Zero Energy Modes in One Dimension: An Introduction to "Hourglass" Modes

David J. Benson

March 9, 2003

Reduced integration does a lot of good things for an element: it reduces the computational cost, it reduces shear and volume locking, and it generally softens the element so that the predicted stress is more accurate. However, reduced integration also makes elements too soft in the sense that modes other than rigid body modes aren't resisted by the element. These modes, which were originally noticed in finite difference calculations in two dimensions in the 1960s, are historically called *hourglass* or *keystone* modes because of their shape. For other elements, the modes don't have these shapes, and the modes are commonly referred to as *zero energy* modes.

1 Mathematical Preliminaries: Vectors and Covectors

When we are working in cartesian space, such as (x_1, x_2) with the basis vectors $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$, an arbitrary vector in the space, $\boldsymbol{a}$, can be written

$$\boldsymbol{a} = (\boldsymbol{a} \cdot \boldsymbol{e}_1)\boldsymbol{e}_1 + (\boldsymbol{a} \cdot \boldsymbol{e}_2)\boldsymbol{e}_2 = a_1\boldsymbol{e}_1 + a_2\boldsymbol{e}_2. \tag{1}$$

Suppose, however, that two other basis vectors, $\hat{\boldsymbol{e}}_1$ and $\hat{\boldsymbol{e}}_2$, are chosen and they aren't perpendicular to each other and, just to make things a little more complicated, they don't have a unit length. Expressing $\boldsymbol{a}$ in terms of these new basis vectors isn't as simple as it was for $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$. Our goal is to determine the coefficients $\hat{a}_1$ and $\hat{a}_2$ so that

$$\boldsymbol{a} = \hat{a}_1\hat{\boldsymbol{e}}_1 + \hat{a}_2\boldsymbol{e}_2. \tag{2}$$

The most direct approach is to solve two linear equations for the coefficients which are generated by taking the dot product of Equation 2 with $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$, namely

$$\begin{bmatrix} \hat{\boldsymbol{e}}_1 \cdot \boldsymbol{e}_1 & \hat{\boldsymbol{e}}_2 \cdot \boldsymbol{e}_1 \\ \hat{\boldsymbol{e}}_1 \cdot \boldsymbol{e}_2 & \hat{\boldsymbol{e}}_2 \cdot \boldsymbol{e}_2 \end{bmatrix} \begin{Bmatrix} \hat{a}_1 \\ \hat{a}_2 \end{Bmatrix} = \begin{Bmatrix} a_1 \\ a_2 \end{Bmatrix}. \tag{3}$$

The solution is

$$\frac{1}{\det} \begin{bmatrix} \hat{\boldsymbol{e}}_2 \cdot \boldsymbol{e}_2 & -\hat{\boldsymbol{e}}_2 \cdot \boldsymbol{e}_1 \\ -\hat{\boldsymbol{e}}_1 \cdot \boldsymbol{e}_2 & \hat{\boldsymbol{e}}_1 \cdot \boldsymbol{e}_1 \end{bmatrix} \begin{Bmatrix} a_1 \\ a_2 \end{Bmatrix} = \begin{Bmatrix} \hat{a}_1 \\ \hat{a}_2 \end{Bmatrix}. \tag{4}$$

The solution can be re-written in the form of a vector dot product if we define the vectors $\hat{\boldsymbol{E}}_1$ and $\hat{\boldsymbol{E}}_2$ as

$$\hat{\boldsymbol{E}}_1 = \frac{1}{\det} \begin{Bmatrix} \hat{\boldsymbol{e}}_2 \cdot \boldsymbol{e}_2 \\ -\hat{\boldsymbol{e}}_2 \cdot \boldsymbol{e}_1 \end{Bmatrix} \quad \hat{\boldsymbol{E}}_2 = \frac{1}{\det} \begin{Bmatrix} -\hat{\boldsymbol{e}}_1 \cdot \boldsymbol{e}_2 \\ \hat{\boldsymbol{e}}_1 \cdot \boldsymbol{e}_1 \end{Bmatrix} \tag{5}$$

$$\hat{a}_1 = \boldsymbol{a} \cdot \hat{\boldsymbol{E}}_1 \quad \hat{a}_2 = \boldsymbol{a} \cdot \hat{\boldsymbol{E}}_2. \tag{6}$$

These special vectors are the *covectors* for $\hat{\boldsymbol{e}}_1$ and $\hat{\boldsymbol{e}}_2$. Some properties are fairly obvious: the covectors of the covectors are the original set of basis vectors, and that for a Cartesian coordinate system, the basis vectors are their own covectors. It is, however, important to remember that basis vectors and covectors both come in sets: the covector $\hat{\boldsymbol{E}}_1$ is not just a function of $\hat{\boldsymbol{e}}_1$, but also of all the other basis vectors. Change the other basis vectors, and $\hat{\boldsymbol{E}}_1$ will also change even if $\hat{\boldsymbol{e}}_1$ remains unchanged. The vectors and their covectors have the property

$$\hat{\boldsymbol{E}}_i \hat{\boldsymbol{e}}_j = \delta_{ij} \tag{7}$$

but, in general,

$$\hat{\boldsymbol{E}}_i \hat{\boldsymbol{E}}_j \neq \delta_{ij} \quad \hat{\boldsymbol{e}}_i \hat{\boldsymbol{e}}_j \neq \delta_{ij}. \tag{8}$$

To make these ideas a little more concrete, let's consider an example in two dimensions, with the basis vectors $\hat{\boldsymbol{e}}_1 = \{1\,0\}^T$ and $\hat{\boldsymbol{e}}_2 = \{\cos(\theta)\ \sin(\theta)\}^T$. The system of equations to be solved for $\hat{a}_1$ and $\hat{a}_2$ is

$$\begin{bmatrix} 1 & \cos(\theta) \\ 0 & \sin(\theta) \end{bmatrix} \begin{Bmatrix} \hat{a}_1 \\ \hat{a}_2 \end{Bmatrix} = \begin{Bmatrix} a_1 \\ a_2 \end{Bmatrix}. \tag{9}$$

Solving the system gives

$$\frac{1}{\sin(\theta)} \begin{bmatrix} \sin(\theta) & -\cos(\theta) \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} a_1 \\ a_2 \end{Bmatrix} = \begin{Bmatrix} \hat{a}_1 \\ \hat{a}_2 \end{Bmatrix} \tag{10}$$

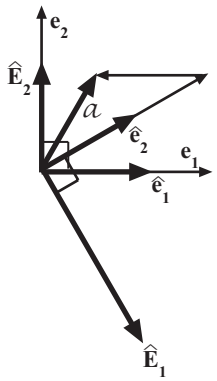

Figure 1: The vectors and covectors for θ equals 30 degrees.

and the covectors are therefore (see Figure 1)

$$\hat{\boldsymbol{E}}_1 = \left\{ \begin{array}{c} 1 \\ -\frac{\cos(\theta)}{\sin(\theta)} \end{array} \right\} \quad \hat{\boldsymbol{E}}_2 = \left\{ \begin{array}{c} 0 \\ \frac{1}{\sin(\theta)} \end{array} \right\}. \tag{11}$$

The reason for introducing vectors and covectors here is $\boldsymbol{B}$ can be thought of as being a collection of covectors corresponding to the nodal displacements associated with the different strain modes of an element. For example, in two dimensions $\epsilon_{11} = \partial u_1/\partial x_1$, and therefore a set of nodal displacements having the form $\{x_1, 0, x_2, 0, \ldots, x_n, 0\}$, when multiplied by the first row of $\boldsymbol{B}$ will give the result $\epsilon_{11} = 1$. If this displacement vector is labelled $\boldsymbol{b}_1$, its covector is $\boldsymbol{B}_1$, the first row in $\boldsymbol{B}$. An arbitrary displacement vector for an element could therefore be expressed as

$$\boldsymbol{d} = (\boldsymbol{B}_1 \cdot \boldsymbol{d})\boldsymbol{b}_1 \ldots = \epsilon_{11}\boldsymbol{b}_1 \ldots \tag{12}$$

Since an element has more degrees of freedom than it has strains, the displacement modes must be expanded to include other modes (e.g., rigid body modes) to form a complete basis for the displacements of the nodes of the element.

A particular type of mode that shows up in under-integrated elements is the zero energy mode. The mathematics of vectors and covectors gives us a language to use for discussing these zero energy modes, and the corresponding extra rows of $\boldsymbol{B}$ required to calculate their amplitude.

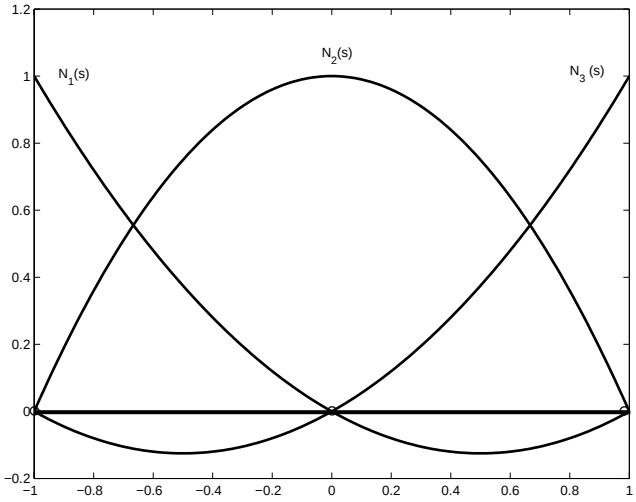

Figure 2: The shape functions for a one-dimensional quadratic element.

2 Zero Energy Modes in One Dimension

All the basic concepts associated with zero energy modes are easily demonstrated in one dimension with a three-node quadratic element using one-point integration. The interpolation functions and their derivatives for the three nodes (see Figure 2) are

$$N_1(s) = \frac{1}{2}s(s-1) \tag{13}$$

$$N_2(s) = 1 - s^2 \tag{14}$$

$$N_3(s) = \frac{1}{2}s(s+1) \tag{15}$$

$$\frac{\partial N_1}{\partial s} = s - \frac{1}{2} \tag{16}$$

$$\frac{\partial N_2}{\partial s} = -2s \tag{17}$$

$$\frac{\partial N_3}{\partial s} = s + \frac{1}{2} \tag{18}$$

The $\boldsymbol{B}$ matrix, for a general location within the element, is therefore

$$\boldsymbol{B}(s) = \frac{1}{\det(J)}[\ s-\tfrac{1}{2} \quad -2s \quad s+\tfrac{1}{2}\ ] \tag{19}$$

$$J = x_1(s-\frac{1}{2}) + x_2(-2s) + x_3(s+\frac{1}{2}) \tag{20}$$

The nodal displacement vector corresponding to a uniform strain of 1 is $\{\ x_1+c \quad x_2+c \quad x_3+c\ \}^T$, where c is any arbitrary constant, and corresponds to a rigid body translation of c. For convenience, we'll call this displacement vector $\boldsymbol{b}$.

A second nodal displacement vector is the rigid body translation

$$\boldsymbol{r} = \{\ 1 \quad 1 \quad 1\ \}^T \tag{21}$$

and its corresponding covector is

$$\boldsymbol{R} = \{\ 1/3 \quad 1/3 \quad 1/3\ \}^T. \tag{22}$$

The product $\boldsymbol{B}\cdot\boldsymbol{r} = 0$, but $\boldsymbol{R}\cdot\boldsymbol{b} = 1/3(x_1+x_2+x_3)+c$. To make $\boldsymbol{R}\cdot\boldsymbol{b} = 0$, we set $c = -1/3(x_1+x_2+x_3)$. We will use the shorthand $x_i^c = x_i - c$ in the subsequent equations.

Since the element has three nodes, the vector space of its nodal displacements has a dimension of three and is spanned by any three linearly independent vectors. This implies that there is a third vector that is linearly independent of $\boldsymbol{b}$ and bfr that can be used to complete the space. Furthermore, there is a covector corresponding to this third vector which completes the covector space for the element.

One choice for the third displacement vector becomes apparent when the element is integrated using one-point integration, where the location and weight are $s = 0$ and $w = 2$, respectively. The strain displacement matrix $\boldsymbol{B}$ simplifies to

$$\boldsymbol{B} = \frac{1}{L}[\ -1 \quad 0 \quad 1\ ]. \tag{23}$$

$$L = x_3 - x_1 \tag{24}$$

The strain in the element is therefore not a function of the displacement at node 2, and if nodes 1 and 3 don't move, the element won't generate any stress regardless of the displacement of node 2. In a similar manner, since the force vector is $\boldsymbol{B}^T\sigma L$, node 2 won't experience any force due to the stress in the element.

By inspection, this peculiar mode, which we'll label $\boldsymbol{h}$, is

$$\boldsymbol{h} = \{\ 0 \quad 1 \quad 0\ \}^T \tag{25}$$

and it is orthogonal to $\boldsymbol{B}(0)$, i.e., $\boldsymbol{B}(0) \cdot \boldsymbol{h} = 0$. The element stiffness matrix for one-point integration is

$$\boldsymbol{K} = EL\boldsymbol{B} \otimes \boldsymbol{B}, \tag{26}$$

and therefore the strain energy associated with this mode is

$$U = \frac{1}{2}\boldsymbol{h}^T \boldsymbol{K} \boldsymbol{h} = 0. \tag{27}$$

The mode $\boldsymbol{h}$ is therefore a *zero energy* mode.

If a calculation is performed using this element, numerical roundoff errors in explicit calculations will possible lead to the motions of the central nodes of the elements being the ones with the largest amplitudes in the calculation. Since these zero energy modes are orthogonal to the stress and strain in the elements, they don't affect the accuracy of the remainder of the solution. In implicit quasi-static calculations, the rows and columns associated with the middle nodes are zero, and the stiffness matrix is singular. In multi-dimensional calculations, the zero energy modes may cause negative Jacobians in the elements, which ends the analysis. A means of reducing the impact of the zero energy modes on the calculation is therefore desirable.

Hourglass modes may be controlled by adding either stiffness or viscosity terms to the equations. In large deformation calculations, a viscous term is usually preferred to avoid building up large elastic forces in the system. Stiffness terms are used in implicit quasi-static calculations to eliminate the singularity of the stiffness matrix. For explicit calculations with small to moderate deformations, and long time periods, a stiffness form of hourglass control is often preferred to prevent hourglass modes from slowly building up in the solution. Combinations of viscous and stiffness hourglass control are also sometimes used, but usually one form of the control is adequate for most calculations.

The simplest formulation to control the zero energy modes, which was used originally in the finite difference community, is to simply add a viscous term to the element force calculations,

$$\boldsymbol{F} = \boldsymbol{B}^T \sigma L + cL\boldsymbol{h} \otimes \boldsymbol{h}\dot{\boldsymbol{u}} \tag{28}$$

where c is a small damping coefficient. In a similar manner, the finite element community added a term to the stiffness matrix,

$$\boldsymbol{K} = EL\boldsymbol{B} \otimes \boldsymbol{B} + eL\boldsymbol{h} \otimes \boldsymbol{h} \tag{29}$$

where e is a small elastic constant.

There is, however, one unfortunate problem with this simple formulation, and that is the zero energy mode $\boldsymbol{h}$ isn't orthogonal to the rigid body mode $\boldsymbol{r}$. The zero energy mode control in Equations 28 and 29 will resist rigid body motion, a highly undesirable, nonphysical response. The appropriate vector to use is the covector $\boldsymbol{H}$. Just as $\boldsymbol{B}$ is orthogonal to $\boldsymbol{r}$ and $\boldsymbol{h}$, and $\boldsymbol{R}$ is orthogonal to $\boldsymbol{b}$ and $\boldsymbol{h}$, $\boldsymbol{H}$ needs to be orthogonal to $\boldsymbol{b}$ and $\boldsymbol{r}$. The required covector is generated using Graham-Schmidt orthogonalization, a procedure for making vectors orthogonal to each other, with $\boldsymbol{h}$ as the initial "guess" for $\boldsymbol{H}$,

$$\begin{aligned} \boldsymbol{H} &= \boldsymbol{h} - (\boldsymbol{r} \cdot \boldsymbol{h})\,\boldsymbol{R} - (\boldsymbol{b} \cdot \boldsymbol{h})\,\boldsymbol{B} && (30)\\ &= \begin{bmatrix} x_2^c/L - 1/3 & 2/3 & -x_2^c/L - 1/3 \end{bmatrix}^T. && (31) \end{aligned}$$

The orthogonality of $\boldsymbol{H}$ with respect to $\boldsymbol{r}$ and bfb is best demonstrated by directly calculating the dot products. First, for the rigid body mode,

$$\begin{aligned} \boldsymbol{H} \cdot \boldsymbol{r} &= x_2^c/L - 1/3 + 2/3 - x_2^c/L - 1/3 && (32)\\ &= 0. && (33) \end{aligned}$$

The product with the strain mode is slightly more complicated,

$$\begin{aligned} \boldsymbol{H} \cdot \boldsymbol{b} &= \boldsymbol{H} \cdot \begin{Bmatrix} x_1^c & x_2^c & x_3^c \end{Bmatrix}^T && (34)\\ &= (x_1^c x_2^c)/L - x_1^c/3 + 2x_2^c/3 - (x_3^c x_2^c)/L - x_3^c/L && (35)\\ &= (x_1^c - x_3^c)x_2^c/L - (x_1^c - 2x_2^c + x_3^c)/3 && (36)\\ &= -x_2^c - (x_1^c - 2x_2^c + x_3^c)/3 && (37)\\ &= -(x_1^c + x_2^c + x_3^c)/3 = 0. && (38)\\ & && (39) \end{aligned}$$

Substituting $\boldsymbol{H}$ for $\boldsymbol{h}$ in Equations 28 and 29 gives two formulations for controlling the hourglass mode without opposing rigid body motion or altering the constant strain response,

$$\begin{aligned} \boldsymbol{F} &= -\boldsymbol{B}^T \sigma L - cL\boldsymbol{H} \otimes \boldsymbol{H}\dot{\boldsymbol{u}} && (40)\\ \boldsymbol{K} &= EL\boldsymbol{B} \otimes \boldsymbol{B} + eL\boldsymbol{H} \otimes \boldsymbol{H} && (41) \end{aligned}$$

Summarizing our results so far, the quadratic element has three basic displacement modes:

1. The rigid body translation mode $\boldsymbol{r} = \{\ 1 \quad 1 \quad 1\ \}^T$.
2. The strain mode $\boldsymbol{b} = \{\ -L/2 \quad 0 \quad L/2\ \}^T$. This mode has been normalized so that $\boldsymbol{B} \cdot \boldsymbol{b} = 1$.
3. The zero energy mode $\boldsymbol{h} = \{\ -1/2 \quad 1 \quad -1/2\ \}^T$, which has been normalized so that $\boldsymbol{H} \cdot \boldsymbol{h} = 1$ and it has been orthogonalized with respect to $\boldsymbol{r}$ and $\boldsymbol{b}$.

Since the element only has three degrees of freedom, and the three modes are linearly dependent, these modes span the entire space of possible deformations for the element.

Corresponding to the three displacement modes are the three covectors that detect the amplitudes of the displacement modes in an arbitrary displacement vector:

1. $\boldsymbol{R} = [\ 1/3 \quad 1/3 \quad 1/3\ ]$ is the rigid body covector.
2. $\boldsymbol{B} = [\ -1/L \quad 0 \quad 1/L\ ]$ is the familiar strain-displacement matrix.
3. $\boldsymbol{H} = [\ -1/3 \quad 2/3 \quad -1/3\ ]$.

Just as the three displacement modes span the element's vector space, the covectors span its covector space. For example, the general form for the strain-displacement matrix, $\boldsymbol{B}(s)$ can be expressed in terms of the covectors as

$$\boldsymbol{B}(s) = \frac{L}{2\det(J)}\boldsymbol{B} - \frac{3s}{\det(J)}\boldsymbol{H}. \tag{42}$$

The element stiffness matrix, for uniformly spaced nodes, with exact integration is

$$\begin{aligned} \boldsymbol{K} &= \int_{-1}^{+1} E\boldsymbol{B}^T(s)\boldsymbol{B}(s)\det(J)\mathrm{d}s & (43)\\ &= \frac{E}{L}\begin{bmatrix} 7/3 & -8/3 & 1/3 \\ -8/3 & 16/3 & -8/3 \\ 1/3 & -8/3 & 7/3 \end{bmatrix} & (44) \end{aligned}$$

Substituting Equation 42 into Equation 43 results in three terms,

$$\boldsymbol{K} = \int_{-1}^{+1} E\frac{L^2}{4\det^2(J)}\boldsymbol{B}^T\boldsymbol{B}\,\det(J)\mathrm{d}s \tag{45}$$

$$- \int_{-1}^{+1} E\frac{3Ls}{\det^2(J)}(\boldsymbol{B}^T\boldsymbol{H} + \boldsymbol{H}^T\boldsymbol{B})\,\det(J)\mathrm{d}s \tag{46}$$

$$+ \int_{-1}^{+1} E\frac{9s^2}{\det^2(J)}\boldsymbol{H}^T\boldsymbol{H}\,\det(J)\mathrm{d}s. \tag{47}$$

The first integral is the stiffness matrix that is obtained using 1-point integration. The second term, which is zero because $\int s\mathrm{d}s = 0$, doesn't contribute to either the exact stiffness or numerically integrated stiffness matrices. The third term is the difference between the exactly integrated stiffness matrix and the one-point integration. Choosing

$$eL = \int_{-1}^{+1} E\frac{9s^2}{\det(J)}\mathrm{d}s = \frac{12E}{L} \tag{48}$$

for Equation 41 gives the exact stiffness matrix (for equally spaced nodes) with one-point integration. This approach is also applicable to elements in two and three dimensions.

3 Implementation in an Explicit Code

The implementation of the viscous and stiffness forms of the zero energy mode control in an explicit finite element code are very similar. Calculating the matrix $\boldsymbol{H} \otimes \boldsymbol{H}$ requires n^2 floating point multiplies for an element with n degrees of freedom. The cost of a matrix-vector multiply $\boldsymbol{H} \otimes \boldsymbol{H}\boldsymbol{u}$ is also high, requiring $2n^2 - n$ floating point operations. The implementation of the zero energy mode control is therefore structured to avoid matrix operations.

For the current discussion, we'll assume that both viscous and stiffness control is used and that it has the form

$$\boldsymbol{F} = -e\boldsymbol{H} \otimes \boldsymbol{H}\boldsymbol{u} - c\boldsymbol{H} \otimes \boldsymbol{H}\dot{\boldsymbol{u}}. \tag{49}$$

The original implementations assumed that e and c are constants, but more recent formulations have made them time dependent functions of the material model, which makes a rate form for the stiffness contribution necessary. The

rate form is also attractive even if e is constant because many explicit codes don't routinely store the total displacement vector, $\boldsymbol{u}$. A generalized force, f_h, is stored for each element,

$$f_h(t) = \int_0^t e\boldsymbol{H} \cdot \boldsymbol{u} \mathrm{d}t. \tag{50}$$

The resulting implementation evaluates, in sequence,

$$\begin{aligned} \dot{\alpha} &= \boldsymbol{H} \cdot \dot{\boldsymbol{u}} && (51) \\ f_h^{n+1} &= f_h^n + \triangle t e \dot{\alpha} && (52) \\ \boldsymbol{F} &= (-f_h^{n+1} - c\dot{\alpha})\boldsymbol{H}. && (53) \end{aligned}$$

The dot product for calculating $\dot{\alpha}$ requires $2n-1$ floating point operations, the update of f_h^{n+1} requires 3, and the final evaluation of the force requires $n+2$, for a total of $3n+4$ operations. This implementation is therefore approximately n times faster than evaluating $\boldsymbol{H} \otimes \boldsymbol{H}$ and performing the matrix multiply. For an element with 3 degrees of freedom, this cost ratio isn't terribly important, but for an 8-node brick element, which has 24 degrees of freedom, the difference in cost is substantial.