9

The Linear Tetrahedron

TABLE OF CONTENTS

Introduction

This Chapter covers the formulation and implementation of the simplest solid element: the four-node tetrahedron. This is also called the *linear tetrahedron* since its shape functions are linear polynomials. Within a programming context, the name is often abbreviated to `Tet4`. We start with this particular element for two reasons: its geometry is the simplest one in three space dimensions, and no numerical integration is needed to construct element equations.

§9.1. The Linear Tetrahedron

The linear tetrahedron, shown in Figure 9.1(a), is often shunned for stress analysis because of its poor performance.[1] Its main value in structural and solid mechanics is educational: it serves as a vehicle to introduce the basic steps of formulation of 3D solid elements, particularly as regards use of natural coordinate systems, node numbering conventions and computational ingredients. It should be noted that 3D visualization is notoriously more difficult than 2D, so we need to proceed somewhat slowly here.

§9.1.1. Tetrahedron Geometry

Figure 9.1 shows a typical four-node tetrahedron. Its geometry is fully defined by giving the position of the four corner nodes[2] with respect to the global RCC system $\{x, y, z\}$:

$$x_i, \quad y_i, \quad z_i \quad (i = 1,\ 2,\ 3,\ 4). \tag{9.1}$$

We will often use the abbreviations for corner coordinate differences:

$$x_{ij} = x_i - x_j, \quad y_{ij} = y_i - y_j, \quad z_{ij} = z_i - z_j, \quad i, j = 1, \ldots 4. \tag{9.2}$$

The four corners are assumed not to be coplanar. The element has six sides (also known as edges) and four faces. Sides are straight because they are defined by two corner points. Faces are planar because they are defined by three corner points. At each corner three sides and three faces meet.

The domain occupied by the tetrahedron is denoted by Ω^e. See Figure 9.1(a).

The volume measure of the tetrahedron is denoted by V, which should not be confused with the domain identifier Ω or Ω^e. The volume is given by the following determinant in terms of the corner coordinate values:

$$V = \int_{\Omega^e} d\Omega^e = \tfrac{1}{6} \det \begin{bmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \\ z_1 & z_2 & z_3 & z_4 \end{bmatrix} = \tfrac{1}{6} \det(\mathbf{J}) = \tfrac{1}{6}\, J. \tag{9.3}$$

[1] Reason: derivatives of shape functions are constant over the element volume. Strains and stresses recovered from displacement derivatives can be highly inaccurate, even exhibiting wrong signs. This deficiency makes the element unreliable for stress analysis when strains and stresses exhibit significant gradients. On the other hand, when the main goal of the simulation is merely to get values of primary (master) variables, as in thermal analysis and computational gas dynamics, the linear tetrahedron is acceptable. It is often used in such applications since there is abundant public-domain and commercial software that can generate a tetrahedral mesh to fill a given volume.

[2] A corner is called *vertex* (pl. *vertices*) in some expositions. “Corner” is, however, more common in the FEM literature.

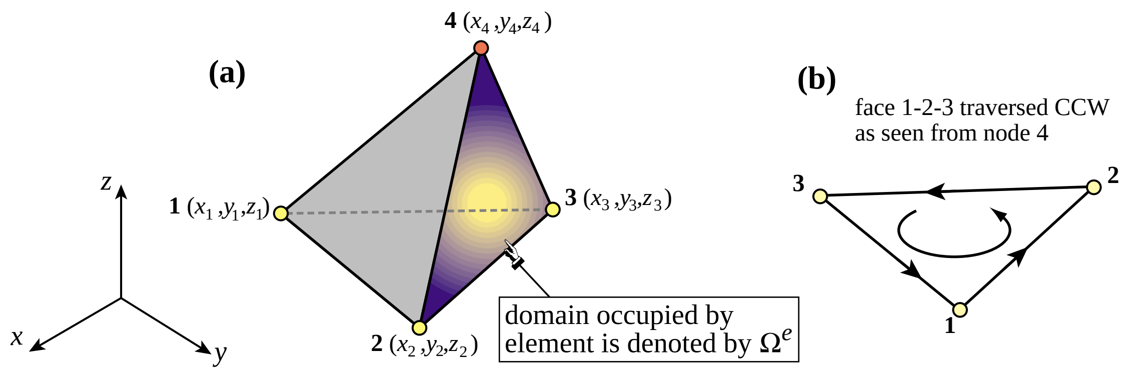

FIGURE 9.1. The four node tetrahedron element, also called the linear tetrahedron, or `Tet4` in programming context: (a) element picture; (b) corner node numbering convention.

The above displayed matrix (without the $1/6$ factor) is called the *Jacobian matrix* **J** and its determinant[3] the *Jacobian determinant* J. Note that J is a *signed* quantity, and so is $V = J/6$. We will always assume that V is *positive*. This can be insured if the nodes are not coplanar, and are appropriately numbered. A numbering rule that achieves this goal is as follows:

(I) Pick a corner (any corner) as initial one. In Figure 9.1(a) this is that numbered 1.

(II) Pick a face to contain the first three corners. The opposite corner will be numbered 4.

(III) Number those three corners in a *counterclockwise* sense when looking at the face from the opposite one. See Figure 9.1(b).

Should the volume computed by (9.3) be zero, this indicates that the four corner points are coplanar. This case should be flagged as an error.[4]

An explicit expression for the tetrahedron volume in terms of corner locations is

$$6V = J = x_{21}\,(y_{23}\,z_{34} - y_{34}\,z_{23}) + x_{32}\,(y_{34}\,z_{12} - y_{12}\,z_{34}) + x_{43}\,(y_{12}\,z_{23} - y_{23}\,z_{12}), \tag{9.4}$$

in which the abbreviations (9.2) for coordinate differences are used.

§9.1.2. Corners, Sides and Faces

The four corners are identified by integers 1 through 4, assigned as described above. The four faces are identified by either 3 integers or just one:

1. By their corner list: 234, 341, 412, 123.
2. By their opposite corner: 1,2,3,4.

See Figure 9.2(a). Which identification scheme is used depends mostly on notational convenience.

Finally, the six sides are identified by two integers: the number of their end (corner) nodes: 12, 23, 31, 14, 24, and 34. See Figure 9.2(b). Sides may also be defined by the intersection of two faces; for example side 12 is the intersection of faces 124 and 123 or, alternatively, faces 3 and 4.

---

[3] For more general geometries, such as the quadratic tetrahedron studied in the next Chapter, J is defined by more complicated expressions and generally varies over the elemet. If so the relation $V = J/6$ is no longer valid.

[4] In floating point computation the coplanarity test is trickier since numerical tresholds (tolerances) must be introduced.

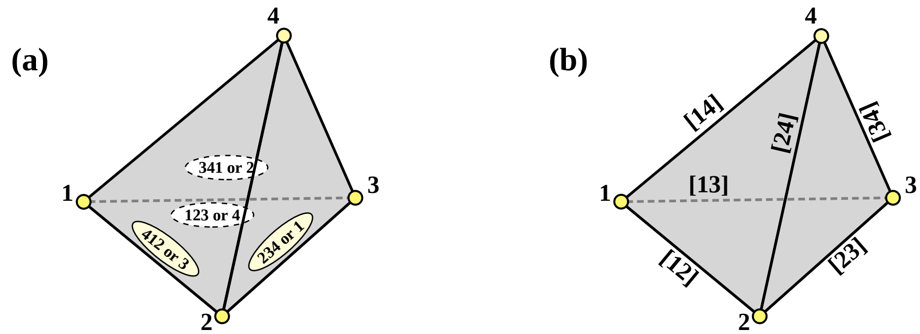

FIGURE 9.2. Identification of (a) tetrahedron faces, and (b) tetrahedron sides.

§9.1.3. Tetrahedral Coordinates

The position of a tetrahedron point may be specified either by its Cartesian coordinates $\{x, y, z\}$, or by its *tetrahedral natural coordinates*. The latter form a set of *four dimensionless numbers* denoted by

$$\zeta_1,\ \zeta_2,\ \zeta_3,\ \zeta_4. \tag{9.5}$$

The set (9.5) is the 3D analog of the triangular natural coordinates introduced in Chapter 15 of IFEM. They will be simply denoted as *tetrahedral coordinates* if there is no danger of confusion.

The value of ζ_i is one at corner i and zero at the other 3 corners, including the entire opposite face. It varies linearly with distance as one traverses the distance from the corner to that face. See Figure 9.3(a) for ζ_1. More precisely, consider an arbitrary point P as shown in Figure 9.3(b). Let h_{Pi} denote the distance of P to the face opposite the i^{th} corner, whereas h_i is the distance of that corner to that face. Then ζ_i is the ratio h_{Pi}/h_i, and similarly for the other three coordinates.

Because four coordinates is one too many for 3D space, there must be a constraint between the ζ_i. And indeed, as in the case of triangular coordinates, their sum must be identically one:

$$\zeta_1 + \zeta_2 + \zeta_3 + \zeta_4 = 1. \tag{9.6}$$

Remark 9.1. An equivalent definition sometimes found in the FEM literature uses the following volume ratios:

$$\zeta_1 = \frac{V_{P1}}{V}, \quad \zeta_2 = \frac{V_{P2}}{V}, \quad \zeta_3 = \frac{V_{P3}}{V}, \quad \zeta_4 = \frac{V_{P4}}{V}, \tag{9.7}$$

Here V^e_{Pi} denotes the volume of the "subtetrahedron" spanned by point P of coordinates $(\zeta_1, \zeta_2, \zeta_3, \zeta_4)$ and the face opposite the i^{th} corner, whereas V is the tetrahedron volume. This is illustrated for ζ_1 in Figure 9.3(b). The equivalence of $\zeta_i = h_{Pi}/h_i$ and $\zeta_i = V_{Pi}/V$ is readily shown on multiplying the former by the i^{th} face area. When the ζ_i are defined by (9.7) they are called *tetrahedral volume coordinates*, or simply *volume coordinates*. Adding the four definitions in (9.7) and noting[5] that $V_{P1} + V_{P2} + V_{P3} + V_{P4} = V$ for any point P, immediately yields (9.6).

This alternative definition has a serious disadvantage: it is not extendible to higher order tetrahedra with curved faces and sides. For example the quadratic tetrahedron described in the next Chapter. For this reason it will not be emphasized here.

[5] See (9.13) below.

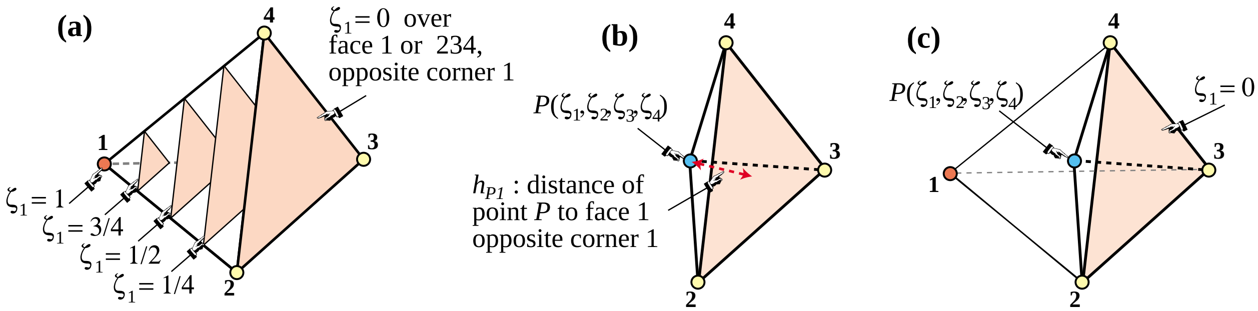

FIGURE 9.3. Tetrahedron natural coordinates: (a) visualization of ζ_1 as planes parallel to face 1 (opposite corner 1); (b) distance h_{P1} of a point P to that face; (c) interpretation as volume coordinates: $\zeta_i = V_{Pi}/V$, where V_{Pi} is the volume of the "subtetrahedron" spanned by $P(\zeta_1, \zeta_2, \zeta_3, \zeta_4)$ and the i^{th} face.

§9.1.4. Intrinsic Features

Intrinsic geometric features of a tetrahedron may be compactly expressed in terms of tetrahedral coordinates, independently of the Cartesian metric. Thus, the four corners 1, 2, 3, and 4 have coordinates (1,0,0,0), (0,1,0,0), (0,0,1,0), and (0,0,0,1), respectively. The opposite faces 1 or 234, 2 or 341, 3 or 412, and 4 or 123 have equations $\zeta_1 = 0$, $\zeta_2 = 0$, $\zeta_3 = 0$ and $\zeta_4 = 0$, respectively. More generally, the equation

$$\zeta_i = constant \tag{9.8}$$

represents planes parallel to the face opposite to the i^{th} corner, as shown in Figure 9.3(a) for ζ_1.

A tetrahedron has six sides, also called edges, defined by their two end corners: 12, 23, 31, 14, 24 and 34. Sides can also be defined by the intersection of two faces; for example side 12 is the intersection of faces 124 and 123, which have equations $\zeta_3 = 0$ and $\zeta_4 = 0$, respectively; consequently a *pair* of equations is required to define a side (or, more generally, a line).

Midpoints of sides 12, 23, 34, 14, 24 and 34 have tetrahedral coordinates $(\frac{1}{2}, \frac{1}{2}, 0, 0)$, $(0, \frac{1}{2}, \frac{1}{2}, 0)$, $(0, 0, \frac{1}{2}, \frac{1}{2})$, $(\frac{1}{2}, 0, 0, \frac{1}{2})$, $(0, \frac{1}{2}, 0, \frac{1}{2})$, and $(0, 0, \frac{1}{2}, \frac{1}{2})$, respectively. The centroid of faces 1 or 234, 2 or 341, 3 or 412, and 4 or 123, have tetrahedral coordinates $(0, \frac{1}{3}, \frac{1}{3}, \frac{1}{3})$, $(\frac{1}{3}, 0, \frac{1}{3}, \frac{1}{3})$, $(\frac{1}{3}, \frac{1}{3}, 0, \frac{1}{3})$, and $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0)$, respectively. Finally, the centroid has coordinates $(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$.

A homogeneous linear equation in tetrahedral coordinates, such as $\alpha_1\zeta_1 + \alpha_2\zeta_2 + \alpha_3\zeta_3 + \alpha_4\zeta_4 = 0$, represents a *plane* in 3D space, as discussed in the next subsection. Of particular interest are equations typified, say, by $\zeta_1 = \zeta_4$; this one represents a *median plane* that passes through side 23 and the midpoint of side 14. There are six median planes, all of which intersect at the centroid.

§9.1.5. Linear Interpolation Over Tetrahedron

Any function *linear* in x, y, z, say $F(x, y, z)$, that takes the values F_i ($i = 1, 2, 3, 4$) at the corners of the linear tetrahedron may be interpolated in terms of the tetrahedron coordinates as

$$F(\zeta_1, \zeta_2, \zeta_3, \zeta_4) = F_1\zeta_1 + F_2\zeta_2 + F_3\zeta_3 + F_4\zeta_4 = F_i\zeta_i. \tag{9.9}$$

(In the last expression the summation convention over $i = 1, 2, 3, 4$ is used.) Geometrically this particular $F(\zeta_1, \zeta_2, \zeta_3, \zeta_4) = 0$ represents a *plane*. The homogeneous form does not entail loss of generality, as explained in the Remark below.

Remark 9.2. Any (linear or nonlinear) polynomial expression in the ζ_i can be put into *homogeneous* form. This can be best shown by an example. Consider $3\zeta_1 + 7\zeta_2 + 12\zeta_3 - 5\zeta_4 = 2$. Replace the RHS by $2 = 2(\zeta_1 + \zeta_2 + \zeta_3 + \zeta_4)$ and combine with the LHS to get $\zeta_1 + 5\zeta_2 + 10\zeta_3 - 7\zeta_4 = 0$.

Example 9.1. Suppose that $F(x, y, z) = 4x + 9y - 8z + 3$ and that the Cartesian coordinates of corners 1,2,3,4 are (0, 0, 0), (1, 0, 0), (0, 1, 0) and (0, 0, 1), respectively. The values of F at the corners are $F_1 = 3$, $F_2 = 7$, $F_3 = 12$ and $F_4 = -5$. Consequently $F(\zeta_1, \zeta_2, \zeta_3, \zeta_4) = 3\zeta_1 + 7\zeta_2 + 12\zeta_3 - 5\zeta_4$.

§9.1.6. Coordinate Transformations

Quantities that are intrinsically linked to the element geometry (e.g., shape functions) are best expressed in tetrahedral coordinates. On the other hand, quantities such as displacement, strain and stress components are expressed in the Cartesian system $\{x, y, z\}$. Ergo, we need transformation equations to pass from one coordinate system to the other. The element geometric description in terms of tetrahedral coordinates follows by applying the linear interpolation (9.9) to x, y and z, so that $x = x_i \zeta_i$, $y = y_i \zeta_i$, and $z = z_i \zeta_i$. Prepending the sum-of-tetrahedral-coordinates identity (9.6) as first row we build the matrix relation

$$\begin{bmatrix} 1 \\ x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \\ z_1 & z_2 & z_3 & z_4 \end{bmatrix} \begin{bmatrix} \zeta_1 \\ \zeta_2 \\ \zeta_3 \\ \zeta_4 \end{bmatrix}. \tag{9.10}$$

As noted in §9.1.1, the above 4×4 matrix is called the *Jacobian* matrix of the linear tetrahedron. Explicit inversion gives

$$\begin{bmatrix} \zeta_1 \\ \zeta_2 \\ \zeta_3 \\ \zeta_4 \end{bmatrix} = \frac{1}{6V} \begin{bmatrix} 6V_{01} & y_{42} z_{32} - y_{32} z_{42} & x_{32} z_{42} - x_{42} z_{32} & x_{42} y_{32} - x_{32} y_{42} \\ 6V_{02} & y_{31} z_{43} - y_{34} z_{13} & x_{43} z_{31} - x_{13} z_{34} & x_{31} y_{43} - x_{34} y_{13} \\ 6V_{03} & y_{24} z_{14} - y_{14} z_{24} & x_{14} z_{24} - x_{24} z_{14} & x_{24} y_{14} - x_{14} y_{24} \\ 6V_{04} & y_{13} z_{21} - y_{12} z_{31} & x_{21} z_{13} - x_{31} z_{12} & x_{13} y_{21} - x_{12} y_{31} \end{bmatrix} \begin{bmatrix} 1 \\ x \\ y \\ z \end{bmatrix}, \tag{9.11}$$

in which the abbreviations (9.2) are used. The first column entries are explicitly given by

$$\begin{aligned} 6V_{01} &= x_2 (y_3 z_4 - y_4 z_3) + x_3 (y_4 z_2 - y_2 z_4) + x_4 (y_2 z_3 - y_3 z_2), \\ 6V_{02} &= x_1 (y_4 z_3 - y_3 z_4) + x_3 (y_1 z_4 - y_4 z_1) + x_4 (y_3 z_1 - y_1 z_3), \\ 6V_{03} &= x_1 (y_2 z_4 - y_4 z_2) + x_2 (y_4 z_1 - y_1 z_4) + x_4 (y_1 z_2 - y_2 z_1), \\ 6V_{04} &= x_1 (y_3 z_2 - y_2 z_3) + x_2 (y_1 z_3 - y_3 z_1) + x_3 (y_2 z_1 - y_1 z_2). \end{aligned} \tag{9.12}$$

These V_{0i} have the following geometric interpretation:[6] signed volumes of the tetrahedra formed by the origin $x = y = z = 0$ and faces 234, 341, 412 and 123, for $i = 1, 2, 3, 4$, respectively. That physical interpretation suggests the identity

$$V = V_{01} + V_{02} + V_{03} + V_{04}. \tag{9.13}$$

This can be easily verified either by comparing the sum of (9.12) to the explicit volume expression (9.4) or, equivalently, expanding the determinant of the Jacobian matrix (9.3) by the 2nd row.

[6] The interpretation is immediate by making $x = y = z = 0$ in (9.11), and comparing the LHS to the definition (9.7) of the ζ_i as volume coordinates, with P placed at the origin of the Cartesian frame. These entries are of limited importance in element stiffness development, for which partial derivative relations such as (9.19) play a more significant role.

To facilitate use of the summation convention it is convenient to represent entries of the rightmost three columns in the foregoing matrix using a more compact notation:

$$\begin{bmatrix} \zeta_1 \\ \zeta_2 \\ \zeta_3 \\ \zeta_4 \end{bmatrix} = \frac{1}{6V} \begin{bmatrix} 6V_{01} & a_1 & b_1 & c_1 \\ 6V_{02} & a_2 & b_2 & c_2 \\ 6V_{03} & a_3 & b_3 & c_3 \\ 6V_{04} & a_4 & b_4 & c_4 \end{bmatrix} \begin{bmatrix} 1 \\ x \\ y \\ z \end{bmatrix}. \tag{9.14}$$

§9.1.7. Cartesian Face Equations

Since the equation of the i^{th} face in terms of tetrahedral coordinates is $\zeta_i = 0$, it follows immediately from (9.14) that its Cartesian equation is

$$6V_{0i} + a_i\, x + b_i\, y + c_i\, z = 0, \quad i = 1, 2, 3, 4. \tag{9.15}$$

This can be transformed to the normal equation of a plane by scaling:

$$\frac{1}{S_i}(6V_{0i} + a_i\, x + b_i\, y + c_i\, z) = 6\bar{V}_{0i} + \bar{a}_i\, x + \bar{b}_i\, y + \bar{c}_i\, z = 0, \quad i = 1, 2, 3, 4, \tag{9.16}$$

in which $S_i = +\sqrt{a_i^2 + b_i^2 + c_i^2}$. Three useful geometric interpretations:

1. S_i is twice the area A_i (in absolute value) of the face opposite to corner i. Consequenty the tetrahedron altitudes are $h_i = 3V/A_i = 6V/S_i$. See Exercise E9.2.
2. $\bar{a}_i = a_i/S_i$, $\bar{b}_i = b_i/S_i$, and $\bar{c}_i = c_i/S_i$ are the direction cosines of the *unit interior normal* to the i^{th} face, with respect to $\{x, y, z\}$. For the *unit exterior normal,* take $\{-\bar{a}_i, -\bar{b}_i, -\bar{c}_i\}$.
3. The signed distance h_P of a point P of coordinates $\{x_P, y_P, z_P\}$ to the i^{th} face is obtained by substituting $x \to x_P$, $y \to y_P$ and $z \to z_P$ into either of the left-hand sides of (9.16).

§9.1.8. Partial Derivatives

From (9.10) and (9.14) we can easily find the following relations that connect partial derivatives of Cartesian and tetrahedral coordinates:

$$\frac{\partial x}{\partial \zeta_i} = x_i, \quad \frac{\partial y}{\partial \zeta_i} = y_i, \quad \frac{\partial z}{\partial \zeta_i} = z_i, \qquad i = 1, 2, 3, 4. \tag{9.17}$$

$$6V \frac{\partial \zeta_i}{\partial x} = a_i, \quad 6V \frac{\partial \zeta_i}{\partial y} = b_i, \quad 6V \frac{\partial \zeta_i}{\partial z} = c_i, \qquad i = 1, 2, 3, 4. \tag{9.18}$$

Partial derivatives of a function $F(\zeta_1, \zeta_2, \zeta_3, \zeta_4)$ with respect to Cartesian coordinates follow from (9.18) and the chain rule:

$$\begin{aligned} \frac{\partial F}{\partial x} &= \frac{\partial F}{\partial \zeta_i}\frac{\partial \zeta_i}{\partial x} = \frac{1}{6V}\left(\frac{\partial F}{\partial \zeta_1}a_1 + \frac{\partial F}{\partial \zeta_2}a_2 + \frac{\partial F}{\partial \zeta_3}a_3 + \frac{\partial F}{\partial \zeta_4}a_4\right) = \frac{a_i}{6V}\frac{\partial F}{\partial \zeta_i}, \\ \frac{\partial F}{\partial y} &= \frac{\partial F}{\partial \zeta_i}\frac{\partial \zeta_i}{\partial y} = \frac{1}{6V}\left(\frac{\partial F}{\partial \zeta_1}b_1 + \frac{\partial F}{\partial \zeta_2}b_2 + \frac{\partial F}{\partial \zeta_3}b_3 + \frac{\partial F}{\partial \zeta_4}b_4\right) = \frac{b_i}{6V}\frac{\partial F}{\partial \zeta_i}, \\ \frac{\partial F}{\partial z} &= \frac{\partial F}{\partial \zeta_i}\frac{\partial \zeta_i}{\partial z} = \frac{1}{6V}\left(\frac{\partial F}{\partial \zeta_1}c_1 + \frac{\partial F}{\partial \zeta_2}c_2 + \frac{\partial F}{\partial \zeta_3}c_3 + \frac{\partial F}{\partial \zeta_4}c_4\right) = \frac{c_i}{6V}\frac{\partial F}{\partial \zeta_i}. \end{aligned} \tag{9.19}$$

Here the summation convention over $i = 1, 2, 3, 4$ applies to the indexed expressions.

```
IsoTet4ShapeFunDer[xyztet_,numer_]:= Module[{
    x1,y1,z1,x2,y2,z2,x3,y3,z3,x4,y4,z4,
    x12,x13,x14,x23,x24,x34,x21,x31,x32,x42,x43,
    y12,y13,y14,y23,y24,y34,y21,y31,y32,y42,y43,
    z12,z13,z14,z23,z24,z34,z21,z31,z32,z42,z43,
    a1,a2,a3,a4,b1,b2,b3,b4,c1,c2,c3,c4,Nfx,Nyy,Nfz,Jdet},
  {{x1,y1,z1},{x2,y2,z2},{x3,y3,z3},{x4,y4,z4}}=xyztet;
   If [numer,{{x1,y1,z1},{x2,y2,z2},{x3,y3,z3},{x4,y4,z4}}=N[xyztet]];
   x12=x1-x2; x13=x1-x3; x14=x1-x4; x23=x2-x3; x24=x2-x4; x34=x3-x4;
   x21=-x12; x31=-x13; x41=-x14; x32=-x23; x42=-x24; x43=-x34;
   y12=y1-y2; y13=y1-y3; y14=y1-y4; y23=y2-y3; y24=y2-y4; y34=y3-y4;
   y21=-y12; y31=-y13; y41=-y14; y32=-y23; y42=-y24; y43=-y34;
   z12=z1-z2; z13=z1-z3; z14=z1-z4; z23=z2-z3; z24=z2-z4; z34=z3-z4;
   z21=-z12; z31=-z13; z41=-z14; z32=-z23; z42=-z24; z43=-z34;
   Jdet=x21*(y23*z34-y34*z23)+x32*(y34*z12-y12*z34)+x43*(y12*z23-y23*z12);
   a1=y42*z32-y32*z42; b1=x32*z42-x42*z32; c1=x42*y32-x32*y42;
   a2=y31*z43-y34*z13; b2=x43*z31-x13*z34; c2=x31*y43-x34*y13;
   a3=y24*z14-y14*z24; b3=x14*z24-x24*z14; c3=x24*y14-x14*y24;
   a4=y13*z21-y12*z31; b4=x21*z13-x31*z12; c4=x13*y21-x12*y31;
   Nfx={a1,a2,a3,a4}; Nfy={b1,b2,b3,b4}; Nfz={c1,c2,c3,c4};
   Return[{Nfx,Nfy,Nfz,Jdet}]];
```

FIGURE 9.4. *Mathematica* module `IsoTet4ShapeFunDer` that returns Cartesian partial derivatives of the linear tetrahedron shape functions (same as tetrahedral coordinates).

```
  ClearAll[x1,y1,z1,x2,y2,z2,x3,y3,z3,x4,y4,z4]; SeedRandom[27];
  RR={{Random[],Random[],Random[]},{Random[],Random[],Random[]},
      {Random[],Random[],Random[]},{Random[],Random[],Random[]}}-0.5;
  {{x1,y1,z1},{x2,y2,z2},{x3,y3,z3},{x4,y4,z4}}=Rationalize[RR,.01];
  encoor={{x1,y1,z1},{x2,y2,z2},{x3,y3,z3},{x4,y4,z4}};
  J={{1,1,1,1},{x1,x2,x3,x4},{y1,y2,y3,y4},{z1,z2,z3,z4}};
  Print["J=",J//MatrixForm];
  {acol,bcol,ccol,Jdet}=IsoTet4ShapeFunDer[encoor,False];
  Print["Jdet=",Jdet];
  V01=x2*(y3*z4-y4*z3)+x3*(y4*z2-y2*z4)+x4*(y2*z3-y3*z2);
  V02=x1*(y4*z3-y3*z4)+x3*(y1*z4-y4*z1)+x4*(y3*z1-y1*z3);
  V03=x1*(y2*z4-y4*z2)+x2*(y4*z1-y1*z4)+x4*(y1*z2-y2*z1);
  V04=x1*(y3*z2-y2*z3)+x2*(y1*z3-y3*z1)+x3*(y2*z1-y1*z2);
  Vcol={V01,V02,V03,V04}; Jinv=Transpose[{Vcol,acol,bcol,ccol}]/Jdet;
  Print["Jinv=",Jinv//MatrixForm];
  Print["check Jinv:",Simplify[Jinv.J]//MatrixForm];
```

FIGURE 9.5. Test statements to exercise module `IsoTet4ShapeFunDer` of Figure 9.4.

§9.1.9. Shape Function Derivatives Module

A *Mathematica* module called `IsoTet4ShapeFunDer`, which returns tetrahedral coordinate derivatives with respect to $\{x, y, z\}$, is listed in Figure 9.4, along with test statements. (The name of the module reflects the fact that tetrahedral coordinates are in fact the shape functions of the linear triangle.)

The module is referenced as

$$\{\,\texttt{Nfx,Nfy,Nfz,Jdet}\,\}\texttt{=IsoTet4ShapeFunDer[xyztet,numer]} \qquad (9.20)$$

$$\text{J} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ -\frac{1}{2} & \frac{1}{16} & -\frac{1}{34} & -\frac{3}{11} \\ -\frac{2}{9} & 0 & -\frac{1}{13} & -\frac{2}{9} \\ \frac{4}{11} & \frac{2}{13} & \frac{1}{3} & -\frac{1}{4} \end{pmatrix}$$

$$\text{Jdet} = \frac{11663081}{600709824}$$

$$\text{Jinv} = \begin{pmatrix} \frac{2168287}{11663081} & -\frac{42621040}{11663081} & \frac{58441383}{11663081} & \frac{3220932}{11663081} \\ \frac{18062408}{11663081} & -\frac{53559792}{11663081} & \frac{169329888}{11663081} & -\frac{19836960}{11663081} \\ -\frac{9787206}{11663081} & \frac{81914976}{11663081} & -\frac{178705683}{11663081} & \frac{30338880}{11663081} \\ \frac{1219592}{11663081} & \frac{14265856}{11663081} & -\frac{49065588}{11663081} & -\frac{13722852}{11663081} \end{pmatrix}$$

$$\text{check Jinv:} \quad \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

FIGURE 9.6. Results from running test script of Figure 9.5.

The arguments are:

`xyztet` Element node coordinates, supplied as the two-dimensional list `{ { x1,y1,z1 },{ x2,y2,z2 },{ x3,y3,z3 },{ x4,y4,z4 } }`.

`numer` A logical flag. `True` to carry out floating-point numeric computations, else `False`.

The function returns are:

`Nfx` List `{ a1,a2,a3,a4 }`, in which $a_i = 6V\dfrac{\partial\zeta_i}{\partial x} = J\dfrac{\partial\zeta_i}{\partial x}$. See (9.14) and (9.18).

`Nfy` List `{ b1,b2,b3,b4 }`, in which $b_i = 6V\dfrac{\partial\zeta_i}{\partial y} = J\dfrac{\partial\zeta_i}{\partial y}$. See (9.14) and (9.18).

`Nfz` List `{ c1,c2,c3,c4 }`, in which $c_i = 6V\dfrac{\partial\zeta_i}{\partial z} = J\dfrac{\partial\zeta_i}{\partial z}$. See (9.14) and (9.18).

`Jdet` Determinant of the Jacobian matrix in (9.10). Here `Jdet` $= J = |\mathbf{J}| = 6V$.

Note that it is convenient to return, say, a_i and not a_i/J to avoid zero-J exceptions inside `IsoTet4ShapeFunDer`. Consequently the module *always* returns the above data, even if `Jdet` is zero or negative. It is up to the calling program to proceed upon testing whether `Jdet` ≤ 0.

Test statements listed in Figure 9.5 generate a "random tetrahedron" using the built-in function `Random`, convert coordinate values to small fractions, execute `IsoTet4ShapeFunDer` in exact arithmetic, form the inverse Jacobian matrix with the return data (the first column is constructed as part of the test since the module does not return those values), and checks that it is the alleged inverse. The results of the test are shown in Figure 9.6.

§9.1.10. Analytical Integration

Exact analytical integration over the linear tetrahedron can be easily done if the integrand is a *polynomial expressed in tetrahedral coordinates*. Fortunately this restriction is met by most of the stiffness and force calculations discussed in §9.3 and §9.4 below. The basic procedure is: the polynomial is broken up into monomial terms, and each such term integrated using the general

```
IntegrateOverTetrahedron[expr_,{ζ1_,ζ2_,ζ3_,ζ4_},V_,max_]:=
  Module[{p,i,j,k,l,c,s=0}, p=Expand[expr];
   For [i=0,i<=max,i++, For[j=0,j<=max,j++,
        For [k=0,k<=max,k++, For [l=0,l<=max,l++,
        c=Coefficient[Coefficient[Coefficient[
          Coefficient[p,ζ1,i],ζ2,j],ζ3,k],ζ4,l];
        s+=6*c*(i!*j!*k!*l!)/((i+j+k+l+3)!) ]]]];
   ClearAll[p]; Return[Simplify[V*s]];
 ];

expr=c0+c1*ζ1+c2*ζ1*ζ2+c3*ζ1*ζ2*ζ3+c4*ζ1*ζ2^2*ζ4+c5*ζ1^2*ζ4^2;
Print[IntegrateOverTetrahedron[expr,{ζ1,ζ2,ζ3,ζ4},V,2]];
```

$$\frac{1}{840}\,(840\,\mathrm{c0} + 210\,\mathrm{c1} + 42\,\mathrm{c2} + 7\,\mathrm{c3} + 2\,\mathrm{c4} + 4\,\mathrm{c5})\,\mathrm{V}$$

FIGURE 9.7. *Mathematica* module that analytically integrates an arbitrary polynomial in tetrahedral coordinates over a tetrahedron domain. See text for usage instructions.

formula

$$\boxed{\int_{\Omega^e} \zeta_1^i\,\zeta_2^j\,\zeta_3^k\,\zeta_4^\ell\;d\Omega^e = \frac{i!\;j!\;k!\;\ell!}{(i+j+k+\ell+3)!}\,6V.} \tag{9.21}$$

Here i, j, k and ℓ are *nonnegative integers*, Ω^e is the tetrahedron domain, and V its volume measure. This formula is only valid for *constant metric* tetrahedra in the sense explained in the previous Chapter, but the linear tetrahedron does qualify as such. Special cases that often crop up:

$$\int_{\Omega^e} d\Omega^e = V, \qquad \int_{\Omega^e} \zeta_i\;d\Omega^e = \tfrac{1}{4}V, \qquad \int_{\Omega^e} \zeta_i\zeta_j\;d\Omega^e = \begin{cases} \frac{1}{10}V & \text{if } i=j, \\ \frac{1}{20}V & \text{if } i\neq j. \end{cases} \tag{9.22}$$

The *Mathematica* module `IntegrateOverTetrahedron`, listed in Figure 9.7, symbolically integrates an arbitrary polynomial expression in tetrahedral coordinates over the tetrahedron domain. The module is referenced as

```
int=IntegrateOverTetrahedron[expr, {ζ1,ζ2,ζ3,ζ4},V,max]
```
(9.23)

The arguments are:

`expr`	The polynomial expression to be integrated, in terms of tetrahedral coordinates.
`ζ1,ζ2,ζ3,ζ4`	A list of the symbols used in `expr` to represent the tetrahedral coordinates.
`V`	A symbol representing the tetrahedron volume.
`max`	An integer equal or greater than the highest tetrahedral coordinate exponent that appears in `expr`. Used to delimit internal loops.

The module name returns the integral as function value.

Example 9.2. Integrate the polynomial $F = c_0 + c_1\zeta_1 + c_2\zeta_1\zeta_2 + c_3\zeta_1\zeta_2\zeta_3 + c_4\zeta_1\zeta_2^2\zeta_4 + c_5\zeta_1^2\zeta_4^2$ over a linear tetrahedron of volume V. This is actually the test carried out at the bottom of the input cell in Figure 9.7. Upon setting up `expr`, the calling sequence is `IntegrateOverTetrahedron[expr,{ζ1,ζ2,ζ3,ζ4},V,2]`. Note that here `max` is 2 because that is the highest exponent that appears in a polynomial independent variable. The output cell in that Figure shows that the result is

$$\int_{\Omega^e} F\, d\Omega^e = \frac{V}{840}\Big(840c_0 + 210c_1 + 42c_2 + 7c_3 + 2c_4 + 7c_5\Big). \tag{9.24}$$

Example 9.3. Evaluate the second (geometric) moment of inertia $I_{xx} = \int_{\Omega^e} x^2\, d\Omega^e$ in terms of the node coordinates. To convert the integrand to a function of tetrahedral coordinates, replace $x = x_1\zeta_1 + x_2\zeta_2 + x_3\zeta_3 + x_4\zeta_4$ in x^2 to form `expr` as a quadratic polynomial. The appropriate *Mathematica* statements are `Ne={ζ1,ζ2,ζ3,ζ4}; x={x1,x2,x3,x4}.Ne; Print[IntegrateOverTetrahedron[x^2,Ne,V,2]];`. The result is

$$\int_{\Omega^e} x^2\, d\Omega^e = \frac{V}{10}\big(x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_1\,x_2 + x_1\,x_3 + x_1\,x_4 + x_2\,x_3 + x_2\,x_4 + x_3\,x_4\big). \tag{9.25}$$

§9.2. The Linear Tetrahedron As Iso-P Elasticity Element

The simplest tetrahedron finite element for problems of variational order $m = 1$ is the four-node tetrahedron with *linear shape functions*. The shape functions are simply the tetrahedral coordinates: $N_i = \zeta_i$, $i = 1, 2, 3, 4$. This finite element is derived now for the elasticity problem, using the Total Potential Energy (TPE) principle as source variational form, with displacements as master field.

§9.2.1. Displacement Interpolation

The displacement field over the tetrahedron is defined by the three components u_x, u_y and u_z. These are linearly interpolated from their node values, by making F equal to u_x, u_y and u_z in (9.9), and arranging it in matrix form:

$$\vec{\mathbf{u}} = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} = \begin{bmatrix} u_{x1} & u_{x2} & u_{x3} & u_{x4} \\ u_{y1} & u_{y2} & u_{y3} & u_{y4} \\ u_{z1} & u_{z2} & u_{z3} & u_{z4} \end{bmatrix} \begin{bmatrix} \zeta_1 \\ \zeta_2 \\ \zeta_3 \\ \zeta_4 \end{bmatrix}. \tag{9.26}$$

Combining this with (9.10) we have the *isoparametric definition* of the four-node tetrahedron as a displacement model:

$$\begin{bmatrix} 1 \\ x \\ y \\ z \\ u_x \\ u_y \\ u_z \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \\ z_1 & z_2 & z_3 & z_4 \\ u_{x1} & u_{x2} & u_{x3} & u_{x4} \\ u_{y1} & u_{y2} & u_{y3} & u_{y4} \\ u_{z1} & u_{z2} & u_{z3} & u_{z4} \end{bmatrix} \begin{bmatrix} \zeta_1 \\ \zeta_2 \\ \zeta_3 \\ \zeta_4 \end{bmatrix}. \tag{9.27}$$

Comparing this to the general iso-P definition given in the previous Chapter, it can be seen that the shape functions are simply $N_i = \zeta_i$.

The 12×1 node displacement vector is configured node-wise as

$$\mathbf{u}^e = [\, u_{x1} \quad u_{y1} \quad u_{z1} \quad u_{x2} \quad u_{y2} \quad u_{z2} \quad \cdots \quad u_{z4} \,]^T \,. \tag{9.28}$$

It is easily shown that the resulting element is C^0 inter-element conforming. It is also complete for the TPE functional, which has a variational index of one. Consequently, the element is *consistent* in the sense discussed in Chapter 19 of IFEM.

§9.2.2. The Strain Field

The element strain field is strongly connected to the displacements by the strain-displacement equations, which in indicial notation read

$$e_{ij} = \tfrac{1}{2}(u_{i,j} + u_{j,i}). \tag{9.29}$$

We translate this to matrix notation as follows. First, the six independent components of the stress tensor are arranged into a 6-component strain vector as follows:

$$\mathbf{e} = [\, e_{11} \quad e_{22} \quad e_{33} \quad 2e_{12} \quad 2e_{23} \quad 2e_{31} \,]^T = [\, e_{xx} \quad e_{yy} \quad e_{zz} \quad \gamma_{xy} \quad \gamma_{yz} \quad \gamma_{zy} \,]^T \,. \tag{9.30}$$

The second expression shows the engineering notation for the shear strains. Second, displacement components u_1, u_2 and u_3 are rewritten as u_x, u_y and u_z, collected into a 3-vector and linked to the displacement field by (9.26):

$$\mathbf{e} = \begin{bmatrix} e_{xx} \\ e_{yy} \\ e_{zz} \\ 2e_{xy} \\ 2e_{yz} \\ 2e_{zx} \end{bmatrix} = \begin{bmatrix} \partial/\partial x & 0 & 0 \\ 0 & \partial/\partial y & 0 \\ 0 & 0 & \partial/\partial z \\ \partial/\partial y & \partial/\partial x & 0 \\ 0 & \partial/\partial z & \partial/\partial y \\ \partial/\partial z & 0 & \partial/\partial x \end{bmatrix} \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} = \mathbf{D}\,\vec{\mathbf{u}}. \tag{9.31}$$

Combining this with (9.26) and using the partial differentiation rules (9.19) we obtain the matrix relation between strains and nodal displacements:

$$\mathbf{e} = \mathbf{B}\,\mathbf{u}^e. \tag{9.32}$$

If the node displacements are arranged node-by-node, as in (9.28), the matrix $\mathbf{B}$ has the following configuration:

$$\mathbf{B} = \frac{1}{6V} \begin{bmatrix} a_1 & 0 & 0 & a_2 & 0 & 0 & a_3 & 0 & 0 & a_4 & 0 & 0 \\ 0 & b_1 & 0 & 0 & b_2 & 0 & 0 & b_3 & 0 & 0 & b_4 & 0 \\ 0 & 0 & c_1 & 0 & 0 & c_2 & 0 & 0 & c_3 & 0 & 0 & c_4 \\ b_1 & a_1 & 0 & b_2 & a_2 & 0 & b_3 & a_3 & 0 & b_4 & a_4 & 0 \\ 0 & c_1 & b_1 & 0 & c_2 & b_2 & 0 & c_3 & b_3 & 0 & c_4 & b_4 \\ c_1 & 0 & a_1 & c_2 & 0 & a_2 & c_3 & 0 & a_3 & c_4 & 0 & a_4 \end{bmatrix}. \tag{9.33}$$

Note that this matrix is constant over the element.

§9.2.3. The Stress Field

The stress field is related to the stress field by the strong connection

$$\sigma_{ij} = E_{ijk\ell} e_{k\ell} \tag{9.34}$$

To convert this to matrix notation we rearrange the 6 independent stress components to correspond to the strains (9.30):

$$\boldsymbol{\sigma} = [\, \sigma_{11} \quad \sigma_{22} \quad \sigma_{33} \quad \sigma_{12} \quad \sigma_{23} \quad \sigma_{31} \,]^T = [\, \sigma_{xx} \quad \sigma_{yy} \quad \sigma_{zz} \quad \sigma_{xy} \quad \sigma_{yz} \quad \sigma_{zx} \,]^T . \tag{9.35}$$

If the material is linearly elastic and no initial strains are considered, the constitutive equation may be compactly expressed as

$$\boldsymbol{\sigma} = \mathbf{E}\,\mathbf{e}. \tag{9.36}$$

where the 6×6 elasticity matrix $\mathbf{E}$ is symmetric. For a general anisotropic material the expanded form of (9.36) is

$$\begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{xy} \\ \sigma_{yz} \\ \sigma_{zx} \end{bmatrix} = \begin{bmatrix} E_{11} & E_{12} & E_{13} & E_{14} & E_{15} & E_{16} \\ & E_{22} & E_{23} & E_{24} & E_{25} & E_{26} \\ & & E_{33} & E_{34} & E_{35} & E_{36} \\ & & & E_{44} & E_{45} & E_{46} \\ & & & & E_{55} & E_{56} \\ \text{symm} & & & & & E_{66} \end{bmatrix} \begin{bmatrix} e_{xx} \\ e_{yy} \\ e_{zz} \\ 2e_{xy} \\ 2e_{yz} \\ 2e_{zx} \end{bmatrix}, \tag{9.37}$$

in which E_{ij} are constitutive moduli. If the material is isotropic, with elastic modulus E and Poisson's ratio ν, the foregoing relation simplifies to

$$\begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{yz} \\ \sigma_{zx} \\ \sigma_{xy} \end{bmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2}-\nu & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2}-\nu & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2}-\nu \end{bmatrix}. \tag{9.38}$$

§9.3. The Element Stiffness Matrix

Introducing $\mathbf{e} = \mathbf{B}\mathbf{u}$ and $\boldsymbol{\sigma} = \mathbf{E}\mathbf{e}$ into the TPE functional restricted to the element volume and rendering the resulting algebraic form stationary with respect to the node displacements $\mathbf{u}^e$ we get the usual expression for the element stiffness matrix

$$\boxed{\mathbf{K}^e = \int_{\Omega^e} \mathbf{B}^T\,\mathbf{E}\,\mathbf{B}\; d\Omega^e.} \tag{9.39}$$

Assuming that elastic moduli do not vary over the element, the foregoing integrand is constant because matrix $\mathbf{B}$ as given by (9.33) is constant. Since $V = \int_{\Omega^e} d\Omega^e$ we get

$$\boxed{\mathbf{K}^e = V\,\mathbf{B}^T\,\mathbf{E}\,\mathbf{B}.} \tag{9.40}$$

This stiffness matrix is 12×12. It can be directly evaluated in closed form using the above expression or, equivalently, by a one-point (centroid) integration rule.

§9.4. The Consistent Node Force Vector

A tetrahedral mesh may be subjected to given body (volume) forces such as gravity, as well as to specified boundary tractions, such as pressure. Both kinds of loading have to be converted to node forces through an energy-based lumping procedure.

```
ClearAll[bz1,bz2,bz3,bz4,ζ1,ζ2,ζ3,ζ4,V];
Ne={{ζ1,0,0,ζ2,0,0,ζ3,0,0,ζ4,0,0},
    {0,ζ1,0,0,ζ2,0,0,ζ3,0,0,ζ4,0},
    {0,0,ζ1,0,0,ζ2,0,0,ζ3,0,0,ζ4}};
zetas={ζ1,ζ2,ζ3,ζ4};
bv={0,0,{bz1,bz2,bz3,bz4}.zetas};
b=Transpose[Ne].bv; fe=Table[0,{12}];
For [i=1,i<=12,i++,
    fe[[i]]=IntegrateOverTetrahedron[b[[i]],zetas,V,3]];
Print["consistent body force =",fe//MatrixForm];
```

FIGURE 9.8. *Mathematica* script for linearly varying z-body forces.

§9.4.1. Body Forces

A body load field over the element, such as gravity or centrifugal forces, is defined by its components

$$\mathbf{b} = [\, b_x \quad b_y \quad b_z \,]^T \, . \tag{9.41}$$

Inserting this into the TPE principle, the body force contribution gives

$$\boxed{\mathbf{f}^e = \int_{\Omega^e} \mathbf{N}^T \, \mathbf{b} \; d\Omega^e.} \tag{9.42}$$

Here $\mathbf{N}$ is the 3×12 shape function matrix that relates element field displacements to node displacements:

$$\vec{\mathbf{u}} = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} = \mathbf{N}\,\mathbf{u}^e. \tag{9.43}$$

For the node-wise displacement ordering (9.28) of the element freedoms, the body force matrix is

$$\mathbf{N} = \begin{bmatrix} \zeta_1 & 0 & 0 & \zeta_2 & 0 & 0 & \zeta_3 & 0 & 0 & \zeta_4 & 0 & 0 \\ 0 & \zeta_1 & 0 & 0 & \zeta_2 & 0 & 0 & \zeta_3 & 0 & 0 & \zeta_4 & 0 \\ 0 & 0 & \zeta_1 & 0 & 0 & \zeta_2 & 0 & 0 & \zeta_3 & 0 & 0 & \zeta_4 \end{bmatrix}. \tag{9.44}$$

Even if nonzero body forces are uniform, the integrand in (9.42) is not constant over the element. See the next example.

Example 9.4. Assume the constant gravity field: $b_x = 0$, $b_y = 0$ and $b_z = -\rho g$, where ρ (density) and g (acceleration of gravity) are constant over the element. Using the second of (9.22) we find

$$\mathbf{f}^e = -\tfrac{1}{4}\rho g V \; [\,0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1\,]^T . \tag{9.45}$$

This means that the total weight force acting along $-z$, which is $\rho g V$, must be divided in four and assigned equally to each corner node. This is the same result as element-by-element lumping.

Example 9.5. Assume a z-directed body force field that varies linearly over the element, which is given by

$$b_x = 0, \quad b_y = 0, \quad b_z = b_{z1}\zeta_1 + b_{z2}\zeta_2 + b_{z3}\zeta_3 + b_{z4}\zeta_4, \tag{9.46}$$

in which b_{zi} are the nodal values of b_z. Using the *Mathematica* script listed in Figure 9.8 we find that

$$f_{xi} = f_{yi} = 0, \quad f_{zi} = \frac{V}{20}(b_{z1} + b_{z2} + b_{z3} + b_{z4} + b_{zi}), \quad i = 1, 2, 3, 4. \tag{9.47}$$

If $b_{z1} = b_{z2} = b_{z3} = b_{z4} = -\rho g$ we recover the result of the previous example: $b_{zi} = -\rho\, g V/4$.

§9.4.2. Surface Tractions

The most practically important case is that of surface tractions normal to an element face. This models the effect of pressure loads. The calculation of node forces for the case of a constant pressure acting on a tetrahedron face is the matter of one exercise.

§9.5. Element Implementation

This section covers the implementation of the stiffness matrix, body force and stress computations for the linear tetrahedron. If the language provides for high level matrix operations, such as *Mathematica* and *Matlab* do, the resulting code is quite compact.

§9.5.1. Element Stiffness Matrix

An implementation of the stiffness matrix computations as a *Mathematica* module is listed in Figure 9.9. The module is invoked as

```
Ke=IsoTet4Stiffness[xyztet,Emat,{ },options];
```
(9.48)

The arguments are

`xyztet` Element node coordinates, arranged as a two-dimensional list: `{{x1,y1,z1},{x2,y2,z2},{x3,y3,z3},{x4,y4,z4}}`.

Emat The 6×6 matrix of elastic moduli (9.37) provided as a two-dimensional list: `{{E11,E12,E13,E14,E15,E16}, ... {E16,E26,E36,E46,E56,E66}}`.

`options` A list containing optional additional information. For this element: `options={numer,,e}`, in which

- `numer` A logical flag: `True` to specify floating-point numeric work, `False` to request exact calculations. If omitted, `False` is assumed.
- `e` Element number. Only used in error messages. If omitted, 0 is assumed.

The third argument is a placeholder and should be set to the empty list { }.

```
IsoTet4Stiffness[xyztet_,Emat_,{},options_]:= Module[{
  e=0,Nfx,Nfy,Nfz,Jdet,a1,a2,a3,a4,b1,b2,b3,b4,c1,c2,c3,c4,
  numer=False,Be,Ke},
  If [Length[options]>=1, numer=options[[1]]];
  If [Length[options]>=3, e=options[[3]]];
  {Nfx,Nfy,Nfz,Jdet}=IsoTet4ShapeFunDer[xyztet,numer];
  If [numer&&(Jdet<=0), Print["IsoTet4Stiffness: Neg "
     "or zero Jacobian, element," e]; Return[Null]];
  {a1,a2,a3,a4}=Nfx; {b1,b2,b3,b4}=Nfy; {c1,c2,c3,c4}=Nfz;
  Be={{a1,0,0,  a2,0,0,  a3,0,0,  a4,0,0},
      {0,b1,0,  0,b2,0,  0,b3,0,  0,b4,0},
      {0,0,c1,  0,0,c2,  0,0,c3,  0,0,c4},
      {b1,a1,0, b2,a2,0, b3,a3,0, b4,a4,0},
      {0,c1,b1, 0,c2,b2, 0,c3,b3, 0,c4,b4},
      {c1,0,a1, c2,0,a2, c3,0,a3, c4,0,a4}};
  Ke=(1/(6*Jdet))*Transpose[Be].(Emat.Be);
  If [!numer,Ke=Simplify[Ke]]; Return[Ke] ];
```

FIGURE 9.9. *Mathematica* module to compute the stiffness matrix of the linear tetrahedron for elasticity.

```
ClearAll[Em,ν]; Em=480; ν=1/3;
Emat=Em/((1+ν)*(1-2*ν))*{{1-ν,ν,ν,0,0,0},
  {ν,1-ν,ν,0,0,0},{ν,ν,1-ν,0,0,0},{0,0,0,1/2-ν,0,0},
  {0,0,0,0,1/2-ν,0},{0,0,0,0,0, 1/2-ν}};
xyztet={{2,3,4},{6,3,2},{2,5,1},{4,3,6}};
Ke=IsoTet4Stiffness[xyztet,Emat,{},{False}];
Print["Ke=",Ke//MatrixForm];
Print["eigs of Ke=",Chop[Eigenvalues[N[Ke]]]];
```

FIGURE 9.10. Test script to exercise the stiffness matrix module listed in Figure 9.9.

$$
\mathrm{Ke} = \begin{pmatrix}
745 & 540 & 120 & -5 & 30 & 60 & -270 & -240 & 0 & -470 & -330 & -180 \\
540 & 1720 & 270 & -120 & 520 & 210 & -120 & -1080 & -60 & -300 & -1160 & -420 \\
120 & 270 & 565 & 0 & 150 & 175 & 0 & -120 & -270 & -120 & -300 & -470 \\
-5 & -120 & 0 & 145 & -90 & -60 & -90 & 120 & 0 & -50 & 90 & 60 \\
30 & 520 & 150 & -90 & 220 & 90 & 60 & -360 & -60 & 0 & -380 & -180 \\
60 & 210 & 175 & -60 & 90 & 145 & 0 & -120 & -90 & 0 & -180 & -230 \\
-270 & -120 & 0 & -90 & 60 & 0 & 180 & 0 & 0 & 180 & 60 & 0 \\
-240 & -1080 & -120 & 120 & -360 & -120 & 0 & 720 & 0 & 120 & 720 & 240 \\
0 & -60 & -270 & 0 & -60 & -90 & 0 & 0 & 180 & 0 & 120 & 180 \\
-470 & -300 & -120 & -50 & 0 & 0 & 180 & 120 & 0 & 340 & 180 & 120 \\
-330 & -1160 & -300 & 90 & -380 & -180 & 60 & 720 & 120 & 180 & 820 & 360 \\
-180 & -420 & -470 & 60 & -180 & -230 & 0 & 240 & 180 & 120 & 360 & 520
\end{pmatrix}
$$

eigs of Ke = {3885.87, 1006.81, 986.364, 214.716, 106.822, 99.4105, 0, 0, 0, 0, 0, 0}

FIGURE 9.11. Results from running test script of Figure 9.10.

The stiffness module calls `IsoTet4ShapeFunDer`, which was described in §9.1.5, and listed in Figure 9.4, to get the Cartesian partial derivatives of the shape functions (same as tetrahedral coordinates for this element) as well as the Jacobian determinant. Note that in the implementation of Figure 9.9, `Be` is $J\,\mathbf{B}^e$ and not $\mathbf{B}^e$. The correct scaling is restored in the `Ke=(1/(6*Jdet))*Transpose[Be].(Emat.Be)` statement, since $V/J^2 = 1/(6J)$.

Module `IsoTet4Stiffness` is exercised by the *Mathematica* script listed in Figure 9.10, which forms the stiffness matrix of a linear tetrahedron with corner coordinates

$$\texttt{xyztet=\{\{2,3,4\},\{6,3,2\},\{2,5,1\},\{4,3,6\}\}.} \tag{9.49}$$

Its volume is $+24$. The material is isotropic with $E = 480$ and $\nu = 1/3$. The results are shown in Figure 9.11. The computation of stiffness matrix eigenvalues is always a good programming test, since six eigenvalues (associated with rigid body modes) must be exactly zero while the other six must be real and positive. This is verified by the results shown in Figure 9.11.

§9.5.2. *The Unit Reference Tetrahedron

The *unit reference tetrahedron has corners* at { 0,0,0 }, { 1,0,0 }, { 0,1,0 }, { 0,0,1 } and volume $V = 1/6$. This geometry is of interest since the stiffness matrix and its eigenvalues can be obtained in explicit symbolic form for an isotropic material. For modulus of elasticity E and Poisson's ratio ν, the stiffness matrix is

$$\mathbf{K}^e = \hat{E}\begin{bmatrix} 4-6\nu & 1 & 1 & -2\hat{\nu} & -\tilde{\nu} & -\tilde{\nu} & -\tilde{\nu} & -2\nu & 0 & -\tilde{\nu} & 0 & -2\nu \\ 1 & 4-6\nu & 1 & -2\nu & -\tilde{\nu} & 0 & -\tilde{\nu} & -2\hat{\nu} & -\tilde{\nu} & 0 & -\tilde{\nu} & -2\nu \\ 1 & 1 & 4-6\nu & -2\nu & 0 & -\tilde{\nu} & 0 & -2\nu & -\tilde{\nu} & -\tilde{\nu} & -\tilde{\nu} & -2\hat{\nu} \\ -2\hat{\nu} & -2\nu & -2\nu & 2\hat{\nu} & 0 & 0 & 0 & 2\nu & 0 & 0 & 0 & 2\nu \\ -\tilde{\nu} & -\tilde{\nu} & 0 & 0 & \tilde{\nu} & 0 & \tilde{\nu} & 0 & 0 & 0 & 0 & 0 \\ -\tilde{\nu} & 0 & -\tilde{\nu} & 0 & 0 & \tilde{\nu} & 0 & 0 & 0 & \tilde{\nu} & 0 & 0 \\ -\tilde{\nu} & -\tilde{\nu} & 0 & 0 & \tilde{\nu} & 0 & \tilde{\nu} & 0 & 0 & 0 & 0 & 0 \\ -2\nu & -2\hat{\nu} & -2\nu & 2\nu & 0 & 0 & 0 & 2\hat{\nu} & 0 & 0 & 0 & 2\nu \\ 0 & -\tilde{\nu} & -\tilde{\nu} & 0 & 0 & 0 & 0 & 0 & \tilde{\nu} & 0 & \tilde{\nu} & 0 \\ -\tilde{\nu} & 0 & -\tilde{\nu} & 0 & 0 & \tilde{\nu} & 0 & 0 & 0 & \tilde{\nu} & 0 & 0 \\ 0 & -\tilde{\nu} & -\tilde{\nu} & 0 & 0 & 0 & 0 & 0 & \tilde{\nu} & 0 & \tilde{\nu} & 0 \\ -2\nu & -2\nu & -2\hat{\nu} & 2\nu & 0 & 0 & 0 & 2\nu & 0 & 0 & 0 & 2\hat{\nu} \end{bmatrix}, \tag{9.50}$$

in which $\hat{E} = E/(12(1-2\nu)(1+\nu))$, $\tilde{\nu} = 1 - 2\nu$ and $\hat{\nu} = 1 - \nu$. The eigenvalues of $\mathbf{K}^e$ are

$$\hat{E}\,(5-4\nu \pm \sqrt{9-24\nu+48\nu^2}), \quad 5\hat{E}\,(1-2\nu)\ \text{(two)}, \quad 2\hat{E}\,(1-2\nu)\ \text{(two)}, \quad 0\ \text{(six)}. \tag{9.51}$$

As $\nu \to 1/2$ (incompressible material) five nonzero eigenvalues tend to $8E/3$, $10E/3$ (twice), and $4E/3$ (twice). The largest one, which is $\hat{E}(5-4\nu + \sqrt{9-24\nu+48\nu^2})$, goes to infinity.

§9.5.3. Element Consistent Force Vector For Body Forces

This is the matter of one Exercise.

§9.5.4. Element Stresses

This is the matter of one Exercise.

§9.5.5. *Unrolled Loop Implementation

The computer time spent into element formation becomes an issue when going to 3D models. The main reason is that 3D models tend to contain lots of elements.[7] On the other hand solution methods for such monsters meshes tend to be iterative, which often work well in 3D. So there is an incentive to cut down on formation time. The issue is examined here for the linear tetrahedron to focus the ideas, but the problem becomes far more serious for numerically integrated higher order elements.

An operation count on the element stiffness computation done in `IsoTet4Stiffness` reveals that over 90% of the total arithmetic computation effort is carried out in the V-scaled $\mathbf{B}^T\mathbf{E}\mathbf{B}$ matrix product, implemented in the module listed in Figure 9.9 as `Ke=(1/(6*Jdet))*Transpose[Be].(Emat.Be)`. An important question for production level FEM programs is: how can this effort be reduced?

For low-level *compiled languages*, such as Fortran or C, there is a well known speed-up technique: *loop unrolling*. This is best illustrated with an example. Unrolling the three loops involved in the $\mathbf{B}^T\mathbf{E}\mathbf{B}$ computation for `IsoTet4Stiffness` results in the module `IsoTet4StiffnessUnrolled` listed in Figures 9.12 and 9.13. As can be seen the code "explodes" from 14 lines to well over 100, but (1) all loops as well as indexing are gone, (2) zero entries in in $\mathbf{B}$ are explicitly accounted for, and (3) symmetry of $\mathbf{K}^e$ is taken advantage of. The number of multiplications is roughly cut by 4 and no indexing computations are needed.

Is this "decompression" useful? Yes and no.

(To be completed)

Notes and Bibliography

The linear tetrahedron for structural and solid mechanics was concurrently developed by several academic researchers, as well as teams in the aerospace industry, during the early and mid 1960s. These include: Argyris (team at ISD, University of Stuttgart, Germany), Gallagher (team at Bell Aerospace), Melosh (Boeing, Ph. D. Thesis at University of Washington under Harold Martin), and Rashid (UC Berkeley and Anatech Corporation). The first journal publication was by Gallagher, Padlog and Bijlard in 1962 [290], closely followed by Melosh in 1963 [492], and Argyris in 1965 [27,29]. Rashid's 1964 thesis under Ray Clough focused on the axisymmetric solid case while his extensions to general geometries came out in the late 1960s [625,626].

Early (pre-1965) formulations used Cartesian coordinates, and as a consequence element development took a inordinately long time. Tetrahedral natural coordinates represent an immediate generalization of triangular natural coordinates (called barycentric or Möbius coordinates in the mathematical literature), a brief history of which is given in Chapter 15 of IFEM.

The analytical integration rule (9.21) can be derived using the generalized beta function and repeated integration by parts. It is unclear where it appeared first.

Tetrahedral elements were put in the shadow of hexahedral (brick) elements when the latter appeared around 1967 as isoparametric models through the work of Irons [401]. For hand-constructed meshes, there is not much point in using tetrahedra unless geometrical complexities call for their use for rounding corners, etc. Revival came in the form of "space filling", 3D mesh generators. As of this writing, linear tetrahedra are still heavily used for applications where master field derivatives (such as strains and stresses in solid mechanics) are of secondary interest.

[7] The 2007 element count record (for a physics aplication) was about 500 million tets, and billion-element models will undoubtedly happen.

```
IsoTet4StiffnessFast[xyztet_,Emat_,{},options_]:= Module[{
  e=0,Nfx,Nfy,Nfz,Jdet,a1,a2,a3,a4,b1,b2,b3,b4,c1,c2,c3,c4,
  E11,E12,E13,E14,E15,E16,E22,E23,E24,E25,E26,
  E33,E34,E35,E36,E44,E45,E46,E55,E56,E66,
  A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12,
  B1,B2,B3,B4,B5,B6,B7,B8,B9,B10,B11,B12,
  C1,C2,C3,C4,C5,C6,C7,C8,C9,C10,C11,C12,
  F1,F2,F3,F4,F5,F6,F7,F8,F9,F10,F11,F12,
  G1,G2,G3,G4,G5,G6,G7,G8,G9,G10,G11,G12,
  H1,H2,H3,H4,H5,H6,H7,H8,H9,H10,H11,H12,numer=False,Ke},
  If [Length[options]>=1, numer=options[[1]]];
  If [Length[options]>=3, e=options[[3]]];
  {Nfx,Nfy,Nfz,Jdet}=IsoTet4ShapeFunDer[xyztet,numer];
  {a1,a2,a3,a4}=Nfx; {b1,b2,b3,b4}=Nfy; {c1,c2,c3,c4}=Nfz;
  If [numer,{{a1,a2,a3,a4},{b1,b2,b3,b4},{c1,c2,c3,c4},Jdet}=
         N[{{a1,a2,a3,a4},{b1,b2,b3,b4},{c1,c2,c3,c4},Jdet}]];
  If [numer&&(Jdet<=0), Print["IsoTet4Stiffness: Neg "
     "or zero Jacobian, element," e]; Return[Null]];
 {{E11,E12,E13,E14,E15,E16},{E12,E22,E23,E24,E25,E26},
  {E13,E23,E33,E34,E35,E36},{E14,E24,E34,E44,E45,E46},
  {E15,E25,E35,E45,E55,E56},{E16,E26,E36,E46,E56,E66}}=Emat;
  {{A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12},
   {B1,B2,B3,B4,B5,B6,B7,B8,B9,B10,B11,B12},
   {C1,C2,C3,C4,C5,C6,C7,C8,C9,C10,C11,C12},
   {F1,F2,F3,F4,F5,F6,F7,F8,F9,F10,F11,F12},
   {G1,G2,G3,G4,G5,G6,G7,G8,G9,G10,G11,G12},
   {H1,H2,H3,H4,H5,H6,H7,H8,H9,H10,H11,H12}}=
  {{a1*E11+b1*E14+c1*E16,b1*E12+a1*E14+c1*E15,c1*E13+b1*E15+a1*E16,
   a2*E11+b2*E14+c2*E16,b2*E12+a2*E14+c2*E15,c2*E13+b2*E15+a2*E16,
   a3*E11+b3*E14+c3*E16,b3*E12+a3*E14+c3*E15,c3*E13+b3*E15+a3*E16,
   a4*E11+b4*E14+c4*E16,b4*E12+a4*E14+c4*E15,c4*E13+b4*E15+a4*E16},
  {a1*E12+b1*E24+c1*E26,b1*E22+a1*E24+c1*E25,c1*E23+b1*E25+a1*E26,
   a2*E12+b2*E24+c2*E26,b2*E22+a2*E24+c2*E25,c2*E23+b2*E25+a2*E26,
   a3*E12+b3*E24+c3*E26,b3*E22+a3*E24+c3*E25,c3*E23+b3*E25+a3*E26,
   a4*E12+b4*E24+c4*E26,b4*E22+a4*E24+c4*E25,c4*E23+b4*E25+a4*E26},
  {a1*E13+b1*E34+c1*E36,b1*E23+a1*E34+c1*E35,c1*E33+b1*E35+a1*E36,
   a2*E13+b2*E34+c2*E36,b2*E23+a2*E34+c2*E35,c2*E33+b2*E35+a2*E36,
   a3*E13+b3*E34+c3*E36,b3*E23+a3*E34+c3*E35,c3*E33+b3*E35+a3*E36,
   a4*E13+b4*E34+c4*E36,b4*E23+a4*E34+c4*E35,c4*E33+b4*E35+a4*E36},
  {a1*E14+b1*E44+c1*E46,b1*E24+a1*E44+c1*E45,c1*E34+b1*E45+a1*E46,
   a2*E14+b2*E44+c2*E46,b2*E24+a2*E44+c2*E45,c2*E34+b2*E45+a2*E46,
   a3*E14+b3*E44+c3*E46,b3*E24+a3*E44+c3*E45,c3*E34+b3*E45+a3*E46,
   a4*E14+b4*E44+c4*E46,b4*E24+a4*E44+c4*E45,c4*E34+b4*E45+a4*E46},
  {a1*E15+b1*E45+c1*E56,b1*E25+a1*E45+c1*E55,c1*E35+b1*E55+a1*E56,
   a2*E15+b2*E45+c2*E56,b2*E25+a2*E45+c2*E55,c2*E35+b2*E55+a2*E56,
   a3*E15+b3*E45+c3*E56,b3*E25+a3*E45+c3*E55,c3*E35+b3*E55+a3*E56,
   a4*E15+b4*E45+c4*E56,b4*E25+a4*E45+c4*E55,c4*E35+b4*E55+a4*E56},
  {a1*E16+b1*E46+c1*E66,b1*E26+a1*E46+c1*E56,c1*E36+b1*E56+a1*E66,
   a2*E16+b2*E46+c2*E66,b2*E26+a2*E46+c2*E56,c2*E36+b2*E56+a2*E66,
   a3*E16+b3*E46+c3*E66,b3*E26+a3*E46+c3*E56,c3*E36+b3*E56+a3*E66,
   a4*E16+b4*E46+c4*E66,b4*E26+a4*E46+c4*E56,c4*E36+b4*E56+a4*E66}};
```

FIGURE 9.12. Unrolled loop implementation of stiffness matrix module, Part 1.

```
Ke=Table[0,{12},{12}];
Ke[[1]]={a1*A1+b1*F1+c1*H1,a1*A2+b1*F2+c1*H2,
 a1*A3+b1*F3+c1*H3,a1*A4+b1*F4+c1*H4,a1*A5+b1*F5+c1*H5,
 a1*A6+b1*F6+c1*H6,a1*A7+b1*F7+c1*H7,a1*A8+b1*F8+c1*H8,
 a1*A9+b1*F9+c1*H9,a1*A10+b1*F10+c1*H10,
 a1*A11+b1*F11+c1*H11,a1*A12+b1*F12+c1*H12};
Ke[[2]]={Ke[[1,2]],b1*B2+a1*F2+c1*G2,
 b1*B3+a1*F3+c1*G3,b1*B4+a1*F4+c1*G4,b1*B5+a1*F5+c1*G5,
 b1*B6+a1*F6+c1*G6,b1*B7+a1*F7+c1*G7,b1*B8+a1*F8+c1*G8,
 b1*B9+a1*F9+c1*G9,b1*B10+a1*F10+c1*G10,
 b1*B11+a1*F11+c1*G11,b1*B12+a1*F12+c1*G12};
Ke[[3]]={Ke[[1,3]],Ke[[2,3]],c1*C3+b1*G3+a1*H3,
 c1*C4+b1*G4+a1*H4,c1*C5+b1*G5+a1*H5,c1*C6+b1*G6+a1*H6,
 c1*C7+b1*G7+a1*H7,c1*C8+b1*G8+a1*H8,c1*C9+b1*G9+a1*H9,
 c1*C10+b1*G10+a1*H10,c1*C11+b1*G11+a1*H11,
 c1*C12+b1*G12+a1*H12};
Ke[[4]]={Ke[[1,4]],Ke[[2,4]],Ke[[3,4]],a2*A4+b2*F4+c2*H4,
 a2*A5+b2*F5+c2*H5,a2*A6+b2*F6+c2*H6,a2*A7+b2*F7+c2*H7,
 a2*A8+b2*F8+c2*H8,a2*A9+b2*F9+c2*H9,A10*a2+b2*F10+c2*H10,
 A11*a2+b2*F11+c2*H11,A12*a2+b2*F12+c2*H12};
Ke[[5]]={Ke[[1,5]],Ke[[2,5]],Ke[[3,5]],Ke[[4,5]],
 b2*B5+a2*F5+c2*G5,b2*B6+a2*F6+c2*G6,b2*B7+a2*F7+c2*G7,
 b2*B8+a2*F8+c2*G8,b2*B9+a2*F9+c2*G9,B10*b2+a2*F10+c2*G10,
 B11*b2+a2*F11+c2*G11,B12*b2+a2*F12+c2*G12};
Ke[[6]]={Ke[[1,6]],Ke[[2,6]],Ke[[3,6]],Ke[[4,6]],Ke[[5,6]],
 c2*C6+b2*G6+a2*H6,c2*C7+b2*G7+a2*H7,c2*C8+b2*G8+a2*H8,
 c2*C9+b2*G9+a2*H9,C10*c2+b2*G10+a2*H10,
 C11*c2+b2*G11+a2*H11,C12*c2+b2*G12+a2*H12};
Ke[[7]]={Ke[[1,7]],Ke[[2,7]],Ke[[3,7]],Ke[[4,7]],Ke[[5,7]],
 Ke[[6,7]],a3*A7+b3*F7+c3*H7,a3*A8+b3*F8+c3*H8,
 a3*A9+b3*F9+c3*H9, A10*a3+b3*F10+c3*H10,
 A11*a3+b3*F11+c3*H11,A12*a3+b3*F12+c3*H12};
Ke[[8]]={Ke[[1,8]],Ke[[2,8]],Ke[[3,8]],Ke[[4,8]],Ke[[5,8]],
 Ke[[6,8]],Ke[[7,8]],b3*B8+a3*F8+c3*G8,b3*B9+a3*F9+c3*G9,
 B10*b3+a3*F10+c3*G10,B11*b3+a3*F11+c3*G11,
 B12*b3+a3*F12+c3*G12};
Ke[[9]]={Ke[[1,9]],Ke[[2,9]],Ke[[3,9]],Ke[[4,9]],Ke[[5,9]],
 Ke[[6,9]],Ke[[7,9]],Ke[[8,9]],c3*C9+b3*G9+a3*H9,
 C10*c3+b3*G10+a3*H10,C11*c3+b3*G11+a3*H11,C12*c3+b3*G12+a3*H12};
Ke[[10]]={Ke[[1,10]],Ke[[2,10]],Ke[[3,10]],Ke[[4,10]],
 Ke[[5,10]],Ke[[6,10]],Ke[[7,10]],Ke[[8,10]],Ke[[9,10]],
 A10*a4+b4*F10+c4*H10,A11*a4+b4*F11+c4*H11,A12*a4+b4*F12+c4*H12};
Ke[[11]]={Ke[[1,11]],Ke[[2,11]],Ke[[3,11]],Ke[[4,11]],
 Ke[[5,11]],Ke[[6,11]],Ke[[7,11]],Ke[[8,11]],Ke[[9,11]],
 Ke[[10,11]],B11*b4+a4*F11+c4*G11,B12*b4+a4*F12+c4*G12};
Ke[[12]]={Ke[[1,12]],Ke[[2,12]],Ke[[3,12]],Ke[[4,12]],
 Ke[[5,12]],Ke[[6,12]],Ke[[7,12]],Ke[[8,12]],Ke[[9,12]],
 Ke[[10,12]],Ke[[11,12]],C12*c4+b4*G12+a4*H12};
Ke=(1/(6*Jdet))*Ke;
If [!numer,Ke=Simplify[Ke]]; Return[Ke] ];
```

FIGURE 9.13. Unrolled loop implementation of stiffness matrix module, Part 2.

Homework Exercises for Chapter 9

The Linear Tetrahedron

EXERCISE 9.1 [T:5] A glance at the calling sequence of the stiffness computation module makes plain that a materially homogeneous tetrahedron (that is, one with constant $\mathbf{E}$) does not require fabrication properties (such as, for instance, the thickness of a plane stress element, or the area of a bar element.) Can you guess why? What could happen if the material is constant within layers, as in composites?

EXERCISE 9.2 [A:20] Given the corner Cartesian coordinates of a linear tetrahedron, how can the areas A_i ($i = 1, 2, 3, 4$) of the four faces be directly expressed in terms of them? Apply your result to compute the 4 face areas for a linear tetrahedron with 4 corners given by the Cartesian position coordinates at $(0, 0, 0)$, $(1, 0, 0)$, $(0, 1, 0)$ and $(0, 0, 1)$, respectively.

Hint: There are three ways to approach this problem (pick one):

1. Longest: express the element volume as $V = \frac{1}{3} A_i\, h_i$, where h_i are the corner-to-opposite-face distances, also called tetrahedron altitudes, and use the second property noted in §9.1.7. The tetrahedron volume can be computed directly by the determinant formula (9.3).
2. Quickest: use cross product of two face sides.
3. Most elegant: project the face area onto the 3 Cartesian frame planes, and use a formula of analytical geometry that says that the square of the face area is the sum of the squares of those projections. In this way show that $A_i = S_i/2$, where S_i is defined in §9.1.7.

EXERCISE 9.3 [A:15] Consider the function defined in tetrahedral natural coordinates as

$$F_b = \zeta_1\, \zeta_2\, \zeta_3\, \zeta_4. \tag{E9.1}$$

This is called a *bubble function* in the FEM literature. It vanishes on all faces.

(a) Show that its partial derivatives $\partial F_b/\partial x$, $\partial F_b/\partial y$, and $\partial F_b/\partial z$ vanish on all edges. *Hint*: read §9.1.8.

(b) Show that the volume integral $\int_{\Omega^e} F_b \; d\Omega$ evaluates to $V/840$. *Hint*: read §9.1.10.

EXERCISE 9.4 [C:20] Implement a body force computation module `IsoTet4BodyForce` that returns the consistent node force vector corresponding to a body force field that varies linear over the element and is defined by the 12 nodal values `{{bx1,by1,bz1},{bx2,by2,bz2},{bx3,by3,bz3,{bx4,by4,bz4}}`. Hint: use the result (9.47) extended to cover the x and y components.

EXERCISE 9.5 [C:20] Implement a stress computation module `IsoTet4Stress` that returns the element stress field (which is constant), given the element node displacements.

EXERCISE 9.6 [A:20] The i^{th} face (face opposite the i^{th} corner) of a linear tetrahedron is under uniform pressure p acting normal to the face (positive pressure: $+p$ means it points *into* the element). Compute the associated node force vector $\mathbf{f}^e$ by node-by-node (NbN) lumping.[8] It is sufficient to work out the expression for face 1, that is, $i = 1$.

Hint: Use the expression of §9.1.7, and the results of Exercise 9.2 for the face area.

[8] For those unfamiliar with NbN lumping: multiply the pressure by the face area to get a total force $F_p = p\, A_i$ acting on the centroid of the face, with F_p positive if going into the element. Then get its $\{x, y, z\}$ Cartesian components on scaling by the 3 direction cosines of the interior normal-to-the-face. Finally, assign 1/3 of each component to the face corner nodes; e.g., if $i = 1$ assign to 2, 3, 4, and so on.