Engineering with Computers (2019) 35:101–113
https://doi.org/10.1007/s00366-018-0586-6

ORIGINAL ARTICLE

Solving viscoelastic problems with cyclic symmetry via a temporally adaptive EFG-SB partitioning algorithm

Xiaofeng Guo[1,2] · **Haitian Yang**[2]

Received: 1 September 2017 / Accepted: 22 January 2018 / Published online: 9 February 2018

Abstract
Based upon a combination of a temporally-piecewise adaptive algorithm with an Element-Free Galerkin Scaled Boundary Method (EFG-SBM), a partitioning algorithm is presented for the two-dimensional viscoelastic analysis of cyclically symmetric structures. By expanding variables at a discretized time interval, the variations of variables can be described more precisely, and a space–time domain-coupled problem can be converted into a series of recurrent boundary value problems which are solved by an EFG-SBM-based partitioning algorithm via an adaptive computing process. Numerical examples are given to verify the proposed algorithm in terms of computing accuracy and efficiency.

Keywords Viscoelasticity · Cyclic symmetry · Element-free Galerkin-scaled boundary method · Temporally piecewise adaptive algorithm

1 1. Introduction

Efficient numerical algorithms are increasingly demanding to solve viscoelastic problems because analytical solutions are very limited, regarding the time-dependent constitutive relationship, complex geometry and boundary conditions [1–7].

The scaled boundary method (SBM) [8–11] is a spatial discretized algorithm that is semi-analytical, and takes advantages in dealing with problems of stress singularities and unbounded domains [12–20]. SBM has been employed in the viscoelastic analysis as a spacial solver at each of time intervals [5].

SBM is more computationally expensive than FEM mainly because an eigenvalue problem needs to be solved in generating system equations, thus as a spatial solver used for every recursive euqation, frequently, it is definitely more necessary and significant for SBM to reduce its computational cost.

In this paper, using an expansion technique in the time domain, an initial boundary value problem of viscoelasticity is decoupled into a series of recurrent boundary value problems which are solved by an EFG-SBM. If the structure of interest is cyclically symmetric [21–24], both the eigenvalue and system equations of EFG-SBM can be solved utilizing a partitioning algorithm, and the computational cost will be reduced by solving a series of independent subproblems instead of the whole EFG-SB equations at each of time intervals. On the other hand, at a discretized time interval, the variations of variables can be described more precisely, computing accuracy can be controlled via an adaptive process. Although there was a previous work concerned with exploitation of the cyclic symmetry in viscoelastic analysis by scaled boundary finite element method (SBFEM) [25], but it seems no report relevant to EFG-SBM.

This paper is organized as follows. Recursive viscoelastic governing equations and EFG-SBM equations are given in Sects. 2 and 3, respectively. Rotationally periodic symmetrical structures are briefly described in Sect. 4, and a piecewise partitioning algorithm is presented in the Sect. 5. Section 6 provides numerical verification via two examples and Sect. 7 summarizes conclusions.

✉ Haitian Yang
haitian@dlut.edu.cn

[1] School of Civil and safety Engineering, Dalian Jiaotong University, Dalian 116028, People's Republic of China

[2] Dept. of Engineering Mechanics, State Key Lab of Structural Analysis for Industrial Equipment, Dalian University of Technology, Dalian 116024, People's Republic of China

2 2. Recurrent governing equations

The governing equations of 2-D viscoelastic problems are described by [26].

Equilibrium equation

$$\mathrm{H}\sigma + b = 0 \tag{1}$$

Relationship of strain and displacement

$$\boldsymbol{\varepsilon} = \mathrm{H}^{\mathrm{T}}\mathbf{u} \tag{2}$$

where $\boldsymbol{\sigma}$ and $\boldsymbol{\varepsilon}$ denote the vectors of stress and strain, respectively, $\mathbf{b}$ is the vector of body force, $\mathbf{u}$ is the vector of displacement,

$$\mathbf{H} = \begin{bmatrix} \partial/\partial x & 0 & \partial/\partial y \\ 0 & \partial/\partial y & \partial/\partial x \end{bmatrix} \tag{3}$$

The viscoelastic constitutive relationships are specified by a three-parameter solid model (see Fig. 1) in a differential form [1, 27, 28].

$$\alpha_1 \mathbf{D}\boldsymbol{\varepsilon}(t) + p_1 \mathbf{D}\frac{\mathrm{d}\boldsymbol{\varepsilon}(t)}{\mathrm{d}t} = (\alpha_1 + \alpha_2)\boldsymbol{\sigma}(t) + p_1 \frac{\mathrm{d}\boldsymbol{\sigma}(t)}{\mathrm{d}t} \quad (t > 0) \tag{4}$$

$$\boldsymbol{\sigma}(t) = \mathbf{D}\boldsymbol{\varepsilon}(t) \quad (t > 0) \tag{5}$$

where

$$\mathbf{D} = \frac{E_2}{1-v^2}\begin{pmatrix} 1 & v & 0 \\ v & 1 & 0 \\ 0 & 0 & \frac{1-v}{2} \end{pmatrix} \quad \text{for the plane stress problem} \tag{6}$$

$$p_1 = \frac{\eta_1}{E_1 + E_2} \quad \alpha_1 = \frac{E_1}{E_1 + E_2} \quad \alpha_2 = \frac{E_2}{E_1 + E_2} \tag{7}$$

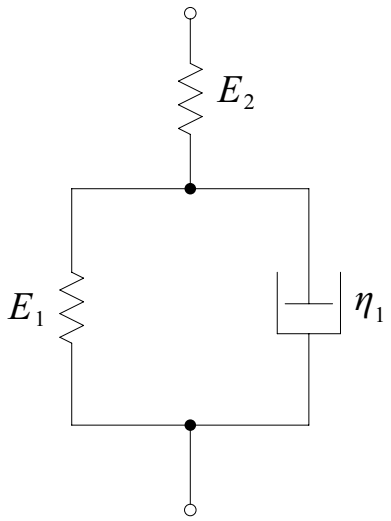

Fig. 1 Three-parameter solid

where v, E_1, E_2, and η_1 are constitutive parameters.

For the plane strain problems, E_1, E_2, η_1 and v need to be replaced by $\frac{E_1}{1-v^2}$, $\frac{E_2}{1-v^2}$, $\frac{\eta_1}{1-v^2}$, and $\frac{v}{1-v}$, respectively.

The boundary conditions are given by [26]

$$\mathbf{u} = \tilde{\mathbf{u}} \quad \text{on } \Gamma_{\mathrm{u}} \tag{8}$$

$$\mathbf{p} = \tilde{\mathbf{p}} \quad \text{on } \Gamma_{\sigma} \tag{9}$$

where $\mathbf{p}$ denotes the vector of traction, $\tilde{\mathbf{u}}$ and $\tilde{\mathbf{p}}$ are the prescribed functions on the boundary.

Divide time domain into a number of time intervals, the initial points and sizes of the time intervals are defined by $t_0, t_1, t_2, \ldots, t_k \ldots$ and $T_1, T_2, \ldots, T_k \ldots$, respectively. At the kth discretized time interval, to describe the variation of variables more precisely, all variables are expanded in terms of s_t

$$\boldsymbol{\sigma} = \sum_{m=0} \boldsymbol{\sigma}^{km} s_t^m \tag{10}$$

$$\boldsymbol{\varepsilon} = \sum_{m=0} \boldsymbol{\varepsilon}^{km} s_t^m \tag{11}$$

$$\mathbf{b} = \sum_{m=0} \mathbf{b}^{km} s_t^m \tag{12}$$

$$\mathbf{u} = \sum_{m=0} \mathbf{u}^{km} s_t^m \tag{13}$$

$$\tilde{\mathbf{u}} = \sum_{m=0} \tilde{\mathbf{u}}^{km} s_t^m \tag{14}$$

$$\mathbf{p} = \sum_{m=0} \mathbf{p}^{km} s_t^m \tag{15}$$

$$\tilde{\mathbf{p}} = \sum_{m=0} \tilde{\mathbf{p}}^{km} s_t^m \tag{16}$$

$$s_t = \frac{t - t_{k-1}}{T_k} \tag{17}$$

where t_{k-1} and T_k represent the initial point and size of the kth time interval, respectively, $\boldsymbol{\sigma}^{km}$ and $\boldsymbol{\varepsilon}^{km}$ represent the expanding coefficients of $\boldsymbol{\sigma}$ and $\boldsymbol{\varepsilon}$ at kth time interval, respectively, m stands for the power order of expansion, $\mathbf{b}^{km}$ denotes the expanding coefficient of $\mathbf{b}$, $\mathbf{u}^{km}$, $\mathbf{p}^{km}$, $\tilde{\mathbf{u}}^{km}$ and $\tilde{\mathbf{p}}^{km}$ are the expanding coefficients of $\mathbf{u}$, $\mathbf{p}$, $\tilde{\mathbf{u}}$ and $\tilde{\mathbf{p}}$, at kth time interval, respectively.

The conversion relationship between the differentiations with respect to t and s_t is

$$\frac{\mathrm{d}}{\mathrm{d}t} = \frac{1}{T_k}\frac{\mathrm{d}}{\mathrm{d}s_t} \tag{18}$$

Substituting Eqs. (10–16) into Eqs. (1, 2, 5, 8, 9) then yields

$$\mathrm{H}\boldsymbol{\sigma}^{km} + \mathbf{b}^{km} = 0 \tag{19}$$

$$\boldsymbol{\varepsilon}^{km} = \mathrm{H}^{\mathrm{T}}\mathbf{u}^{km} \tag{20}$$

$$\mathbf{u}^{km} = \tilde{\mathbf{u}}^{km} \quad \text{on } \Gamma_u \tag{21}$$

$$\mathbf{p}^{km} = \tilde{\mathbf{p}}^{km} \quad \text{on } \Gamma_\sigma \tag{22}$$

At the first time interval ($k=1$), when $m=0$, i.e. $t=0$,

$$\boldsymbol{\sigma}^{km} = \boldsymbol{\sigma}^{10} = \mathbf{D}\boldsymbol{\varepsilon}^{km} = \mathbf{D}\boldsymbol{\varepsilon}^{10} \tag{23}$$

$\boldsymbol{\sigma}^{10}$, $\boldsymbol{\varepsilon}^{10}$, and $\mathbf{u}^{10}$ can be obtained by solving Eqs. (19–23).

At other time intervals ($k=2, 3,\ldots$)

$$\boldsymbol{\varepsilon}^{k0} = \sum_{m=0} \boldsymbol{\varepsilon}^{(k-1)m} \tag{24}$$

$$\boldsymbol{\sigma}^{k0} = \sum_{m=0} \boldsymbol{\sigma}^{(k-1)m} \tag{25}$$

$$\mathbf{u}^{k0} = \sum_{m=0} \mathbf{u}^{(k-1)m} \tag{26}$$

For $m \neq 0$, substituting Eqs. (10–11) in Eq. (4) then yields

$$\boldsymbol{\sigma}^{km} = \mathbf{D}\boldsymbol{\varepsilon}^{km} + \mathbf{C}^{km} \quad k = 1, 2, 3, \ldots \quad m = 1, 2, \ldots \tag{27}$$

$$\mathbf{C}^{km} = \frac{T_k E_1}{m\eta_1}\mathbf{D}\boldsymbol{\varepsilon}^{k(m-1)} - \frac{T_k}{mp_1}\boldsymbol{\sigma}^{k(m-1)} \tag{28}$$

Furthermore,

$$\boldsymbol{\sigma}^{km} = \mathbf{D}\boldsymbol{\varepsilon}^{km} + \beta_m\left(\alpha_2\mathbf{D}\sum_{i=1}^{m-1}\gamma_i\boldsymbol{\varepsilon}^{ki} - \alpha_1\mathbf{D}\boldsymbol{\varepsilon}^{k0} + \boldsymbol{\sigma}^{k0}\right) \tag{29}$$

where

$$\beta_m = \frac{1}{m!}\left(\frac{-T_k}{p_1}\right)^{(m)} \tag{30}$$

$$\gamma_i = i!\left(\frac{-p_1}{T_k}\right)^{i} \tag{31}$$

3 Recursive equations of EFG-SBM

The scaled boundary method introduces such a ξ–s coordinate system by scaling a defining curve relative to a scaling center (x_0, y_0) selected within the domain, as shown in Fig. 2. The coordinate ξ runs from the scaling center towards the curve, and has values of zero at the scaling center and unity at the curve. The coordinate s specifies a distance around the boundary from an origin on the boundary [12, 13].

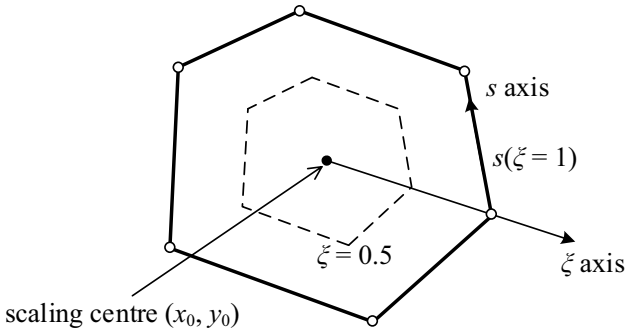

Fig. 2 Definition of the scaled coordinate system

The scaled boundary and Cartesian coordinate systems are related by the scaling equations [12]

$$x = x_0 + \xi x_s(s) \tag{32a}$$

$$y = y_0 + \xi y_s(s) \tag{32b}$$

For Eqs. (19–22)

$$\mathbf{u}^{km}(\xi, s) = \mathbf{N}(s)\mathbf{u}_h^{km}(\xi) \tag{33}$$

where $\mathbf{u}^{km}(\xi, s)$ represents the expanding coefficient of $\mathbf{u} = \sum_{m=0}\mathbf{u}^{km}s_t^m$, $\mathbf{N}(s)$ is a matrix of shape function and $\mathbf{u}_h^{km}(\xi)$ is a set of n functions analytical in ξ.

The expanding coefficient of the strain $\boldsymbol{\varepsilon} = \sum_{m=0}\boldsymbol{\varepsilon}^{km}s_t^m$ in the scaled coordinate system is described by [12, 13]

$$\boldsymbol{\varepsilon}^{km}(\xi, s) = \mathbf{B}_1(s)\mathbf{u}^{km}(\xi)_{,\xi} + \frac{1}{\xi}\mathbf{B}_2(s)\mathbf{u}^{km}(\xi) \quad k = 1, 2, \ldots \quad m = 0, 1, 2, \ldots \tag{34}$$

$$\mathbf{B}_1(s) = \mathbf{b}_1(s)\mathbf{N}(s) = \frac{1}{|\mathbf{J}|}\begin{bmatrix} \frac{\partial y_s(s)}{\partial s} & 0 & -\frac{\partial x_s(s)}{\partial s} \\ 0 & -\frac{\partial x_s(s)}{\partial s} & \frac{\partial y_s(s)}{\partial s} \end{bmatrix}^{\mathrm{T}} \cdot \mathbf{N}(s) \tag{35}$$

$$\mathbf{B}_2(s) = \mathbf{b}_2(s)\mathbf{N}(s)_{,s} = \frac{1}{|\mathbf{J}|}\begin{bmatrix} -y_s(s) & 0 & x_s(s) \\ 0 & x_s(s) & -y_s(s) \end{bmatrix}^{\mathrm{T}} \cdot \mathbf{N}(s)_{,s} \tag{36}$$

$$|\mathbf{J}| = x_s(s)y_s(s)_{,s} - y_s(s)x_s(s)_{,s} \tag{37}$$

where $\mathbf{N}(s)_{,s}$ refers to the derivative of $\mathbf{N}(s)$ with respect to s.

Utilizing the virtual displacement principle then gives (in the absence of body force) [12, 13]

$$\int_V \delta\boldsymbol{\varepsilon}(\xi, s)^{\mathrm{T}}\boldsymbol{\sigma}^{km}(\xi, s)\,\mathrm{d}V - \int_S \delta\mathbf{u}(s)^{\mathrm{T}}\mathbf{p}^{km}(s)\,\mathrm{d}s = 0 \tag{38}$$

$m = 0, 1, 2, 3, \ldots$ $k = 1, 2, 3, \ldots$ where

$$\delta\mathbf{u}(\xi, s) = \mathbf{N}(s)\delta\mathbf{u}(\xi) \tag{39}$$

denotes the vector of virtual displacement, and

$$\delta\boldsymbol{\varepsilon}(\xi, s) = \mathbf{B}_1(s)\delta\mathbf{u}(\xi)_{,\xi} + \frac{1}{\xi}\mathbf{B}_2(s)\delta\mathbf{u}(\xi) \tag{40}$$

The second term in Eq. (38) becomes

$$\int_S \delta\mathbf{u}(s)^{\mathrm{T}}\mathbf{p}^{km}(s)\,\mathrm{d}\,s = \delta\mathbf{u}^{\mathrm{T}}\int_S \mathbf{N}(s)^{\mathrm{T}}\,\mathbf{p}^{km}(s)\mathrm{d}s = \delta\mathbf{u}^{\mathrm{T}}\mathbf{P}^{km} \tag{41}$$

$k = 1, 2, \ldots \quad m = 0, 1, 2, \ldots$

Substituting Eqs. (23) and (29) into Eq. (38), respectively, then gives

$$\int_V \delta\boldsymbol{\varepsilon}(\xi, s)^{\mathrm{T}}\mathbf{D}\boldsymbol{\varepsilon}^{k0}(\xi, s)\,\mathrm{d}\,V - \int_S \delta\mathbf{u}(s)^{\mathrm{T}}\mathbf{p}^{k0}(s)\,\mathrm{d}\,s = 0 \tag{42}$$

$t = 0$, i.e. $k = 1$ and $m = 0$

$$\int_V \delta\boldsymbol{\varepsilon}(\xi, s)^{\mathrm{T}}\mathbf{D}\boldsymbol{\varepsilon}^{km}\,\mathrm{d}\,V + \beta_m\alpha_2\sum_{i=1}^{m-1}\gamma_i\int_V \delta\boldsymbol{\varepsilon}(\xi, s)^{\mathrm{T}}\mathbf{D}\boldsymbol{\varepsilon}^{ki}\,\mathrm{d}\,V$$
$$-\beta_m\alpha_1\int_V \delta\boldsymbol{\varepsilon}(\xi, s)^{\mathrm{T}}\mathbf{D}\boldsymbol{\varepsilon}^{k0}\,\mathrm{d}\,V + \beta_m\int_V \delta\boldsymbol{\varepsilon}(\xi, s)^{\mathrm{T}}\boldsymbol{\sigma}^{k0}\,\mathrm{d}\,V - \int_S \delta\mathbf{u}(s)^{\mathrm{T}}\mathbf{p}^{km}(s)\,\mathrm{d}\,s = 0 \quad \text{at } t > 0 \tag{43}$$

Using Eqs. (34), (39–41) and Green's Theorem then gives [5, 12]

$$\delta\mathbf{u}^{\mathrm{T}}\left\{\mathbf{E}_0\mathbf{u}_{h,\xi}^{k0} + \mathbf{E}_1^{\mathrm{T}}\mathbf{u}_h^{k0}\right\} - \delta\mathbf{u}^{\mathrm{T}}\mathbf{P}^{k0}$$
$$-\int_0^1 \delta\mathbf{u}(\xi)^{\mathrm{T}}\left\{\mathbf{E}_0\xi\mathbf{u}_h^{k0}(\xi)_{,\xi\xi} + \left[\mathbf{E}_0 + \mathbf{E}_1^{\mathrm{T}} - \mathbf{E}_1\right]\mathbf{u}_h^{k0}(\xi)_{,\xi} - \mathbf{E}_2\frac{1}{\xi}\mathbf{u}_h^{k0}(\xi)\right\}\mathrm{d}\,\xi = 0 \quad \text{at } t > 0 \tag{44}$$

$$\delta\mathbf{u}^{\mathrm{T}}\left\{\mathbf{E}_0\mathbf{u}_{h,\xi}^{km} + \mathbf{E}_1^{\mathrm{T}}\mathbf{u}_h^{km}\right\}$$
$$+\delta\mathbf{u}^{\mathrm{T}}\left\{\beta_m\alpha_2\sum_{i=1}^{m-1}\gamma_i\left\{\mathbf{E}_0\mathbf{u}_{h,\xi}^{ki} + \mathbf{E}_1^{\mathrm{T}}\mathbf{u}_h^{ki}\right\} - \beta_m\alpha_1\left\{\mathbf{E}_0\mathbf{u}_{h,\xi}^{k0} + \mathbf{E}_1^{\mathrm{T}}\mathbf{u}_h^{k0}\right\} + \beta_m\mathbf{P}^{k0} - \mathbf{P}^{km}\right\}$$
$$-\beta_m\alpha_2\sum_{i=1}^{m-1}\gamma_i\int_0^1 \delta\mathbf{u}(\xi)^{\mathrm{T}}\left\{\mathbf{E}_0\xi\mathbf{u}_h^{ki}(\xi)_{,\xi\xi} + [\mathbf{E}_0 + \mathbf{E}_1^{\mathrm{T}} - \mathbf{E}_1]\mathbf{u}_h^{ki}(\xi)_{,\xi} - \mathbf{E}_2\frac{1}{\xi}\mathbf{u}_h^{ki}(\xi)\right\}\mathrm{d}\,\xi \quad \text{at } t > 0 \tag{45}$$
$$+\beta_m\alpha_1\int_0^1 \delta\mathbf{u}(\xi)^{\mathrm{T}}\left\{\mathbf{E}_0\xi\mathbf{u}_h^{k0}(\xi)_{,\xi\xi} + [\mathbf{E}_0 + \mathbf{E}_1^{\mathrm{T}} - \mathbf{E}_1]\mathbf{u}_h^{k0}(\xi)_{,\xi} - \mathbf{E}_2\frac{1}{\xi}\mathbf{u}_h^{k0}(\xi)\right\}\mathrm{d}\,\xi$$
$$-\int_0^1 \delta\mathbf{u}(\xi)^{\mathrm{T}}\left\{\mathbf{E}_0\xi\mathbf{u}_h^{km}(\xi)_{,\xi\xi} + [\mathbf{E}_0 + \mathbf{E}_1^{\mathrm{T}} - \mathbf{E}_1]\mathbf{u}_h^{km}(\xi)_{,\xi} - \mathbf{E}_2\frac{1}{\xi}\mathbf{u}_h^{km}(\xi)\right\}\mathrm{d}\,\xi = 0$$

where

$$\mathbf{E}_0 = \int_S \mathbf{B}_1(s)^{\mathrm{T}}\mathbf{D}\mathbf{B}_1(s)|\mathbf{J}(s)|\,\mathrm{d}\,s \tag{46a}$$

$$\mathbf{E}_1 = \int_S \mathbf{B}_2(s)^{\mathrm{T}}\mathbf{D}\mathbf{B}_1(s)|\mathbf{J}(s)|\,\mathrm{d}\,s \tag{46b}$$

$$\mathbf{E}_2 = \int_S \mathbf{B}_2(s)^{\mathrm{T}}\mathbf{D}\mathbf{B}_2(s)|\mathbf{J}(s)|\,\mathrm{d}\,s \tag{46c}$$

In Eqs. (44) and (45) and the following section, $\mathbf{u}_{h,\xi}^{km}$ and $\mathbf{u}_h^{km}$ are used to represent $\mathbf{u}_{h,\xi}^{km}(\xi = 1)$ and $\mathbf{u}_h^{km}(\xi = 1)$.

In Eq. (44),

$$\mathbf{P}^{k0} = \int_S \mathbf{N}(s)^{\mathrm{T}}\,\mathbf{p}^{k0}(s)\mathrm{d}s \quad k = 1 \tag{47}$$

In Eq. (45),

$$\mathbf{P}^{k0} = \int_S \mathbf{N}(s)^{\mathrm{T}}\, \mathbf{p}^{k0}(s)\mathrm{d}s = \sum_{m=0} \mathbf{P}^{(k-1)m} \quad k = 2, 3, \ldots \tag{48}$$

Furthermore [5],

$$P^{*km} = E_0 u^{km}_{h,\xi} + E_1^T u^{km}_h \quad k = 1, 2, \ldots \quad m = 0, 1, 2, \ldots \tag{49}$$

$$\mathbf{E}_0 \xi^2 \mathbf{u}^{km}_h(\xi)_{,\xi\xi} + [\mathbf{E}_0 + \mathbf{E}_1^{\mathrm{T}} - \mathbf{E}_1]\xi \mathbf{u}^{km}_h(\xi)_{,\xi} - \mathbf{E}_2 \mathbf{u}^{km}_h(\xi) = 0 \tag{50}$$

where

$$\mathbf{P}^{*km} = \mathbf{P}^{k0} \quad \text{at } t = 0 \tag{51}$$

$$\mathbf{P}^{*km} = \mathbf{P}^{km} - \beta_m \alpha_2 \sum_{i=1}^{m-1} \gamma_i \mathbf{P}^{*ki} + \beta_m \alpha_1 \mathbf{P}^{*k0} - \beta_m \mathbf{P}^{k0} \tag{52}$$

$$att > 0 \text{ and } m = 1, 2, \ldots$$

$$\mathbf{P}^{*k0} = \sum_{i=0} \mathbf{P}^{*(k-1)i} \quad att > 0 \text{i.e.} k = 2, 3, \ldots \tag{53}$$

$\mathbf{u}^{km}(\xi)$ is approximated by

$$\mathbf{u}^{km}(\xi) = c_1^{km} \xi^{-\lambda_1} \boldsymbol{\phi}_1 + c_2^{km} \xi^{-\lambda_2} \boldsymbol{\phi}_2 + \ldots = \sum_{i=1}^{n} c_i^{km} \xi^{-\lambda_i} \boldsymbol{\phi}_i \tag{54}$$

$$k = 1, 2, \ldots \quad m = 0, 1, 2, \ldots$$

When $\xi = 1$

$$\mathbf{u}^{km}_h = c_1^{km} \boldsymbol{\phi}_1 + c_2^{km} \boldsymbol{\phi}_2 + \ldots = \sum_{i=1}^{n} c_i^{km} \boldsymbol{\phi}_i = \boldsymbol{\Phi} \mathbf{c}^{km} \tag{55}$$

where

$$\boldsymbol{\Phi} = \left[\boldsymbol{\phi}_1 \ \boldsymbol{\phi}_2 \ \cdots \ \boldsymbol{\phi}_n \right] \tag{56}$$

$$\mathbf{c}^{km} = \left[c_1^{km}, c_2^{km}, \ldots, c_n^{km} \right]^{\mathrm{T}} \tag{57}$$

Therefore,

$$\left[\lambda_i^{\,2} \mathbf{E}_0 - \lambda_i \left[\mathbf{E}_1^{\mathrm{T}} - \mathbf{E}_1 \right] - \mathbf{E}_2 \right] \boldsymbol{\phi}_i = 0 \tag{58}$$

$$\mathbf{P}^{*km} = \mathbf{Q} \mathbf{c}^{km} \tag{59}$$

where

$$\mathbf{Q} = \left[\mathbf{q}_1 \ \mathbf{q}_2 \ \cdots \ \mathbf{q}_n \right] \tag{60}$$

$$\mathbf{q}_i = \left[\mathbf{E}_1^{\mathrm{T}} - \lambda_i \mathbf{E}_0 \right] \boldsymbol{\phi}_i \tag{61}$$

Equations (58) and (61) can be further expressed by [5, 12, 13]

$$\begin{bmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{bmatrix} \begin{Bmatrix} \boldsymbol{\phi} \\ \mathbf{q} \end{Bmatrix} = \lambda \begin{Bmatrix} \boldsymbol{\phi} \\ \mathbf{q} \end{Bmatrix} \tag{62}$$

where

$$\mathbf{Z}_{11} = \mathbf{E}_{\mathbf{0}}^{-1} \mathbf{E}_1^{\mathrm{T}} \tag{63a}$$

$$\mathbf{Z}_{12} = -\mathbf{E}_{\mathbf{0}}^{-1} \tag{63b}$$

$$\mathbf{Z}_{21} = \mathbf{E}_1 \mathbf{E}_0^{-1} \mathbf{E}_1^{\mathrm{T}} - \mathbf{E}_2 \tag{63c}$$

$$\mathbf{Z}_{22} = -\mathbf{E}_1 \mathbf{E}_0^{-1} \tag{63d}$$

Left multiplying both sides of Eq. (55) with $\boldsymbol{\Phi}^{-1}$ then yields

$$\mathbf{c}^{km} = \boldsymbol{\Phi}^{-1} \mathbf{u}^{km}_h \tag{64}$$

Thus,

$$Q\mathbf{c}^{km} = Q\boldsymbol{\Phi}^{-1} \mathbf{u}^{km}_h = \mathbf{K}\mathbf{u}^{km}_h = \mathbf{P}^{*km} \quad k = 1, 2, \ldots \quad m = 0, 1, 2, \ldots \tag{65}$$

The boundary is divided into several segments at each of which $N(s)$ is defined by EFGM [5, 13, 29, 30].

$$\left[N_1(s), \ \ldots \ , N_{n_l}(s) \right] = \boldsymbol{\varphi}(s)^{\mathrm{T}} \boldsymbol{\Theta}^{-1} \tag{66}$$

where

$$\boldsymbol{\varphi}(s)^{\mathrm{T}} = [\varphi_1(s) \ \ \varphi_2(s) \ \ \ldots \ \varphi_{n_l}(s)] \tag{67}$$

$$\boldsymbol{\Theta} = \begin{bmatrix} \varphi_1(s_1) & \ldots & \varphi_{n_l}(s_1) \\ \ldots & \ldots & \ldots \\ \varphi_1(s_{n_l}) & \ldots & \varphi_{n_l}(s_{n_l}) \end{bmatrix} \tag{68}$$

$$\varphi_I(s) = \mathbf{p}(s)^{\mathrm{T}} \mathbf{A}(s)^{-1} \mathbf{H}_I(s) \tag{69}$$

$$\mathbf{A}(s) = \sum_{I=1}^{n_x} w_I(s) \mathbf{p}(s_I) \mathbf{p}(s_I)^{\mathrm{T}} \tag{70}$$

$$\mathbf{H}_I(s) = w_I(s) \mathbf{p}(s_I) \tag{71}$$

$$\mathbf{p}(s)^{\mathrm{T}} = [1, s] \tag{72}$$

n_L is the total number of nodes at the lth segment. n_x is the total number of nodes in the domain of definition of point s at the segment.

A spline weight function [5, 13] is employed in the form of

$$w_I(s)=\begin{cases}1-6\left(\dfrac{|s-s_I|}{r_I}\right)^2+8\left(\dfrac{|s-s_I|}{r_I}\right)^3-3\left(\dfrac{|s-s_I|}{r_I}\right)^4 & 0<|s-s_I|<r_I\\ 0 & |s-s_I|\geq r_I\end{cases} \tag{73}$$

where s_I is s coordinate of Ith node, and r_I is the size of support domain.

Therefore,

$$\left[N_1(s)_{,s},\ \cdots,N_{n_l}(s)_{,s}\right]=\boldsymbol{\varphi}(s)_{,s}^{\mathrm{T}}\boldsymbol{\Theta}^{-1} \tag{74}$$

where

$$\boldsymbol{\varphi}(s)_{,s}^{\mathrm{T}}=\mathbf{p}(s)_{,s}^{\mathrm{T}}\mathbf{A}(s)^{-1}\mathbf{H}(s)+\mathbf{p}(s)^{\mathrm{T}}\left[\mathbf{A}(s)_{,s}^{-1}\mathbf{H}(s)+\mathbf{A}(s)^{-1}\mathbf{H}(s)_{,s}\right] \tag{75}$$

$$\mathbf{A}(s)_{,s}^{-1}=-\mathbf{A}(s)^{-1}\mathbf{A}(s)_{,s}\mathbf{A}(s)^{-1} \tag{76}$$

In the recursive solution of Eq. (65), a self-adaptive computation is carried out at each of the time intervals with a convergence criterion

$$\mathrm{Abs}\left(\left(u_{hj}^{kR}s_t^R\Big/\sum_{m=0}^{R-1}u_{hj}^{km}s_t^m\right)_{s_i=1}\right)\leq\beta \tag{77}$$

where β is an error bound, u_{hj}^{km} denotes the jth component of $\mathbf{u}_h^{km}$ ($m=1, 2,\ldots, R$). Every $\mathbf{u}_h^{kR}$($R=1, 2,\ldots$) is required to be checked with the above criterion, if the criterion is satisfied consecutively three times, computing will stop at current time interval, and step into the next one. If the criterion is not met, the next order $(R+1)$ computation will continue untill Eq. (77) meets.

In the computation, *mm* and *mim*, the upper and lower bound of R, will be prescribed in advance. If condition (77) is not satisfied when $R=mm$, a size decrement of time step is necessary to restart the recursive procedure at the current time interval; if condition (77) is satisfied when $R<mim$, a size increment of the time step can be considered at the next time interval.

In this paper, $mm=20$, $mim=5$, size increment and decrement are both half size of current time step.

4 Rotationally periodic symmetry

A structure or a computational region Ω is said to possess rotationally periodic symmetry of order N when its geometry and physical properties are invariant under the following N symmetry transformations [23, 24]

$$\Psi_i=(i-1)\theta\quad i=1,2,\ldots,N \tag{78}$$

where Ψ_i represents a rotation of Ω about its axis of rotation, $\theta=2\pi/N$, and N is defined as the order of symmetry. In Fig. 3, $N=6$.

Designate the boundary of Ω as Γ that is naturally divided into N identical partsΓ_i ($i=1,\ \ldots\ ,N$). Γ_i is arranged in an anti-clock wise sequence, i.e.

$$\Gamma=\Gamma_1\cup\Gamma_2\cdots\cup\Gamma_N,\quad \Gamma_i=\Psi_i:\Gamma_1 \tag{79}$$

Equation (79) means that Γ_i can be obtained from Γ_1, which is called 'basic region' and can be arbitrarily chosen from those identical parts. For any node or integration point in the basic region, there are certainly other $N-1$ different nodes or integration points, which are located symmetrically on the other $N-1$ symmetry regions. All these N nodes constitute a set of symmetric nodes, which is called symmetric node orbit, and is designated as O_A

$$O_A=\{A_1,A_2,...,A_N\} \tag{80}$$

In Fig. 3, it is readily seen that the six interface nodes B_i ($i=1,\ \ldots\ ,N$) constitute orbit O_B. Only those nodes that are located on the internal part and the 'right' interface of Γ_1 are regarded as belonging to the basic region. Only B_1 is regarded

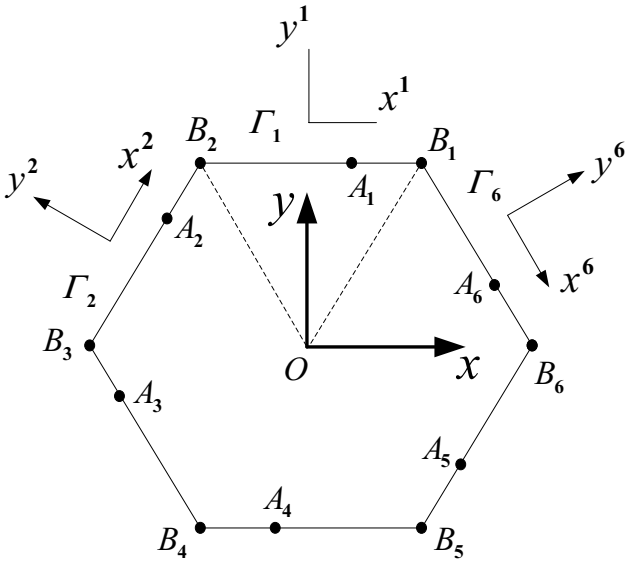

Fig. 3 A rotationally periodic plane plate with $N=6$. The symmetric node orbits O_A, O_B and their corresponding symmetric coordinate system for reference

as belonging to Γ_1 and B_2 will then be regarded as belonging to Γ_2.

5 An EFG-SBM-based partitioning algorithm

Consider a cyclically symmetric structure with order N and divide Γ into$N$ cyclically symmetric segments. There are M nodes at each segment.

$\mathbf{u}_h^{km}$ in the x–y coordinate system is described by

$$\left[\mathbf{u}_h^{km}\right]^{\mathrm{T}} = \left[\left[\mathbf{u}_h^{km}\right]^{1\mathrm{T}}, \left[\mathbf{u}_h^{km}\right]^{2\mathrm{T}}, ..., \left[\mathbf{u}_h^{km}\right]^{N\mathrm{T}}\right] \quad k = 1, 2, \ldots \quad m = 0, 1, 2, \ldots \tag{81}$$

where sub-vector $\left[\mathbf{u}_h^{km}\right]^i$ belongs to the ith symmetry region.

In such an ordering way, the coefficient matrices $\mathbf{Z}_{11}$, $\mathbf{Z}_{12}$, $\mathbf{Z}_{21}$ and $\mathbf{Z}_{22}$ in Eq. (62) can be written as

$$Z_{rs} = \begin{bmatrix} Z_{rs}^{11} & Z_{rs}^{12} & \cdots & Z_{rs}^{1N} \\ Z_{rs}^{21} & Z_{rs}^{22} & \cdots & Z_{rs}^{2N} \\ \vdots & \vdots & \ddots & \vdots \\ Z_{rs}^{N1} & Z_{rs}^{N2} & \cdots & Z_{rs}^{NN} \end{bmatrix}_{N*2M\times N*2M} \quad r,s = 1, 2 \tag{82}$$

In a symmetry-adapted reference coordinate system, $\mathbf{u}_h^{km}$ and $\mathbf{P}^{*km}$ are replaced by

$$\mathbf{u}_h^{km} = \mathbf{T}^M \bar{\mathbf{u}}_h^{km} \tag{83a}$$

$$\mathbf{P}^{*km} = \mathbf{T}^M \bar{\mathbf{P}}^{*km} \tag{83b}$$

where

$$\mathbf{T}^M = \begin{bmatrix} \bar{\mathbf{T}}_1^M & & & \mathbf{0} \\ & \bar{\mathbf{T}}_2^M & & \\ & & \cdots & \\ \mathbf{0} & & & \bar{\mathbf{T}}_N^M \end{bmatrix}_{N*2M\times N*2M} \quad \bar{\mathbf{T}}_i^M = \begin{bmatrix} \mathbf{T}_i & & & \mathbf{0} \\ & \mathbf{T}_i & & \\ & & \cdots & \\ \mathbf{0} & & & \mathbf{T}_i \end{bmatrix}_{2M\times 2M} \tag{84}$$

$$\mathbf{T}_i = \begin{bmatrix} \cos(i-1)\theta & -\sin(i-1)\theta \\ \sin(i-1)\theta & \cos(i-1)\theta \end{bmatrix}, \quad \theta = \frac{2\pi}{N} \tag{85}$$

where $\bar{\mathbf{u}}_h^{km}$ and $\bar{\mathbf{P}}^{*km}$ are defined via Eq. (80).

Substituting Eqs. (83a, 83b) into Eq. (65) and left-multiplying both sides of Eq. (65) with $\mathbf{T}^{M\mathrm{T}}$ then yield

$$\bar{\mathbf{K}}\bar{\mathbf{u}}_h^{km} = \bar{\mathbf{P}}^{*km} \tag{86}$$

where

$$\bar{\mathbf{K}} = \mathbf{T}^{MT}\mathbf{K}\mathbf{T}^M = \mathbf{T}^{MT}\mathbf{Q}\boldsymbol{\Phi}^{-1}\mathbf{T}^M = \bar{\mathbf{Q}}\bar{\boldsymbol{\Phi}}^{-1} \tag{87}$$

$$\bar{\boldsymbol{\Phi}} = \mathbf{T}^{MT}\boldsymbol{\Phi} \tag{88a}$$

$$\bar{Q} = \mathbf{T}^{MT}Q \tag{88b}$$

$$\boldsymbol{\phi} = \mathbf{T}^M\bar{\boldsymbol{\phi}} \tag{89a}$$

$$\mathbf{q} = \mathbf{T}^M\bar{\mathbf{q}} \tag{89b}$$

Substituting Eqs. (89a, b) into Eq. (62) and left-multiplying both sides of Eq. (62) with $\mathbf{T}^{M\mathrm{T}}$ then yield

$$\begin{bmatrix} T^{MT} & 0 \\ 0 & T^{MT} \end{bmatrix}\begin{bmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{bmatrix}\begin{bmatrix} T^M & 0 \\ 0 & T^M \end{bmatrix}\begin{Bmatrix} \bar{\phi} \\ \bar{q} \end{Bmatrix} = \lambda\begin{Bmatrix} \bar{\phi} \\ \bar{q} \end{Bmatrix} \tag{90}$$

i.e.

$$\begin{bmatrix} \bar{\mathbf{Z}}_{11} & \bar{\mathbf{Z}}_{12} \\ \bar{\mathbf{Z}}_{21} & \bar{\mathbf{Z}}_{22} \end{bmatrix}\begin{Bmatrix} \bar{\phi} \\ \bar{\mathbf{q}} \end{Bmatrix} = \lambda\begin{Bmatrix} \bar{\phi} \\ \bar{\mathbf{q}} \end{Bmatrix} \tag{91}$$

where

$$\bar{\mathbf{Z}}_{rs} = \mathbf{T}^{MT}\mathbf{Z}_{rs}\mathbf{T}^M \quad r,s = 1, 2 \tag{92}$$

It can be proved that $\bar{\mathbf{Z}}_{11}$, $\bar{\mathbf{Z}}_{12}$, $\bar{\mathbf{Z}}_{21}$ and $\bar{\mathbf{Z}}_{22}$ are block-circulant [23], i.e.

$$\bar{\mathbf{Z}}_{rs} = \begin{bmatrix} \bar{\mathbf{Z}}_{rs}^1 & \bar{\mathbf{Z}}_{rs}^2 & \cdots & \bar{\mathbf{Z}}_{rs}^N \\ \bar{\mathbf{Z}}_{rs}^N & \bar{\mathbf{Z}}_{rs}^1 & & \vdots \\ \vdots & & \ddots & \bar{\mathbf{Z}}_{rs}^2 \\ \bar{\mathbf{Z}}_{rs}^2 & \cdots & \bar{\mathbf{Z}}_{rs}^N & \bar{\mathbf{Z}}_{rs}^1 \end{bmatrix}, \quad r,s = 1, 2 \tag{93}$$

In addition to this, $\bar{\mathbf{E}}_0 = \mathbf{T}^{MT}\mathbf{E}_0\mathbf{T}^M$, $\bar{\mathbf{E}}_1 = \mathbf{T}^{MT}\mathbf{E}_1\mathbf{T}^M$, and $\bar{\mathbf{E}}_2 = \mathbf{T}^{MT}\mathbf{E}_2\mathbf{T}^M$ are also block circulant [23].

To partition the computing processes of Eqs. (86) and (91), $\bar{\mathbf{u}}_h^{km}$ and $\bar{\mathbf{P}}^{*km}$ are transformed via $[e_1, e_2, \ldots, e_N]$ that is a group of complete orthogonal basis vectors [23, 24]

$$\bar{\mathbf{u}}_h^{km} = \mathbf{E}^M\underline{\mathbf{u}}_h^{km} \tag{94a}$$

$$\bar{\mathbf{P}}^{*km} = \mathbf{E}^M\underline{\mathbf{P}}^{*km} \tag{94b}$$

where

$$\mathbf{E}^M = [e_{\mathrm{rs}} \cdot \mathbf{I}^M]^{\mathrm{T}} \tag{95}$$

$\mathbf{I}^M$ is a $2M$-dimensional unit matrix. e_{rs} is the sth element of $\mathbf{e}_r$ defined by

$$\mathbf{e}_1 = [1, 1, \ldots, 1]^{\mathrm{T}}\Big/\sqrt{N}$$

$$\mathbf{e}_{2i} = \sqrt{2/N}\left[\cos i\theta_1, \cos i\theta_2, \ldots, \cos i\theta_N\right]^{\mathrm{T}}$$

$$\mathbf{e}_{2i+1} = \sqrt{2/N}\left[\sin i\theta_1, \sin i\theta_2, \ldots, \sin i\theta_N\right]^{\mathrm{T}}$$

$$\theta_k = (k-1)\theta \ \ (k = 1, 2, \ldots, N)$$

$$\mathbf{e}_N = [1, -1, 1, \ldots, -1]^{\mathrm{T}}\Big/\sqrt{N} \quad \text{(when is even)} \tag{96}$$

where $\left[(N-1)/2\right]$ refers to the largest integer that does not exceed $(N-1)/2$.

Substituting Eqs. (94a, b) into Eq. (86) and left-multiplying both sides of Eq. (86) with $\mathbf{E}^{MT}$ then yield

$$\underline{\mathbf{K}}\underline{\mathbf{u}}_h^{km} = \underline{\mathbf{P}}^{*km} \tag{97}$$

where

$$\underline{\mathbf{K}} = \underline{\mathbf{Q}}\underline{\mathbf{\Phi}}^{-1} = \mathbf{E}^{MT}\bar{\mathbf{K}}\mathbf{E}^{M} = \mathbf{E}^{MT}\bar{\mathbf{Q}}\bar{\mathbf{\Phi}}^{-1}\mathbf{E}^{M} \tag{98}$$

$$\underline{Q} = \mathbf{E}^{MT}\bar{Q} \tag{99a}$$

$$\underline{\mathbf{\Phi}} = \mathbf{E}^{MT}\bar{\mathbf{\Phi}} \tag{99b}$$

$$\bar{\phi} = \mathbf{E}^{M}\underline{\phi} \tag{100a}$$

$$\bar{\mathbf{q}} = \mathbf{E}^{M}\underline{\mathbf{q}} \tag{100b}$$

Left multiplying two sides of Eq. (91) with $\mathbf{E}^{MT}$ and substituting Eqs. (100a, b) into Eq. (91), one has

$$\begin{bmatrix} \underline{\mathbf{Z}}_{11} & \underline{\mathbf{Z}}_{12} \\ \underline{\mathbf{Z}}_{21} & \underline{\mathbf{Z}}_{22} \end{bmatrix} \begin{Bmatrix} \underline{\phi} \\ \underline{\mathbf{q}} \end{Bmatrix} = \lambda \begin{Bmatrix} \underline{\phi} \\ \underline{\mathbf{q}} \end{Bmatrix} \tag{101}$$

where

$$\underline{\mathbf{Z}}_{ij} = \mathbf{E}^{M\mathrm{T}}\bar{\mathbf{Z}}_{ij}\mathbf{E}^{M} \quad i,j = 1,2 \tag{102}$$

$\underline{\mathbf{Z}}_{11}$, $\underline{\mathbf{Z}}_{12}$, $\underline{\mathbf{Z}}_{21}$ and $\underline{\mathbf{Z}}_{22}$ are block-diagonal, i.e.

$$\underline{\mathbf{Z}}_{ij} = \sum_{r=0}^{[N/2]} \oplus \underline{\mathbf{Z}}_{ij}^{rr} \quad i,j = 1,2 \tag{103}$$

where $\oplus$ represents the direct sum of matrices.

Equation (103) indicates that Eq. (101) can be partitioned into $[(N+2)/2]$ independent sub-eigenvalue-problems, i.e.

$$\begin{bmatrix} \underline{\mathbf{Z}}_{11}^{rr} & \underline{\mathbf{Z}}_{12}^{rr} \\ \underline{\mathbf{Z}}_{21}^{rr} & \underline{\mathbf{Z}}_{22}^{rr} \end{bmatrix} \begin{Bmatrix} \underline{\phi}_r^* \\ \underline{\mathbf{q}}_r^* \end{Bmatrix} = \lambda \begin{Bmatrix} \underline{\phi}_r^* \\ \underline{\mathbf{q}}_r^* \end{Bmatrix} \quad r = 0,1,2,\ldots,\left[N/2\right] \tag{104}$$

Based on Laplace theorem [31], it can be proved that the eigenvalues of Eq. (101) are the eigenvalues of Eq. (104). $\underline{\mathbf{\Phi}}$ consists of

$$\underline{\mathbf{\Phi}} = \left[\underline{\mathbf{\Phi}}^1, \underline{\mathbf{\Phi}}^2, \ldots, \underline{\mathbf{\Phi}}^N\right] \tag{105a}$$

$$\underline{\Phi}^j = [\underline{\phi}_1^j, \ldots \underline{\phi}_i^j, \ldots \underline{\phi}_{2M}^j] \quad (j = 1,2,\ ,N) \tag{105b}$$

$\underline{\mathbf{\Phi}}^1$ is related to $r = 0$ via

$$\underline{\phi}_i^{1\mathrm{T}} = \left[\phi_{0i}^{*\mathrm{T}}, 0^{2\mathrm{T}}, \ldots, 0^{N\mathrm{T}}\right] \tag{106a}$$

When N is even, $\underline{\mathbf{\Phi}}^N$ is related to $r = N/2$ via

$$\underline{\phi}_i^{N\mathrm{T}} = \left[0^{1\mathrm{T}}, \ldots 0^{(N-1)\mathrm{T}}, \underline{\phi}_{ri}^{*\mathrm{T}}\right] \tag{106b}$$

$\underline{\mathbf{\Phi}}^j$ ($j = 2r$, or $2r+1$) is related to $r = 1, 2, \ldots, \left[(N-1)/2\right]$ via

$$\underline{\phi}_i^{2r\mathrm{T}} = \left[0^{1\mathrm{T}}, \ldots, 0^{(2r-1)\mathrm{T}}, \underline{\phi}_{ri}^{*\mathrm{T}}, 0^{(2r+2)\mathrm{T}}, \ldots, 0^{N\mathrm{T}}\right] \tag{106c}$$

$$\underline{\phi}_i^{(2r+1)\mathrm{T}} = \left[0^{1\mathrm{T}}, \ldots, 0^{(2r-1)\mathrm{T}}, \underline{\phi}_{r(2M+i)}^{*\mathrm{T}}, 0^{(2r+2)\mathrm{T}}, \ldots, 0^{N\mathrm{T}}\right] \tag{106d}$$

Definitely, $\underline{\phi}_i^{j\mathrm{T}}$ ($i = 1, 2, \ldots 2M$; $j = 1, 2, \ldots N$) is linearly independent of each other.

$\underline{\mathbf{Q}}$ is constituted by

$$\underline{\mathbf{Q}} = \left[\underline{\mathbf{Q}}^1, \underline{\mathbf{Q}}^2, \ldots, \underline{\mathbf{Q}}^N\right] \tag{107}$$

where $\underline{\mathbf{Q}}^j = [\underline{\mathbf{q}}_1^j, \ldots \underline{\mathbf{q}}_i^j, \ldots \underline{\mathbf{q}}_{2M}^j]$, and $\underline{\mathbf{q}}_i^j$ can be constituted in a similar way.

Therefore,

$$\underline{\mathbf{\Phi}} = \begin{bmatrix} \underline{\mathbf{\Phi}}_0^* & & & \\ & \underline{\mathbf{\Phi}}_1^* & & \\ & & \cdots & \\ & & & \underline{\mathbf{\Phi}}_{nsp}^* \end{bmatrix} \tag{108a}$$

$$\underline{\mathbf{Q}} = \begin{bmatrix} \underline{\mathbf{Q}}_0^* & & & \\ & \underline{\mathbf{Q}}_1^* & & \\ & & \cdots & \\ & & & \underline{\mathbf{Q}}_{nsp}^* \end{bmatrix} \tag{108b}$$

The dimension of $\underline{\mathbf{\Phi}}_0^*$ is $2M$, the dimension of $\underline{\mathbf{\Phi}}_i^*$, $(i = 1, \ldots, nsp-1)$ is $2 \times 2M$, and the dimension of $\underline{\mathbf{\Phi}}_{nsp}^*$ is $2M$ or $2 \times 2M$ if N is even or odd.

Substituting Eqs. (108a, b) into Eq. (98) then gives

$$\underline{\mathbf{K}} = \begin{bmatrix} \underline{\mathbf{Q}}_0^*\underline{\mathbf{\Phi}}^{*-1}{}_0 & & & \\ & \underline{\mathbf{Q}}_1^*\underline{\mathbf{\Phi}}^{*-1}{}_1 & & \\ & & \cdots & \\ & & & \underline{\mathbf{Q}}_{nsp}^*\ \underline{\mathbf{\Phi}}^{*-1}{}_{nsp} \end{bmatrix} \tag{109}$$

and

$$\underline{\mathbf{K}}_{rr}\left[\underline{\mathbf{u}}_h^{km}\right]_r = \left[\underline{\mathbf{P}}^{*km}\right]_r \quad r = 0, 1, 2, \ldots, [N/2] \tag{110}$$

where

$$\left[\underline{\mathbf{u}}_h^{km}\right]^1 = \left[\underline{\mathbf{u}}_h^{km}\right]_0; \left[\underline{\mathbf{u}}_h^{km}\right]^N = \left[\underline{\mathbf{u}}_h^{km}\right]_{N/2} \text{ (when } N \text{ is even)}$$

$$\left[\left[\underline{\mathbf{u}}_h^{km}\right]^{p\mathrm{T}}, \left[\underline{\mathbf{u}}_h^{km}\right]^{q\mathrm{T}}\right] = \left[\underline{\mathbf{u}}_h^{km}\right]_r^{\mathrm{T}} \tag{111}$$
$$\left(p = 2r,\ q = 2r+1;\ r = 1, 2, \ldots, [(N-1)/2]\right)$$

When load distribution and displacement constraint are both rotationally symmetric,

$$\left[\overline{P^{*km}}\right]^1 = \left[\overline{P^{*km}}\right]^2 = \cdots = \left[\overline{P^{*km}}\right]^N \tag{112}$$

$$\left[\underline{P^{*km}}\right]^2 = \left[\underline{P^{*km}}\right]^3 = \cdots = \left[\underline{P^{*km}}\right]^N = \mathbf{0}, \quad \left[\underline{P^{*km}}\right]^1 = \sqrt{N}\left[\underline{P^{*km}}\right]^1 \tag{113}$$

One only needs to solve the first sub-problem in Eq. (113) because other [*N*/2] subproblems have zero solutions.

No matter the load distribution and displacement constraint are rotationally symmetric or not, the solution of Eq. (62) can be computationally partitioned via Eq. (101), and **K**can be generated via

$$\bar{\mathbf{K}} = \mathbf{E}^M \underline{\mathbf{K}} \mathbf{E}^{M\mathrm{T}} \tag{114}$$

and

$$\mathbf{K} = \mathbf{T}^M \bar{\mathbf{K}} \mathbf{T}^{M\mathrm{T}} \tag{115}$$

When displacement constraints are rotationally symmetric, Eq. (65) can be solved by solving a series of smaller independent problems described by Eq. (110). When displacement constraints are not rotationally symmetric, Eq. (65) can be solved using **K** generated via Eqs. (114–115).

Consequently, the computing expense of EFG-SBM via a combination with a temporally piecewise adaptive algorithm can be significantly reduced by the exploration of cyclic symmetry.

6 Numerical examples

Two numerical examples are given to verify the proposed algorithm, and all the computations are carried out on a PC with 3.10 GHz CPU and 4G RAM.

Both are the creep analyses, assuming that $\mathbf{b} = \mathbf{0}$, $\tilde{\mathbf{u}} = 0$, $\tilde{\mathbf{p}}$ is time independent and $\beta = 1 \times 10^{-5}$.

6.1 Numerical example 1

Consider a regular octagonal gear subjected to a uniform pressure on one edge as shown in Fig. 4 where $q = 1$ N/m,

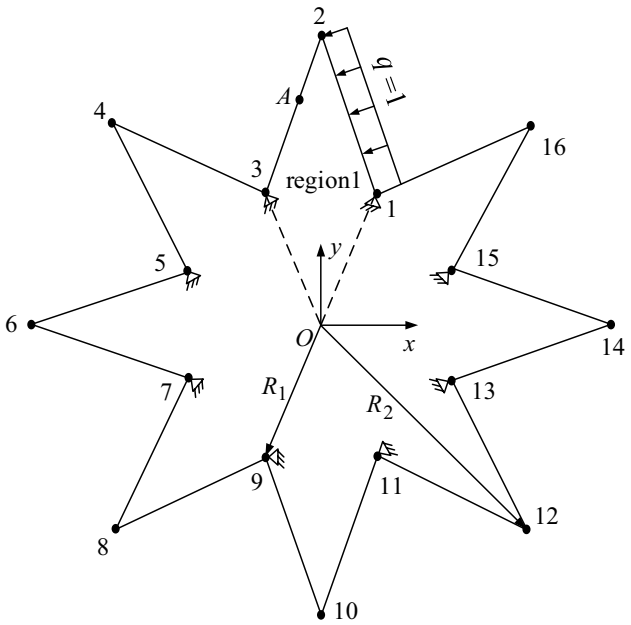

Fig. 4 A regular octagonal gear in the numerical example 1

$R_1 = 1$ m and $R_2 = 2$ m. There are eight hinged supports at concave corner points. Plane stress conditions are assumed with $E_1 = 1000$ Pa, $E_2 = 2000$ Pa, $\eta_1 = 2 \times 10^4$ N s/m^2 and $\nu = 0.3$. 30 nodes are uniformly arranged at every edge. All geometric, physical properties and constraints are cyclic symmetry with $N = 8$.

The result obtained by the proposed approach is compared with a reference solution [28] whose elastic solution is given by ANSYS with 12,779 nodes and 4078 8-node-quadrangle finite elements. The solution of u_x at point 2 is presented in Table 1 and Fig. 5. The maximum relative error between the proposed algorithm and reference solution is about 3.76%. The solution of u_y at node *A* is presented in

Table 1 Numerical comparison of u_x at point 2 ($x_2 = 0$, $y_2 = 2$ m)

t (s)	Proposed algorithm (mm)	Reference solution (mm)
0	− 11.186	− 11.623
50	− 31.722	− 32.961
100	− 33.408	− 34.712
150	− 33.546	− 34.856
200	− 33.558	− 34.868
250	− 33.558	− 34.869
325	− 33.559	− 34.869
400	− 33.559	− 34.869
475	− 33.559	− 34.869
550	− 33.559	− 34.869
625	− 33.559	− 34.869

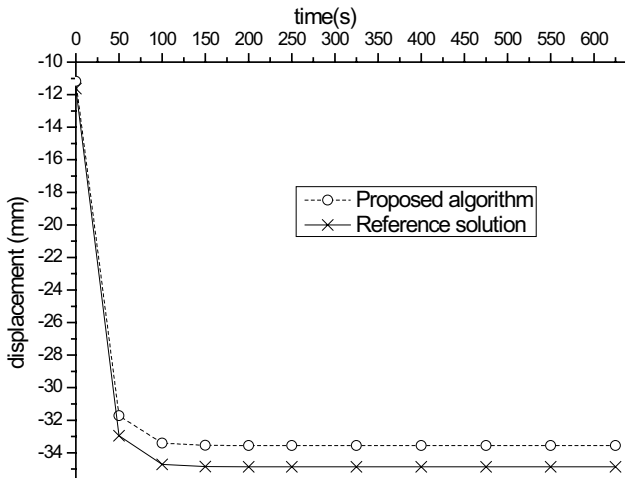

Fig. 5 Numerical comparison of u_x at point 2

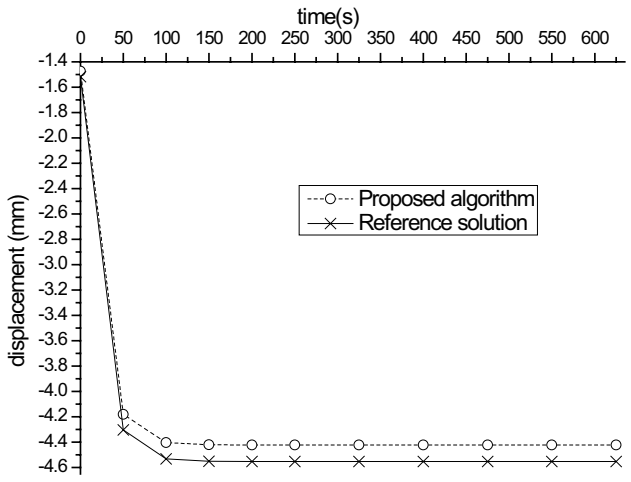

Fig. 6 Numerical comparison of u_y at node A

Table 2 Numerical comparison of u_y at node A ($x_A = -0.15$ m, $y_A = 1.57$ m)

t (s)	Proposed algorithm (mm)	Reference solution (mm)
0	−1.4740	−1.5174
50	−4.1801	−4.3031
100	−4.4022	−4.5318
150	−4.4204	−4.5505
200	−4.4219	−4.5521
250	−4.4220	−4.5522
325	−4.4220	−4.5522
400	−4.4221	−4.5522
475	−4.4220	−4.5522
550	−4.4220	−4.5522
625	−4.4221	−4.5522

Table 2 and Fig. 6. The maximum relative error is about 2.86%. Initial time step is assumed with $T_0 = 100$ s and then the algorithm would adaptively decrease half size of initial time step to restart computing at the first time interval. Size of time step is adaptively incremented to 75 s at the sixth time interval. A comparison of computing expense is given in Table 3.

6.2 Numerical example 2

Consider a regular octagonal plane plate subjected to a uniform tangential force $p = 1$ N/m along the boundaries, as shown in Fig. 7, where $R = 1$ m. Plane stress conditions are assumed with $E_1 = 1000$ Pa, $E_2 = 2000$ Pa, $\eta_1 = 2 \times 10^4$ N s/m^2 and $\nu = 0.3$. Sizes of time steps are adaptive and initial time step is assumed with $T_0 = 100$ s. There are eight hinged supports along the boundaries. 100 nodes are uniformly

Table 3 A comparison of CPU time

EFG-SBM solution method	Computing expense (s)	
	Without partitioning	Partitioning
Generating $\mathbf{E}_r$ ($r = 0, 1, 2$) and $\mathbf{Z}_{ij}$ ($i, j = 1, 2$)	23.582	4.612
Solving Eigenvalue-equation	19.735	1.868
Solving system equation at the beginning time	0.070	0.019
1st time interval	2.703	0.525
2nd time interval	0.936	0.190
3rd time interval	0.922	0.170
4th time interval	0.854	0.178
5th time interval	0.661	0.098
6th time interval	1.452	0.218
7th time interval	0.463	0.105
8th time interval	0.674	0.122
9th time interval	0.879	0.100
10th time interval	0.637	0.104
Full time	60.234	24

arranged at each sub-cyclic symmetry region. All geometric, physical properties, constraints and load conditions are cyclic symmetry with $N = 8$.

The result obtained by the proposed approach is compared with a reference solution [28] whose elastic solution is given by ANSYS with 73,033 nodes and 24,448 8-node-quadrangle finite elements. The solution of u_y at node A is presented in Table 4 and Fig. 8. The maximum relative error between the proposed algorithm and reference solution is about 2.07%. The solution of u_x at point B is presented in Table 5 and Fig. 9. The maximum relative error is about 1.87%. A comparison of computing expense is given in

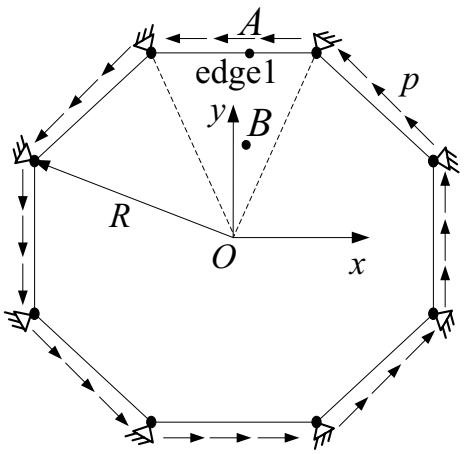

Fig. 7 A regular 8-edge plate subjected to a tangential distributed force along the boundaries

Table 4 Numerical comparison of u_y at node A ($x_A = 0.07654$ m, $y_A = 0.92388$ m)

t (s)	Proposed algorithm (10^{-5} m)	Analytical solution (10^{-5} m)
50	5.7075	5.8282
100	6.0108	6.1379
150	6.0357	6.1633
200	6.0378	6.1654
250	6.0379	6.1656
325	6.0379	6.1656
400	6.0379	6.1656
475	6.0379	6.1656
550	6.0379	6.1656
625	6.0379	6.1656

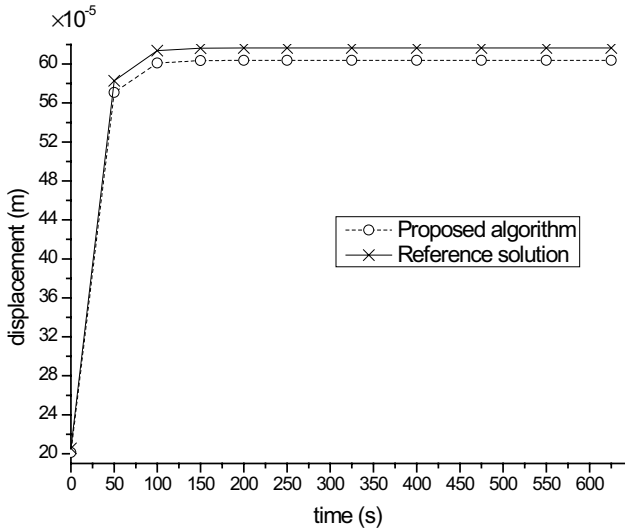

Fig. 8 Numerical comparison of u_y at node A

Table 5 Numerical comparison of u_x at point B ($x_B = 0.08419$ m, $y_B = 0.46194$ m)

t (s)	Proposed algorithm (10^{-4} m)	Analytical solution (10^{-4} m)
50	−3.0470	−2.9912
100	−3.2090	−3.1502
150	−3.2223	−3.1632
200	−3.2233	−3.1643
250	−3.2234	−3.1644
325	−3.2234	−3.1644
400	−3.2234	−3.1644
475	−3.2234	−3.1644
550	−3.2234	−3.1644
625	−3.2234	−3.1644

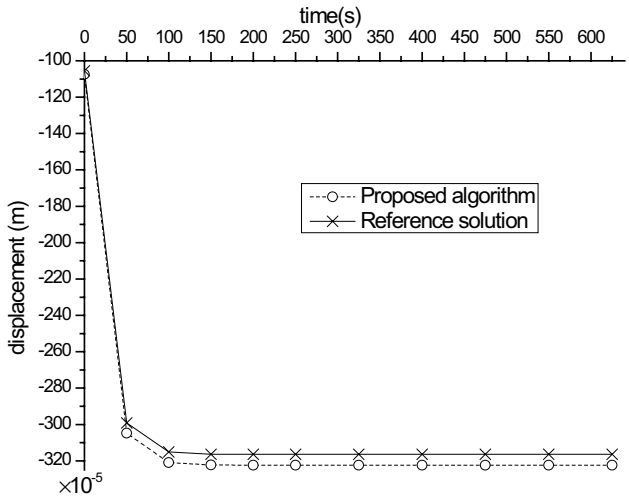

Fig. 9 Numerical comparison of u_x at point B

Table 6. Figures 10 and 11 exhibit the adaptive procedures of size of time interval and power order, respectively.

7 Computing remarks

1. In comparison with ANSYS, a sufficient computing accuracy can be provided by the proposed algorithm, as shown in Tables 1, 2, 4 and 5, and Figs. 5, 6, 8 and 9. Both examples need a FEM-based convergent elastic solution that requires more DOF of unknowns in comparison with EFG-SBM. In example 2, the proposed algorithm needs only 1600 DOF, the convergent elastic solution given by ANSYS needs 146,066 DOF. An ANSYS-based viscoelastic analysis for this problem may lead to more DOF.

Table 6 A comparison of CPU time

EFG-SBM solution method	Computing expense (s)		
	Without partitioning	Partitioning (1)	Partitioning (2)
Generating $\mathbf{E}_r$ ($r=0, 1, 2$) and $\mathbf{Z}_{ij}$ ($i,j=1, 2$)	78.734	15.603	15.603
Solving eigenvalue-equation	69.939	5.068	0.415
Solving system equation at the beginning time	0.273	0.048	0.002
1st time interval	9.315	1.271	0.420
2nd time interval	3.382	0.450	0.154
3rd time interval	3.152	0.441	0.144
4th time interval	2.705	0.368	0.131
5th time interval	1.969	0.279	0.098
6th time interval	3.898	0.525	0.174
7th time interval	1.665	0.264	0.087
8th time interval	2.180	0.327	0.107
9th time interval	1.941	0.267	0.097
10th time interval	2.073	0.276	0.105
Full time	218.922	103.359	60.250

The cyclic symmetry of loads distribution is exploited in Eq. (2)

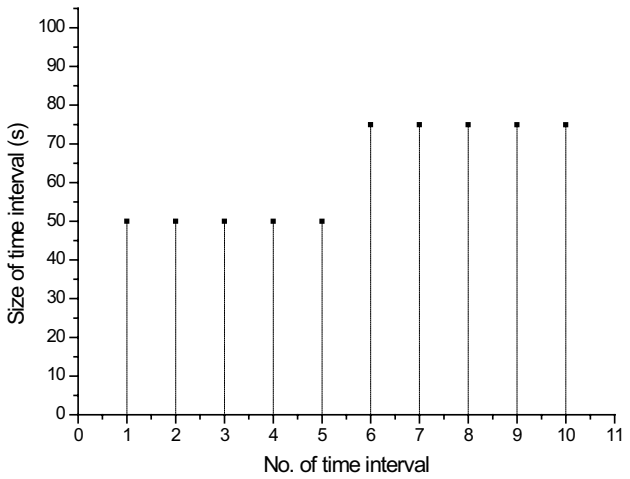

Fig. 10 Size of time interval

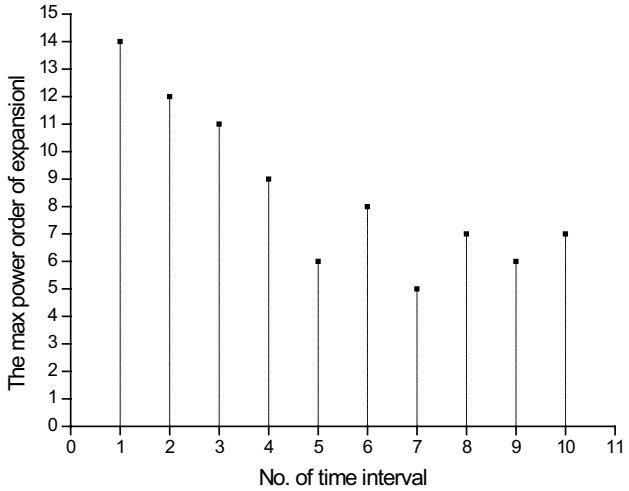

Fig. 11 The max power order of expansion at each time interval

2. Figure 10 exhibits the adaptive procedure of size of time interval. When creep tends to a stable state, the variation of displacement becomes less and less, and the size of time step gradually becomes relative bigger in the adaptive process. When the size of time step is enlarged, more recursive computing may be required to satisfy Eq. (77) as shown in Fig. 11. In Table 6 of example 2, we see that the computational expense at the first time interval is the most more than other time intervals. It is because that the convergence criterion (77) could not be satisfied even when $R=$ mm and then the algorithm would adaptively decrease half size of initial time step to restart computing as shown in Fig. 10.

3. Exploiting cyclical symmetry, a substantial speed-up can be achieved in solving viscoelastic problems by EFG-SBM, as shown in Tables 3 and 6.

8 Conclusions

The objective of this paper is to utilize the cyclic symmetry to improve the computational efficiency in a EFG-SBM based recursive solution process for the numerical viscoelastic analysis. The major contributions include:

1. An EFG-SBM-based partitioning algorithm via a combination with a temporally piecewise adaptive algorithm is presented. Then both eigenvalue and system equations can be partitioned into a number of smaller problems which can be solved independently, and thereby the computational efficiency is significantly improved;
2. In the whole recursive process of the partitioning algorithm, the eigenvalue equations and the stiffness matrix need to be solved only one time.
3. Since the proposed partitioning algorithm facilitates to be parallelized, a higher computing efficiency can be expected.

Acknowledgements The research leading to this paper is funded by NSF [11572068], NKBRSF [2015CB057804], Natural Science Funding of Liaoning Province [201602115].

References

1. Christensen RM (1982) Theory of viscoelasticity: an introduction. Academic Press, Cambridge
2. Ahmadi E, Barikloo H, Kashfi M (2016) Viscoelastic finite element analysis of the dynamic behavior of apple under impact loading with regard to its different layers. Comput Electron Agr 121:1–11
3. Ashrafia H, Shariyata M, Khalilia SMR, Asemib K (2013) A boundary element formulation for the heterogeneous functionally graded viscoelastic structures. Appl Math Comput 225:246 – 62
4. Han Z, Yang HT, Liu L (2006) Solving viscoelastic problems with cyclic symmetry via a precise algorithm and EFGM. Acta Mech Sinica 22:170–176
5. Guo XF, Yang HT (2016) A combination of EFG-SBM and a temporally-piecewise adaptive algorithm to solve viscoelastic problems. Eng Anal Bound Elem 67:43–52
6. Xu QW, Prozzi JA (2015) A time-domain finite element method for dynamic viscoelastic solution of layered-half-space responses under loading pulses. Comput Struct 160:20–39
7. Kalyania VK, Pallavikab SK, Chakrabortyb (2014) Finite-difference time-domain method for modelling of seismic wave propagation in viscoelastic media. Appl Math Comput 237:133 – 45
8. Wolf JP, Song Ch (1996) Consistent infinitesimal finite element cell method: three dimensional vector wave equation. Int J Numer Methods Eng 39:2189–2208
9. Wolf JP, Song Ch (1996) Finite-element modelling of unbounded media. Wiley, Chichester
10. Wolf JP, Song Ch (2000) The scaled boundary finite-element method—a primer: derivation. Comput Struct 78:191–210
11. Wolf JP, Song Ch (2001) The scaled boundary finite-element method—a fundamental solution-less boundary-element method. Comput Methods Appl Mech Eng 190:5551–5568
12. Deeks AJ, Wolf JP (2002) A virtual work derivation of the scaled boundary finite-element method for elastostatics. Comput Mech 28:489–504
13. Deeks AJ, Augarde CE (2005) A meshless local Petrov-Galerkin scaled boundary method. Comput Mech 36:159–170
14. Lin G, Lu S, Liu J (2015) Duality system-based derivation of the modified scaled boundary finite element method in the time domain and its application to anisotropic soil. Appl Math Model 40:5230–5255
15. Ooi ET, Yang ZJ (2010) Modelling crack propagation reinforced concrete using a hybrid finite element-scaled boundary finite element method. Eng Fract Mech 78:252 – 73
16. Ooi ET, Yang ZJ (2010) A hybrid finite element-scaled boundary finite element method for crack propagation modelling. Comput Method Appl Mech Eng 199:1178–1192
17. Ooi ET, Yang ZJ (2011) Modelling dynamic crack propagation using the scaled boundary finite element method. Int J Numer Method Eng 88:329–349
18. Yang ZJ, Deeks AJ, Hao H (2007) Transient dynamic fracture analysis using scaled boundary finite element method: a frequency domain approach. Eng Fract Mech 74:669 – 87
19. Bazyar MH, Talebi A (2015) Scaled boundary finite-element method for solving nonhomogeneous anisotropic heat conduction problems. Appl Math Model 39:7583–7599
20. Hell S, Becker W (2015) The scaled boundary finite element method for the analysis of 3D crack interaction. J Comput Sci 9:76–81
21. Zhong WX, Qiu CH (1983) Analysis of symmetric or partially symmetric structures. Comput Methods Appl Mech Eng 38:1–18
22. Wu GF, Yang HT (1994) The use of cyclic symmetry in two-dimensional elastic stress analysis by BEM. Int J Solids Struct 31:279–290
23. Guo XF, He YQ, Yang HT (2015) An EFG-SBM based partitioning algorithm for two-dimensional elastic analysis of cyclically symmetrical structures. Eng Computation 32(2):452–472
24. Yang HT, Guo XF, He YQ (2015) An EFG-SBM based partitioning algorithm for heat transfer analysis of cyclically symmetrical structures. Finite Elem Anal Des 93:42–49
25. Chongshuai Wang Y, He H Yang (2016) A piecewise partitioning scaled boundary finite element algorithm to solve viscoelastic problems with cyclic symmetry. Eng Anal Bound Elem 73:120–125
26. Zienkiewicz OC, Morgan K (1983) Finite element and approximation. Wiley, New York
27. Shames IH, Cozzarelli FA (1992) Elasitc and inelastic stress analysis. Prentice Hall, Englewood Cliffs
28. Cai E (1989) The foundation of viscoelastic mechanics. Beihang University Press, Beijing
29. Belytschko T, Lu YY, Gu L (1994) Element-free Galerkin methods. Int J Numer Methods Eng 37:229–256
30. Belytschko T, Krongauz Y, Organ D (1996) Meshless methods: an overview and recent developments. Comput Methods Appl Mech Eng 139:3–47
31. Lin HT, Li CY, Zhang NY (1994) Linear algebra. Tian Jin University Press, Tian Jin