THE USE OF ENRICHED HEXAHEDRAL ELEMENTS WITH BUBBLE FUNCTIONS FOR FINITE ELEMENT ANALYSIS

Shi-Pin Ho and Yen-Liang Yeh
Department of Mechanical Engineering, National Cheng Kung University, Tainan, Taiwan, R.O.C.
Contact: spho@mail.ncku.edu.tw

Received January 2006, Accepted December 2006
No. 06-CSME-01, E.I.C. Accession 2920

Abstract

In this paper, the concept that adds the interior nodes of the Lagrange elements to the serendipity elements is described and a family of enriched elements is presented to improve the accuracy of finite element analysis. By the use of the static condensation technique at the element level, the extra computation time in using these elements can be ignored. Three-dimensional elastic problems are used as examples in this paper. The numerical results show that these enriched elements are more accurate than the traditional serendipity elements. The convergence rate of the enriched elements is the same as the traditional serendipity elements. In the numerical example, the error norm of the first order enriched elements can be reduced when compared with the use of the traditional serendipity element, but the computation time is increased a little. The use of enriched second and third order hexahedral elements does not only improve accuracy, but also saves the computation time for solving the system of equations, when the precondition conjugate gradient method is used to solve the system of equations. The saving of computation time is due to the decrease in the number of iteration for the iteration method.

L'UTILISATION DES ELEMENTS HEXAEDRES ENRICHIS AVEC LA FONCTION DE BULLE POUR L'ANALYSE D'ELEMENT FINI

Résumé

Dans cet article, nous rajoutons le concept des noeuds intérieurs des éléments Lagrange aux éléments serendipity; une famille des éléments enrichis y est présentée également pour améliorer l'exactitude de l'analyse d'élément fini. Par l'utilisation de la technique de la condensation statique sur des éléments, le temps de calcul est légèrement augmenté, mais peut être ignoré. Des problèmes élastiques tridimensionnels se servent ici comme des exemples. Les résultats numériques prouvent que ces éléments enrichis sont plus précis que ceux serendipity traditionnels. Le taux de convergence des éléments enrichis est identique que ceux serendipity traditionnels. Dans l'exemple numérique, l'erreur de la norme des éléments enrichis du premier ordre peut être réduite par rapport à l'utilisation de l'élément serendipity traditionnel, mais le temps de calcul est augmenté. L'utilisation des éléments hexaèdres enrichis du second et troisième ordre améliore non seulement l'exactitude, mais gagne le temps de calcul pour résoudre le système des equations quand la méthode du gradient conjugé est employée de condition préalable. L'économie du temps de calcul est due à la diminution du nombre d'itération pour la méthode d'itération.

INTRODUCTION

The selection and use of elements is very important in finite element analysis, affecting both accuracy and computation time. Finite element methodology has been employing Lagrange elements and Serendipity elements for three decades. However, the research for improvements in computational speed and accuracy has continued. In the 1970's, Herrmann [1] presented a two-dimensional incompatible element and evaluated its efficiency. In the same decade Wilson et al [2] presented the incompatible models method, aiming to improve the behavior of elements in bending situations. Wilson's method was later improved by Taylor [3]. Many people adopted Taylor's method, and modifications soon followed. Chen [4] applied Taylor's method to solve the axisymmetric solid problem. Taiebat [5] extended the idea to produce three dimensional non-conform elements and de Sousa et al [6] recommended low order elements for three-dimensional analysis.

The use and improvement of higher order elements is another way to improve the accuracy of finite element analysis. Rathod et al [7, 8] derived the interpolation polynomials for the general serendipity elements that allow arbitrarily placed nodes along edges. Celia [9] showed that side nodes positioned at the same relative distance from corner nodes in both local and global space and interior nodes positioned so that they don't affect the Jacobian of the coordinate transformation provide improved accuracy over nodes positioned locally without regard for the global configuration. Universal serendipity elements [10, 11] defined as isoparametric elements at element edges in an arbitrary manner is another application. Kikuchi [12] developed a general principle to derive 8-node elements by eliminating the interior node of a 9-node element with all the Cartesian quadratic polynomials preserved.

Adding bubble functions can also greatly improve element performance. A bubble function is defined as a function that disappears along the element boundaries. Accordingly, the inter-element compatibility requirements for displacement are maintained for the element, and the bubble function parameters can be eliminated at the element level by static condensation and can be recovered for subsequent calculations [13]. Bubble functions have been introduced to construct plate element models [14, 15, 16, 17]. For example, Pinsky and Jasti [14] developed four-node and nine-node plate bending elements that incorporated bubble function displacements. Kemp et al [16] used the bubble function displacements to develop a four-node solid shell element based on the assumed strain formulation and verified that the bubble function displacement alleviates sensitivity to mesh distortion as well as element locking. On the other hand, bubble functions are employed to solve advection-diffusion problems by Brezzi, Franca and Farhat [18, 19, 20, 21] and are shown to help in stabilizing the advective operator without recourse to upwinding or any other numerical strategy. Furthermore, the limitation of bubble functions is discussed by Franca [22] and error analysis of residual-free bubbles is discussed by Brezzi [23] and Sanglli [24].However, the introduction of the higher order bubble functions may result in element models that require a higher-order numerical integration rule for the element stiffness matrix. Therefore accuracy and computation time must be considered when we evaluate the performance of the elements.

The goal of this paper is to derive a family of enriched hexahedral elements that improve the accuracy of serendipity elements and do not increase computation time. To do so, the interior nodes of Lagrange elements are added to serendipity elements and then the polynomial order of the shape functions is increased to improve accuracy. In the finite element calculation process, static condensation is used to reduce the degree of freedom at interior nodes. Since static condensation is performed at the element level, only a small increase in computation time is needed.

The shape functions of the enriched hexahedral elements are derived in Section 2. Subparametric formulation and static condensation of the enriched elements are discussed in Section 3. In Section 4, three-dimensional elastic problems are used to demonstrate the performance of the proposed elements. Conclusions are provided in Section 5.

FORMULATION OF THE ENRICHED HEXAHEDRAL ELEMENTS

In this paper, a nodal based basis is constructed by adding the interior nodes of Lagrange basis to a serendipity basis. The serendipity basis is then corrected so that the Kronecker delta property is

satisfied at the interior nodes, and a family of enriched elements is presented.

We use the notation $N_i^{(L)}$ to indicate the shape function that corresponds to the i-th node of the Lagrange element. After the numbers of all edge nodes are assigned, the interior nodes are assigned in turn. On the other hand, serendipity elements have no interior nodes. In this methodology the notation $N_i^{(S)}$ is used to indicate the shape function which corresponds to the i-th node of the serendipity element.

For the general case of an n-th order enriched element, we combine the n-th order serendipity elements with the interior nodes of the m-th order Lagrange element. This method can provide higher order shape functions for the interior of lower order serendipity elements.

We need to keep the relation $N_i(\xi_j,\eta_j,\varsigma_j)=\delta_{ij}$ for every shape function within the element, so the serendipity shape functions corresponding to the edge nodes must be modified. The enriched element, designated nSmL, is derived by combining the n-th order serendipity element and the interior nodes of m-th order Lagrange element (m≥2). The n-th order serendipity elements have 12n-4 nodes and the shape functions are represented as $N_i^{(S)}, i=1\sim 12n-4$. The m-th order Lagrange elements have $(m-1)^3$ interior nodes. Since the $6m^2+2$ nodes on the surfaces are numbered first, the $(m-1)^3$ interior nodes of the m-th order Lagrange element are numbered from $6m^2+3$ to $(m+1)^3$. The shape functions of the Lagrange element interior nodes $N_i^{(L)}$, $i=6m^2+3\sim(m+1)^3$, are used as the interior shape functions of the nSmL element. Since the edges of the nSmL element have 12n-4 nodes, these shape functions are numbered from 12n-3 to $12n-4+(m-1)^3$. Therefore, the shape functions of the enriched element can be represented as

$$N_i(\xi,\eta,\varsigma)=\begin{cases} N^{(L)}_{(6m^2-12n+6+i)} & for \quad i=12n-3\sim 12n-4+(m-1)^3 \\ N_i^{(S)}-\sum_{j=12n-4}^{12n-4+(m-1)^3} a_{ij}N_j & for \quad i=1\sim 12n-4 \end{cases} \tag{1}$$

where $a_{ij}=N_j^{(S)}(\xi_i,\eta_i,\varsigma_i)$. Because $N_j^{(L)}, j=6m^2+3\sim(m+1)^3$, is zero at all edge nodes, the modification of $a_{ij}N_j$ makes N_i satisfy the $N_i(\xi_j,\eta_j,\varsigma_j)=\delta_{ij}$ condition.

For example, the shape functions of the 1S2L element can be derived using above definition and these shape function are shown as follows

$$N_1(\xi,\eta,\varsigma)=\frac{1}{8}(1-\xi)(1-\eta)(1-\varsigma)-\frac{1}{8}(1-\xi^2)(1-\eta^2)(1-\varsigma^2) \tag{2}$$

$$N_2(\xi,\eta,\varsigma)=\frac{1}{8}(1+\xi)(1-\eta)(1-\varsigma)-\frac{1}{8}(1-\xi^2)(1-\eta^2)(1-\varsigma^2) \tag{3}$$

$$N_3(\xi,\eta,\varsigma)=\frac{1}{8}(1+\xi)(1+\eta)(1-\varsigma)-\frac{1}{8}(1-\xi^2)(1-\eta^2)(1-\varsigma^2) \tag{4}$$

$$N_4(\xi,\eta,\varsigma)=\frac{1}{8}(1-\xi)(1+\eta)(1-\varsigma)-\frac{1}{8}(1-\xi^2)(1-\eta^2)(1-\varsigma^2) \tag{5}$$

$$N_5(\xi,\eta,\varsigma)=\frac{1}{8}(1-\xi)(1-\eta)(1+\varsigma)-\frac{1}{8}(1-\xi^2)(1-\eta^2)(1-\varsigma^2) \tag{6}$$

$$N_6(\xi,\eta,\varsigma)=\frac{1}{8}(1+\xi)(1-\eta)(1+\varsigma)-\frac{1}{8}(1-\xi^2)(1-\eta^2)(1-\varsigma^2) \tag{7}$$

$$N_7(\xi,\eta,\varsigma)=\frac{1}{8}(1+\xi)(1+\eta)(1+\varsigma)-\frac{1}{8}(1-\xi^2)(1-\eta^2)(1-\varsigma^2) \tag{8}$$

$$N_8(\xi,\eta,\varsigma)=\frac{1}{8}(1-\xi)(1+\eta)(1+\varsigma)-\frac{1}{8}(1-\xi^2)(1-\eta^2)(1-\varsigma^2) \tag{9}$$

$$N_9(\xi,\eta,\varsigma)=(1-\xi^2)(1-\eta^2)(1-\varsigma^2) \tag{10}$$

It will be seen that the use of $N_9(\xi,\eta,\varsigma)$ improves the accuracy of the element. Because the shape function of the 1S2L element interior node vanishes on all edge sides, the 1S2L element is conforming.

SUBPARAMETRIC FORMULATION AND STATIC CONDENSATION

Since the two methods, which are subparametric formulation and static condensation, regulate the implementation of the nSmL element, the nSmL element is easily added into an existing finite element code by programmers. That employs the shape functions of the n-th serendipity element to approximate the geometry so the particular mesh generation is unnecessary, and this condenses the interior degree of freedom so the element coefficient matrix can be reduced at the element level. Therefore, we will describe the two methods in this section.

Subparametric formulation

The dependent variable u is approximated in the nSmL element by the expression

$$u = \sum_{i=1}^{12n-4+(m-1)^3} u_i N_i(\xi,\eta,\varsigma) \tag{11}$$

Because the shape functions of the nSmL element interior nodes vanish at the element's edges, the shape functions at the element's edges are the same as the character of the n-th order serendipity element. The shape functions of the n-th serendipity element can be employed to approximate the geometry. Therefore, we can use the following formulation to approximate the geometry of the three dimensional nSmL elements as

$$x = \sum_{i=1}^{12n-4} x_i N_i^{(S)}(\xi,\eta,\varsigma) \tag{12}$$

$$y = \sum_{i=1}^{12n-4} y_i N_i^{(S)}(\xi,\eta,\varsigma) \tag{13}$$

$$z = \sum_{i=1}^{12n-4} z_i N_i^{(S)}(\xi,\eta,\varsigma) \tag{14}$$

Where the (x_i, y_i, z_i) is the location of the i-th node in the edge.

This kind of formulation is called a subparametric formulation in which the geometry is represented by lower order elements than those used to approximate the dependent variables. By the subparametric formulation, the coordinates of the interior nodes of the enriched element are not needed, so it is easy to implement nSmL elements in existing finite element code which use serendipity elements. Therefore, the particular mesh generation is unnecessary.

If the interior nodes are necessary and the isoparametric formulation must be employed in some situations, for example a large deformation problem, the locations of the interior nodes can be calculated by the shape functions of the n-the order serendipity shown in the following equations after the edge nodes have been generated.

$$x_j = \sum_{i=1}^{12n-4} x_i N_i^{(S)}(\xi_j,\eta_j,\varsigma_j) \qquad j = 12n-3 \sim 12n-4+(m-1)^3 \tag{15}$$

$$y_j = \sum_{i=1}^{12n-4} y_i N_i^{(S)}(\xi_j,\eta_j,\varsigma_j) \qquad j = 12n-3 \sim 12n-4+(m-1)^3 \tag{16}$$

$$z_j = \sum_{i=1}^{12n-4} z_i N_i^{(S)}(\xi_j,\eta_j,\varsigma_j) \qquad j = 12n-3 \sim 12n-4+(m-1)^3 \tag{17}$$

Where the (x_j, y_j, z_j) is the location of the j-the interior node and the (x_i, y_i, z_i) is the location of the i-th node at the edge. And the $(\xi_j, \eta_j, \varsigma_j)$ is the location of $(6m^2 - 12n + 6 + j)$-th node of the m-th order Lagrange element in the master coordinate. So the interior nodes of the nSmL element can easily be inserted to approximate the geometry.

Static condensation

The degrees of freedom of an nSmL element can be classified into two groups. Those that are not connected to the freedoms of another element are named "interior freedoms" and those that are connected to at least one other element are named "edge freedoms." For example, when the 1S2L element is used in the finite element model, the element coefficient matrix $\mathbf{K}^e$ and the element loading vector $\mathbf{F}^e$ of the element e are

$$\begin{bmatrix} K_{11}^e & K_{12}^e & K_{13}^e & K_{14}^e & K_{15}^e & K_{16}^e & K_{17}^e & K_{18}^e & K_{19}^e \\ K_{21}^e & K_{22}^e & K_{23}^e & K_{24}^e & K_{25}^e & K_{26}^e & K_{27}^e & K_{28}^e & K_{29}^e \\ K_{31}^e & K_{32}^e & K_{33}^e & K_{34}^e & K_{35}^e & K_{36}^e & K_{37}^e & K_{38}^e & K_{39}^e \\ K_{41}^e & K_{42}^e & K_{43}^e & K_{44}^e & K_{45}^e & K_{46}^e & K_{47}^e & K_{48}^e & K_{49}^e \\ K_{51}^e & K_{52}^e & K_{53}^e & K_{54}^e & K_{55}^e & K_{56}^e & K_{57}^e & K_{58}^e & K_{59}^e \\ K_{61}^e & K_{62}^e & K_{63}^e & K_{64}^e & K_{65}^e & K_{66}^e & K_{67}^e & K_{68}^e & K_{69}^e \\ K_{71}^e & K_{72}^e & K_{73}^e & K_{74}^e & K_{75}^e & K_{76}^e & K_{77}^e & K_{78}^e & K_{79}^e \\ K_{81}^e & K_{82}^e & K_{83}^e & K_{84}^e & K_{85}^e & K_{86}^e & K_{87}^e & K_{88}^e & K_{89}^e \\ K_{91}^e & K_{92}^e & K_{93}^e & K_{94}^e & K_{95}^e & K_{96}^e & K_{97}^e & K_{98}^e & K_{99}^e \end{bmatrix} \begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \\ u_4^e \\ u_5^e \\ u_6^e \\ u_7^e \\ u_8^e \\ u_9^e \end{Bmatrix} = \begin{Bmatrix} F_1^e \\ F_2^e \\ F_3^e \\ F_4^e \\ F_5^e \\ F_6^e \\ F_7^e \\ F_8^e \\ F_9^e \end{Bmatrix} \tag{18}$$

Because the interior freedoms of the element e are not coupled with the other elements, the entries of the row and column that correspond to the interior freedoms of element e are not used in the algorithm for the following assembly procedure. If we use the Gaussian elimination method to solve the system of equations, we choose K_{99}^e as the pivot. The modified global coefficient matrix and the global loading vector are

$$\begin{bmatrix} \overline{K_{11}^e} & \overline{K_{12}^e} & \overline{K_{13}^e} & \overline{K_{14}^e} & \overline{K_{15}^e} & \overline{K_{16}^e} & \overline{K_{17}^e} & \overline{K_{18}^e} & 0 \\ \overline{K_{21}^e} & \overline{K_{22}^e} & \overline{K_{23}^e} & \overline{K_{24}^e} & \overline{K_{25}^e} & \overline{K_{26}^e} & \overline{K_{27}^e} & \overline{K_{28}^e} & 0 \\ \overline{K_{31}^e} & \overline{K_{32}^e} & \overline{K_{33}^e} & \overline{K_{34}^e} & \overline{K_{35}^e} & \overline{K_{36}^e} & \overline{K_{37}^e} & \overline{K_{38}^e} & 0 \\ \overline{K_{41}^e} & \overline{K_{42}^e} & \overline{K_{43}^e} & \overline{K_{44}^e} & \overline{K_{45}^e} & \overline{K_{46}^e} & \overline{K_{47}^e} & \overline{K_{48}^e} & 0 \\ \overline{K_{51}^e} & \overline{K_{52}^e} & \overline{K_{53}^e} & \overline{K_{54}^e} & \overline{K_{55}^e} & \overline{K_{56}^e} & \overline{K_{57}^e} & \overline{K_{58}^e} & 0 \\ \overline{K_{61}^e} & \overline{K_{62}^e} & \overline{K_{63}^e} & \overline{K_{64}^e} & \overline{K_{65}^e} & \overline{K_{66}^e} & \overline{K_{67}^e} & \overline{K_{68}^e} & 0 \\ \overline{K_{71}^e} & \overline{K_{72}^e} & \overline{K_{73}^e} & \overline{K_{74}^e} & \overline{K_{75}^e} & \overline{K_{76}^e} & \overline{K_{77}^e} & \overline{K_{78}^e} & 0 \\ \overline{K_{81}^e} & \overline{K_{82}^e} & \overline{K_{83}^e} & \overline{K_{84}^e} & \overline{K_{85}^e} & \overline{K_{86}^e} & \overline{K_{87}^e} & \overline{K_{88}^e} & 0 \\ K_{91}^e & K_{92}^e & K_{93}^e & K_{94}^e & K_{95}^e & K_{96}^e & K_{97}^e & K_{98}^e & K_{99}^e \end{bmatrix} \begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \\ u_4^e \\ u_5^e \\ u_6^e \\ u_7^e \\ u_8^e \\ u_9^e \end{Bmatrix} = \begin{Bmatrix} \overline{F_1^e} \\ \overline{F_2^e} \\ \overline{F_3^e} \\ \overline{F_4^e} \\ \overline{F_5^e} \\ \overline{F_6^e} \\ \overline{F_7^e} \\ \overline{F_8^e} \\ F_9^e \end{Bmatrix} \tag{19}$$

where $\overline{K_{ij}^e} = K_{ij}^e - \frac{K_{ii}^e}{K_{99}^e} K_{9j}^e$ and $\overline{F_i^e} = F_i^e - \frac{K_{ii}^e}{K_{99}^e} F_9^e$.

Only the entries, which are connected with the other elements, are modified, while the other entries remain unchanged. Therefore we use the interior freedoms to do the Gaussian elimination procedure called static condensation in the element level. After the edge freedoms are assembled and solved, we get the interior freedom by

$$u_9^e = (F_9^e - \sum_{i=1}^{8} K_{9i}^e u_i^e) / K_{99}^e \tag{20}$$

which is also done at the element level.

In general, all the interior freedoms can be made invisible to the user by applying a small dose of static condensation. The element coefficient matrix can be reduced at the element level and divided into several partitions

$$\begin{bmatrix} K_{ee} & K_{ei} \\ K_{ie} & K_{ii} \end{bmatrix} \begin{Bmatrix} u_e \\ u_i \end{Bmatrix} = \begin{Bmatrix} F_e \\ F_i \end{Bmatrix} \tag{21}$$

where u_e is the undetermined variables vector of the edge nodes and u_i is the undetermined variables vector of the interior nodes. K_{ee}, K_{ei}, K_{ie} and K_{ii} are submatrices. Because the interior nodes have no relation with other elements, we can let

$$\begin{bmatrix} \overline{K_{ee}} & 0 \\ K_{ie} & K_{ii} \end{bmatrix} \begin{Bmatrix} u_e \\ u_i \end{Bmatrix} = \begin{Bmatrix} \overline{F_e} \\ F_i \end{Bmatrix} \tag{22}$$

where $\overline{K_{ee}} = K_{ee} - K_{ei}K_{ii}^{-1}K_{ie}$ and $\overline{F_e} = F_e - K_{ei}K_{ii}^{-1}F_i$ are assembled to the global coefficient matrix and force. After u_e is solved, we get u_i as

$$u_i = K_{ii}^{-1}F_i - K_{ii}^{-1}K_{ie}u_e \tag{23}$$

Therefore we can see that Gaussian elimination is used to produce the $\overline{K_{ee}}$ and $\overline{F_e}$, then store the parts $K_{ii}^{-1}F_i$ and $K_{ii}^{-1}K_{ie}$ which are needed to solve u_i.

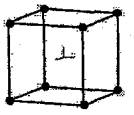
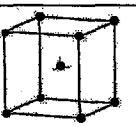
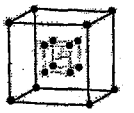
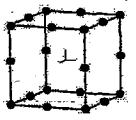
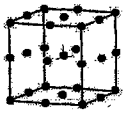
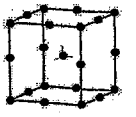
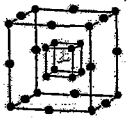
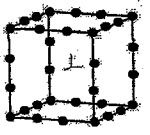
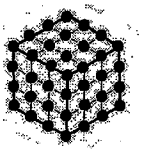
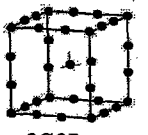
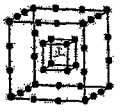
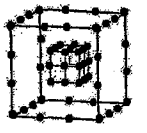
In the coefficient matrix $\overline{K_{ee}} = K_{ee} - K_{ei}K_{ii}^{-1}K_{ie}$ it can be seen that K_{ee} applies partial factorization and this procedure can be regarded as a preconditioner [25]. Notably, this preconditioner is applied directly to the entities of the coefficient matrix, so the procedure for solving the system of equations can be applied to another preconditioner if the precondition conjugate gradient method is used. Therefore, in the following numerical experiment, the enriched elements, static condensation and precondition conjugate gradient method cooperate in the finite element analysis procedure.

NUMERICAL EXAMPLES

Elastic problems are considered in this section to verify the efficiency and accuracy of the enriched elements. The C++ program to solve these examples is compiled with Microsoft Visual C++ 6.0 and the operating system is Windows XP with SP2. All the computational work is carried out on a PC running Pentium4 1.6GHz with 512 MB RAM and IDE-ATA100 hard disk. The isotropic material is used in the numerical examples. First we test which formulae of Gauss-Legendre quadrature for the enriched elements are applicable in the analytic procedure. Then we evaluate the error norm and the computation time for those proposed elements. The precondition conjugate gradient method (PCG) is used to solve the system of equations and the converged tolerance of PCG iteration is 10^{-8}. After comparing different preconditioners, the SSOR preconditioner [26] is used in the following numerical examples.

We will compare the performance of the serendipity elements (herein designated 1S, 2S and 3S), Lagrange elements (herein designated 2L and 3L) and the enriched elements (herein designated 1S2L, 1S3L, 2S2L, 2S3L, 3S2L, 3S3L and 3S4L) used in finite element analysis. The diagrams of the proposed elements are shown in Table 1.

Table 1. The elements are used in the numerical experiment

	Serendipity	Lagrange	Enriched		
First order	 1S		 1S2L	 1S3L	
Second order	 2S	 2L	 2S2L	 2S3L	
Third order	 3S	 3L	 3S2L	 3S3L	 3S4L

MacNeal and Harder's test problems

A curved cantilever beam loaded with an in-plane shear load, which is described by MacNeal and Harder [27], is considered (Figure 1) and is solved using the proposed elements. In this section, we also check which formulae of Gauss-Legendre quadrature should be used when using the proposed elements. Different Gauss-Legendre quadrature formulae (herein including 1-point, 8-point, 27-point and 64-point formulae) are adopted in the following numerical testing. Comparison is also made with the Solid45 element and Solid95 element of ANSYS 9.0. The Solid45 element is an eight-node brick element and the excluded extra displacement shapes formula as well as the included extra displacement shapes formula are considered in the numerical experiments. The Solid95 element is a twenty-node brick element but uses a peculiar 14-point integration rule.

The elements are tested for its principal deformation modes under curved geometry. Normalized tip deflections with respect to theoretical values are shown in Table 2. The iteration count for precondition conjugate gradient method is listed in the parenthesis and the symbol X means that the Gauss-Legendre quadrature formula cannot be used for the targeted element because a serious error occurred. The performances of the proposed elements are comparable with those of the other elements. All the third order elements predict a very accurate solution. The 2S3L shows a more accurate solution than the other second order element in the curved beam test when using the 27-point Gauss-Legendre quadrature formula. In the cases of the first order elements, we find that the 1S element shows its inability to represent the bending state and the 1S2L and 1S3L elements also present the same deficiency. Although the Solid45 element included extra displacement shapes is able to represent the bending state in the regular and parallelogram shape mesh, it is incapable in the trapezoidal shape mesh [27].

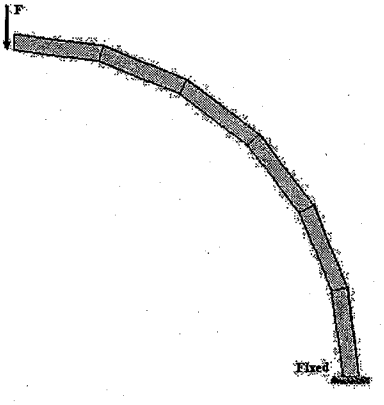


Figure 1. Curved beam:
Inner radius = 4.12;
outer radius = 4.32;
arc = 90°;
thickness = 0.1;
$E = 1.0 \times 10^7$;
$\nu = 0.25$;
mesh = 6×1×1.
A unit-shear force F is applied at the tip.

However, when both accuracy and efficiency is considered, we should select the lowest and most usable formula in order to save computation time. The 1S and 1S2L elements should adopt the 8-point Gauss-Legendre quadrature formula, but the 1S3L element should adopt the 27-point Gauss-Legendre quadrature formula. All the second order elements should adopt the 27-point Gauss-Legendre quadrature formula. In the third order elements, 3L and 3S4L elements should adopt the 64-point Gauss-Legendre quadrature formula and the others should adopt the 27-point Gauss-Legendre quadrature formula.

Table 2. Normalized tip deflection of curved beam

	1-point	8-point	27-point	64-point
1S	X	0.07324 (27)	0.07324 (27)	0.07324 (27)
1S2L	X	0.06389 (26)	0.06886 (26)	0.06683 (26)
1S3L	X	X	0.07373 (27)	0.07374 (27)
2S	X	X	0.87455 (99)	0.87468 (98)
2L	X	X	0.87782 (113)	0.87796 (112)
2S2L	X	X	0.87486 (100)	0.87499 (100)
2S3L	X	X	0.88207 (98)	0.87967 (97)
3S	X	X	1.00649 (185)	1.00445 (137)
3L	X	X	X	1.00970 (163)
3S2L	X	X	1.00687 (192)	1.00481 (158)
3S3L	X	X	1.00725 (196)	1.00500 (135)
3S4L	X	X	X	1.00549 (143)
Solid 45 (excluding extra displacement shapes)				0.07324
Solid 45 (including extra displacement shapes)				0.87970
Solid 95 (14-point integration rule)				0.87472

* The above results are the computational tip deflections to be divided by the theoretical solution 0.08734.

An elastic problem that is devised an exact solution

A small deformation field is assumed for the region bounded by $0 \le x \le 1$, $0 \le y \le 1$ and $0 \le z \le 1$. The geometry is a unit block that is fixed to the entire boundary and a body force that is derived from a designed analytic solution is applied to the block. The exact solutions of the displacement field are sine functions of the Cartesian coordinates (x, y, z), shown as

$$u = v = w = 0.001 \times \sin(\pi x) \times \sin(\pi y) \times \sin(\pi z) \tag{24}$$

u, v and w are displacement functions in the x, y and z directions. These displacement fields satisfy the condition that the Jacobian is positive in the field.

When the deformation functions are substituted into the equations of stress-strain and strain-deformation, the stress components are

$$\sigma_x = 0.001*\pi*[C_{12}*sin(\pi x)*\cos(\pi y+\pi z)+C_{11}*\cos(\pi x)*\sin(\pi y)*\sin(\pi z)] \tag{25}$$

$$\sigma_y = 0.001*\pi*[C_{12}*\cos(\pi x+\pi z)*\sin(\pi y)+C_{11}*sin(\pi x)*\cos(\pi y)*\sin(\pi z)] \tag{26}$$

$$\sigma_z = 0.001*\pi*[C_{12}*\cos(\pi x+\pi y)*\sin(\pi z)+C_{11}*sin(\pi x)*\sin(\pi y)*\cos(\pi z)] \tag{27}$$

$$\tau_{yz} = \pi*C_{44}*\sin(\pi x)*\sin[\pi(y+z)] \tag{28}$$

$$\tau_{xz} = \pi*C_{44}*\sin(\pi y)*\sin[\pi(x+z)] \tag{29}$$

$$\tau_{xy} = \pi*C_{44}*\sin(\pi z)*\sin[\pi(x+y)] \tag{30}$$

The coefficient of stress-strain relation are $C_{11} = E(1-\nu)/(1+\nu)(1-2\nu)$, $C_{44} = E/2(1+\nu)$ and

$C_{12} = E\nu/(1+\nu)(1-2\nu)$, where E is the elastic modulus and ν is the Poisson ratio.

When the stress components are substituted into the equilibrium equations, we get the body force functions as

$$f_x = 0.001*\pi^2*\{C_{11}*\sin(\pi x)*\sin(\pi y)*\sin(\pi z) - C_{12}*\cos(\pi x)*\sin[\pi(x+y)] - C_{44}\sin(\pi z)*\cos[\pi(x+y)] - C_{44}\sin(\pi y)*\cos[\pi(x+z)]\} \quad (31)$$

$$f_y = 0.001*\pi^2*\{C_{11}*\sin(\pi x)*\sin(\pi y)*\sin(\pi z) - C_{12}*\cos(\pi y)*\sin[\pi(x+z)] - C_{44}\sin(\pi z)*\cos[\pi(x+y)] - C_{44}\sin(\pi x)*\cos[\pi(y+z)]\} \quad (32)$$

$$f_z = 0.001*\pi^2*\{C_{11}*\sin(\pi x)*\sin(\pi y)*\sin(\pi z) - C_{12}*\cos(\pi z)*\cos[\pi(x+y)] - C_{44}\sin(\pi x)*\cos[\pi(y+z)] - C_{44}\sin(\pi y)*\cos[\pi(x+z)]\} \quad (33)$$

Thus, a block that is fixed to the entire boundary and to which a body force is applied is solved by the finite element method.

This example is solved for regular mesh but different element sizes h (0.25, 0.2, 0.125 and 0.1). The results show solution error as error energy norm $\|e\|_{eng}$ and error L_2 norm $\|e\|_{L2}$ with the following definitions [28]:

Error energy norm : $$\|e\|_{eng} = \left(\int_{\Omega} (\varepsilon - \varepsilon_h)^T (\sigma - \sigma_h) dV \right)^{\frac{1}{2}} \quad (34)$$

Error L_2 norm : $$\|e\|_{L2} = \left(\int_{\Omega} |u - u_h|^2 dV \right)^{\frac{1}{2}} \quad (35)$$

where **u** is the exact solution vector which includes u_x, u_y and u_z components. σ is the exact stress vector which includes σ_x, σ_y, σ_z, τ_{yz}, τ_{xz} and τ_{xy} components. ε is the exact strain vector which includes ε_x, ε_y, ε_z, γ_{yz}, γ_{xz} and γ_{xy} components. u_h is the finite element approximate solution vector, σ_h is the finite element approximate stress vector and ε_h is the finite element approximate strain vector. The approximate stress and strain are directly evaluated by all nodal deformations and shape functions using the stress-strain and strain-deformation relations without using any extrapolation or interpolation technique. The error norms and computation time of the proposed elements are compared with the traditional serendipity and Lagrange elements. The ratio of the reduced error norm is defined so that the change in the error norm of the elements is divided by the error norm of the serendipity elements or the Lagrange elements.

In the finite element analysis procedure we must deal with producing the element coefficient matrix, assembling the global coefficient matrix and solving the system of equations. Therefore, computation times are separated into three groups in the following tables: matrix formation time (T_f), static condensation time (T_c) and equation solution time (T_s). The iteration count for precondition conjugate gradient method (N_{PCG}) and the total computation time (T_{total}) are also shown in the following tables. Matrix formation time includes the computation time for producing the element coefficient matrix and assembling the global coefficient matrix. Static condensation time includes the computation time for calculating the static condensation and interior node dependent variables at the element level. The time for solving the system of equations is listed in the equation solution time.

Error norm and computation time of first order hexahedral elements

The error energy norm of the first order hexahedral elements is shown in Table 3. From Table 3 the 1S2L element, with one extra node in the center, reduces the error energy norm about 32%, while the 1S3L element, with eight extra nodes, reduces the energy norm about 46% when compared with the 1S element. The error L_2 norm in Table 4 shows that 1S2L and 1S3L elements reduce the error L_2

norm about 60%.

Table 3. Error energy norm of the first order hexahedral elements

h(m)	1S	1S2L	1S3L
0.25	2.872E+02	1.966E+02	1.552E+02
0.2	2.298E+02	1.555E+02	1.226E+02
0.125	1.437E+02	9.606E+01	7.556E+01
0.1	1.150E+02	7.665E+01	6.026E+01

Table 4. Error L2 norm of the first order hexahedral elements

h(m)	1S	1S2L	1S3L
0.25	4.105E-05	1.318E-05	1.556E-05
0.2	2.649E-05	7.964E-06	9.691E-06
0.125	1.046E-05	2.925E-06	3.678E-06
0.1	6.711E-06	1.846E-06	2.338E-06

Table 5. Computation time of the first order hexahedral elements

h(m)	T(sec)	1S	1S2L	1S3L
0.25	T_f	0.01	0.01	0.02
	T_c	0	0	0
	T_s	0.01	0.01	0.01
	N_{PCG}	7	7	7
	T_{total}	0.02	0.02	0.03
0.2	T_f	0.02	0.02	0.05
	T_c	0	0	0.02
	T_s	0.02	0.02	0.02
	N_{PCG}	8	8	8
	T_{total}	0.04	0.04	0.09
0.125	T_f	0.09	0.11	0.2
	T_c	0	0	0.07
	T_s	0.06	0.06	0.06
	N_{PCG}	12	12	12
	T_{total}	0.15	0.17	0.33
0.1	T_f	0.17	0.18	0.3
	T_c	0	0.03	0.23
	T_s	0.14	0.14	0.131
	N_{PCG}	14	14	14
	T_{total}	0.31	0.35	0.661

The computation time for the first order hexahedral elements are shown in Table 5. From the table we can see the equation solution time for the 1S2L and 1S3L elements is quite similar to that of the 1S element. The total computation time of the 1S2L element is as same as the 1S element. But the total computation time of the 1S3L element is much more than the 1S element. From Table 5, we can find the matrix formation time of the 1S3L element increases significantly. It is because the 1S3L element has much larger coefficient matrix sizes than the 1S element and needs the 27-point Gauss-Legendre quadrature formula.

Error norm and computation time of second order hexahedral elements

The error energy norm of the second order hexahedral elements is shown in Table 6 and the error L_2 norm of the second order hexahedral elements is shown in Table 7. The error norm of the 2S2L and 2L element is very close to the error norm of the 2S element. From Table 6, the 2S3L element, with eight extra nodes, reduces the error energy norm about 32% when compared with the 2S element and

reduces the energy norm about 31% when compared with the 2L element. Table 7 shows that 2S3L element reduces the error L_2 norm about 40% when compared with both the 2S element and the 2L element. The proposed 2S3L elements provide significantly superior accuracy compared with the traditional comparable serendipity and Lagrange elements.

Table 6. Error energy norm of the second order hexahedral elements

h(m)	2S	2L	2S2L	2S3L
0.25	3.098E+01	2.906E+01	2.929E+01	1.975E+01
0.2	1.932E+01	1.856E+01	1.866E+01	1.271E+01
0.125	7.374E+00	7.277E+00	7.286E+00	5.010E+00
0.1	4.697E+00	4.659E+00	4.662E+00	3.213E+00

Table 7. Error L2 norm of the second order hexahedral elements

h(m)	2S	2L	2S2L	2S3L
0.25	3.024E-06	2.995E-06	2.985E-06	1.596E-06
0.2	1.542E-06	1.530E-06	1.529E-06	8.609E-07
0.125	3.730E-07	3.716E-07	3.716E-07	2.224E-07
0.1	1.904E-07	1.899E-07	1.899E-07	1.154E-07

Table 8. Computation time of the second order hexahedral elements

h(m)	T(sec)	2S	2L	2S2L	2S3L
	T_f	0.13	0.24	0.15	0.19
	T_c	0	0	0	0.04
0.25	T_s	0.15	0.13	0.11	0.1
	N_{PCG}	31	15	22	20
	T_{total}	0.28	0.37	0.26	0.33
	T_f	0.27	0.48	0.28	0.37
	T_c	0	0	0.01	0.09
0.2	T_s	0.34	0.341	0.22	0.201
	N_{PCG}	40	20	25	23
	T_{total}	0.61	0.821	0.51	0.661
	T_f	1.051	2.163	1.121	1.651
	T_c	0	0.04	0.04	0.272
0.125	T_s	2.223	2.083	1.302	1.241
	N_{PCG}	64	31	37	35
	T_{total}	3.274	4.286	2.463	3.164
	T_f	2.083	4.447	2.263	3.145
	T_c	0	0.52	0.05	0.58
0.1	T_s	5.388	5.048	3.165	2.895
	N_{PCG}	80	39	46	42
	T_{total}	7.471	10.015	5.478	6.62

Table 8 shows the computation times for the second order hexahedral elements. The equation solution time for the 2S2L and 2S3L elements is smaller than the 2S element. The equation solution time is always smaller for the second order elements relative to the 2S element since the number of iterations is reduced. However, the matrix formation time of the 2S3L element is greater than the 2S element, so total computation time of the 2S3L element is slightly smaller than the serendipity elements. Although the 2S2L element does not improve the accuracy, its total computation is much less than the 2S element. Therefore the 2S2L element can replace the 2S element since it can save total computation time when the second order element is chosen.

Error norm and computation time of third order hexahedral elements

The error energy norm and error L_2 norm of the third order hexahedral elements are shown in

Tables 9 and 10. From Tables 9 and 10, the 3S2L and 3S3L elements reduce the error energy norm and error L_2 norm about 60% when compared with the 3S element. The 3S4L element reduces the error energy norm and error L_2 norm about 72% when compared with the 3S element. The 3L element has the best accuracy of the third order hexahedral elements.

Table 9. Error energy norm of the third order hexahedral elements

h(m)	3S	3L	3S2L	3S3L	3S4L
0.25	1.027E+01	1.906E+00	4.061E+00	3.697E+00	2.878E+00
0.2	5.105E+00	9.774E-01	1.958E+00	1.831E+00	1.398E+00
0.125	1.206E+00	2.391E-01	4.449E-01	4.311E-01	3.215E-01
0.1	6.125E-01	1.225E-01	2.238E-01	2.189E-01	1.623E-01

Table 10. Error L2 norm of the third order hexahedral elements

h(m)	3S	3L	3S2L	3S3L	3S4L
0.25	6.660E-07	1.075E-07	2.259E-07	2.134E-07	1.872E-07
0.2	2.579E-07	4.410E-08	8.525E-08	8.234E-08	7.191E-08
0.125	3.706E-08	6.729E-09	1.183E-08	1.176E-08	1.024E-08
0.1	1.497E-08	2.755E-09	4.735E-09	4.741E-09	4.128E-09

Table 11. Computation time of the third order hexahedral elements

h(m)	T(sec)	3S	3L	3S2L	3S3L	3S4L
0.25	T_f	0.37	2.504	0.37	0.44	1.663
	T_c	0	0.21	0	0.05	0.39
	T_s	0.451	0.921	0.441	0.301	0.28
	N_{PCG}	38	22	33	25	24
	T_{total}	0.821	3.635	0.811	0.791	2.333
0.2	T_f	0.661	4.516	0.681	0.871	3.234
	T_c	0	0.371	0.01	0.1	0.681
	T_s	1.001	2.243	0.891	0.641	0.641
	N_{PCG}	44	28	39	28	28
	T_{total}	1.662	7.13	1.582	1.612	4.556
0.125	T_f	2.794	18.689	2.764	3.435	13.27
	T_c	0	1.58	0.04	0.571	2.682
	T_s	6.419	13.71	5.117	3.885	3.706
	N_{PCG}	72	43	57	43	41
	T_{total}	9.213	33.979	7.921	7.891	19.658
0.1	T_f	5.498	37.425	5.488	6.93	26.651
	T_c	0	2.663	0.08	1.031	5.345
	T_s	15.433	52.035	11.858	9.023	8.923
	N_{PCG}	90	54	69	52	51
	T_{total}	20.931	92.123	17.426	16.984	40.919

The computation times of the third order hexahedral elements are shown in Table 11. It is seen that the equation solution times of the proposed elements are smaller than the serendipity elements because the number of iterations for the precondition conjugate gradient method decreased significantly as the number of interior nodes increase. Use of the enriched third order hexahedral elements not only improves accuracy but also saves total computation time.

Summary

From above numerical testing, we find that the 1S2L element is better than the 1S element because the 1S2L element clearly improves the accuracy when compared with the 1S element. Although the 1S3L element improved about 60% error norm, it requires still more interior nodes and

more computation time than the 1S2L element. The well-know deficiency associated with linear hexahedral element is its inability to represent the bending state. The 1S2L and 1S3L elements also present the same deficiency in modeling the bending state.

From the error norm, the 1S element, the 1S2L element and the 1S3L element show that they have the same convergence rate. The 2S, 2L, 2S2L and 2S3L elements likewise have similar convergence rate. Similarly, the 3S, 3S2L, 3S3L and 3S4L elements have equivalent convergence rate. The enriched terms cannot provide the complete high order polynomial so that convergence rates of the proposed element are just as same as the ones of the comparable serendipity element, but the numerical results show that the error can be reduced proportionately when using the proposed elements.

As the result of second and third order elements, we find that the number of iterations for precondition conjugate gradient method is decreased by static condensation. Because of the decrease in the number of iterations for PCG, use of the 2S3L, 3S2L and 3S3L elements not only improves accuracy but also saves total computation time. Although the 2S2L element does not improve the accuracy, it can save the total computation time. Therefore the increase in static condensation computation time can be ignored when the equation solution time is much greater than the matrix formulation time.

A practical elastic problem

A hexagonal wrench, applied a loading $F = 1000\,\mathrm{N}$, is considered in this section to verify the performance of the proposed elements. The finite element mesh and the boundary condition are shown in Figure 2. The boundary conditions are that twelve inside corners of one end are fixed and applied a loading to another end. The total number of elements is 1215. The material properties are set as $E = 2\times10^{11}$ and $\nu = 0.3$.

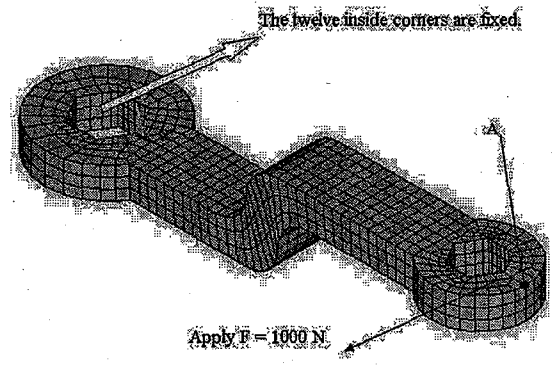


Figure 2. The mesh and the boundary condition of the hexagonal wrench

Table 12. The numerical results of the hexagonal wrench example

	DOF	T_f	T_c	T_s	N_{PCG}	T_{total}	deflection
1S	5892	0.19	0	3.395	277	3.585	0.67447
1S2L	9537	0.22	0.01	3.365	274	3.595	0.68654
1S3L	35052	0.431	0.16	3.284	267	3.875	0.73784
2S	21075	2.313	0	125.451	1454	127.764	0.77020
2L	37653	5.226	0.853	103.98	636	110.059	0.82241
2S2L	24720	2.414	0.06	66.065	765	68.539	0.77549
2S3L	50235	3.634	0.722	61.799	720	66.155	0.80190
3S	36258	6.539	0	359.808	1615	366.347	0.81001
3L	117150	57.514	6.558	1122.29	954	1186.362	0.96442
3S2L	39903	6.228	0.152	268.185	1201	274.565	0.81459
3S3L	65418	7.981	1.544	286.431	1281	295.956	0.92286
3S4L	134673	31.576	5.659	201.479	903	238.714	0.85608

* The above deflections are the computational deflections to be divided by the reference deflection 9.121×10^{-3}.

The computation time and the deflection of point A to force direction are shown in Table 12. DOF in the table means the total degree of freedom, including both interior freedoms and edge freedoms. A reference deflection 9.1216×10^{-3} is pointed out by ANSYS, using element type is Solid95 and number of elements is 87672 (total 377635 nodes). From the numerical results, the solution procedure dominates the computation time on a complex geometry, distortion elements and large number of nodes. The proposed elements show the advantage on both the accuracy and efficient. Especially, the 2S3L and 3S3L elements can provide a more accurate solution and save a lot of computation time when comparing with the comparable serendipity elements.

CONCLUSION

We have presented a family of enriched elements for finite element analysis. These elements are a combination of several of the characteristics of traditional Lagrange and serendipity elements. The proposed elements have demonstrated improved accuracy and computation time relative to their Lagrange and serendipity counterparts for at least certain cases.

Numerical experiments showed that error norms in all the tested cases were reduced from 40% to over 60% and use of the enriched second and third order hexahedral elements not only improves accuracy but also saves total computation time when compared with the serendipity elements. In addition, the convergence rate of the nSmL element is as same as the traditional serendipity element.

Since the enriched elements, static condensation and precondition conjugate gradient method cooperate, the accuracy is improved and the computation time is saved in finite element analysis. By the use of the static condensation technique at the element level, extra computation time using these enriched elements can be ignored and the procedure can be regarded as a preconditioner. When the precondition conjugate gradient method is used to solve the system of equations, the number of iterations can be reduced. Hence the computation time can be cut even though the dimension of the coefficient matrix is as same as the serendipity element.

The use of the 1S2L and 1S3L elements can improve the accuracy, but we should notice that the 1S2L and 1S3L elements present a deficiency in modeling the bending state. Although the 2S2L element does not improve the accuracy, its total computation is much less than the 2S element. Therefore the 2S2L element can be used to save the total computation time. The 2S3L element is able to predict greater accuracy and just needs a little more time than the 2S2L element, so the 2S3L element is also a good choice. Since the 3S3L element provides significantly superior accuracy and less computation time compared with the 3S element, it is recommended for the third order element.

REFERENCES

[1] Herrmann, L.R., Efficiency Evaluation of A Two-Dimensional Incompatible Finite Element, Computers and Structures, 3, 1973, 1377-1395.

[2] Wilson, E.L., Taylor, R.L., Doherty, W.P. and Ghaboussi, J., Incompatible Displacement Models, in: S.J. Fenves, et al, ed., Numerical and Computer Methods in Structural Mechanics, Academic Press, New York, 1973, 43-57.

[3] Taylor, R.L., Beresford, P.J. and Wilson, E.L., A Non-Conforming Element for Stress Analysis, International Journal for Numerical Methods in Engineering, 10, 1976, 1211-1219.

[4] Chen, W. and Cheung, Y.K., The Nonconforming Element Method and Refined Hybrid Element Method for Axisymmetric Solid, International Journal for Numerical Methods in Engineering, 39, 1995, 2509-2529.

[5] Taiebat, H.H. and Carter, J.P., Three-Dimension Non-Conforming Elements, Research Report R808, Department of Civil Engineering, The University of Sydney, 2001.

[6] Alves de Sousa, R.J., Natal Jorge, R.M., Areias, P.M.A., Fontes Valente, R.A. and César de Sá, J.M.A., Low Order Elements for 3D Analysis, in: Proceedings of the FifthWorld, Congress on Computational Mechanics, Vienna, Austria, 2002, 1-13.

[7] Rathod, H.T. and Sridevi, K., General complete Lagrange family for the cube in finite element interpolations, Computer Methods in Applied Mechanics and Engineering, 181, 2000, 295-344.

[8] Rathod, H.T. and Sridevi, K., General complete Lagrange interpolations with applications to

three-dimensional finite element analysis, Computer Methods in Applied Mechanics and Engineering, 190, 2001, 3325-3368.
[9] Celia, M.A. and Gray, W.G., An improved isoparametric transformation for finite element analysis, International Journal for Numerical Methods in Engineering, 20, 1984, 1443-1459.
[10] Citipitioglu, E., Universal serendipity elements, International Journal for Numerical Methods in Engineering, 19, 1983, 803-810.
[11] Utku, M., An improved transformation for universal serendipity elements, Computers and Structures, 73, 1999, 199-206.
[12] Kikuchi, F., Okabe, M. and Fujio, H., Modification of the 8-node serendipity element, Computer Methods in Applied Mechanics and Engineering, 179, 1999, 91-109.
[13] Cook, R.D., Malkus, D.S., Plesha, M.E. and Witt, R.J., Concepts and applications of finite element analysis, John Wiley & Sons, New York, 2002.
[14] Pinsky, P.M. and Jasti, R.V., A mixed finite element formulation for Reissner-Mindlin plates based on the use of bubble functions, International Journal for Numerical Methods in Engineering, 28, 1989, 1677-1702.
[15] Hong, W.I., Kim, Y.H. and Lee, S.W., An assumed strain triangular solid shell element with bubble function displacements for analysis of plates and shells, International Journal for Numerical Methods in Engineering, 52, 2001, 455-469.
[16] Kemp, B.L., Cho, C. and Lee, S.W., A four-node solid shell element formulation with assumed strain, International Journal for Numerical Methods in Engineering, 43, 1998, 909-924.
[17] Auricchio, F. and Taylor, R.L., A triangular thick plate finite element with an exact thin limit, Finite Elements in Analysis and Design, 19, 1995, 57-68.
[18] Brezzi, F., Bristeau, M.O., Franca, L.P., Mallet, M. and Roge, G, A relationship between stabilized finite element methods and the Galerkin method with bubble functions, Computer Methods in Applied Mechanics and Engineering, 96, 1992, 117-129.
[19] Baiocchi, C., Brezzi, F. and Franca, L.P., Virtual bubbles and Galerkin-least -squares type methods(Ga.L.S), Computer Methods in Applied Mechanics and Engineering, 105, 1993, 125-141.
[20] Franca, L.P. and Farhat, C., Bubble functions prompt unusual stabilized finite element methods, Computer Methods in Applied Mechanics and Engineering, 123, 1995, 299-308.
[21] Brezzi, F. and Russo, A., Choosing bubbles for advection-diffusion problems, Mathematical Models and Methods in Applied Sciences, 4, 1994, 571-587.
[22] Franca, L. and Farhat. C., On the limitation of bubble functions, Computer Methods in Applied Mechanics and Engineering, 117, 1994, 225-230.
[23] Brezzi, F., Hughes, T.J.R., Marin, L.D., Russo, A. and Suli, E., A priori error analysis of residual-free bubbles for advection-diffusion problems, SIAM Journal on Numerical Analysis, 36, 1999, 1933-1948.
[24] Sangalli, G., Global and local error analysis for the residual-free bubbles method applied to advection-dominated problems, SIAM Journal on Numerical Analysis, 38, 2000, 1496-1522.
[25] Farhat, C. and Sobh, N., A coarse/fine preconditioner for very ill-conditioned finite element problems, International Journal for Numerical Methods in Engineering, 28, 1989, 1715-1723.
[26] Golub, G.H. and Van Loan, C.F., Matrix computations, third edition, Johns Hopkins, London, 1996.
[27] Macneal, R.H. and Harder, R.L., A proposed standard set of problems to test finite element accuracy, Finite Element in Analysis and Design, 1, 1985, 3-20.
[28] Zienkiewicz, O.C. and Taylor, R.L., The finite element method - volume 1 basic formulation and linear problems, McGraw-Hill, London, 1989.