Springer

Journal of Mechanical Science and Technology 23 (2009) 2224~2235

Journal of Mechanical Science and Technology

www.springerlink.com/content/1738-494x
DOI 10.1007/s12206-009-0510-5

An adaptive procedure based on combining finite elements with meshless methods †

Hyun-Gyu Kim[1,*] and Hye-Young Kim[2]

[1]*Department of Mechanical Engineering, Seoul National University of Technology, Seoul, 139-743, Korea*
[2]*The Graduate School of Energy & Environment, Seoul National University of Technology, Seoul, 139-743, Korea*

(Manuscript Received January 11, 2008; Revised December 17, 2008; Accepted April 1, 2009)

Abstract

This paper presents a new method for combining finite elements with meshless methods, which increases the accuracy of computational solutions in a coarse mesh by adding nodes in the domain of interest. The present method shares the features of the finite element and meshless methods such as (a) the meshless interpolation of the MLS type is employed; (b) integration domains are consistent with support domains; and (c) essential boundary conditions can be applied directly. In the present method, a ground mesh with triangular or quadrilateral elements is constructed to define polygonal support domains, and then additional nodes are placed arbitrarily in a domain without the reconstruction of a mesh. The method is very useful in an adaptive calculation, because nodes can be easily added or removed without any remeshing process.

Keywords: Adaptive computations; Finite element methods; Meshless methods; Moving least square

1. Introduction

Meshing in the finite element method is one of the time-consuming works in the field of computational mechanics. The fundamental idea of meshless methods is quite promising for numerical simulations as alternative approaches which involve no mesh. Meshless techniques are also appealing because of their potential in adaptive calculations, where one can add or remove nodes in a particular region to control the accuracy of results. Although a number of meshless methods [1-5] have been proposed over the last decade, these methods have some difficulties in constructing support and integration domains to preserve a regularity of shape functions and a consistency in numerical integrations of a weak form over the problem domain, because the determination of support regions associated with nodes irregularly distributed in a domain invokes a complexity of the so-called meshless connectivity [6] to build approximation functions. For non-uniform node distribution, in particular, a careful concern is required for multiple coverings of support domains up to the order of basis. In addition, numerical integrations of a weak form in a domain essentially have to use subdivided regions such as a background mesh even though shape functions are defined without the aid of a mesh. As a result, standard meshless methods need a connectivity to define support and integration domains. The choice of support and integration domains in meshless methods still remains open.

Meshless methods have some difficulties in the definition of support and integration domains consistent with the characteristics of shape functions. The first difficulty is related to the number of coverages of support domains for constructing shape functions. Some methods [7, 8] using the Delaunay tessellation have been proposed for the definition of influence

† This paper was recommended for publication in revised form by Associate Editor Maenghyo Cho
*Corresponding author. Tel.: +82 2 970 6309, Fax.: +82 2 949 1458
E-mail address: khg@snut.ac.kr

domains to develop meshless interpolants. The second difficulty is related to the consistency of support and integration domains. The standard finite element methods use support and integration domains based on finite elements, and the shape functions of finite elements are easily well defined with polynomial functions to yield a consistent numerical integration. Most of the meshless methods, however, may not give accurate results for a problem with irregular node distributions, and should be revised support domains and background meshes for adaptive calculations by adding or removing nodes in a domain [9].

There have been several efforts to develop a method for combining salient features of finite element and meshless methods. Oden et al. [10] introduced a new hp-finite element method, by using a combination of the conventional FEM and the partition of unity, to achieve a different order of basis for each node. In this method, however, a new global mesh is needed to add nodes for refinements, and a careful choice of the basis functions has to be made to prevent their linear dependence. The so-called generalized finite element method (GFEM) [11] uses special functions from known analytical solutions in order to improve the FEM solution. Duarte et al. [12] proposed FE-based partition of unity with the same supports as corresponding FE shape functions, which has a similar concept to the method proposed here in that numerical integrations of shape functions and their products can be done by using the finite elements. The moving particle finite element method (MPFEM) [13] and the reproducing kernel element method (RKEM) [14] have been developed to combine the strengths of both finite elements and meshless methods by using the reproducing conditions. The MPFEM and the RKEM introduced a concept of general shape functions with a high order smoothness. Some methods [15, 16] have been proposed to enrich the finite element method with meshless methods, which have similar features to the present method in adding additional nodes without any remeshing process. The use of circular supports in these methods [15, 16] may produce a difficulty in numerical integration due to complex intersections of finite elements and circular supports. In the present scheme, polygonal supports for additional nodes are used to be aligned with the finite elements.

A new method using the moving least square based on a ground mesh with triangular or quadrilateral elements is proposed in this paper. The previous work [17] only considered a convex polygonal support domain with quadrilateral elements, but the present method makes it possible to solve problems using a ground mesh with generally-shaped triangular and quadrilateral elements. The finite elements connecting primary nodes are constructed as a ground mesh in a domain, and secondary nodes can be placed arbitrarily without reconstruction of a mesh. The key idea of the present method is that polygonal support domains of secondary nodes are defined on the basis of a ground mesh to be aligned with integration domains. Since most meshless methods do not take into consideration the background mesh in the construction of shape functions, an inaccurate result may be obtained because of a difficulty in numerical integrations of complex functions. The proposed method leads to a consistency of numerical integrations by aligning support and integration domains, and does not need a new ground mesh when nodes are added or removed in a domain. As a consequence, the present scheme is very useful in controlling errors by adding or removing nodes in a domain. In addition, a simple technique is proposed to apply essential boundary conditions directly. Numerical examples of two-dimensional solids are presented to demonstrate the efficiency of the proposed method.

2. Support and integration domains of a weak form

In the following, support and integration domains for the weak form of elastostatic problems are described. The equilibrium equations of linear elasticity, in a global domain Ω bounded by Γ, are given by

$$\sigma_{ij,j}+b_i=0 \quad \text{in} \quad \Omega \tag{1}$$

where σ_{ij} is the stress tensor, b_i are the body forces, $(\,)_{,j}$ denotes $\partial(\,)/\partial x_j$, and a summation over a repeated index is implied. The boundary conditions are assumed to be

$$u_i=\overline{u}_i \quad \text{on} \quad \Gamma_u \tag{2a}$$

$$\sigma_{ij}n_j=\overline{t}_i \quad \text{on} \quad \Gamma_t \tag{2b}$$

where Γ_u and Γ_t are the global boundaries with prescribed displacements and tractions, respectively. Using the divergence theorem, the weak form of the equilibrium equation is written as

$$\int_{\Omega}\sigma_{ij}\,v_{i,j}\,d\Omega=\int_{\Gamma_t}\bar{t}_i v_i\,d\Gamma+\int_{\Omega}b_i v_i\,d\Omega \tag{3}$$

where v_i is the test function, and $t_i=\sigma_{ij}n_j$ and n_j is the outward unit normal to the boundary $\partial\Omega$. To obtain the discrete equations from the formulation, based on meshless interpolations, the global forms of interpolations for displacements can be written as

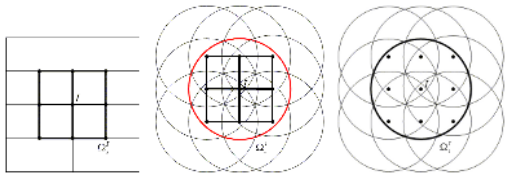
$$u_i^h(\boldsymbol{x})=\sum_{I=1}^{N}\phi^I(\boldsymbol{x})\hat{u}_i^I \tag{4}$$

where $\phi^I(\boldsymbol{x})$ is the nodal shape function centered at node I. In general, in meshless interpolations, $\hat{u}_i^I$ are fictitious nodal values.

Let $\{\Omega_s^I\}$ be a system of overlapping patches which covers the global domain Ω, where I $(=1,2,\cdots,N)$ indicates a node, and N is the total number of nodes. The sub-domain Ω_s^I is thus called the support domain of node I. To perform numerical integrations of the weak form, an integration domain is required for constructing the system stiffness matrix. Let Ω_D denote an integration domain which is not overlapping each other. The system stiffness matrix $\mathbf{K}$ can then be obtained by

$$\mathbf{K}=\sum_{D=1}^{N_D}\int_{\Omega_D}\sigma_{ij}v_{i,j}\,d\Omega \tag{5}$$

where N_D is the number of integration domains. Note that the values of shape functions at $\boldsymbol{x}$ inside Ω_D are influenced by a set of nodes in the vicinity of $\boldsymbol{x}$. The finite element method usually performs the numerical integration on the basis of support domains; the boundaries of support domains subdivide a global domain into integration domains of finite elements. Most of the meshless methods use background cells to integrate the weak form over a domain, in which integration domains do not match with the boundaries of support domains. The MLPG method [5] uses the support domains as integration domains, which may be a natural choice (see Fig. 1(c)). Similarly, the present method also uses support domains to define integration domains. Fig. 1 shows three methods used in integrating the weak form. The support domains in the FEM do not intersect each other in an integration domain or in an element. However, circular support domains as shown in Fig. 1b intersect each other in an integration domain.



(a) (b) (c)

Fig. 1. Support and integration domains: (a) finite element methods, (b) meshless methods with background cells, (c) meshless methods with support domains consistent with integration domains.

3. Characteristics of moving least square approximation

In this section, we review the characteristics of the moving least square (MLS) approximation, including some difficulties in the connectivity of meshless methods and numerical integrations. The MLS approximation always preserves the completeness up to the order of the basis, and reasonably interpolates the distributed nodal information. However, the nodal shape functions that arise from the MLS approximation have a very complex nature. This complexity results in difficulties with numerical integrations of a weak form. In addition, a definition of influence domains is required to preserve the connectivity of the MLS approximation.

We consider the approximation of a function $u(\boldsymbol{x})$ in a local region centered at $\tilde{\boldsymbol{x}}$ in a domain Ω. The moving least-square approximation starts from a local approximation in the neighborhood of $\tilde{\boldsymbol{x}}$, such as

$$u^{local}(\boldsymbol{x},\tilde{\boldsymbol{x}})=\mathbf{p}^T(\boldsymbol{x})a(\tilde{\boldsymbol{x}})\quad\forall\boldsymbol{x}\in B(\tilde{\boldsymbol{x}}) \tag{6}$$

where $B(\tilde{\boldsymbol{x}})$ is a sphere centered at $\tilde{\boldsymbol{x}}$, $\mathbf{p}^T(\boldsymbol{x})=[p_1(\boldsymbol{x}),p_2(\boldsymbol{x}),...,p_m(\boldsymbol{x})]$ is a complete monomial basis of order m; and $\mathbf{a}(\tilde{\boldsymbol{x}})$ is a vector containing the coefficients $a_j(\tilde{\boldsymbol{x}})$, $j=1,2,...,m$. The coefficient vector $\mathbf{a}(\tilde{\boldsymbol{x}})$ is determined by minimizing a weighted discrete L_2-norm, defined as

$$\begin{aligned}Y(\tilde{\boldsymbol{x}})&=\sum_{I=1}^{N}w^I(\tilde{\boldsymbol{x}})\left[\mathbf{p}^T(\boldsymbol{x}_I)\mathbf{a}(\tilde{\boldsymbol{x}})-\hat{u}^I\right]^2\\&=\left[\mathbf{P}\cdot\mathbf{a}(\tilde{\boldsymbol{x}})-\hat{\mathbf{u}}\right]^T\cdot\mathbf{W}(\tilde{\boldsymbol{x}})\cdot\left[\mathbf{P}\cdot\mathbf{a}(\tilde{\boldsymbol{x}})-\hat{\mathbf{u}}\right]\end{aligned} \tag{7}$$

where $w^I(\tilde{\boldsymbol{x}})$ is a weight function defined in a support domain Ω_s^I, with $w^I(\tilde{\boldsymbol{x}})>0$ for all $\tilde{\boldsymbol{x}}$ in Ω_s^I and $w^I(\tilde{\boldsymbol{x}})=0$ at the boundary of Ω_s^I, $\boldsymbol{x}^I$ denotes the value of $\boldsymbol{x}$ at node I, and the matrices $\mathbf{P}$, $\mathbf{W}$

and $\hat{\mathbf{u}}$ are defined as

$$\mathbf{P}=\begin{bmatrix}\mathbf{p}^T(\boldsymbol{x}_1)\\ \mathbf{p}^T(\boldsymbol{x}_2)\\ \cdots\\ \mathbf{p}^T(\boldsymbol{x}_N)\end{bmatrix}_{N\times m},$$

$$\mathbf{W}(\tilde{\boldsymbol{x}})=\begin{bmatrix}w^1(\tilde{\boldsymbol{x}}) & \cdots & 0\\ \cdots & \cdots & \cdots\\ 0 & \cdots & w^N(\tilde{\boldsymbol{x}})\end{bmatrix}_{N\times N},$$

$$\hat{\mathbf{u}}^T=\left[\hat{u}^1,\hat{u}^2,\cdots,\hat{u}^N\right] \tag{8}$$

For convenience, $\tilde{\boldsymbol{x}}$ in the above relations is replaced by $\boldsymbol{x}$, because a local point $\tilde{\boldsymbol{x}}$ can be extended to all points in whole domain. This is the basic concept of the "moving" procedure, and we can finally obtain a global approximation.

The stationary condition of $Y(\boldsymbol{x})$ with respect to the coefficients $\mathbf{a}(\boldsymbol{x})$ leads to the following relation:

$$\mathbf{A}(\boldsymbol{x})\mathbf{a}(\boldsymbol{x})=\mathbf{B}(\boldsymbol{x})\hat{\mathbf{u}} \tag{9}$$

where the matrices $\mathbf{A}(\boldsymbol{x})$ and $\mathbf{B}(\boldsymbol{x})$ are given by

$$\mathbf{A}(\boldsymbol{x})=\mathbf{P}^T\mathbf{W}\mathbf{P}=\sum_{I=1}^{N}w^I(\boldsymbol{x})\mathbf{p}(\boldsymbol{x}^I)\mathbf{p}^T(\boldsymbol{x}^I) \tag{10a}$$

$$\mathbf{B}(\boldsymbol{x})=\mathbf{P}^T\mathbf{W}=\left[w^1(\boldsymbol{x})\mathbf{p}(\boldsymbol{x}^1),\cdots,w^N(\boldsymbol{x})\mathbf{p}(\boldsymbol{x}^N)\right] \tag{10b}$$

The global approximation $u^h(\boldsymbol{x})$ can then be expressed as

$$u^h(\boldsymbol{x})=\sum_{I=1}^{N}\phi^I(\boldsymbol{x})\hat{u}^I \tag{11}$$

where the nodal shape functions are given by

$$\phi^I(\boldsymbol{x})=\mathbf{p}^T(\boldsymbol{x})\mathbf{A}^{-1}(\boldsymbol{x})\mathbf{B}(\boldsymbol{x}) \tag{12}$$

In the traditional FEM, the nodal shape functions have a value of unity at the respective nodes. However, in the MLS approximation, $\hat{u}^I$ are fictitious, and are not exactly equal to the nodal values of field variables. The MLS interpolation is well defined only when the matrix $\mathbf{A}$ in Eq. (10) is non-singular. A necessary condition for a well-defined MLS interpolation is that at least m weight functions are non-zero for each sample point $\boldsymbol{x}\in\Omega$. Consequently, the influence domains of non-uniform node distributions should be defined carefully to prevent from ill-conditioning of the construction of shape functions based on the MLS approximation. As a result, meshless connectivity is required even though the nodal connectivity as in the FEM is not needed in the MLS approximation. To preserve the regularity of MLS shape functions, the support domains should overlap at least three times in two dimensions with the linear basis. A careful choice of support domains for a non-uniform node distribution is required to guarantee the regularity of MLS shape functions.

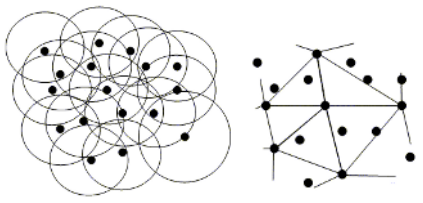
We can obtain an explicit form of nodal shape functions, with a linear basis, in a two-dimensional problem, in order to better understand the characteristics of nodal shape functions:

$$\phi^I(x,y)=w^I(x,y)\left[c_0(x,y),c_1(x,y),c_2(x,y)\right]\begin{Bmatrix}1\\ x\\ y\end{Bmatrix} \tag{13}$$

where the coefficients $c_0(x,y)$, $c_1(x,y)$ and $c_2(x,y)$ are given by Atluri et al. [18]. In general, the coefficients in Eq. (13) are not of a single type of functions, because the influence domains of neighboring nodes related to the support domains Ω_s^I make complex intersections due to crossing of the boundaries of influence domains. Consequently, the nodal shape functions $\phi^I(\boldsymbol{x})$ consist of many different forms of rational functions in the support domains Ω_s^I. It seems to be difficult to integrate these kinds of complex functions, by using a simple Gaussian quadrature rule, and this causes a difficulty in numerical integrations of a weak form.

4. Construction of polynomial supports and weight functions

This section first describes the procedure for constructing polygonal support domains on the basis of triangular and quadrilateral finite elements, and corresponding weight functions. Later, some advantages of the present method in the treatment of essential



(a) (b)

Fig. 2. Schematic representations of supports of randomly distributed nodes: (a) meshless methods with circular support domains, (b) the present method with polygonal support domains.

boundary conditions are pointed out. A ground mesh is constructed by connecting primary nodes, and the support domains of secondary nodes distributed arbitrarily in a domain are defined on the basis of the ground mesh. The proposed method always guarantees the regularity of MLS shape functions and the consistency of support and integration domains.

4.1 Polygonal support domains and weight functions

As explained in the previous section, the definition of support domains for meshless connectivity and the complex interaction of support domains in an integration domain are difficulties in meshless methods. In this study, the concept of a ground mesh is introduced to preserve the regularity of MLS shape functions, and to prevent the crossing of the boundaries of support domains in an integration domain. The polygonal support domains defined by a ground mesh connecting anchor or primary nodes are used for the support domains of nodes, called the secondary nodes, added or removed in a domain. As a result, no additional mesh is required for secondary nodes, which makes it possible to extend the meshless concept to be a useful tool for error controls and adaptive calculations. Fig. 2 shows two types of support domains: circular support domains and polygonal support domains defined by the primary nodes. Since the types of shape functions may change across the boundaries of support domains, the method shown in Fig. 2(a) may lead to a complexity in the weak form, for a non-uniform node distribution, due to the intersections of support domains. However, the present method in Fig. 2(b) uses simple polygonal intersections of support domains. As a result, shape functions in a polygonal intersection have a single type of rational function, because of the alignment of the boundaries of support domains. The support and integration domains of secondary nodes are defined on the basis of a ground mesh, and there is no modification of the ground mesh by adding or removing secondary nodes in a domain. Hence, the method presented here has an advantage in that one does not need to modify the mesh in adaptive computations.

In the present scheme, secondary nodes can be placed at arbitrary locations in the domain of interest, in order to improve the deformations. Secondary nodes placed randomly in the domain take the support domains to be the supports of the nearest primary or anchor nodes. Fig. 3 illustrates how to choose primary

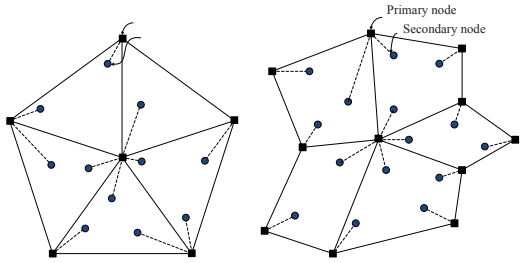


Fig. 3. Polygonal support domains for primary and secondary nodes: solid rectangles are the primary nodes and solid circles are the secondary nodes.

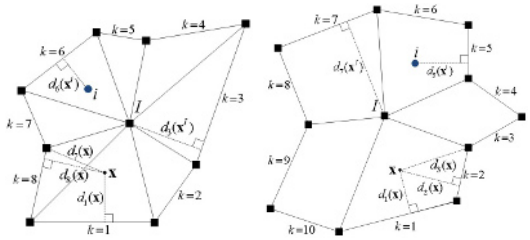


Fig. 4. Distances from the position $\mathbf{x}$ to the segments k: (a) a ground mesh with triangular elements, (b) a ground mesh with quadrilateral elements.

nodes that would be anchors for secondary nodes. The support domain of a primary node becomes a polygon consisting of triangles or rectangles in two dimensions. Therefore, it is very easy to find the support domains of secondary nodes once a ground mesh is constructed by using primary nodes. It should be noted that the polygonal support domains with triangular or quadrilateral elements can have convex and concave shapes.

We denote by I the primary node and by i the secondary nodes, related to the primary node I. Weight functions in the MLS approximation, which ultimately govern the shape functions (see Eqs. (10) and (12)), should be defined appropriately. It is favorable that there is no discontinuity of the derivatives of weight functions in a ground mesh. Moreover, it is desirable that the maximum values of weight functions are located at nodal positions. Weight functions of primary nodes are constructed by a set of functions which are appropriately zero along a given boundary of polygonal support domains. On a convex domain, a function which is zero along the boundary can be defined by the triangular simplex [19], in which the partition of unity was used to construct shape functions for polygonal domains with interior nodes. The

product of boundary functions $f_k^I(\mathbf{x})$ for triangles or rectangles k consisting a convex or concave polygonal support domain is used to build weight functions which vanish at the boundary of the support domain, i.e.,

$$w_I(\mathbf{x}) = \prod_{k=1}^{n_s} f_k^I(\mathbf{x}) \quad \text{on} \quad \Omega_s^I \tag{14}$$

with

$$f_k^I(\mathbf{x}) = q_k^I(\mathbf{x}) e^{1-q_k^I(\mathbf{x})}, \quad q_k^I(\mathbf{x}) = \frac{d_k(\mathbf{x})}{d_k(\mathbf{x}^I)} \tag{15}$$

where n_s is the number of edge segments of a polygon Ω_s^I, and $d_k(\mathbf{x})$ are the nearest distances from the position x to the segments k. Fig.4 illustrates the distances to define the boundary functions $f_k^I(\mathbf{x})$. The definition of the boundary functions $f_k^I(\mathbf{x})$ in Eq. (15) is similar to the cloud boundary function by Duarte et al. [12]. Fig. 5 plots the boundary functions $f_k^I(\mathbf{x})$ which have the maximum at $q_k^I(\mathbf{x}) = 1$. As a consequence, the weight functions at the nodes $\mathbf{x}^I$ have the maximum, $w_I(\mathbf{x}^I) = 1$, and zero at the boundaries of polygonal support domains. Note that the proposed procedure can be applied to both a convex and a concave polygonal support. The boundary functions increase 0 to 1 as the position $\mathbf{x}$ moves from the boundary of a polygon or a support domain to the nodal position $\mathbf{x}^I$. As a result, the product of the boundary functions constructs a smooth weight function of the node I with zero value at the boundary of a polygonal support domain Ω_s^I.

We now discuss the construction of weight functions of secondary nodes. Weight functions of secondary nodes should be zero at the boundaries of polygonal support domains of primary nodes, because the support domains of secondary nodes are those of primary nodes. For this purpose, we take weight functions of secondary nodes:

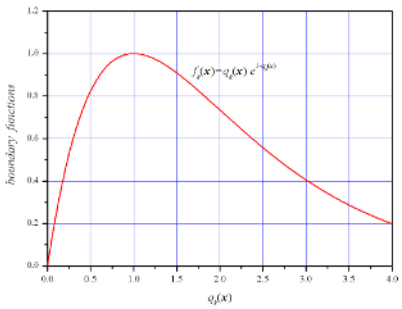
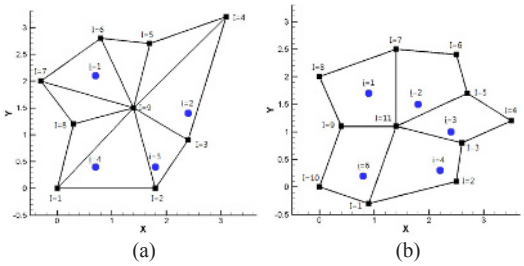


Fig. 5. Boundary functions $f_k(\mathbf{x})$ as a function of $q_k(\mathbf{x})$.

$$w_i(\mathbf{x}) = \prod_{k=1}^{n_s} f_k^i(\mathbf{x}) \quad \text{on} \quad \Omega_s^I \tag{16}$$

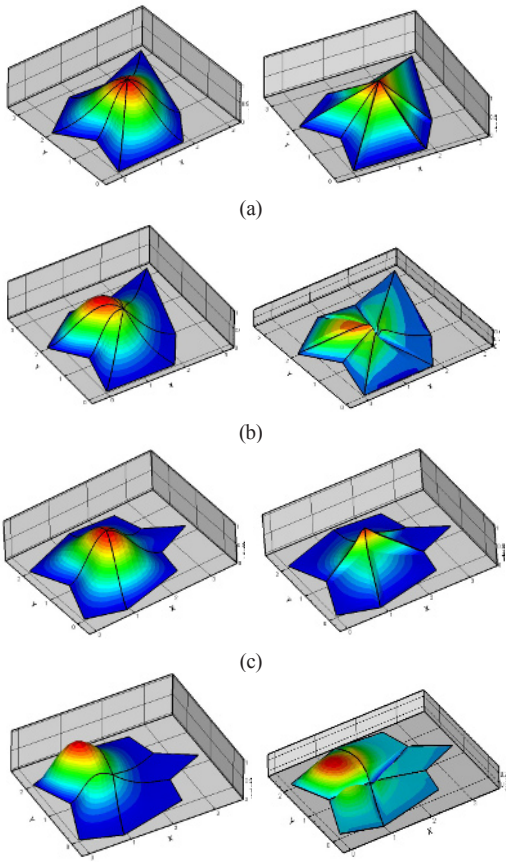
with

$$f_k^i(\mathbf{x}) = q_k^i(\mathbf{x}) e^{1-q_k^i(\mathbf{x})}, \quad q_k^i(\mathbf{x}) = \frac{d_k(\mathbf{x})}{d_k(\mathbf{x}^i)} \tag{17}$$



(a) (b)

Fig. 6. Primary and secondary nodes in polygonal domains: (a) a triangular ground mesh, (b) a quadrilateral ground mesh.



(a) (b) (c) (d)

Fig. 7. Weight (left) and shape (right) functions of the primary and secondary nodes in polygonal support domains of triangular and quadrilateral ground meshes; (a) primary node *I*=6, (b) secondary node *i*=1, (c) primary node *I*=9, (d) secondary node *i*=4.

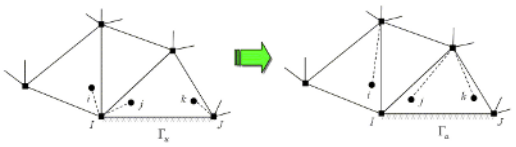
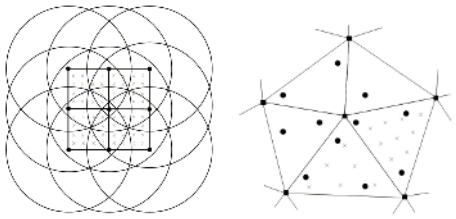


Fig. 8. Modification of anchor nodes for secondary nodes near by primary nodes prescribed essential boundary conditions.



(a) (b)

Fig. 9. Integration domains: (a) background integration cells, (b) a ground mesh.

It should be noted that the values of weight functions in Eq. (16) are zero at the boundary of the support domains $\Omega_s^I = \Omega_s^i$. The location of the maximum of weight functions $w_i(\mathbf{x})$ is the nodal position $\mathbf{x}^i$, and its value is 1. The derivatives of weight functions across the boundaries of the support domains are not continuous because the derivatives of the boundary functions are not zero on the boundaries. Thus, the stresses and strains across inter-element boundaries of a ground mesh are not continuous.

Fig. 6 shows secondary nodes within triangular and quadrilateral ground meshes constructed by connecting primary nodes. The weight and shape functions of primary and secondary nodes are plotted in Fig. 7. In these figures, weight functions are zero on the boundaries of polygonal support domains. Note that the weight functions of secondary nodes have skewed forms with the maximum values at $\boldsymbol{x}^i$. The shape functions of primary and secondary nodes depend on the number of nodes in the support domains and their locations. The shape functions of primary nodes are different from those of finite elements when secondary nodes exist in the support domains of primary nodes. It should be emphasized that the shape functions of primary nodes on a triangular ground mesh become the finite element shape functions unless secondary nodes are involved in the MLS approximation.

Again, emphasis is placed on the fact that the shape functions in the present method have a single form of rational functions all over the elements in a ground mesh, because there is no crossing of the boundaries of the support domains. In summary, the secondary nodes can be added at arbitrary positions in a domain, after the initial polygonal support domains are constructed from a ground mesh, and errors in numerical results can be controlled by adding or removing secondary nodes in a domain. Hence, to start with, only a simple ground mesh may be used, and later, a random pattern of secondary nodes may be introduced, in an adaptive fashion to control the error of numerical solutions.

4.2 Essential boundary conditions

One major difficulty in meshless methods is considered to be the imposition of essential boundary conditions, because, in general, approximation functions do not satisfy the Kronecker-delta condition, $\phi^I(\boldsymbol{x}_J) = \delta_{IJ}$. Most of the meshless methods have used Lagrange multipliers or penalty methods to impose essential boundary conditions. In some cases, meshless interpolations and FE shape functions have been combined, leading to a complex interface element in the regions of intersection of FE and meshless shape functions [20]. Similar to the conventional meshless methods, the present method also requires one of these techniques to impose essential boundary conditions because primary nodes are multiply overlaid by the support domains of the associated secondary nodes. An important feature is that primary nodes not associated with secondary nodes have the Kronecker-delta condition. Therefore, we can impose essential boundary conditions directly by changing the anchor nodes of secondary nodes related to primary nodes to impose essential boundary conditions. Fig. 8 illustrates the secondary nodes i, j and k related to the primary nodes I and J on the global boundary imposing essential boundary conditions. In order to preserve the Kronecker-delta property on the primary nodes I and J, the secondary nodes i, j and k do not take these primary nodes as anchor nodes. Through this simple modification, shape functions on the global boundaries become the finite element shape functions because the influence domains of secondary nodes do not overlay the boundaries connecting these primary nodes. As a result, we can impose essential boundary conditions directly at the primary nodes I and J.

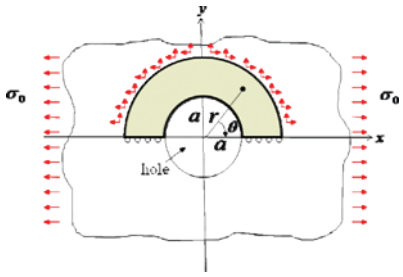
4.3 Numerical integrations

To evaluate the stiffness matrix from a weak form, it is necessary to use a numerical quadrature since analytical integration is all but impossible in general. The numerical integration of a weak form usually plays an important role in the convergence of numerical solutions in meshless methods. Fig. 9 shows that the schematic features of two integration methods in meshless methods. The first method, using a background mesh, has been used in most of the meshless methods. Dolbow and Belytschko [21] have already indicated that the integration using the background mesh is not adequate, to accurately integrate the terms in the stiffness matrix, when irregularly distributed nodes are used. They presented a method to reduce errors in numerical integrations, by making the integration cell to be aligned with the boundaries of support domains. As explained in the previous section, the present method uses the intersections of the supports as the integration domains to obtain a consistent numerical integration for the stiffness matrix. As a result, shape functions inside an integration domain become a single type of rational function.

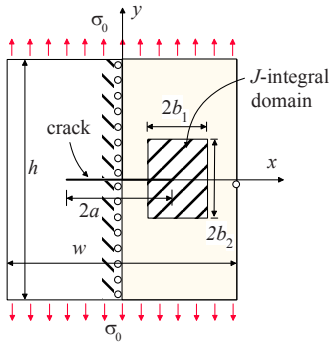
In this study, a symmetric quadrature [22] is used to numerically evaluate the integrals in a weak form. Since shape functions inside an integration domain are rational functions in the present method, the symmetric quadrature may not be adequate to evaluate integrals properly. However, the accuracy of numerical integrations may be controlled by the number of integration points, taking a proper level of polynomials as an approximation for these rational functions. In general, the more secondary nodes are involved in the support domains, the more integration points are required. Development of an efficient integration rule for rational functions is still an open question, to improve the performance of this method while using only a small number of integration points.

5. Numerical examples

In this section, numerical results of linear elastic problems, specifically a cantilever beam and a center cracked plate, are presented to illustrate the effectiveness of the present method. Fig. 10 describes the numerical examples. Young's modulus and Poisson's ratio are $E = 1.0 \times 10^{10}$ and $\nu = 0.25$, respectively. We use the displacement and energy norms defined as



(a)



(b)

Fig. 10. Geometric descriptions of numerical examples; (a) a plate with a circular hole, (b) a center cracked plate in tension.

$$\|\boldsymbol{u}\| = \left(\int_{\Omega} \boldsymbol{u}^T \boldsymbol{u} \, d\Omega \right)^{\frac{1}{2}} \tag{18a}$$

$$\|\boldsymbol{\varepsilon}\| = \left(\frac{1}{2} \int_{\Omega} \boldsymbol{\varepsilon}^T \mathbf{D} \boldsymbol{\varepsilon} \, d\Omega \right)^{\frac{1}{2}} \tag{18b}$$

The relative errors for $\|\boldsymbol{u}\|$ and $\|\boldsymbol{\varepsilon}\|$ are defined as

$$r_u = \frac{\left\| \boldsymbol{u}^{num} - \boldsymbol{u}^{exact} \right\|}{\left\| \boldsymbol{u}^{exact} \right\|} \tag{19a}$$

$$r_e = \frac{\left\| \boldsymbol{\varepsilon}^{num} - \boldsymbol{\varepsilon}^{exact} \right\|}{\left\| \boldsymbol{\varepsilon}^{exact} \right\|} \tag{19b}$$

The linear basis in the MLS approximation is used in the numerical examples. To verify the present scheme, secondary nodes are placed randomly in a domain.

5.1 Infinite plate with a circular hole

We first consider an infinite plate with a circular hole of radius a. The plate is subjected to a uniform tension, $\sigma_0 = 1.0 \times 10^9$, in the x-direction, at infinity as shown in Fig. 10a. The exact solutions for stresses are

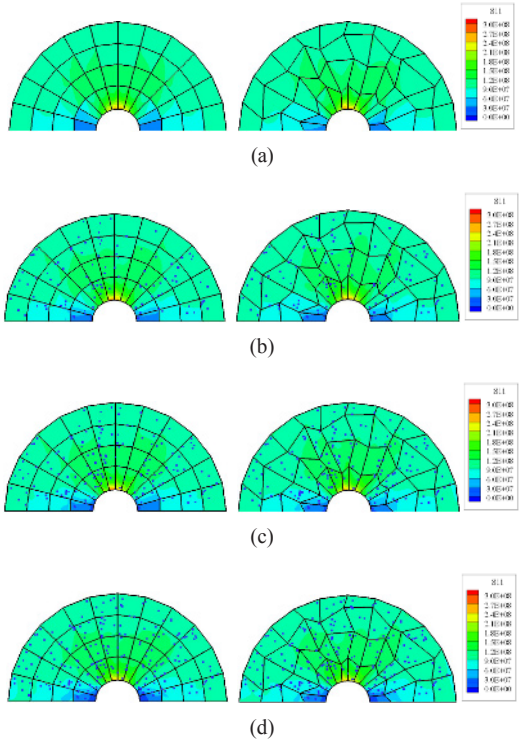


Fig. 11. Stress σ_{11} in a plate using ground meshes with regularly- and irregularly-shaped quadrilateral elements: (a) 0 secondary nodes, (b) 100 secondary nodes, (c) 150 secondary nodes, (d) 200 secondary nodes.

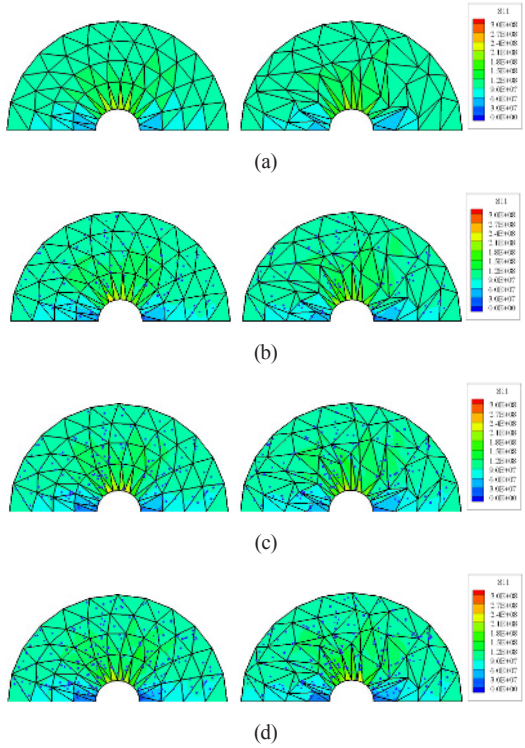


Fig. 12. Stress σ_{11} in a plate using ground meshes with regularly- and irregularly-shaped triangular elements: (a) 0 secondary nodes, (b) 100 secondary nodes, (c) 150 secondary nodes, (d) 200 secondary nodes.

$$\sigma_{11}=\sigma_0\left[1-\frac{a^2}{r^2}\left(\frac{3}{2}\cos 2\theta+\cos 4\theta\right)+\frac{3a^4}{2r^4}\cos 4\theta\right] \quad (20a)$$

$$\sigma_{12}=\sigma_0\left[-\frac{a^2}{r^2}\left(\frac{1}{2}\sin 2\theta+\sin 4\theta\right)+\frac{3a^4}{2r^4}\sin 4\theta\right] \quad (20b)$$

$$\sigma_{22}=\sigma_0\left[-\frac{a^2}{r^2}\left(\frac{1}{2}\cos 2\theta-\cos 4\theta\right)-\frac{3a^4}{2r^4}\cos 4\theta\right] \quad (20c)$$

where (r,θ) are the polar coordinates, and θ is measured from the positive x-axis. The corresponding displacements in the plane stress are given by

$$u_1=\frac{1+\nu}{E}\sigma\left(\frac{1}{1+\nu}r\cos\theta+\frac{2}{1+\nu}\frac{a^2}{r}\cos\theta+\frac{1}{2}\frac{a^2}{r}\cos 3\theta-\frac{1}{2}\frac{a^4}{r^3}\cos 3\theta\right) \quad (21a)$$

$$u_2=\frac{1+\nu}{E}\sigma\left(-\frac{\nu}{1+\nu}r\sin\theta-\frac{1-\nu}{1+\nu}\frac{a^2}{r}\sin\theta+\frac{1}{2}\frac{a^2}{r}\sin 3\theta-\frac{1}{2}\frac{a^4}{r^3}\sin 3\theta\right) \quad (21b)$$

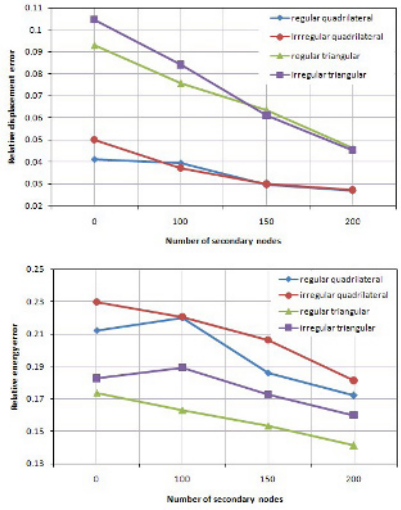


Fig. 13. Relative errors of displacement and energy for a plate with a circular hole problem.

Due to symmetry, only a part, $0\le r\le 4$, of the upper half of the plate is modeled as shown in Fig. 10(a).

Symmetry conditions are imposed on the bottom edge, i.e., $u_2 = 0$, $t_1 = 0$ on the bottom edge, and the inner boundary at $a = 1.0$ is traction free. The traction boundary conditions, as given by the exact solutions, are imposed on the outer boundary at $r = 4$. First, ground meshes with regularly and irregularly shaped quadrilateral elements are considered. The ground mesh is constructed by connecting the primary nodes. Particularly, enrichment of solutions using an irregular ground mesh will show the possibility of the present method to solve a problem using generally shaped elements. We distribute 100, 150 and 200 secondary nodes randomly in the domain through generating random numbers. We use 25 integration points in an integration domain for obtaining an accurate numerical integration of the products of MLS shape functions. The stress distributions are plotted in Fig. 11. The stress changes to a better one as the secondary nodes are added in the domain. Note that the stresses are discontinuous across the boundaries of the ground mesh. Secondly, we solve the problem using ground meshes with regularly and irregularly shaped triangular elements. Fig. 12 show the stress distributions when 100, 150 and 200 secondary nodes are placed randomly in the domain. Similar to the results for the quadrilateral elements, better results can be obtained by adding secondary nodes in the domain. The stresses are not continuous across the boundaries of the ground mesh. The relative errors of displacement and energy decrease by adding secondary nodes in the domain, as shown in Fig. 13. In particular, secondary nodes added on irregularly shaped ground meshes improve the deformations and stresses in numerical results. The numerical results show a good performance even though secondary nodes are distributed randomly in the domain.

5.2 A center cracked plate in tension

Next, we consider a center cracked plate in tension. Due to symmetry, the right half as shown in Fig. 10b is modeled under plane stress condition. The size of model is $h = w = 4.0$, and the crack length is $a = 2.0$. The applied stress σ_0 at the top and the bottom is 5.0×10^8 in this example. The symmetric condition is applied on the left side. Of primary importance in a crack problem is the determination of the parameters which characterize the singularity of the stress fields in the vicinity of a crack tip. The mode I stress intensity factor K_I, as a characterizing parameter for the crack, is computed from the J-integral by domain integration [23]. The size of the J-integral domain is chosen as $2b_1 \times 2b_2 = 2.0 \times 2.0$. The stress intensity factor K_I is evaluated by $K_I = \sqrt{JE}$ for plane stress, and the target solution for this problem is $K_I / K_0 = 1.325$ where $K_0 = \sigma_0\sqrt{\pi a}$ [24]. In this numerical example, we use quadrilateral and triangular ground meshes, and 25, 50 and 100 secondary nodes are added randomly near the crack tip. The J-integral domain is indicated in Fig. 10b. The stress σ_{22} at the crack tip is higher as the number of secondary nodes near the crack tip increases. By adding secondary nodes on the crack lines, improved deformations near the crack tip are obtained as shown in Figs. 14 and 15. In particular, the crack line is not straight when the secondary nodes are involved in the deformations near the crack tip. In Fig. 16, the errors in the stress intensity factors evaluated from J-integral are plotted against the number of secondary nodes.

The stress intensity factor approaches the target solution as the number of secondary nodes in the domain increases. Consequently, a better solution can be obtained by adding secondary nodes in the domain of interest.

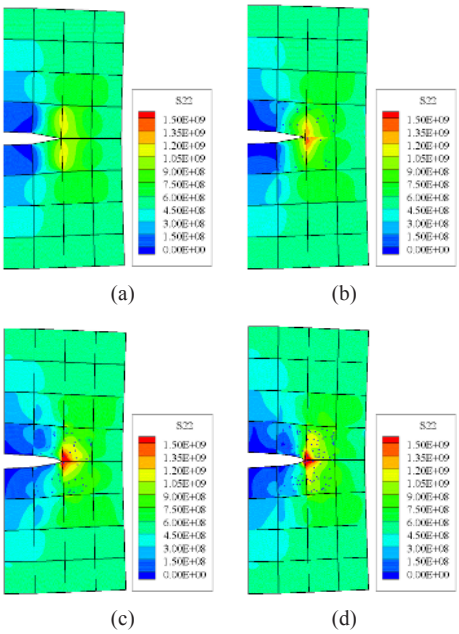


Fig. 14. Stress σ_{22} in a center cracked plate using a ground mesh with quadrilateral elements: (a) 0 secondary nodes, (b) 25 secondary nodes, (c) 50 secondary nodes, (d) 100 secondary nodes.

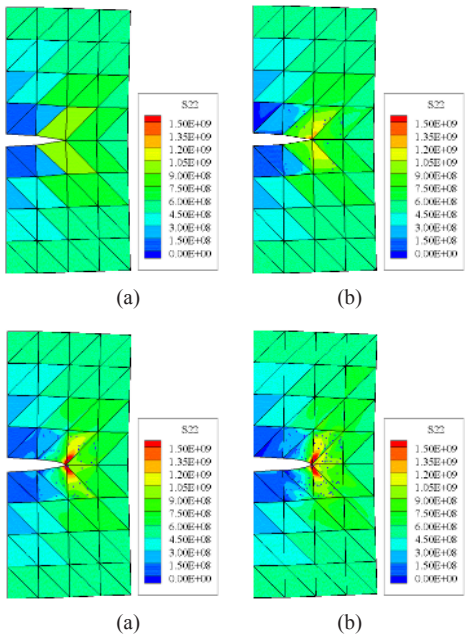


Fig. 15. Stress σ_{22} in a center cracked plate using a ground mesh with triangular elements: (a) 0 secondary nodes, (b) 25 secondary nodes, (c) 50 secondary nodes, (d) 100 secondary nodes.

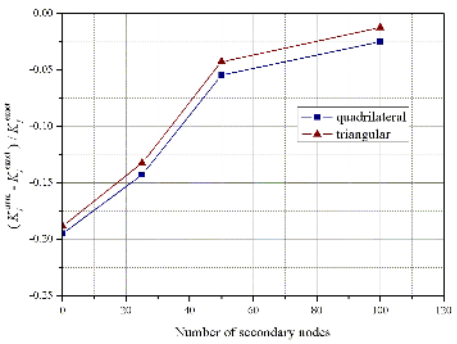


Fig. 16. Errors in the evaluation of stress intensity factor, versus the number of secondary nodes for a center cracked plate problem.

6. Concluding remarks

A new adaptive method has been presented to enrich the finite element solutions with the meshless concept. Arbitrary placement of secondary nodes makes the method in many respects closer to meshless methods. We present a new method to define the weight functions of primary and secondary nodes when polygonal supports are convex and concave domains. Consequently, an important aspect of the present scheme is the capability for solving problems using ground meshes with generally shaped triangular and quadrilateral elements, which allows an efficient adaptive calculation by adding secondary nodes in a domain. The approach presented here alleviates a major difficulty in meshless connectivity, complex intersections of support domains, and the definition of integration domains. Moreover, the present approach can directly take care of essential boundary conditions. A clear advantage of the adaptive scheme in this paper is that secondary nodes can be placed arbitrarily in the domain of interest, without the burdensome task of constructing a new mesh to enrich the solution. Therefore, the present method can be a useful tool for error controls and adaptive calculations in the field of computational mechanics.

Acknowledgment

This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD) (KRF-2007-331-D00013).

References

[1] T. Belytschko, Y. Y. Lu and L. Gu, Element-free Galerkin methods, *International Journal for Numerical Methods in Engineering*, 37 (1994) 229-256.

[2] W. K. Liu, S. Jun and Y. F. Zhang, Reproducing kernel particle methods, *International Journal for Numerical Methods in Fluids*, 20 (1995) 1081-1106.

[3] I. Babuska and J. M. Melenk, The partition of unity method, *International Journal for Numerical Methods in Engineering*, 40 (1997) 727-758.

[4] C. A. Duarte and J. T. Oden, Hp-cloud–a meshless method to solve boundary-value problems, *Computer Methods in Applied Mechanics and Engineering*, 139 (1996) 237-262.

[5] S. N. Atluri and T. Zhu, A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics, *Computational Mechanics*, 22 (1998) 117-127.

[6] S. R. Idelsohn and E. Onate, To mesh or not to mesh. That is the question…, *Computer Methods in Applied Mechanics and Engineering*, 195 (2006) 4681-4696.

[7] N. Sukumar, B. Moran, A. Y. Semenov and V. Belikov, Natural neighbour Galerkin methods, *International Journal for Numerical Methods in Engineering*, 50 (2001) 1-27.

[8] S. R. Idelsohn, E. Onate, N. Calvo and F. D. Pin, The meshless finite element method, *International Journal for Numerical Methods in Engineering*, 58 (2003) 893-912.

[9] G. R. Liu and Z. H. Tu, An adaptive procedure based on background cells for meshless methods, *Computer Methods in Applied Mechanics and Engineering*, 191 (2002) 1923-1943.

[10] J. T. Oden, C. A. Duarte and O. C. Zienkiewicz, A new cloud-based hp finite element method, *Computer Methods in Applied Mechanics and Engineering*, 153 (1998) 117-126.

[11] T. Strouboulis, K. Copps and I. Babuska, The generalized finite element method, *Computer Methods in Applied Mechanics and Engineering*, 190 (2001) 4081-4193.

[12] C. A. Duarte, D. J. Kim and D. M. Quaresma, Arbitrarily smooth generalized finite element approximations, *Computer Methods in Applied Mechanics and Engineering*, 196 (2006) 33-56.

[13] S. Hao, W. K. Liu and T. Belytschko, Moving particle finite element method with global smoothness, *International Journal for Numerical Methods in Engineering*, 59 (2004) 1007-1020.

[14] H. Lu, S. Li, D. C. Simkins, W. K. Liu and J. Cao, Reproducing kernel element method Part III: Generalized enrichment and applications, *Computer Methods in Applied Mechanics and Engineering*, 193 (2004) 989-1011.

[15] W. K. Liu, R. A. Uras and Y. Chen, Enrichment of the finite element method with the reproducing kernel particle method, *Journal of Applied Mechanics ASME*, 64 (1998) 861-870.

[16] A. Huerta and S. Fernández-Méndez, Enrichment and coupling of the finite element and meshless methods, *International Journal for Numerical Methods in Engineering*, 48 (2000) 1615-1636.

[17] H.-G. Kim and S. N. Atluri, Arbitrary placement of secondary nodes, and error control, in the mesh less local Petrov-Galerkin (MLPG) method, *Computer Modeling in Engineering and Sciences*, 1 (3) (2000) 11-32.

[18] S. N. Atluri, H.-G. Kim and J. Y. Cho, A critical assessment of the truly meshless local Petrov-Galerkin (MLPG), and local boundary integral equation (LBIE) methods, *Computational Mechanics*, 24 (1999) 348-372.

[19] E. A. Malsch and G. Dasgupta, Shape functions for polygonal domains with interior nodes, *International Journal for Numerical Methods in Engineering*, 61 (2004) 1153-1172.

[20] T. Belytschko, D. Organ and Y. Krongauz, A coupled finite element–element-free Galerkin method, *Computational Mechanics*, 17 (1995) 186-195.

[21] J. Dolbow and T. Belytschko, Numerical integration of the Galerkin weak form in meshless methods, *Computational Mechanics*, 23 (1999) 219-230.

[22] S. Wandzura and H. Xiao, Symmetric quadrature rules on a triangle, *Computational Applied Mathematics*, 45 (2003) 1829-1840.

[23] T. L. Anderson, *Fracture mechanics fundamentals and applications*, CRC Press (1991).

[24] X. R. Wu and A. J. Carlsson, *Weight functions and stress intensity factor solutions*, Pergamon Press (1991).

Hyun-Gyu Kim received his B.S. degree from Seoul National University in 1990. He then received M.S. and Ph.D. degrees from KAIST in 1993 and 1998, respectively. Dr. Kim is currently a Professor at the Department of Mechanical Engineering in Seoul National University of Technology, Korea. His research interests include multi-physics coupling analysis, interfacing non-matching meshes, development of special elements, and inverse problems.