Engineering Analysis with Boundary Elements 83 (2017) 204–216

Contents lists available at ScienceDirect

Engineering Analysis with Boundary Elements

journal homepage: www.elsevier.com/locate/enganabound

A meshless complex variable Galerkin boundary node method for potential and Stokes problems

Yaozong Tang[a], Xiaolin Li[b,c,*]

[a] College of Mathematics and Statistics, Kashgar University, Kashgar 844000, PR China
[b] School of Mathematical Sciences, Chongqing Normal University, Chongqing 400047, PR China
[c] Key Laboratory for Optimization and Control Ministry of Education, Chongqing Normal University, Chongqing 400047, PR China

ARTICLE INFO

Keywords:
Meshless methods
Complex variable moving least squares approximation
Boundary integral equations
Galerkin boundary node method
Error estimates
Stokes flows

ABSTRACT

In this study, combining the boundary integral equations (BIEs) with the complex variable moving least squares (CVMLS) approximation, a symmetric and boundary-only meshless method, the complex variable Galerkin boundary node method (CVGBNM), is developed. Numerical applications and theoretical error estimates of the CVGBNM are derived for BIEs, potential problems and Stokes problems. Finally, numerical examples are given to demonstrate the efficacy of the method.

1. Introduction

Over the past half century, the numerical solutions of many physics and engineering problems have been dominated by mesh-based methods such as the finite element methods and the boundary element methods (BEMs). In these methods, the approximation of unknown variables is related exactly to the geometry of elements. Meshless (or meshfree) methods [1,2], in which the approximation of unknown variables requires only nodes, can overcome the meshing-related drawbacks.

The main difference between mesh-based methods and meshless methods is the way in which the shape function is formulated. The moving least squares (MLS) approximation [3] is extensively used to formulate meshless shape functions. Many meshless methods, such as the element-free Galerkin (EFG) method [4,5], the meshless local Petrov–Galerkin (MLPG) method [6], the boundary node method (BNM) [7,8] and the hybrid boundary node method (HBNM) [9] have been developed using the MLS approximation. Besides, some MLS variants [2], such as the interpolating MLS [10,11], the improved MLS [12,13] and the improved interpolating MLS [14,15] have also been proposed. These MLS approximations have many advantages, such as good smoothness and high computational accuracy. Like other numerical methods, these MLS approximations also have their own deficiencies. The main deficiency is that many nodes are required to form the shape function [2,16], which leads to the increase in computational cost and the decrease in computational efficiency.

To enhance the computational efficiency of the MLS approximation, a complex variable moving least squares (CVMLS) approximation [16] has been developed by introducing the complex variable theory into the MLS approximation. A distinguished feature of the CVMLS approximation is that two-dimensional functions can be approximated by one-dimensional basis functions. Then, the number of coefficients in the approximation function of the CVMLS is less than that of the MLS, and therefore the CVMLS has lower computational cost than the MLS. Some meshless methods, such as the complex variable EFG method [2,17], the complex variable MLPG method [18], and the complex variable boundary element-free method (CVBEFM) [19] have been developed based on the CVMLS approximation. It has been shown in Refs. [2,16–19] that, the numerical error of these CVMLS methods is much less than that of the MLS methods for the same node distribution, and the CVMLS methods need fewer nodes than the MLS-based methods for the same numerical precision. Therefore, the CVMLS methods have greater computational precision and efficiency than the MLS meshless methods.

The functional used in the CVMLS approximation to solve the unknown coefficients is defined directly in a weighted squares form. Based on the CVMLS approximation, an improved CVMLS approximation [20] has been developed by using the complex modulus to define a new functional. Compared with the CVMLS, the improved CVMLS has a more specific physical meaning because of the use of the new functional and the polynomial conjugated basis. By combining the improved CVMLS approximation with the global Galerkin weak form, the improved CVEFG method has been developed [20–22]. Besides, an inter-

* Corresponding author at: School of Mathematical Sciences, Chongqing Normal University, Chongqing 400047, PR China.
E-mail address: lxlmath@163.com (X. Li).

http://dx.doi.org/10.1016/j.enganabound.2017.07.025
Received 30 January 2017; Received in revised form 25 July 2017; Accepted 26 July 2017
0955-7997/

polating CVMLS method [23] has been presented based on the improved CVMLS approximation with a singular weight function.

Recently, a shifted and scaled polynomial basis function has been developed to stabilize the MLS [24]. Theoretical analysis and numerical verification show that the stabilized MLS prevents the instability occurrence [25]. The inherent instability of the interpolating MLS is also studied theoretically and numerically [11]. In this paper, shifted and scaled complex variable basis functions are used to stabilize the CVMLS approximation.

Boundary integral equations (BIEs) and BEMs have been widely used for the numerical solution of boundary value problems. The BNM [7] and the HBNM [9] are boundary-only meshless methods formulated using the MLS approximation and BIEs, while the CVBEFM [19] is a CVMLS-based boundary-only meshless method. Besides, the boundary element-free method (BEFM) [12,26,27] is formulated combining BIEs with the improved MLS approximation. These methods take the advantages of both the BIEs in dimension reduction and the MLS in element elimination. However, since shape functions generated by the MLS and the CVMLS lack the delta function property, it is difficult to impose boundary conditions in these methods. In fact, the implementation of boundary conditions in the BNM and the HBNM adds the number of system equations, while boundary conditions in the BEFM and the CVBEFM are implemented with constraints [14].

The Galerkin boundary node method (GBNM) [28,29] is another boundary-only meshless method that combines the variational form of BIEs with the MLS approximation. In this method, the implementation of boundary conditions does not present any difficulty. Besides, the GBNM can yield symmetric and positive definite system matrix. Numerical applications and theoretical error estimates of the GBNM have been presented for problems in potential theory [28–30] and fluid mechanics [31,32].

As in the BNM, curvilinear coordinates are used in the GBNM to formulate meshless shape functions. In fact, if one attempts to formulate shape functions directly using Cartesian coordinates, the moment matrix involved in the MLS will be singular or ill-conditioned. For example, the matrix is singular whenever the boundary nodes lie on a straight line [7]. Nevertheless, it is burdensome and time-consuming to obtain the curvilinear coordinates for some problems, such as complicated boundary problems and moving boundary problems [33].

In this paper, the CVMLS approximation is introduced into the GBNM to produce a meshless complex variable Galerkin boundary node method (CVGBNM). Details of numerical implementation of the CVGBNM are presented for general BIEs, potential problems and Stokes problems. Based on the error results of the CVMLS approximation, asymptotic error estimates of the CVGBNM are derived.

In the CVGBNM, the CVMLS approximation is used to generate the trial and test functions of the variational form of BIEs. Then, the CVGBNM is expected to have higher computational efficiency than the GBNM. Besides, because Cartesian coordinates can be used directly and easily in the CVMLS to accommodate the curved boundary, the CVGBNM avoids the curvilinear coordinates required in the BNM and the GBNM. Moreover, compared with the BNM and the CVBEFM, boundary conditions in the CVGBNM are applied directly and exactly, and the system matrix in the CVGBNM is symmetric and positive definite. Accordingly, the CVGBNM overcomes the disadvantages of the BNM, the CVBEFM and the GBNM, while possesses the advantages of the three methods.

An outline of this paper is as follows. Section 2 presents and analyzes the CVMLS approximation. Then, a detailed analysis and application of the CVGBNM for general BIEs is given in Section 3, while numerical implementations and asymptotic error estimates of the CVGBNM for potential problems and Stokes problems are presented in Section 4. Finally, numerical examples and conclusions are provided in Sections 5 and 6, respectively.

2. The complex variable moving least squares (CVMLS) approximation

2.1. Formulations

Let $\Omega \subset \mathbb{R}^2$ be a bounded domain with a Lipschitz continuous boundary Γ. A generic point in Ω is denoted by $\mathbf{x} = (x_1, x_2)^{\mathrm{T}}$ or $z = x_1 + ix_2$, where $i = \sqrt{-1}$ is an imaginary number. Let $\{z_I\}_{I=1}^{N}$ be a set of N nodes on Γ. This set is used to define a finite open covering $\{\mathfrak{R}_I\}_{I=1}^{N}$ of Γ composed of N disks $\mathfrak{R}_I$ centered at z_I, where

$$\mathfrak{R}_I = \{\tilde{z} \in \Gamma : |z_I - \tilde{z}| < h_I\}, \quad I = 1, 2, \cdots, N$$

denotes the influence domain of z_I, and h_I is the radius of $\mathfrak{R}_I$.

Let u_1 and u_2 be two real value functions defined on Γ. To obtain the CVMLS approximation of the complex value function $u(z) = u_1 + iu_2$, define the following local approximation

$$u(z) \approx \mathcal{M}u(z, z^*) = \sum_{j=0}^{m} \bar{p}_j(z^*) a_j(z) = \bar{\mathbf{p}}^{\mathrm{T}}(z^*)\mathbf{a}(z), \quad \forall z \in \Gamma \tag{1}$$

where $\mathcal{M}$ is an approximation operator, the point z^* can either be the evaluation point z or a nodal point z_I in the influence domain of z, and

$$\bar{\mathbf{p}}(z) = [\bar{p}_0(z), \bar{p}_1(z), \bar{p}_2(z), ..., \bar{p}_m(z)]^{\mathrm{T}} = [1, \bar{z}, \bar{z}^2, ..., \bar{z}^m]^{\mathrm{T}}$$

is the basis function vector which equals the conjugate of the conventional basis function vector

$$\mathbf{p}(z) = [p_0(z), p_1(z), p_2(z), ..., p_m(z)]^{\mathrm{T}} = [1, z, z^2, ..., z^m]^{\mathrm{T}} \tag{2}$$

The vector $\mathbf{a}(z) = [a_0(z), a_1(z), ..., a_m(z)]^{\mathrm{T}}$ in Eq. (1) can be obtained by minimizing the functional

$$\begin{aligned} \mathcal{J}(z) &= \sum_{I \in \wedge(z)} w_I(z) \left| \mathcal{M}u(z, z_I) - u_I \right|^2 \\ &= \sum_{I \in \wedge(z)} w_I(z) \left(\sum_{j=0}^{m} \bar{p}_j(z_I) a_j(z) - u_I \right) \overline{\left(\sum_{j=0}^{m} \bar{p}_j(z_I) a_j(z) - u_I \right)} \end{aligned}$$

as [2,20–22]

$$\mathbf{a}(z) = \mathbf{A}^{-1}(z)\mathbf{B}(z)\mathbf{u} \tag{3}$$

where $\wedge(z) \triangleq \{I_1, I_2, \cdots, I_\tau\} \subseteq \{1, 2, \cdots, N\}$ is the set of the global sequence numbers of nodes whose influence domains cover the point z, and $\mathbf{u} = [u_{I_1}, u_{I_2}, \cdots, u_{I_\tau}]^{\mathrm{T}}$ with $u_I = u_{1I} + iu_{2I}$ are the nodal values of u at the node z_I. Besides,

$$\mathbf{A}(z) = \mathbf{P}^{\mathrm{T}}\mathbf{W}(z)\bar{\mathbf{P}} \tag{4}$$

$$\mathbf{B}(z) = \mathbf{P}^{\mathrm{T}}\mathbf{W}(z) \tag{5}$$

$$\mathbf{P} = \left[\mathbf{p}\left(z_{I_1}\right), \mathbf{p}\left(z_{I_2}\right), \cdots, \mathbf{p}\left(z_{I_\tau}\right)\right]^{\mathrm{T}} \tag{6}$$

$$\mathbf{W}(z) = \mathrm{diag}\left(w_{I_1}(z), w_{I_2}(z), \cdots, w_{I_\tau}(z)\right)$$

Moreover,

$$w_I(z) = \varphi\left(\frac{|z - z_I|}{h_I}\right), \quad I = 1, 2, \cdots, N \tag{7}$$

are weight functions, where the function φ is nonnegative, γth times continuously differentiable, and its derivatives up to order γ are bounded. Here, γ is a positive integer. The weight functions $w_I(z)$ also satisfy $w_I(z) \in C_0^{\gamma}(\mathfrak{R}_I)$. Clearly, we have

$$z \in \mathfrak{R}_I \quad \Leftrightarrow \quad I \in \wedge(z) \quad \Leftrightarrow \quad w_I(z) > 0, \quad \forall z \in \Gamma, \quad I = 1, 2, \cdots, N \tag{8}$$

Substituting Eq. (3) into Eq. (1) yields

$$u(z) \approx \mathcal{M}u(z, z^*) = \mathbf{p}^{\mathrm{T}}(z^*)\mathbf{A}^{-1}(z)\mathbf{B}(z)\mathbf{u} \tag{9}$$