

Research Article

A Modification of the Moving Least-Squares Approximation in the Element-Free Galerkin Method

Yang Cao,¹ Jun-Liang Dong,² and Lin-Quan Yao³

¹ School of Transportation, Nantong University, Nantong 226019, China

² College of Applied Sciences, Beijing University of Technology, Beijing 100124, China

³ School of Urban Rail Transportation, Soochow University, Suzhou 215006, China

Correspondence should be addressed to Jun-Liang Dong; dongjl@bjut.edu.cn

Received 8 November 2013; Revised 30 January 2014; Accepted 4 February 2014; Published 19 March 2014

Academic Editor: Debasish Roy

Copyright © 2014 Yang Cao 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.

The element-free Galerkin (EFG) method is one of the widely used meshfree methods for solving partial differential equations. In the EFG method, shape functions are derived from a moving least-squares (MLS) approximation, which involves the inversion of a small matrix for every point of interest. To avoid the calculation of matrix inversion in the formulation of the shape functions, an improved MLS approximation is presented, where an orthogonal function system with a weight function is used. However, it can also lead to ill-conditioned or even singular system of equations. In this paper, aspects of the IMLS approximation are analyzed in detail. The reason why singularity problem occurs is studied. A novel technique based on matrix triangular process is proposed to solve this problem. It is shown that the EFG method with present technique is very effective in constructing shape functions. Numerical examples are illustrated to show the efficiency and accuracy of the proposed method. Although our study relies on monomial basis functions, it is more general than existing methods and can be extended to any basis functions.

1. Introduction

In recent years, the meshfree (meshless) method has been developed rapidly as a computational technique for solving partial differential equations. In the meshfree method, only a mesh of nodes and a boundary description are needed to develop the discrete equations. This makes it much flexible in solving problems with moving discontinuities (e.g., fracture of solids) and/or moving boundaries (e.g., shape optimization problems) that cannot be solved easily by conventional numerical methods, such as the finite element method (FEM), finite difference method (FDM), and boundary element method (BEM).

A group of meshfree methods have been proposed and developed, such as the diffuse element method (DEM) [1], the element-free Galerkin method (EFG) [2, 3], the meshfree point interpolation method [4], the meshless method based on radial basis functions [5–7], the meshless local Petrov-Galerkin method [8], and the meshfree weak-strong (MWS) method [9]. Many details about meshfree methods have been presented [10]. Among these meshfree methods, the EFG

method is a very promising method and is currently widely used in computational mechanics and other areas.

In the EFG method, moving least-squares (MLS) approximation is used to construct shape functions for each computational point. One disadvantage of MLS approximation is that a set of linear algebraic equations must be solved for every computational point of interest [11]. If m basis functions were used to construct the shape functions, then an $m \times m$ moment matrix, say \mathbf{A} , must be inverted for every computational point when the discrete equations are assembled. With inappropriate nodes distribution and basis functions, \mathbf{A} may be ill-conditioned or even be singular. Besides, in the postprocessing, \mathbf{A} also must be inverted at each node when the displacements, strains, and stresses are computed. The computational cost associated with this is quite burdensome. Moreover, in order to retain the high accuracy of MLS approximation, the moment matrix \mathbf{A} must be inverted accurately. If the inverse of \mathbf{A} is obtained inaccurately, then the accuracy of EFG method is compromised. In order to ameliorate this shortcoming, an improved MLS (IMLS) approximation is described and studied in

[12–17], where the authors used weighted orthogonal basis functions for constructing MLS approximation by using Schmidt orthogonalization process. With the use of weighted orthogonal basis functions, the moment matrix becomes a diagonal matrix; thus the burden of inverting \mathbf{A} at each computational point is totally eliminated. This idea is further studied and applied in elasticity problems [12, 16], dynamics analysis [12], fracture problems [13], and so on. The use of weighted orthogonal basis functions can also be applied to other meshfree methods based on MLS approximation, such as MLPG method [17].

In [16], some aspects of the MLS approximation with orthogonal basis functions are revisited. The authors found that this technique can be used to avoid computing the inversion of \mathbf{A} but cannot be used to overcome the singularity problems which may occur in the MLS approximation. They used the perturbation method to avoid inversion but did not overcome the singularity problem, either. In [18], the reason for singularity problem was discussed and a method to solve the singularity problem by finding optimal radius of the support domain was proposed. This method is useful for regular nodes distribution and monomial basis functions. For irregular nodes distribution and general basis functions, this method cannot avoid singularity problems. Sunilkumar and Roy proposed another class of techniques in [19], where support domain automatically allows for invertibility of the moment matrix in all cases. In this paper, aspects of MLS approximation with orthogonal basis functions are analyzed in detail. The reason why singularity problem occurs is studied, where the proof is different from that in [18]. A novel technique, which is similar to matrix triangular process [20], is proposed to solve this problem. It is shown that the EFG method with the present technique is very effective in constructing shape functions. Some numerical examples are illustrated to show its efficiency and accuracy. Our study also relies on monomial basis functions, but it is more general than the existing methods and can be extended to any basis functions.

The remainder of this paper is arranged as follows. In Section 2, the MLS approximation with orthogonal functions is reviewed. In Section 3, a comprehensive discussion of the MLS approximation with orthogonal basis functions is analyzed; the cause of singularity problem is derived and a new technique is proposed to solve this problem. The standard EFG formulation is presented for 2D linear elastic problem in Section 4. In Section 5, some numerical examples are presented to show the effectiveness of the proposed method. Finally, this paper is ended with some conclusions and outlook in Section 6.

2. The MLS Approximation with Orthogonal Functions

Consider a subdomain $\Omega_{\mathbf{x}}$, the neighborhood of a point \mathbf{x} and denoted as the domain of definition of the MLS approximation for the trial function at \mathbf{x} , which is located in the problem domain Ω . To approximate the distribution of function u in $\Omega_{\mathbf{x}}$, over a number of randomly located

nodes $\{\mathbf{x}_i\}$, $i = 1, \dots, n$, the moving least-squares (MLS) approximation $u^h(\mathbf{x})$ of u , for all $\mathbf{x} \in \Omega_{\mathbf{x}}$, can be defined as

$$u^h(\mathbf{x}) = \sum_{j=1}^m p_j(\mathbf{x}) a_j(\mathbf{x}) = \mathbf{p}^T(\mathbf{x}) \mathbf{a}(\mathbf{x}), \quad (1)$$

where $\mathbf{p}^T(\mathbf{x}) = [p_1(\mathbf{x}), p_2(\mathbf{x}), \dots, p_m(\mathbf{x})]$ are basis functions and $\mathbf{a}(\mathbf{x})$ is a vector containing the coefficient $a_j(\mathbf{x})$ ($j = 1, 2, \dots, m$), which are also functions of \mathbf{x} . The basis functions can be chosen widely. In general, they can be chosen as the monomial bases, which are defined as follows:

(i) linear basis

$$\begin{aligned} \mathbf{p}^T &= \{1, x\}, & \text{in } 1D \\ \mathbf{p}^T &= \{1, x, y\}, & \text{in } 2D; \end{aligned} \quad (2)$$

(ii) quadratic basis

$$\begin{aligned} \mathbf{p}^T &= \{1, x, x^2\}, & \text{in } 1D \\ \mathbf{p}^T &= \{1, x, y, x^2, xy, y^2\}, & \text{in } 2D. \end{aligned} \quad (3)$$

The coefficient vector $\mathbf{a}(\mathbf{x})$ can be obtained at any point \mathbf{x} by minimizing a weighted discrete L_2 norm, which can be defined as

$$\begin{aligned} \mathbf{J}(\mathbf{x}) &= \sum_{i=1}^n w_i(\mathbf{x}) [\mathbf{p}^T(\mathbf{x}_i) \mathbf{a}(\mathbf{x}) - \hat{u}_i]^2, \\ &= [\mathbf{P}\mathbf{a}(\mathbf{x}) - \hat{\mathbf{u}}]^T \mathbf{W} [\mathbf{P}\mathbf{a}(\mathbf{x}) - \hat{\mathbf{u}}], \end{aligned} \quad (4)$$

where $w_i(\mathbf{x}) = w(\mathbf{x} - \mathbf{x}_i)$ is the weight function associated with the node i , with $w_i(\mathbf{x}) \geq 0$ for all \mathbf{x} in the support domain of $w_i(\mathbf{x})$, \mathbf{x}_i denotes the value of \mathbf{x} at node i , and n is the number of nodes in $\Omega_{\mathbf{x}}$. In this paper, the cubic spline weight function is used

$w_i(x)$

$$= \begin{cases} \frac{2}{3} - 4\left(\frac{d_i}{r_w}\right)^2 + 4\left(\frac{d_i}{r_w}\right)^3, & d_i \leq 0.5r_w, \\ \frac{4}{3} - 4\left(\frac{d_i}{r_w}\right) + 4\left(\frac{d_i}{r_w}\right)^2 - \frac{4}{3}\left(\frac{d_i}{r_w}\right)^3, & 0.5r_w < d_i \leq r_w, \\ 0, & d_i > r_w, \end{cases} \quad (5)$$

where $d_i = |\mathbf{x} - \mathbf{x}_i|$ is the distance from node \mathbf{x}_i to point \mathbf{x} and r_w is the size of the support domain for the weight function. The matrices \mathbf{P} and \mathbf{W} in (4) are

$$\begin{aligned} \mathbf{P} &= \begin{bmatrix} p_1(\mathbf{x}_1) & p_1(\mathbf{x}_2) & \cdots & p_1(\mathbf{x}_n) \\ p_2(\mathbf{x}_1) & p_2(\mathbf{x}_2) & \cdots & p_2(\mathbf{x}_n) \\ \vdots & \vdots & \ddots & \vdots \\ p_m(\mathbf{x}_1) & p_m(\mathbf{x}_2) & \cdots & p_m(\mathbf{x}_n) \end{bmatrix}_{m \times n}, \\ \mathbf{W} &= \begin{bmatrix} w_1(\mathbf{x}) & 0 & \cdots & 0 \\ 0 & w_2(\mathbf{x}) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & w_n(\mathbf{x}) \end{bmatrix}_{n \times n} \end{aligned} \quad (6)$$

and $\hat{\mathbf{u}} = [\hat{u}_1, \hat{u}_2, \dots, \hat{u}_n]$. Because the number of nodes n used in the MLS approximation is usually much larger than the number of unknown coefficients m , the approximated function does not pass through the nodal values. Thus \hat{u}_i ($i = 1, 2, \dots, n$) are the fictitious nodal values, not the nodal values in general.

To find $\mathbf{a}(\mathbf{x})$, we obtain the extremum of \mathbf{J} in (4) by [2, 10]

$$\frac{\partial \mathbf{J}}{\partial \mathbf{a}} = \mathbf{A}(\mathbf{x}) \mathbf{a}(\mathbf{x}) - \mathbf{B}(\mathbf{x}) \hat{\mathbf{u}} = 0, \quad (7)$$

where

$$\begin{aligned} \mathbf{A}(\mathbf{x}) &= \mathbf{P}\mathbf{W}\mathbf{P}^T = \sum_{i=1}^n w_i(\mathbf{x}) \mathbf{p}(\mathbf{x}_i) \mathbf{p}^T(\mathbf{x}_i), \\ \mathbf{B}(\mathbf{x}) &= \mathbf{P}\mathbf{W} \\ &= [w_1(\mathbf{x}) \mathbf{p}(\mathbf{x}_1), w_2(\mathbf{x}) \mathbf{p}(\mathbf{x}_2), \dots, w_n(\mathbf{x}) \mathbf{p}(\mathbf{x}_n)]. \end{aligned} \quad (8)$$

If $\mathbf{A}(\mathbf{x})$ is nonsingular, then from (7) we obtain

$$\mathbf{a}(\mathbf{x}) = \mathbf{A}^{-1}(\mathbf{x}) \mathbf{B}(\mathbf{x}) \hat{\mathbf{u}}. \quad (9)$$

Substituting $\mathbf{a}(\mathbf{x})$ into (1), the expression of the local approximation $u^h(\mathbf{x})$ is thus

$$u^h(\mathbf{x}) = \Phi(\mathbf{x}) \hat{\mathbf{u}} = \sum_{i=1}^n \phi_i(\mathbf{x}) \hat{u}_i, \quad (10)$$

where $\Phi(\mathbf{x}) = \{\phi_1(\mathbf{x}) \cdots \phi_n(\mathbf{x})\} = \mathbf{p}^T(\mathbf{x}) \mathbf{A}^{-1}(\mathbf{x}) \mathbf{B}(\mathbf{x})$ is the shape function and

$$\phi_i(\mathbf{x}) = \sum_{j=1}^m p_j(\mathbf{x}) [\mathbf{A}^{-1}(\mathbf{x}) \mathbf{B}(\mathbf{x})]_{ji}. \quad (11)$$

The derivative of these shape functions can also be obtained

$$\begin{aligned} \phi_{i,k}(\mathbf{x}) &= \sum_{j=1}^m [p_{j,k}(\mathbf{A}^{-1}\mathbf{B})_{ji} + p_j(\mathbf{A}^{-1}_{,k}\mathbf{B} + \mathbf{A}^{-1}\mathbf{B}_{,k})_{ji}], \\ & \quad i = 1, \dots, n, \end{aligned} \quad (12)$$

where $\mathbf{A}_{,k}^{-1} = -\mathbf{A}^{-1} \mathbf{A}_{,k} \mathbf{A}^{-1}$ and the index following a comma is a spatial derivative. Since $\mathbf{A}(\mathbf{x}) = \mathbf{P}\mathbf{W}\mathbf{P}^T$, we should only solve the derivative of \mathbf{P} and \mathbf{W} to get the derivative of $\mathbf{A}(\mathbf{x})$. It should be noted that the above expression is a standard way to compute the derivative. Some novel and numerically accurate schemes for computations of derivatives of shape functions have also been studied; see [21].

As we mentioned above, (7) must be solved accurately for every point to retain the accuracy of the MLS approximation. In the meshfree methods, it may be time consuming and even cannot obtain desired accuracy when $\mathbf{A}(\mathbf{x})$ is ill-conditioned. To overcome these shortcomings, weighted orthogonal basis functions are proposed to derive the improved moving least-squares approximation (IMLS). Using weighted orthogonal basis functions in the MLS approximation, the moment matrix $\mathbf{A}(\mathbf{x})$ in (7) is a diagonal matrix and the necessity

for solving (7) can be eliminated. Moreover, the condition number of the matrix $\mathbf{A}(\mathbf{x})$ can be improved to some extent, but we cannot obtain a well-conditioned matrix $\mathbf{A}(\mathbf{x})$ when ill-conditioned matrix or singular problems occur; see the discussion in [16] and Section 3.

To diagonalize the moment matrix $\mathbf{A}(\mathbf{x})$ for arbitrary computational point \mathbf{x} , the following orthogonality condition is imposed at any point \mathbf{x} where $\mathbf{a}(\mathbf{x})$ is to be computed:

$$\begin{aligned} (q_k, q_j) &= \sum_{i=1}^n w_i(\mathbf{x}) q_k(\mathbf{x}_i) q_j(\mathbf{x}_i) \\ &= \begin{cases} 0, & k \neq j, \\ A_k, & k = j, \end{cases} \quad (k, j = 1, 2, \dots, n), \end{aligned} \quad (13)$$

where the function set $\{q_1(\mathbf{x}), q_2(\mathbf{x}), \dots, q_m(\mathbf{x})\}$ can be termed a weighted orthogonal function set with weight functions $\{w_i\}$ about points $\{\mathbf{x}_i\}$.

For the given arbitrary basis functions $p_k(\mathbf{x})$ ($k = 1, 2, \dots, m$), the orthogonal basis functions $q_k(\mathbf{x})$ can be obtained by using the Schmidt orthogonalization procedure as follows:

$$q_k(\mathbf{x}) = p_k(\mathbf{x}) - \sum_{j=1}^{k-1} \alpha_{kj}(\mathbf{x}) q_j(\mathbf{x}), \quad k = 1, \dots, m, \quad (14)$$

where

$$\alpha_{kj}(\mathbf{x}) = \frac{(p_k, q_j)}{(q_j, q_j)} = \frac{\sum_{i=1}^n w_i(\mathbf{x}) p_k(\mathbf{x}_i) q_j(\mathbf{x}_i)}{\sum_{i=1}^n w_i(\mathbf{x}) q_j^2(\mathbf{x}_i)}. \quad (15)$$

If the weighted orthogonal basis function set about $\{\mathbf{x}_i\}$ is used, then (7) becomes

$$\begin{aligned} \tilde{\mathbf{A}}(\mathbf{x}) \tilde{\mathbf{a}}(\mathbf{x}) &= \begin{bmatrix} (q_1, q_1) & 0 & \cdots & 0 \\ 0 & (q_2, q_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & (q_m, q_m) \end{bmatrix} \begin{bmatrix} \tilde{a}_1(\mathbf{x}) \\ \tilde{a}_2(\mathbf{x}) \\ \vdots \\ \tilde{a}_m(\mathbf{x}) \end{bmatrix} \\ &= \begin{bmatrix} (q_1, \hat{u}_1) \\ (q_2, \hat{u}_2) \\ \vdots \\ (q_m, \hat{u}_m) \end{bmatrix} = \tilde{\mathbf{B}}(\mathbf{x}) \hat{\mathbf{u}}, \end{aligned} \quad (16)$$

where $\tilde{\mathbf{A}}(\mathbf{x}) = \mathbf{Q}\mathbf{W}\mathbf{Q}^T$, $\tilde{\mathbf{B}}(\mathbf{x}) = \mathbf{Q}\mathbf{W}$, and

$$\mathbf{Q} = \begin{bmatrix} q_1(\mathbf{x}_1) & q_1(\mathbf{x}_2) & \cdots & q_1(\mathbf{x}_n) \\ q_2(\mathbf{x}_1) & q_2(\mathbf{x}_2) & \cdots & q_2(\mathbf{x}_n) \\ \vdots & \vdots & \ddots & \vdots \\ q_m(\mathbf{x}_1) & q_m(\mathbf{x}_2) & \cdots & q_m(\mathbf{x}_n) \end{bmatrix}. \quad (17)$$

From (16), we can see that $\tilde{\mathbf{A}}(\mathbf{x})$ is a diagonal matrix. If $(q_k, q_k) \neq 0$ ($k = 1, 2, \dots, m$), then we can obtain the coefficients $\tilde{a}_i(\mathbf{x})$ directly:

$$\tilde{a}_i(\mathbf{x}) = \frac{(q_i, \hat{u}_i)}{(q_i, q_i)}, \quad i = 1, 2, \dots, m. \quad (18)$$

That is,

$$\tilde{\mathbf{a}}(\mathbf{x}) = \bar{\mathbf{A}}(\mathbf{x}) \tilde{\mathbf{B}}(\mathbf{x}) \hat{\mathbf{u}}, \quad (19)$$

where

$$\bar{\mathbf{A}}(\mathbf{x}) = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ (q_1, q_1) & 1 & \cdots & 0 \\ 0 & (q_2, q_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 \\ & & & (q_m, q_m) \end{bmatrix}. \quad (20)$$

Equations (17) and (1) give the following expressions of the approximation function $u^h(\mathbf{x})$ as

$$u^h(\mathbf{x}) = \tilde{\Phi}(\mathbf{x}) \hat{\mathbf{u}} = \sum_{i=1}^n \tilde{\phi}_i(\mathbf{x}) \hat{u}_i, \quad (21)$$

where $\tilde{\phi}_i(\mathbf{x}) = \sum_{j=1}^m q_j(\mathbf{x}) [\bar{\mathbf{A}}(\mathbf{x}) \tilde{\mathbf{B}}(\mathbf{x})]_{ji}$ ($i = 1, 2, \dots, n$) are the shape functions. The derivative of these shape functions can also be obtained:

$$\tilde{\phi}_{i,k}(\mathbf{x}) = \sum_{j=1}^m \left[q_{j,k} (\bar{\mathbf{A}}\mathbf{B})_{ji} + q_j (\bar{\mathbf{A}}_{,k}\mathbf{B} + \bar{\mathbf{A}}\mathbf{B}_{,k})_{ji} \right], \quad (22)$$

$$i = 1, \dots, n,$$

where $\bar{\mathbf{A}}_{,k} = -\bar{\mathbf{A}}\mathbf{A}_{,k}\bar{\mathbf{A}}$ and the index following a comma is a spatial derivative.

3. A Modified Moving Least-Squares Approximation

In this section, more comprehensive analysis of the use of orthogonal basis functions in the MLS approximation is discussed, although some of them are described in [16]. MLS approximation with orthogonal basis functions possesses some distinguished advantages, but there is also a possible singularity problem of the moment matrix $\tilde{\mathbf{A}}$ in its way. A typical example which shows the MLS approximation with orthogonal basis functions fails is presented. The main reason for the singularity problem is analyzed. Then a modified MLS approximation is proposed to overcome this problem.

3.1. Aspects of Use of Orthogonalization. As discussed in [12, 13, 16, 17], the aim of the orthogonalization process is to enhance the computational efficiency and remove inaccuracies in the MLS approximation by avoiding the inversion of an ill-conditioned matrix. Using the orthogonal basis functions, the moment matrix $\tilde{\mathbf{A}}$ is a diagonal matrix; thus the matrix inversion at each computational point is avoided, which can accelerate the computation especially for large problem. Moreover, the condition number of the diagonal moment matrix $\tilde{\mathbf{A}}$ can be improved greatly for a lot of problems. But this process fails when the original moment matrix \mathbf{A} is a singular (ill-conditioned) matrix. That is to say the source of

inaccuracy causing the ill-conditioning is not removed by this procedure.

It is shown in [12] that orthogonal basis functions $q_k(\mathbf{x})$ can be obtained from Pascal basis $p_k(\mathbf{x})$ by using the Schmidt orthogonalization. Thus, there is a linear relation between them; that is, each $q_j(\mathbf{x})$ can be written as

$$q_j(\mathbf{x}) = \sum_{i=1}^m g_{ji} p_i(\mathbf{x}), \quad j = 1, \dots, m, \quad (23)$$

or in matrix form

$$\mathbf{q}(\mathbf{x}) = \mathbf{G}\mathbf{p}(\mathbf{x}), \quad (24)$$

where $\mathbf{G} \in \mathbb{R}^{m \times m}$ is a transform matrix mapping $\mathbf{p}(\mathbf{x})$ to $\mathbf{q}(\mathbf{x})$. The Schmidt orthogonalization makes \mathbf{G} a lower triangular matrix with all diagonal entries of unit value. Thus we obtain $\det(\mathbf{G}) = 1$, where $\det(\#)$ means the determinant of matrix $\#$. Equation (24) makes the following equality hold:

$$\mathbf{Q} = \mathbf{G}\mathbf{P}, \quad (25)$$

where \mathbf{P} and \mathbf{Q} are defined in (6) and (17), respectively. Then we have

$$\bar{\mathbf{A}} = \mathbf{Q}\mathbf{W}\mathbf{Q}^T = \mathbf{G}\mathbf{P}\mathbf{W}\mathbf{P}^T\mathbf{G}^T = \mathbf{G}\mathbf{A}\mathbf{G}^T, \quad (26)$$

$$\tilde{\mathbf{B}} = \mathbf{Q}\mathbf{W} = \mathbf{G}\mathbf{P}\mathbf{W} = \mathbf{G}\mathbf{B}.$$

From (26), we can find the following.

- (1) $\bar{\mathbf{A}}$ is congruent with \mathbf{A} , and these two matrices have the same inertia, which means these two matrices have same numbers of positive, negative, and zero eigenvalues.
- (2) Solving (16) is equivalent to solving

$$\mathbf{G}\mathbf{A}\mathbf{G}^T\mathbf{G}^{-T}\mathbf{a}(\mathbf{x}) = \mathbf{G}\mathbf{B}, \quad (27)$$

which means the matrix \mathbf{G} can be viewed as a split preconditioner [22] for solving (7). Using a preconditioner, the condition number of the system matrix may be reduced. That is to say, the condition number of $\bar{\mathbf{A}} = \mathbf{G}\mathbf{A}\mathbf{G}^T$ may be improved from matrix theory. This has been proved for a lot of problems in applications when \mathbf{A} is ill-conditioned.

- (3) Since \mathbf{G} is a lower triangular matrix with all diagonal entries of unit value, that is, $\det(\mathbf{G}) = 1$, therefore

$$\begin{aligned} |\det(\bar{\mathbf{A}})| &= |\det(\mathbf{G}) \det(\mathbf{A}) \det(\mathbf{G}^T)| \\ &= |\det(\mathbf{G})| |\det(\mathbf{A})| |\det(\mathbf{G})| = |\det(\mathbf{A})|, \end{aligned} \quad (28)$$

which means if \mathbf{A} is ill-conditioned, then the determinant of $\bar{\mathbf{A}}$ is close to zero. And we can get that $\det(\bar{\mathbf{A}})$ is also close to zero.

- (4) If \mathbf{A} and $\bar{\mathbf{A}}$ are well defined, then the shape functions derived using either bases are identical; that is,

$$\begin{aligned} \tilde{\Phi}(\mathbf{x}) &= \mathbf{q}^T(\mathbf{x}) \tilde{\mathbf{A}}^{-1}(\mathbf{x}) \tilde{\mathbf{B}}(\mathbf{x}) = \mathbf{p}^T(\mathbf{x}) \mathbf{G}^T \mathbf{G}^{-T} \mathbf{A}^{-1} \mathbf{G}^{-1} \mathbf{G}\mathbf{B} \\ &= \mathbf{p}^T(\mathbf{x}) \mathbf{A}^{-1} \mathbf{B}(\mathbf{x}) = \Phi(\mathbf{x}). \end{aligned} \quad (29)$$

TABLE 1: Degradation cases and perturbation of shape functions.

Node $8'$	$\text{cond}(A)$	$\text{cond}(\tilde{A})$	$\min \tilde{A}$	ϕ_1	ϕ_2	ϕ_3	ϕ_4	$\sum_i \phi_i$
				ϕ_5	ϕ_6	ϕ_7	ϕ_8	
(3.0, 1.0)	1.70e28	1.08e34	0.0	NaN	NaN	NaN	NaN	NaN
(2.999, 0.999)	4.47e9	1.57e8	7.86e - 11	-99.96	100.39	99.85	-100.02	1.00
(2.99, 0.99)	3.87e7	1.33e6	9.29e - 9	-50.02	350.03	-549.52	250.25	1.00
(2.9, 0.9)	3.47e5	7.47e3	1.87e - 6	-9.96	10.39	9.85	-10.03	1.00
(3.0, 0.9)	3.83e5	9.46e3	1.39e - 6	-5.01	35.03	-54.52	25.25	1.00
				-0.96	1.38	0.82	-1.03	1.00
				-0.51	3.53	-5.01	2.78	1.00
				-1.10	1.55	0.95	-1.18	1.00
				-0.62	3.98	-5.36	2.78	1.00

Moreover, the derivatives of these shape functions are also identical; that is,

$$\begin{aligned}
\tilde{\Phi}_{,k}(\mathbf{x}) &= \mathbf{q}_{,k}^T \tilde{\mathbf{A}}^{-1} \tilde{\mathbf{B}} + \mathbf{q}^T \tilde{\mathbf{A}}_{,k}^{-1} \tilde{\mathbf{B}} + \mathbf{q}^T \tilde{\mathbf{A}}^{-1} \tilde{\mathbf{B}}_{,k} \\
&= \mathbf{q}_{,k}^T \tilde{\mathbf{A}}^{-1} \tilde{\mathbf{B}} - \mathbf{q}^T \tilde{\mathbf{A}}^{-1} \tilde{\mathbf{A}}_{,k} \tilde{\mathbf{A}}^{-1} \tilde{\mathbf{B}} + \mathbf{q}^T \tilde{\mathbf{A}}^{-1} \tilde{\mathbf{B}}_{,k} \\
&= \mathbf{p}_{,k}^T \mathbf{G}^T \mathbf{G}^{-T} \mathbf{A}^{-1} \mathbf{G}^{-1} \mathbf{G} \mathbf{B} \\
&\quad - \mathbf{p}^T \mathbf{G}^T \mathbf{G}^{-T} \mathbf{A}^{-1} \mathbf{G}^{-1} \mathbf{G} \mathbf{A}_{,k} \mathbf{G}^T \mathbf{G}^{-T} \mathbf{A}^{-1} \mathbf{G}^{-1} \mathbf{G} \\
&\quad + \mathbf{p}^T \mathbf{G}^T \mathbf{G}^{-T} \mathbf{A}^{-1} \mathbf{G}^{-1} \mathbf{G} \mathbf{B}_{,k} \\
&= \mathbf{p}_{,k}^T \mathbf{A}^{-1} \mathbf{B} - \mathbf{p}^T \mathbf{A}^{-1} \mathbf{A}_{,k} \mathbf{A}^{-1} \mathbf{B} + \mathbf{p}^T \mathbf{A}^{-1} \mathbf{B}_{,k} \\
&= \Phi_{,k}(\mathbf{x}).
\end{aligned} \tag{30}$$

From these aspects, we know that if \mathbf{A} is not well defined (ill-conditioned or even singular), then we cannot obtain a well-conditioned diagonal moment matrix $\tilde{\mathbf{A}}$. But the accuracy may be improved for many applications; this is because the transformation matrix \mathbf{G} , which can be considered as a preconditioner, can reduce the condition number in some extent (see also Table 1). That is why many researchers believe that using the orthogonal basis functions in MLS approximation can improve the accuracy of solution [13–15]. The following example also shows the above discussion. Consider a computational point \mathbf{x} located at (1.5, 0.5), whose influence domain contains eight nodes; see Figure 1. These eight nodes are located in parallel lines in both x and y directions. The rectangle domain was chosen as the support domain of the computational point \mathbf{x} . The radiuses of the support domain are $\alpha_x = 2.0$ in x direction and $\alpha_y = 0.6$ in y direction. These values are not of significance but mean that all eight nodes are located in the influence domain of \mathbf{x} .

When quadratic basis and cubic weight functions are utilized to automatically construct the shape function, we find that the moment matrix is

$$\begin{aligned}
\mathbf{A}(\mathbf{x}) &= \mathbf{P} \mathbf{W} \mathbf{P}^T \\
&= \begin{bmatrix} 0.01235 & 0.01852 & 0.00617 & 0.03189 & 0.00926 & 0.00617 \\ 0.01852 & 0.03189 & 0.00926 & 0.06019 & 0.01595 & 0.00926 \\ 0.00617 & 0.00926 & 0.00617 & 0.01595 & 0.00926 & 0.00617 \\ 0.03189 & 0.06019 & 0.01595 & 0.12139 & 0.03009 & 0.01595 \\ 0.00926 & 0.01595 & 0.00926 & 0.03009 & 0.01595 & 0.00926 \\ 0.00617 & 0.00926 & 0.00617 & 0.01595 & 0.00926 & 0.00617 \end{bmatrix}
\end{aligned} \tag{31}$$

$$\begin{aligned}
\tilde{\mathbf{A}}(\mathbf{x}) &= \text{diag}(0.01235, 0.00412, 0.00309, 0.00197, \\
&\quad 0.00103, 0.00000).
\end{aligned} \tag{32}$$

It is easy to see that $\tilde{\mathbf{A}}$ is also rank deficiency. This example shows that it is possible to form an ill-conditioned or singular linear system of equations, even if orthogonalization procedure is used. If \mathbf{A} is singular or there is a zero on the diagonal of $\tilde{\mathbf{A}}$, then no shape functions can be derived. If some nodes are perturbed, for example, node 8 is perturbed to node $8'$ (see Figure 1), then \mathbf{A} becomes nonsingular but may be an ill-conditioned matrix. It should be noted that node $8'$ shown in Figure 1 is just one case; it can be perturbed in any direction indeed. In the perturbed case, using the orthogonalization process may be more effective than the traditional MLS approximation, since the condition number decreases and no matrix inversion is needed to compute. Table 1 presents the minimum values of the condition number of \mathbf{A} and $\tilde{\mathbf{A}}$, the minimum values of $\tilde{\mathbf{A}}_{jj}$, and the values of shape functions. $\phi_1 - \phi_8$ stand for the shape functions of node 1–node 8, respectively. Σ indicates the sum of these shape functions.

3.2. The Cause of Singularity Problem. From above discussion, we know that the MLS approximation with orthogonal basis functions may also fail, although the number of

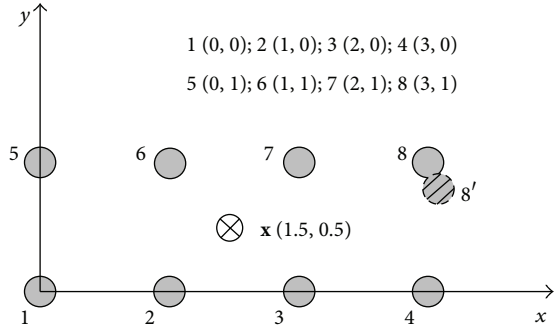


FIGURE 1: An example with eight nodes and their coordinates.

nodes in the support domain of a computational point \mathbf{x} is larger than the number of bases. The cause of singularity problem has been discussed in the original paper of the MLS approximation [11] and a recent paper [18], where the authors also presented the optimal radius of the support domain of a computational point for polynomial basis. In this subsection, we discuss the cause of singularity problem from matrix theory.

Close examination has shown that the problem of singularity arises from rank deficiency of matrix \mathbf{P} . In fact, the matrix $\mathbf{W} = \text{diag}(w(\mathbf{x} - \mathbf{x}_1), \dots, w(\mathbf{x} - \mathbf{x}_n))$ is a diagonal positive definite matrix for a given set of nodes in the support domain. If we further assume that the number of these nodes is larger than the number of monomial bases and \mathbf{P} has full row rank, then we can obtain a nonsingular moment matrix $\mathbf{A}(\mathbf{x})$ from the matrix theory of view. The following proposition gives the nonsingular condition of the moment matrix $\mathbf{A}(\mathbf{x})$. The proof of this proposition is given in the appendix.

Proposition 1. *The moment matrix $\mathbf{A}(\mathbf{x})$ is symmetric. Assume that a set of nodes are located in the support domain of a computational node \mathbf{x} ; then the moment matrix $\mathbf{A}(\mathbf{x})$ is a nonsingular matrix provided that*

$$n \geq m, \quad \text{rank}(\mathbf{P}) = m; \quad (33)$$

that is, \mathbf{P} has full row rank, where n is the number of nodes in the support domain and m is the number of basis functions.

3.3. Techniques to Avoid the Singularity. To avoid a singular moment matrix, some techniques have been proposed, such as the perturbation method [16] and finding appropriate support domain for a computational point [18]. Using a perturbation method, one could not determine the disturbance beforehand. If the disturbance is too small, the condition number will be very large; see Table 1 for a degradation example. If the disturbance is too large, the moment matrix may be well defined, but the nodes are not the original nodes; this will take large error for discrete linear systems. Finding appropriate support domain for a computational point is a good choice to avoid the singularity problem, but it is only efficient for regular nodes distribution and special basis functions, such as monomial bases. For irregular nodes distribution and general basis functions, this method is not

very efficient. In this subsection, we discuss a new technique, which is similar to the matrix triangularization algorithm (MTA) [20].

As discussed above, the reason of the singularity problem of the moment matrix $\mathbf{A}(\mathbf{x})$ is the rank deficiency of \mathbf{P} due to the improper enclosure of nodes and basis functions. For a given set of nodes, the basis functions are very important. The main idea of this technique is, therefore, to automatically find the monomials that are responsible for the rank deficiency for a given set of nodes. These monomials should be done specially to ensure a full rank moment matrix. But this technique should be an automatic procedure, and it has to be done without increasing too much computational cost and should not be too complex.

The details of the algorithm are as follows.

- (1) Suppose that m monomials are chosen to obtain the basis functions and n nodes are selected in the support domain of an interpolation point \mathbf{x} . Using (6), the matrix \mathbf{P} can be obtained. It should be noted that the rows of \mathbf{P} correspond to the monomials in the basis, and its columns associate with the nodes in the support domain.
- (2) The matrix \mathbf{P} is triangularized to determine the row rank r .
 - (i) If $r = m$, it means that \mathbf{P} has full row rank. From Proposition 1, we know that the moment matrices $\mathbf{A}(\mathbf{x})$ and $\tilde{\mathbf{A}}(\mathbf{x})$ are generally well defined. In this case, the orthogonal basis functions $\mathbf{q}(\mathbf{x})$ can be obtained by using the Schmidt orthogonalization process. Thus, a diagonal moment matrix $\tilde{\mathbf{A}}(\mathbf{x})$ and its inversion $\bar{\mathbf{A}}(\mathbf{x})$ are obtained. Therefore, shape functions can be easily got. In particular, the moment matrix $\mathbf{A}(\mathbf{x})$ may be ill-conditioned. In this case, using the Schmidt orthogonalization process is a good choice and shape functions can also be easily got (see discussion in Section 3.1).
 - (ii) If $r < m$, it means that there is a rank deficiency in \mathbf{P} . Thus, the moment matrix is singular and we cannot obtain the shape functions. The reason of the rank deficiency is that some rows of \mathbf{P} ($m - r$ rows) are linearly related to other rows. Therefore, the $m - r$ rows should be done with some special techniques. The procedure is as follows.
 - (a) In the row triangularization process, the permutations of rows are recorded. From the diagonal elements of the triangularized matrix, we can find out which rows (total $m - r$ rows) are related to other rows. This means that these rows (which corresponded to the monomials) have no effect in forming shape functions.
 - (b) We can do the Schmidt orthogonalization process for the remaining r basis functions. Since basis functions are corresponding to

the rows of \mathbf{P} , we can set the corresponding $m - r$ rows and $m - r$ columns of \mathbf{A} or $\tilde{\mathbf{A}}$ to be zero except the diagonal entry to be one and set the $m - r$ rows of \mathbf{B} or $\tilde{\mathbf{B}}$ to be zero.

- (c) Through the above operation, we can obtain a diagonal moment matrix $\tilde{\mathbf{A}}(\mathbf{x})$, which has full rank now. Thus the shape functions can be obtained easily.

3.4. Flowchart of the Technique. The flowchart of the technique is briefly given as follows.

- (1) Determine the support domain for point \mathbf{x} to obtain an $m \times n$ matrix \mathbf{P} and an $n \times n$ weight matrix \mathbf{W} .
- (2) Transform \mathbf{P} to be a row trapezoidal matrix to get row rank r and record the exchanges of rows.
- (3) If $r = m$, then go to the next step. If $r < m$, then determine which $m - r$ rows are related to other rows. The remaining r basis functions are used to construct orthogonal basis functions. Set these $m - r$ rows and $m - r$ columns of \mathbf{A} or $\tilde{\mathbf{A}}$ to be zero except the diagonal entry to be one, and set the $m - r$ rows of \mathbf{B} or $\tilde{\mathbf{B}}$ to be zero.
- (4) Compute the shape functions from (16)–(18).

3.5. Technique for an Eight-Node Approximation. In order to show how this technique works, an example of eight-node approximation, as shown in Figure 1, is presented here in detail. These eight nodes are included in the support domain of an interpolation point $\mathbf{x} = [1.5, 0.5]$. The complete quadratic basis and the cubic weight function (5) are used. Thus $m = 6$ and $n = 8$.

- (1) Construct the matrix \mathbf{P} : using (6), the matrix \mathbf{P} is constructed as follows:

$$\mathbf{P} = \begin{matrix} \text{node 1} & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \text{node 8} \\ \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 & 0 & 1 & 2 & 3 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 4 & 9 & 0 & 1 & 4 & 9 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2 & 3 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix} & \begin{matrix} 1 \\ x_i \\ y_i \\ x_i^2 \\ x_i y_i \\ y_i^2 \end{matrix} \end{matrix} \quad (34)$$

It can be easy to find that rows of \mathbf{P} correspond to the monomials in the bases, and its columns associate with the eight nodes in the support domain.

- (2) Row rank and row determination: using some permutation matrices, the matrix \mathbf{P} is transformed to a row trapezoidal matrix to determine the row rank. The row trapezoidal matrix $\hat{\mathbf{P}}$ is

$$\hat{\mathbf{P}} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 3 & 1 & 2 & 2 & 3 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 6 & 0 & 2 & 2 & 6 \\ 0 & 0 & 0 & 0 & 1 & 0 & 2 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (35)$$

TABLE 2: Shape functions obtain by matrix triangular process.

ϕ_i				$\sum \phi_i$
$\phi_1 = -0.03125$	$\phi_2 = 0.28125$	$\phi_3 = 0.28125$	$\phi_4 = -0.03125$	1.0000
$\phi_5 = -0.03125$	$\phi_6 = 0.28125$	$\phi_7 = 0.28125$	$\phi_8 = -0.03125$	

It can be found that the sixth row of $\hat{\mathbf{P}}$ is zero and the row rank is 5. It means that there is a rank deficiency in \mathbf{P} . Therefore, there is a rank deficiency in the moment matrix $\mathbf{A}(\mathbf{x})$ from Proposition 1. From $\hat{\mathbf{P}}$ we can also find that the sixth row is linearly related to other five rows. It should be noted that there is no row changed in the row triangularization process, which means that the sixth row of $\hat{\mathbf{P}}$ is corresponding to the sixth row of \mathbf{P} and the sixth row of \mathbf{P} is also linearly related to other five rows.

- (3) Since the sixth row is corresponding to the basis function y^2 , the remaining basis functions $\{1, x, y, x^2, xy\}$ are used to construct orthogonal basis functions $\mathbf{q}(\mathbf{x})$ by Schmidt orthogonalization process. Now, the sixth row and the sixth column of $\tilde{\mathbf{A}}(\mathbf{x})$ are set to be zero except the diagonal entry to be one, and the sixth row of $\tilde{\mathbf{B}}(\mathbf{x})$ is set to be zero. In this case, we have the moment matrix

$$\tilde{\mathbf{A}}(\mathbf{x}) = \text{diag}(0.01235, 0.00412, 0.00309, 0.00197, 0.00103, 1.00000). \quad (36)$$

- (4) Compute the MLS shape functions from (16)–(18): after the above steps, the moment matrix $\tilde{\mathbf{A}}$ has full rank. The shape functions can be obtained easily. They are listed in Table 2.

4. Element-Free Galerkin Formulation

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

$$\nabla \cdot \boldsymbol{\sigma} + \mathbf{b} = 0, \quad \text{in } \Omega, \quad (37)$$

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

$$\begin{aligned} \boldsymbol{\sigma} \cdot \mathbf{n} &= \bar{\mathbf{t}} \quad \text{on } \Gamma_t, \\ \mathbf{u} &= \bar{\mathbf{u}} \quad \text{on } \Gamma_u, \end{aligned} \quad (38)$$

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 .

Using the standard principle of minimum potential energy for (37)–(38), that is, to find $\mathbf{u} \in (H^1(\omega))^2$, such that

$$\Pi = \frac{1}{2} \int_{\Omega} \boldsymbol{\varepsilon}^T \mathbf{D} \boldsymbol{\varepsilon} \, d\Omega - \int_{\Omega} \mathbf{u}^T \cdot \mathbf{b} \, d\Omega - \int_{\Gamma_t} \mathbf{u}^T \cdot \bar{\mathbf{t}} \, d\Gamma \quad (39)$$

is stationary, where $H^i(\omega)$ denotes the Sobolev space of order i and ω is a function in the Sobolev space, $\boldsymbol{\varepsilon}$ and $\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon}$ are strain and stress vectors, and \mathbf{D} is the strain-stress matrix. MLS equation (10) or (21) is used to approximate the displacements in the Galerkin procedure. Then we can get

$$\mathbf{u} = \begin{Bmatrix} u \\ v \end{Bmatrix} = \sum_{j=1}^n \begin{bmatrix} \phi_j & 0 \\ 0 & \phi_j \end{bmatrix} \begin{Bmatrix} u_j \\ v_j \end{Bmatrix} = \sum_{j=1}^n \Phi_j \mathbf{u}_j. \quad (40)$$

Then substituting (40) into (39) leads to the following total potential energy in the matrix form, as

$$\Pi = \frac{1}{2} \mathbf{u}^T \cdot \mathbf{K} \cdot \mathbf{u} - \mathbf{u}^T \cdot \mathbf{f} \quad (41)$$

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

$$\mathbf{K}\mathbf{u} = \mathbf{f}, \quad (42)$$

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

$$\begin{aligned} \mathbf{K}_{ij} &= \int_{\Omega} \mathbf{B}_i^T \mathbf{D} \mathbf{B}_j d\Omega, \quad i, j = 1, 2, \dots, n, \\ \mathbf{f}_i &= \int_{\Omega} \phi_i \mathbf{b} d\Omega + \int_{\Gamma_i} \phi_i \bar{\mathbf{t}} d\Gamma, \quad i = 1, 2, \dots, n, \end{aligned} \quad (43)$$

where

$$\begin{aligned} \mathbf{B}_i &= \begin{bmatrix} \phi_{i,x} & 0 \\ 0 & \phi_{i,y} \\ \phi_{i,y} & \phi_{i,x} \end{bmatrix}, \\ \mathbf{D} &= \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \text{ (for plane stress problem)}. \end{aligned} \quad (44)$$

In the sequel and unless mentioned otherwise, EFG method indicates the element-free Galerkin method with the original MLS approximation (10) and MEFG method stands for the element-free Galerkin method with the modified MLS approximation (21). These methods employ the moving least-squares (MLS) approximation or its modified form to approximate the trial functions. Another problem is that, in general, they do not have the property of nodal interpolants as in the FEM; that is, $\phi_i(\mathbf{x}_j) \neq \delta_{ij}$, where $\phi_i(\mathbf{x}_j)$ is the shape function corresponding to the node at \mathbf{x}_i , evaluated at a nodal point, \mathbf{x}_j , and δ_{ij} is the Kronecker data, unless the weight functions used in the MLS approximation are singular at nodal points [11]. Therefore, essential boundary conditions should be imposed with special techniques [23, 24]. In this paper, we use the penalty method [23].

5. Numerical Experiments

In this section, two numerical examples are presented to show the efficiency of the EFG method with the modified

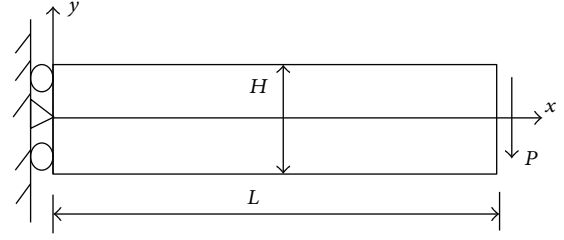


FIGURE 2: Cantilever beam.

MLS approximation studied in this paper. In these examples, quadratic basis function and cubic spline weight function are used. All runs are performed in MATLAB 7.0 on an Intel Pentium 4 (2 G RAM) Windows XP system.

5.1. Cantilever Beam. The first example is a cantilever beam of length L and height H subjected to traction at the free end (see Figure 2). The beam has a unit thickness and hence a plane stress problem is considered here. The closed-form solution is available for parabolic traction of force P

$$\begin{aligned} u_x &= \frac{Py}{6EI} \left[(6L - 3x)x + (2 + \nu) \left(y^2 - \frac{H^2}{4} \right) \right], \\ u_y &= -\frac{P}{6EI} \left[3\nu y^2 (L - x) + (4 + 5\nu) \frac{H^2 x}{4} + (3L - x)x^2 \right], \end{aligned} \quad (45)$$

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

$$\sigma_x = \frac{P(L-x)y}{I}, \quad \tau_{xy} = -\frac{P}{2I} \left(\frac{H^2}{4} - y^2 \right), \quad \sigma_y = 0. \quad (46)$$

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

To evaluate the accuracy of the coupled method, the following error norms are used:

$$r_u = \frac{\|\hat{\mathbf{u}} - \mathbf{u}\|_2}{\|\mathbf{u}\|_2}, \quad r_\sigma = \frac{\|\hat{\boldsymbol{\sigma}} - \boldsymbol{\sigma}\|_2}{\|\boldsymbol{\sigma}\|_2}, \quad (47)$$

where $\|\#\|_2$ is 2-norm of vector “#”, $\hat{\mathbf{u}}$ and \mathbf{u} are the approximation and exact solution of displacements, and $\hat{\boldsymbol{\sigma}}$ and $\boldsymbol{\sigma}$ are the approximation and exact value of stresses.

To show the effectiveness of EFG method with the modified MLS approximation studied in this paper, a typical regular node distribution (with 37×13 nodes) is shown in Figure 3. To evaluate the integrals, 15×8 background cells are used. For each background 4×4 Gauss points are employed. The rectangle domain is chosen as the support domain of the computational point \mathbf{x} . We adopt two kinds of radius of the support domain. The first case is $\alpha_x = 2.2$ in the x direction and $\alpha_y = 1.3$ in the y direction. The second case is $\alpha_x = \alpha_y = 3.0$ in x and y directions, respectively. In the first case, singularity problem of the moment matrix $\mathbf{A}(\mathbf{x})$ occurs;

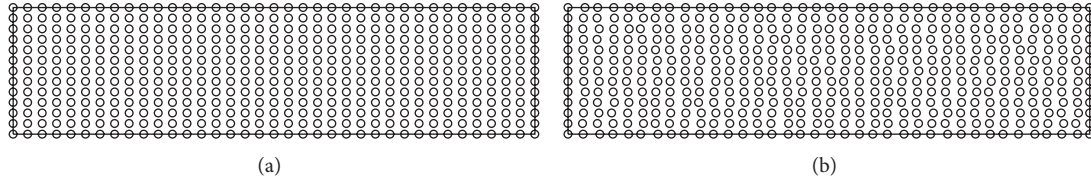


FIGURE 3: Typical node distribution: regular 37×13 nodes (a); irregular 481 nodes (b).

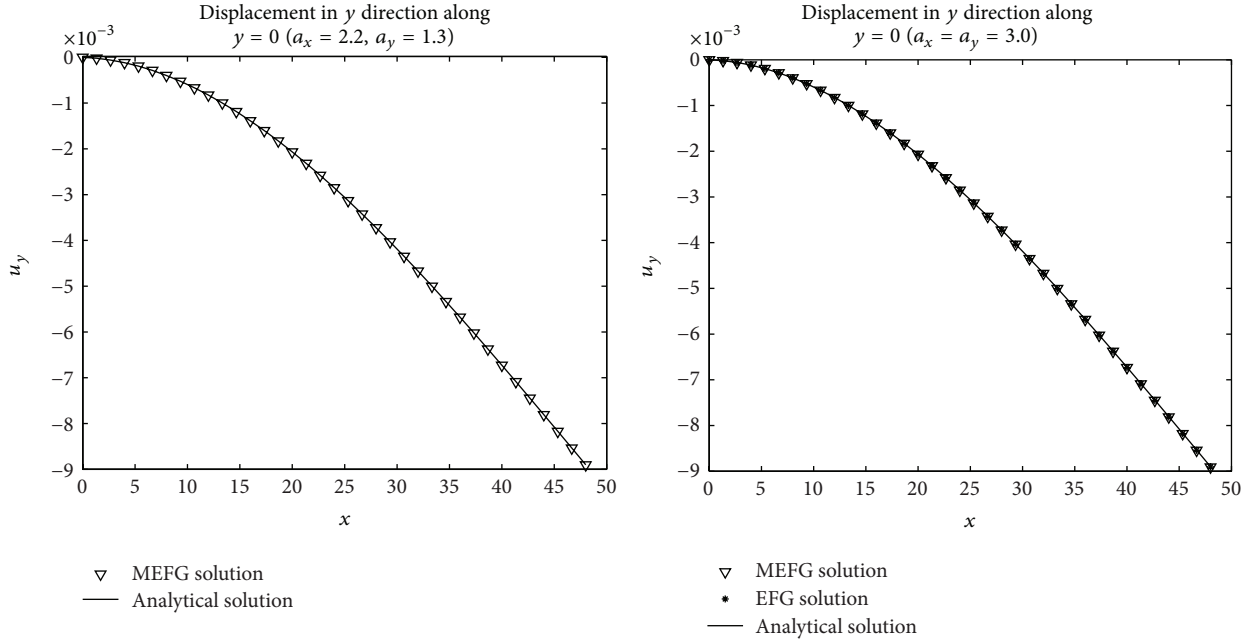


FIGURE 4: Comparison of displacement u_y along $y = 0$ between analytical and numerical solution.

TABLE 3: Convergence study for Example 4.1.

Regular nodes	13×4	25×7	37×13	49×16
Background cells	5×4	10×4	15×8	20×15
r_u	0.0010	$9.7847e - 4$	$5.5448e - 4$	$1.8456e - 4$
r_σ	0.0932	0.0373	0.0150	0.0104
Irregular nodes	52	175	481	784
Background cells	5×4	10×4	15×8	20×15
r_u	0.0763	0.0207	0.0239	0.0014
r_σ	0.3934	0.1618	0.1677	0.0490

thus the EFG method fails, while the MEFG method can also obtain excellent agreement with the analytical solution; see Figures 4 and 5. In the second case, no singularity problem appears; that is, the moment matrix $\mathbf{A}(\mathbf{x})$ is always well defined. Thus both the EFG method and the MEFG method can obtain excellent agreement with the analytical solution; see also Figures 4 and 5. It should be also noted that, in the second case, the results of EFG method and MEFG method are almost the same, which confirms the theoretical analysis in Section 3.1.

The convergence of the MEFG method and the accuracy of the MEFG method for irregular node distribution are also

studied. Four different regular node distributions with 52 (13×4 : 13 nodes in the x direction, 4 nodes in the y direction), 175 (25×7) nodes, 481 (37×13) nodes, and 784 (49×16) nodes and four different irregular node distributions are considered. Figure 3 also plots a typical irregular node distribution (with 481 nodes). For intergration for each problem, $5 \times 4, 10 \times 4, 15 \times 8, 20 \times 15$ background cells are used. Here, the radius of the support domain is taken as $\alpha_x = 2.2$ in the x direction and $\alpha_y = 2.3$ in the y direction. The computational results are listed in Table 3. The convergence curves for displacements and stresses with different node distributions are plotted in Figure 6. In Figure 6, h is the maximum size of node arrangement. From Table 3 and Figure 6, we can see that the MEFG method possesses convergence. Convergence for the regular node distribution problem is better than that for the irregular distribution problem.

5.2. Poisson Equation. Consider Poisson equation,

$$-\Delta u = 2(x + y - x^2 - y^2) \quad \text{in } \Omega = [0, 1] \times [0, 1] \quad (48)$$

with boundary conditions

$$u = 0.0 \quad \text{on } \partial\Omega \quad (49)$$

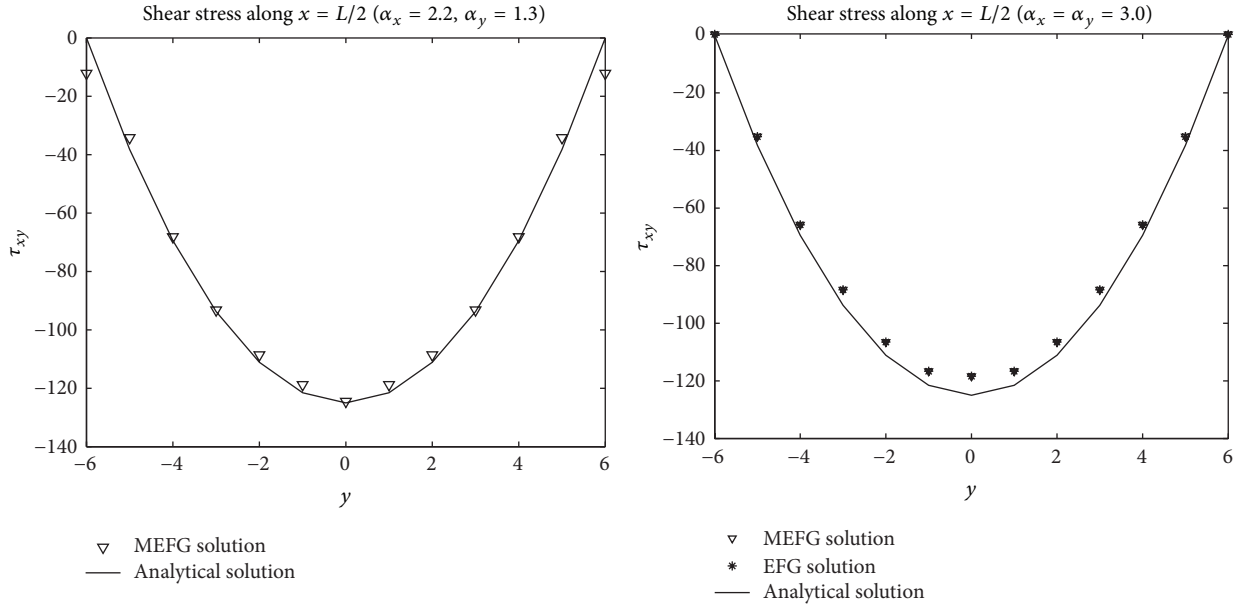


FIGURE 5: Comparison of shear stress along $x = L/2$ between analytical and numerical solution.

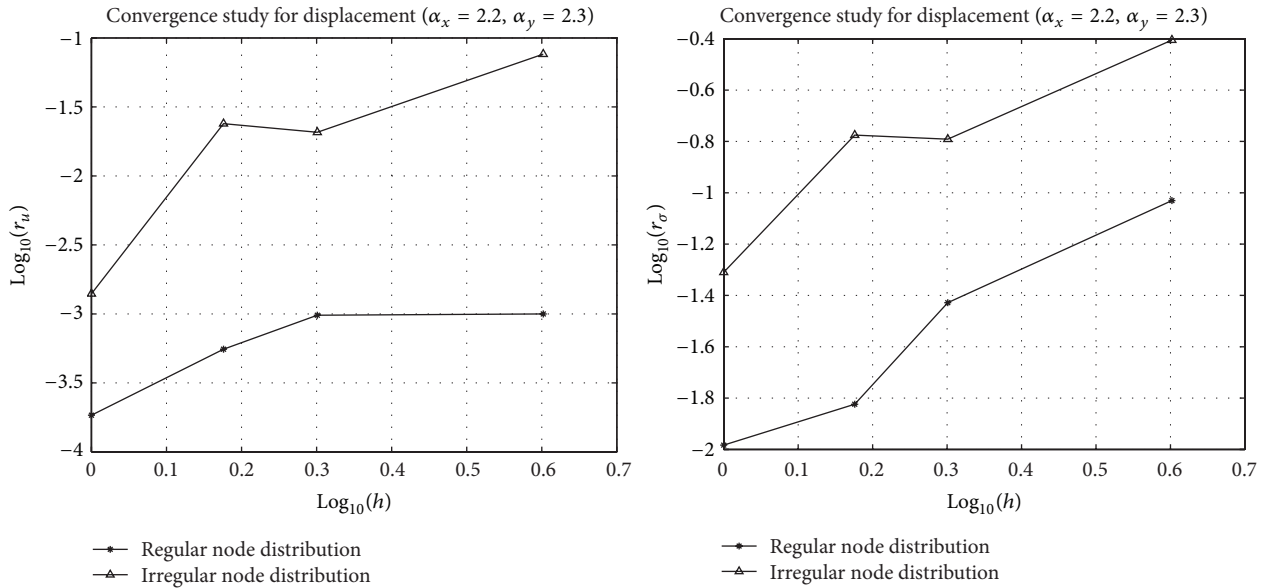


FIGURE 6: Convergence study for regular and irregular node distributions.

whose analytical solution is given by

$$u = (x - x^2)(y - y^2). \tag{50}$$

The EFG formulation of Poisson equation can be easily obtained from the same procedure (39)–(41). A typical 21×21 regular nodes discretization and 10×10 background cells are shown in Figure 7. We use two kinds of radius of the support domain to show the effectiveness of the MEFG method. The first case is $\alpha_x = 3$ in the x direction and $\alpha_y = 1.5$ in the y direction. The second case is $\alpha_x = \alpha_y = 3.0$ in both x and y directions, respectively. Figures 8 and 9 plot the numerical solution and analytical solution along $x = 1/2$ and $y = 0.2$.

From these figures, we can see that the EFG method fails for the first case while the MEFG method can also obtain good numerical results, since singularity problem for the moment matrix $\mathbf{A}(\mathbf{x})$ occurs when the original MLS approximation is used. For the second case, both the EFG method and the MEFG method can obtain excellent agreement with the analytical solution, and the results of EFG method and MEFG method are almost the same.

For convergence studies, four different regular node distributions with 121 (11×11) nodes, 441 (21×21) nodes, 961 (31×31) nodes, and 1681 (41×41) nodes are employed. Two kinds of radius of the support domain studied above

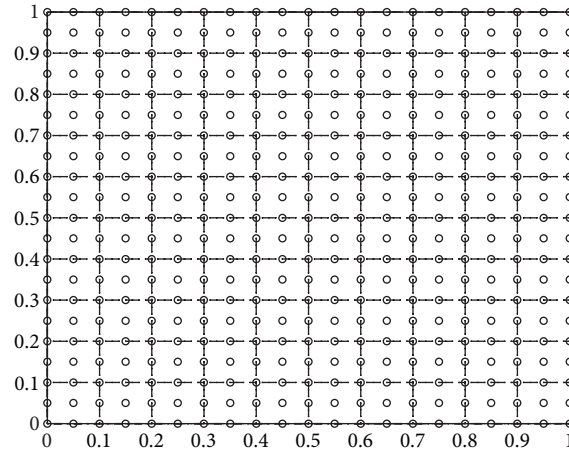


FIGURE 7: Typical node distribution for Poisson equation.

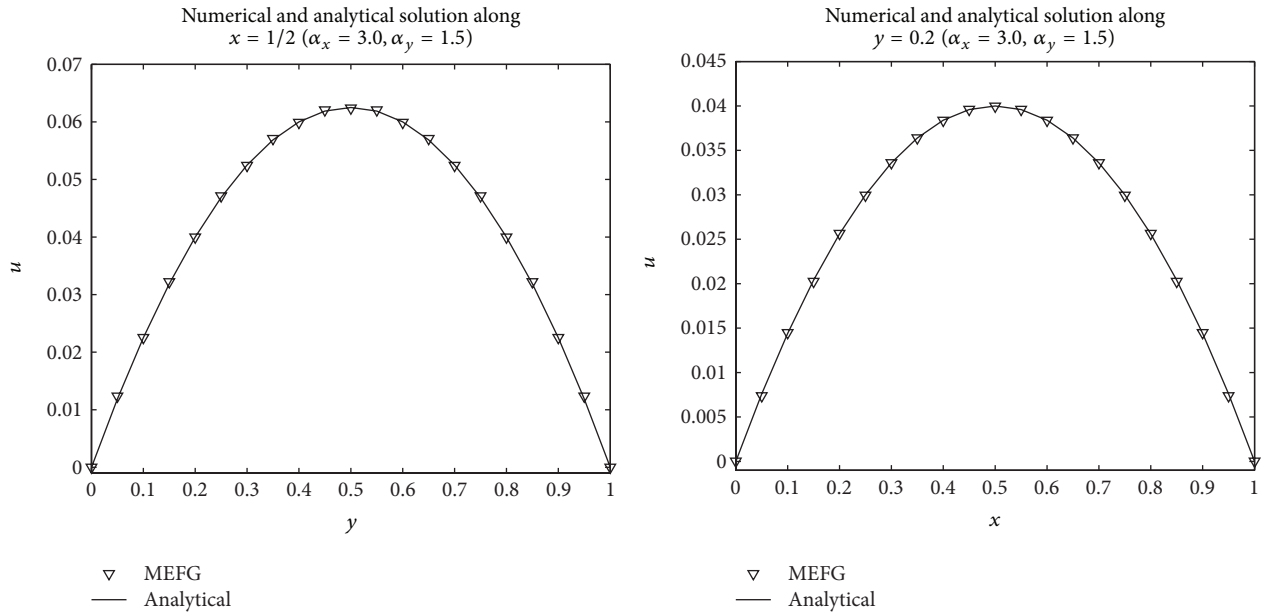


FIGURE 8: Comparison of analytical and MEFG solution.

are also considered here. The computational results are listed in Table 4, where r_1 is the relative error for the case $\alpha_x = \alpha_y = 3.0$ (which corresponds to the nonsingularity problem) and r_2 is the relative error for the cases $\alpha_x = 3.0$ and $\alpha_y = 1.5$ (which corresponds to the singularity problem). The convergence curves are depicted in Figure 10. From Table 4 and Figure 10, we can also see that the MEFG method also possesses convergence.

6. Conclusion and Outlook

In this paper, aspects of MLS approximation with orthogonal basis functions, which are used to construct shape functions in the element-free Galerkin method, are analyzed in detail. It is shown that this method has some advantages; for example, it avoids computing inversion of moment matrices;

the condition number of moment matrix constructed by MLS approximation with orthogonal basis functions can be improved greatly for many problems. But it does not overcome the singularity problem of moment matrix, either. The reason why singularity problem occurs is studied. A novel technique based on matrix triangular process is proposed to solve this problem in detail. Our study relies on monomial basis functions, but it can be extended to any basis functions. Numerical examples are illustrated to show the effectiveness of the new approach. Although the proposed method has been investigated in the EFG method, it is also suitable for other meshfree methods based on MLS approximation.

Appendix

In this appendix, we prove Proposition 1.

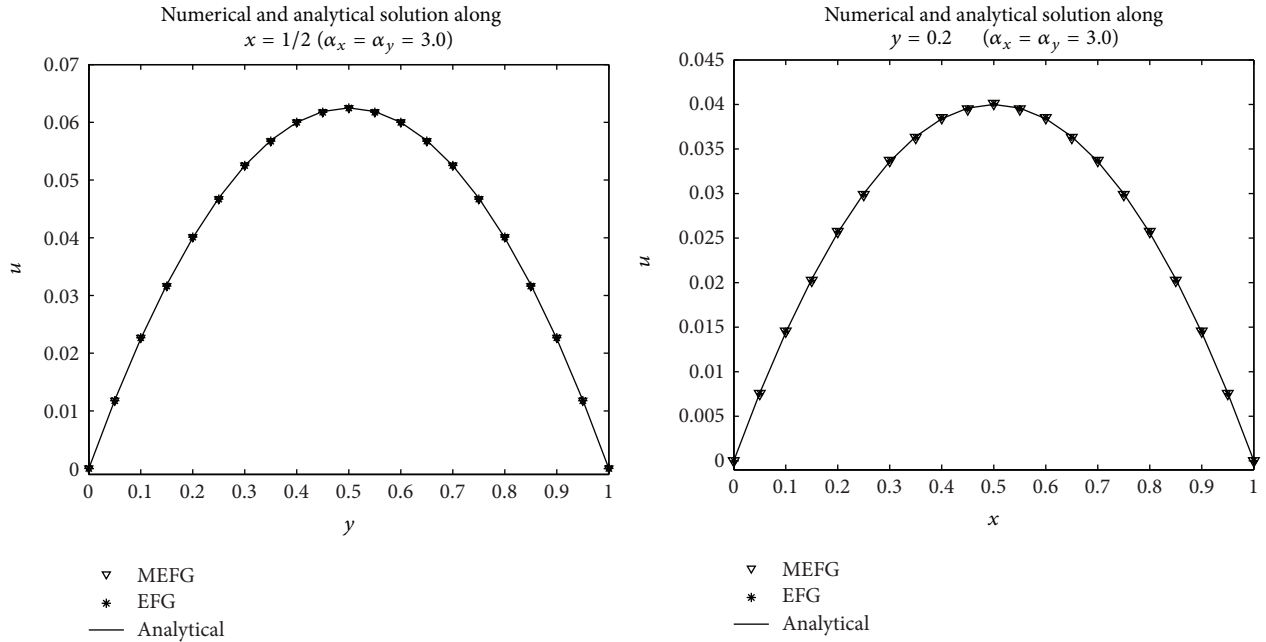


FIGURE 9: Comparison of analytical and numerical solution.

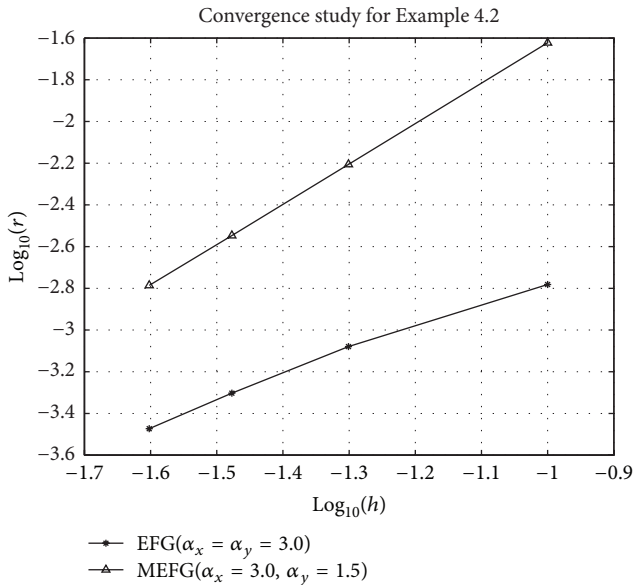


FIGURE 10: Convergence study for Example 4.2.

TABLE 4: Convergence study for Example 4.2.

Node distribution	10×10	20×20	30×30	40×40
Background cells	5×5	10×10	15×15	20×20
r_1	0.0017	$8.3331e-4$	$4.9831e-4$	$3.3669e-4$
r_2	0.0239	0.0062	0.0028	0.0016

Proof. The first part of Proposition 1 is easy to verify since $\mathbf{W}(\mathbf{x})$ is a diagonal matrix. Moreover, from the definition of the weight function $w(\mathbf{x})$, we know that if a set of nodes are in

the support domain of a computational node \mathbf{x} , then $w_i(\mathbf{x}) > 0$. Thus, $\mathbf{W}(\mathbf{x})$ is a positive definite matrix. Because

$$\mathbf{A}(\mathbf{x})_{m \times m} = \mathbf{P}_{m \times n} \mathbf{W}(\mathbf{x})_{n \times n} \mathbf{P}_{n \times m}^T, \quad (\text{A.1})$$

we obtain

$$\begin{aligned} \text{rank}(\mathbf{A}(\mathbf{x})) &\leq \min\{\text{rank}(\mathbf{P}), \text{rank}(\mathbf{W}), \text{rank}(\mathbf{P}^T)\} \\ &= \min\{\text{rank}(\mathbf{P}), \text{rank}(\mathbf{W})\}. \end{aligned} \quad (\text{A.2})$$

If $n < m$, then $\text{rank}(\mathbf{P}) \leq n$ and $\min\{\text{rank}(\mathbf{P}), \text{rank}(\mathbf{W})\}$ is at most n . Therefore, $\text{rank}(\mathbf{A}(\mathbf{x})) \leq n < m$. Thus, $\mathbf{A}(\mathbf{x})$ is a singular matrix. If $n \geq m$, we show that $\mathbf{A}(\mathbf{x})$ is a positive semidefinite matrix. If we further assume that $\text{rank}(\mathbf{P}) = m$, then $\mathbf{A}(\mathbf{x})$ is a positive definite matrix; thus $\mathbf{A}(\mathbf{x})$ is a nonsingular matrix. In fact, for a given nonzero vector $z \in \mathbb{R}^m$

$$z^T \mathbf{A}(\mathbf{x}) z = z^T \mathbf{P} \mathbf{W}(\mathbf{x}) \mathbf{P}^T z = (\mathbf{P}^T z)^T \mathbf{W}(\mathbf{x}) \mathbf{P}^T z \geq 0 \quad (\text{A.3})$$

since $\mathbf{W}(\mathbf{x})$ is a diagonal positive definite matrix. The equality in (A.3) holds if and only if $\mathbf{P}^T z = 0$. This cannot be true when \mathbf{P}^T has full column rank and $z \neq 0$. Thus $z^T \mathbf{A}(\mathbf{x}) z > 0$ always holds for a nonzero vector z . Therefore, $\mathbf{A}(\mathbf{x})$ is a positive definite matrix.

Thus, we complete the proof. \square

Conflict of Interests

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

Acknowledgments

The authors would like to express their great thankfulness to the referees for the comments and constructive suggestions very much, which are valuable in improving the quality of their paper. This work is supported by the National Natural Science Foundation of China under the Grant nos. 11172192 and 11301290. The research of Jun-Liang Dong is supported by The Basic Development Foundation for Mathematics and Statistics (no. 006000542213501), Beijing University of Technology, China.

References

- [1] B. Nayroles, G. Touzot, and P. Villon, "Generalizing the finite element method: diffuse approximation and diffuse elements," *Computational Mechanics*, vol. 10, no. 5, pp. 307–318, 1992.
- [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] Y. Yin, L.-Q. Yao, and Y. Cao, "A 3D shell-like approach using element-free Galerkin method for analysis of thin and thick plate structures," *Acta Mechanica Sinica*, vol. 29, no. 1, pp. 85–98, 2013.
- [4] G.-R. Liu and Y. T. Gu, "A point interpolation method for two-dimensional solids," *International Journal for Numerical Methods in Engineering*, vol. 50, no. 4, pp. 937–951, 2001.
- [5] 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.
- [6] C. Franke and R. Schaback, "Solving partial differential equations by collocation using radial basis functions," *Applied Mathematics and Computation*, vol. 93, no. 1, pp. 73–82, 1998.
- [7] Y. C. Hon and R. Schaback, "On unsymmetric collocation by radial basis functions," *Applied Mathematics and Computation*, vol. 119, no. 2-3, pp. 177–186, 2001.
- [8] S. N. Atluri and T. Zhu, "A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics," *Computational Mechanics*, vol. 22, no. 2, pp. 117–127, 1998.
- [9] G.-R. Liu and Y.-T. Gu, "A meshfree method: meshfree weak-strong (MWS) form method, for 2-D solids," *Computational Mechanics*, vol. 33, no. 1, pp. 2–14, 2003.
- [10] G. R. Liu and Y. T. Gu, *An Introduction to Meshfree Methods and Their Programming*, Springer, Amsterdam, The Netherlands, 2005.
- [11] P. Lancaster and K. Salkauskas, "Surfaces generated by moving least squares methods," *Mathematics of Computation*, vol. 37, no. 155, pp. 141–158, 1981.
- [12] Y.-Y. Lu, T. Belytschko, and L. Gu, "A new implementation of the element free Galerkin method," *Computer Methods in Applied Mechanics and Engineering*, vol. 113, no. 3-4, pp. 397–414, 1994.
- [13] Z. Zhang, K. M. Liew, Y.-M. Cheng, and Y. Y. Lee, "Analyzing 2D fracture problems with the improved element-free Galerkin method," *Engineering Analysis with Boundary Elements*, vol. 32, no. 3, pp. 241–250, 2008.
- [14] 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.
- [15] R.-J. Cheng and K. M. Liew, "Analyzing modified equal width (MEW) wave equation using the improved element-free Galerkin method," *Engineering Analysis with Boundary Elements*, vol. 36, no. 9, pp. 1322–1330, 2012.
- [16] X.-Y. Zhuang and C. Augarde, "Aspects of the use of orthogonal basis functions in the element-free Galerkin method," *International Journal for Numerical Methods in Engineering*, vol. 81, no. 3, pp. 366–380, 2010.
- [17] Y. Cao, L. Q. Yao, and Y. Yin, "The meshless local Petrov-Galerkin method based on the improved moving leastsquares approximation," in *Proceedings of the International Conference on Computational & Experimental Engineering and Sciences*, Nanjing, China, 2011.
- [18] Y.-F. Nie, S. N. Atluri, and C.-W. Zuo, "The optimal radius of the support of radial weights used in moving least squares approximation," *Computer Modeling in Engineering and Sciences*, vol. 12, no. 2, pp. 137–147, 2006.
- [19] N. Sunilkumar and D. Roy, "A smooth finite element method based on reproducing kernel DMS-splines," *Computer Modeling in Engineering and Sciences*, vol. 65, no. 2, pp. 107–153, 2010.
- [20] G.-R. Liu and Y.-T. Gu, "A matrix triangularization algorithm for the polynomial point interpolation method," *Computer Methods in Applied Mechanics and Engineering*, vol. 192, no. 19, pp. 2269–2295, 2003.
- [21] A. Shaw and D. Roy, "A NURBS-based error reproducing kernel method with applications in solid mechanics," *Computational Mechanics*, vol. 40, no. 1, pp. 127–148, 2007.
- [22] Y. Saad, *Iterative Methods for Sparse Linear Systems*, SIAM, Philadelphia, Pa, USA, 2nd edition, 2003.
- [23] 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.
- [24] Y. Cao, L. Q. Yao, and Y. Yin, "New treatment of essential boundary conditions in EFG method by coupling with RPIM," *Acta Mechanica Solida Sinica*, vol. 26, no. 3, pp. 302–316, 2013.