On error estimation and adaptive refinement for element free Galerkin method. Part II : adaptive refinement

Lee, Chi King; Zhou, C. E.

2003

Lee, C. K., & Zhou, C. E. (2004). On error estimation and adaptive refinement for element free Galerkin method. *Computers & Structures*, 82(4-5), 429-443.

<https://hdl.handle.net/10356/103287>

<https://doi.org/10.1016/j.compstruc.2003.10.017>

© 2003 Elsevier. This is the author created version of a work that has been peer reviewed and accepted for publication by *Computers & Structures*, Elsevier. It incorporates referee's comments but changes resulting from the publishing process, such as copyediting, structural formatting, may not be reflected in this document. The published version is available at: [Article DOI: <http://dx.doi.org/10.1016/j.compstruc.2003.10.017>].

Downloaded on 26 Apr 2023 20:11:09 SGT

**On error estimation and adaptive refinement for
element free Galerkin method
Part II: Adaptive refinement**

C. K. Lee* and C. E. Zhou

*School of Civil and Environmental Engineering
Nanyang Technological University
Nanyang Avenue
Singapore 639798*

*E-mail address: ccklee@ntu.edu.sg

Pre-printed version of the paper appeared in the journal: **Computers and Structures**, Vol. 82, No. 4-5, pp429-443

Summary

In this paper, an adaptive refinement procedure using the element free Galerkin method (EFGM) for the solution of 2D linear elastostatic problems is suggested. Based on the numerical experiments done in Part I of the current study, in the proposed adaptive refinement scheme, the Zienkiewicz and Zhu (Z-Z) error estimator using the T-Belytschko (TB) stress recovery scheme is employed for the *a posteriori* error estimation of EFGM solution. By considering the *a priori* convergence rate of the EFGM solution and the estimated error norm, an adaptive refinement strategy for the determination of optimal node spacing is proposed. A simple point mesh generation scheme using pre-defined templates to generate new nodes inside the integration cells for adaptive refinement is also developed. The performance of the suggested refinement procedure is tested by using it to solve several benchmark problems. Numerical results obtained indicate that the suggested procedure can lead to the generation of nearly optimal meshes and the effects of singular points inside the problem domain are largely eliminated. The optimal convergence rate of the EFGM analysis is restored and the effectivity indices of the Z-Z error estimator are converging towards the ideal value of unity as the meshes are refined.

KEYWORDS: *Element free Galerkin method, stress recovery, a priori and a posteriori error estimation, point mesh generation, adaptive refinement*

1. Introduction

After more than two decades of research and development, highly efficient adaptive finite element refinement procedures had been developed to solve many problems in different areas of engineering applications [1-6]. In fact, effective adaptive finite element refinement schemes have been implemented in many research programs [7] and even in commercial analysis packages[8]. When developing an adaptive finite element procedure, an automatic mesh generation scheme is one of the most important components. However, to develop and implement automatic mesh generators for with good control of element size and grading is not an easy task. During the last decade, much research efforts were devoted to this area [9,10] and yet it still remains an intensive research topics in computational mechanics and computational geometry. When compared with the finite element method, the meshless methods enjoy much more flexibility in model generation since no or very few elements are needed to be prepared. Hence, the development of an automatic adaptive refinement procedure for the meshless methods is a relatively easy task provided that the following essential ingredients are available:

- (1) Reliable error (*a priori* and *a posteriori*) estimation schemes
- (2) An efficient adaptive refinement strategy (appropriate choice of node density and nodal support size)

In this paper, an automatic adaptive refinement procedure for the element free Galerkin method will be suggested. The developed refinement procedure will employ the T-Belytschko (TB) stress recovery scheme and the Zienkiewicz and Zhu (Z-Z) error estimator tested in Part I of the current study [11] for the *a posteriori* error estimation. An effective adaptive refinement strategy will be suggested to control the discretization error of the EFGM solution. Based on the *a priori* convergence rate prediction and the quantitative error distribution information predicted by the *a posteriori* error estimator, the suggested adaptive refinement strategy will determinate the new point spacing required for the refinement procedure. A simple yet effective point mesh generation procedure is then employed for the generation of nearly optimal refinement point meshes over the problem domain. The suggested refinement scheme will be applied for the solution of some standard benchmark 2D elastostatic problems and its performance will be assessed by comparing the convergence rates obtained with those obtained by uniform refinement.

2. *A priori and a posteriori error estimations for the EFGM*

2.1 *A priori error estimation for the EFGM*

Similar to the case of finite element analysis, in the EFGM, an *a priori* error estimator provides a guideline for predicting the convergence rate of the solution when the point mesh undergoes refinements. An *a priori* error estimator for the EFGM can usually be devised based on (i) the mathematical formulation of the analysis method and (ii) the *a priori* knowledge of the exact solution. In this section, a summary for the basic theoretical results available will be given here. For more details of the *a priori* error estimator for EFGM analysis, one may refer to references [12].

For the analysis for 2D elastostatic problem using the EFGM, the *a priori* error estimator can be written in the form

$$\|\mathbf{e}_u\| \leq Ch^{\min(p,\lambda)} \quad (1)$$

In Eqn. 1, C is a constant independent of h , the typical spacing of node in the point mesh. λ is the strength of singularity of the exact solution which is less than unity. p is the order of moving-least-square (MLS) fitting used in the EFGM analysis. $\|\mathbf{e}_u\|$ is the global error in energy norm of the EFGM solution defined as

$$\|\mathbf{e}_u\| = \left\{ \int_{\Omega} (\boldsymbol{\sigma} - \boldsymbol{\sigma}^h)^T \mathbf{D}^{-1} (\boldsymbol{\sigma} - \boldsymbol{\sigma}^h) d\Omega \right\}^{1/2} \quad (2)$$

In Eqn. 2, $\mathbf{D}$ is the usual material matrix while $\boldsymbol{\sigma}$ and $\boldsymbol{\sigma}^h$ are, respectively, the exact and the EFGM stresses.

From Eqn. 1, it can be seen that whenever there is any singularity present in the problem domain (e.g. a crack line or a re-entrant corner), the convergence rate of the EFGM solution will be reduced. Furthermore, many previous studies [13,14] done on the performance of the EFGM showed that the accuracy of the EFGM solution could be increased by raising the value of p . However, it should be mentioned that the total degree of freedoms of the EFGM analysis is independent of the value of p used in the MLS fitting.

For 2D problems, the relationship between h and NN , the number of nodes of the point mesh, is given by

$$h \propto NN^{-1/2} \quad (3)$$

Hence, Eqn. 1 can also be written in the form of

$$\|\mathbf{e}_u\| \leq C' NN^{-\frac{\min(p,\lambda)}{2}} \quad (4)$$

In Eqn. 4, C' is another constant independent of NN . It is hoped that the adaptive refinement procedure will be able to construct an optimal point mesh such that the density of the error norm of the solution will be uniformly distributed over the problem domain. In such a case, the effect of any singularity will be eliminated and the convergence rate will be improved to [15]

$$\|\mathbf{e}_u\| \leq C'' NN^{-\frac{p}{2}} \quad (5)$$

Note that, in general, the value of $\|\mathbf{e}_u\|$ will vary for different problems. Hence, during the adaptive EFGM analysis, the relative error in energy norm, η , defined as

$$\eta = \frac{\|\mathbf{e}_u\|}{\|\mathbf{u}\|} \quad (6)$$

will be adopted as a measure of the accuracy of the solution. In Eqn. 6, $\|\mathbf{u}\|$ is the energy norm of the exact solution defined as

$$\|\mathbf{u}\| = \left\{ \int_{\Omega} \boldsymbol{\sigma}^T \mathbf{D}^{-1} \boldsymbol{\sigma} d\Omega \right\}^{1/2} \quad (7)$$

2.2 *A posteriori error estimation for the EFGM*

In this study, the Zienkiewicz-Zhu (Z-Z) error estimator [15] is selected as the a posteriori error estimator so that the estimate error norm, $\overline{\|\mathbf{e}_u\|}$ is computed as

$$\overline{\|\mathbf{e}_u\|} = \left\{ \int_{\Omega} (\boldsymbol{\sigma}^p - \boldsymbol{\sigma}^h)^T \mathbf{D}^{-1} (\boldsymbol{\sigma}^p - \boldsymbol{\sigma}^h) d\Omega \right\}^{1/2} \quad (8)$$

In Eqn. 8, $\boldsymbol{\sigma}^p$ is a smoothed stress field obtained by a smoothing or recovery process. If both the exact and estimated error norms are known, the effectivity index, θ , can be computed as

$$\theta = \frac{\overline{\|\mathbf{e}_u\|}}{\|\mathbf{e}_u\|} \quad (9)$$

In Part I of the current study [11], a number of commonly used recovery procedures for the construction of $\boldsymbol{\sigma}^p$ were tested. It was found that, in general, the T-Belytschko (TB) recovery scheme is the most preferable one in the sense that while it is the simplest scheme tested, in most cases it can generate the most accurate recovery stress and the

best effectivity index. Hence, the TB stress recovery scheme will be used in this paper for the construction of the recovery stress field. By using the TB scheme, the smoothed stress, $\boldsymbol{\sigma}^p$, can be expressed as

$$\boldsymbol{\sigma}^p(\mathbf{x}) = \sum_{i=1}^{NN} \Phi_i(\mathbf{x}_i) \boldsymbol{\sigma}^h(\mathbf{x}_i) \quad (10)$$

In Eqn. 10, $\boldsymbol{\sigma}^h(\mathbf{x}_i)$ is the EFGM stress evaluated at node i , Φ_i is the set of MLS shape functions used in the EFGM. From Eqn. 10, it can be seen that the recovery stress is obtained simply by performing a MLS fitting over the nodal EFGM stress.

3. The adaptive refinement strategy for EFGM

3.1 Acceptable EFGM solution

The objective of the adaptive refinement scheme is to achieve an EFGM solution with relative error in energy norm, η_{fin} , less than some user prescribed value, η_{tar} :

$$\eta_{fin} \leq \eta_{tar} \quad (11)$$

Since, in general, the exact energy norm $\|\mathbf{u}\|$ is not known in advance, during the adaptive refinement, it will be approximated by $\|\mathbf{u}^h\|$, the energy norm of the EFGM solution. That is

$$\|\mathbf{u}\| \approx \|\mathbf{u}^h\| = \left\{ \int_{\Omega} (\boldsymbol{\sigma}^h)^T \mathbf{D}^{-1} (\boldsymbol{\sigma}^h) d\Omega \right\}^{1/2} \quad (12)$$

From Eqn. 12, the estimated relative error of the problem, $\bar{\eta}$, can be calculated as

$$\eta \approx \bar{\eta} = \frac{\|\mathbf{e}_u\|}{\|\mathbf{u}^h\|} \quad (13)$$

and the adaptive refinement EFGM procedure will be terminated if the following accuracy is reached

$$\bar{\eta} \leq \eta_{tar} \quad (14)$$

3.2 Global and local refinement indicators

From the experience obtained from adaptive finite element analysis [1-6], an adaptive refinement procedure will be optimal only when both the global and local errors of the solution are controlled properly. From Eqns. 11 and 13, a global refinement indicator can be defined as

$$\xi_g = \frac{\overline{\|\mathbf{e}_u\|}}{\eta_{\text{tar}} \|\mathbf{u}^h\|} \quad (15)$$

Clearly, a value of $\xi_g \leq 1$ implies that the global error criterion (Eqn. 14) is satisfied whereas $\xi_g > 1$ indicates that further refinement is needed.

The global refinement indicator can only indicate whether further refinement is needed or not. It alone cannot indicate where and how to refine the mesh locally. In order to construct an optimal point mesh that will be able to eliminate the effect of singularity, a local criterion based on the error norm density or the *specific error* [16] will be used.

If Ω is the area of the whole problem domain and Ω_i is the area of a sub-domain of Ω , then φ_i , the specific error over Ω_i , is defined as the square of error norm per unit area

$$\varphi_i = \frac{\|\mathbf{e}_u\|_i^2}{\Omega_i} \quad (16a)$$

In Eqn. 16a, $\|\mathbf{e}_u\|_i$ is the error norm over Ω_i . Similarly, the specific error of the EFGM solution over the whole problem domain Ω , φ , is defined as

$$\varphi = \frac{\|\mathbf{e}_u\|^2}{\Omega} \quad (16b)$$

The uniform error density (or uniform specific error) requirements implies that, for any subdomain Ω_i

$$\varphi_i = \varphi \quad \text{or} \quad \frac{\|\mathbf{e}_u\|_i^2}{\Omega_i} = \frac{\|\mathbf{e}_u\|^2}{\Omega} \quad (17)$$

By using the global error requirement (Eqns. 11 and 14), the local error criterion for the sub-domain Ω_i can be expressed as

$$\|\mathbf{e}_u\|_i \leq \eta_{\text{tar}} \|\mathbf{u}\| \sqrt{\frac{\Omega_i}{\Omega}} \quad (18a)$$

During the adaptive analysis, the exact error norms in Eqn. 18a are not known in advance and they are replaced by their estimated values. That is

$$\overline{\|\mathbf{e}_u\|}_i \leq \eta_{\text{tar}} \|\mathbf{u}^h\| \sqrt{\frac{\Omega_i}{\Omega}} \quad (18b)$$

In Eqn. 18b, $\overline{\|\mathbf{e}_u\|}_i$ is the estimated error norm over Ω_i and is defined as (c.f. Eqn. 8)

$$\|\overline{\mathbf{e}_u}\|_i = \left\{ \int_{\Omega_i} (\boldsymbol{\sigma}^p - \boldsymbol{\sigma}^h)^T \mathbf{D}^{-1} (\boldsymbol{\sigma}^p - \boldsymbol{\sigma}^h) d\Omega \right\}^{1/2} \quad (19)$$

Hence, the local refinement indicator, ξ_i , for Ω_i can now be written in the form (c.f. Eqn. 15)

$$\xi_i = \sqrt{\frac{\Omega}{\Omega_i}} \frac{\|\overline{\mathbf{e}_u}\|_i}{\eta_{\text{tar}} \|\mathbf{u}^h\|} \quad (20)$$

Based on the above local refinement indicator, one can determine whether the node spacing in the subdomain Ω_i should be refined ($\xi_i > 1$) or not ($\xi_i \leq 1$). A value of $\xi_i = 1.0$ implies that an optimal local node spacing is obtained in Ω_i .

In the EFGM analysis, since no element will be used to discretize and cover the problem domain, the most convenient and effective way to impose the above local error condition is to define the local refinement indicator over the integration cells of the background grid. Note that in practice, in order to integrate the global stiffness equation accurately so that the convergence rate of the EFGM solution is maintained, the size of the integration cells used must be of the same order as the node spacing. As a result, using the integration cells as the subdomains for the definition of the local refinement indicators will ensure that the local error criterion will be imposed almost over everywhere in the whole problem domain. (The only exception is at the singular point in which theoretically the specific error will be unbounded there [16].) Furthermore, if the integration cells cover the problem domain exactly, the estimated error norm for the whole problem domain can be computed by summing up the contributions from all the integration cells. That is

$$\|\overline{\mathbf{e}_u}\|^2 = \sum_{i=1}^{\text{NIC}} \|\overline{\mathbf{e}_u}\|_i^2 \quad (21)$$

In Eqn, 21, NIC is the number of integration cells which cover the problem domain.

Finally, it should be noted that the local error criterion (Eqn. 20) will only be used for the determination of new node spacing during the new point mesh generation process. It is not the termination criterion for the whole adaptive refinement process. Only the global error criterion (Eqns. 14 and 15) will be used as the termination criterion for the refinement procedure. An overall flow chart for the adaptive refinement procedure is shown in Fig. 1.

4. New adaptive point mesh generation

Since in the current adaptive refinement scheme the local refinement indicators are defined over the integration cells of the point mesh, a new adaptive point mesh will be generated by simply re-defining the nodes in the background integration cells in a cell-by-cell manner. For each integration cell in the point mesh, based on the local (cell) refinement indicator and the *a priori* error estimator (Eqns. 4 and 5), the required number of nodes within the cell will first be calculated. After that, the positions of the new nodes in the cell will be determined by using some pre-defined templates.

4.1 Definition of number of new nodes in an integration cell

Assume that the optimal convergence rate (Eqn. 5) is achieved and let $NN_{old,i}$ be the current number of nodes in the i th integration cell. Then by using Eqns. 18 and 20, the relation between the local refinement indicator and $NN_{new,i}$, the new number of nodes needed, can be simply expressed as

$$\frac{NN_{new,i}}{NN_{old,i}} = (\xi_i)^{2/p} \quad (22)$$

4.2 Pre-defined templates for new nodes generation in integration cells

After the number of new nodes required for the integration cell, $NN_{new,i}$, is computed, its value will be compared with some pre-defined templates for the determination of $NN_{fin,i}$, the final number of nodes used in the cell. In general, in order to generate an optimal point mesh, it is desirable that the change in nodal density should be as smooth as possible. As a result, one can expect that the more pre-defined templates are used, the better (or smoother) the node density transition will be resulted. However, the use of too many different templates will complicate the implementation task. In this study, it is found that optimal points meshes are generated in almost all practical cases if the following templates are used

$$NN_{fin,i} = \left\{ \begin{array}{ll} 0 & \text{if } NN_{new,i} \leq 0.5 \\ 1 & \text{if } 0.5 < NN_{new,i} \leq 1 \\ 3 & \text{if } 1 < NN_{new,i} \leq 4 \\ 5 & \text{if } 4 < NN_{new,i} \leq 7 \\ 9 & \text{if } 7 < NN_{new,i} \leq 11 \\ 13 & \text{if } 11 < NN_{new,i} \leq 16 \\ 16 & \text{if } 16 < NN_{new,i} \end{array} \right\} \quad (23)$$

The distributions of newly generated nodes in the cell for the templates stated in Eqn. 23 are shown in Fig. 2.

By using the pre-defined templates, the new point mesh generation is accomplished by placing new nodes to the integration cell in a simple cell-by-cell manner. The only complication in which special care is needed arises along the edge between two adjacent cells where different templates are used. In such case, the template that leads to a smaller number of nodes will be used. An example for such a situation is shown in Fig. 3. In Fig. 3, according to the template of the right-hand-side cell ($NN_{fin,i}=16$), four nodes (nodes 1, 2, 3 and 4) should be generated on the edge CD. However, as according to the left-hand-side template in which $NN_{fin,i}=13$, only three nodes (nodes 1, 5 and 4) will be needed. Therefore, only the nodes 1, 5 and 4 will be retained in the new point mesh.

4.3 Finite elements and integration cell generations

Since the coupled finite element-element free Galerkin method (FE-EFGM) is adopted in the current study for the imposition of essential boundary conditions [11], after the new point mesh is generated, a layer of 4-node linear quadrilateral (Q4) finite elements will be inserted along all the essential boundaries. The sizes of the elements will follow the local node spacing along the essential boundary as shown in Fig. 4. Note that since the convergence rate of the Q4 element is lower than the EFG solution when quadratic polynomial terms are used in the EFGM (Eqn. 5), one can expect that the overall convergence rate of the FE-EFGM analysis will be slightly lower than the one predicted in Eqn. 5 during quadratic analyses.

After the point mesh and the finite elements for the treatment of essential boundary conditions are generated, integration cells will be constructed to generate the global stiffness equations system. In general, in order to ensure the stability of the global stiffness equations system and maintain the convergence rate of the analysis, the sizes of the integration cells should not be larger than the spacing of the nodes. During an

adaptive analysis when the spacings of nodes are not uniform, an efficient way to define the integration cells is to employ the quadtree approach [17] so that every integration cell in the mesh will contain at most four nodes. However, in this study, in order to compute the local error norms (Eqn. 19) and the effectivity indices (Eqn. 9) accurately, dense uniform integration grids of size proportional to the smallest spacing of nodes which fit exactly the problem domains will be used instead[11]. In addition, during the computation of error norms for individual integration cell, the higher order (8×8) integration rule are employed while the normal 4×4 integration rule will be used to generate the global stiffness equations system.

4.4 *Variable support size determination*

For adaptive refinement, unlike the case of uniform refinement in which the support sizes for all the nodes in the mesh are the same, the support sizes of nodes in an adaptive mesh will vary from node to node in the current implementation. The reason is that a constant support size will not be able to reflect the local node spacing of an adaptive mesh. If one uses a value based on the smallest node spacing in the mesh, the local MLS fitting procedure will be unstable at where the node spacing is large. However, setting the support size based on the largest node spacing will increase the computational costs and reduce the accuracy of the EFGM solution [18]. Thus, in the current implementation, the support sizes of the nodes will be determined by imposing the condition that every integration point must be contained inside the supports of a minimum number of nodes (3 and 6 for linear and quadratic analyses respectively) in the mesh. This can be done by scanning over all the integration points used for forming the global stiffness equations. For a given integration point $\mathbf{x}_G$, a list of the minimum number of nodes that (i) are closest to $\mathbf{x}_G$ and (ii) will make the MLS fitting stable at $\mathbf{x}_G$ are determined. The distances from this set of nodes to $\mathbf{x}_G$ are then computed. These distances are obviously equal to the minimum support sizes for this set of nodes with respect to $\mathbf{x}_G$. After checking over all the integration points in the mesh, the minimum support size for a given node can be taken as the largest value obtained from all the integration points. It can be seen the support size generated by the current scheme is locally defined and variable throughout the problem domain.

It should be mentioned here the conceptual difficulty arose due to the suggested remeshing scheme and the used of background integration cells. In principle, the approximation properties (and thus refinement strategies) are solely related to the size of

the supports of the shape functions, i.e. to the nodal density and are in no way connected to the quadrature rules used. However, well-behaved MLS approximation requires that the union of supports of all nodes be sufficiently overlapped and such overlap area have to be accounted for in the numerical integration scheme [19]. In view of the above two different concepts, the current variable support scheme can be interpreted as a trade-off selection in which integration scheme is involved in the choice of support size.

5 Numerical Examples

5.1 Overview

In this section, the four benchmark problems which are used in Part I of the current study [11] will be used again to investigate the performance of the suggested adaptive EFGM refinement procedure. Detailed descriptions of these problems can be found in reference [11]. A summary of the key properties of these four benchmark problems is listed in Table 1.

In order to investigate the effectiveness of the suggested adaptive EFGM procedure, both uniform and adaptive refinements will be carried out. Detailed descriptions of the FE-EFGM analysis procedure can be found in reference [11] and will not be repeated here. The point meshes used during the uniform refinement are the same as those used in reference [11] for the testing of different stress recovery schemes. In all the benchmark problems, the first uniform meshes used are chosen as the initial meshes for the adaptive refinement process. The target relative errors (η_{tar} , Eqn. 14) for the four benchmark problems tested are listed in Table 1. The adaptive procedure will be stopped whenever it is found that Eqn. 14 is satisfied or when totally four analyses (three refinements) are carried out. For all the uniform and adaptive analyses done, the exact relative error η (Eqn. 6), the estimated relative error $\bar{\eta}$ (Eqn. 13) and the effectivity index θ (Eqn. 9) are computed. In addition, $R(\eta)$, the rate of convergence of the relative error obtained for both uniform and adaptive refinements are computed as

$$R(\eta) = \frac{\log\left(\eta_i / \eta_{i+1}\right)}{\log\left(NTD_{i+1} / NTD_i\right)} \quad (24)$$

In Eqn. 24, η_i and NTD_i are the exact relative error and the total degree of freedoms of the i th mesh respectively.

5.2 Numerical results obtained

5.1.1 Problem 1: Timoshenko beam

For this problem, the results obtained and the density of integration cell used for error estimation for uniform and adaptive refinements are summarized in Tables 2 and 3 respectively. The rates of convergence for both refinements are plotted in Fig. 5 while the adaptive point meshes generated are shown in Fig. 6.

From the results of uniform refinement, it can be seen that the theoretical convergence rate are achieved for linear while the convergence rate for quadratic analyses is slightly lower than the theoretical value. As mentioned in Section 4.3, such a reduction in convergence rate of quadratic analysis is due to the fact that linear Q4 elements (with convergence rate equal to 0.5) are employed for the treatment of essential boundary condition. For an adaptive analysis, two and one refinements (i.e. totally three and two analyses) are needed to achieve the desired accuracies for the linear and the quadratic analyses respectively. From Table 3 and Fig. 6, it can be seen when adaptive refinement is used, the theoretical convergence rates are achieved for both linear and quadratic analyses and the accuracy of the solutions are improved as well. In addition, even this problem contains no singular point, from Fig. 6, one can see that the node spacing is reduced near the end of the support in which high stress gradients are located.

Regarding the effectivity indices obtained, from Tables 2 and 3, it can be seen that for this problem in which the exact solution is smooth, all the effectivity indices converged to the ideal value of unity independent of the orders of the analysis and the types of refinements used.

5.2.2 Problem 2: Infinite plate with a hole

For this problem, the results obtained for uniform and adaptive refinements are shown in Tables 4 and 5 respectively. The convergence rates are plotted in Fig. 7 while the adaptive point meshes generated are shown in Fig. 8.

For uniform refinement, similar to the case of example one, nearly optimal convergence rate is obtained for linear analysis while there is a reduction in convergence rate of quadratic analysis.

For adaptive refinement, two refinements are needed to achieve the desired accuracies for both linear ($\eta_{\text{tar}}=5\%$) and quadratic ($\eta_{\text{tar}}=2.5\%$) analyses. It can be seen that denser nodes are generated near the opening in which the stress gradient is high. Similar to Example 1, when adaptive refinement is used, there is a slightly increase in the

convergence rates and the accuracy of the solutions for both linear and quadratic analyses. In addition, regardless the order of the EFGM analysis, the effectivity indices are also converging to unity for both uniform and adaptive refinements.

5.2.3 Problem 3: Region near the tip of a crack

For this problem, the results obtained for uniform and adaptive refinements are listed in Tables 6 and 7 respectively. The convergence rates are plotted in Fig. 9 while the adaptive point meshes used are shown in Figs. 10 and 11.

For uniform refinement, from Fig. 9 and Table 7, it is obvious that the convergence rates for both linear and quadratic analyses are affected by the presence of the singular point and are reduced to 0.25. Furthermore, from Table 6, one can also see that no obvious convergence of the effectivity indices can be observed.

For adaptive refinement, three refinements are carried out for both linear and quadratic analysis. From Table 7, it can be seen that significant improvements in the rates of convergence are obtained. For the linear analysis, the theoretical optimal convergence rate is restored. For the quadratic analysis, due to the use of linear elements for treatment of boundary condition, the actual convergence rate is again slightly lower than the optimal value. From Fig. 9, one can also see that the actual convergence rate is increased as the mesh is refined. This can be explained by the fact that the density of nodes of the initial mesh used (Fig. 10a) is too coarse near the vicinity of the singular point. Thus, the adaptive procedure requires a couple of refinement steps to generate a properly refined point meshes with reduced node spacing near the singular point.

For the performance of the error estimator, it is found that when the point meshes are refined adaptively, convergence of the effectivity indices can be observed. Thus, it can be concluded that the adaptive refinement strategy and the point mesh refinement scheme suggested successfully led to the generation of nearly optimal point meshes.

5.2.4 Problem 4: L-shape domain

For this problem, the results obtained from uniform and adaptive refinements are listed in Tables 8 and 9 respectively while the convergence rates plots are shown in Fig. 12. The adaptive point meshes generated are shown in Figs. 13 and 14.

For uniform refinement, similar to Problem 3, due to the presence of the singular point, the convergence rates for both linear and quadratic analyses are reduced approximately

to 0.27. Again, the effectivity indices obtained do not converge to the ideal value of unity as the meshes are refined.

For adaptive refinement, three refinements are carried out for both linear and quadratic cases. From Figs. 13 and 14, it can be seen that densely spaced nodes are generated near the re-entrant corner of the problem domain. From Table 9 and Fig. 12, one can see that the optimal value of 0.5 is obtained for linear adaptive analysis. The convergence rate obtained for quadratic adaptive analysis is largely improved and again increased as the mesh is refined such that the convergence rate obtained at the last step of refinement (0.931) is closed to the optimal value of unity. One can thus conclude that the effect of the singular point is again almost eliminated.

Finally, since nearly optimal meshes are generated, convergence of the effectivity indices again is observed as the meshes are refined.

6. Conclusion

In this paper, a simple and automatic adaptive refinement procedure for the element free Galerkin method (EFGM) is suggested. In this procedure, the Zienkiewicz and Zhu error estimator based on the T-Belytschko stress recovery scheme is employed for the *a posteriori* error estimation. The local refinement indicators are computed based on the target relative error required and the estimated error norm. A new and effective refinement strategy is suggested to determine the new node spacing of the adaptive point mesh. Adaptive refinement meshes are generated by a simple refinement scheme in which pre-defined templates are employed to define the new node spacing and distribution in the integration cells of the mesh. Numerical studies are carried out on four benchmark problems with and without singularities. From the numerical studies, it is found that the user prescribed target relative error can usually be reached in a few numbers of refinements. For problems without singularity, even though there is no significant improvement in the convergence rate, the suggested adaptive scheme can lead to solutions with higher accuracy. For problems in which singular point exists, it is found that for both linear and quadratic analyses, well graded and nearly optimal meshes are generated. As a result, when compared with uniform refinement, by using the proposed adaptive refinement scheme, far less degrees of freedom are needed to achieve the desired accuracy. Regarding the effects of the adaptive refinement process on the performance of the *a posteriori* error estimator, numerical results indicate that the

effectivity indices are converging to the ideal value of unity even when singular point is present while such a convergence is not observed in the case of uniform refinement.

References

1. Lohner R, Morgan K and Zienkiewicz OC (1985) "An adaptive finite element procedure for compressible high speed flows", Computer Methods in Applied Mechanics and engineering, Vol. 51, pp. 441-65
2. Zienkiewicz OC, Huang GC and Liu YC (1990) "Adaptive FEM computation of forming processes - Application to porous and non-porous materials", International Journal for Numerical Methods in Engineering, Vol. 30, pp. 1527-1553
3. Lee CK and Lo SH (1992) "An automatic adaptive refinement finite element procedure for 2D elastostatic analysis", International Journal for Numerical Methods in Engineering, Vol. 35, pp. 1967-1989
4. Lee CK and Hobbs RE (1997), "Automatic adaptive refinement for shell analysis using nine-node assumed strain element", International Journal for Numerical Methods in Engineering, Vol. 40, pp. 3601-3638
5. Lee CK and Hobbs RE (1998) "On solving nearly incompressible 2D problems using an adaptive refinement procedure", Communications in Numerical Methods in Engineering, Vol. 41 pp. 409-418
6. Lee CK and Hobbs RE (1998) "Automatic adaptive refinement for plate bending problems using Reissner-Mindlin plate bending elements", International Journal for Numerical Methods in Engineering, Vol. 41, pp. 1-63
7. Taylor RL, FEAP, A Finite Element Analysis Program Version 7.4 Released, <http://www.ce.berkeley.edu/~rlt/feap/>
8. LUSAS, MYSTRO User Guide, FEA, Ltd, UK, 1997
9. Special Issue on Adaptive Meshing, Part II, Finite Elements in Analysis and Design Vol. 25, (1997), Guest Editor: S. H. Lo
10. Special Issue on Adaptive meshing, International Journal for Numerical Methods in Engineering, Vol. 32, (1991) Guest Editors: Shephard MS and Weatherill NP
11. Lee CK and Zhou CE, "On error estimation and automatic adaptive refinement for element free Galerkin method. Part I: Stress recovery and a posteriori error estimation". To appear in Computers and Structures.
12. Krysl P and Belytschko T (1997), "Element-free Galerkin method: Convergence of the continuous and discontinuous shape functions", Computer Methods in Applied Mechanics and Engineering, Vol. 148, pp. 257-277.

13. Belytschko T, Lu YY, Gu L (1994) "Element-Free Galerkin Methods", International Journal for Numerical Methods in Engineering, Vol. 37, pp. 229-256.
14. Lu YY, Belytschko T, Gu L (1994) "A New Implementation of the Element Free Galerkin Method", Computer Methods in Applied Mechanics and Engineering, Vol. 113, pp. 397-414.
15. Zienkiewicz, OC and Zhu JZ. (1987) "A Simple Error Estimator and Adaptive Procedure for Practical Engineering Analysis", International Journal for Numerical Methods in Engineering, Vol. 24, pp. 337-357.
16. Onate E and Bugeda G (1993), "A study of mesh optimality criteria in adaptive finite element analysis", Engineering Computations, Vol. 10, pp. 307-321.
17. Baehmann PL, Wittchen SL, Shephard MS, Grice KR and Yerry MA (1987), "Robust, geometrically based, automatic two-dimensional mesh generation", International Journal for Numerical Methods in Engineering, Vol. 24, pp. 1043-1087.
18. Sukumar N, Moran B and Belytschko T (1998), "The natural element method in solid mechanics", International Journal for Numerical Methods in Engineering, Vol. 43, pp. 839-887.
19. Igor K, Sunil S (1997) "An Improved Element Free Galerkin Formulation", International Journal for Numerical Methods in Engineering, Vol.40, pp. 2953-2974.

Figure 1. Flow chart for the adaptive EFGM refinement procedure

Figure 2. Templates for node refinement in integration cell

Figure 3. Special case for node placement along an edge between two adjacent cells with different templates

Figure 4. Treatment of essential boundary by couple finite element-element free Galerkin method

Figure 5. Rate of convergence for Problem 1

(a) Initial mesh for both linear and quadratic adaptive refinement, 27 nodes.

(b) Mesh 1 for linear adaptive refinement, 252 nodes

(c) Mesh 2 for linear adaptive refinement, 885 nodes

(d) Mesh 1 for quadratic adaptive refinement, 182 nodes

Figure 6. Adaptive meshes generated from Problem 1

Figure 7. Rate of convergence for Problem 2

Figure 8. Adaptive meshes generated from Problem 2

Figure 9. Rate of convergence for Problem 3

Figure 12. Rate of convergence for Problem 4

Figure 10. Adaptive meshes generated from Problem 3, linear analysis

(a) Initial mesh, 45 nodes

(b) Mesh 1, 97 nodes

(c) Mesh 2, 202 nodes

(d) Mesh 3, 383 nodes

Figure 11. Adaptive meshes generated from Problem 3, quadratic analysis

Figure 13. Adaptive meshes generated from Problem 4, linear analysis

Figure 14. Adaptive meshes generated from Problem 4, quadratic analysis

Problem no.	Descriptions	Reference figures	Presence of singularity	λ	$\eta_{tar} (\%)$	
					Linear	quadratic
1	Timoshenko beam	*Fig. 9	No	---	5	2.5
2	Infinite plate with a hole	*Fig. 11	No	---	5	2.5
3	Region near the tip of a crack	*Fig. 13	Yes	0.5	5	5
4	L-shape domain	*Fig. 15	Yes	0.544	5	5

* Figure numbers of reference [11] in which the definition of the problems are given

Table 1. Summary description of benchmark problems

Mesh (integration cell)	DOF	Linear analyses		Quadratic analysis		θ	
		$\eta (\%)$	$R(\eta)$	$\eta (\%)$	$R(\eta)$	Linear	Quadratic
8×2	54	23.03	---	15.50	---	1.13	0.89
16×4	179	12.60	0.50	5.63	0.85	1.07	0.99
24×6	350	8.45	0.60	3.31	0.79	1.05	1.00
32×8	594	6.35	0.54	2.31	0.68	1.04	1.01

Table 2. Results for Problem 1: uniform refinement

Mesh	Integration cell density	DOF	$\eta (\%)$	$R(\eta)$	θ
Linear analysis					
Initial	8×2	54	23.03	---	1.13
1	32×8	504	6.43	0.57	1.03
2	128×32	1770	3.20	0.56	1.01
Quadratic analysis					
Initial	8×2	54	15.50	---	0.89
1	32×8	364	2.30	1.00	1.02

Table 3. Results for Problem 1: adaptive refinement

Mesh (integration cell)	DOF	Linear analyses		Quadratic analysis		θ	
		η (%)	R(η)	η (%)	R(η)	Linear	Quadratic
4×4	50	10.23	---	12.02	---	0.76	1.09
8×8	162	5.74	0.49	4.78	0.78	0.90	1.07
16×16	578	3.45	0.40	2.21	0.61	1.04	1.01

Table 4. Results for Problem 2: uniform refinement

Mesh	Integration cell density	DOF	η (%)	R(η)	θ
Linear analysis					
Initial	4×4	50	10.23	---	0.76
1	8×8	148	5.77	0.53	0.93
2	16×16	422	3.48	0.48	1.02
Quadratic analysis					
Initial	4×4	50	12.02	---	1.09
1	8×8	156	4.73	0.82	1.07
2	16×16	412	2.52	0.65	1.03

Table 5. Results for Problem 2: adaptive refinement

Mesh (integration cell)	DOF	Linear analyses		Quadratic analysis		θ	
		η (%)	R(η)	η (%)	R(η)	Linear	Quadratic
8×4	90	28.29	---	23.27	---	0.91	0.89
16×8	306	20.70	0.26	19.11	0.16	0.89	0.81
32×16	1122	14.88	0.25	13.69	0.26	0.89	0.81

Table 6. Results for Problem 3: uniform refinement

Mesh	Integration cell density	DOF	η (%)	R(η)	θ
Linear analysis					
Initial	8×4	90	28.29	---	0.91
1	24×12	306	17.08	0.41	0.92
2	72×36	568	10.35	0.81	0.93
3	216×108	1050	6.45	0.77	0.95
Quadratic analysis					
Initial	8×4	90	23.27	---	0.89
1	24×12	194	16.57	0.44	1.27
2	72×36	404	9.18	0.81	1.19
3	216×108	766	5.90	0.69	1.13

Table 7. Results for Problem 3: adaptive refinement

Mesh (integration cell)	DOF	Linear analyses		Quadratic analysis		θ	
		η (%)	$R(\eta)$	η (%)	$R(\eta)$	Linear	Quadratic
6×6	80	22.82	---	19.42	---	0.82	0.78
10×10	192	18.00	0.27	15.67	0.25	0.72	0.82
20×20	682	12.83	0.27	11.07	0.27	0.71	0.79

Table 8. Results for Problem 4: uniform refinement

Mesh	Integration cell density	DOF	η (%)	$R(\eta)$	θ
Linear analysis					
Initial	6×6	80	22.82	---	0.82
1	14×14	190	13.61	0.60	0.82
2	32×32	358	9.80	0.52	0.83
3	96×96	692	6.56	0.61	0.87
Quadratic analysis					
Initial	6×6	80	19.42	---	0.78
1	12×12	172	14.77	0.36	1.30
2	24×24	418	8.50	0.62	1.10
3	72×72	696	5.29	0.93	1.09

Table 9. Results for Problem 4: adaptive refinement