4\,I_{zzf} & -d & -\dfrac{6\,I_{zzf}}{L^e} & 2\,I_{zzf} \\
-A_f & 0 & 0 & A_f & 0 & 0 \\
0 & -\dfrac{12\,I_{zzf}}{(L^e)^2} & -\dfrac{6\,I_{zzf}}{L^e} & 0 & \dfrac{12\,I_{zzf}}{(L^e)^2} & -\dfrac{6\,I_{zzf}}{L^e} \\
0 & \dfrac{6\,I_{zzf}}{L^e} & 2\,I_{zzf} & 0 & -\dfrac{6\,I_{zzf}}{L^e} & 4\,I_{zzf}
\end{bmatrix}, \quad
\mathbf{T}_k =
\begin{bmatrix}
1 & 0 & d_f & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & d_f \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}, \quad k = 1, 2.
$$
(20.25)

where, in the notation of Figure 20.16, $A_f = bt_f$, $I_{xxf} = bt_f^3/12$, $d_1 = -(c + t_f)/2$ and $d_2 = (c + t_f)/2$. The stiffness of the sandwich beam is $\mathbf{K}^e = \mathbf{T}_1^T\,K bold_1^e\,\mathbf{T}_1 + \mathbf{T}_2^T\,K bold_2^e\,\mathbf{T}_2$ into which the numerical values given in Figure 20.16 may be inserted. There is no need to use here the template form and of adjusting the higher order stiffness.

§20.6. The Space Beam Element

A second example in 3D is the general beam element shown in Figure 20.17. The element is prismatic and has two end nodes: 1 and 2, placed at the centroid of the end cross sections.

These define the local $\bar{x}$ axis as directed from 1 to 2. For simplicity the cross section will be assumed to be doubly symmetric, as is the case in commercial I and double-T profiles. The principal moments of inertia are defined by these symmetries. The local $\bar{y}$ and $\bar{z}$ axes are aligned with the symmetry lines of the cross section forming a RH system with $\bar{x}$. Consequently the principal moments of inertia are I_{yy} and I_{zz}, the bars being omitted for convenience.

The global coordinate system is $\{x, y, z\}$. To define the orientation of $\{\bar{y}, \bar{z}\}$ with respect to the global system, a third orientation node 3, which must not be colinear with 1–2, is introduced. See Figure 20.17. Axis $\bar{y}$ lies in the 1–2–3 plane and $\bar{z}$ is normal to 1–2–3.
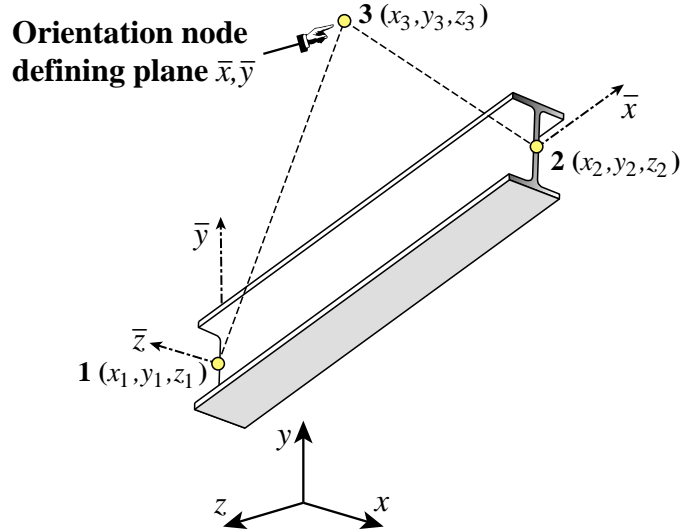

FIGURE 20.17. The space (3D) beam element.

Six global DOF are defined at each node i: the 3 translations u_{xi}, u_{yi}, u_{zi} and the 3 rotations θ_{xi}, θ_{yi}, θ_{zi}.

§20.6.1. Stiffness Matrix

The element global node displacements and conjugate forces are arranged as

$$
\begin{aligned}
\mathbf{u}^e &= [\, u_{x1} \quad u_{y1} \quad u_{z1} \quad \theta_{x1} \quad \theta_{y1} \quad \theta_{z1} \quad u_{x2} \quad u_{y2} \quad u_{z2} \quad \theta_{x2} \quad \theta_{y2} \quad \theta_{z2} \,]^T, \\
\mathbf{f}^e &= [\, f_{x1} \quad f_{y1} \quad f_{z1} \quad m_{x1} \quad m_{y1} \quad m_{z1} \quad f_{x2} \quad f_{y2} \quad f_{z2} \quad m_{x2} \quad m_{y2} \quad m_{z2} \,]^T.
\end{aligned}
\tag{20.26}
$$

The beam material is characterized by the elastic modulus E and the shear modulus G (the latter appears in the torsional stiffness). Four cross section properties are needed: the cross section area A, the moment of inertia J that characterizes torsional rigidity,[2] and the two principal moments of inertia I_{yy} and I_{zz} taken with respect to $\bar{y}$ and $\bar{z}$, respectively. The length of the element is denoted by L. The Bernoulli-Euler model is used; thus the effect of tranverse shear on the beam stiffness is neglected.

To simplify the following expressions, define the following "rigidity" combinations by symbols: $R^a = EA/L$, $R^t = GJ/L$, $R_{y3}^b = EI_{yy}/L^3$, $R_{y2}^b = EI_{yy}/L^2$, $R_y^b = EI_{yy}/L$, $R_{z3}^b = EI_{zz}/L^3$, $R_{z2}^b = EI_{zz}/L^2$, $R_z^b = EI_{zz}/L$. Note that R^a is the axial rigidity, R^t the torsional rigidity, while

---

[2] For circular and annular cross sections, J is the polar moment of inertia of the cross section wrt $\bar{x}$. For other sections J has dimensions of (length)^4 but must be calculated according to St. Venant's theory of torsion, or approximate theories.

```
SpaceBeamColumn2Stiffness[ncoor_,{Em_,Gm_},{A_,Izz_,Iyy_,Jxx_},
  options_]:= Module[
 {x1,x2,y1,y2,z1,z2,x21,y21,z21,xm,ym,zm,x0,y0,z0,dx,dy,dz,
  EA,EIyy,EIzz,GJ,numer,ra,ry,ry2,ry3,rz,rz2,rz3,rx,
  L,LL,LLL,yL,txx,txy,txz,tyx,tyy,tyz,tzx,tzy,tzz,T,Kebar,Ke},
  {x1,y1,z1}=ncoor[[1]]; {x2,y2,z2}=ncoor[[2]];
  {x0,y0,z0}={xm,ym,zm}={x1+x2,y1+y2,z1+z2}/2;
  If [Length[ncoor]<=2,{x0,y0,z0}+={0,1,0}];
  If [Length[ncoor]==3,{x0,y0,z0}=ncoor[[3]] ];
  {x21,y21,z21}={x2-x1,y2-y1,z2-z1}; {numer}=options;
  EA=Em*A; EIzz=Em*Izz; EIyy=Em*Iyy; GJ=Gm*Jxx;
  LL=Simplify[x21^2+y21^2+z21^2]; L=Sqrt[LL];
  If [numer, {x21,y21,z21,EA,EIyy,EIzz,GJ,LL,L}=
          N[{x21,y21,z21,EA,EIyy,EIzz,GJ,LL,L}]];
  If [!numer, L=PowerExpand[L]]; LLL=Simplify[LL*L];
  ra=EA/L; rx=GJ/L;
  ry=2*EIyy/L; ry2=6*EIyy/LL; ry3=12*EIyy/LLL;
  rz=2*EIzz/L; rz2=6*EIzz/LL; rz3=12*EIzz/LLL;
  Kebar={
  { ra,   0,   0,  0,   0,   0, -ra,   0,   0,   0,   0,   0},
  {  0, rz3,   0,  0,   0, rz2,   0,-rz3,   0,   0,   0, rz2},
  {  0,   0, ry3,  0,-ry2,   0,   0,   0,-ry3,   0,-ry2,   0},
  {  0,   0,   0, rx,   0,   0,   0,   0,   0, -rx,   0,   0},
  {  0,   0,-ry2,  0,2*ry,   0,   0,   0, ry2,   0,  ry,   0},
  {  0, rz2,   0,  0,   0,2*rz,   0,-rz2,   0,   0,   0,  rz},
  {-ra,   0,   0,  0,   0,   0,  ra,   0,   0,   0,   0,   0},
  {  0,-rz3,   0,  0,   0,-rz2,   0, rz3,   0,   0,   0,-rz2},
  {  0,   0,-ry3,  0, ry2,   0,   0,   0, ry3,   0, ry2,   0},
  {  0,   0,   0,-rx,   0,   0,   0,   0,   0,  rx,   0,   0},
  {  0,   0,-ry2,  0,  ry,   0,   0,   0, ry2,   0,2*ry,   0},
  {  0, rz2,   0,  0,   0,  rz,   0,-rz2,   0,   0,   0,2*rz}};
  {dx,dy,dz}={x0-xm,y0-ym,z0-zm};If[numer,{dx,dy,dz}=N[{dx,dy,dz}]];
  tzx=dz*y21-dy*z21; tzy=dx*z21-dz*x21; tzz=dy*x21-dx*y21;
  zL=Sqrt[tzx^2+tzy^2+tzz^2];
  If [!numer,zL=Simplify[PowerExpand[zL]]];
  {tzx,tzy,tzz}={tzx,tzy,tzz}/zL;   {txx,txy,txz}={x21,y21,z21}/L;
  tyx=tzy*txz-tzz*txy; tyy=tzz*txx-tzx*txz; tyz=tzx*txy-tzy*txx;
  Te={{txx,txy,txz,  0, 0, 0,  0, 0, 0,  0, 0, 0},
      {tyx,tyy,tyz,  0, 0, 0,  0, 0, 0,  0, 0, 0},
      {tzx,tzy,tzz,  0, 0, 0,  0, 0, 0,  0, 0, 0},
      {  0,  0,  0, txx,txy,txz,  0, 0, 0,  0, 0, 0},
      {  0,  0,  0, tyx,tyy,tyz,  0, 0, 0,  0, 0, 0},
      {  0,  0,  0, tzx,tzy,tzz,  0, 0, 0,  0, 0, 0},
      {  0,  0,  0,  0, 0, 0, txx,txy,txz,  0, 0, 0},
      {  0,  0,  0,  0, 0, 0, tyx,tyy,tyz,  0, 0, 0},
      {  0,  0,  0,  0, 0, 0, tzx,tzy,tzz,  0, 0, 0},
      {  0,  0,  0,  0, 0, 0,  0, 0, 0, txx,txy,txz},
      {  0,  0,  0,  0, 0, 0,  0, 0, 0, tyx,tyy,tyz},
      {  0,  0,  0,  0, 0, 0,  0, 0, 0, tzx,tzy,tzz}};
   Ke=Transpose[Te].Kebar.Te;
   Return[Ke]
 ];
```

FIGURE 20.18. Module to form stiffness of space (3D) beam.

the R^b's are bending rigities scaled by the length in various ways. Then the 12×12 local stiffness matrix can be written as[3]

---

[3] Cf. page 79 of Pzremieniecki [321]. The presentation in this book includes transverse shear effects as per Timoshenko's beam theory. The form (20.27) results from neglecting those effects.

$$\bar{\mathbf{K}}^e = \begin{bmatrix}
R^a & 0 & 0 & 0 & 0 & 0 & -R^a & 0 & 0 & 0 & 0 & 0 \\
0 & 12R^b_{z3} & 0 & 0 & 0 & 6R^b_{z2} & 0 & -12R^b_{z3} & 0 & 0 & 0 & 6R^b_{z2} \\
0 & 0 & 12R^b_{y3} & 0 & -6R^b_{y2} & 0 & 0 & 0 & -12R^b_{y3} & 0 & -6R^b_{y2} & 0 \\
0 & 0 & 0 & R^t & 0 & 0 & 0 & 0 & 0 & -R^t & 0 & 0 \\
0 & 0 & -6R^b_{y2} & 0 & 4R^b_y & 0 & 0 & 0 & 6R^b_{y2} & 0 & 2R^b_y & 0 \\
0 & 6R^b_{z2} & 0 & 0 & 0 & 4R^b_z & 0 & -6R^b_{z2} & 0 & 0 & 0 & 2R^b_z \\
-R^a & 0 & 0 & 0 & 0 & 0 & R^a & 0 & 0 & 0 & 0 & 0 \\
0 & -12R^b_{z3} & 0 & 0 & 0 & -6R^b_{z2} & 0 & 12R^b_{z3} & 0 & 0 & 0 & -6R^b_{z2} \\
0 & 0 & -12R^b_{y3} & 0 & 6R^b_{y2} & 0 & 0 & 0 & 12R^b_{y3} & 0 & 6R^b_{y2} & 0 \\
0 & 0 & 0 & -R^t & 0 & 0 & 0 & 0 & 0 & R^t & 0 & 0 \\
0 & 0 & -6R^b_{y2} & 0 & 2R^b_y & 0 & 0 & 0 & 6R^b_{y2} & 0 & 4R^b_y & 0 \\
0 & 6R^b_{z2} & 0 & 0 & 0 & 2R^b_z & 0 & -6R^b_{z2} & 0 & 0 & 0 & 4R^b_z
\end{bmatrix} \quad (20.27)$$

The transformation to the global system is the subject of Exercise 20.8.

§20.6.2. Stiffness Module

The computation of the stiffness matrix $\mathbf{K}^e$ of the two-node, prismatic space beam-column element is done by *Mathematica* module `SpaceBeamColumn2Stiffness`. This is listed in Figure 20.18. The module is invoked as

```
Ke = SpaceBeamColumn2Stiffness[ncoor, {Em,Gm}, {A,Izz,Iyy,Jxx}, options]
```
$$(20.28)$$

The arguments are

ncoor	Node coordinates of element arranged as {{x1,y1,z1},{x2,y2,z2},{x3,y3,z3}}, in which {x3,y3,z3} specifies an orientation node 3 that defines the local frame. See §20.4.1.
Em	Elastic modulus.
Gm	Shear modulus.
A	Cross section area.
Izz	Moment of inertia of cross section area respect to axis $\bar{z}$.
Iyy	Moment of inertia of cross section area respect to axis $\bar{y}$.
Jxx	Inertia with respect to $\bar{x}$ that appears in torsional rigidity GJ.
options	A list of processing options. For this element is has only one entry: {numer}. This is a logical flag with the value True or False. If True the computations are carried out in floating-point arithmetic. If False symbolic processing is assumed.

The module returns the 12×12 element stiffness matrix as function value.

The implementation logic and testing of this element is the subject of Exercises 20.8 and 20.9.

Notes and Bibliography

All elements implemented here are formulated in most books dealing with matrix structural analysis. Przemie-niecki [321] has been recommended in Chapter 1 on account of being inexpensive. The implementation and testing procedures are rarely covered.

The use of rigid links for offsetting degrees of freedom is briefly covered in §7.8 of the textbook [72].

Homework Exercises for Chapter 20
Implementation of One-Dimensional Elements

EXERCISE 20.1 [C:15] Download the plane bar stiffness module and their testers and verify the test results reported here. Comment on whether the stiffness matrix Ke has the correct rank of 1.

EXERCISE 20.2 [C:15] Download the space bar stiffness module and their testers and verify the test results reported here. Comment on whether the computed stiffness matrix Ke has the correct rank of 1.

EXERCISE 20.3 [C:15] Download the plane beam-column stiffness module and their testers and verify the test results reported here. Comment on whether the computed stiffness matrix Ke has the correct rank of 3.

EXERCISE 20.4 [A+C:30] Explain why the space bar element has rank 1 although it has 6 degrees of freedom and 6 rigid body modes. (According to the formula given in Chapter 19, the correct rank should be $6 - 6 = 0$.)

EXERCISE 20.5 [C:25] Implement the plane bar, plane beam-column and space bar stiffness element module in a lower level programming language and check them by writing a short test driver. [Do not bother about the mass modules.] Your choices are C, Fortran 77 or Fortran 90. (C++ is overkill for this kind of software).

EXERCISE 20.6 [A:25] Explain why the eigenvalues of $\mathbf{K}^e$ of any the elements given here do not change if the $\{x, y, z\}$ global axes change.

EXERCISE 20.7 [A+C:30] (Advanced) Implement a 3-node space bar element. *Hint:* use the results of Exercise 16.5 and transform the local stiffness to global coordinates via a 3×9 transformation matrix. Test the element and verify that it has two nonzero eigenvalues.

EXERCISE 20.8 [D+A:25] Explain the logic of the space beam module listed in Figure 20.18. Assume that the local stiffness matrix $\bar{\mathbf{K}}^e$ stored in Kebar is correct (it has been trranscribed from [321]). Instead, focus on how the local to global transformation is built and applied.

EXERCISE 20.9 [C:25] Test the space beam element of Figure 20.18 using the scripts given in Figures E20.1 and E20.2, and report results. Comment on whether the element exhibits the correct rank of 6.

```
ClearAll[L,Em,Gm,A,Izz,Iyy,Jxx];
ncoor={{0,0,0},{1,8,4}}; Em=54; Gm=30;
A=18; Izz=36; Iyy=72; Jxx=27;
Ke= SpaceBeamColumn2Stiffness[ncoor,{Em,Gm},{A,Izz,Iyy,Jxx},{True}];
Print["Numerical Elem Stiff Matrix: "];
Print[SetPrecision[Ke,4]//MatrixForm];
Print["Eigenvalues of Ke=",Chop[Eigenvalues[Ke]]];
```

FIGURE E20.1. Script for numeric testing of the space beam module of Figure 20.18.

```
ClearAll[L,Em,Gm,A,Izz,Iyy,Jxx];
ncoor={{0,0,0},{2*L,2*L,L}/3};
Ke=SpaceBeamColumn2Stiffness[ncoor,{Em,Gm},{A,Izz,Iyy,Jxx},{False}];
kfac=Em; Ke=Simplify[Ke/kfac];
Print["Numerical Elem Stiff Matrix: "];
Print[kfac," ",Ke//MatrixForm];
Print["Eigenvalues of Ke=",kfac,"*",Eigenvalues[Ke]];
```

FIGURE E20.2. Script for symbolic testing of the space beam module of Figure 20.18.