GRANT/TR/IN/39

Final Technical Report

On the grant

"Development of MLPG and LBIE Methods for Nonlinear Problems of Fracture"

NASA Grant Number: NAG-1-2274

Principal Investigator:
Satya N. Atluri
7704 Boelter Hall, UCLA
Los Angeles, CA 90095-1600

Effective Date: 3/15/00
Expiration Date: 12/14/00

NASA Technical Officer:
Dr. Ivatury S. Raju
Mail Stop 240
NASA Langley Research Center
Hampton VA 23681-2199

Arbitrary Placement of Secondary Nodes, and Error Control, in the Meshless Local Petrov-Galerkin (MLPG) Method

H.-G. Kim, S. N. Atluri[1]

Abstract: The truly meshless local Petrov-Galerkin (MLPG) method holds a great promise in solving boundary value problems, using a local symmetric weak form as a natural approach. In the present paper, in the context of MLPG and the meshless interpolation of a moving least squares (MLS) type, a method which uses primary and secondary nodes in the domain and on the global boundary is introduced, in order to improve the accuracy of solution. The secondary nodes can be placed at any location where one needs to obtain a better resolution. The sub-domains for the shape functions in the MLS approximation are defined only from the primary nodes, and the secondary nodes use the same sub-domains. The shape functions based on the MLS approximation, in an integration domain, have a single type of a rational function, which reduces the difficulty of numerical integration to evaluate the weak form. The present method is very useful in an adaptive calculation, because the secondary nodes can be easily added and/or moved without an additional mesh. The essential boundary conditions can be imposed exactly, and non-convex boundaries can be treated without special techniques. Several numerical examples are presented to illustrate the performance of the present method.

keyword: meshless method, MLPG method, local symmetric weak form, MLS, primary node, secondary node.

1 Introduction

Meshless methods are attractive in adaptive error-control in computations to solve boundary value problems, by adding or removing nodes without the burdensome task of remeshing, each time. Several meshless methods have been proposed, each with certain advantages and disadvantages. These include: the smooth particle hydrodynamics (SPH) [Gingold and Monaghan (1977)], the element free Galerkin (EFG) method [Nayroles, Touzot, and Villon (1992)], the reproducing kernel particle method (RKPM) [Liu, Jun, and Zhang (1995)], the hp-cloud method [Duarte and Oden (1996)], the finite point method [Oñate, Idelsohn, Zienkiewicz, and Taylor (1996)], the partition of unity [Babuška and Melenk (1997)], the local boundary integral equation (LBIE) method [Zhu, Zhang, and Atluri (1998a,b)], and the meshless local Petrov-Galerkin (MLPG) method [Atluri and Zhu (1998a,b)]. In these methods, the construction of a trial approximation, which does not rely on element connectivity, is a significant development. However, most meshless methods, except the LBIE/MLPG methods, are not truly meshless approaches, since these methods require background meshes for numerical integration of the weak form. In these methods, an additional cost is associated with the construction of a background mesh, if nodes are added or deleted in a domain. The MLPG/LBIE methods, however, are more natural approaches, because these methods use a local weak form, and use numerical integration over sub-domains, which can be of arbitrary shapes such as circles, ellipses, rectangulars and parallelopipeds in a 2-dimensional geometry.

In spite of the novel concepts embodied in the MLPG method, difficulties in the numerical integration for evaluation of the weak form still persist, as reported by Atluri, Cho, and Kim (1999a) and Atluri, Kim, and Cho (1999b). This is due to the complexity of the non-element interpolation functions, which result from the moving least squares (MLS), the partition of unity, and the hp-cloud methods. In addition, circular subdomains make the numerical integration difficult, because the intersections between such sub-domains result in highly complex functions in the integration domain. As a result, a large number of integration points may be required to obtain accurate solutions [Atluri, Cho, and Kim (1999a) and Atluri, Kim, and Cho (1999b)]. In this paper, we present a viable method, based on the MLPG, that use secondary nodes to obtain an improvement in the accuracy of solution, without an additional mesh. The sub-domains for the MLS shape functions are generated only from the primary nodes, and the secondary nodes use the same sub-domains. The secondary nodes in the domain, and on the global boundary, do not necessitate the creation of new sub-domains, and the shape functions for the secondary nodes can be easily defined on the original sub-domains, using the MLS approximation. Numerical integration is carried out over a polygonal cell, which is the intersection of the sub-domains constructed only from the primary nodes. As a consequence of the alignment of the boundaries of sub-domains and integration domains, the shape functions have a single type of a rational function in a domain of integration. This greatly alleviates the difficulty in the numerical integration of the weak form in the MLPG method. The essential boundary conditions can be imposed exactly, and the non-convex boundaries can be treated without using

[1] Center for Aerospace Research and Education, 7704 Boelter Hall, University of California, Los Angeles, CA 90095-1600, USA.

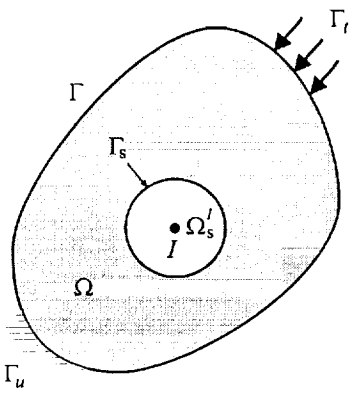


Figure 1 : A schematic representation of the sub-domain Ω_s^I, with node I as its center, and $\partial\Omega_s^I$ as its boundary. The global domain is Ω, with a global boundary Γ, where displacements are prescribed at Γ_u, and tractions are prescribed at Γ_t.

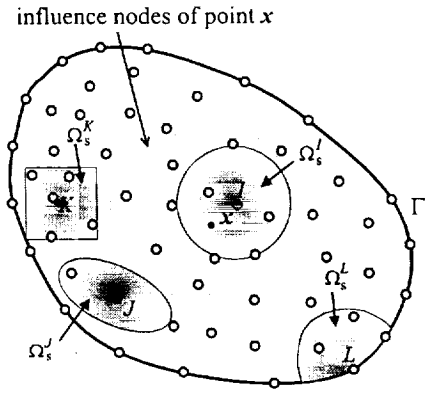


Figure 2 : A schematic illustrating various shapes for sub-domains, and the region bounding all the nodes in Ω which may influence the interpolation at a generic **x** in a meshless approximation.

any special techniques. The primary advantage of the present MLPG method is that the secondary nodes can be easily added and/or removed, without the burdensome task of constructing a new mesh, because the secondary nodes use the same sub-domains defined from the original primary nodes. The use of the secondary nodes to improve the computational solutions to a boundary value problem is independent of the mesh derived from the primary nodes. The present MLPG method offers a very useful tool for an adaptive calculation, by controlling errors in the computed results.

There have been several efforts to develop ways to improve the accuracy of a numerical solution, using a coarse primary mesh. Oden, Duarte, and Zienkiewicz (1998) 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 (GFEM) [Strouboulis, Babuska, and Copps (1998)] uses special functions from known analytical solutions in order to improve the FEM solution, in a way that is similar to the conventional hybrid FEM [Atluri, Gallagher, and Zienkiewicz (1983)]. Liu, Uras, and Chen (1998) used a coupling of the RKPM and the FEM to achieve an adaptive calculation by adding nodes. Although this coupling of the RKPM and the FEM has features that are similar to the present MLPG method, the basic approaches of two methods are quite different. The enrichment using the RKPM is not based on a local weak-formulation, and the coupling of the RKPM and the FEM may not give rise to consistent solutions, because of the difficulty of numerical integration over an integration domain when the boundaries of sub-domains and the integration domains are not aligned with each other.

The outline of this paper is as follows. In section 2, the local symmetric weak form is explained, as a key concept in the MLPG method. In section 3, we review the characteristics of the MLS approximation and of the numerical integration. In this part, we emphasis the difficulty of numerical integration to evaluate the weak form in meshless methods. In section 4, the concept of "primary" and "secondary" nodes is introduced. To construct proper shape functions for the primary and the secondary nodes in the domain and on the global boundary, weight functions in the MLS approximation should be defined appropriately over sub-domains, in order to preserve a single type of a rational function in each integration domain. Towards this end, we present a method to construct the weight functions in the MLS approximation, for the primary and the secondary nodes. Numerical examples in linear elasticity are presented in section 5. Finally, the conclusions are given in section 6.

2 The meshless local Petrov-Galerkin (MLPG) formulation

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

$$u_i = \bar{u}_i \quad \text{at} \quad \Gamma_u \tag{2}$$

$$\sigma_{ij} n_j = \bar{t}_i \quad \text{at} \quad \Gamma_t \tag{3}$$

where Γ_u and Γ_t are the global boundaries with prescribed displacements and tractions, respectively. In a conventional Galerkin finite element formulation, the global weak form is

used to solve the boundary value problem numerically. However, the MLPG method starts from a weak form over a local sub-domain, or a patch, Ω_s^I inside the global domain Ω as shown in Fig. 1. 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. We implicitly introduce the concept of "nodes" with "local domains". The sub-domain Ω_s^I is thus called the sub-domain of node I. The sub-domain Ω_s^I can be a circle, a rectangle, or an ellipse in two dimensions (or a sphere, a parallelopiped, or an ellipsoid in three dimensions) in the MLPG formulation, but it can be extended to any kinds of geometry as shown in Fig. 2.

A generalized local weak form of the equilibrium equation is written as

$$\int_{\Omega_s^I}(\sigma_{ij,j}+b_i)v_i\,d\Omega=0 \tag{4}$$

where v_i is the test function. Using the divergence theorem in Eq. 4, we obtain the following local weak-form:

$$\int_{\partial\Omega_s^I}\sigma_{ij}n_jv_i\,d\Gamma-\int_{\Omega_s^I}(\sigma_{ij}v_{i,j}+b_iv_i)\,d\Omega=0 \tag{5}$$

where n_j is the outward unit normal to the boundary $\partial\Omega_s^I$.

In the MLPG method, the trial and test functions can be chosen from different spaces. In particular, the test functions need not vanish on the boundary where the essential boundary conditions are specified. Also, in order to simplify Eq. 5, we deliberately select the test functions v_i such that they vanish over $\partial\Omega_s^I$, except when $\partial\Omega_s^I$ intersects with the global boundary Γ. This can be easily accomplished in the MLPG method by using the nodal-test-shape function whose value at the local boundary $\partial\Omega_s^I$ is zero, as long as $\partial\Omega_s^I$ does not intersect with Γ. Using these test functions in Eq. 5, we obtain the following Local Symmetric Weak Form (LSWF):

$$\int_{\Omega_s^I}\sigma_{ij}v_{i,j}\,d\Omega=\int_{\Gamma_{st}^I}\bar{t}_iv_i\,d\Gamma+\int_{\Omega_s^I}b_iv_i\,d\Omega \tag{6}$$

where $t_i=\sigma_{ij}n_j$, and Γ_{st}^I is the intersection of Γ_t and the boundary $\partial\Omega_s^I$ of Ω_s^I. The MLPG method based on the local formulation in Eq. 4 makes clear the basic concepts for integrating the local weak form in Eq. 6. The MLPG formulation enables us to use different interpolations for the trial and the test functions. Furthermore, the sizes and shapes of the sub-domains of trial and test functions, respectively, do not need to be the same in the MLPG formulation. Therefore, the MLPG method can include all other meshless methods as special cases. In the present method, we use the same function space for the trial and the test functions as a special case. Note that the value of the trial function at each point $\mathbf{x}$ inside Ω_s^I, is influenced by a set of values of the function at an arbitrary number of nodes in the vicinity of each $\mathbf{x}$, in a non-element, diffuse interpolation of the moving least square (MLS) type. Thus, Eq. 6 leads, for each Ω_s^I, to the I^{th} equation in the system stiffness matrix, involving all the J nodes, whose sub-domains Ω_s^J intersect with Ω_s^I, such that the integrand in Eq. 6 is non-zero.

To obtain the discrete equations from the MLPG formulation in Eq. 6, based on meshless interpolations, which are explained in the next section, the following interpolations are used. The global forms of interpolations for the trial and the test functions, respectively, can be written as

$$u_i^h(\mathbf{x})=\sum_{J=1}^{N}\phi^J(\mathbf{x})\hat{u}_i^J \tag{7}$$

$$v_i^h(\mathbf{x})=\sum_{I=1}^{N}\psi^I(\mathbf{x})\hat{v}_i^I \tag{8}$$

where $\phi^J(\mathbf{x})$ and $\psi^I(\mathbf{x})$ are the nodal shape functions for the trial and the test functions centered at nodes J and I, respectively. In general, in meshless interpolations, $\hat{v}_i^I$ and $\hat{u}_i^J$ are fictitious nodal values. Substitution of Eq. 6 into the MLPG formulation in Eq. 5 leads to following discretized system of linear equation:

$$\sum_{J=1}^{N}\int_{\Omega_s^I}(\mathbf{B}_v^I)^T\mathbf{D}\mathbf{B}^J\hat{\mathbf{u}}^J\,d\Omega=\int_{\Gamma_{st}^I}\mathbf{V}^I\bar{\mathbf{t}}\,d\Gamma+\int_{\Omega_s^I}\mathbf{V}^I\mathbf{b}\,d\Omega \tag{9}$$

where, in two-dimensional space,

$$\mathbf{B}_v^I=\begin{bmatrix}\psi_{,1}^I & 0\\ 0 & \psi_{,2}^I\\ \psi_{,2}^I & \psi_{,1}^I\end{bmatrix},\mathbf{B}^J=\begin{bmatrix}\phi_{,1}^J & 0\\ 0 & \phi_{,2}^J\\ \phi_{,2}^J & \phi_{,1}^J\end{bmatrix},\mathbf{N}=\begin{bmatrix}n_1 & 0 & n_2\\ 0 & n_2 & n_1\end{bmatrix}$$

$$\mathbf{V}^I=\begin{bmatrix}\psi^I & 0\\ 0 & \psi^I\end{bmatrix},\bar{\mathbf{u}}^J=\begin{Bmatrix}\hat{u}_1^J\\ \hat{u}_2^J\end{Bmatrix}$$

$$\mathbf{D}=\frac{\bar{E}}{1-\bar{\nu}^2}\begin{bmatrix}1 & \bar{\nu} & 0\\ \bar{\nu} & 1 & 0\\ 0 & 0 & (1-\bar{\nu})/2\end{bmatrix}$$

with

$$\bar{E}=\begin{cases}E & \\ \frac{E}{1-\nu^2} & \end{cases}\quad\text{and}\quad\bar{\nu}=\begin{cases}\nu & \text{for plane stress}\\ \frac{\nu}{1-\nu} & \text{for plane strain}\end{cases}$$

In the above equations, (n_1,n_2) is the normal vector at the boundary, and E and ν are the Young modulus and Poisson's ratio, respectively. The local symmetric weak form in Eq. 9 makes the "stiffness" entries, $\mathbf{K}_{IJ}$ (which is the stiffness matrix in multi-dimensional space), in the row corresponding to the node I, and to the nodes J, depending only on the non-zero values of the integrands in the weak form, over the intersection of Ω_s^I and Ω_s^J. Then, the global equation can be written as

$$\sum_{J=1}^{N}\mathbf{K}_{IJ}\hat{\mathbf{u}}^J=\mathbf{f}_I \tag{10}$$

where

$$\mathbf{K}_{IJ}=\int_{\Omega_s^I}(\mathbf{B}_v^I)^T\mathbf{D}\mathbf{B}^J\,d\Omega$$

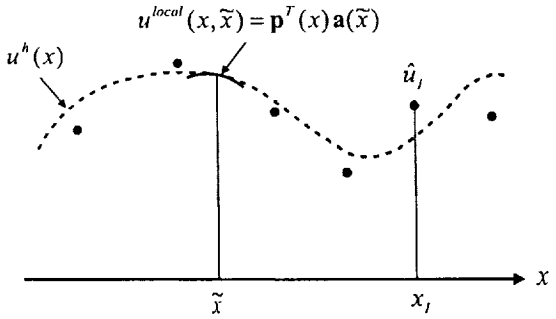


Figure 3 : Conceptual explanation of the moving least square interpolation, in one-dimension.

$$\mathbf{f}_I = \int_{\Gamma_{st}^I} \mathbf{V}^I \bar{\mathbf{t}} d\Gamma + \int_{\Omega_s^I} \mathbf{V}^I \mathbf{b} d\Omega$$

Therefore, in the MLPG method, the usual assembly process is not required to form a global stiffness matrix. Theoretically, as long as the union of all local domains covers the global domain, the equilibrium equation and the boundary conditions will be satisfied in the global domain Ω and its boundary Γ, respectively.

3 Characteristics of moving least square (MLS) approximation & numerical integration

In this section, we review the characteristics of the moving least square (MLS) approximation, including some difficulties in the numerical integration of the weak form to evaluate the stiffness matrix, which was well explained in Atluri, Cho, and Kim (1999a) and Atluri, Kim, and Cho (1999b). The MLS method is generally considered to be one of the schemes to interpolate random data with a reasonable accuracy. Nayroles, Touzot, and Villon (1992) were the first to use the MLS interpolation in a Galerkin formulation, which they called the "diffuse element method". The MLS approximation always preserves completeness up to the order of the basis, and reasonably interpolates the irregularly distributed nodal information. However, the nodal shape functions that arise from the MLS approximation have a very complex nature. They are not only rational, but they are also of different types across the boundaries of sub-domains. This complexity results in difficulties with the numerical integration of the weak form in the MLPG method.

We consider the approximation of a function $u(\mathbf{x})$ in a local region centered at $\bar{\mathbf{x}}$ in a domain Ω as shown in Fig. 3.

The moving least-square approximation starts from a local approximation in the neighborhood of $\bar{\mathbf{x}}$, such as

$$u^{local}(\mathbf{x},\bar{\mathbf{x}}) = \mathbf{p}^T(\mathbf{x})\mathbf{a}(\bar{\mathbf{x}}) \quad \forall \mathbf{x} \in B(\bar{\mathbf{x}}) \tag{11}$$

where, $B(\bar{\mathbf{x}})$ is a sphere centered at $\bar{\mathbf{x}}$, $\mathbf{p}^T(\mathbf{x}) = [p_1(\mathbf{x}), p_2(\mathbf{x}), \ldots, p_m(\mathbf{x})]$ is a complete monomial basis of order m; and $\mathbf{a}(\bar{\mathbf{x}})$ is a vector containing the coefficients $a_j(\bar{\mathbf{x}})$, $j = 1,2,\ldots,m$ which are functions of the space coordinates $\mathbf{x} = [x,y,z]^T$. The commonly used bases in 2-D problems are the linear basis:

$$\mathbf{p}^T(\mathbf{x}) = [1,x,y] \tag{12}$$

or the quadratic basis:

$$\mathbf{p}^T(\mathbf{x}) = [1,x,y,x^2,xy,y^2] \tag{13}$$

The coefficient vector $\mathbf{a}(\bar{\mathbf{x}})$ is determined by minimizing a weighted discrete L_2-norm, defined as:

$$\begin{aligned}\mathbf{Y}(\bar{\mathbf{x}}) &= \sum_{I=1}^{N} w_I(\bar{\mathbf{x}})[\mathbf{p}^T(\mathbf{x}_I)\mathbf{a}(\bar{\mathbf{x}}) - \hat{u}^I]^2 \\ &= [\mathbf{P}\cdot\mathbf{a}(\bar{\mathbf{x}}) - \hat{\mathbf{u}}]^T \cdot \mathbf{W}(\bar{\mathbf{x}}) \cdot [\mathbf{P}\cdot\mathbf{a}(\bar{\mathbf{x}}) - \hat{\mathbf{u}}]\end{aligned} \tag{14}$$

where $w_I(\bar{\mathbf{x}})$ is a weight function defined in a sub-domain Ω_s^I, with the node I as its center; and with $w_I(\bar{\mathbf{x}}) > 0$ for all $\bar{\mathbf{x}}$ in the support of $w_I(\bar{\mathbf{x}})$ and $w_I(\bar{\mathbf{x}}) = 0$ at the boundary of Ω_s^I, $\mathbf{x}_I$ denotes the value of $\mathbf{x}$ at node I, and the matrices $\mathbf{P}$ and $\mathbf{W}$ are defined as

$$\mathbf{P} = \begin{bmatrix} \mathbf{p}^T(\mathbf{x}_1) \\ \mathbf{p}^T(\mathbf{x}_2) \\ \cdots \\ \mathbf{p}^T(\mathbf{x}_N) \end{bmatrix}_{N\times m} \tag{15}$$

$$\mathbf{W} = \begin{bmatrix} w_1(\bar{\mathbf{x}}) & \cdots & 0 \\ \cdots & \cdots & \cdots \\ 0 & \cdots & w_N(\mathbf{x}) \end{bmatrix}_{N\times N} \tag{16}$$

and

$$\hat{\mathbf{u}}^T = [\hat{u}^1, \hat{u}^2, \cdots, \hat{u}^N] \tag{17}$$

Here it should be noted that $\hat{u}^I, I = 1,2,\cdots,N$ are the fictitious nodal values. In fact, only those neighboring nodes J, whose sub-domains Ω_s^J intersect with the sub-domain Ω_s^I of node I, have an influence in constructing the shape function for node I. For convenience, $\bar{\mathbf{x}}$ in the above relations is replaced by $\mathbf{x}$, because a local point $\bar{\mathbf{x}}$ 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 conceptual explanation for the MLS is given in Fig. 3.

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

$$\mathbf{A}(\mathbf{x})\mathbf{a}(\mathbf{x}) = \mathbf{B}(\mathbf{x})\hat{\mathbf{u}} \tag{18}$$

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

$$\mathbf{A}(\mathbf{x}) = \mathbf{P}^T\mathbf{W}\mathbf{P} = \sum_{I=1}^{N} w_I(\mathbf{x})\mathbf{p}(\mathbf{x}_i)\mathbf{p}^T(\mathbf{x}_i) \tag{19}$$

$$\mathbf{B}(\mathbf{x}) = \mathbf{P}^T\mathbf{W} = [w_1(\mathbf{x})\mathbf{p}(\mathbf{x}_1), w_2(\mathbf{x})\mathbf{p}(\mathbf{x}_2), \ldots, w_N(\mathbf{x})\mathbf{p}(\mathbf{x}_N)] \tag{20}$$

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

$$u^h(\mathbf{x}) = \sum_{I=1}^{N} \phi^I(\mathbf{x})\hat{u}^I \tag{21}$$

where the nodal shape functions are given by

$$\Phi^T(\mathbf{x}) = \mathbf{p}^T(\mathbf{x})\mathbf{A}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x}) \tag{22}$$

In the traditional Galerkin FEM, the 'nodal shape functions' have a value of unity at the respective node, and an approximation of the type of Eq. 21 would involve the directly the 'nodal value' of the field variable. However, in the present MLS approximation, $\hat{u}^I$ are fictitious, and are not exactly equal to the nodal values of the field variable (see Fig. 3). Inspite of this, it is instructive to call $\phi^I(\mathbf{x})$ in Eq. 21 'a nodal shape function'. The MLS interpolation is well defined only when the matrix $\mathbf{A}$ is non-singular. A necessary condition for a well-defined MLS interpolation is that at least m weight functions are nonzero (i.e. $N \geq m$) for each sample point $\mathbf{x} \in \Omega$. The partial derivative of $\phi^I(\mathbf{x})$ can be obtained as follows:

$$\phi^I_{,k} = \sum_{j=1}^{m} [p_{j,k}(\mathbf{A}^{-1}\mathbf{B})_{jI} + p_j(\mathbf{A}^{-1}\mathbf{B}_{,k} + \mathbf{A}^{-1}_{,k}\mathbf{B})_{jI}] \tag{23}$$

in which $\mathbf{A}^{-1}_{,k} = (\mathbf{A}^{-1})_{,k}$ represents the derivative of the inverse of $\mathbf{A}$ with respect to $\mathbf{x}$, which is given by

$$\mathbf{A}^{-1}_{,k} = -\mathbf{A}^{-1}\mathbf{A}_{,k}\mathbf{A}^{-1} \tag{24}$$

and the index following a comma indicates a spatial derivative. Considering that $\phi^I(\mathbf{x}) = 0$ whenever $w_I(\mathbf{x}) = 0$, the support sizes for the nodal shape function and the weight function have the same value. The nodal shape functions obtained by the MLS interpolation with mth order basis can reproduce any mth order polynomials $g(\mathbf{x})$ exactly [Belytschko, Krongauz, Organ, Fleming, and Krysl (1996)], i.e.,

$$\sum_{I=1}^{N} = \phi^I(\mathbf{x})g(\mathbf{x}_I) = g(\mathbf{x}) \tag{25}$$

Eq. 25 indicates that the nodal shape function is complete up to the order of the basis. In order to guarantee the convergence of the weak formulation with successive increase in the number of nodes, the shape functions have to be complete. The smoothness of the nodal shape function $\phi^I(\mathbf{x})$ is determined by those of the basis and of the weight function. The choice of the weight function is more or less arbitrary as long as the weight function is positive and continuous.

We can obtain an explicit form for the nodal shape functions, with a linear basis, in a two-dimensional problem, in order to better understand the characteristics of the nodal shape functions. The nodal shape functions can be written as

$$\phi^I(x,y) = \frac{w_I(x,y)}{b(x,y)}[c_0(x,y), c_1(x,y), c_2(x,y)] \begin{Bmatrix} 1 \\ x \\ y \end{Bmatrix} \tag{26}$$

where the coefficients $c_0(x,y)$, $c_1(x,y)$, $c_2(x,y)$, and $b(x,y)$ are given in Atluri, Kim, and Cho (1999). In general, $c_0(x,y)$, $c_1(x,y)$, $c_2(x,y)$, and $b(x,y)$ are not of the single type of functions, because the sub-domains related to the sub-domain Ω^I_s make complex intersections as shown in Fig. 4. For example, the weight functions in the expressions for the coefficients $c_0(x,y)$, $c_1(x,y)$, $c_2(x,y)$, and $b(x,y)$ are different at $\mathbf{x}_1$ and $\mathbf{x}_2$ in Fig. 4. In other words, the nodal shape function $\phi^I(\mathbf{x}$ consists of a different form of a rational function in each small intersection region, as indicated in Fig. 4.

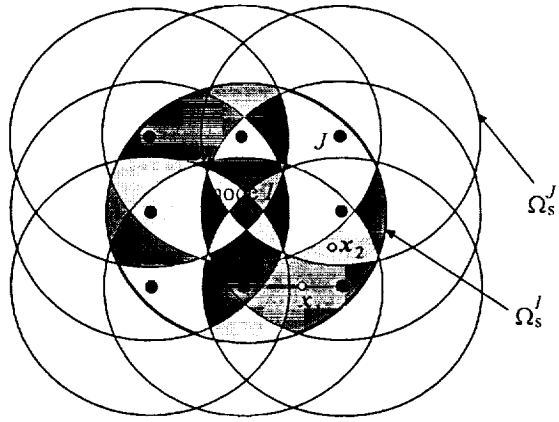


Figure 4 : The nodal shape function $\phi^I(\mathbf{x})$, from the MLS interpolation, which is nonzero in Ω^I_s, and which has a different form in each small intersection, as divided by several circular sub-domains.

Although the smoothness of $\phi^I(\mathbf{x})$ can be achieved if a sufficient order of spline function is used as a weight function, the shape function in the sub-domain Ω^I_s consists of several types of rational functions. It seems to be difficult to integrate these kinds of complex functions in the sub-domain, by using a simple Gaussian quadrature rule, and this causes a difficulty in the numerical integration of weak forms. As a result, an accurate integration of the shape functions for the construction of the stiffness matrix is not as trivial as for the finite element method. Due to different characteristics of functions in different sub-regions, we cannot expect accurate numerical integration even though many numbers of integration points are introduced. Two typical integration methods in meshless methods are illustrated in Fig 5. Although integration over sub-domain Ω^I_s (Fig. 5b) is more natural than the integration using a background cell (Fig. 5a), the difficulty of numerical integration may not be avoided in a simple way.

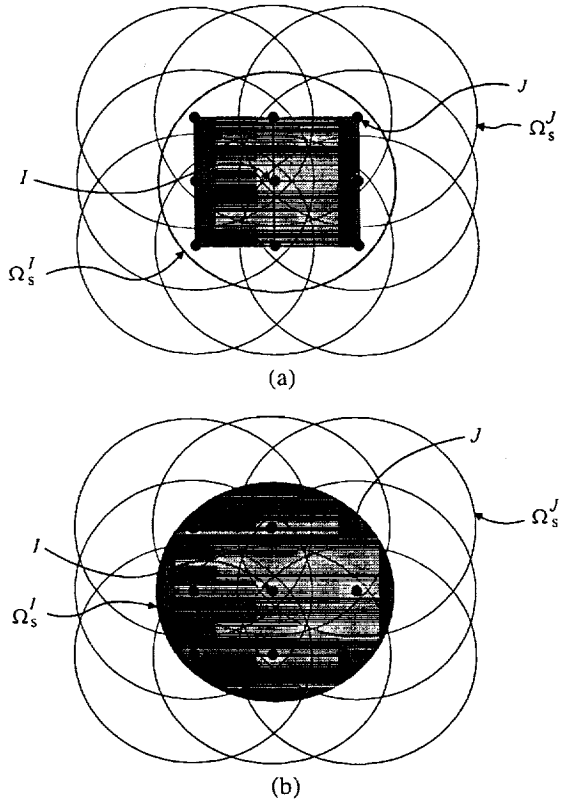


Figure 5 : Integration methods in meshless methods: (a) using background cell and (b) integration over sub-domain Ω_s^I for node I.

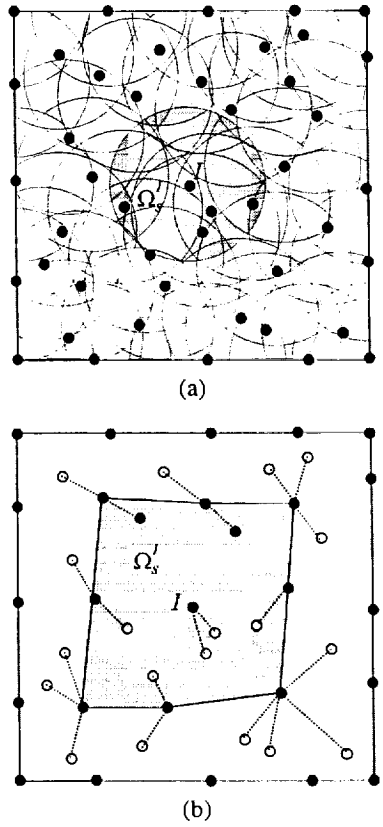


Figure 6 : Schematic representations for sub-domains with randomly distributed nodes: (a) for the MLPG method with circular sub-domains and (b) for the MLPG method with polygonal sub-domains; in (b), the sub-domain for the secondary nodes is taken to be the same as that for the nearest primary node in the present method (The solid circles are primary nodes; and the open circles are secondary nodes.

To surmount this difficulty, in the present version of the MLPG approach, polygonal sub-domains are employed so as to lead to a single type of the MLS shape function all over the integration domain. The detailed explanation on how to construct the MLS shape functions, in the polygonal sub-domain Ω_s^I, is given in the following section.

4 Error control in the MLPG method

This section first introduces the concept of primary and secondary nodes, and describes the way to construct polygonal sub-domains Ω_s^I, and the corresponding weight functions, for the present version of the MLPG method. Later, the advantages of the present approach in the treatment of essential boundary conditions, and non-convex boundaries, are pointed out. Next, the simplified procedures for the numerical evaluation of the stiffness matrix is described.

4.1 Polygonal sub-domains and weight functions for primary and secondary nodes

As explained in the previous section, the difficulty of numerical integration of the weak form is due to the complexity of the MLS shape functions in an integration domain. To evaluate the stiffness matrix with accuracy, it is necessary to make the shape functions simple in an integration domain. In our approach, we introduce the concept of primary and secondary nodes in the MLPG method, in order to prevent the crossing of the boundaries of sub-domains in an integration domain, such that the polygonal sub-domains defined by a mesh from only the primary nodes also become the sub-domains for the secondary nodes. As a result, no additional mesh is required for the secondary nodes, which makes it possible to extend the original MLPG concept to be a useful tool for error control and adaptive calculation. The shape functions in a sub-domain in the present version of the MLPG method may have a simpler form than those in other meshless methods, because of the alignment of the boundaries of sub-domains. Fig. 6 shows two types of sub-domains in the MLPG method: circular sub-domains and polygonal sub-domains that are used by both the

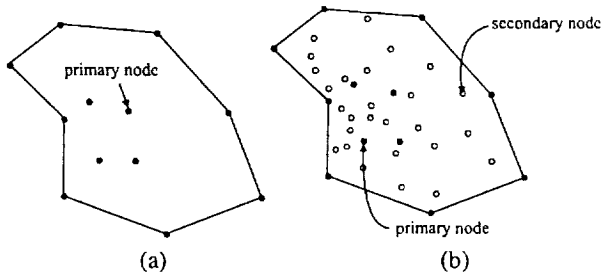


Figure 7 : Arbitrary secondary-node placement in the MLPG: (a) primary nodes, and (b) primary and secondary nodes in the domain. Primary nodes are solid circles, and secondary nodes are open circles.

primary and the secondary nodes. Since the types of the shape functions may change across the boundaries of sub-domains, the first method shown in Fig. 6a may lead to great difficulties in integrating the weak form accurately, for randomly distributed nodes, due to complex intersections of sub-domains. However, the present method in Fig. 6b uses simple polygonal intersections of sub-domains, in order to avoid this difficulty. As a result, the shape functions in a polygonal intersection have a single type of a rational function, because of the alignment of the boundaries of sub-domains. The advantage of the second method, in Fig. 6b, is that the secondary nodes can be placed at any arbitrary locations, where it is needed to improve the accuracy of the solution, as shown in Fig. 7. This second method, which is attractive for adaptive error-control algorithm, is pursued in the present paper. Fig. 8 illustrates the comparison between finite element mesh/background mesh that is commonly used in other meshless methods such as EFG, RKPM and h-p cloud methods; and a background mesh of the primary nodes only as in the present method. It is important to note that in the present MLPG method, the secondary nodes do not involve an additional mesh. In addition, the secondary nodes can be added and/or moved without changing a coarse background mesh constructed from the primary nodes. Fig. 8a shows a complex and a distorted feature for randomly distributed nodes in the FEM/EFG/RKPM/h-p cloud methods, and it is to be noted that a new mesh is required to change nodal information.

To meet the stated goal, it is important to develop simple forms of the shape functions for the primary and the secondary nodes. The weight functions in the MLS approximation, which ultimately govern the shape functions (see Eqs. 19, 20, and 22), should be defined appropriately to generate shape functions with simple forms. In the present approach, the secondary nodes can be placed at arbitrary locations in the domain Ω and on the global boundary Γ, in order to improve the deformations in each region, respectively. First, the secondary nodes which may be placed randomly, in the domain

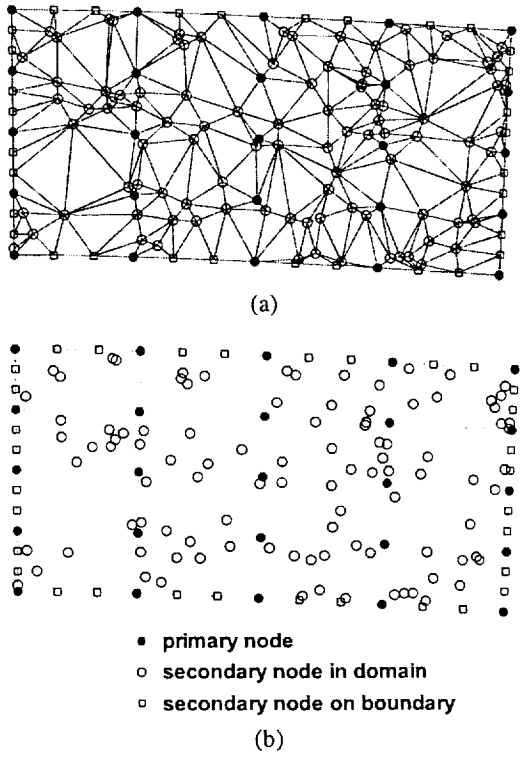


Figure 8 : Comparison between the FEM mesh/background mesh, and the MLPG: (a) the mesh from the usual Finite Element Method as well as the background mesh in EFG, RKPM, and h-p cloud method, when primary and secondary nodes are used, and (b) the mesh from primary nodes in the MLPG approach, whereas the secondary nodes in the present method do not involve an additional mesh.

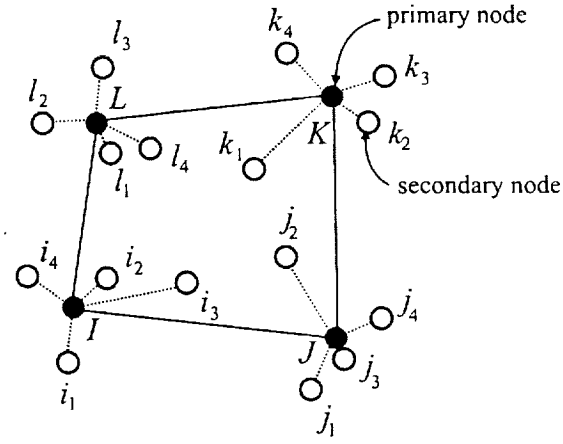


Figure 9 : A primary-node anchor for a secondary node is the nearest primary node. I, J, K and L indicate the primary nodes, and i, j, k and l indicate the set of secondary nodes.

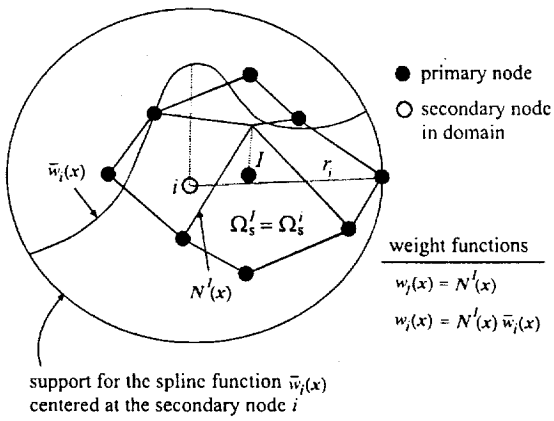


Figure 10 : The definition of sub-domains for the primary node I and the secondary node i in the domain Ω. r_i is the radius of support for spline function to construct weight function for the secondary node i in the domain.

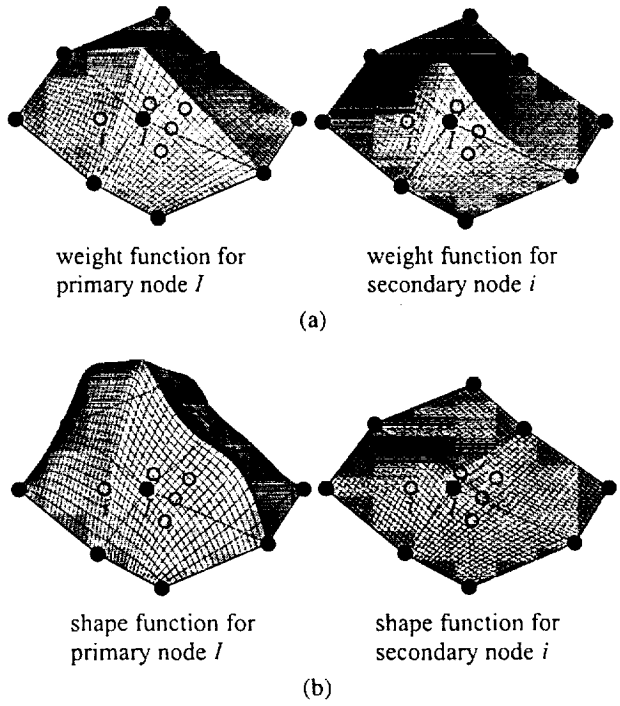


Figure 11 : Weight and shape functions: (a) weight functions and (b) shape functions for the primary node I and for the secondary node i, respectively in the domain. The weight function for the primary node is the finite element shape function, and the weight function for the secondary node has a skewed form which is obtained by multiplying the finite element shape function centered at I, and the 4^{th} order spline function centered at i.

Ω, take the sub-domains to be the supports of the nearest primary nodes, as shown in Fig. 6. The primary nodes on the global boundary Γ are excluded from the set of candidate primary nodes that act as anchors for the secondary nodes in the domain Ω, so that the essential boundary conditions can be imposed directly on the primary nodes on the global boundary Γ. Fig. 9 indicates how to choose the primary nodes that would be anchors for the secondary nodes in the domain, in a simple way. Therefore, it is very easy to find the sub-domains for the secondary nodes in the domain, with an initial coarse background mesh, constructed only from the primary nodes. We denote by I the primary node and by i the set of the secondary nodes, related to the primary node I, as shown in Fig. 10.

There are several methods to construct the weight functions on polygonal sub-domains. In the present paper, we use the linear finite element shape functions as weight functions in the MLS approximation for primary nodes, i.e.,

$$w_I(\mathbf{x}) = N^I(\mathbf{x}) \quad \text{on} \quad \Omega_s^I \tag{27}$$

where $N^I(\mathbf{x})$ is the linear finite element shape function for the primary node I. Note that the weight function inside a cell all over the sub-domain Ω_s^I has only one type of a simple form. We now discuss the construction of the weight functions in the MLS approximation for the secondary nodes. The weight functions for the secondary nodes should be zero at the boundaries of polygonal sub-domains as defined for the original primary nodes, which is the fundamental requirement in the MLPG formulation. For this purpose, we choose the weight functions in the MLS approximation, for the secondary nodes, to be the product of the linear finite element shape function $N^I(\mathbf{x})$ centered at a primary node I as in Eq. 27, and the 4^{th} order spline function $\bar{w}_i(\mathbf{x})$ with a circular support centered at the secondary node i, as indicated in Fig. 10. Within this circular support centered at the secondary node i, the maximum value of the spline function is at the position of the secondary node i, and the size of the circular support is set by the maximum distance from the secondary node i to the primary nodes in the sub-domain Ω_s^I. We take the weight function for the secondary node i to be

$$w_i(\mathbf{x}) = N^I(\mathbf{x})\bar{w}_i(\mathbf{x}) \quad \text{on} \quad \Omega_s^I = \Omega_s^i \tag{28}$$

with

$$\bar{w}_i(x,y) = \begin{cases} 1 - 6(\frac{d_i}{r_i})^2 + 8(\frac{d_i}{r_i})^3 - 3(\frac{d_i}{r_i})^4, & 0 \le d_i \le r_i \\ 0, & d_i > r_i \end{cases} \tag{29}$$

where I and i are the primary and the secondary nodes in the domain Ω, respectively, $d_i = |\mathbf{x} - \mathbf{x}_i|$ is the distance from the secondary node $\mathbf{x}_i$, and the radius r_i is the size of the support chosen in such a way that the circular support for $\bar{w}_i(\mathbf{x})$ covers the polygonal sub-domain $\Omega_s^I = \Omega_s^i$. Note that the values of weight functions in Eq. 28 for the secondary nodes are zero

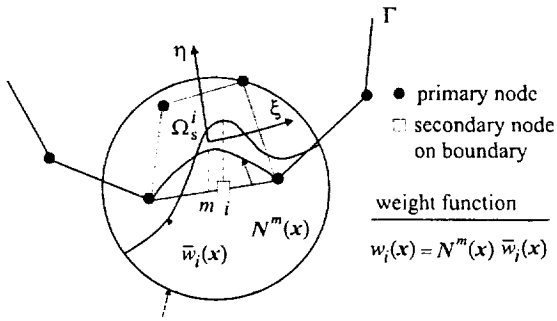


Figure 12 : The definition of the sub-domains for the primary node I and the secondary node i on the global boundary Γ. r_i is the radius of support for spline function to construct the weight function for the secondary node i.

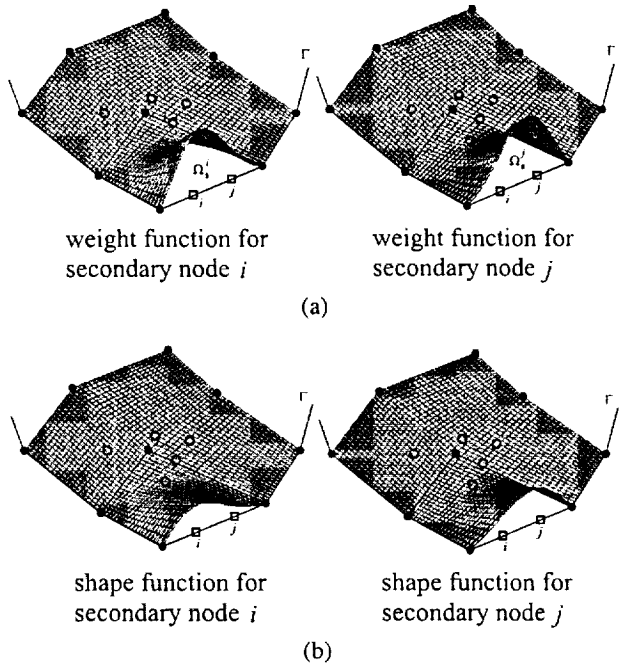


Figure 13 : (a) weight functions and (b) shape functions for secondary nodes i and j on the global boundary Γ.

at the boundary of the sub-domain $\Omega_s^I = \Omega_s^i$. The weight and the shape functions for the primary and the secondary nodes in the domain are plotted in Fig. 11. In these figures, the weight function for the secondary node i has a skewed form, depending on the location of $\mathbf{x}_i$. The construction of the shape functions and derivatives for the secondary nodes is identical to the procedure used for the primary nodes. The weight functions for secondary nodes in the domain satisfy the following conditions:

$w_i(\mathbf{x}) > 0$ on the sub-domain $\Omega_s^I = \Omega_s^i$,

$w_i(\mathbf{x}) = 0$ at the boundary of the sub-domain $\Omega_s^I = \Omega_s^i$,

$w_i(\mathbf{x})$ has a simple form of a continuous function inside a cell all over the sub-domain $\Omega_s^I = \Omega_s^i$.

The deformations on the global boundary cannot be improved, by adding secondary nodes only in the domain, because the primary nodes on the global boundary are excluded as candidate primary nodes for the secondary nodes in the domain. This limitation may spoil the quality of numerical solutions, due to the limitation of only a linear deformation between the primary nodes on the global boundary. However, improved deformations on the global boundary are important, in order to obtain the stresses or the strains correctly on the global boundary. Also, the convergence of energy may not be attained, without an improvement of the deformations on the global boundary, even though the convergence of displacements may be obtained using only the secondary nodes in the domain. Hence, in order to improve the deformations on the global boundary, some secondary nodes are placed also on the boundary-segments connecting the primary nodes on the global boundary. The sub-domains for the secondary nodes on the global boundary are the cells connecting primary nodes on the global boundary, as shown in Fig. 12. The weight functions for the secondary nodes on the global boundary can be constructed as

$$w_i(\mathbf{x}) = N^m(\mathbf{x})\bar{w}_i(\mathbf{x}) \quad \text{in} \quad \Omega_s^i \tag{30}$$

where $\bar{w}_i(\mathbf{x})$ is the 4^{th} order spline function in Eq. 29, where r_i is the maximum distance from the secondary node i to the primary nodes in the sub-domain Ω_s^i, and $N^m(\mathbf{x})$ in the present method is taken by

$$N^m(\mathbf{x}) = \tilde{N}^m(\xi,\eta) = \frac{1}{2}(1-\xi^2)(1-\eta) \tag{31}$$

In the above equation, the ξ and η are the coordinates defined on the master domain, as in the finite element method. The weight and the shape functions for the secondary nodes on the global boundary are plotted in Fig. 13. In these figures, the secondary nodes i and j on the global boundary between two primary nodes have the same sub-domain Ω_s^i and Ω_s^j. Consequently, the linear deformations on the global boundary can be improved by adding the secondary node on the global boundary.

To alleviate the difficulty in the numerical integration of the weak form, it is important to preserve a single type of a continuous function in an integration domain. Since the MLS shape functions, which are the rational functions as in Eq. 26, are

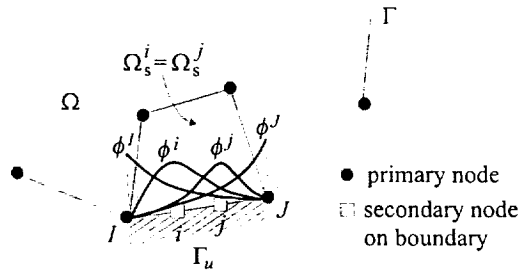


Figure 14 : Schematic representation of the shape functions for the secondary nodes i and j, which are located between the primary node I and J on the global boundary Γ.

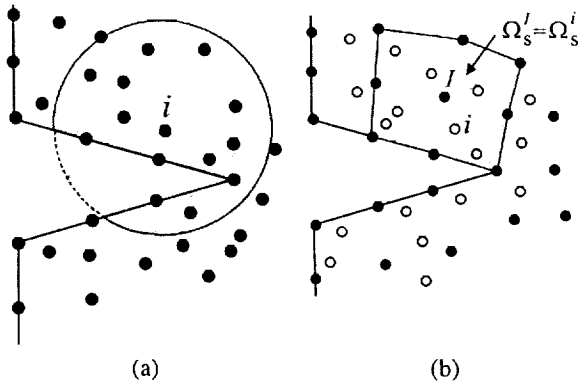


Figure 15 : Sub-domains for a domain with non-convex boundaries (a) for other meshless methods with circular sub-domains, and (b) for the present MLPG method.

expressed in terms of the weight functions and the basis in the MLS approximation, the integrand in the weak form within an integration domain consists of a continuous rational function in the presently described formulation. Again, the emphasis is placed on the fact that the shape functions for the primary and the secondary nodes have a single form of a rational function all over the integration domain, because there is no crossing of the boundaries of the sub-domains in the domain for integration. Although the MLS shape functions in a sub-domain become more complex as the number of the secondary nodes increases, a single form of a rational function may be much easier to integrate numerically, than the shape functions with different types of functions as in other meshless methods.

In summary, the secondary nodes can be added at arbitrary positions in the domain and on the global boundary, after constructing the initial polygonal sub-domains from the primary nodes, and errors in numerical results can be controlled by adding and moving the secondary nodes. Thus, to start with,

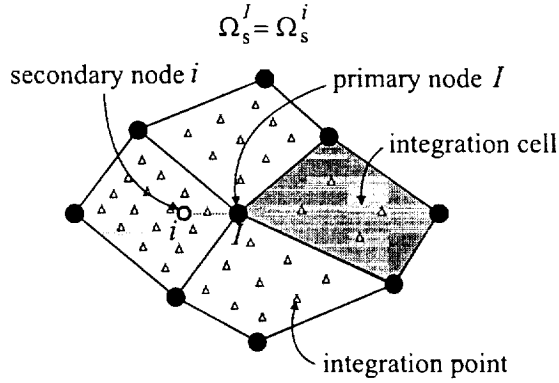


Figure 16 : Numerical integration inside each integration cell in the sub-domain $\Omega_s^I = \Omega_s^i$.

only a few primary nodes may be used, and later, a random pattern of the secondary nodes may be introduced, in an adaptive fashion to control the error of the numerical solution.

4.2 Treatments of boundary conditions and non-convex boundaries

One major difficulty in the meshless methods is considered to be the imposition of the essential boundary conditions, because, in general, the approximation functions do not satisfy the Kronecker-delta condition $\phi^I(\mathbf{x}_J) = \delta_{IJ}$ at the boundary. Most meshless methods have used Lagrange multipliers or penalty methods to impose the essential boundary conditions [Zhu and Atluri (1998)]. In some cases, meshless interpolations and FEM shape functions have been combined [Belytschko, Organ, and Krongauz (1995)], leading to a complex interface element in the regions of intersection of FEM and meshless shape functions. However, in the present method the Kronecker-delta condition is satisfied at the primary nodes on the global boundary, because the primary nodes on the global boundary are not candidates as anchors for the sub-domains for the secondary nodes in the domain, and because the weight functions, except those for the primary nodes on the global boundary, are zero at this point. Therefore, the essential boundary conditions can be imposed exactly at the primary nodes on the global boundary. On the contrary, the Kronecker-delta condition may not be satisfied at the secondary nodes on the global boundary, as a result of non-zero values for the weight functions at the secondary nodes. Since the deformations on the global boundary depend only on the primary and the secondary nodes on the boundary-segment between the primary nodes on the global boundary, the fictitious nodal values for the secondary nodes on the global boundary can be evaluated in an easy way. Fig. 14 illustrates two secondary nodes on the boundary-segment between two primary nodes on the

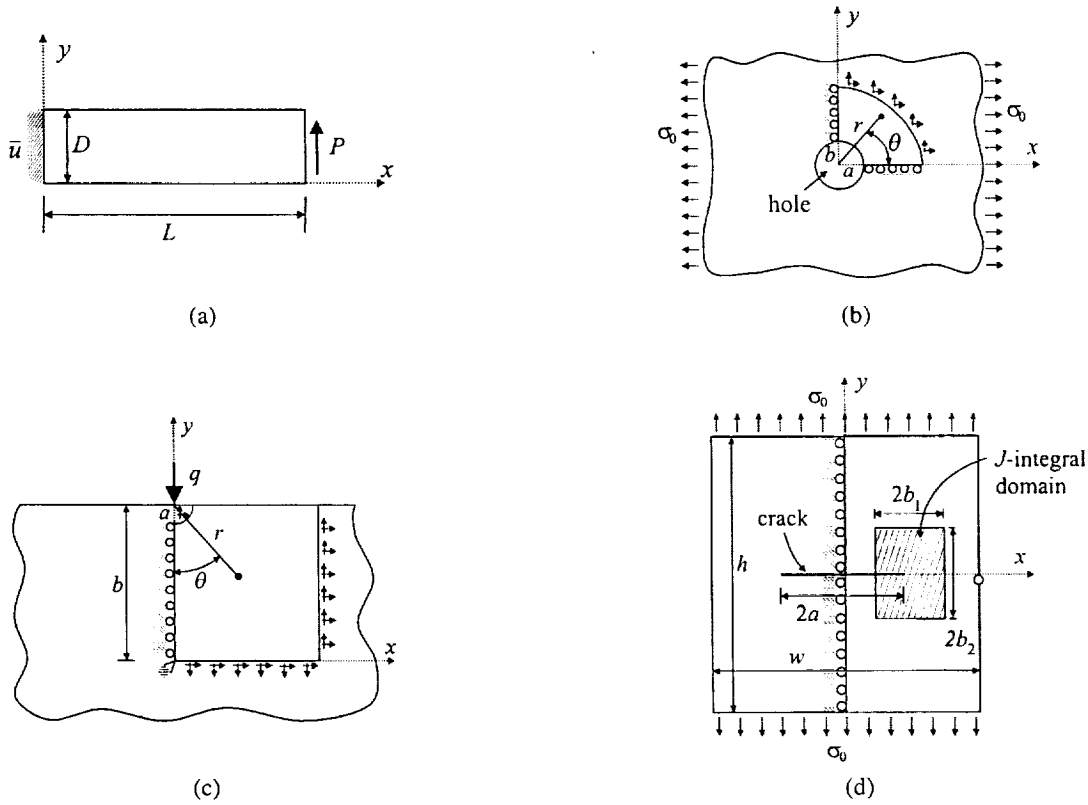


Figure 17 : Geometric description for numerical experiments: (a) a cantilever beam, (b) a plate with a hole, (c) concentrated load on a semi-infinite plate, and (d) a center cracked plate in tension.

global boundary, and the following equation can be written as

$$\begin{bmatrix} \phi^I(\mathbf{x}_I) & \phi^i(\mathbf{x}_I) & \phi^j(\mathbf{x}_I) & \phi^J(\mathbf{x}_I) \\ \phi^I(\mathbf{x}_I) & \phi^i(\mathbf{x}_I) & \phi^j(\mathbf{x}_I) & \phi^J(\mathbf{x}_I) \\ \phi^I(\mathbf{x}_I) & \phi^i(\mathbf{x}_I) & \phi^j(\mathbf{x}_I) & \phi^J(\mathbf{x}_I) \\ \phi^I(\mathbf{x}_I) & \phi^i(\mathbf{x}_I) & \phi^j(\mathbf{x}_I) & \phi^J(\mathbf{x}_I) \end{bmatrix} \begin{Bmatrix} \hat{u}^I \\ \hat{u}^i \\ \hat{u}^j \\ \hat{u}^J \end{Bmatrix} = \begin{Bmatrix} \bar{u}^I \\ \bar{u}^i \\ \bar{u}^j \\ \bar{u}^J \end{Bmatrix} \tag{32}$$

where $\hat{u}$ is the fictitious nodal displacement in MLS approximation, and $\bar{u}$ is the prescribed displacement on the global boundary. As explained before, the following conditions are satisfied at the primary nodes:

$$\hat{u}^I = \bar{u}^I \quad \text{and} \quad \hat{u}^J = \bar{u}^J \tag{33}$$

We can evaluate the fictitious nodal values $\hat{u}^i$ and $\hat{u}^j$ at the secondary nodes on the global boundary from Eq. 32, and the essential boundary conditions can be imposed directly in the computation. Consequently, we can impose the essential boundary conditions exactly at the primary and the secondary nodes on the global boundary.

The other meshless methods with circular sub-domains may lead to a difficulty near boundaries when the domain is not strictly convex [Organ, Fleming, Terry, and Belytschko (1996)]. In Fig. 15a, the deformation at the node i on the upper part of edge directly affects the deformations on the lower part, which is invisible to an observer at the node i. Some special treatments, as introduced by Organ, Fleming, Terry, and Belytschko (1996) to solve the problems with non-convex boundaries, are required in other meshless methods. In the present method, however, there is no this difficulty with a non-convex boundary, because the sub-domains for the primary and the secondary nodes are simple polygonal-local domains as shown in Fig. 15b. Therefore, we can easily deal with problems with non-convex boundaries using the present method.

4.3 Numerical integration of weak forms

To evaluate the stiffness matrix from the weak form, it is necessary to use a numerical quadrature since analytical integration is all but impossible in general. The numerical integration of the stiffness matrix usually plays an important role in the convergence of numerical solutions in meshless methods. Fig. 5 shows that the schematic features of two integration methods in meshless methods. The first method, using a background mesh, has been used in most meshless meth-

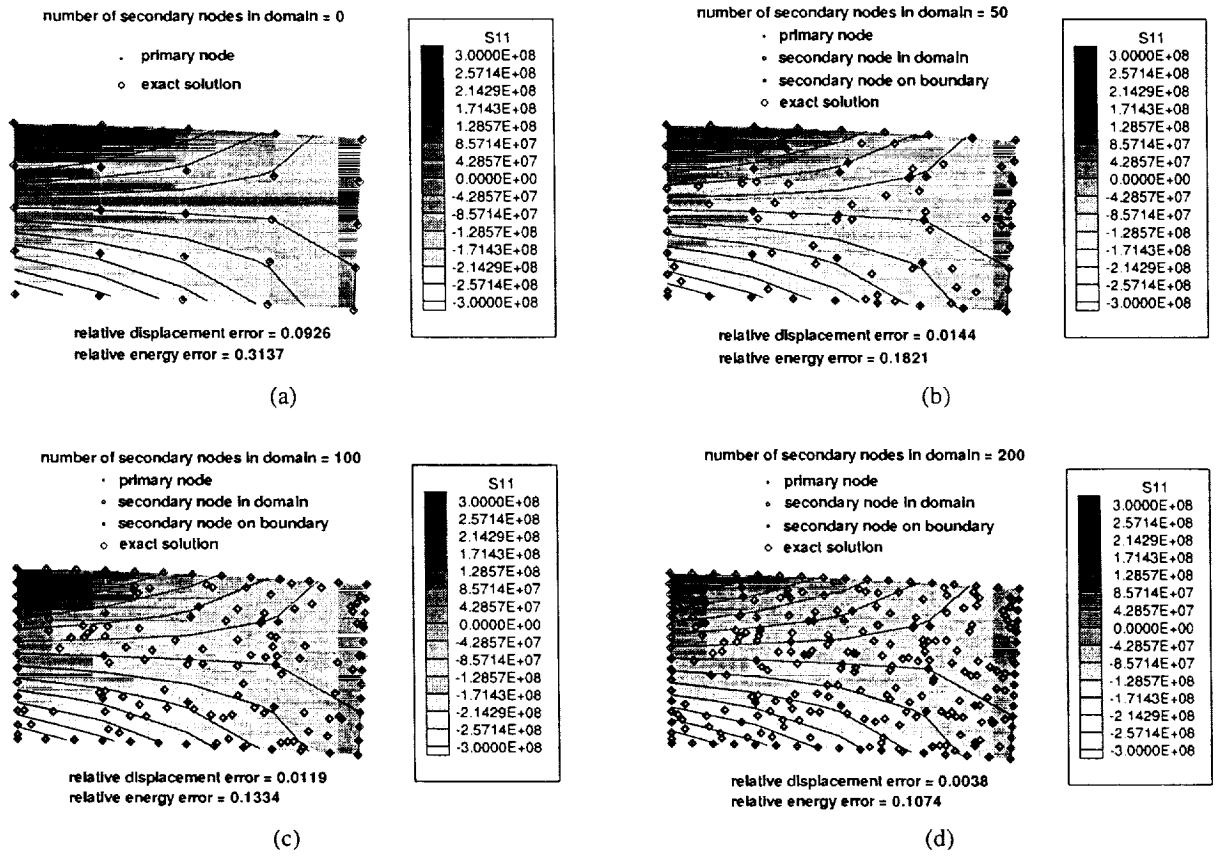


Figure 18 : Comparison of the MLPG and the exact solutions, for the problem of a cantilever beam. Displacements and distribution of stress σ_{11} (a) without secondary nodes, (b) with 50 secondary nodes, (c) with 100 secondary nodes, and (d) with 200 secondary nodes in the domain.

ods including the EFG, RKPM, and hp-clouds. Dolbow and Belytschko (1999) 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 integration, by making the integration cell to be aligned with the boundaries of sub-domains. As explained before, the present method, however, leads to a simple type of a rational function inside a cell in a sub-domain. Therefore, a cell (a quadrilateral for example) in a sub-domain ($\Omega_s^I = \Omega_s^i$ for the MLPG method) is taken as an integration domain in the present MLPG formulation, to obtain accurate numerical integration for the stiffness matrix. The schematic of the integration method is presented in Fig. 16.

Gaussian quadrature is commonly employed to numerically evaluate the integrals in the weak forms. The Gaussian quadrature can exactly integrate the polynomials of order $2n-1$ in a spatial direction, where n is the number of integration points in that spatial coordinate. Since inside an integration domain the shape functions are rational functions in the present method, the Gaussian quadrature may not be adequate to evaluate integrals in the present method properly. However, the accuracy of numerical integration may be controlled by the number of integration points, taking a proper level of polynomial as an approximation for these rational functions. In general, the more secondary nodes are involved in the sub-domain, the more integration points are required. To find 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 experiments

Several problems in two-dimensional linear elasticity are solved to illustrate the effectiveness of the present method. The numerical results of the MLPG method as applied to problems in two-dimensional elasto-statics, specifically a cantilever beam, a plate with a hole (circular and elliptic holes),

concentrated load on a semi-infinite plate, and a center cracked plate in tension as shown in Fig. 17, are now discussed. The Young modulus and the Poisson's ratio are $E = 1.0 \times 10^{10}$ and $\nu = 0.25$, respectively. We use the displacement and energy norms defined as

$$\| \mathbf{u} \| = (\int_\Omega \mathbf{u}^T \mathbf{u} \, d\Omega)^{\frac{1}{2}} \tag{34}$$

$$\| \varepsilon \| = (\frac{1}{2} \int_\Omega \varepsilon^T \mathbf{D} \varepsilon \, d\Omega)^{\frac{1}{2}} \tag{35}$$

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

$$r_u = \frac{\| \mathbf{u}^{num} - \mathbf{u}^{exact} \|}{\| \mathbf{u}^{exact} \|} \tag{36}$$

$$r_\varepsilon = \frac{\| \varepsilon^{num} - \varepsilon^{exact} \|}{\| \varepsilon^{exact} \|} \tag{37}$$

The linear basis in the MLS approximation is used in the numerical examples.

5.1 *Cantilever beam*

We first consider a cantilever beam problem shown in Fig. 17a. The exact solution for this problem is given in Timoshenko and Goodier (1970) as

$$u_1 = -\frac{P}{6EI}(y - \frac{D}{2})[3x(2L - x) + (2 + \nu)y(y - D)] \tag{38}$$

$$u_2 = \frac{P}{6EI}[x^2(3L - x) + 3\nu(L - x)(y - \frac{D}{2})^2 + \frac{4 + 5\nu}{4} D^2 x] \tag{39}$$

where

$$I = \frac{D^3}{12}$$

The stresses corresponding to the above are

$$\sigma_{11} = -\frac{P}{I}(L - x)(y - \frac{D}{2}) \tag{40}$$

$$\sigma_{22} = 0 \tag{41}$$

$$\sigma_{12} = -\frac{Py}{2I}(y - D) \tag{42}$$

We use regularly distributed primary nodes for a model with $D = 4.0$ and $L = 8.0$ to examine the effects of the secondary nodes. The essential and traction boundary conditions are applied at the left and right sides of the beam, respectively, and P in Eqs. 38-42 is $(-)1.0 \times 10^8$. The solution without the secondary nodes is exactly the same as that of the finite element method, as shown in Fig. 18a. We distribute 50, 100, and 200 secondary nodes randomly in the domain through generating random numbers. To improve the deformations on the global boundary, the secondary nodes are placed between the primary nodes on the global boundary, as shown in Fig. 18. We use 3×3, 5×5, and 8×8 integration points in an integration domain for 50, 100, and 200 secondary nodes in the domain, respectively. It is required to increase the number of integration points as the secondary nodes are added. Comparison between the MLPG and the exact solutions are given in Fig. 18. In these results, the relative displacement error changes from 9.26% to 0.38%, and the relative energy error decreases from 31.37% to 10.74% by adding 200 secondary nodes. In Fig. 19, the convergences in the relative displacement and energy norms for the MLPG are presented with a representative nodal density $\sqrt{N_s}$, where N_s is the number of secondary nodes in the domain. The present method gives better results than the solutions without the secondary nodes. Particularly, the relative displacement error decreases rapidly even though the secondary nodes are added randomly.

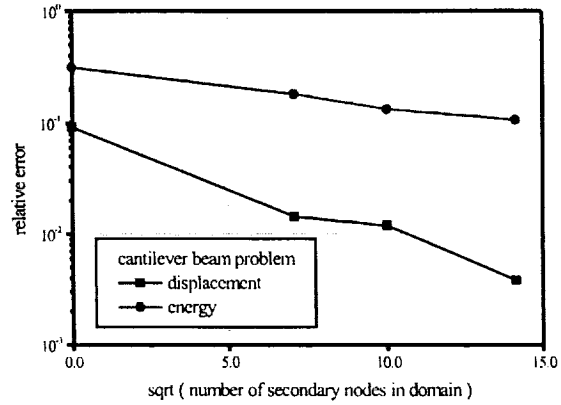


Figure 19 : Convergence for the displacement and the energy norms in a cantilever beam problem, with irregularly distributed secondary nodes in the domain.

5.2 *Infinite plate with a hole*

We consider an infinite plate with a circular hole of radius a ($b = a$ in Fig. 17b). 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. 17b. The exact solutions for stresses are

$$\sigma_{11} = \sigma_0[1 - \frac{a^2}{r^2}(\frac{3}{2}\cos 2\theta + \cos 4\theta) + \frac{3a^4}{2r^4}\cos 4\theta] \tag{43}$$

$$\sigma_{12} = \sigma_0[-\frac{a^2}{r^2}(\frac{1}{2}\sin 2\theta + \sin 4\theta) + \frac{3a^4}{2r^4}\sin 4\theta] \tag{44}$$

$$\sigma_{22} = \sigma_0[-\frac{a^2}{r^2}(\frac{1}{2}\cos 2\theta - \cos 4\theta) - \frac{3a^4}{2r^4}\cos 4\theta] \tag{45}$$

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

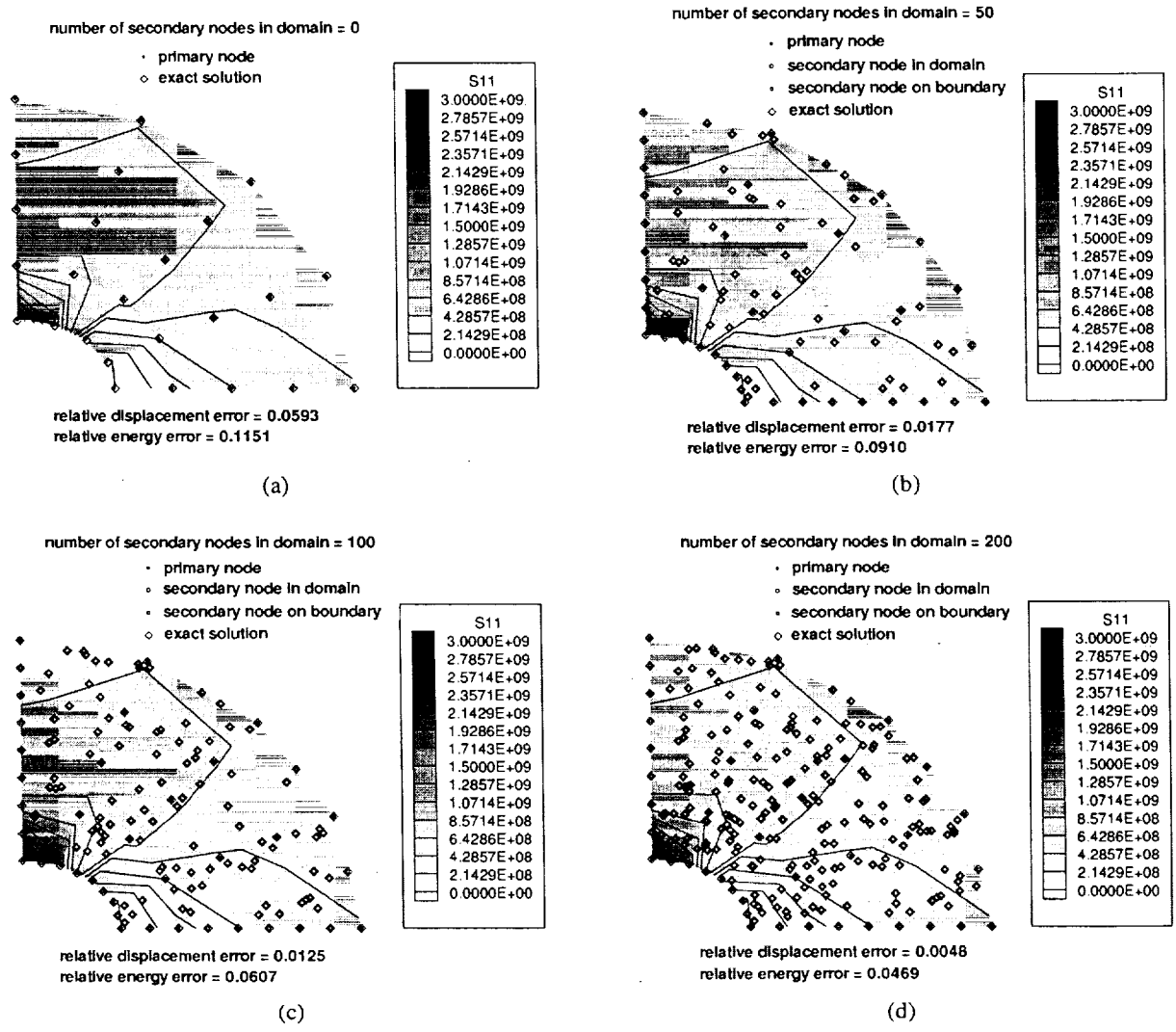


Figure 20 : Comparison of the MLPG and the exact solutions, for the problem of a plate with a circular hole. Displacements and distribution of stress σ_{11} (a) without secondary nodes, (b) with 50 secondary nodes, (c) with 100 secondary nodes, and (d) with 200 secondary nodes in the domain.

by

$$u_1 = \frac{1+\nu}{E}\sigma\Big(\frac{1}{1+\nu}r\cos\theta + \frac{2}{1+\nu}\frac{a^2}{r}\cos\theta + \frac{1}{2}\frac{a^2}{r}\cos 3\theta - \frac{1}{2}\frac{a^4}{r^3}\cos 3\theta\Big) \quad (46)$$

$$u_2 = \frac{1+\nu}{E}\sigma\Big(-\frac{\nu}{1+\nu}r\sin\theta - \frac{1-\nu}{1+\nu}\frac{a^2}{r}\sin\theta + \frac{1}{2}\frac{a^2}{r}\sin 3\theta - \frac{1}{2}\frac{a^4}{r^3}\sin 3\theta\Big) \quad (47)$$

Due to symmetry, only a part, $0 \le r \le 4$, of the upper right quadrant of the plate is modeled as shown in Fig. 17b. Symmetry conditions are imposed on the left and bottom edges, i.e. $u_1 = 0, t_2 = 0$ is prescribed on the left edge and $u_2 = 0, t_1 = 0$ 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 $r = 4.0$. Secondary nodes are distributed randomly in the domain, as shown in Fig. 20. 4×4, 6×6, and 9×9 integration points in an integration domain are used for models with 50, 100, and 200 secondary nodes in the domain, respectively. One secondary node, on each boundary-segment between the primary nodes on the global boundary, is added to improve the deformations on the global boundary. The comparison between the MLPG and the exact solutions are shown in Fig. 20.

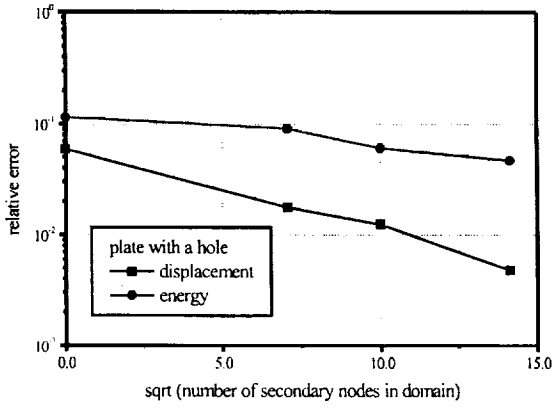
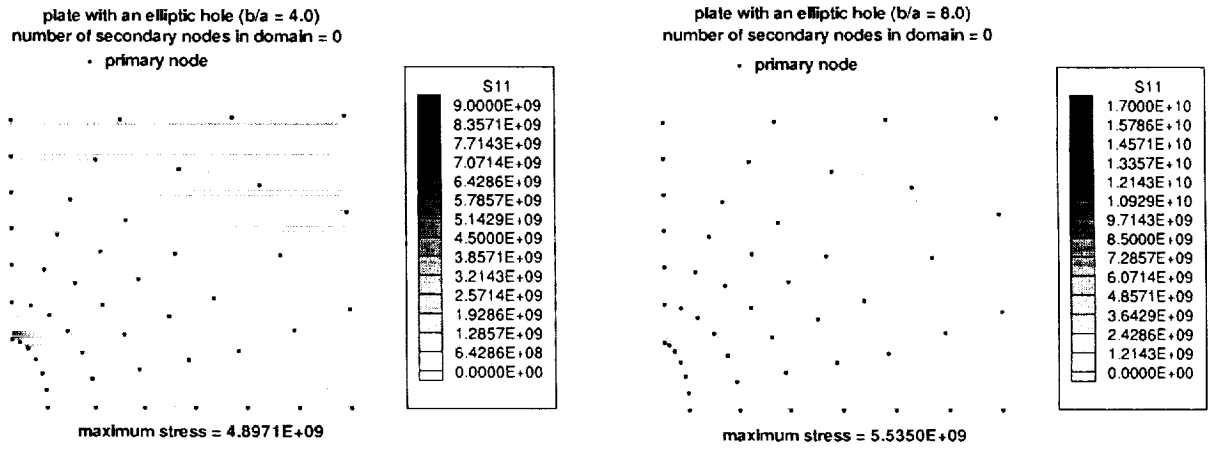


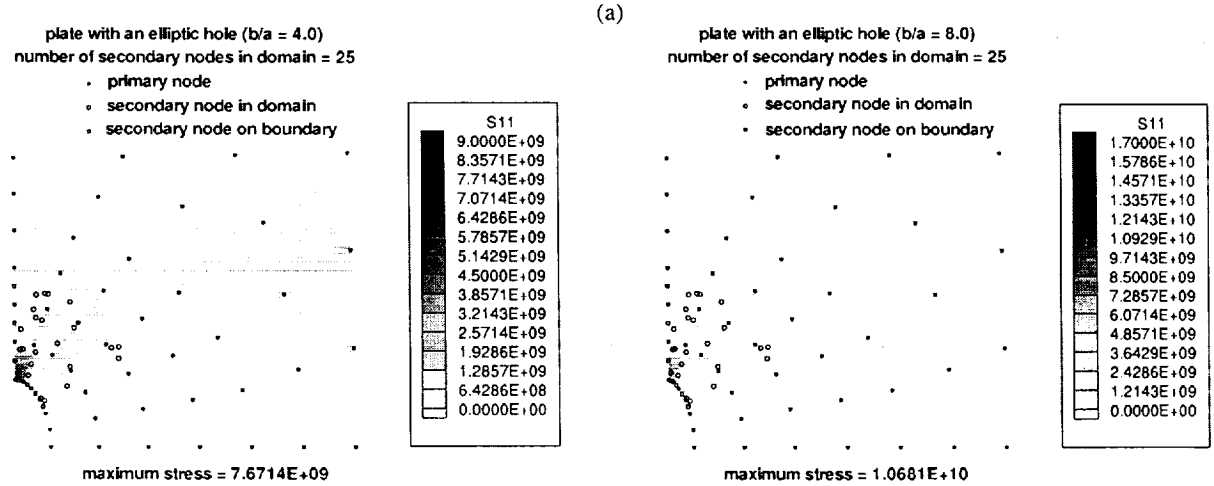
Figure 21 : Convergence for the displacement and the energy norms in the problem of a plate with a circular hole, with irregularly distributed secondary nodes in the domain.

The relative displacement error decreases from 5.93% to 0.48%, and relative energy error decreases from 11.51% to 4.69% by adding 200 secondary nodes in the domain. The convergence with the number of nodes is shown in Fig. 21 with a representative nodal density $\sqrt{N_s}$ for the irregularly distributed secondary nodes in the domain.

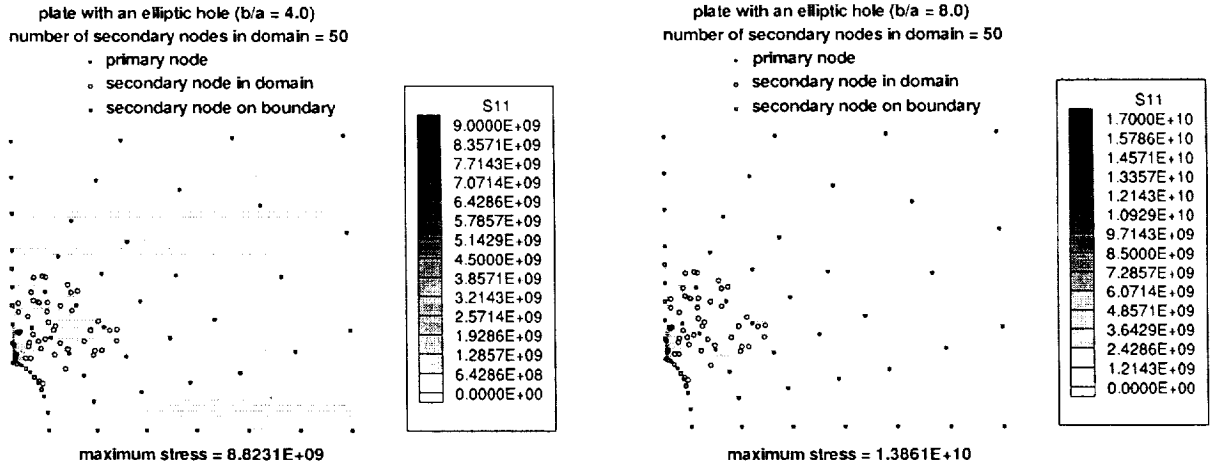
Agreement between the MLPG and the exact solutions is excellent in this example. As a more severe case, we consider a plate with an elliptic hole for aspect ratios of 4 : 1, and 8 : 1, respectively. The stress concentration factors for these elliptical-hole problems in an infinite plate under uniform traction are 9.0 and 17.0, respectively. The secondary nodes in the domain and on the global boundary are added near the elliptical hole where the major-axis is of the ellipse intersects the hole surface. The models with randomly distributed (25, 50, 100 and 200) secondary nodes in the domain, use 4×4, 6×6, 9×9 and 11×11 integration points in an integration cell, respectively.



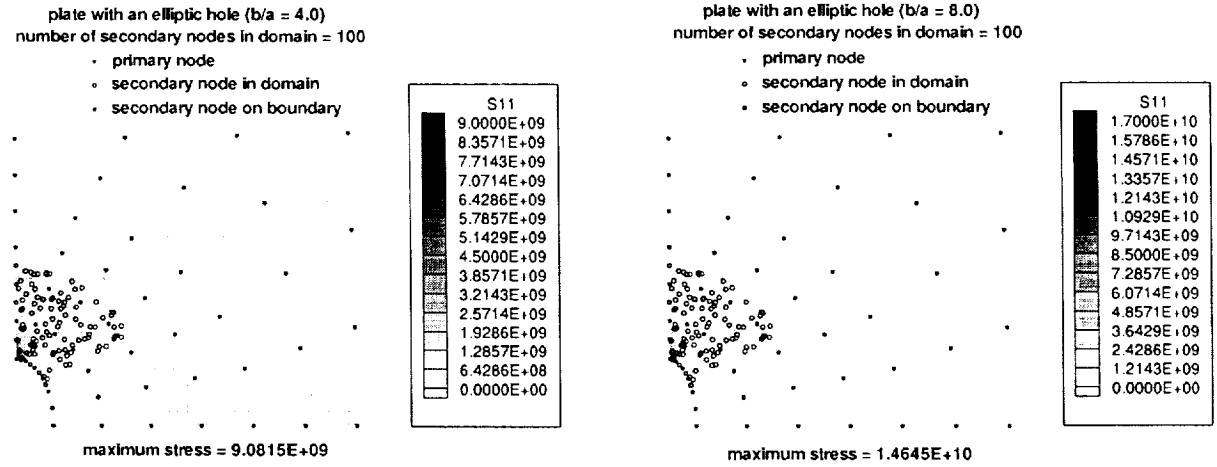
(a)



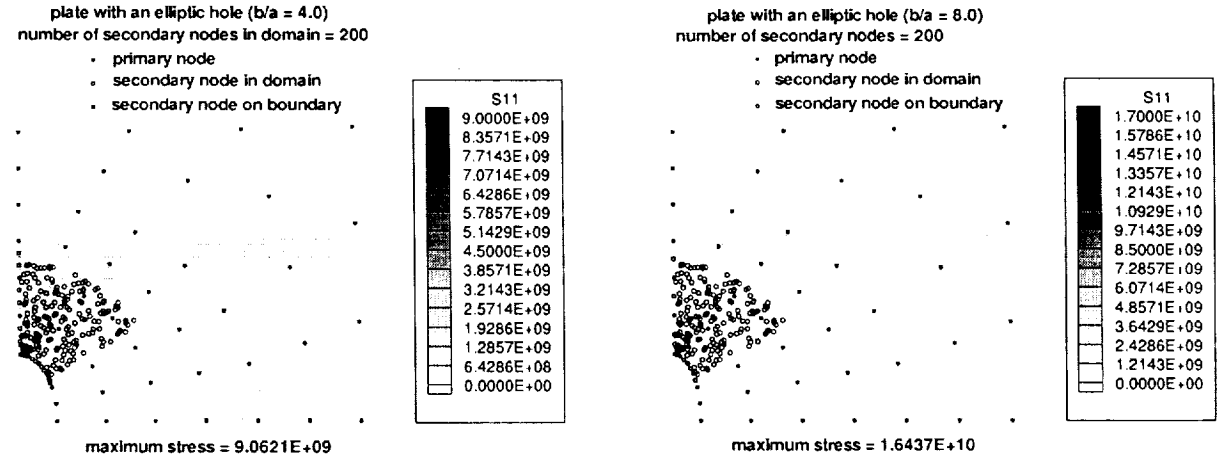
(b)



(c)



(d)



(e)

Figure 22 : The MLPG solutions for the problem of a plate with an elliptical hole: ($b/a = 4.0$ and $b/a = 8.0$). Displacements and distribution of stress σ_{11} (a) without secondary nodes, (b) with 25 secondary nodes, (c) with 50 secondary nodes, (d) with 100 secondary nodes, and (e) with 200 secondary nodes in the domain.

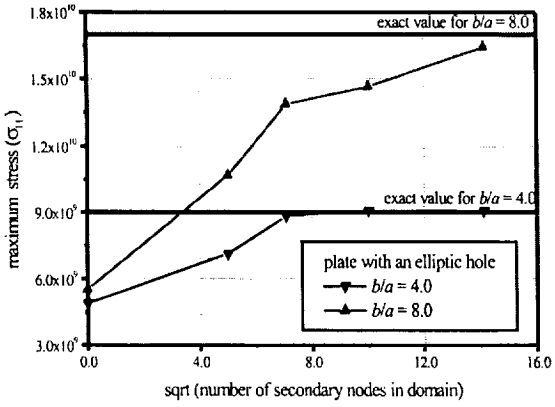


Figure 23 : Convergence for the displacement and the energy norms in the problem of a plate with an elliptical hole, with irregularly distributed secondary nodes in the domain.

Note that the sub-domains not influenced by the secondary nodes require only 2×2 integration points because of simple shape functions in this region. Fig. 22 shows the stress distributions by adding 25, 50, 100 and 200 secondary nodes. The stress σ_{11} near the tip of the major-axis of the elliptical hole is plotted in Fig. 23 with a representative nodal density $\sqrt{N_s}$. The results show that the stress concentration factors approach the exact values as the number of the secondary nodes increases.

5.3 *Concentrated load on a semi-infinite plate*

Consider the concentrated load on an infinite plate as shown in Fig. 17c. The plate is assumed to be of unit width so that the concentrated load is equal to q. The exact solution for this problem [Saada (1974)] is given by

$$\sigma_{rr} = -\frac{2q}{\pi}\frac{\cos\theta}{r} \tag{48}$$

$$\sigma_{\theta\theta} = \sigma_{r\theta} = \sigma_{zz} = 0 \tag{49}$$

$$e_{rr} = -\frac{2q}{\pi E}\frac{\cos\theta}{r} \tag{50}$$

$$e_{\theta\theta} = \frac{2\nu q}{\pi E}\frac{\cos\theta}{r} \tag{51}$$

$$e_{r\theta} = 0 \tag{52}$$

The displacements are

$$u_r = \frac{2q}{\pi E}\cos\theta ln\frac{d}{r} - \frac{(1-\nu)q}{\pi E}\theta\sin\theta \tag{53}$$

$$u_\theta = \frac{q(1+\nu)}{\pi E}\sin\theta - \frac{2q}{\pi E}\sin\theta ln\frac{d}{r} - \frac{(1-\nu)q}{\pi E}\theta\cos\theta \tag{54}$$

where d is a distance from the point of the concentrated load q. We use the right half model with 4.0×4.0 size as shown in Fig. 17c. The concentrated load q is 1.0×10^9 in this example. Since the displacements are singular at the point of application of the concentrated load q, we use the model with a small circular hole $a = 0.5$. The symmetric boundary condition is applied at the left side of the half model, and tractions are applied at the other boundaries from the exact solution. In numerical calculation, the distance d is set to be 4.0 to give the fixed boundary condition at the center. Convergence studies are carried out using three different numbers of secondary nodes, namely 50, 100, and 200. 4×4, 6×6, and 9×9 integration points are used in an integration domain for these three models. The comparison between the MLPG and the exact solutions is plotted in Fig. 24, and the convergence is shown in Fig. 25. In this example, we obtain the similar trends as shown in the problems of a cantilever beam and a plate with a hole.

5.4 *Center cracked plate in tension*

Next, we consider a center cracked plate in tension. Due to symmetry, the right half as shown in Fig. 17d 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 J-integral using domain integration [Nikishkov and Atluri (1987) and Anderson (1991)]. 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 $K_0 = \sigma_0\sqrt{\pi a}$ [Tada, Paris, and Irwin (1977) and Wu and Carlsson (1991)]. In this numerical example, we use regularly distributed primary nodes as shown in Fig. 26, and secondary nodes are added randomly near the crack tip. 4×4, 6×6 and 8×8 integration points are used for models with 25, 50 and 100 secondary nodes in the domain, respectively. However, only 2×2 integration points are used in the sub-domains which are not influenced by the secondary nodes. The stress σ_{22} at the crack tip is higher as the number of the secondary nodes near the crack tip increases. By adding secondary nodes on the crack lines, the improved deformations near the crack tip are obtained as shown in Fig. 27. The deformations of the crack lines are very important to recover the stresses correctly near the crack tip. In Fig. 28, the errors in the stress intensity factors evaluated from J-integral are plotted against the number of the secondary nodes. The stress intensity factor approaches the target solution as the number of secondary nodes in the domain increase.

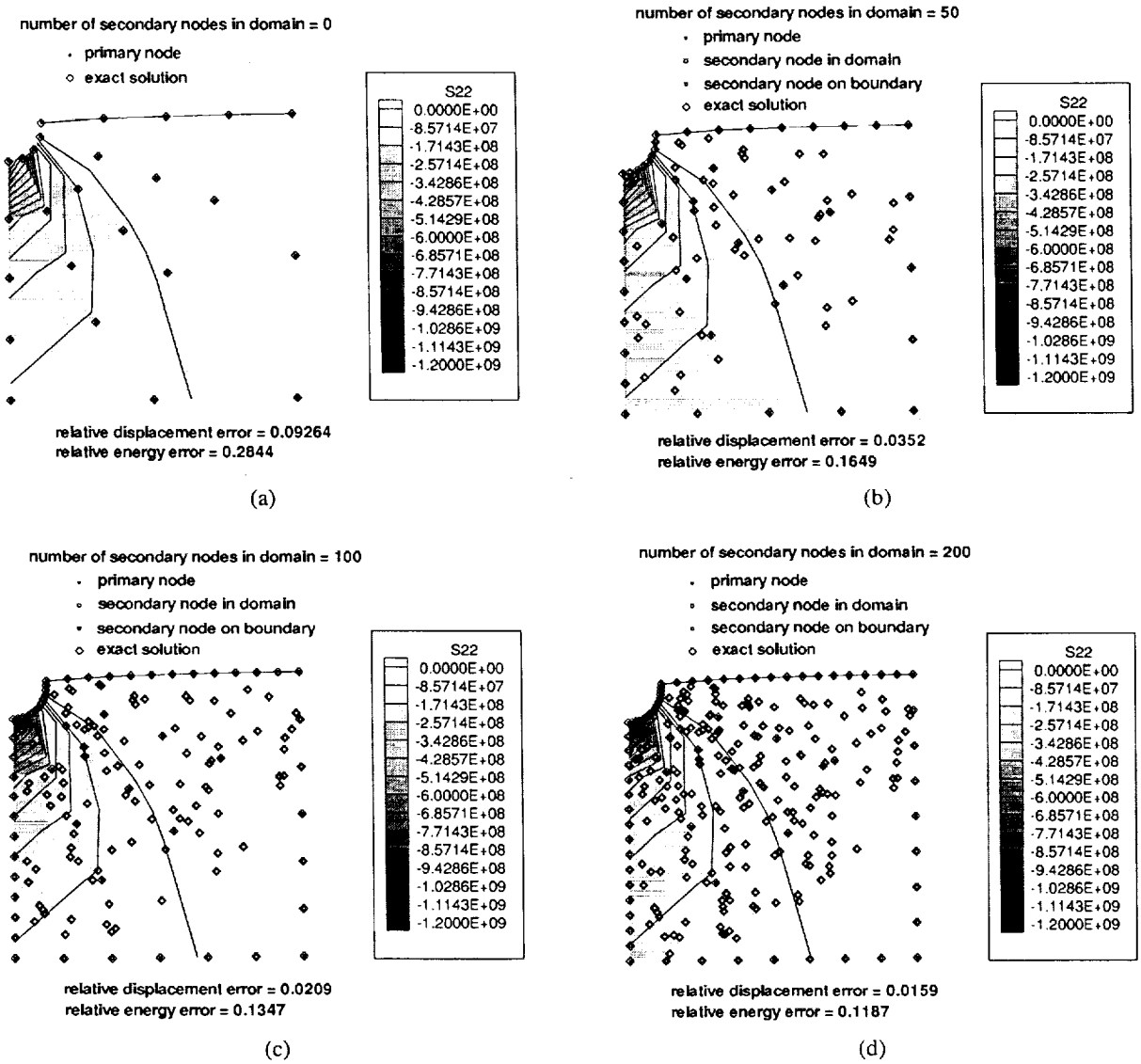


Figure 24 : Comparison of the MLPG and the exact solutions for the problem of a concentrated load on a semi-infinite plate. Displacements and distribution of stress σ_{22} (a) without secondary nodes, (b) with 50 secondary nodes, (c) with 100 secondary nodes, and (d) with 200 secondary nodes in the domain.

6 Concluding remarks

A method, based on the meshless local Petrov-Galerkin (MLPG) concept for solving boundary value problems has been presented in this paper. The concept of primary and secondary nodes was introduced, and the weight functions for primary and secondary nodes were defined to yield a single type of a rational function all over the integration domain. The approach presented in this paper alleviates a major difficulty in the numerical integration to evaluate weak forms in other meshless methods. Additional mesh for the secondary nodes is not required to improve the accuracy of solution. The essential boundary conditions are easily taken care of as in FEM, and the non-convex boundaries can be treated without a special technique.

A clear advantage of the present method is that the secondary nodes can be placed at any random locations, without the burdensome task of constructing a new mesh to enrich the solution. The present MLPG method can control errors in numerical results by adding the secondary nodes, using only the sub-domains constructed from the primary nodes. Therefore, the present method can be a useful tool for error control and adaptive calculation in the field of computational mechanics. The numerical experiments show that the present method is very efficient for adaptive error control by placing

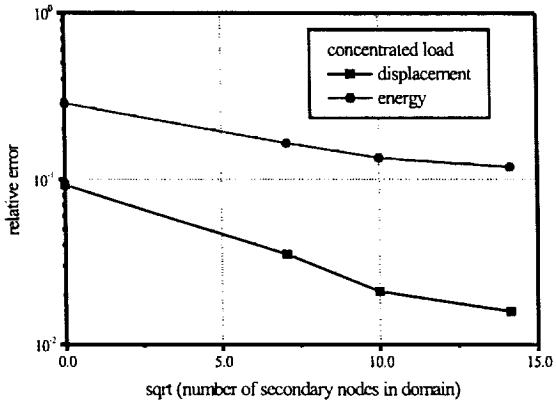


Figure 25 : Convergence for the displacement and the energy norms in the problem of a concentrated load on a semi-infinite plate, with irregularly distributed secondary nodes in the domain.

secondary nodes arbitrarily in the domain and on the global boundary. The present method can be easily implemented in three-dimensional problems with the same advantages. Further results based on the current approach will be presented in a series of forthcoming papers.

Acknowledgement: This work was supported by the Office of Naval Research, with Dr. Y.D.S. Rajapakse as the cognizant program official, and by NASA Langley Research Center, with Dr. I.S. Raju as the cognizant program official.

References

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

Atluri, S. N.; Cho, J. Y.; Kim, H. (1999a): Analysis of thin beams, using the meshless local petrov-galerkin method, with generalized moving least squares interpolations. *Comput. Mech.*, vol. 24, pp. 334–347.

Atluri, S. N.; Gallagher, R. H.; Zienkiewicz, O. C. (1983): *Hybrid & mixed finite element methods*. J. Wiley & Sons.

Atluri, S. N.; Kim, H.; Cho, J. Y. (1999b): A critical assessment of the truly meshless local petrov-galerkin (mlpg), and local boundary integral equation (lbie) methods. *Comput. Mech.*, vol. 24, pp. 348–372.

Atluri, S. N.; Zhu, T. (1998a): A new meshless local petrov-galerkin (mlpg) approach in computational mechanics. *Comput. Mech.*, vol. 22, pp. 117–127.

Atluri, S. N.; Zhu, T. (1998b): A new meshless local petrov-galerkin (mlpg) approach to nonlinear problems in computer modeling and simulation. *Computer Modeling and Simulation in Engrg.*, vol. 3, pp. 187–196.

Babuška, I.; Melenk, J. (1997): The partition of unity method. *Int. J. Num. Meth. Eng.*, vol. 40, pp. 727–758.

Belytschko, T.; Organ, D.; Krongauz, Y. (1995): A coupled finite element - element-free galerkin method. *Comput. Mech.*, vol. 17, pp. 186–195.

Dolbow, J.; Belytschko, T. (1999): Numerical integration of the galerkin weak form in meshless methods. *Comput. Mech.*, vol. 23, pp. 219–230.

Duarte, C. A.; Oden, J. T. (1996): Hp-cloud - a meshless method to solve boundary-value problems. *Comput. Methods Appl. Mech. Engrg.*, vol. 139, pp. 237–262.

Gingold, R. A.; Monaghan, J. J. (1977): Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Mon. Not. Roy. Astron. Soc.*, vol. 181, pp. 375–389.

Liu, W. K.; Jun, S.; Zhang, Y. (1995): Reproducing kernel particle methods. *Int. J. Num. Meth. Fluids*, vol. 20, pp. 1081–1106.

Liu, W. K.; Uras, R. A.; Chen, Y. (1998): Enrichment of the finite element method with the reproducing kernel particle method. *J. Appl. Mech. ASME*, vol. 64, pp. 861–870.

Nayroles, B.; Touzot, G.; Villon, P. (1992): Generalizing the finite element method: diffuse approximation and diffuse elements. *Comput. Mech.*, vol. 10, pp. 307–318.

Nikishkov, G. P.; Atluri, S. N. (1987): An equivalent domain integral method for computing crack-tip integral parameters in nonelastic, thermo-mechanical fracture. *Engrg. Fracture Mech.*, vol. 26, pp. 851–867.

Oñate, E.; Idelsohn, S.; Zienkiewicz, O. C.; Taylor, R. L. (1996): A finite point method in computational mechanics. application to convective transport and fluid flow. *Int. J. Num. Meth. Eng.*, vol. 39, pp. 3839–3866.

Oden, J. T.; Duarte, C. A. M.; Zienkiewicz, O. C. (1998): A new cloud-based hp finite element method. *Comput. Methods Appl. Mech. Engrg.*, vol. 153, pp. 117–126.

Organ, D.; Fleming, M.; Terry, T.; Belytschko, T. (1996): Continuous meshless approximations for nonconvex bodies by diffraction and transparency. *Comput. Mech.*, vol. 18, pp. 225–235.

Saada, A. S. (1974): *Elasticity theory and application*. Pergamon Press.

Strouboulis, T.; Babuska, I.; Copps, K. (1998): The design and analysis of the generalized finite element method. *Comput. Methods Appl. Mech. Engrg.* In press.

Tada, H.; Paris, P. C.; Irwin, G. R. (1977): *The stress analysis of cracks handbook.* Del Research Corporation.

Timoshenko, S. P.; Goodier, J. N. (1970): *Theory of Elasticity.* 3rd Edn. McGraw-Hill.

Wu, X. R.; Carlsson, A. J. (1991): *Weight functions and stress intensity factor solutions.* Pergamon press.

Zhu, T.; Atluri, S. N. (1998): A modified collocation & a penalty formulation for enforcing the essential boundary conditions in the element free galerkin method. *Comput. Mech.*, vol. 21, pp. 211–222.

Zhu, T.; Zhang, J. D.; Atluri, S. N. (1998a): A local boundary integral equation (lbie) method in computational mechanics, and a meshless discretization approach. *Comput. Mech.*, vol. 21, pp. 223–235.

Zhu, T.; Zhang, J. D.; Atluri, S. N. (1998b): A meshless local boundary integral equation (lbie) method for solving nonlinear problems. *Comput. Mech.*, vol. 22, pp. 174–186.

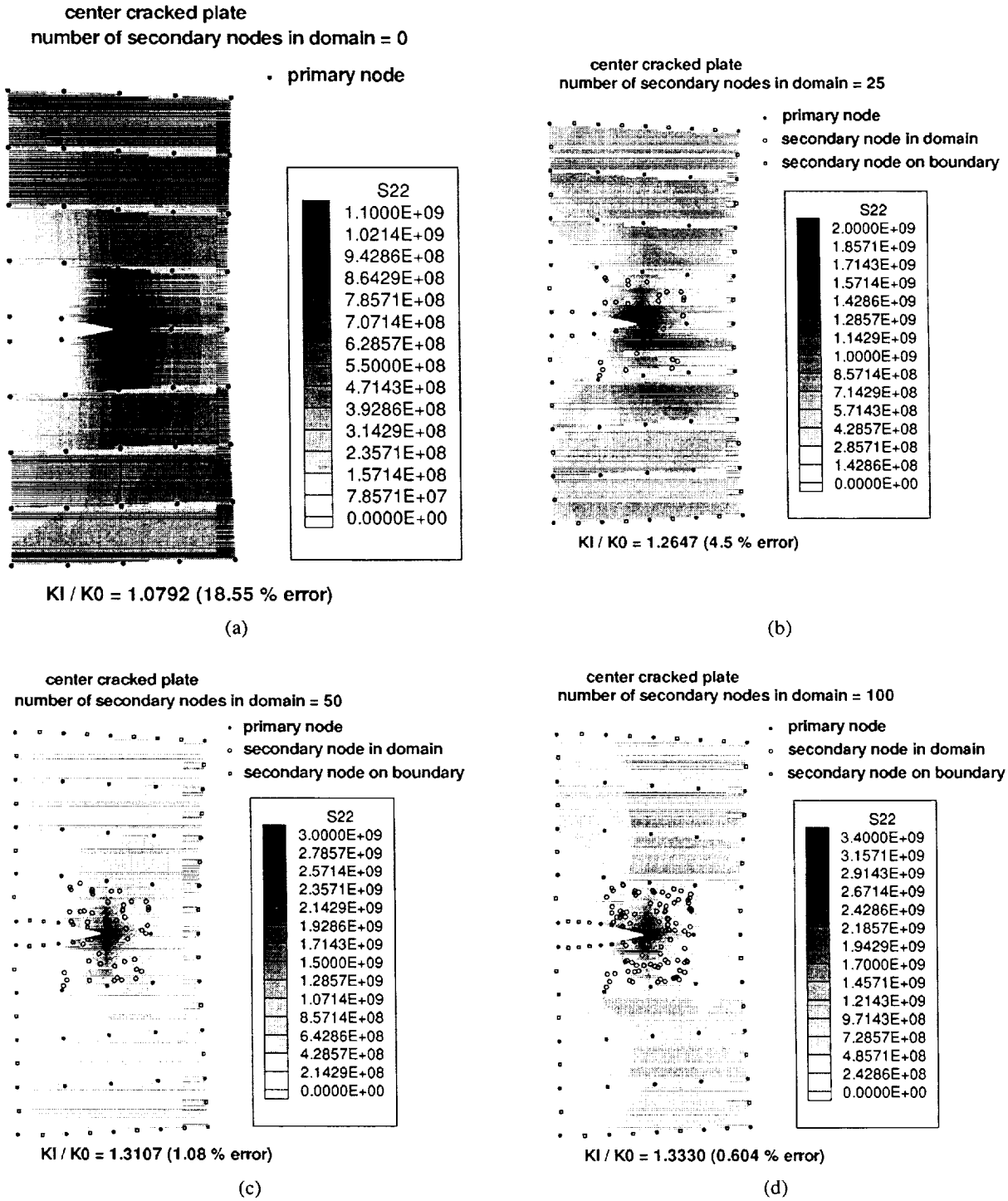


Figure 26 : The MLPG solutions for the problem of a center cracked plate. Displacements and distribution of stress σ_{22} (a) without secondary nodes, (b) with 25 secondary nodes, (c) with 50 secondary nodes, and (d) with 100 secondary nodes in the domain.

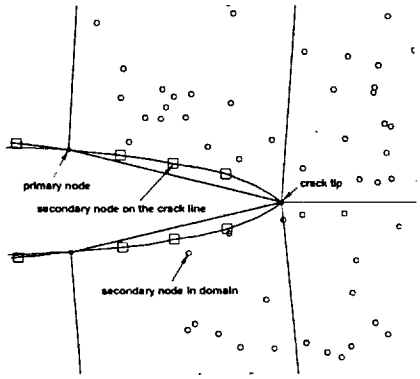


Figure 27 : Deformation near crack tip, including secondary nodes on the crack line.

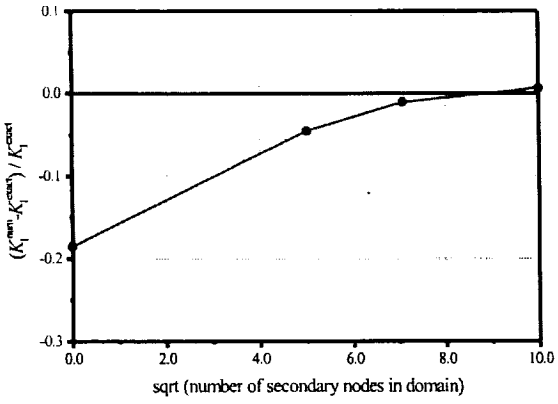


Figure 28 : Errors in the evaluation of stress intensity factor, versus the number of secondary nodes in the domain for a center cracked plate problem.