

Research Article

Radial Basis Point Interpolation Method with Reordering Gauss Domains for 2D Plane Problems

Shi-Chao Yi,¹ Fu-jun Chen,¹ and Lin-Quan Yao²

¹*School of Mathematical Sciences, Soochow University, Suzhou 215006, China*

²*School of Urban Rail Transportation, Soochow University, Suzhou 215137, China*

Correspondence should be addressed to Fu-jun Chen; chenfujun20@163.com

Received 13 June 2014; Revised 28 August 2014; Accepted 28 August 2014; Published 23 December 2014

Academic Editor: Guan H. Yeoh

Copyright © 2014 Shi-Chao Yi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We present novel Gauss integration schemes with radial basis point interpolation method (RPIM). These techniques define new Gauss integration scheme, researching Gauss points (RGD), and reconstructing Gauss domain (RGD), respectively. The developments lead to a curtailment of the elapsed CPU time without loss of the accuracy. Numerical results show that the schemes reduce the computational time to 25% or less in general.

1. Introduction

In the past two decades, the development and application of meshfree methods have attracted much attention. One of the reasons is the versatility of meshfree methods for complex geometry of solids and flexibility for different engineering problems [1]. Element-free Galerkin (EFG) method, which is originated by Belytschko et al. [2–5], is one of the most widely used meshfree methods. The key advantage of EFG method is that only nodal data is required and no element connectivity is needed, when moving least-squares (MLS) interpolation is used to construct trial and test functions. It is currently widely used in computational mechanics and other areas, such as by Sarboland and Aminataei [6] for nonlinear nonhomogeneous Burgers Equation and Pirali et al. [7] for crack discontinuities problem. In the meantime, some new techniques are used to improve the performance of the MLS, complex variable moving least-squares [8–10] and improved complex variable moving least-squares [11], the moving least-squares with singular weight function [12], and so forth. But shape functions constructed by MLS interpolation do not possess Kronecker delta function property; the treatment of essential boundary conditions is one of the typical drawbacks. Thus, many special techniques have been proposed to impose essential boundary conditions [13–15], such as point collocation [13], Lagrange multipliers [2], singular weighting functions [14], and penalty method [15]. None of these methods is fully satisfactory, as

they still need additional efforts to enforce essential boundary conditions.

In order to totally eliminate the drawback associated with EFG method for imposing essential boundary conditions, Liu and Gu have developed the point interpolation methods (PIM) by using polynomial basis or/and radial basis function (RBF) [16–19]. However, singularity may occur if arrangement of the nodes is not consistent with the order of polynomial basis, while the inverse of moment matrix always exists for arbitrarily scattered nodes with radial basis [20]. In this paper, we mainly care about the radial point interpolation method (RPIM) studied by Wang and Liu in [21], which is based on the global weak form.

On the other hand, high computational cost is still one of the main drawbacks in meshfree method with Galerkin weak form. In the RPIM method for a given computational point, an $n \times n$ linear system (the coefficient matrix is called the moment matrix) should be solved to construct shape functions if radial basis functions are selected, where n is the number of nodes in the support domain. Furthermore, if m monomial basis functions are added, the linear system will be extended to $(n + m) \times (n + m)$. This is very time-consuming especially for the meshless methods based on weak forms, where integration should be employed and the number of computational points is much bigger than the number of nodes.

In fact, common meshless method is based on the numerical integration for Gauss domains with RPIM. It is a tedious process when the integration points are extremely more than the nodes. In practical work, some Gauss point has the same nodes as some neighboring computational points. Thus, they only need one shape function with them. A process will be needed to find these nodes. This scheme is called researching Gauss points method (RGP). But the storage of searching program will increase quickly with the increasing number of nodes and Gauss points. Thus, more computational time is spent. To avoid this, reconstructing Gauss domains (RGD) is presented. All computational points share one same influence domain (the same nodes) in the domain. This scheme offsets the RGP's disadvantage without the searching program. The reason why we call it researching Gauss points method is that the RGP method needs a searching nodes program same as in the RPIM. The two techniques are collectively named reordering Gauss domain (RG) method.

The remainder of the paper is arranged as follows. In Section 2, the radial basis point interpolation method (RPIM) is presented to get the shape functions. In Section 3, a Galerkin weak form and its numerical algorithm are studied for 2D solid mechanics problems. The reordering Gauss domains methods are then constructed in Section 4. In Section 5, several examples are presented to show the effectiveness of the proposed method and some parameters' performances of the proposed method are also investigated. Finally, we end this paper with some conclusions in Section 6.

2. Radial Basis Point Interpolation Method (RPIM)

The RPIM interpolation $u^h(x)$ of u , for all $x \in \Omega_x$, can be defined by

$$u^h(x) = \sum_{i=1}^n R_i(x) a_i + \sum_{j=1}^m p_j(x) b_j \quad (1)$$

with the constraint condition

$$\sum_{i=1}^n p_j(x_i) a_i = 0, \quad j = 1, 2, \dots, m, \quad (2)$$

where $R_i(x)$ is the radial basis function (RBF), n is the number of nodes in the neighborhood of x , $p_j(x)$ is the monomial in the space coordinates $\mathbf{p}^T(x) = [1, x, y]$, m is the number of polynomial basis functions, and coefficients a_i and b_j are interpolation constants. The variable x of the radial basis function $R_i(x)$ is the distance between the interpolation point x and a node x_i . It is necessary to construct the interpolation function here to solve the equations.

There are a number of types of radial basis functions. Characteristics of radial basis functions have been widely investigated in [20, 22]. In this paper, the following multi-quadrics (MQ) radial basis function is used:

$$R_i(x) = (r_i^2 + C^2)^q, \quad (3)$$

where $r_i = \sqrt{(x - x_i)^2 + (y - y_i)^2}$ and C and q are two parameters. C is defined as

$$C = \alpha_0 d_i, \quad (4)$$

where α_0 is a dimensionless coefficient and d_i is a parameter of the nodal distance. For regularly distributed nodal case, d_i is the shortest distance between the node x_i and its neighbor nodes. Effects of α_0 and q have been studied in detail in [22]. In static analysis of 2D solid problems, it has been found that $\alpha_0 = 1.0$ and $q = 1.03$ lead to good results. Hence, these numbers are used in this paper.

The second term in (1) consists of polynomials. To ensure invertible interpolation matrix of RBF, the polynomial added into the RBF cannot be arbitrary. A low-degree polynomial is often needed to augment RBF to guarantee the nonsingularity of the moment matrix. In addition, the linear polynomial added into the RBF can also ensure linear consistency and improve the interpolation accuracy [22]. Thus, the linear polynomial is added into the MQ RBF in the following discussion.

Coefficients a_i and b_j in (1) can be determined by enforcing that (1) and (2) be satisfied at the n nodes surrounding point x . Equations (1) and (2) can be rewritten in the matrix form:

$$\mathbf{G}\mathbf{a}_0 \triangleq \begin{bmatrix} \mathbf{R}_0 & \mathbf{P}_0 \\ \mathbf{P}_0^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} = \begin{bmatrix} \mathbf{u}_s \\ \mathbf{0} \end{bmatrix}, \quad (5)$$

where

$$\mathbf{a}_0^T = [\mathbf{a}^T \quad \mathbf{b}^T]^T = \{a_1, a_2, \dots, a_n, b_1, b_2, b_3\},$$

$$\mathbf{u}_s^T = \{u_1, u_2, \dots, u_n\},$$

$$\mathbf{R}_0 = \begin{bmatrix} R_1(r_1) & R_2(r_1) & \cdots & R_n(r_1) \\ R_1(r_2) & R_2(r_2) & \cdots & R_n(r_2) \\ \vdots & \vdots & \ddots & \vdots \\ R_1(r_n) & R_2(r_n) & \cdots & R_n(r_n) \end{bmatrix}_{n \times n}, \quad (6)$$

$$\mathbf{P}_0^T = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \\ y_1 & y_2 & \cdots & y_n \end{bmatrix}_{3 \times n}.$$

The matrix \mathbf{R}_0 is symmetric, and thus the matrix \mathbf{G} is symmetric. Then, the interpolation equation (1) is finally expressed as

$$u^h(x) = [\mathbf{R}^T(x) \quad \mathbf{p}^T(x)] \mathbf{G}^{-1} \mathbf{u}_s = \Phi^R(x) \mathbf{u}_s, \quad (7)$$

where the shape function $\Phi(x)$ is defined by

$$\Phi(x) = [\phi_1(x), \phi_2(x), \dots, \phi_n(x)] = [\mathbf{R}^T(x) \quad \mathbf{p}^T(x)] \mathbf{G}^{-1}. \quad (8)$$

And $\mathbf{R}^T(x) = [R_1(x), R_2(x), \dots, R_n(x)]$, $\mathbf{p}^T(x) = [1, x, y]$. It can be found from the above discussion that RPIM passes through the nodal values. Therefore, RPIM shape functions given in (8) satisfy the Kronecker delta condition. Thus,

$$\phi_i(x_j) = \delta_{ij} = \begin{cases} 1, & i = j, \\ 0, & i \neq j. \end{cases} \quad (9)$$

3. Variational Form of 2D Plane Problem

Consider the 2D problem of the deformation of a linear elastic medium from an undeformed domain Ω , enclosed by Γ :

$$\nabla \cdot \boldsymbol{\sigma} + \mathbf{b} = 0, \quad \text{in } \Omega, \quad (10)$$

where $\boldsymbol{\sigma}$ is the stress tensor corresponding to the displacement field \mathbf{u} and \mathbf{b} is a body force vector, and boundary conditions are as follows:

$$\begin{aligned} \boldsymbol{\sigma} \cdot \mathbf{n} &= \bar{\mathbf{t}} \quad \text{on } \Gamma_t, \\ \mathbf{u} &= \bar{\mathbf{u}} \quad \text{on } \Gamma_u, \end{aligned} \quad (11)$$

in which $\bar{\mathbf{t}}$ and $\bar{\mathbf{u}}$ are prescribed tractions and displacements, respectively, on the traction boundary Γ_t and on the displacement boundary Γ_u and \mathbf{n} is the unit outward normal matrix to the boundary Γ_t .

Using the standard principle of minimum potential energy for (10)-(11), that is, to find $\mathbf{u} \in (H^1(\omega))^3$ such that

$$\Pi = \frac{1}{2} \int_{\Omega} \boldsymbol{\varepsilon}^T \mathbf{D} \boldsymbol{\varepsilon} d\Omega - \int_{\Omega} \mathbf{u}^T \cdot \mathbf{b} d\Omega - \int_{\Gamma_t} \mathbf{u}^T \cdot \bar{\mathbf{t}} d\Gamma \quad (12)$$

is stationary, where $H^1(\omega)$ denotes the Sobolev space of order i , $\boldsymbol{\varepsilon}$ and $\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon}$ are strain-stress vectors, and \mathbf{D} is the strain-stress matrix. RPIM equation (1) is used to approximate the displacements in the Galerkin procedure. Then we can get

$$\mathbf{u} = \begin{Bmatrix} u_x \\ u_y \end{Bmatrix} = \sum_{i=1}^k \begin{bmatrix} \phi_i & 0 \\ 0 & \phi_i \end{bmatrix} \begin{Bmatrix} u_x^i \\ u_y^i \end{Bmatrix} = \sum_{i=1}^k \Phi_i \mathbf{u}_i. \quad (13)$$

Substituting (13) into (12) leads to the following total potential energy in the matrix form:

$$\Pi = \frac{1}{2} \mathbf{u}^T \cdot \mathbf{K} \cdot \mathbf{u} - \mathbf{u}^T \cdot \mathbf{f}, \quad (14)$$

and invoking $\delta\Pi = 0$ results in the following linear system of \mathbf{u} :

$$\mathbf{K}\mathbf{u} = \mathbf{f}, \quad (15)$$

in which the stiffness matrix \mathbf{K} is built from 2×2 matrices \mathbf{K}_{ij} and the right-hand side vector \mathbf{f} is built from the 2×1 vectors \mathbf{f}_i . These matrices and vectors are defined by

$$\begin{aligned} \mathbf{K}_{ij} &= \int_{\Omega} \mathbf{B}_i^T \mathbf{D} \mathbf{B}_j d\Omega, \quad i, j = 1, 2, \dots, k, \\ \mathbf{f}_i &= \int_{\Omega} \phi_i \mathbf{b} d\Omega + \int_{\Gamma_t} \phi_i \bar{\mathbf{t}} d\Gamma, \quad i = 1, 2, \dots, k, \end{aligned} \quad (16)$$

where

$$\mathbf{B}_i = \begin{bmatrix} \phi_{i,x} & 0 \\ 0 & \phi_{i,y} \\ \phi_{i,y} & \phi_{i,x} \end{bmatrix},$$

$$\mathbf{D} = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \quad (\text{for plane stress problem}). \quad (17)$$

Whether the RPIM method or other meshless methods based on global weak form, background cells are necessary to obtain the numerical integration of (16). Different from the finite element method (FEM), which uses the same nodes for both interpolation and numerical integration, background cells in meshless methods are independent of the interpolations. In this paper, quadrilateral cells and Gauss quadrature are used for the numerical integration.

4. Reconstructing Gauss Domains Methods

A weak form, in contrast to a strong form (collocation method in general), requires weaker consistency on the assumed field functions. The consistency requirement on the assumed functions for field variables is very different from the strong form. For a $2k$ th-order differential governing system equation, the strong formulation requires a consistency of the $2k$ th order, while the weak formulation requires a consistency of only the k th order. But the meshfree method usually uses the integral representation of field variable functions for solving strong form system equations and the numerical integration is extremely time-consuming. Numerical accuracy mainly depends on the number of Gauss points in the corresponding domain; the more the Gauss points, the better the results in general. Thus, one of the most time-consuming steps in the meshless method is the construction of shape functions, since for every point of interest a linear system should be computed. As discussed in Section 2, using radial basis functions to construct shape functions, an $n \times n$ linear system should be computed for every computational point. If m monomials are added, an $(n+m) \times (n+m)$ linear system should be solved. This is very time-consuming especially for meshless methods based on weak form, where a large number of integration points are used. In this section, we propose the researching Gauss points (RGP) and reconstructing Gauss domains (RGD) methods, which are together named reordering Gauss domains (RG) methods and can partly reduce the computation cost for the meshless methods compared with the RPIM.

4.1. Researching Gauss Points (RGP) with the Same Nodes. In the RPIM approximation, shape function consists of two parts: $[\mathbf{R}^T(x) \mathbf{p}^T(x)]$ and \mathbf{G}^{-1} (8). The time consumption of $[\mathbf{R}^T(x) \mathbf{p}^T(x)]$ is less than that of \mathbf{G}^{-1} , because the computational complexities are $o(n+m)$ and $o((n+m)^3)$, respectively.

Every Gauss point has its own $[\mathbf{R}^T(x) \mathbf{p}^T(x)]$ and is different from the rest, but it may have the same \mathbf{G}^{-1} as the other. Thus, we need a storage place, containing the public nodes and the Gauss points' information. The data is obtained by searching the Gauss points. Thus, the method is called researching Gauss points (RGP) method.

The red area represents the nodes set (involved red nodes \circ) and the blue area represents the Gauss points set (involved blue gauss points $*$) in Figure 1(a). It should be pointed out that one-to-one mapping exists between the Gauss points sets and the nodes points sets (can be verified by logical

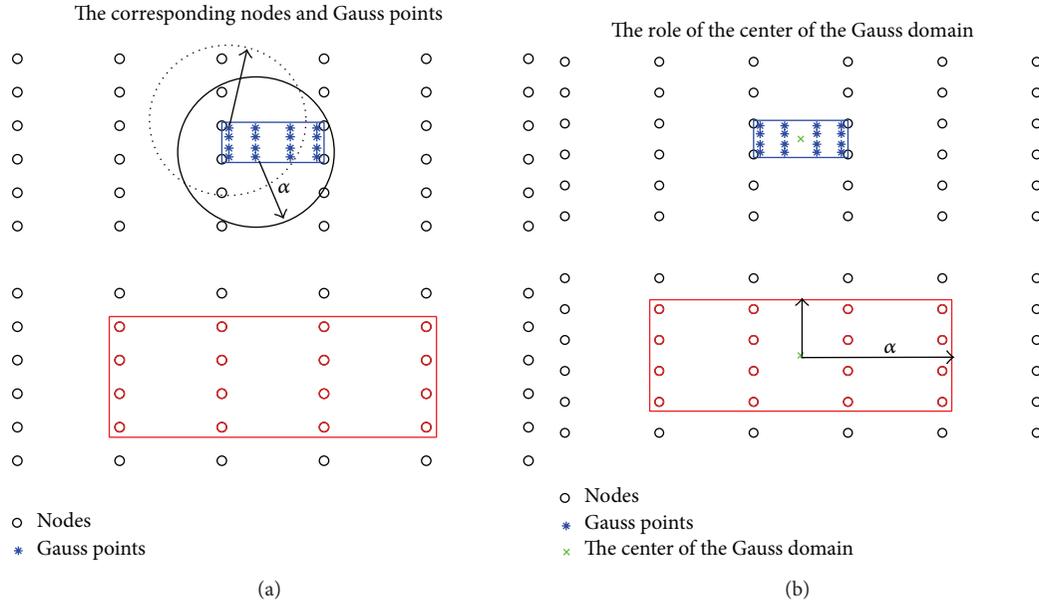


FIGURE 1: The distribution of the nodes and the Gauss points: find the 1-1 mapping (a); assign the 1-1 mapping (b).

deduction). Thus, the key work is how to get the relationship. First of all, we construct $N_n * N_g$ matrix $S = [s_{ij}]$, where N_n, N_g denote the number of the nodes and the number of the Gauss points, respectively. The element $S = [s_{ij}]$ is δ function:

$$s_{ij} = \begin{cases} 1 & \text{the } i\text{th nodes locate in the support domain} \\ & \text{of the } j\text{th gauss point;} \\ 0 & \text{otherwise.} \end{cases} \quad (18)$$

Thus, S is written as

$$S = \begin{bmatrix} \dots & \dots & \dots & \dots & \dots \\ 1 & 1 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 1 & 1 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \end{bmatrix}_{N_n * N_g} \quad (19)$$

If the j th row elements are the same as the k th row elements, we confirm the j th and k th Gauss points have the same basis functions $R(x_j) R(x_k)$. It is a rule, which gives one-to-one mappings between the two types of sets by comparing the rows of S . Some codes (isequal.m, unique.m, etc.) could be used to find these mappings in MATLAB. The process of one-to-one mapping is also called the searching program distinguishing the searching nodes program in advance. For the sake of simplicity, the RGP method is divided into two parts: Part I (the researching program) and the rest.

4.2. *Reconstructing Gauss Domains (RGD)*. The searching program saves the time consumption in Gauss integration, but it needs extra time and storage for the searching process. With the rapid increase of the nodes and the Gauss points, the searching process costs also grow exponentially. Then,

we present the reconstructing Gauss domains (RGD) method without the searching process. In the fixed Gauss domain, all Gauss points are mandatorily contacted with some neighbouring nodes. These nodes lie in the support domain of the Gauss domain's centroid. That is, every Gauss domain has its own basis functions or G^{-1} corresponding to the centroid.

A center point (\times) of the Gauss domain is added in Figure 1(b) compared with Figure 1(a). Without the searching process, we mandatorily consider that the center point is seen by not only the barycentre of the Gauss domain, but also the barycentre of the domain involving some nodes. These nodes locate in the support domain with α on the barycentre point. Thus, the one-to-one mappings are ascertained by the corresponding barycentre points.

5. Numerical Experiments

In this section, several numerical examples are selected to demonstrate the applicability of the RG meshless methods. The numerical results for these examples are compared with the analytical solutions and the RPIM solution. Square support domains are used for calculations in the present paper. d_c stands for the average node distance. The MQ radial basis function and the linear polynomial are used to construct shape functions. All runs are performed in MATLAB 7.0 on an Intel Pentium 4 (2 GB RAM) Windows XP system.

5.1. *Patch Test*. The first numerical example is the standard patch test shown in Figure 2. The patch test consists of thirteen nodes including five interior nodes. A 2×2 rectangular background mesh is used for numerical integration and 4×4 Gauss points are used in each mesh. In this patch test, the displacements are prescribed on all outside boundaries by a linear function of x and y on the patches of the dimension. The parameters are taken as $E = 1.0$ and $\nu = 0.3$. The linear

TABLE 1: Results at interior nodes located irregularly in the 13 nodes' patch.

Internal node	Coordinates		Exact	RGP/RPIM	RGD
5	(1.0, 1.0)	(u_x, u_y)	(0.600, 0.600)	(0.601, 0.599)	(0.601, 0.599)
		$(\sigma_x, \sigma_y, \tau_{xy})$	(0.857, 0.857, 0)	(0.853, 0.853, 0)	(0.853, 0.853, 0)
10	(0.65, 1.0)	(u_x, u_y)	(0.390, 0.600)	(0.391, 0.599)	(0.391, 0.599)
		$(\sigma_x, \sigma_y, \tau_{xy})$	(0.857, 0.857, 0)	(0.861, 0.853, 0.002)	(0.861, 0.853, 0.002)
11	(0.70, 1.5)	(u_x, u_y)	(0.420, 0.900)	(0.422, 0.897)	(0.421, 0.897)
		$(\sigma_x, \sigma_y, \tau_{xy})$	(0.857, 0.857, 0)	(0.856, 0.855, 0.001)	(0.856, 0.855, 0.001)
12	(1.3, 1.2)	(u_x, u_y)	(0.780, 0.720)	(0.779, 0.718)	(0.779, 0.718)
		$(\sigma_x, \sigma_y, \tau_{xy})$	(0.857, 0.857, 0)	(0.846, 0.852, -0.002)	(0.846, 0.852, -0.002)
13	(1.2, 0.6)	(u_x, u_y)	(0.720, 0.360)	(0.720, 0.359)	(0.720, 0.359)
		$(\sigma_x, \sigma_y, \tau_{xy})$	(0.857, 0.857, 0)	(0.854, 0.856, 0)	(0.854, 0.856, 0)

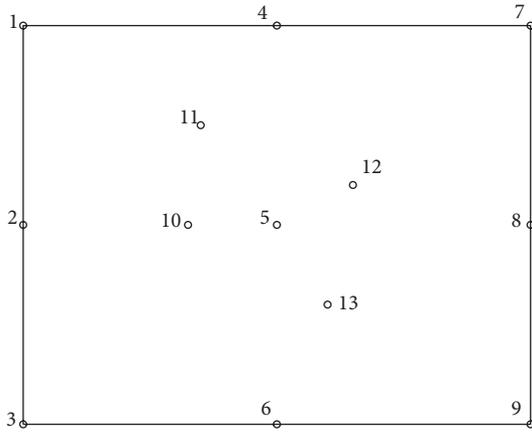


FIGURE 2: A patch with 13 nodes.

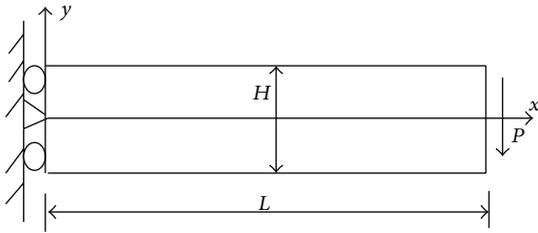


FIGURE 3: Cantilever beam.

displacement functions are $u_x = 0.6x$ and $u_y = 0.6y$. RG methods of the patch test require that the displacement of any interior node be given by the same linear function and that the strains and stresses be constant in the patch. The RG methods pass the patch test when linear polynomials are added ($m = 3$). However, when polynomial term is not included ($m = 0$), the patch test does not easily pass. The same is for the RPIM method; see [21]. Computed results at interior nodes for the RG methods and the RPIM method and the exact results are listed in Table 1.

5.2. Cantilevered Beam. The second example is a cantilever beam problem; see Figure 3. Consider a beam of length L and height H subjected to traction at the free end. The

beam has a unit thickness, and thus a plane stress problem is considered here. The closed-form solution is available for parabolic traction of force P :

$$u_x = \frac{Py}{6EI} \left[(6L - 3x)x + (2 + \nu) \left(y^2 - \frac{H^2}{4} \right) \right],$$

$$u_y = -\frac{P}{6EI} \left[3\nu y^2 (L - x) + (4 + 5\nu) \frac{H^2 x}{4} + (3L - x)x^2 \right], \quad (20)$$

where the moment of inertia I of the beam is given by $I = H^3/12$. The stresses corresponding to the above displacements are

$$\sigma_x = \frac{P(L - x)y}{I}, \quad \tau_{xy} = -\frac{P}{2I} \left(\frac{H^2}{4} - y^2 \right), \quad \sigma_y = 0. \quad (21)$$

The beam parameters are taken as $E = 3.0 \times 10^7$, $\nu = 0.3$, $H = 12$, $L = 48$, and $P = 1000$ in computation.

To evaluate effects of various parameters, we use the two-norm relative errors r_u and r_σ for displacement and stress, respectively. They are defined by

$$r_u = \frac{\|\mathbf{u} - \hat{\mathbf{u}}\|_2}{\|\mathbf{u}\|_2}, \quad r_\sigma = \frac{\|\hat{\boldsymbol{\sigma}} - \boldsymbol{\sigma}\|_2}{\|\boldsymbol{\sigma}\|_2}, \quad (22)$$

where $\hat{\mathbf{u}}$ and \mathbf{u} are the approximate and exact solution of displacements and $\hat{\boldsymbol{\sigma}}$ and $\boldsymbol{\sigma}$ are the approximate and exact value of stresses.

5.2.1. Effect of Irregular Node Distribution. Node distributions with 325 irregular nodes and 481 irregular nodes are shown in Figure 4. A background mesh is necessary to obtain the numerical integrations of (15). This mesh is independent of nodes for interpolations, while FEM uses the same nodes for both interpolation and numerical integration. For this problem, 10×6 and 15×6 background cells are used for the 325 irregular nodes' problem and 481 irregular nodes' problem, respectively. In each cell, a 4×4 Gauss quadrature is used to evaluate the stiffness matrix. Figure 5(a) shows a comparison of the analytical solution and numerical solution

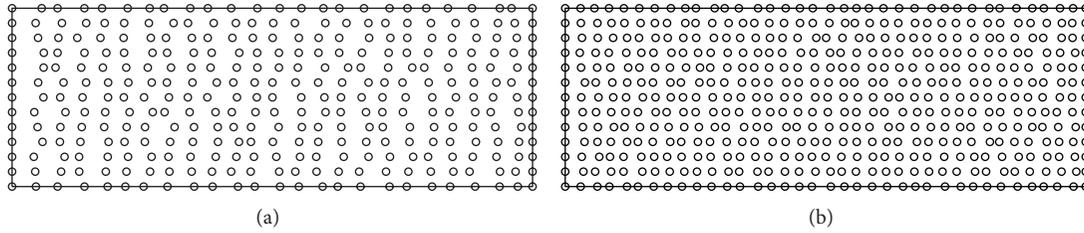


FIGURE 4: Meshless models with irregular data points: 325 nodes (a), 481 nodes (b).

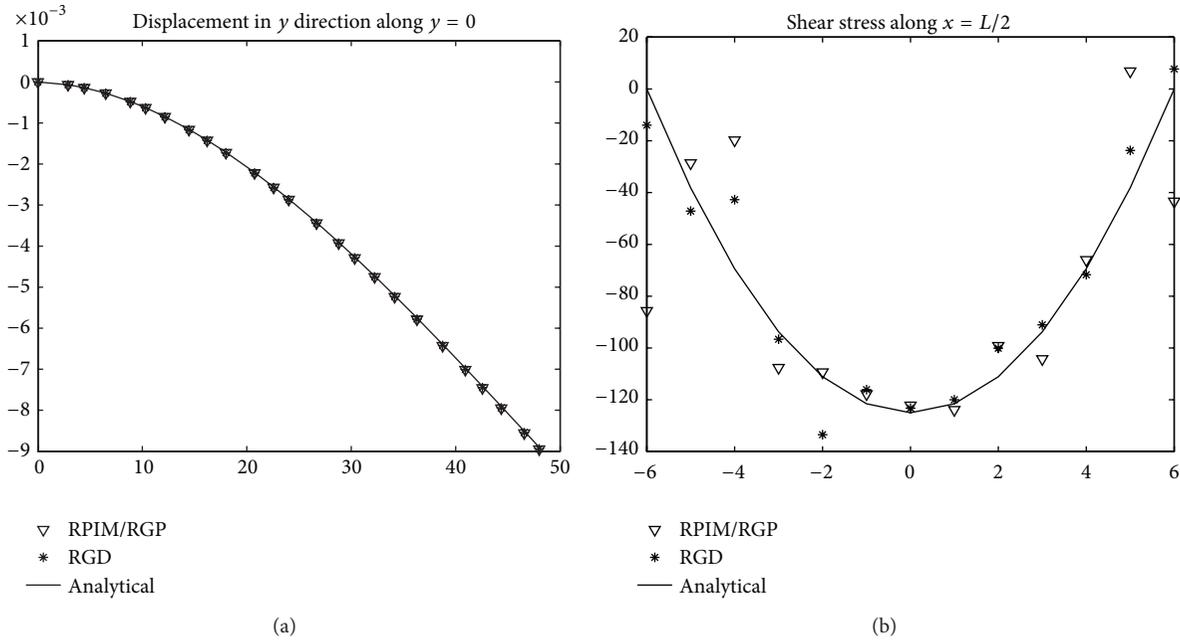


FIGURE 5: Results for 325 irregular nodes: (a) displacement and (b) shear stress.

of u_y along $y = 0$ for this problem with 325 irregular nodes. The plot shows an excellent agreement between the analytical and numerical results for each method when an irregular node distribution is used. Figure 5(b) is the sectional distribution of shear stress τ_{xy} along the $x = 24$ section for this problem with 325 irregular nodes. The closed-form solution is also plotted for comparison.

Computational results including error and computational cost are listed in Table 2 for the cantilevered beam problem with 325 irregular nodes and 481 irregular nodes. From Figure 5 and Table 2, we can find that the RG methods have better accuracy and are less time-consuming than the RPIM method.

5.2.2. Effect of the Size of Support Domain of RGD. As discussed in Section 4.2, the centroid of every Gauss cell has a support domain. The range of the support domain α is an important parameter. The 325 regular nodes' distribution (25×13 : 25 nodes in the x direction and 13 nodes in the y direction) is used to study the effect of the size of the support domain for the RG methods.

In Table 3, we list the numerical results for the RGD method with different α . To compare with the RPIM method,

TABLE 2: Computational results for irregular node distribution.

Irregular nodes	RPIM	RGP (Part I + the rest)	RGD
325			
r_u		0.0046	0.0013
r_σ		0.0793	0.0582
CPU	3.034	2.917 = 1.424 + 1.493	1.162
481			
r_u		0.0080	0.0019
r_σ		0.0824	0.0472
CPU	4.521	6.468 = 4.354 + 2.114	1.668

we also list the computational results for the RPIM method with different radiuses of the support domain. From Table 3, we can see that the RGD method has better steadiness, computational efficiency, and computational accuracy than the RPIM method. Particularly, with $\alpha = 4.5$, the elapsed CPU time is reduced to 12.7% (see the data in the boxes) of the previous one. This shows that the size of the support

TABLE 3: Effect of the size of support domain of RGD (Example 5.2).

α	RGD			RPIM and RGP			
	r_u	r_σ	CPU	r_u	r_σ	RPIM CPU	RGP CPU (Part I + the rest)
2.5	0.0513	0.2033	0.413	0.0074	0.0522	1.808	2.151 = 1.023 + 1.128
3.0	0.0129	0.1000	0.646	0.0031	0.0620	3.003	2.537 = 1.131 + 1.406
3.5	0.0109	0.0945	0.733	0.0031	0.0305	4.752	3.277 = 1.798 + 1.479
4.0	0.0006	0.0285	1.182	0.0002	0.0533	7.135	4.071 = 2.608 + 1.463
4.5	0.0004	0.0274	1.336	0.0021	0.0191	10.303	5.195 = 3.720 + 1.475
5.0	0.0002	0.0202	2.112	0.0004	0.0294	14.599	6.451 = 4.911 + 1.540

TABLE 4: Convergence study on regular nodes for Example 5.2.

Regular nodes	13×4	25×7	37×10	49×13	61×16	
RGD						
r_u		0.0384	0.0063	0.0035	0.0025	0.0015
r_σ		0.1891	0.0399	0.0676	0.0668	0.0182
CPU		0.440	0.839	1.972	2.730	5.420
RGP/RPIM						
r_u		0.0358	0.0028	0.0024	0.0196	0.0240
r_σ		0.2057	0.0522	0.0559	0.1434	0.1738
RPIM CPU		1.864	2.450	5.637	5.981	12.302
RGP CPU (Part I + the rest)		0.219 + 0.350	1.404 + 0.168	5.943 + 2.147	9.975 + 3.049	59.036 + 6.468

TABLE 5: Convergence study on irregular nodes for Example 5.2.

Irregular nodes	52	175	370	637	976	
RGD						
r_u		0.0413	0.0062	0.0021	0.0055	0.0175
r_σ		0.1836	0.0399	0.0578	0.0832	0.1138
CPU		0.419	0.831	2.055	2.627	4.387
RGP/RPIM						
r_u		0.0386	0.0003	0.0020	0.0038	0.0003
r_σ		0.1758	0.0553	0.0422	0.0610	0.0108
RPIM CPU		1.637	2.486	5.600	6.178	12.814
RGP CPU (Part I + the rest)		0.203 + 0.332	0.832 + 0.734	5.989 + 1.667	10.018 + 2.480	58.798 + 3.672

domain has increased and its CPU time takes up a larger proportion of the entire time, in which the consumption surged from 48% to 76% of the entire cost. The searching process has become the major expenditure of the entire program. From Table 3, we can find that the RGD method has a stable convergency with the increase of the α , but it does not appear in the RPIM. In this paper, we take the suitable α to be 4.

5.2.3. *Convergence Study.* For convergence studies, five different regular node distributions with 52 (13×4 : 13 nodes in the x direction and 4 nodes in the y direction) nodes, 175 (25×7) nodes, 370 (37×10) nodes, 637 (49×13) nodes, and 976 (61×16) nodes and five different irregular node distributions are considered. The size of the support domain is taken as $\alpha = 4$. The computational results are listed in Tables 4 and 5. In Tables 4 and 5, we also list the computational results

of the RPIM method. The convergence curves with different node distributions are plotted in Figure 5. In Figure 5, h is the maximum size of node arrangement. From Tables 4 and 5 and Figure 5 we can see that the present method possesses good convergence and less computational times than the RPIM method in general. However, the searching process consumption overtakes 88% of the entire process in the 976 regular/irregular nodes, which become cumbersome. The oscillatory behavior of the convergence curve of RGD method exists in regular node, and oscillatory behaviors appear in the two node distributions of the RPIM method. Figure 5 shows that RGD method is steadier than the RPIM method. The reason of oscillatory appearance is that the accuracy of most meshless methods is closely related to the integration scheme, the number of Gauss points, the radiuses of the support domain and influence domain, and so on. Thus, it is a difficult task to get the best accuracy for all cases.

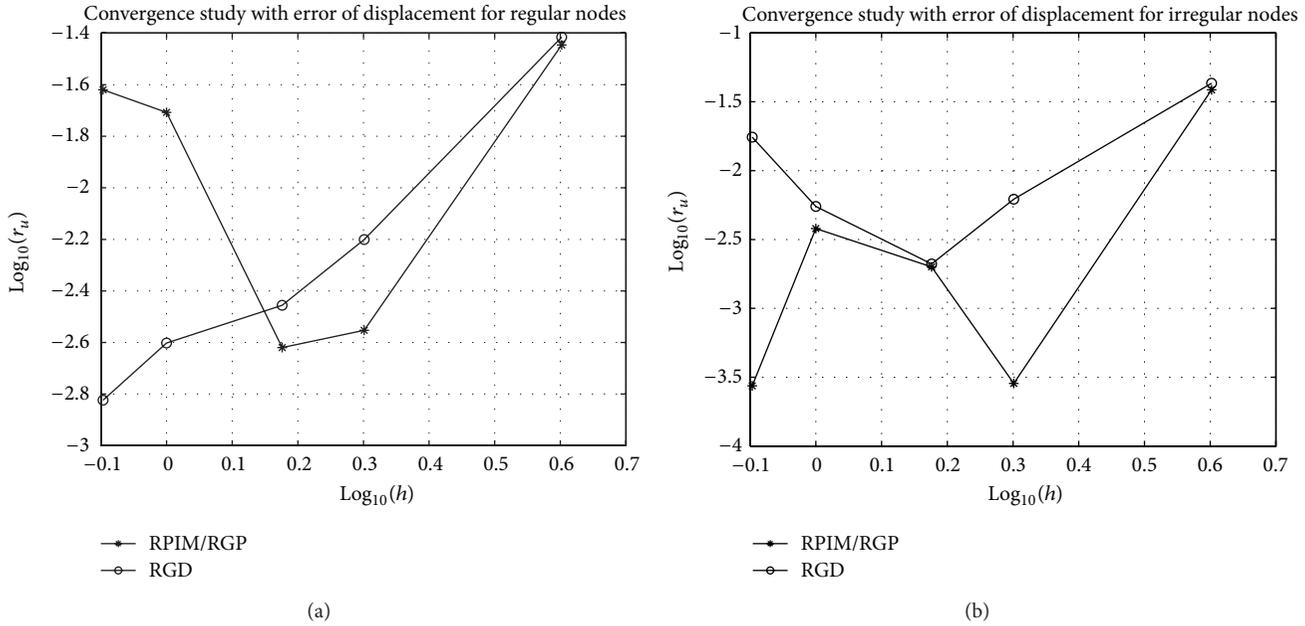


FIGURE 6: Convergence study with error of displacement: regular nodes (a), irregular nodes (b).

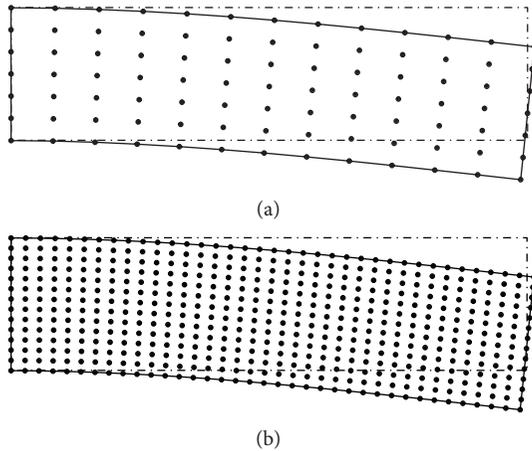


FIGURE 7: Displacement 13×7 (a) and 36×13 (b) regular nodes distribution (scaled 400).

5.2.4. Discussion on the Computational Results. To further discuss the effectiveness of the RG methods, we discuss the computational results of the cantilever beam problem in this subsection. Figure 6 shows a comparison of analytical solution and the present RGD numerical solution for the beam deflection. Two nodal distributions, 91 (13×7) regular nodes and 481 (37×13) regular nodes, are used. The stress results are also obtained. Figures 8 and 9 illustrate the analytical solution and the RG methods solutions for the stress σ_x and the shear stress τ_{xy} of the beam using 481 regular nodes, respectively. Figures 6–8 show the RG methods and RPIM have good performance for the stress σ_x and poor performance for the stress τ_{xy} . But in terms of time, the RGD method is better than the RGP method and RPIM method

(Tables 4 and 5 and Figure 6). Thus, the RGP method is not considered in the next section.

5.2.5. A Special Case of the RD Methods. As a special case of the RD methods, the vertices of the Gauss domains coincide with the nodes in regular nodes distribution; see Figure 11. Figure 11 shows that a 5×5 uniform grid and a 4×4 Gauss quadrature in each Gauss domain are used. The Gauss points $*$ in the Gauss domain have the same nodes \circ (red nodes) with $\alpha = 2$. We consider that the process of constructing shape function has three parts: the $[R(x) \ P(x)]$, solving the inverse of \mathbf{G} , and computing $[R(x) \ P(x)] * \mathbf{G}^{-1}$. These computational complexities are $O((m+n))$, $O((m+n)^3)$, and $O((m+n)^2)$, respectively. Table 6 shows the complexity of the integration for a Gauss domain with 16 Gauss points.

The efficiency ratio is defined as $\text{CPU}_{\text{RPIM}}/\text{CPU}_{\text{RG}}$. To discuss the efficiency ratio, two nodal distributions, 325 (25×13) regular nodes and 629 (37×17) regular nodes, are used. Table 7 shows that the efficiency ratios are 11.832 and 13.362, respectively. The efficiency ratios are also expressed as

efficiency ratios

$$\begin{aligned}
 &= \left(\text{NG} * \left(16 * O((m+n)^3) + 16 * O((m+n)^2) \right. \right. \\
 &\quad \left. \left. + 16 * O(m+n) \right) \right. \\
 &\quad \left. + \text{preprocess} + \text{postprocess} \right) \\
 &\times \left(\text{NG} * \left(O((m+n)^3) + 16 * O((m+n)^2) \right. \right. \\
 &\quad \left. \left. + 16 * O(m+n) \right) \right. \\
 &\quad \left. + \text{preprocess} + \text{postprocess} \right)^{-1}.
 \end{aligned} \tag{23}$$

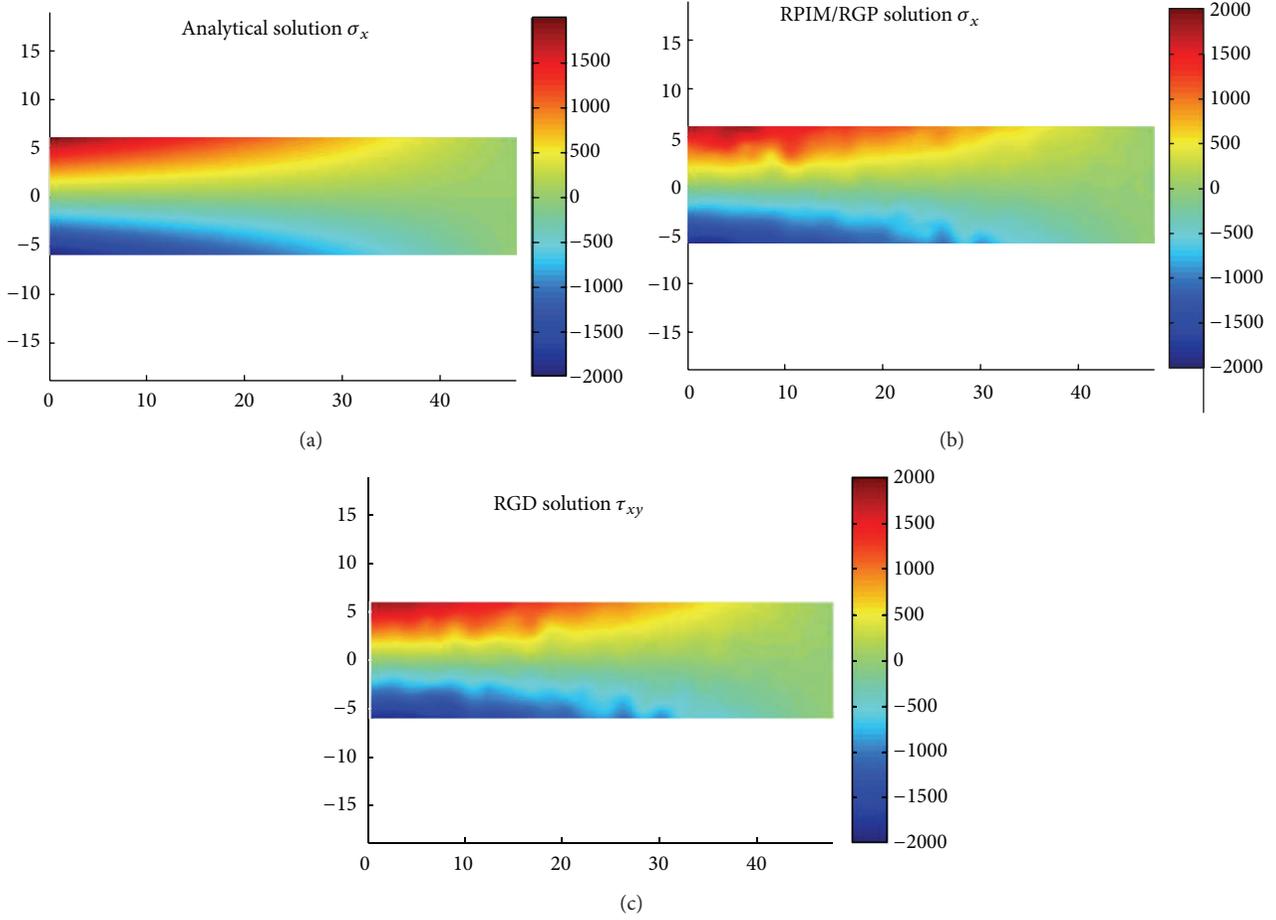


FIGURE 8: Stress σ_x : analytical results (a), RGP numerical results (b), and RGD numerical results (c).

To refine the nodes domain, the NG will be a large number. The elapsed CPU times of compute shape functions is far than the rest. Thus,

$$\begin{aligned}
 \text{efficiency ratios} &\rightarrow (16 * O((m+n)^3)) \\
 &+ 16 * O((m+n)^2) \\
 &+ 16 * O(m+n) \\
 &\times (O((m+n)^3) + 16 * O((m+n)^2) \\
 &+ 16 * O(m+n))^{-1} \\
 &\rightarrow 16.
 \end{aligned}
 \tag{24}$$

The formula of the efficiency ratios fully explains that the ratio is 11.832 for 325 (25×13) regular nodes (NG = 288) and 13.362 for 629 (37×17) regular nodes (NG = 576). Numerical results show that the rates are 11.832 and 13.362 which gradually close to the theory upper 16 with the increase

of NG (the number of the Gauss domains). If we want to improve the accuracy of numerical results with refining the nodes domain, the number of the Gauss points should be added and the efficiency ratios will be improved further.

5.3. Plate with a Central Circular Hole. Consider now a plate with a central circular hole subjected to a unidirectional tensile load of 1.0 in the x direction (see Figure 11(a)). This is a typical plane stress problem.

The closed-form solution of stresses is

$$\begin{aligned}
 \sigma_x &= 1 - \frac{a^2}{r^2} \left[\frac{3}{2} \cos 2\theta + \cos 4\theta \right] + \frac{3a^4}{2r^4} \cos 4\theta, \\
 \sigma_y &= -\frac{a^2}{r^2} \left[\frac{1}{2} \cos 2\theta - \cos 4\theta \right] - \frac{3a^4}{2r^4} \cos 4\theta, \\
 \tau_{xy} &= -\frac{a^2}{r^2} \left[\frac{1}{2} \sin 2\theta + \sin 4\theta \right] + \frac{3a^4}{2r^4} \sin 4\theta,
 \end{aligned}
 \tag{25}$$

where (r, θ) are polar coordinates, θ is the measured counterclockwise from the positive x -axis, and a is the radius of

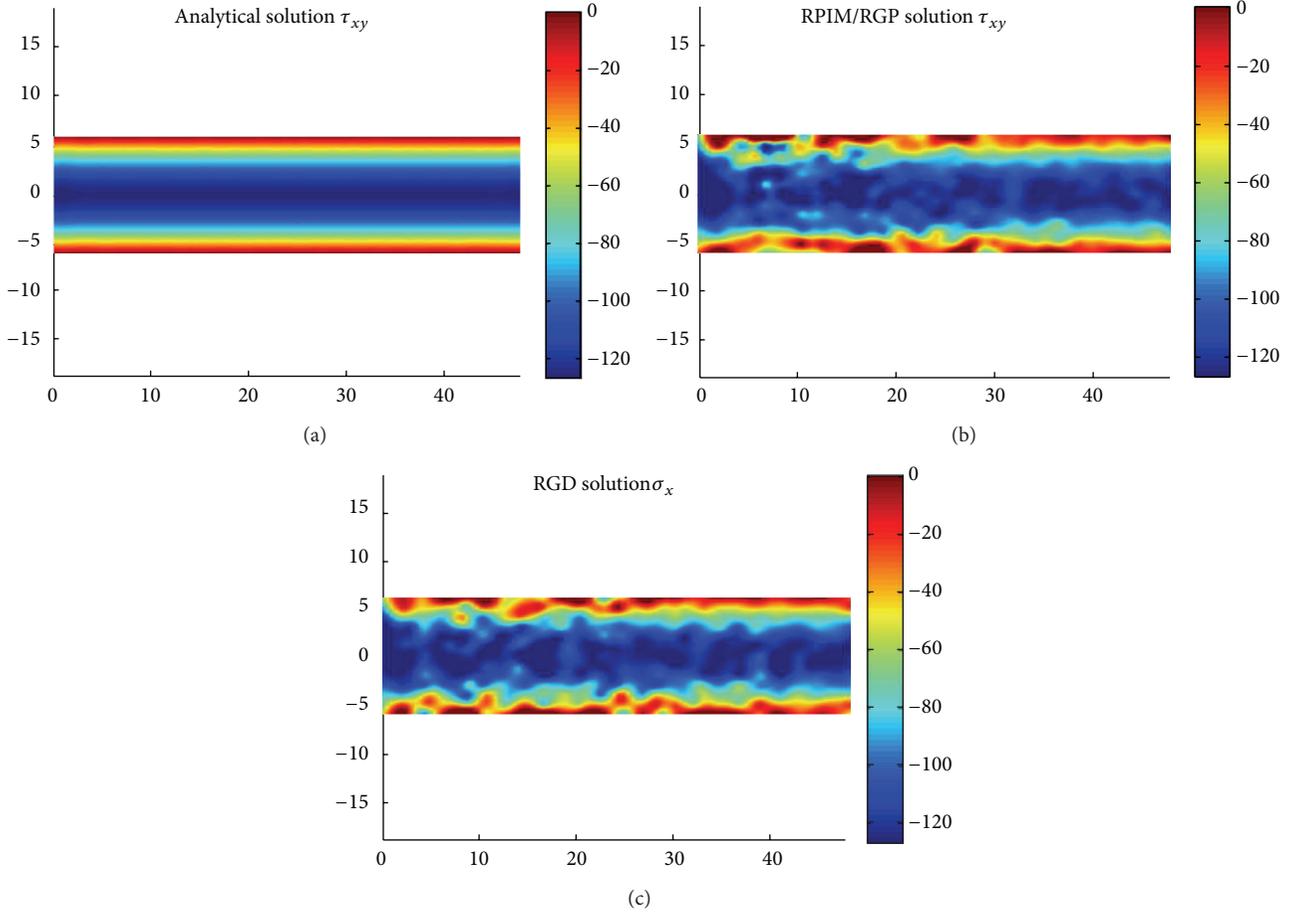


FIGURE 9: Shear stress τ_{xy} : analytical results (a), RGP numerical results (b), and RGD numerical results (c).

the hole. The corresponding displacements, in the plain stress case, are given by

$$\begin{aligned}
 u_r &= \frac{1}{4G} \left\{ r \left[\frac{\kappa - 1}{2} + \cos 2\theta \right] + \frac{a^2}{r} [1 + (1 + \kappa) \cos 2\theta] \right. \\
 &\quad \left. - \frac{a^4}{r^3} \cos 2\theta \right\}, \\
 u_\theta &= \frac{1}{4G} \left[(1 - \kappa) \frac{a^2}{r} - r - \frac{a^4}{r^3} \right] \sin 2\theta,
 \end{aligned}
 \tag{26}$$

where $G = E/(2(1 + \nu))$ and $\kappa = (3 - \nu)/(1 + \nu)$. The material properties are $E = 1.0 \times 10^3$ and $\nu = 0.3$.

By taking advantage of its symmetry, only a quarter of the model is considered in the analysis (see Figure 11(b)). Symmetry conditions are imposed on the left and the bottom edges, and the inner boundary of the hole is traction-free. These boundary conditions include (i) essential boundary conditions on the bottom and left edges on which displacement \bar{u} is computed from the exact displacement given in (26) and (ii) natural boundary conditions on the right and top edges on which traction \bar{t} is computed from the exact stress given in (25). A typical node distribution with 99 nodes

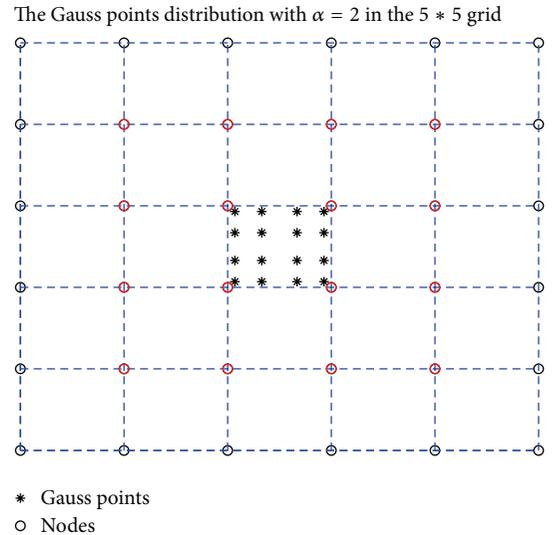


FIGURE 10: The diagram of 5 * 5 domains for the special case.

and the corresponding background mesh constructed for the numerical integration are shown in Figure 12(a). For each background cell, a 4 * 4 Gauss quadrature is employed. It

TABLE 6: The complexity of the integration for a Gauss domain.

Complexity	Part I $O((m+n))$	Part II $O((m+n)^3)$	Part III $O((m+n)^2)$	Sum
RPIM	16 times	16 times	16 times	$16 * O((m+n)^3) + 16 * O((m+n)^2) + 16 * O(m+n)$
RG	16 times	1 time	16 times	$O((m+n)^3) + 16 * O((m+n)^2) + 16 * O(m+n)$

TABLE 7: Effect of the size of support domain for the RG method (Example 4.2).

Schemes	Grid					
	25 * 13			37 * 17		
	r_u	r_σ	CPU	r_u	r_σ	CPU
RG			1.267			2.373
RPIM	4.7796e - 004	0.0204	14.991	7.3559e - 005	0.0054	31.708
Efficiency ratio			11.832			13.362

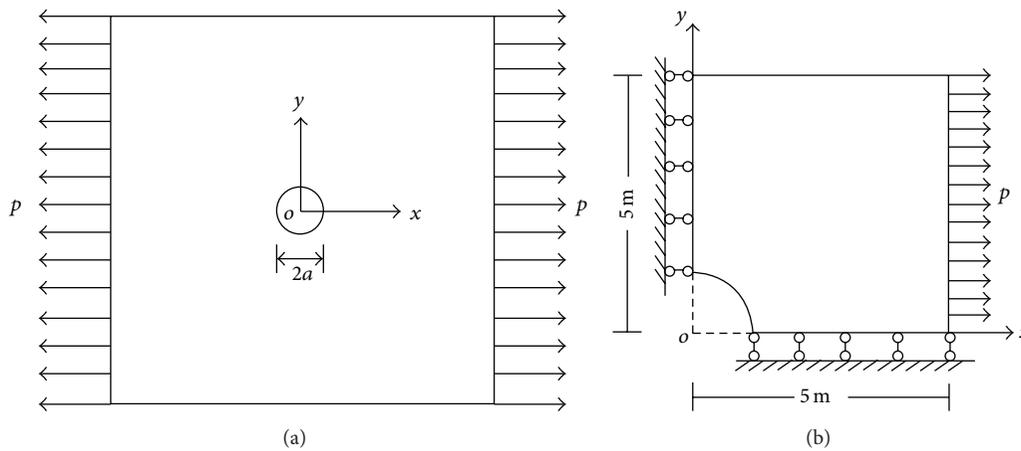


FIGURE 11: Plate with a central circular hole and its model problem: (a) a plate with a hole, (b) a quarter of the plate and boundary conditions.

TABLE 8: Effect of the size of support domain for the RG methods for the 99 nodes (Example 5.3).

α	RGD			RPIM		
	r_u	r_σ	CPU	r_u	r_σ	CPU
1.1	0.0149	0.1101	0.880	0.0219	0.1005	0.968
1.2	0.0097	0.0881	0.963	0.0205	0.1004	1.052
1.5	0.0094	0.0623	1.175	0.0271	0.1526	1.547
1.8	0.0056	0.0886	1.580	0.0133	0.2052	2.383
2.0	0.0054	0.1154	2.043	0.0115	0.2226	3.052
3.0	0.0044	0.0590	4.600	0.0057	0.1500	9.286

TABLE 9: Convergence study for Example 5.3.

Scheme	99 nodes	289 nodes	625 nodes	1089 nodes
RGD				
r_u	0.0149	0.0053	0.0025	0.0016
r_σ	0.1101	0.0534	0.0384	0.0343
CPU	0.880	2.030	3.480	13.516
RPIM				
r_u	0.0219	0.0057	0.0031	0.0022
r_σ	0.1005	0.0541	0.0343	0.0283
CPU	0.968	4.523	17.419	55.530

is also of great interest to further study the effect of the nodal influence domain size and the computational support domain size. It should be noted that the support domain for all nodes is a circle with varying radius. They are chosen such that the support is small for nodes near the hole and bigger for nodes near the edges. Table 8 presents the relative errors and elapsed CPU times with different parameters α for

the 99 nodes. In this case, the computational results show relatively good accuracy and less elapsed CPU time. For convergence studies, four different node distributions with 99 nodes, 289 nodes, 625 nodes, and 1089 nodes are considered in Table 9. The last three node distributions are taken from [23]. Due to singular property for RPIM, α was advised as 1.1 compared with the RGD method. The computational results

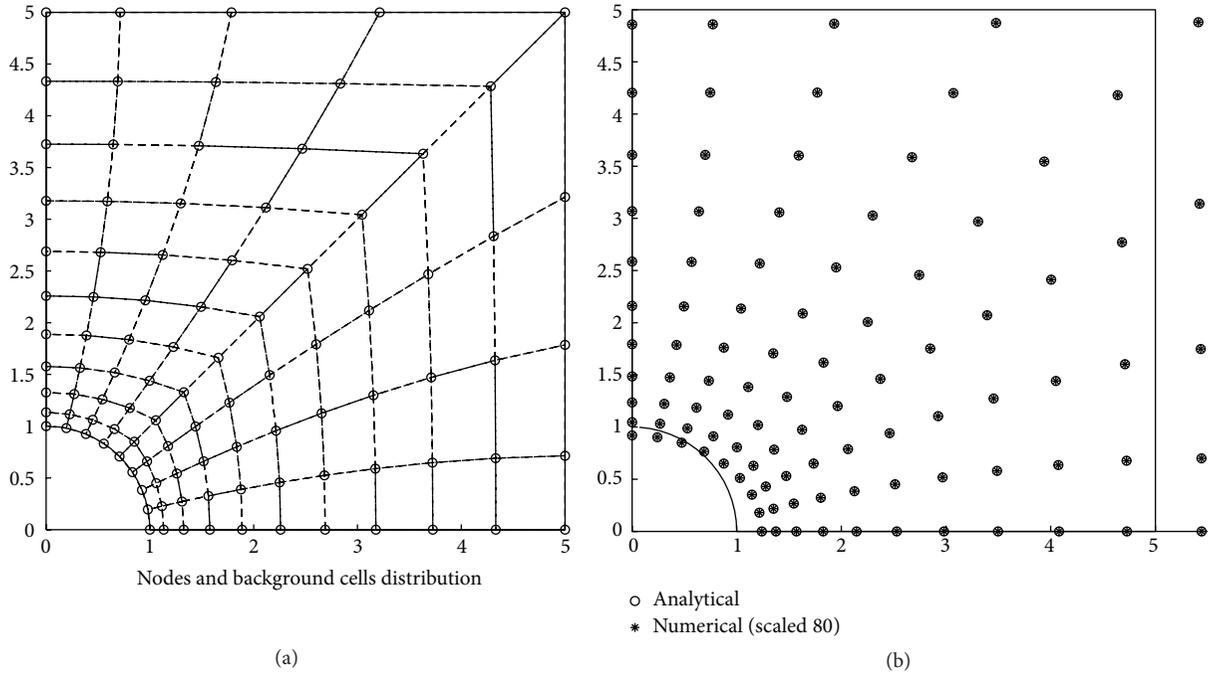


FIGURE 12: Plate with a central circular hole: (a) 99 nodes' distribution, (b) analytical and RGD solution of displacement.

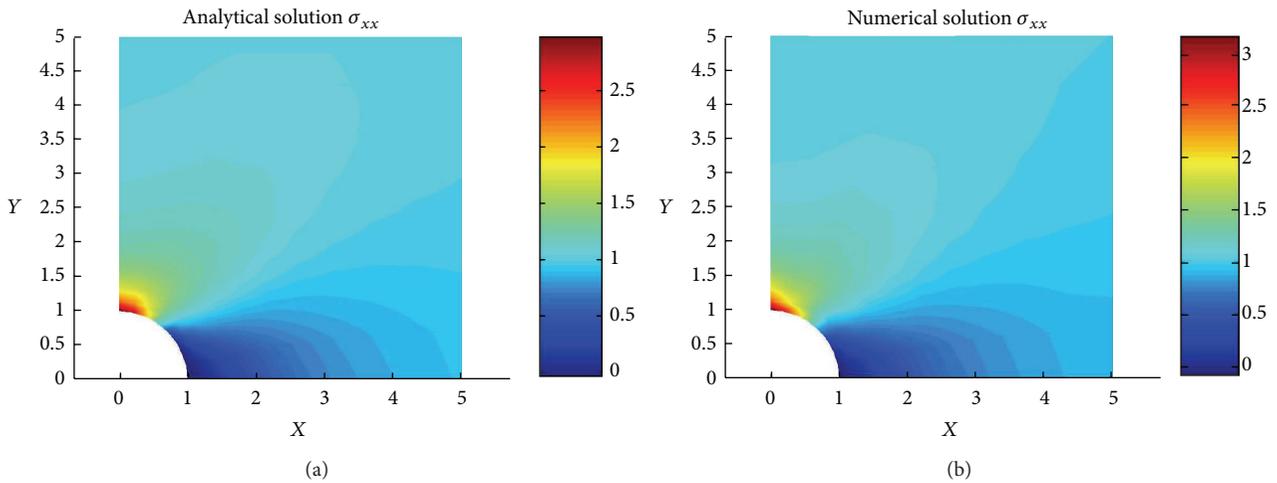


FIGURE 13: Stress σ_x : analytical results (a) and RGD numerical results (b).

are presented in Table 9, which also shows that the RGD method possesses high convergence. The elapsed CPU times of the four node distributions fall 9%, 55%, 80%, and 76%, respectively. The computed displacements and stresses are also shown for 99 nodes' distribution problem in Figures 10–13. The displacements computed by the RGD method at nodes are plotted and compared to the exact solution (see Figure 12(b)). Figures 13 and 14 compare the analytical solution and RGD solution of stress σ_x and shear stress τ_{xy} , respectively. Figure 15(a) compares the stresses σ_x at stations along the left edge of the plate for the RPIM solution, the RGD solution, and the closed-form solution. Figure 15(b) plots the stresses σ_y at stations along the bottom edge of the plate for

these methods. From Figures 11–13, it is shown that the RGD has a higher accuracy compared with the RPIM solution. In particular, the RGD solution of stress σ_x at (0, 5) agrees with the analytical result, but the RPIM result is far away from the analytical result.

6. Conclusions

In this paper, we have studied researching Gauss points and reordering Gauss domain technique (GD methods), where the computational processes are more reasonable. Secondly, this technique is compatible with most of the common RPIM methods, and this aspect is worth studying. Finally,

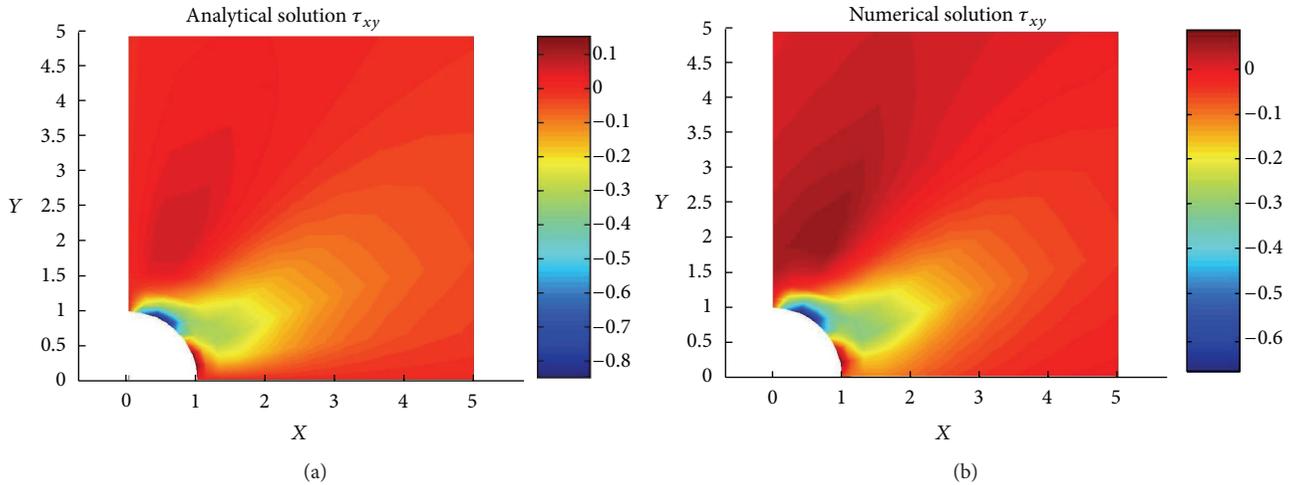


FIGURE 14: Shear stress τ_{xy} : analytical results (a) and RGD numerical results (b).

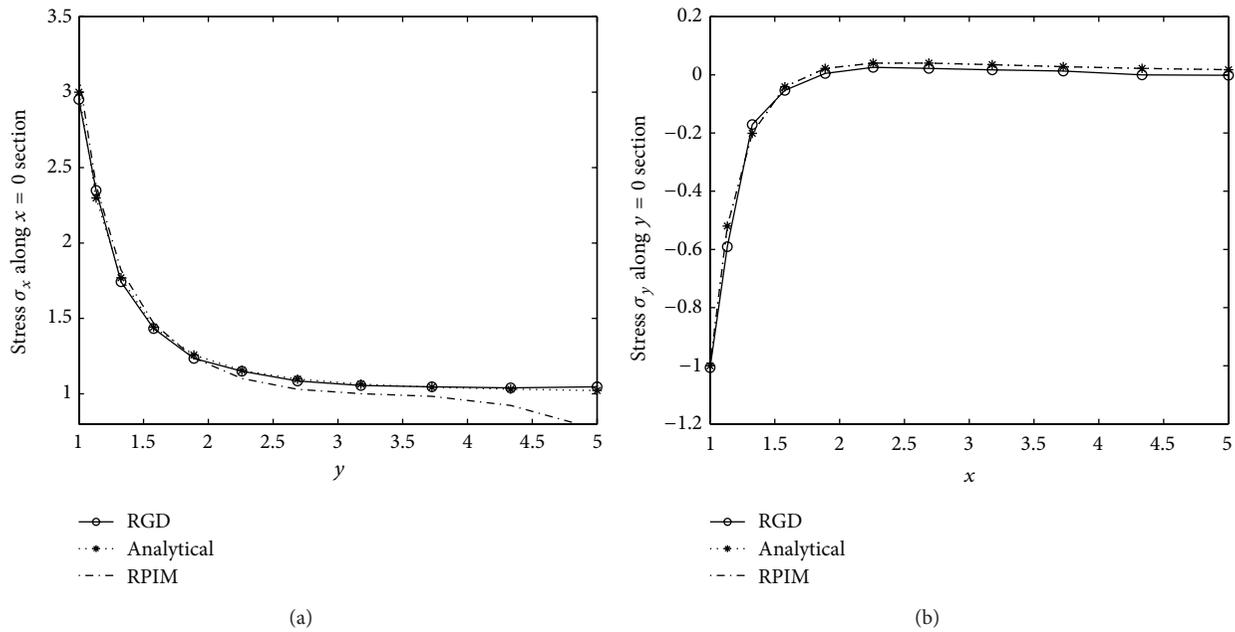


FIGURE 15: Comparisons of RGD and analytical solution: stress σ_x at the stations along the left edge of the plate (a) and stress σ_y at the stations along the bottom edge of the plate (b).

3D problems could be solved with this scheme to save the computation time.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant nos. 11172192 and 11301290. Finally, the authors wish to thank the editors and the

anonymous referees for their hard work which has greatly improved the original paper.

References

- [1] X. Zhang, K. Z. Song, M. W. Lu, and X. Liu, “Meshless methods based on collocation with radial basis functions,” *Computational Mechanics*, vol. 26, no. 4, pp. 333–343, 2000.
- [2] T. Belytschko, Y. Y. Lu, and L. Gu, “Element-free Galerkin methods,” *International Journal for Numerical Methods in Engineering*, vol. 37, no. 2, pp. 229–256, 1994.
- [3] Z. Zhang, P. Zhao, and K. M. Liew, “Improved element-free Galerkin method for two-dimensional potential problems,”

- Engineering Analysis with Boundary Elements*, vol. 33, no. 4, pp. 547–554, 2009.
- [4] Y. Cao, J.-L. Dong, and L.-Q. Yao, “A modification of the moving least-squares approximation in the element-free galerkin method,” *Journal of Applied Mathematics*, vol. 2014, Article ID 528082, 13 pages, 2014.
- [5] F. Z. Louai, N. Naït-Said, and S. Drid, “Implementation of an efficient element-free Galerkin method for electromagnetic computation,” *Engineering Analysis with Boundary Elements*, vol. 31, no. 3, pp. 191–199, 2007.
- [6] M. Sarboland and A. Aminataei, “On the numerical solution of one-dimensional nonlinear nonhomogeneous Burgers’ equation,” *Journal of Applied Mathematics*, vol. 2014, Article ID 598432, 15 pages, 2014.
- [7] H. Pirali, F. Djevanroodi, and M. Haghpanahi, “Combined visibility and surrounding triangles method for simulation of crack discontinuities in meshless methods,” *Journal of Applied Mathematics*, vol. 2012, Article ID 715613, 16 pages, 2012.
- [8] Y. M. Cheng and J. H. Li, “A meshless method with complex variables for elasticity,” *Acta Physica Sinica*, vol. 54, no. 10, pp. 4463–4471, 2005.
- [9] L. Chen and Y. M. Cheng, “Reproducing kernel particle method with complex variables for elasticity,” *Acta Physica Sinica*, vol. 57, no. 1, pp. 1–10, 2008.
- [10] M. Peng, P. Liu, and Y. Cheng, “The complex variable element-free Galerkin (CVEFG) method for two-dimensional elasticity problems,” *International Journal of Applied Mechanics*, vol. 1, no. 2, pp. 367–385, 2009.
- [11] B. Funong, L. Dongming, W. Jianfei, and C. Yumin, “An improved complex variable element-free Galerkin method for two-dimensional elasticity problems,” *Chinese Physics B*, vol. 21, no. 2, Article ID 020204, 2012.
- [12] I. Kaljevic and S. Saigal, “An improved delement free Galerkin formulation,” *International Journal for Numerical Methods in Engineering*, vol. 40, no. 16, pp. 2953–2974, 1997.
- [13] G. J. Wagner and W. K. Liu, “Application of essential boundary conditions in mesh-free methods: a corrected collocation method,” *International Journal for Numerical Methods in Engineering*, vol. 47, no. 8, pp. 1367–1379, 2000.
- [14] I. Kaljević and S. Saigal, “An improved element free Galerkin formulation,” *International Journal for Numerical Methods in Engineering*, vol. 40, no. 16, pp. 2953–2974, 1997.
- [15] T. Zhu and S. N. Atluri, “A modified collocation method and a penalty formulation for enforcing the essential boundary conditions in the element free Galerkin method,” *Computational Mechanics*, vol. 21, no. 3, pp. 211–222, 1998.
- [16] G. R. Liu and Y. T. Gu, “A point interpolation method for two-dimensional solid,” *International Journal for Numerical Methods in Engineering*, vol. 50, no. 4, pp. 937–951, 2001.
- [17] Y. T. Gu and G. R. Liu, “A local point interpolation method for static and dynamic analysis of thin beams,” *Computer Methods in Applied Mechanics and Engineering*, vol. 190, no. 42, pp. 5515–5528, 2001.
- [18] G. R. Liu and Y. T. Gu, “A local radial point interpolation method (LRPIM) for free vibration analyses of 2-D solids,” *Journal of Sound and Vibration*, vol. 246, no. 1, pp. 29–46, 2001.
- [19] G. R. Liu, G. Y. Zhang, Y. T. Gu, and Y. Y. Wang, “A meshfree radial point interpolation method (RPIM) for three-dimensional solids,” *Computational Mechanics*, vol. 36, no. 6, pp. 421–430, 2005.
- [20] H. Wendland, “Error estimates for interpolation by compactly supported radial basis functions of minimal degree,” *Journal of Approximation Theory*, vol. 93, no. 2, pp. 258–272, 1998.
- [21] J. G. Wang and G. R. Liu, “A point interpolation meshless method based on radial basis functions,” *International Journal for Numerical Methods in Engineering*, vol. 54, no. 11, pp. 1623–1648, 2002.
- [22] J. G. Wang and G. R. Liu, “On the optimal shape parameters of radial basis functions used for 2-D meshless methods,” *Computer Methods in Applied Mechanics and Engineering*, vol. 191, no. 23–24, pp. 2611–2630, 2002.
- [23] X. Zhang and Y. Liu, *Meshless Method*, Tsinghua University Press, Beijing, China, 2004.