A new complex variable element-free Galerkin method for two-dimensional potential problems*

Cheng Yu-Min(程玉民)†, Wang Jian-Fei(王健菲), and Bai Fu-Nong(白福浓)

[a] *Shanghai Institute of Applied Mathematics and Mechanics, Shanghai University, Shanghai* 200072, *China*

[b] *Shanghai Key Laboratory of Mechanics in Energy Engineering, Shanghai University, Shanghai* 200072, *China*

(Received 23 December 2011; revised manuscript received 23 February 2012)

In this paper, based on the element-free Galerkin (EFG) method and the improved complex variable moving least-square (ICVMLS) approximation, a new meshless method, which is the improved complex variable element-free Galerkin (ICVEFG) method, for two-dimensional potential problems is presented. In the method, the integral weak form of control equations is employed, and the Lagrange multiplier is used to apply the essential boundary conditions. Then the corresponding formulas of the ICVEFG method for two-dimensional potential problems are obtained. Compared with the complex variable moving least-square (CVMLS) approximation proposed by Cheng, the functional in the ICVMLS approximation has an explicit physical meaning. Furthermore, the ICVEFG method has greater computational precision and efficiency. Three numerical examples are given to show the validity of the proposed method.

Keywords: meshless method, improved complex variable moving least-square approximation, improved complex variable element-free Galerkin method, potential problem

PACS: 02.60.Cb, 02.60.Lj, 02.30.Em **DOI:** 10.1088/1674-1056/21/9/090203

1. Introduction

Meshless methods have been used in a variety of scientific and engineering problems.[1−13] The methods only employ the information at nodes, and do not need the subdivision of the solution domain into many smaller regions. Then they have simple preprocessings and a high computing precision.

The moving least-square (MLS) approximation[14] is one of the methods used to form the shape function in the meshless method, and is most widely used. It has been successfully used for the simulation of a broad range of systems in the last decades.[15−19] The shape function formed with the MLS approximation is able to obtain a quite precise solution. However, the MLS approximation is the approximation of a scalar function, thus the meshless method derived requires a lot of nodes in the domain, in addition, the final algebra equation system is sometimes ill-conditioned.

In order to enhance the computational efficiency, a new approximation method, the complex variable moving least-square (CVMLS) approximation, has been developed.[20] The CVMLS approximation is the approximation of a vector function, and with this approximation, the trial function of a two-dimensional problem is formed with a one-dimensional basis function. The number of unknown coefficients in the trial function of the CVMLS approximation is less than that in the trial function of the MLS approximation, and we can thus select fewer nodes in the meshless method formed from the CVMLS approximation than that from the MLS approximation without loss of precision.

However, the physical meaning of the functional in the CVMLS approximation is not clear. To overcome this insufficiency, the improved complex variable moving least-square (ICVMLS) approximation has been discussed,[21] and the functional in this new method has an explicit physical meaning.

In this paper, based on the element-free Galerkin (EFG) method and the ICVMLS approximation, the improved complex variable element-free Galerkin (ICVEFG) method for two-dimensional (2D) potential problems is presented. In the method, the integral weak form of control equations is employed, and the Lagrange multiplier is used to apply the

*Project supported by the National Natural Science Foundation of China (Grant No. 11171208), the Shanghai Leading Academic Discipline Project, China (Grant No. S30106), and the Innovation Fund Project for Graduate Student of Shanghai University, China (Grant No. SHUCX112359).

†Corresponding author. E-mail: ymcheng@shu.edu.cn

© 2012 Chinese Physical Society and IOP Publishing Ltd

essential boundary conditions. Then the corresponding formulas of the ICVEFG method for two-dimensional potential problems are obtained. Compared with the CVMLS approximation proposed by Cheng, in the ICVMLS approximation, the functional has an explicit physical meaning. Furthermore, the ICVEFG method has greater computational precision and efficiency than the complex variable element-free Galerkin (CVEFG) method. Three numerical examples are given to show the validity of the ICVEFG method.

2. Improved complex variable moving least-square approximation

Compared with the CVMLS approximation proposed by Cheng,[20] a new functional J, which has an explicit mathematical or physical meaning, is introduced in the ICVMLS approximation.

The trial function is defined as

$$\begin{aligned} u^{\rm h}(z) &= u_1^{\rm h}(z) + {\rm i}u_2^{\rm h}(z) = \sum_{i=1}^{m} \bar{p}_i(z)a_i(z) \\ &= \bar{\boldsymbol{p}}^T(z)\boldsymbol{a}(z), \quad z = x_1 + {\rm i}x_2 \in \Omega, \end{aligned} \tag{1}$$

where $u^{\rm h}(z)$ is the approximation to function $u(z)$, $\bar{\boldsymbol{p}}^{\rm T}(z) = (\bar{p}_1(z), \bar{p}_2(z), \ldots, \bar{p}_m(z))$ is the basis function vector, which equals the conjugate of the basis function vector $\boldsymbol{p}^{\rm T}(z) = (p_1(z), p_2(z), \ldots, p_m(z))$, $\boldsymbol{a}^{\rm T}(z) = (a_1(z), a_2(z), \ldots, a_m(z))$ is the corresponding coefficient vector, and m is the number of terms in the basis function vector.

In general, the linear and the quadratic basis function vectors in 2D are given respectively by

$$\bar{\boldsymbol{p}}^{\rm T} = (1, \bar{z}), \tag{2}$$

$$\bar{\boldsymbol{p}}^{\rm T} = (1, \bar{z}, \bar{z}^2). \tag{3}$$

The local approximation at point z is defined as

$$u^{\rm h}(z, \hat{z}) = \sum_{i=1}^{m} \bar{p}_i(\hat{z}) \cdot a_i(z) = \bar{\boldsymbol{p}}^{\rm T}(\hat{z})\boldsymbol{a}(z), \tag{4}$$

where $\hat{z}$ is the node in the local domain of z. The $a_i(z)$ $(i = 1, 2, \ldots, m)$ is determined by minimizing the following functional:

$$\begin{aligned} J &= \sum_{I=1}^{n} w(z - z_I) \left| u^{\rm h}(z, z_I) - u(z_I) \right|^2 \\ &= \sum_{I=1}^{n} w(z - z_I) \Big(\sum_{i=1}^{m} \bar{p}_i(z_I) \cdot a_i(z) - u(z_I) \Big) \\ &\quad \times \overline{\Big(\sum_{i=1}^{m} \bar{p}_i(z_I) \cdot a_i(z) - u(z_I) \Big)}, \end{aligned} \tag{5}$$

where $w(z - z_I)$ is a weight function with compact support, z_I $(I = 1, 2, \ldots, n)$ is the node in a compact support domain of point z, and

$$u(z_I) = u_1(z_I) + {\rm i}u_2(z_I). \tag{6}$$

For functions $f(z)$ and $g(z)$, define inner product

$$(f, g)_z = \sum_{I=1}^{n} w(z - z_I) f(z_I) \bar{g}(z_I), \quad \forall f, g \in C^0(\bar{\Omega}), \tag{7}$$

where $\bar{g}(z_I)$ is the conjugate of $g(z_I)$, $\bar{\Omega} = \Omega \cup \partial\Omega$, and $C^0(\bar{\Omega})$ is the set of all continuous functions in domain $\bar{\Omega}$. Correspondingly, the norm is

$$||f||_z = [(f, f)_z]^{1/2}. \tag{8}$$

Equation (5) can then be expressed as

$$\begin{aligned} J &= (\bar{\boldsymbol{P}}\boldsymbol{a}(z) - \boldsymbol{u}^*(z))^{\rm T} \boldsymbol{W}(z) \overline{(\bar{\boldsymbol{P}}\boldsymbol{a}(z) - \boldsymbol{u}^*(z))} \\ &= \left\| \bar{\boldsymbol{p}}^{\rm T}(z) \cdot \boldsymbol{a}(z) - \boldsymbol{u} \right\|_z^2, \end{aligned} \tag{9}$$

where

$$\bar{\boldsymbol{P}} = \begin{bmatrix} \bar{p}_1(z_1) & \bar{p}_2(z_1) & \ldots & \bar{p}_m(z_1) \\ \bar{p}_1(z_2) & \bar{p}_2(z_2) & \ldots & \bar{p}_m(z_2) \\ \vdots & \vdots & \ddots & \vdots \\ \bar{p}_1(z_n) & \bar{p}_2(z_n) & \ldots & \bar{p}_m(z_n) \end{bmatrix}_{n \times m}, \tag{10}$$

$$\boldsymbol{W}(z) = \begin{bmatrix} w(z - z_1) & 0 & \ldots & 0 \\ 0 & w(z - z_2) & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & w(z - z_n) \end{bmatrix}, \tag{11}$$

$$\boldsymbol{u}^* = (u(z_1), u(z_2), \ldots, u(z_n))^{\rm T} = \boldsymbol{Q}\boldsymbol{u}, \tag{12}$$

$$\begin{aligned} \boldsymbol{u} = (&u_1(z_1), u_2(z_1), u_1(z_2), u_2(z_2), \ldots, \\ &u_1(z_n), u_2(z_n))^{\rm T}, \end{aligned} \tag{13}$$

$$\boldsymbol{Q} = \begin{bmatrix} 1 & {\rm i} & 0 & 0 & 0 & 0 & \ldots & 0 & 0 \\ 0 & 0 & 1 & {\rm i} & 0 & 0 & \ldots & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & {\rm i} & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \ldots & 1 & {\rm i} \end{bmatrix}_{n \times 2n}. \tag{14}$$

Equation (5) can be also written as

$$J = \|a_1\bar{p}_1 + a_2\bar{p}_2 + \ldots + a_m\bar{p}_m - u\|_z^2, \tag{15}$$

where $\bar{p}_i = \bar{p}_i(z)$, $i = 1, 2, \ldots, m$.

Let $u^{(1)}$ be the projection of u in $\mathrm{span}(\bar{p}_1, \bar{p}_2, \ldots, \bar{p}_m)$, obviously,

$$J^{(1)} = \left\| u^{(1)} - u \right\|_z^2 \tag{16}$$

is the minimum of functional J.

The function u could be decomposed orthogonally as

$$u = u^{(1)} + u^{(2)}, \tag{17}$$

where $u^{(2)} \perp \mathrm{span}(\bar{p}_1, \bar{p}_2, \ldots, \bar{p}_m)$. Obviously,

$$(\bar{p}_i, u^{(2)})_z = 0, \quad (i = 1, 2, \ldots, m), \tag{18}$$

then, we have

$$(\bar{p}_i, u)_z = (\bar{p}_i, u^{(1)})_z + (\bar{p}_i, u^{(2)})_z = (\bar{p}_i, u^{(1)})_z, \quad (i = 1, 2, \ldots, m), \tag{19}$$

and

$$\begin{aligned}(\bar{p}_i, u)_z &= (\bar{p}_i, u^{(1)})_z \\ &= (\bar{p}_i, a_1\bar{p}_1 + a_2\bar{p}_2 + \cdots + a_m\bar{p}_m)_z. \end{aligned} \tag{20}$$

The matrix form of Eq. (20) is

$$\begin{bmatrix} (\bar{p}_1, \bar{p}_1)_z & (\bar{p}_1, \bar{p}_2)_z & \cdots & (\bar{p}_1, \bar{p}_m)_z \\ (\bar{p}_2, \bar{p}_1)_z & (\bar{p}_2, \bar{p}_2)_z & \cdots & (\bar{p}_2, \bar{p}_m)_z \\ \vdots & \vdots & \ddots & \vdots \\ (\bar{p}_m, \bar{p}_1)_z & (\bar{p}_m, \bar{p}_2)_z & \cdots & (\bar{p}_m, \bar{p}_m)_z \end{bmatrix} \times \begin{bmatrix} \bar{a}_1(z) \\ \bar{a}_2(z) \\ \vdots \\ \bar{a}_m(z) \end{bmatrix} = \begin{bmatrix} (\bar{p}_1, u)_z \\ (\bar{p}_2, u)_z \\ \vdots \\ (\bar{p}_m, u)_z \end{bmatrix}, \tag{21}$$

i.e.,

$$(\bar{\boldsymbol{P}}^{\mathrm{T}}\boldsymbol{W}(z)\boldsymbol{P})\bar{\boldsymbol{a}}(z) = \bar{\boldsymbol{P}}^{\mathrm{T}}\boldsymbol{W}(z)\bar{\boldsymbol{u}}^*. \tag{22}$$

The conjugate of Eq. (22) can be written as

$$(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{W}(z)\bar{\boldsymbol{P}})\boldsymbol{a}(z) = \boldsymbol{P}^{\mathrm{T}}\boldsymbol{W}(z)\boldsymbol{u}^*, \tag{23}$$

then we obtain

$$\boldsymbol{a}(z) = \boldsymbol{C}^{-1}(z)\boldsymbol{B}(z)\boldsymbol{u}^*, \tag{24}$$

where

$$\boldsymbol{C}(z) = \boldsymbol{P}^{\mathrm{T}}\boldsymbol{W}(z)\bar{\boldsymbol{P}}, \tag{25}$$

$$\boldsymbol{B}(z) = \boldsymbol{P}^{\mathrm{T}}\boldsymbol{W}(z). \tag{26}$$

Combining Eq. (24) with Eq. (4), the expression for the local approximation $u^{\mathrm{h}}(z, \hat{z})$ is obtained, namely,

$$u^{\mathrm{h}}(z, \hat{z}) = \boldsymbol{\Phi}(z)\boldsymbol{u}^* = \sum_{I=1}^{n} \Phi_I(z)u(z_I), \tag{27}$$

where $\boldsymbol{\Phi}(z)$ is called the shape function, and

$$\begin{aligned}\boldsymbol{\Phi}(z) &= (\Phi_1(z), \Phi_2(z), \ldots, \Phi_n(z)) \\ &= \bar{\boldsymbol{p}}^{\mathrm{T}}(z)\boldsymbol{C}^{-1}(z)\boldsymbol{B}(z). \end{aligned} \tag{28}$$

From Eq. (27), we have

$$u_1^{\mathrm{h}}(z) = \mathrm{Re}[\boldsymbol{\Phi}(z)\boldsymbol{u}^*] = \mathrm{Re}\left[\sum_{I=1}^{n} \Phi_I(z)u(z_I)\right], \tag{29}$$

$$u_2^{\mathrm{h}}(z) = \mathrm{Im}[\boldsymbol{\Phi}(z)\boldsymbol{u}^*] = \mathrm{Im}\left[\sum_{I=1}^{n} \Phi_I(z)u(z_I)\right]. \tag{30}$$

In the ICVMLS approximation, the trial function of a two-dimensional problem is formed with a one-dimensional basis function; therefore, its number of unknown coefficients is less than that in the MLS approximation. In the MLS approximation, the linear basis is $\boldsymbol{p}^{\mathrm{T}} = (1, x_1, x_2)$, and three unknown coefficients are included; whereas in the ICVMLS approximation, $\bar{\boldsymbol{p}}^{\mathrm{T}} = (1, \bar{z})$, and two unknown coefficients are included. Moreover, in terms of quadratic basis, in the MLS approximation, $\boldsymbol{p}^{\mathrm{T}} = (1, x_1, x_2, x_1^2, x_1x_2, x_2^2)$, and six unknown coefficients are included; whereas in the ICVMLS approximation, $\bar{\boldsymbol{p}}^{\mathrm{T}} = (1, \bar{z}, \bar{z}^2)$, and three unknown coefficients are included. In this manner, for an arbitrary point in the domain, we need fewer nodes with domains of influence that cover the point, and thus we can require fewer nodes in the whole domain.

Compared with the CVMLS approximation presented by Cheng, the above-mentioned ICVMLS approximation has not only all the excellent features of the former but also a more explicit physical meaning. From the ICVMLS approximation, a distinct expression for the approximation function is obtained. Because the polynomial's conjugated basis is adopted to make the shape function, the method has higher computing precision and efficiency.

3. The ICVEFG method for two-dimensional potential problems

In this section, we present the ICVEFG method for the two-dimensional Poisson's equation

$$\frac{\partial^2 u(\boldsymbol{x})}{\partial x_1^2}+\frac{\partial^2 u(\boldsymbol{x})}{\partial x_2^2}=Q(\boldsymbol{x}),\quad (\boldsymbol{x}\in\Omega), \tag{31}$$

with boundary conditions of the Dirichlet type, i.e.,

$$u(\boldsymbol{x})=\hat{u}(\boldsymbol{x}),\quad (\boldsymbol{x}\in\Gamma_u), \tag{32}$$

and the Neumann type, i.e.,

$$q(\boldsymbol{x})=\frac{\partial u(\boldsymbol{x})}{\partial \boldsymbol{n}}=\hat{q},\quad (\boldsymbol{x}\in\Gamma_q), \tag{33}$$

where the potential $u(\boldsymbol{x})$ is an unknown function, $Q(\boldsymbol{x})$ is a known function, $\boldsymbol{n}$ is the unit outward normal to boundary Γ, $\hat{u}(\boldsymbol{x})$ is the prescribed value of function $u(\boldsymbol{x})$ over the essential boundary Γ_u, and $\hat{q}$ is the prescribed normal derivative of $u(\boldsymbol{x})$ over the natural boundary Γ_q, and $\Gamma=\Gamma_u\cup\Gamma_q$, $\Gamma_u\cap\Gamma_q=\phi$.

Let

$$\begin{aligned}\Pi=&\int_\Omega\left[\frac{1}{2}\left(\frac{\partial u}{\partial x_1}\right)^2+\frac{1}{2}\left(\frac{\partial u}{\partial x_2}\right)^2-uQ\right]\mathrm{d}\Omega\\&-\int_{\Gamma_q}u\cdot\hat{q}\mathrm{d}\Gamma.\end{aligned} \tag{34}$$

The potential $u(\boldsymbol{x})$ that makes $\delta\Pi$ reach zero is exactly the solution satisfying control equation (31) and boundary condition (33). The Lagrange multiplier is used to apply the essential boundary conditions. Then a modified Π^* is obtained as

$$\Pi^*=\Pi+\int_{\Gamma_u}\lambda(u-\hat{u})\mathrm{d}\Gamma, \tag{35}$$

where λ is the Lagrange multiplier. The stationary condition of Π^* is

$$\begin{aligned}\delta\Pi^*=&\delta\Pi+\int_{\Gamma_u}\delta\lambda(u-\hat{u})\mathrm{d}\Gamma\\&+\int_{\Gamma_u}\lambda\delta u\mathrm{d}\Gamma=0.\end{aligned} \tag{36}$$

Substituting Eq. (34) into Eq. (36), we have the following integral weak form:

$$\begin{aligned}&\int_\Omega\delta(\boldsymbol{L}u)^{\mathrm{T}}(\boldsymbol{L}u)\mathrm{d}\Omega-\int_\Omega\delta\boldsymbol{u}Q\mathrm{d}\Omega-\int_{\Gamma_q}\delta u\hat{q}\mathrm{d}\Gamma\\&+\int_{\Gamma_u}\delta\lambda(u-\hat{u})\mathrm{d}\Gamma+\int_{\Gamma_u}\lambda\delta u\mathrm{d}\Gamma=0,\end{aligned} \tag{37}$$

where

$$\boldsymbol{L}=\begin{bmatrix}\frac{\partial}{\partial x_1}\\ \frac{\partial}{\partial x_2}\end{bmatrix}. \tag{38}$$

From Eq. (29), $u(\boldsymbol{x})$ can be expressed as

$$\begin{aligned}u(\boldsymbol{x})=&\mathrm{Re}[\boldsymbol{\Phi}(z)\boldsymbol{u}^*]\\=&\mathrm{Re}\left[\sum_{I=1}^{n}\Phi_I(z)u(z_I)\right]=\tilde{\boldsymbol{\Phi}}(z)\boldsymbol{u}^*,\end{aligned} \tag{39}$$

$$\tilde{\boldsymbol{\Phi}}(z)=(\tilde{\Phi}_1(z),\tilde{\Phi}_2(z),\ldots,\tilde{\Phi}_n(z)), \tag{40}$$

$$\tilde{\Phi}_I(z)=\mathrm{Re}[\Phi_I(z)]. \tag{41}$$

Moreover,

$$\begin{aligned}\boldsymbol{L}u(\boldsymbol{x})=&\boldsymbol{L}\sum_{I=1}^{n}\mathrm{Re}[\Phi_I(z)]u(z_I)\\=&\sum_{I=1}^{n}\begin{bmatrix}\frac{\partial}{\partial x_1}\\ \frac{\partial}{\partial x_2}\end{bmatrix}\mathrm{Re}[\Phi_I(z)]u(z_I)=\boldsymbol{B}(z)\boldsymbol{u}^*,\end{aligned} \tag{42}$$

where

$$\boldsymbol{B}(z)=(\boldsymbol{B}_1(z),\boldsymbol{B}_2(z),\ldots,\boldsymbol{B}_n(z)), \tag{43}$$

$$\boldsymbol{B}_I(z)=\begin{bmatrix}\mathrm{Re}[\Phi_{I,1}(z)]\\ \mathrm{Re}[\Phi_{I,2}(z)]\end{bmatrix}. \tag{44}$$

In order to obtain the discrete system equations, λ is expressed in the Lagrange interpolation as

$$\lambda(\boldsymbol{x})=\sum_{I=1}^{n_\lambda}N_I(s)\lambda_I=\boldsymbol{N}\boldsymbol{\lambda},\quad (x\in\Gamma_u), \tag{45}$$

where n_λ is the node number on the boundary, $N_I(s)$ is a Lagrange interpolation, s is the arc length along the boundary, and

$$\boldsymbol{N}=(N_1,N_2,\ldots,N_{n_\lambda}), \tag{46}$$

$$\boldsymbol{\lambda}^{\mathrm{T}}=(\lambda_1,\lambda_2,\ldots,\lambda_{n_\lambda}). \tag{47}$$

From Eq. (45), we have

$$\delta\lambda(\boldsymbol{x})=\sum_{I=1}^{n_\lambda}N_I(s)\delta\lambda_I=\delta\boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{N}^{\mathrm{T}}. \tag{48}$$

Substituting Eqs. (39), (42), (45), and (48) into Eq. (37) yields

$$\begin{aligned}&\int_\Omega\delta[\boldsymbol{B}(z)\boldsymbol{u}^*]^{\mathrm{T}}[\boldsymbol{B}(z)\boldsymbol{u}^*]\mathrm{d}\Omega\\&-\int_\Omega\delta[\tilde{\boldsymbol{\Phi}}(z)\boldsymbol{u}^*]Q\mathrm{d}\Omega-\int_{\Gamma_q}\delta[\tilde{\boldsymbol{\Phi}}(z)\boldsymbol{u}^*]\hat{q}\mathrm{d}\Gamma\\&+\int_{\Gamma_u}(\delta\boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{N}^{\mathrm{T}})[(\tilde{\boldsymbol{\Phi}}(z)\boldsymbol{u}^*-\hat{u}]\mathrm{d}\Gamma\end{aligned}$$

$$+\int_{\Gamma_u}(\boldsymbol{N}\boldsymbol{\lambda})\delta[\tilde{\boldsymbol{\Phi}}(z)\boldsymbol{u}^*]\mathrm{d}\Gamma=0. \tag{49}$$

The integral terms in Eq. (49) will be analyzed respectively. The first term of Eq. (49) is

$$\begin{aligned}&\int_{\Omega}\delta[\boldsymbol{B}(z)\boldsymbol{u}^*]^{\mathrm{T}}[\boldsymbol{B}(z)\boldsymbol{u}^*]\mathrm{d}\Omega\\&=\delta\boldsymbol{u}^{*\mathrm{T}}\Big[\int_{\Omega}\boldsymbol{B}^{\mathrm{T}}(z)\boldsymbol{B}(z)\mathrm{d}\Omega\Big]\boldsymbol{u}^*=\delta\boldsymbol{u}^{*\mathrm{T}}\boldsymbol{K}\boldsymbol{u}^*,\end{aligned} \tag{50}$$

where

$$\boldsymbol{K}=\int_{\Omega}\boldsymbol{B}^{\mathrm{T}}(z)\boldsymbol{B}(z)\mathrm{d}\Omega, \tag{51}$$

$\boldsymbol{K}$ is a $n_{\mathrm{t}}\times n_{\mathrm{t}}$ matrix with n_{t} being the total number of nodes in Ω. The second term of Eq. (49) is

$$\begin{aligned}&\int_{\Omega}\delta[\tilde{\boldsymbol{\Phi}}(z)\boldsymbol{u}^*]Q\mathrm{d}\Omega\\&=\delta\boldsymbol{u}^{*\mathrm{T}}\int_{\Omega}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}(z)Q\mathrm{d}\Omega=\delta\boldsymbol{u}^{*\mathrm{T}}\boldsymbol{F}^{(1)},\end{aligned} \tag{52}$$

where $\boldsymbol{F}^{(1)}$ is a load matrix , and

$$\boldsymbol{F}^{(1)}=\int_{\Omega}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}(z)Q\mathrm{d}\Omega. \tag{53}$$

The third term of Eq. (49) is

$$\begin{aligned}&\int_{\Gamma_q}\delta[\tilde{\boldsymbol{\Phi}}(z)\boldsymbol{u}^*]\hat{q}\mathrm{d}\Gamma\\&=\delta\boldsymbol{u}^{*\mathrm{T}}\int_{\Gamma_q}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}(z)\hat{q}\mathrm{d}\Gamma=\delta\boldsymbol{u}^{*\mathrm{T}}\boldsymbol{F}^{(2)},\end{aligned} \tag{54}$$

where $\boldsymbol{F}^{(2)}$ is a load matrix, and

$$\boldsymbol{F}^{(2)}=\int_{\Gamma_q}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}(z)\hat{q}\mathrm{d}\Gamma. \tag{55}$$

The fourth term of Eq. (49) is

$$\begin{aligned}&\int_{\Gamma_u}(\delta\boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{N}^{\mathrm{T}})[\tilde{\boldsymbol{\Phi}}(z)\boldsymbol{u}^*-\hat{u}]\mathrm{d}\Gamma\\&=\delta\boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{G}^{\mathrm{T}}\boldsymbol{u}^*-\delta\boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{r},\end{aligned} \tag{56}$$

where

$$\boldsymbol{r}=\int_{\Gamma_u}\boldsymbol{N}^{\mathrm{T}}\hat{u}\mathrm{d}\Gamma, \tag{57}$$

$$\boldsymbol{G}=\int_{\Gamma_u}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}(z)\boldsymbol{N}\mathrm{d}\Gamma. \tag{58}$$

The fifth term of Eq. (49) is

$$\begin{aligned}&\int_{\Gamma_u}(\boldsymbol{N}\boldsymbol{\lambda})\delta[\tilde{\boldsymbol{\Phi}}(z)\boldsymbol{u}^*]\mathrm{d}\Gamma\\&=\delta\boldsymbol{u}^{*\mathrm{T}}\Big[\int_{\Gamma_u}\tilde{\boldsymbol{\Phi}}^{\mathrm{T}}(z)\boldsymbol{N}\mathrm{d}\Gamma\Big]\boldsymbol{\lambda}=\delta\boldsymbol{u}^{*\mathrm{T}}\boldsymbol{G}\boldsymbol{\lambda}.\end{aligned} \tag{59}$$

Substituting Eq. (50), (52), (54), (56), and (59) into Eq. (49) yields

$$\delta\boldsymbol{u}^{*\mathrm{T}}[\boldsymbol{K}\boldsymbol{u}^*+\boldsymbol{G}\boldsymbol{\lambda}-\boldsymbol{F}]+\delta\boldsymbol{\lambda}^{\mathrm{T}}[\boldsymbol{G}^{\mathrm{T}}\boldsymbol{u}^*-\boldsymbol{r}]=0, \tag{60}$$

where

$$\boldsymbol{F}=\boldsymbol{F}^{(1)}+\boldsymbol{F}^{(2)}. \tag{61}$$

Because of the randomicity of $\delta\boldsymbol{u}^{*\mathrm{T}}$ or $\delta\boldsymbol{\lambda}^{\mathrm{T}}$, we have the final linear system

$$\boldsymbol{K}\boldsymbol{u}^*+\boldsymbol{G}\boldsymbol{\lambda}-\boldsymbol{F}=0, \tag{62}$$

$$\boldsymbol{G}^{\mathrm{T}}\boldsymbol{u}^*-\boldsymbol{r}=0. \tag{63}$$

Equations (62) and (63) can be expressed as

$$\begin{bmatrix}\boldsymbol{K}&\boldsymbol{G}\\\boldsymbol{G}^{\mathrm{T}}&\boldsymbol{0}\end{bmatrix}\begin{bmatrix}\boldsymbol{u}^*\\\boldsymbol{\lambda}\end{bmatrix}=\begin{bmatrix}\boldsymbol{F}\\\boldsymbol{r}\end{bmatrix}. \tag{64}$$

Thus the ICVEFG method for two-dimensional potential problems is presented.

4. Numerical examples

Three examples are solved with the ICVEFG method in this section. The node distribution for each example is regular. And the linear basis function and the cubic spline weight function are employed. For the purpose of error estimation and convergence studies, we define the relative error

$$e=\sum_i\left|\frac{u_i^{\mathrm{num}}-u_i^{\mathrm{exact}}}{u_i^{\mathrm{exact}}}\right|^2, \tag{65}$$

where u_i^{num} and u_i^{exact} are the numerical and the exact solutions, respectively.

4.1. Laplace's equation with mixed boundary conditions on a rectangle

The first example is a 2D Laplace's equation with mixed boundary conditions. For a steady temperature field on a rectangular domain, the government equation is

$$\begin{aligned}\nabla^2T&=\frac{\partial^2T}{\partial x_1^2}+\frac{\partial^2T}{\partial x_2^2}=0,\\&x_1\in[0,5],\quad x_2\in[0,10].\end{aligned} \tag{66}$$

The boundary conditions are

$$T(x_1,0)=0,\quad 0<x_1<5, \tag{67}$$

$$T(0,x_2)=0,\quad 0<x_2<10, \tag{68}$$

$$T(x_1,10)=100\sin(\pi x_1/10),\quad 0<x_1<5, \tag{69}$$

$$\frac{\partial T(5,x_2)}{\partial x_1}=0, \quad 0<x_2<10. \tag{70}$$

The analytical solution of this problem is

$$T(x_1,x_2) = \frac{100\sin(\pi x_1/10)\mathrm{sh}(\pi x_2/10)}{\mathrm{sh}(\pi)}. \tag{71}$$

The node distribution on domain $\Omega = [0,5]\times[0,10]$ is shown in Fig. 1, and 9×17 nodes are selected.

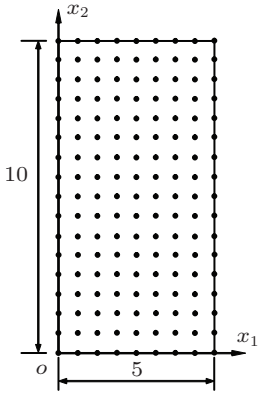


Fig. 1. Node distribution.

In order to show the validity of the proposed method, we compare the analytical solution and the numerical solution obtained by the ICVEFG method at $x_1 = 2.5$ and $x_2 = 5.5$ in Figs. 2 and 3, respectively. In addition, the comparison of $T_{,1}$ between the analytical and the numerical solutions at $x_1 = 2.5$ is shown in Fig. 4. It is evident that the present numerical results are in excellent agreement with the analytical solution.

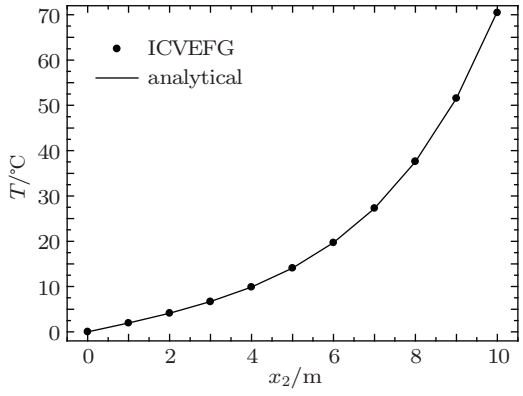


Fig. 2. Analytical and numerical solutions of temperature T at $x_1 = 2.5$.

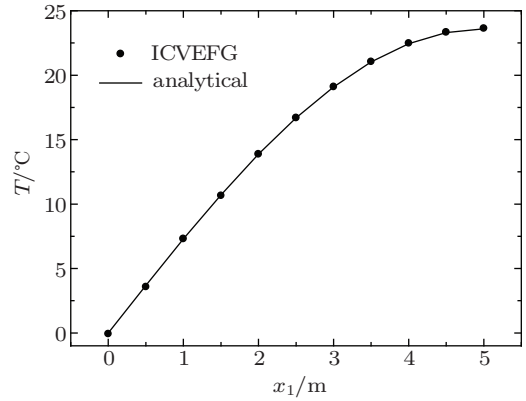


Fig. 3. Analytical and numerical solutions of temperature T at $x_2 = 5.5$.

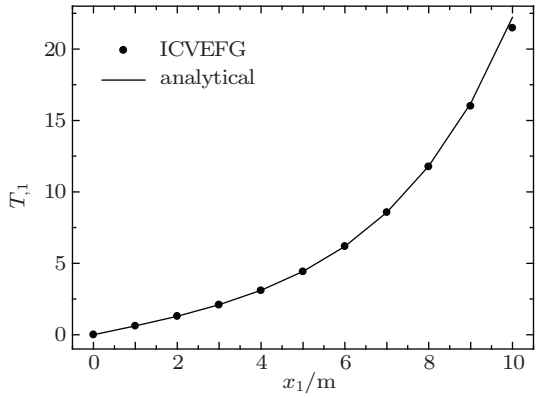


Fig. 4. Analytical and numerical solutions of $T_{,1}$ at $x_1 = 2.5$.

For the purpose of error estimation and convergence studies, we compare the relative errors with different node distributions at $x_1 = 2.5$ and $x_2 = 5.5$, which are shown in Figs. 5 and 6, respectively. The present results show that the more nodes there are, the greater precision we obtain.

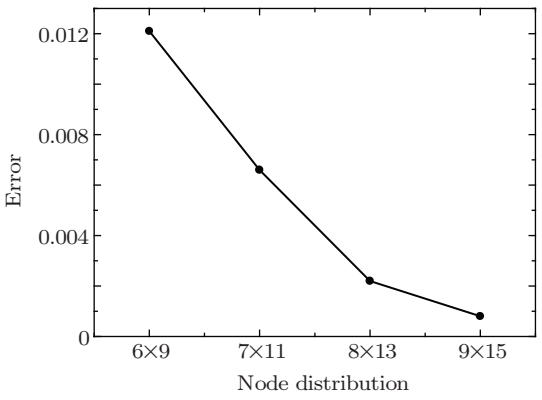


Fig. 5. Error of T at $x_1 = 2.5$.

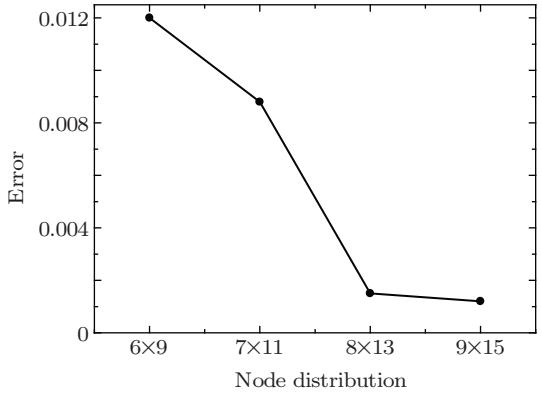


Fig. 6. Error of T at $x_2 = 5.5$.

4.2. Poisson's equation with Dirichlet condition on a rectangle

The second example is a 2D Poisson's equation with the Dirichlet boundary condition on a rectangle. The Poisson's equation is

$$\nabla^2 T(x_1, x_2) = -2(x_1 + x_2 - x_1^2 - x_2^2),$$
$$0 \le x_1 \le 1, \quad 0 \le x_2 \le 1. \tag{72}$$

The boundary condition is

$$T(x_1, x_2) = 0. \tag{73}$$

The analytical solution of this problem is

$$T(x_1, x_2) = (x_1 - x_1^2)(x_2 - x_2^2). \tag{74}$$

The node distribution on plate $\Omega = [0, 1] \times [0, 1]$ is shown in Fig. 7, and 9×15 nodes are selected.

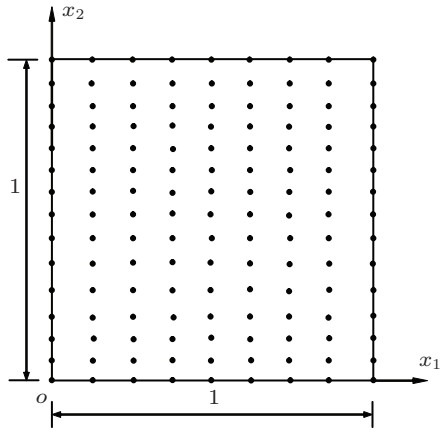


Fig. 7. Node distribution.

The comparison of the analytical solution and the numerical solutions obtained by the CVEFG method and the ICVEFG method at $x_1 = 0.6$ is presented in Fig. 8, the relative errors are 0.0025 and 0.0004, respectively. Moreover, the corresponding derivative $T_{,1}$ is shown in Fig. 9, and the relative errors are 0.0064 and 0.0024, respectively. In addition, the analytical solution and the numerical solutions obtained by the CVEFG method and the ICVEFG method at $x_2 = 0.65$ are presented in Fig. 10, the relative errors are 0.0114 and 0.0009 respectively. Moreover, the corresponding derivative $T_{,2}$ is shown in Fig. 11, and the relative errors are 0.0090 and 0.0008, respectively. It is evident that the present numerical solution agrees with the analytical solution well. And the ICVEFG method has greater accuracy than the CVEFG method.

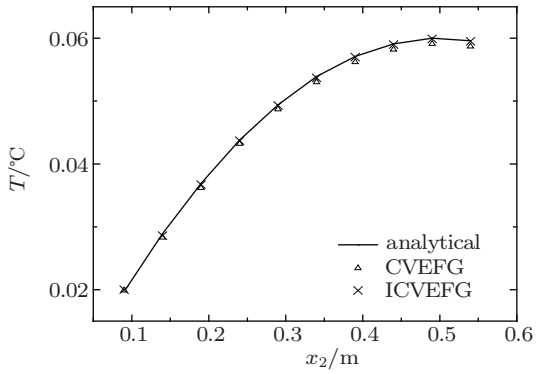


Fig. 8. Values of T at $x_1 = 0.6$.

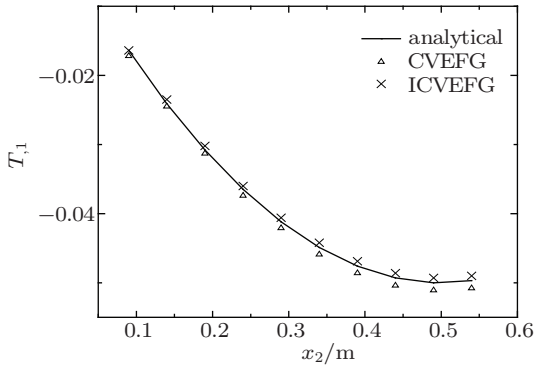


Fig. 9. Values of $T_{,1}$ at $x_1 = 0.6$.

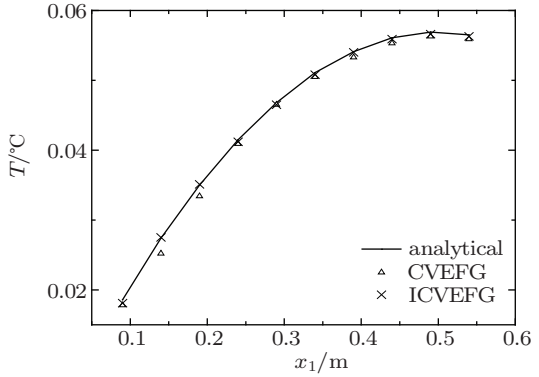


Fig. 10. Values of T at $x_2 = 0.65$.

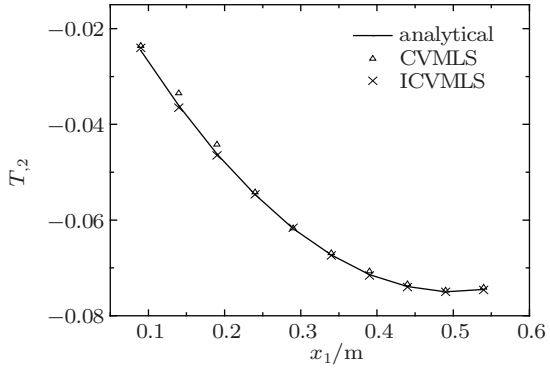


Fig. 11. Values of $T_{,2}$ at $x_2 = 0.65$.

For the purpose of convergence studies, we compare the relative errors with different node distributions at $x_2 = 0.6$ for the ICVEFG and the CVEFG methods, which are shown in Figs. 12 and 13. The present results show that the more nodes there are, the greater precision we obtain. With the same node distribution, the ICVEFG method has smaller relative errors than the CVEFG method. And to obtain the same precision, the ICVEFG method needs fewer nodes. The ICVEFG method's error with 5×8 node distribution is approaching that of the CVEFG method with 8×14 node distribution, as shown in Figs. 12 and 13. As a consequence, the results show that the ICVEFG method has a better convergence characteristic. The CPU time spent by the former is 7.9 s, while it is 8.8 s by the latter. We conclude that the ICVEFG method has not only the better convergence characteristic but also the greater computational precision and efficiency than the CVEFG method.

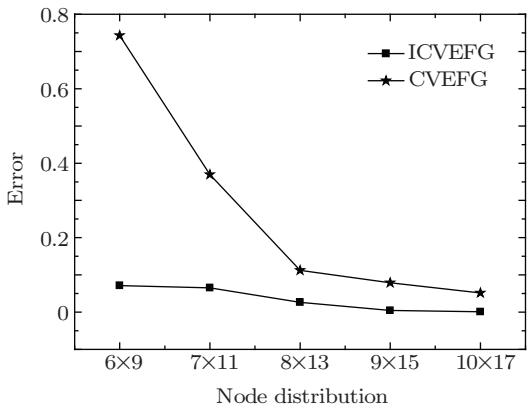


Fig. 12. Errors of T at $x_2 = 0.6$.

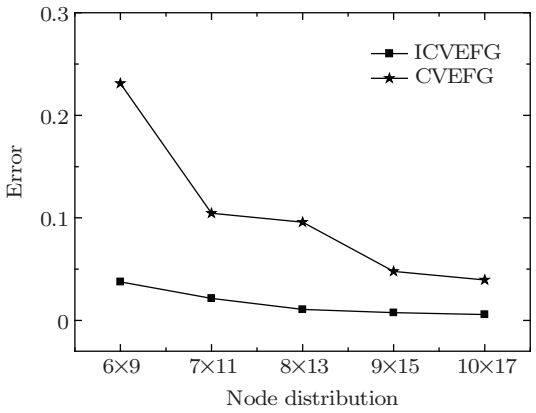


Fig. 13. Errors of $T_{,1}$ at $x_2 = 0.6$.

4.3. Poisson's equation with Dirichlet condition on a torus

The third example is a 2D Poisson's equation on a torus. The Poisson's equation is

$$\nabla^2 u = \frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2} = 4, \quad a < r < b, \quad 0 < \theta < 2\pi. \tag{75}$$

The boundary condition is

$$u(a, \theta) = 0, \quad u(b, \theta) = 0. \tag{76}$$

The analytical solution of this problem is

$$u(r, \theta) = (r^2 - a^2) + (b^2 - a^2)\frac{\log r - \log a}{\log a - \log b}. \tag{77}$$

Here, we suppose $a = 1$ and $b = 5$. By taking advantage of the symmetry of the model, only a quarter of the model needs to be considered in the analysis. Figure 14 shows the distribution of the nodes in the domain for this problem, and 11×17 nodes are selected.

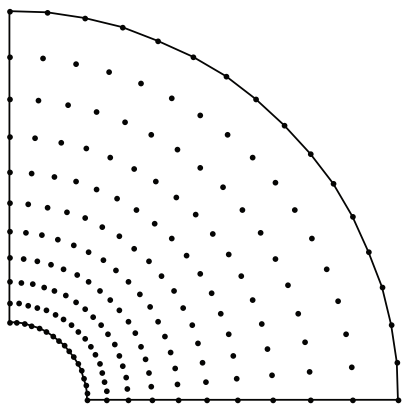


Fig. 14. Node distribution.

The comparison between the analytical solution and the numerical solutions obtained by the ICVEFG and the EFG methods along the r axis at $\theta = \frac{9}{32}\pi$ is presented in Fig. 15. It is evident that the numerical solutions both are in agreement with the analytical solution. However, the CPU time spent in the EFG method is 21.26 s, while it is 10.02 s in the ICVEFG method. We compare CPU time spent in the ICVEFG and the EFG methods at $\theta = 0$ under different node distributions, as shown in Fig. 16. It is shown that the ICVEFG method has greater computational efficiency than the EFG method.

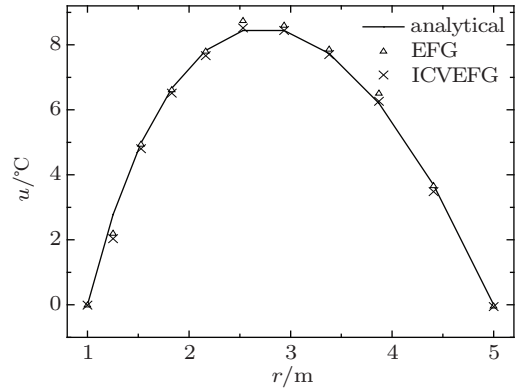


Fig. 15. Values of u at $\theta = \frac{9}{32}\pi$.

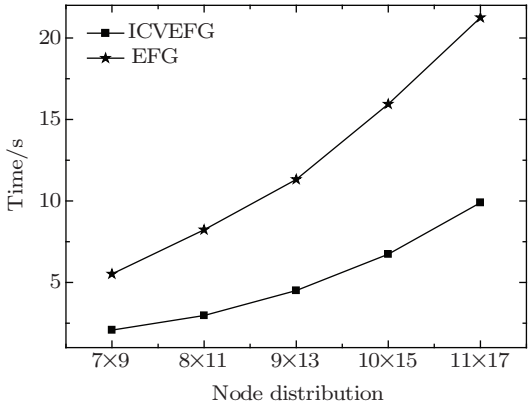


Fig. 16. The CPU time at $\theta = 0$.

Because of the fact that the analytical solution equals to zero at both endpoints, we remove the endpoints when calculating the relative error. The present results show that the relative error in the ICVEFG method is similar to that in the EFG method, and they are both small. Then the ICVEFG and the EFG methods both have good convergence characteristics. The results also show that under the same node distribution, the ICVEFG method spends less time; within the same CPU time, the ICVEFG method needs less nodes.

Figure 16 shows that the CPU time spent in ICVEFG method with 10×15 node distribution is almost equal to that in the EFG method with 7×9 node distribution. Therefore, the EFG and the ICVEFG methods can give similar results, but need different computer time. Compared with the EFG method, the ICVEFG method has a greater computational efficiency.

5. Conclusion

The ICVMLS approximation has all the advantages of the CVMLS approximation. Compared with the CVMLS approximations, the ICVMLS approximation has an explicit physical meaning and greater computational precision and efficiency.

Based on the EFG method and the ICVMLS approximation, the ICVEFG method is presented for two-dimensional potential problems. Three numerical examples are given to show the validity of the ICVEFG method. With the same node distribution, the numerical error in the ICVEFG method is much smaller than that in the CVEFG method, and the CPU time spent in the ICVEFG method is much less than that in the EFG method. Thus, the ICVEFG method has not only better convergence characteristics but also greater computational precision and efficiency.

References

[1] Belytschko T, Krongauz Y, Organ D, Fleming M and Krysl P 1996 *Comput. Methods Appl. Mech. Engrg.* **139** 3

[2] Qin Y X and Cheng Y M 2006 *Acta Phys. Sin.* **55** 3215 (in Chinese)

[3] Cheng R J and Cheng Y M 2011 *Acta Phys. Sin.* **60** 070206 (in Chinese)

[4] Li S C, Cheng Y M and Li S C 2006 *Acta Phys. Sin.* **55** 4760 (in Chinese)

[5] Cheng Y and Li J 2006 *Sci. Chin. Ser.* G: *Phys. Mech. Astron.* **49** 46

[6] Li D M, Peng M J and Cheng Y M 2011 *Sci. Sin. Phys. Mech. Astron.* **41** 1003 (in Chinese)

[7] Dai B D and Cheng Y M 2007 *Acta Phys. Sin.* **56** 597 (in Chinese)

[8] Cheng R J and Cheng Y M 2008 *Acta Phys. Sin.* **57** 6037 (in Chinese)

[9] Cheng R J and Cheng Y M 2011 *Chin. Phys.* B **20** 070206

[10] Chen L and Cheng Y M 2010 *Chin. Phys.* B **19** 090204

[11] Chen L and Cheng Y M 2008 *Acta Phys. Sin.* **57** 6047 (in Chinese)

[12] Chen L and Cheng Y M 2008 *Acta Phys. Sin.* **57** 1 (in Chinese)

[13] Cheng Y M and Li J H 2005 *Acta Phys. Sin.* **54** 4463 (in Chinese)

[14] Lancaster P and Salkauskas K 1981 *Math. Comput.* **37** 141

[15] Afshar M H, Naisipour M and Amani J 2011 *Finite Elem. Anal. Des.* **47** 1315

[16] Wang J F, Bai F N and Cheng Y M 2011 *Chin. Phys.* B **20** 030206

[17] Zhang Z, Zhao P and Liew K M 2009 *Eng. Anal. Bound. Elem.* **33** 547

[18] Li X L and Zhu J L 2011 *Adv. Eng. Softw.* **42** 927

[19] Ren H P, Cheng Y M and Zhang W 2009 *Chin. Phys.* B **18** 4065

[20] Cheng Y M, Peng M J and Li J H 2005 *Chin. J. Theor. Appl. Mech.* **37** 719 (in Chinese)

[21] Bai F N, Li D M, Wang J F and Cheng Y M 2012 *Chin. Phys.* B **21** 020204