Engineering Analysis with Boundary Elements 139 (2022) 152–168

Contents lists available at ScienceDirect

Engineering Analysis with Boundary Elements

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

Implementation of a dual interpolation boundary face method by discontinuous meshes

Jianming Zhang *, Pengfei Chai, Rui He, WeiCheng Lin, Chuangming Ju, Baotao Chi

State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, College of Mechanical and Vehicle Engineering, Hunan University, Changsha 410082, China

ARTICLE INFO

Keywords:
Discontinuous mesh
Binary-tree structure
Hang points
Dual interpolation boundary face method
Hermite-type approximation

ABSTRACT

In this paper, the discontinuous meshes are applied to CAE analysis. Besides that, a dual interpolation boundary face method based on the Hermite-type moving-least-squares method (DiBFM-HMLS) for the 3D elasticity problem is firstly proposed. The DiBFM-HMLS offers an interpolation scheme for the hang points in the discontinuous meshes. Different from the previous mesh division method, the binary-tree structure is used to obtain the discontinuous meshes. Our mesh generation method can effectively avoid model repairing and simplification for geometric structures with small features and defects while ensuring a real automatic mesh division. Compared with the continuous mesh, the discontinuous meshes possess strong geometric adaptability for arbitrarily complicated structures and can provide the possibility for the full-automatic CAE analysis. Successful numerical examples are used to illustrate the accuracy and reliability of the presented methods in solving various kinds of problems and reveal the excellent performance of the discontinuous meshes.

1. Introduction

The automatic grids division technology is one of the keys to CAD/CAE integration. Many mesh generation methods are potential for this integration process such as AFM [1,2], Delaunay triangulation [3,4], and mapping method [5] for continuous meshes, the quad-tree method [6] for discontinuous meshes. As the widely used tool in numerical algorithms, continuous meshes can be divided into two categories: (I) structured grids, and (II) unstructured grids. Structured grids can reach high calculation accuracy and are frequently used in critical portions of a component such as the stress concentration areas. However, its automatic property is just available in geometric models with no defect. Unstructured grids are easily implemented and usually employed in CAE analysis, but they still can't be performed smoothly when the CAD models are not "clean".

The discontinuous meshes not only are flexible to be carried out but also can largely decrease the difficulty of mesh division. The boundary integral equation (BIE) of the boundary element method (BEM) is versatile [7–10] and allows the trial functions to be discontinuous. Thus, different kinds of elements can be used in BEM analysis (e.g. continuous and discontinuous elements [11–13]). These features of BIE provide convenience for the application of the discontinuous mesh in CAE analysis by BEM. Besides, it is not necessary to fill in the transition areas (see Fig. 1) for discontinuous meshes (e.g. the Quad-tree grid generation [14–16]) to acquire high-quality computational grids. Based on the above advantages of BIE, the discontinuous mesh is employed in our research directly and without additional processing.

For the non-interpolation hang points in the discontinuous grids, the meshless interpolation method is used to approximate the physical variables. So far, there are many methods for reference concerning meshless interpolation. The moving-least-squares (MLS) approximation proposed by Lancaster and Salkauskas [17] offers a practical measure for scattered point interpolation. Based on the MLS approximation, Zhang [18] presented the dual interpolation boundary face method (DiBFM-MLS), which has shown superiority in solving various kinds of physical problems [18–21]. In this method, the BFM [22] guarantees that the coordinate of Gauss integral points and Jacobian are taken directly from real geometric boundary instead of an element, thus more precise results can be obtained. However, as the MLS is built on the parameter space, an obvious limitation for DiBFM-MLS exists in dealing with the structure with small feature sizes [23,24]. In addition, the stability of the MLS will be affected following the increase of the

* Corresponding author at: Jianming Zhang, College of Mechanical and Vehicle Engineering, Hunan University, Changsha 410082, China. Telephone: +86-731-88823061.

E-mail address: zhangjm@hnu.edu.cn (J. Zhang).

https://doi.org/10.1016/j.enganabound.2022.03.020
Received 29 November 2021; Received in revised form 23 February 2022; Accepted 20 March 2022
Available online 31 March 2022

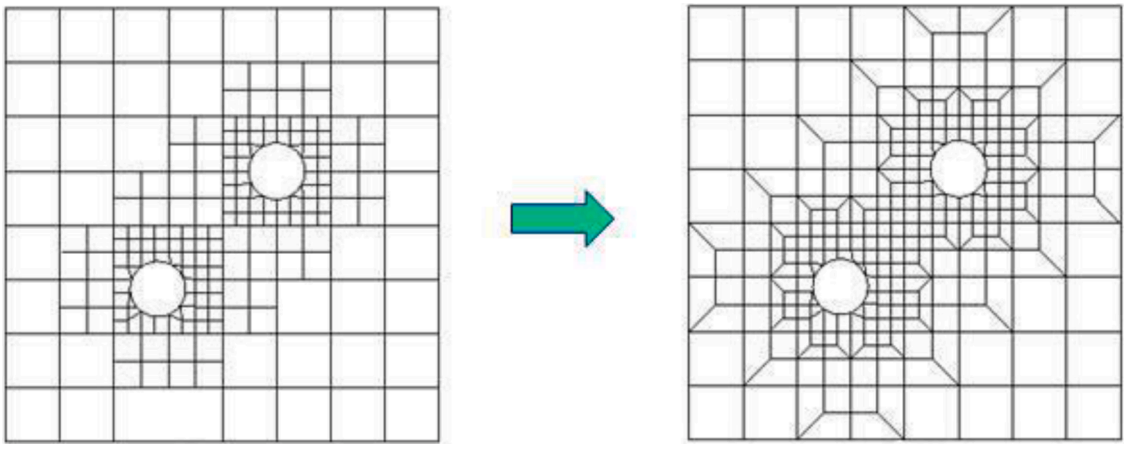


Fig. 1. Mesh transition

interpolation points [25,26], which also restricts the applicability of DiBFM-MLS.

Different from the MLS, the Hermite-type moving-least-squares (HMLS) presented by Li and Aluru [27,28] introduces the derivative component into the least square method, in which the singularity of the momentum matrix is effectively avoided. Furthermore, the normal equations are established on the Cartesian coordinate rather than the parameter space, which guarantees that HMLS can be applied to analyze the structures with small feature size. Since the employment of the linear polynomial basis in HMLS by Li and Aluru, the accuracy can't always be guaranteed in some cases. Zhang [23,24] proposed the dual interpolation boundary face method with HMLS (DiBFM-HMLS) by incorporating incomplete quadratic basis in HMLS, and substantial numerical results for 2D potential and elasticity problems have been achieved.

Previous researches [23,24], about DiBFM-HMLS, are mainly based on the continuous meshes, and only for 2D problems. In this paper, DiBFM-HMLS based on the discontinuous meshes for the 3D elasticity problem is proposed, and the binary-tree structure [29] is used to get the discontinuous meshes. Due to strong geometric adaptability and easy implementation, the discontinuous meshes are very suitable for geometric structures with small features and defects. Simultaneously, DiBFM-HMLS provides an effective measure for dealing with hang points in discontinuous meshes and guarantees the successful application of the discontinuous mesh in solving arbitrarily complicated structures. Moreover, it can promote the process of CAD/CAE integration.

The paper is arranged as follows: the process of binary-tree mesh generation is illustrated in Section 2. The detail of DiBFM-HMLS for 3D elasticity problems and the measure for dealing with hang points are described in Section 3. In Section 4, the discretization and the reassembled form of the BIE for solving 3D elasticity problems are deduced. Several numerical examples are shown in Section 5. Conclusions and some discussions are given in Section 6.

2. Binary-tree mesh generation

In this section, an adaptive binary-tree mesh generation method is introduced. In the beginning, the initial root element is divided into two sub-elements, and this procedure is recursive according to the surface curvature, the curve curvature, or the area ratio of adjacent sub-domains. The annulus is an example to illustrate the binary-tree mesh generation (see Fig. 2). The process is mainly divided into five steps as described in Table 1.

The example of discontinuous meshes using binary-tree method for the flange is shown in Fig. 3.

The flow chart of binary-tree mesh generation process is given in Fig. 4: where S_i (i=1, 2) represents the area ratio between adjacent sub-domains. C_j (j=1, 2) is the curvature parameter (surface curvature and curve curvature). r_k (k=1, 2, 3) is the limitation prescribed by the user. l is the level between adjacent elements.

3. Interpolation scheme in DiBFM-HMLS

In this section, DiBFM-HMLS for the 3D elasticity problem is implemented based on the discontinuous meshes. In the dual interpolation elements, the hang points (see Fig. 5) are divided into two types according to the function: (i) those points are regarded as interpolation points for one element but not for another, and (ii) those points are considered as the common interpolation points for adjacent elements. All the hang points and geometric discontinuous boundary points are regarded as virtual points in DiBFM-HMLS. The approximation for unknown physical variables using dual interpolation element is described as follows:

3.1. The first-layer interpolation in DiBFM-HMLS

The conventional element interpolation is regarded as the first-layer interpolation in DiBFM-HMLS. For 3D elasticity problems, the displacement u_k and traction t_k, (k=1, 2, 3) in one element can be approximated by:

$$u_k(x,y,z) = u_k(\xi,\eta) = \sum_{m=1}^{ns} N_m^s(\xi,\eta) u_k\left(Q_m^s\right) + \sum_{n=1}^{nv} N_n^v(\xi,\eta) u_k\left(Q_n^v\right), \tag{1}$$

$$t_k(x,y,z) = t_k(\xi,\eta) = \sum_{m=1}^{ns} N_m^s(\xi,\eta) t_k\left(Q_m^s\right) + \sum_{n=1}^{nv} N_n^v(\xi,\eta) t_k\left(Q_n^v\right), \tag{2}$$

in Eqs (1), (2), nsandnv represent the total number of source and virtual points in the dual interpolation element. ξ,η is the coordinate of the unknown field point in the parametric space. Q_m^s and Q_n^v represent the m^{th} source point and n^{th} virtual point in the dual interpolation element. $N_m^s(\xi,\eta)$, $N_n^v(\xi,\eta)$are shape functions corresponding to $Q_m^s$and Q_n^v. $u_k(Q_m^s)$,$t_k(Q_m^s)$, $u_k(Q_n^v)$ and $t_k(Q_n^v)$are displacement and traction components along k^{th} direction, respectively. Since the values of virtual point are not independent, the second-layer interpolation for $u_k(Q_n^v)$ and $t_k(Q_n^v)$ will be constructed.

3.2. The Second-layer interpolation in DiBFM-HMLS

For 2D cases, the detail of the second-layer interpolation in DiBFM-HMLS has been illustrated [18,19]. The HMLS form of $u_k(Q_n^v)$ and $t_k(Q_n^v)$(k=1, 2, 3) in Eqs (1), (2) for 3D elasticity problems are defined as follows:

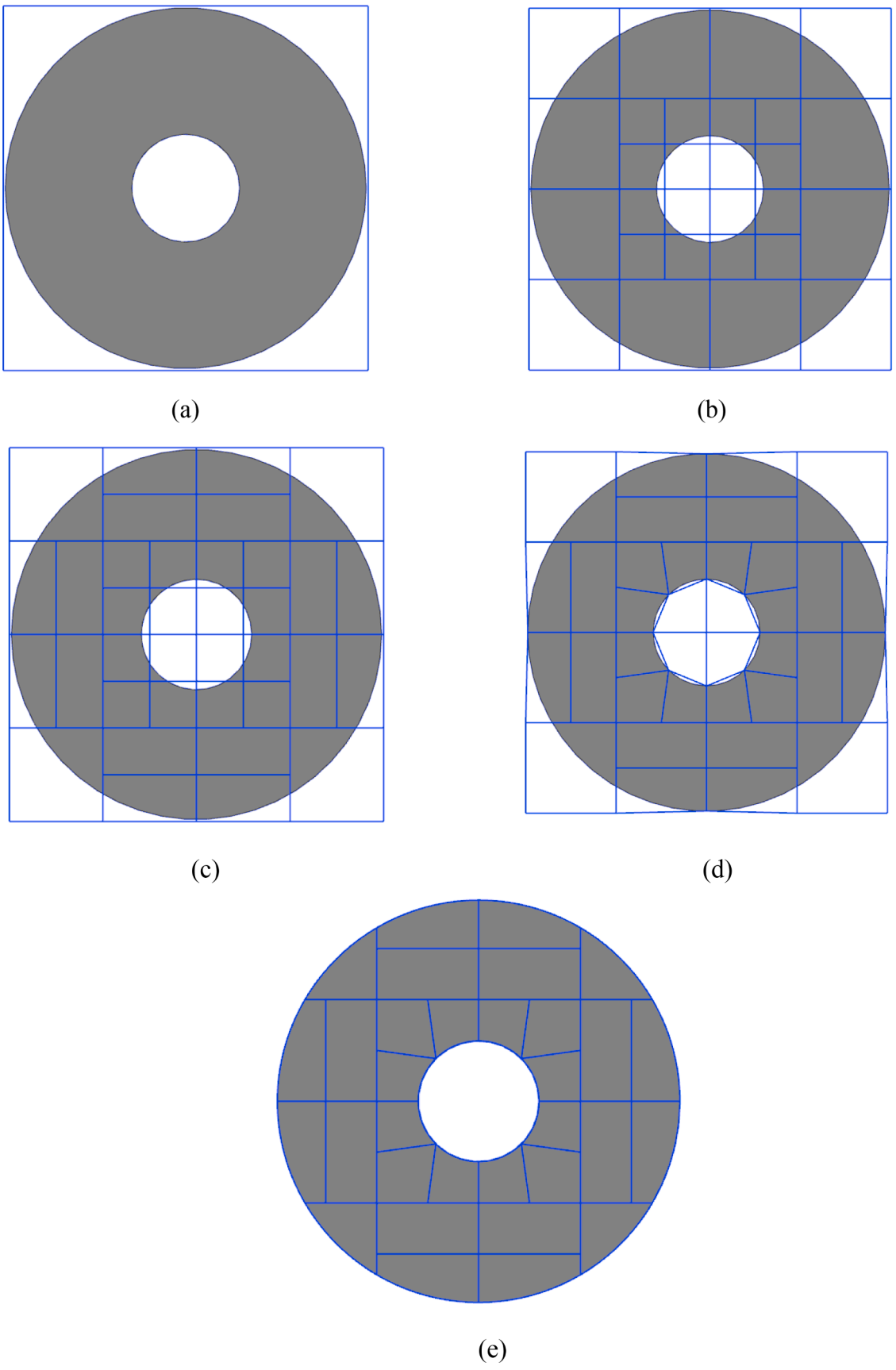


Fig. 2. Binary-tree mesh generation

Table 1
Process of binary-tree mesh generation

***Step* 1**: Getting the target domain of the parameter space of a 2D or 3D surface, and the target domain is surrounded by a rectangular box as small as possible. This rectangular box is called the root element (see Fig. 2(a)).
***Step* 2**: Subdividing the root element according to the surface curvature and the curve curvature, then judging whether each sub-element meets the criterion if the condition is satisfied, then subdividing the sub-domains, otherwise stopping it. The procedure is recursive until all the specified parameters by a user are satisfied. At the same time, in each subdivision, the area ratio between adjacent sub-domains is used as a restriction (see Fig. 2(b)).
***Step* 3**: Balancing the binary-tree mesh, the purpose of this work is to guarantee a natural transition in dimensions between adjacent leaf-elements (see Fig. 2(c)).
***Step* 4**: The node attributes of each sub-element are judged to determine whether the left-element is located in the domain, outside the domain, or on the domain boundary. After that, the mesh nodes of the boundary element are moved to the nearest boundary curve (see Fig. 2(d)).
***Step* 5**: All the boundary elements are trimmed using the template method, then final meshes are obtained (see Fig. 2(e)).

$$u_k(Q_n^v) = \sum_{j=1}^{3} \psi^{u_k u_j}(x^v, y^v, z^v)\widehat{\mathbf{u}}_j(Q^s) + \sum_{j=1}^{3} \psi^{u_k t_j}(x^v, y^v, z^v)\widehat{\mathbf{t}}_j(Q^s), \tag{3}$$

$$t_k(Q_n^v) = \sum_{j=1}^{3} \psi^{t_k u_j}(x^v, y^v, z^v, n^v)\widehat{\mathbf{u}}_j(Q^s) + \sum_{j=1}^{3} \psi^{t_k t_j}(x^v, y^v, z^v, n^v)\widehat{\mathbf{t}}_j(Q^s), \tag{4}$$

where Q^s is the source point in the influence domain of virtual point Q_n^v. The displacement component and traction component of $Q^s$are written as $\widehat{\mathbf{u}}_j(Q^s)$ and$\widehat{\mathbf{t}}_j(Q^s)$, respectively.$(x^v,y^v,z^v)$ is the coordinates of Q_n^v in Euclidean space and $n^v$is the outward normal. $\psi^{u_k u_j}(x^v,y^v,z^v)$,$\psi^{u_k t_j}(x^v,y^v, z^v)$,$\psi^{t_k u_j}(x^v,y^v,z^v,n^v)$ and $\psi^{t_k t_j}(x^v,y^v,z^v,n^v)$are shape function matrices, which can be written as:

$$\psi^{u_k u_j}(x^v, y^v, z^v) = \mathbf{p}_{u_k}^{\mathrm{T}}(x^v, y^v, z^v)\mathrm{C}_t^{-1}\mathbf{P}_{u_j}^{\mathrm{T}}, \tag{5}$$

$$\psi^{u_k t_j}(x^v, y^v, z^v) = \mathbf{p}_{u_k}^{\mathrm{T}}(x^v, y^v, z^v)\mathrm{C}_t^{-1}\mathbf{P}_{t_j}^{\mathrm{T}}, \tag{6}$$

$$\psi^{t_k u_j}(x^v, y^v, z^v, n^v) = \mathbf{p}_{t_k}^{\mathrm{T}}(x^v, y^v, z^v, n^v)\mathrm{C}_t^{-1}\mathbf{P}_{u_j}^{\mathrm{T}}, \tag{7}$$

$$\psi^{t_k t_j}(x^v, y^v, z^v, n^v) = \mathbf{p}_{t_k}^{\mathrm{T}}(x^v, y^v, z^v, n^v)\mathrm{C}_t^{-1}\mathbf{P}_{t_j}^{\mathrm{T}}, \tag{8}$$

where $\mathbf{p}_{u_k}^{\mathrm{T}}$ and $\mathbf{p}_{t_k}^{\mathrm{T}}$ are the polynomial basis vectors corresponding to displacement and traction. In [27], C_t is called as the moment matrix. $\mathbf{P}_{u_j}^{\mathrm{T}}$ and $\mathbf{P}_{t_j}^{\mathrm{T}}$ are matrices defined as follows:

$$\mathbf{P}_{u_j}^{\mathrm{T}} = \left[\mathbf{p}_{u_k}(x_1, y_1, z_1) \quad \mathbf{p}_{u_k}(x_2, y_2, z_2) \quad \cdots \quad \mathbf{p}_{u_k}(x_M, y_M, z_M)\right], \tag{9}$$

$$\mathbf{P}_{t_j}^{\mathrm{T}} = \left[\mathbf{p}_{t_k}(x_1, y_1, z_1, n_1) \quad \mathbf{p}_{t_k}(x_2, y_2, z_2, n_2) \quad \cdots \quad \mathbf{p}_{t_k}(x_M, y_M, z_M, n_M)\right], \tag{10}$$

The discretization form of Eqs (3), (4) can be represented as follows:

$$u_k(Q_n^v) = \sum_{j=1}^{3}\sum_{i=1}^{M} \psi_i^{u_k u_j}(x^v, y^v, z^v)u_j(Q_i^s) + \sum_{j=1}^{3}\sum_{i=1}^{M} \psi_i^{u_k t_j}(x^v, y^v, z^v)t_j(Q_i^s), \tag{11}$$

$$t_k(Q_n^v) = \sum_{j=1}^{3}\sum_{i=1}^{M} \psi_i^{t_k u_j}(x^v, y^v, z^v, n^v)u_j(Q_i^s) + \sum_{j=1}^{3} \times \sum_{i=1}^{M} \psi_i^{t_k t_j}(x^v, y^v, z^v, n^v)t_j(Q_i^s), \tag{12}$$

where k=1,2,3. The number of source points $Q^s$is represent as M. $u_j(Q_i^s)$, $t_j(Q_i^s)$ are displacement and traction component of source point Q_i^salong j^{th} direction. $\psi_i^{u_k u_j}(x^v,y^v,z^v)$, $\psi_i^{u_k t_j}(x^v,y^v,z^v)$, $\psi_i^{t_k u_j}(x^v,y^v,z^v,n^v)$ and $\psi_i^{t_k t_j}(x^v, y^v, z^v, n^v)$ are shape functions corresponding to Q_i^s.

For hang points in type-1, the compatibility condition is used to construct the second-layer interpolation for$u_k(Q_n^v)$:

$$u_k(Q_n^v) = u_k(Q^s), (k = 1, 2, 3), \tag{13}$$

but for $t_k(Q_n^v)$, HMLS approximation (see Eqs (3), (4)) are employed.

For hang points in type-2, the second-layer interpolation for $u_k(Q_n^v)$ and $t_k(Q_n^v)$are all built by HMLS.

4. Discretization and reassembly of the BIE

4.1. Boundary integral equation

Considering a 3D elasticity body Ω with the boundary Γ(see Fig. 6), the boundary integral equation (BIE) for the 3D elasticity problem in BEM [30] can be written as:

$$c_{ij}(P)u_j(P) = \int_\Gamma U_{ij}(P, Q)t_j(Q)d\Gamma(Q) - \int_\Gamma T_{ij}(P, Q)u_j(Q)d\Gamma(Q), \qquad P, Q \in \Gamma, \tag{14}$$

where u_j and t_j (i, j=1, 2, 3) represent the displacement component and traction component of field point Q, respectively. The coefficient $c_{ij}(P) = 1/2\delta_{ij}$ if Γ is smooth at source node P. For 3D elasticity problems, the fundamental solution $U_{ij}(P,Q)$ and $T_{ij}(P,Q)$ are defined as:

$$U_{ij}(P, Q) = \frac{1}{16\pi G(1-\nu)r}\left[(3-4\nu)\delta_{ij} + r_{,i}r_{,j}\right], \tag{15}$$

$$T_{ij}(P, Q) = -\frac{1}{8\pi(1-\nu)r^2}\left\{\frac{\partial r}{\partial n}\left[(1-2\nu)\delta_{ij} + 3r_{,i}r_{,j}\right] - (1-2\nu)\left(r_{,i}n_j - r_{,j}n_i\right)\right\}, \tag{16}$$

where P and Q are the source point and field point; r is the distance between P and Q; while, n_i and n_j are the components of the outward

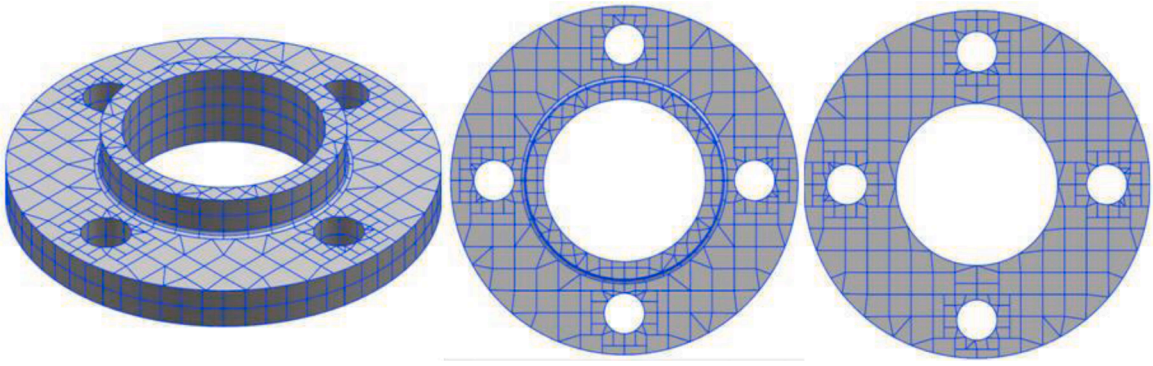


Fig. 3. Binary-tree mesh for flange

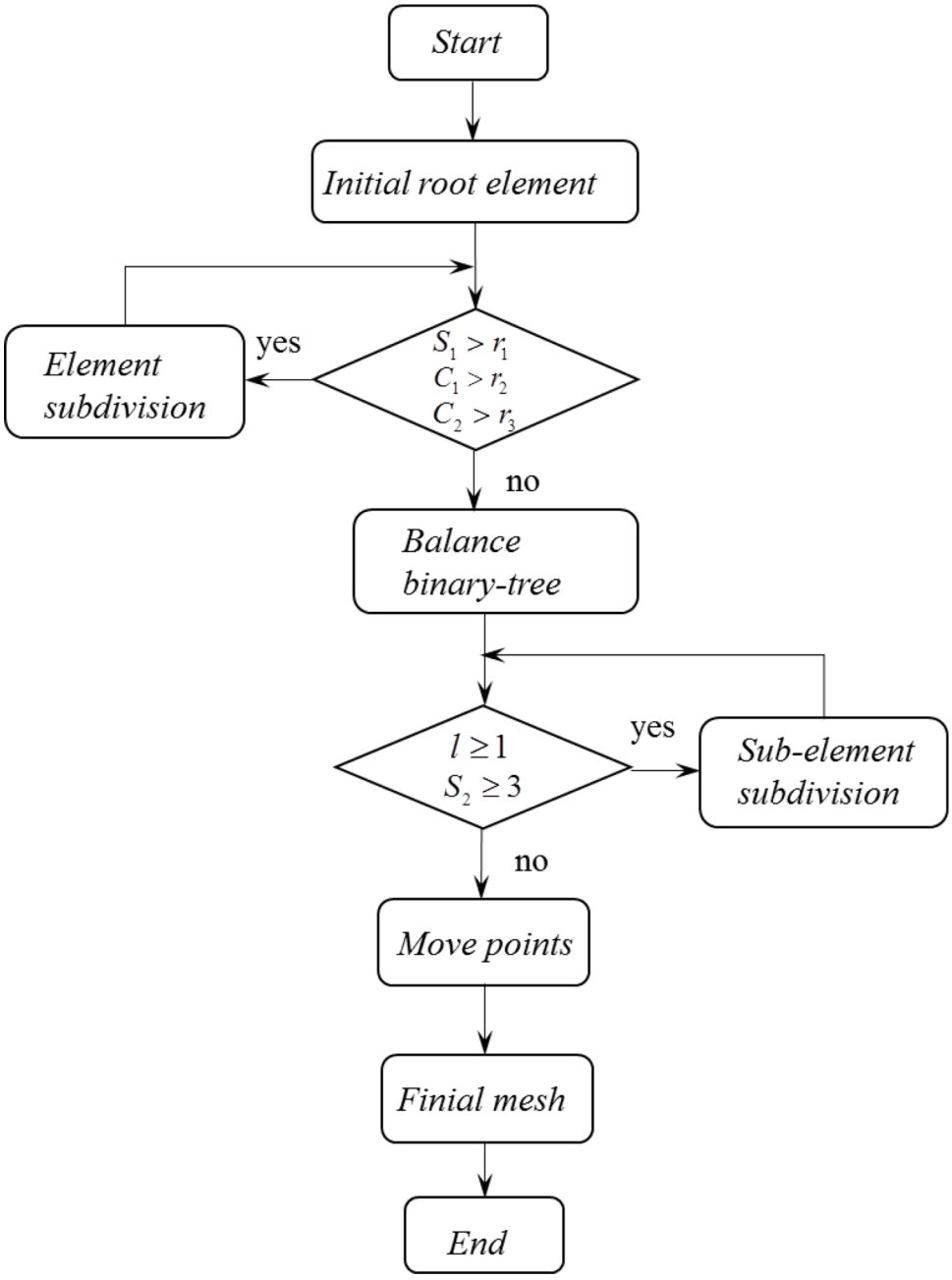


Fig. 4. Flow chart of binary-tree mesh generation

normal *n* at *Q; v* and *G* are the material parameters (the Poisson's ratio and the shear modulus).

4.2. Discretization form of the BIE

Hereinafter, the symbols *NE, NS, NV*, and *NF* (*NF*= *NS*+ *NV*) represent the number of elements, source points, virtual points, and total field points, respectively. In this research, the BIE is discretized by the dual interpolation elements and the traditional continuous elements (see Fig. 5). The discretization form of Eq. (14) concerning collocation point P_k (k=1, 2, …, *NS*) can be written as:

$$\sum_{e=1}^{NE}\left[\sum_{m=1}^{ns} h_{ij}^{ss}(P_k)u_j\left(Q_{e(m)}^{s}\right)+\sum_{n=1}^{nv} h_{ij}^{sv}(P_k)u_j\left(Q_{e(n)}^{v}\right)\right] = \sum_{e=1}^{NE}\left[\sum_{m=1}^{ns} g_{ij}^{ss}(P_k)t_j\left(Q_{e(m)}^{s}\right)+\sum_{n=1}^{nv} g_{ij}^{sv}(P_k)t_j\left(Q_{e(n)}^{v}\right)\right] \quad (i,j=1,2,3), \tag{17}$$

the symbols *ns* and *nv* represent the number of source points and virtual points in one interpolation element.

$$h_{ij}^{ss}(P_k)=\int_{\Gamma_e} T_{ij}(P_k,Q)N_{e(m)}^{s}(Q)d\Gamma(Q)+\frac{1}{2}\delta_{ij},$$

$$h_{ij}^{sv}(P_k)=\int_{\Gamma_e} T_{ij}(P_k,Q)N_{e(n)}^{v}(Q)d\Gamma(Q),$$

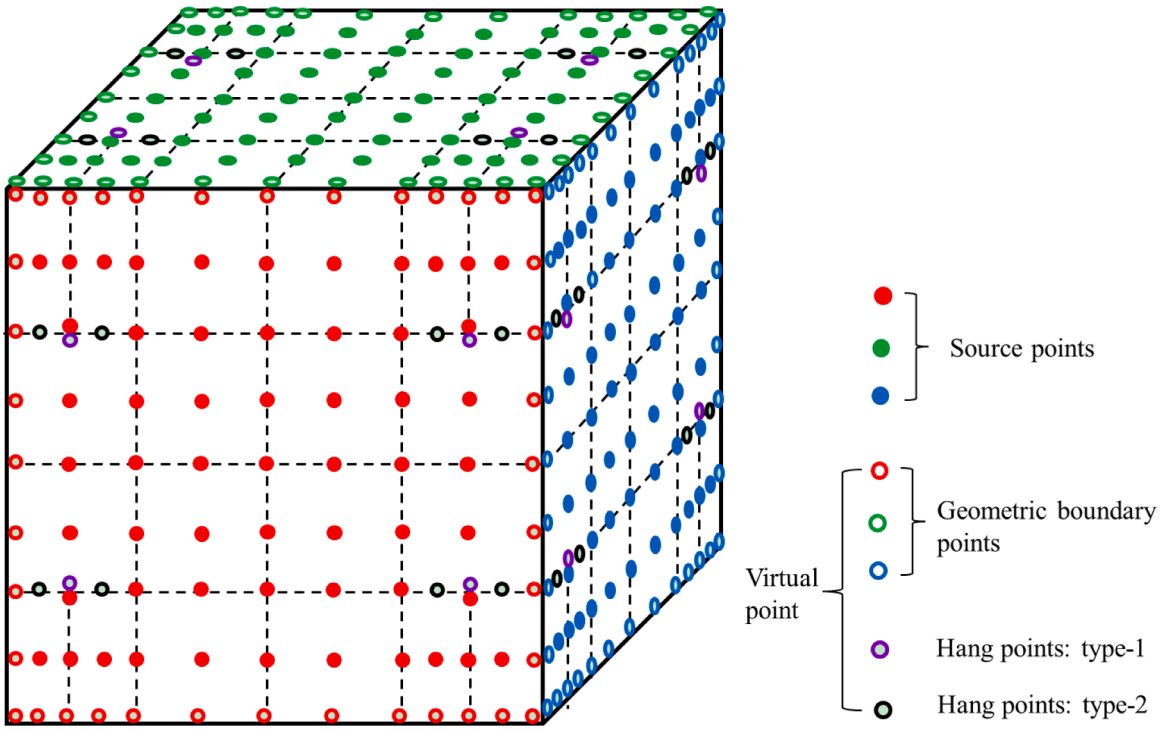


Fig. 5. Distribution of interpolation elements

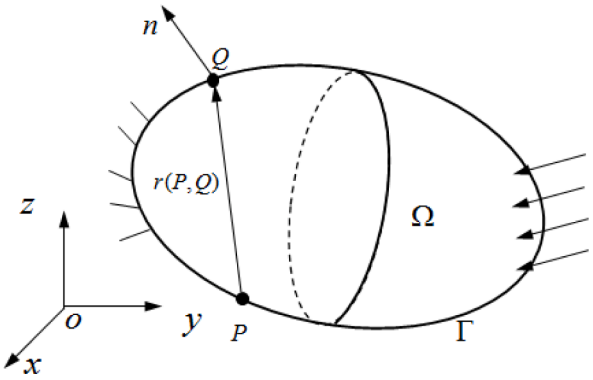


Fig. 6. A 3-D finite domain Ω with boundary Γ.

$$g_{ij}^{ss}(P_k)=\int_{\Gamma_e}U_{ij}(P_k,Q)N_{e(m)}^{s}(Q)d\Gamma(Q),$$

$$g_{ij}^{sv}(P_k)=\int_{\Gamma_e}U_{ij}(P_k,Q)N_{e(n)}^{v}(Q)d\Gamma(Q), \tag{18}$$

and

$$\delta_{ij}=\begin{cases}1, \text{for} i=j\\ 0, \quad \text{for} i\neq j\end{cases}, \tag{19}$$

the symbol Γ_e are e^{th} boundary element of domain Ω. $N_{e(m)}^{s}$ and $N_{e(n)}^{v}$denote the shape function of m^{th} source point and n^{th} virtual point on Γ_e. $u_j(Q_{e(m)}^{s})$ and $u_j(Q_{e(n)}^{v})$ are the displacement components. $t_j(Q_{e(m)}^{s})$ and $t_j(Q_{e(n)}^{v})$ denote the traction components.

The matrix form of Eq. (17) is:

$$\begin{bmatrix}\mathbf{H}_{11}&\mathbf{H}_{12}&\mathbf{H}_{13}\\ \mathbf{H}_{21}&\mathbf{H}_{22}&\mathbf{H}_{23}\\ \mathbf{H}_{31}&\mathbf{H}_{32}&\mathbf{H}_{33}\end{bmatrix}\begin{Bmatrix}\mathbf{u}_1\\ \mathbf{u}_2\\ \mathbf{u}_3\end{Bmatrix}=\begin{bmatrix}\mathbf{G}_{11}&\mathbf{G}_{12}&\mathbf{G}_{13}\\ \mathbf{G}_{21}&\mathbf{G}_{22}&\mathbf{G}_{23}\\ \mathbf{G}_{31}&\mathbf{G}_{32}&\mathbf{G}_{33}\end{bmatrix}\begin{Bmatrix}\mathbf{t}_1\\ \mathbf{t}_2\\ \mathbf{t}_3\end{Bmatrix}, \tag{20}$$

where $\mathbf{H}_{ij}(3NS\times 3NF)$ and $\mathbf{G}_{ij}(3NS\times 3NF)$ (i, j=1, 2, 3) are coefficient matrices, $\mathbf{u}_i(3NF\times 1)$ and $\mathbf{q}_i(3NF\times 1)$ are displacement and traction vectors for all field points.

Since the virtual points are not regarded as collocation points in the BIE, the matrices in Eq. (20) are not squares any more. To get a solvable system of linear algebraic equations, the second-layer interpolation in Section 3.2 is employed.

Through separating the source and virtual points from Eq. (20), the matrix form can be rewritten as:

$$\begin{aligned}&\begin{bmatrix}\mathbf{H}_{11}^{ss}&\mathbf{H}_{12}^{ss}&\mathbf{H}_{13}^{ss}\\ \mathbf{H}_{21}^{ss}&\mathbf{H}_{22}^{ss}&\mathbf{H}_{23}^{ss}\\ \mathbf{H}_{31}^{ss}&\mathbf{H}_{32}^{ss}&\mathbf{H}_{33}^{ss}\end{bmatrix}\begin{Bmatrix}\mathbf{u}_1^{s}\\ \mathbf{u}_2^{s}\\ \mathbf{u}_3^{s}\end{Bmatrix}+\begin{bmatrix}\mathbf{H}_{11}^{sv}&\mathbf{H}_{12}^{sv}&\mathbf{H}_{13}^{sv}\\ \mathbf{H}_{21}^{sv}&\mathbf{H}_{22}^{sv}&\mathbf{H}_{23}^{sv}\\ \mathbf{H}_{31}^{sv}&\mathbf{H}_{32}^{sv}&\mathbf{H}_{33}^{sv}\end{bmatrix}\begin{Bmatrix}\mathbf{u}_1^{v}\\ \mathbf{u}_2^{v}\\ \mathbf{u}_3^{v}\end{Bmatrix}\\ &=\begin{bmatrix}\mathbf{G}_{11}^{ss}&\mathbf{G}_{12}^{ss}&\mathbf{G}_{13}^{ss}\\ \mathbf{G}_{21}^{ss}&\mathbf{G}_{22}^{ss}&\mathbf{G}_{23}^{ss}\\ \mathbf{G}_{31}^{ss}&\mathbf{G}_{32}^{ss}&\mathbf{G}_{33}^{ss}\end{bmatrix}\begin{Bmatrix}\mathbf{t}_1^{s}\\ \mathbf{t}_2^{s}\\ \mathbf{t}_3^{s}\end{Bmatrix}+\begin{bmatrix}\mathbf{G}_{11}^{sv}&\mathbf{G}_{12}^{sv}&\mathbf{G}_{13}^{sv}\\ \mathbf{G}_{21}^{sv}&\mathbf{G}_{22}^{sv}&\mathbf{G}_{23}^{sv}\\ \mathbf{G}_{31}^{sv}&\mathbf{G}_{32}^{sv}&\mathbf{G}_{33}^{sv}\end{bmatrix}\begin{Bmatrix}\mathbf{t}_1^{v}\\ \mathbf{t}_2^{v}\\ \mathbf{t}_3^{v}\end{Bmatrix}\end{aligned}, \tag{21}$$

where $\mathbf{u}_j^{s}$ and $\mathbf{u}_j^{v}$ are displacement vectors corresponding to all source and virtual points. $\mathbf{t}_j^{s}$and $\mathbf{t}_j^{v}$ are traction vectors. The symbols$\mathbf{H}_{ij}^{ss}$, $\mathbf{G}_{ij}^{ss}$, $\mathbf{H}_{ij}^{sv}$ and $\mathbf{G}_{ij}^{sv}$ represent coefficient matrices corresponding to $\mathbf{u}_j^{s}$, $\mathbf{t}_j^{s}$, $\mathbf{u}_j^{v}$ and $\mathbf{t}_j^{v}$, respectively. According to the boundary conditions, the vectors $\mathbf{u}_j^{v}$ and $\mathbf{t}_j^{v}$can be decomposed into two parts:

$$\mathbf{u}_j^{v}=\overset{\frown}{\mathbf{u}}_j^{v}\cup\bar{\mathbf{u}}_j^{v}, \tag{22}$$

$$\mathbf{t}_j^{v}=\overset{\frown}{\mathbf{t}}_j^{v}\cup\bar{\mathbf{t}}_j^{v}, \tag{23}$$

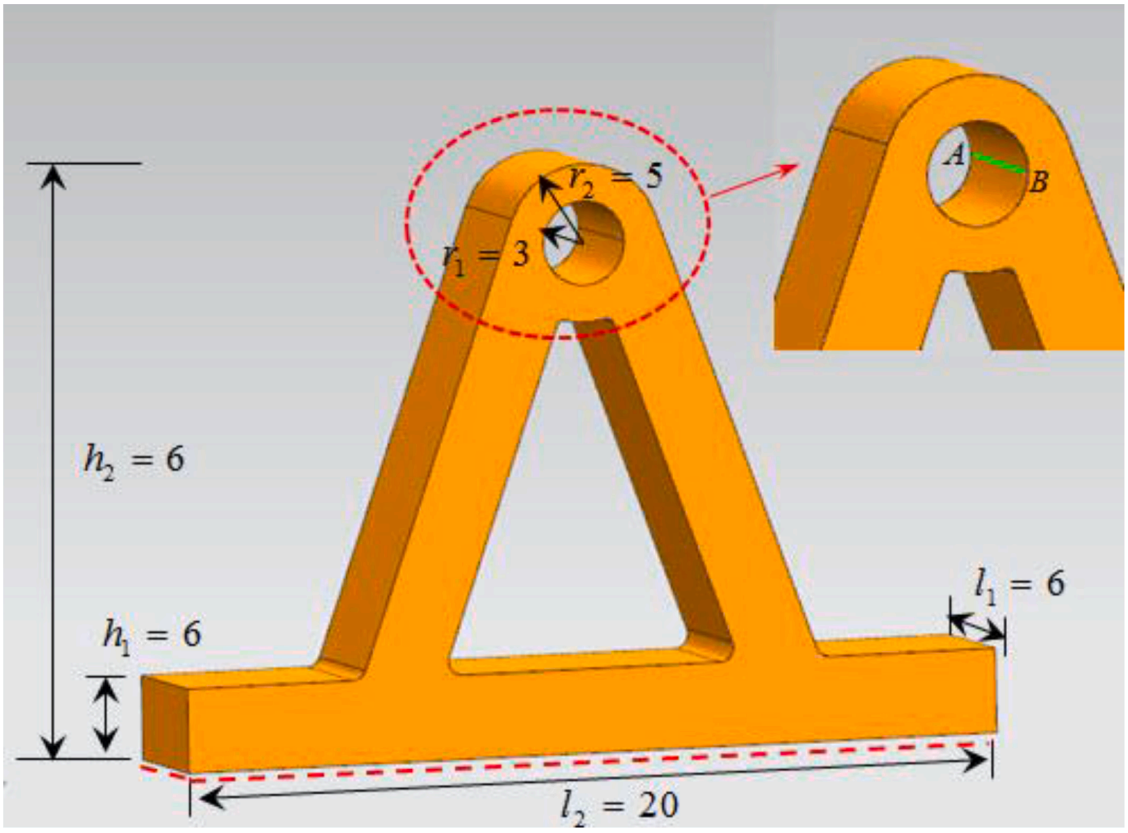


Fig. 7. Geometric model of bearing support

where $\widehat{\mathbf{u}}_j^v, \widehat{\mathbf{t}}_j^v$ and $\overline{\mathbf{u}}_j^v, \overline{\mathbf{t}}_j^v$ represent the unknown and know displacement and traction components, respectively.

According to Eqs (22), (23), Eq. (21) can be written as:

$$\begin{bmatrix} \mathbf{H}_{11}^{ss} & \mathbf{H}_{12}^{ss} & \mathbf{H}_{13}^{ss} \\ \mathbf{H}_{21}^{ss} & \mathbf{H}_{22}^{ss} & \mathbf{H}_{23}^{ss} \\ \mathbf{H}_{31}^{ss} & \mathbf{H}_{32}^{ss} & \mathbf{H}_{33}^{ss} \end{bmatrix} \begin{Bmatrix} \mathbf{u}_1^s \\ \mathbf{u}_2^s \\ \mathbf{u}_3^s \end{Bmatrix} + \begin{bmatrix} \mathbf{H}_{11}^{sv} & \mathbf{H}_{12}^{sv} & \mathbf{H}_{13}^{sv} \\ \mathbf{H}_{21}^{sv} & \mathbf{H}_{22}^{sv} & \mathbf{H}_{23}^{sv} \\ \mathbf{H}_{31}^{sv} & \mathbf{H}_{32}^{sv} & \mathbf{H}_{33}^{sv} \end{bmatrix} \begin{Bmatrix} \widehat{\mathbf{u}}_1^v \\ \widehat{\mathbf{u}}_2^v \\ \widehat{\mathbf{u}}_3^v \end{Bmatrix} + \begin{bmatrix} \mathbf{H}_{11}^{sv} & \mathbf{H}_{12}^{sv} & \mathbf{H}_{13}^{sv} \\ \mathbf{H}_{21}^{sv} & \mathbf{H}_{22}^{sv} & \mathbf{H}_{23}^{sv} \\ \mathbf{H}_{31}^{sv} & \mathbf{H}_{32}^{sv} & \mathbf{H}_{33}^{sv} \end{bmatrix} \begin{Bmatrix} \overline{\mathbf{u}}_1^v \\ \overline{\mathbf{u}}_2^v \\ \overline{\mathbf{u}}_3^v \end{Bmatrix}$$
$$= \begin{bmatrix} \mathbf{G}_{11}^{ss} & \mathbf{G}_{12}^{ss} & \mathbf{G}_{13}^{ss} \\ \mathbf{G}_{21}^{ss} & \mathbf{G}_{22}^{ss} & \mathbf{G}_{23}^{ss} \\ \mathbf{G}_{31}^{ss} & \mathbf{G}_{32}^{ss} & \mathbf{G}_{33}^{ss} \end{bmatrix} \begin{Bmatrix} \mathbf{t}_1^s \\ \mathbf{t}_2^s \\ \mathbf{t}_3^s \end{Bmatrix} + \begin{bmatrix} \mathbf{G}_{11}^{sv} & \mathbf{G}_{12}^{sv} & \mathbf{G}_{13}^{sv} \\ \mathbf{G}_{21}^{sv} & \mathbf{G}_{22}^{sv} & \mathbf{G}_{23}^{sv} \\ \mathbf{G}_{31}^{sv} & \mathbf{G}_{32}^{sv} & \mathbf{G}_{33}^{sv} \end{bmatrix} \begin{Bmatrix} \widehat{\mathbf{t}}_1^v \\ \widehat{\mathbf{t}}_2^v \\ \widehat{\mathbf{t}}_3^v \end{Bmatrix} + \begin{bmatrix} \mathbf{G}_{11}^{sv} & \mathbf{G}_{12}^{sv} & \mathbf{G}_{13}^{sv} \\ \mathbf{G}_{21}^{sv} & \mathbf{G}_{22}^{sv} & \mathbf{G}_{23}^{sv} \\ \mathbf{G}_{31}^{sv} & \mathbf{G}_{32}^{sv} & \mathbf{G}_{33}^{sv} \end{bmatrix} \begin{Bmatrix} \overline{\mathbf{t}}_1^v \\ \overline{\mathbf{t}}_2^v \\ \overline{\mathbf{t}}_3^v \end{Bmatrix}, \tag{24}$$

4.3. Eliminating the degrees of freedom for all virtual points

Through Eqs (11)-(13), the second-layer interpolation for all virtual points in Eq. (24) can be written as matrix forms:

$$\mathbf{u}_i^v = \Phi_{u_i u_j}^{vs} \mathbf{u}_j^s + \Phi_{u_i t_j}^{vs} \mathbf{t}_j^s, \quad i,j = 1 \sim 3, \tag{25}$$

$$\mathbf{t}_i^v = \Phi_{t_i u_j}^{vs} \mathbf{u}_j^s + \Phi_{t_i t_j}^{vs} \mathbf{t}_j^s, \quad i,j = 1 \sim 3, \tag{26}$$

$$\mathbf{u}_i^v = \mathbf{I}_{ij} \mathbf{u}_j^s, \quad i,j = 1 \sim 3, \tag{27}$$

where $\Phi_{u_i u_j}^{vs}, \Phi_{u_i t_j}^{vs}, \Phi_{t_i u_j}^{vs}$ and $\Phi_{t_i t_j}^{vs}$ are shape matrices formed by HMLS. $\mathbf{I}_{ij}$is the identity matrix.

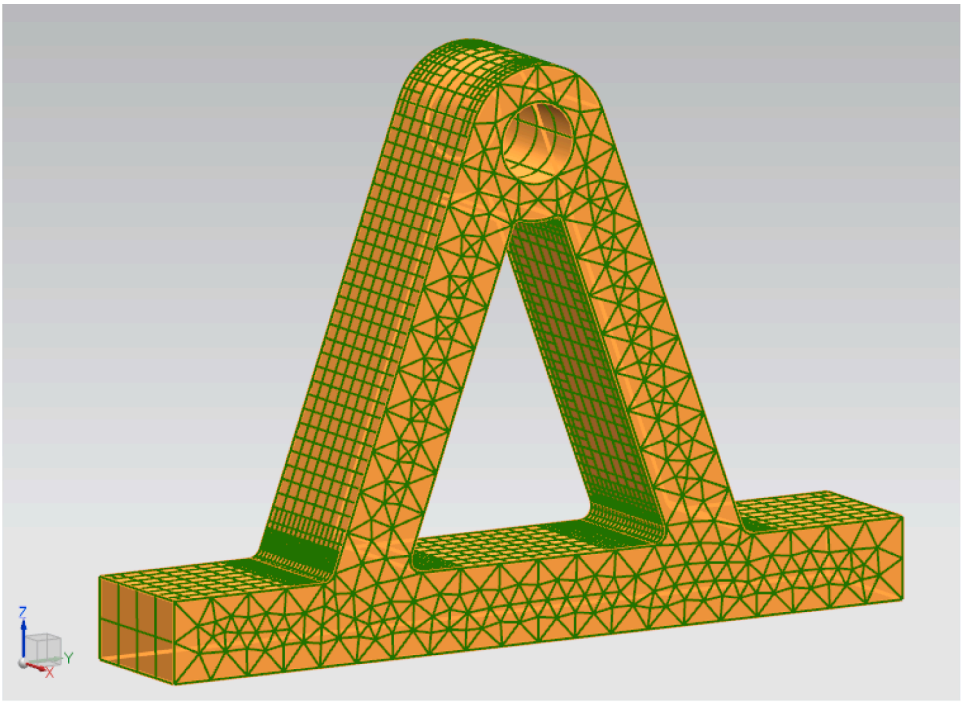
According to Eqs (25)-(27), Eq. (24) can be rewritten as:

$$\begin{bmatrix} \widetilde{\mathbf{H}}_{11}^{ss} & \widetilde{\mathbf{H}}_{12}^{ss} & \widetilde{\mathbf{H}}_{13}^{ss} \\ \widetilde{\mathbf{H}}_{21}^{ss} & \widetilde{\mathbf{H}}_{22}^{ss} & \widetilde{\mathbf{H}}_{23}^{ss} \\ \widetilde{\mathbf{H}}_{31}^{ss} & \widetilde{\mathbf{H}}_{32}^{ss} & \widetilde{\mathbf{H}}_{33}^{ss} \end{bmatrix} \begin{Bmatrix} \mathbf{u}_1^s \\ \mathbf{u}_2^s \\ \mathbf{u}_3^s \end{Bmatrix} + \begin{bmatrix} \mathbf{H}_{11}^{sv} & \mathbf{H}_{12}^{sv} & \mathbf{H}_{13}^{sv} \\ \mathbf{H}_{21}^{sv} & \mathbf{H}_{22}^{sv} & \mathbf{H}_{23}^{sv} \\ \mathbf{H}_{31}^{sv} & \mathbf{H}_{32}^{sv} & \mathbf{H}_{33}^{sv} \end{bmatrix} \begin{Bmatrix} \overline{\mathbf{u}}_1^v \\ \overline{\mathbf{u}}_2^v \\ \overline{\mathbf{u}}_3^v \end{Bmatrix}$$
$$= \begin{bmatrix} \widetilde{\mathbf{G}}_{11}^{ss} & \widetilde{\mathbf{G}}_{12}^{ss} & \widetilde{\mathbf{G}}_{13}^{ss} \\ \widetilde{\mathbf{G}}_{21}^{ss} & \widetilde{\mathbf{G}}_{22}^{ss} & \widetilde{\mathbf{G}}_{23}^{ss} \\ \widetilde{\mathbf{G}}_{31}^{ss} & \widetilde{\mathbf{G}}_{32}^{ss} & \widetilde{\mathbf{G}}_{33}^{ss} \end{bmatrix} \begin{Bmatrix} \mathbf{t}_1^s \\ \mathbf{t}_2^s \\ \mathbf{t}_3^s \end{Bmatrix} + \begin{bmatrix} \mathbf{G}_{11}^{sv} & \mathbf{G}_{12}^{sv} & \mathbf{G}_{13}^{sv} \\ \mathbf{G}_{21}^{sv} & \mathbf{G}_{22}^{sv} & \mathbf{G}_{23}^{sv} \\ \mathbf{G}_{31}^{sv} & \mathbf{G}_{32}^{sv} & \mathbf{G}_{33}^{sv} \end{bmatrix} \begin{Bmatrix} \overline{\mathbf{t}}_1^v \\ \overline{\mathbf{t}}_2^v \\ \overline{\mathbf{t}}_3^v \end{Bmatrix}, \tag{28}$$

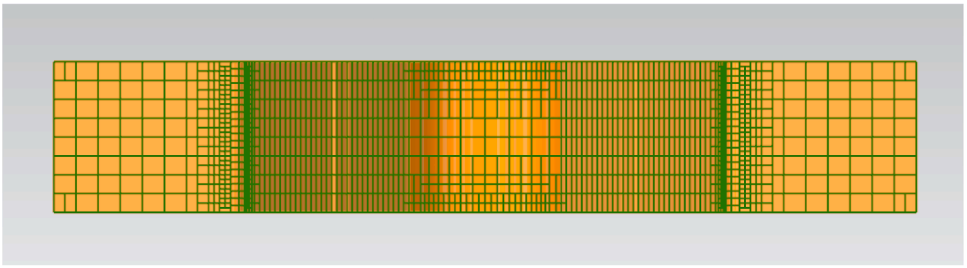
in which

$$\begin{bmatrix} \widetilde{\mathbf{H}}_{11}^{ss} & \widetilde{\mathbf{H}}_{12}^{ss} & \widetilde{\mathbf{H}}_{13}^{ss} \\ \widetilde{\mathbf{H}}_{21}^{ss} & \widetilde{\mathbf{H}}_{22}^{ss} & \widetilde{\mathbf{H}}_{23}^{ss} \\ \widetilde{\mathbf{H}}_{31}^{ss} & \widetilde{\mathbf{H}}_{32}^{ss} & \widetilde{\mathbf{H}}_{33}^{ss} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{11}^{ss} & \mathbf{H}_{12}^{ss} & \mathbf{H}_{13}^{ss} \\ \mathbf{H}_{21}^{ss} & \mathbf{H}_{22}^{ss} & \mathbf{H}_{23}^{ss} \\ \mathbf{H}_{31}^{ss} & \mathbf{H}_{32}^{ss} & \mathbf{H}_{33}^{ss} \end{bmatrix} + \begin{bmatrix} \mathbf{H}_{11}^{sv} & \mathbf{H}_{12}^{sv} & \mathbf{H}_{13}^{sv} \\ \mathbf{H}_{21}^{ss} & \mathbf{H}_{22}^{ss} & \mathbf{H}_{23}^{ss} \\ \mathbf{H}_{31}^{sv} & \mathbf{H}_{32}^{sv} & \mathbf{H}_{33}^{sv} \end{bmatrix} \begin{bmatrix} \overline{\Phi}_{u_1 u_1}^{vs} & \overline{\Phi}_{u_1 u_2}^{vs} & \overline{\Phi}_{u_1 u_3}^{vs} \\ \overline{\Phi}_{u_2 u_1}^{vs} & \overline{\Phi}_{u_2 u_2}^{vs} & \overline{\Phi}_{u_2 u_3}^{vs} \\ \overline{\Phi}_{u_3 u_1}^{vs} & \overline{\Phi}_{u_3 u_2}^{vs} & \overline{\Phi}_{u_3 u_3}^{vs} \end{bmatrix}$$
$$- \begin{bmatrix} \mathbf{G}_{11}^{sv} & \mathbf{G}_{12}^{sv} & \mathbf{G}_{13}^{sv} \\ \mathbf{G}_{21}^{sv} & \mathbf{G}_{22}^{sv} & \mathbf{G}_{23}^{sv} \\ \mathbf{G}_{31}^{sv} & \mathbf{G}_{32}^{sv} & \mathbf{G}_{33}^{sv} \end{bmatrix} \begin{bmatrix} \overline{\Phi}_{t_1 u_1}^{vs} & \overline{\Phi}_{t_1 u_2}^{vs} & \overline{\Phi}_{t_1 u_3}^{vs} \\ \overline{\Phi}_{t_2 u_1}^{vs} & \overline{\Phi}_{t_2 u_2}^{vs} & \overline{\Phi}_{t_2 u_3}^{vs} \\ \overline{\Phi}_{t_3 u_1}^{vs} & \overline{\Phi}_{t_3 u_2}^{vs} & \overline{\Phi}_{t_3 u_3}^{vs} \end{bmatrix}, \tag{29}$$

the entities of matrices $\overline{\Phi}_{u_i u_j}^{vs}, \overline{\Phi}_{t_i u_j}^{vs}$ (i, j=1, 2, 3) come from Eqs (25)-(27).



(a)



(b)

Fig. 8. Discontinuous meshes for bearing support: (a) oblique view, (b) vertical view.

Table 2
Relative errors and CPU time of the results along line AB with different methods.

DiBFM-discontinuous meshes				DiBFM-continuous meshes			
NE	NS	*Err_Mises*	*time*(*s*)	NE	NS	*Err_Mises*	*time*(*s*)
318	318	6.21E-03	13	346	346	6.20E-03	17
546	546	4.10E-03	30	585	585	4.01E-03	40
984	984	3.24E-03	100	1,076	1,076	3.20E-03	120
2,743	2,743	2.42E-03	710	2,852	2,852	2.40E-03	790

$$\begin{bmatrix} \widetilde{\mathbf{G}}_{11}^{ss} & \widetilde{\mathbf{G}}_{12}^{ss} & \widetilde{\mathbf{G}}_{13}^{ss} \\ \widetilde{\mathbf{G}}_{21}^{ss} & \widetilde{\mathbf{G}}_{22}^{ss} & \widetilde{\mathbf{G}}_{23}^{ss} \\ \widetilde{\mathbf{G}}_{31}^{ss} & \widetilde{\mathbf{G}}_{32}^{ss} & \widetilde{\mathbf{G}}_{33}^{ss} \end{bmatrix} = \begin{bmatrix} \mathbf{G}_{11}^{ss} & \mathbf{G}_{12}^{ss} & \mathbf{G}_{13}^{ss} \\ \mathbf{G}_{21}^{ss} & \mathbf{G}_{22}^{ss} & \mathbf{G}_{23}^{ss} \\ \mathbf{G}_{31}^{ss} & \mathbf{G}_{32}^{ss} & \mathbf{G}_{33}^{ss} \end{bmatrix} + \begin{bmatrix} \mathbf{G}_{11}^{sv} & \mathbf{G}_{12}^{sv} & \mathbf{G}_{13}^{sv} \\ \mathbf{G}_{21}^{sv} & \mathbf{G}_{22}^{sv} & \mathbf{G}_{23}^{sv} \\ \mathbf{G}_{31}^{sv} & \mathbf{G}_{32}^{sv} & \mathbf{G}_{33}^{sv} \end{bmatrix} \begin{bmatrix} \Phi_{t_1t_1}^{vs} & \Phi_{t_1t_2}^{vs} & \Phi_{t_1t_3}^{vs} \\ \Phi_{t_2t_1}^{vs} & \Phi_{t_2t_2}^{vs} & \Phi_{t_2t_3}^{vs} \\ \Phi_{t_3t_1}^{vs} & \Phi_{t_3t_2}^{vs} & \Phi_{t_3t_3}^{vs} \end{bmatrix} - \begin{bmatrix} \mathbf{H}_{11}^{sv} & \mathbf{H}_{12}^{sv} & \mathbf{H}_{13}^{sv} \\ \mathbf{H}_{21}^{sv} & \mathbf{H}_{22}^{sv} & \mathbf{H}_{23}^{sv} \\ \mathbf{H}_{31}^{sv} & \mathbf{H}_{32}^{sv} & \mathbf{H}_{33}^{sv} \end{bmatrix} \begin{bmatrix} \Phi_{u_1t_1}^{vs} & \Phi_{u_1t_2}^{vs} & \Phi_{u_1t_3}^{vs} \\ \Phi_{u_2t_1}^{vs} & \Phi_{u_2t_2}^{vs} & \Phi_{u_2t_3}^{vs} \\ \Phi_{u_3t_1}^{vs} & \Phi_{u_3t_2}^{vs} & \Phi_{u_3t_3}^{vs} \end{bmatrix}, \tag{30}$$

Applying the boundary conditions to source points in Eq. (28), and switching the known quantities to right-hand side, the unknown to left-hand side, a standard system of linear equations is formed as follows:

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{31}$$

in which

$$\mathbf{A} = \begin{bmatrix} \widehat{\mathbf{H}}_{11}^{ss} & \widehat{\mathbf{H}}_{12}^{ss} & \widehat{\mathbf{H}}_{13}^{ss} & -\widehat{\mathbf{G}}_{11}^{ss} & -\widehat{\mathbf{G}}_{12}^{ss} & -\widehat{\mathbf{G}}_{13}^{ss} \\ \widehat{\mathbf{H}}_{21}^{ss} & \widehat{\mathbf{H}}_{22}^{ss} & \widehat{\mathbf{H}}_{23}^{ss} & -\widehat{\mathbf{G}}_{21}^{ss} & -\widehat{\mathbf{G}}_{22}^{ss} & -\widehat{\mathbf{G}}_{23}^{ss} \\ \widehat{\mathbf{H}}_{31}^{ss} & \widehat{\mathbf{H}}_{32}^{ss} & \widehat{\mathbf{H}}_{33}^{ss} & -\widehat{\mathbf{G}}_{31}^{ss} & -\widehat{\mathbf{G}}_{32}^{ss} & -\widehat{\mathbf{G}}_{33}^{ss} \end{bmatrix},$$

$$\mathbf{x} = \left\{ \widehat{\mathbf{u}}_1^s \quad \widehat{\mathbf{u}}_2^s \quad \widehat{\mathbf{u}}_3^s \quad \widehat{\mathbf{t}}_1^s \quad \widehat{\mathbf{t}}_2^s \quad \widehat{\mathbf{t}}_3^s \right\}^{\mathrm{T}},$$

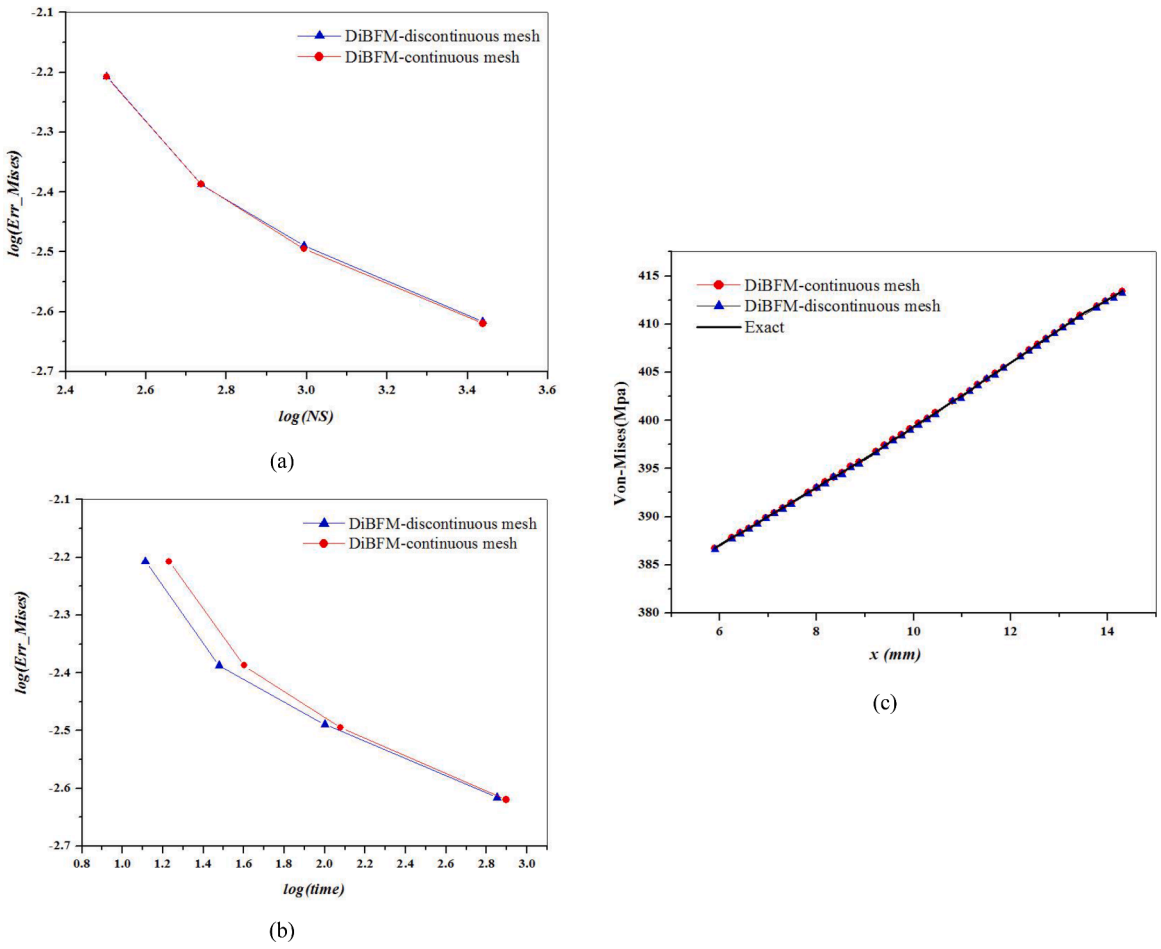


Fig. 9. Numerical comparison: (a) relative errors for Von-mises, (b) computational efficiency and (c) accuracy for Von-mises along line AB.

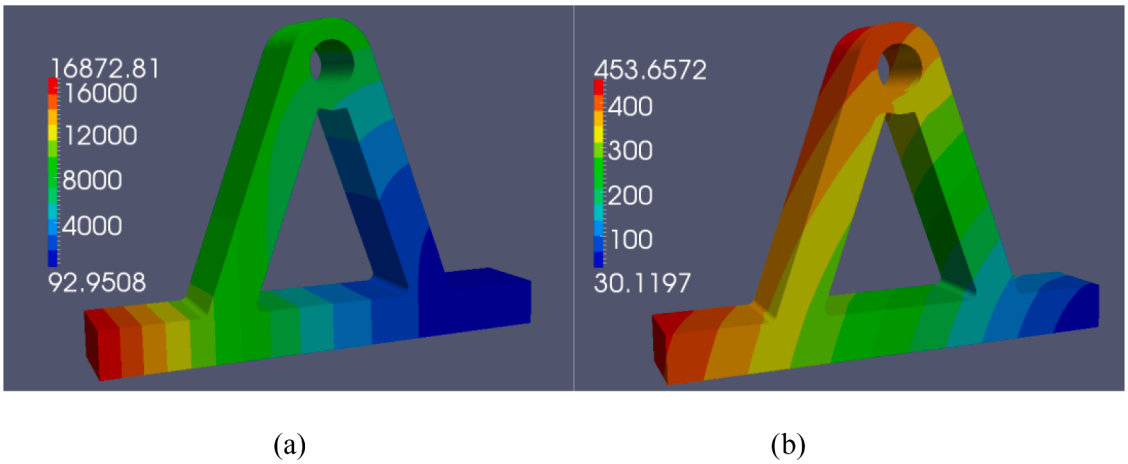


Fig. 10. Contour plots of DiBFM by discontinuous meshes with 4,216 elements, 4,216 source nodes: (a) displacement and (b) Von-Mises stress.

$$\mathbf{b} = -\begin{bmatrix} \overline{\mathbf{H}}_{11}^{ss} & \overline{\mathbf{H}}_{12}^{ss} & \overline{\mathbf{H}}_{13}^{ss} \\ \overline{\mathbf{H}}_{21}^{ss} & \overline{\mathbf{H}}_{22}^{ss} & \overline{\mathbf{H}}_{23}^{ss} \\ \overline{\mathbf{H}}_{31}^{ss} & \overline{\mathbf{H}}_{32}^{ss} & \overline{\mathbf{H}}_{33}^{ss} \end{bmatrix} \begin{Bmatrix} \overline{\mathbf{u}}_1^s \\ \overline{\mathbf{u}}_2^s \\ \overline{\mathbf{u}}_3^s \end{Bmatrix} + \begin{bmatrix} \overline{\mathbf{G}}_{11}^{ss} & \overline{\mathbf{G}}_{12}^{ss} & \overline{\mathbf{G}}_{13}^{ss} \\ \overline{\mathbf{G}}_{21}^{ss} & \overline{\mathbf{G}}_{22}^{ss} & \overline{\mathbf{G}}_{23}^{ss} \\ \overline{\mathbf{G}}_{31}^{ss} & \overline{\mathbf{G}}_{32}^{ss} & \overline{\mathbf{G}}_{33}^{ss} \end{bmatrix} \begin{Bmatrix} \overline{\mathbf{t}}_1^s \\ \overline{\mathbf{t}}_2^s \\ \overline{\mathbf{t}}_3^s \end{Bmatrix} - \begin{bmatrix} \mathbf{H}_{11}^{sv} & \mathbf{H}_{12}^{sv} & \mathbf{H}_{13}^{sv} \\ \mathbf{H}_{21}^{sv} & \mathbf{H}_{22}^{sv} & \mathbf{H}_{23}^{sv} \\ \mathbf{H}_{31}^{sv} & \mathbf{H}_{32}^{sv} & \mathbf{H}_{33}^{sv} \end{bmatrix} \begin{Bmatrix} \overline{\mathbf{u}}_1^v \\ \overline{\mathbf{u}}_2^v \\ \overline{\mathbf{u}}_3^v \end{Bmatrix} + \begin{bmatrix} \mathbf{G}_{11}^{sv} & \mathbf{G}_{12}^{sv} & \mathbf{G}_{13}^{sv} \\ \mathbf{G}_{21}^{sv} & \mathbf{G}_{22}^{sv} & \mathbf{G}_{23}^{sv} \\ \mathbf{G}_{31}^{sv} & \mathbf{G}_{32}^{sv} & \mathbf{G}_{33}^{sv} \end{bmatrix} \begin{Bmatrix} \overline{\mathbf{t}}_1^v \\ \overline{\mathbf{t}}_2^v \\ \overline{\mathbf{t}}_3^v \end{Bmatrix}. \tag{32}$$

In Eq. (32), **A**(3NS × 3NS) represents the final coefficient matrix, and **b** represents the known right-hand-side vector. **x** is the vector concerning the unknown variables of source points.

5. Numerical examples

In this section, the excellent performance of the discontinuous meshes in solving 3D elasticity problems will be illustrated by four numerical examples. The first and second examples demonstrate the

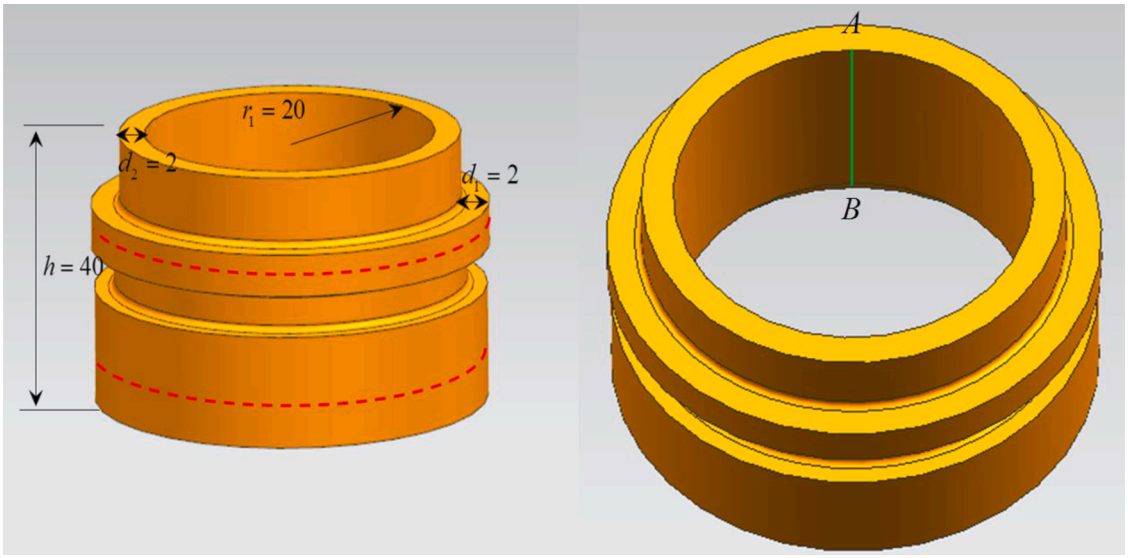


Fig. 11. Geometric model of shaft sleeve

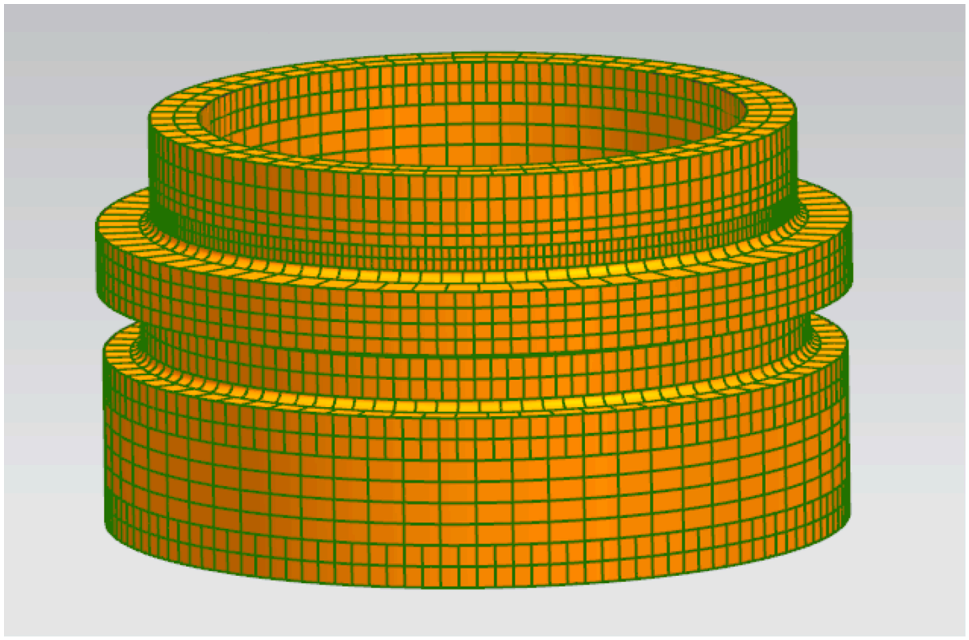


Fig. 12. Discontinuous meshes for shaft sleeve

Table 3
Relative errors and CPU time of the results along line AB with different methods.

DiBFM-discontinuous meshes				DiBFM-continuous meshes			
NE	NS	*Err_Mises*	*time(s)*	NE	NS	*Err_Mises*	*time(s)*
2,472	2,472	7.30E-03	648	2,506	2,506	7.00E-03	671
3,288	3,288	6.60E-03	971	3,364	3,364	6.50E-03	992
4,878	4,878	6.31E-03	2122	4,986	4,986	6.30E-03	2141
10,012	10,012	2.60E-03	5174	10,159	10,159	2.40E-03	5204

accuracy of the presented method in different analytical field problems. The third and fourth examples highlight the ability and the practicability of the proposed method in dealing with complex geometry structures with a real engineering background.

Error estimation and convergence of the presented method are measured by the relative error, defined as:

$$error = \frac{1}{|v^{(e)}|_{\max}} \sqrt{\frac{1}{M} \sum_{i=1}^{M} \left[v_i^{(e)} - v_i^{(n)} \right]^2}, \tag{1}$$

where $|v^{(e)}|_{\max}$ is the maximum value of the exact displacement u or traction t over M sample points, and superscripts e and n denote the exact

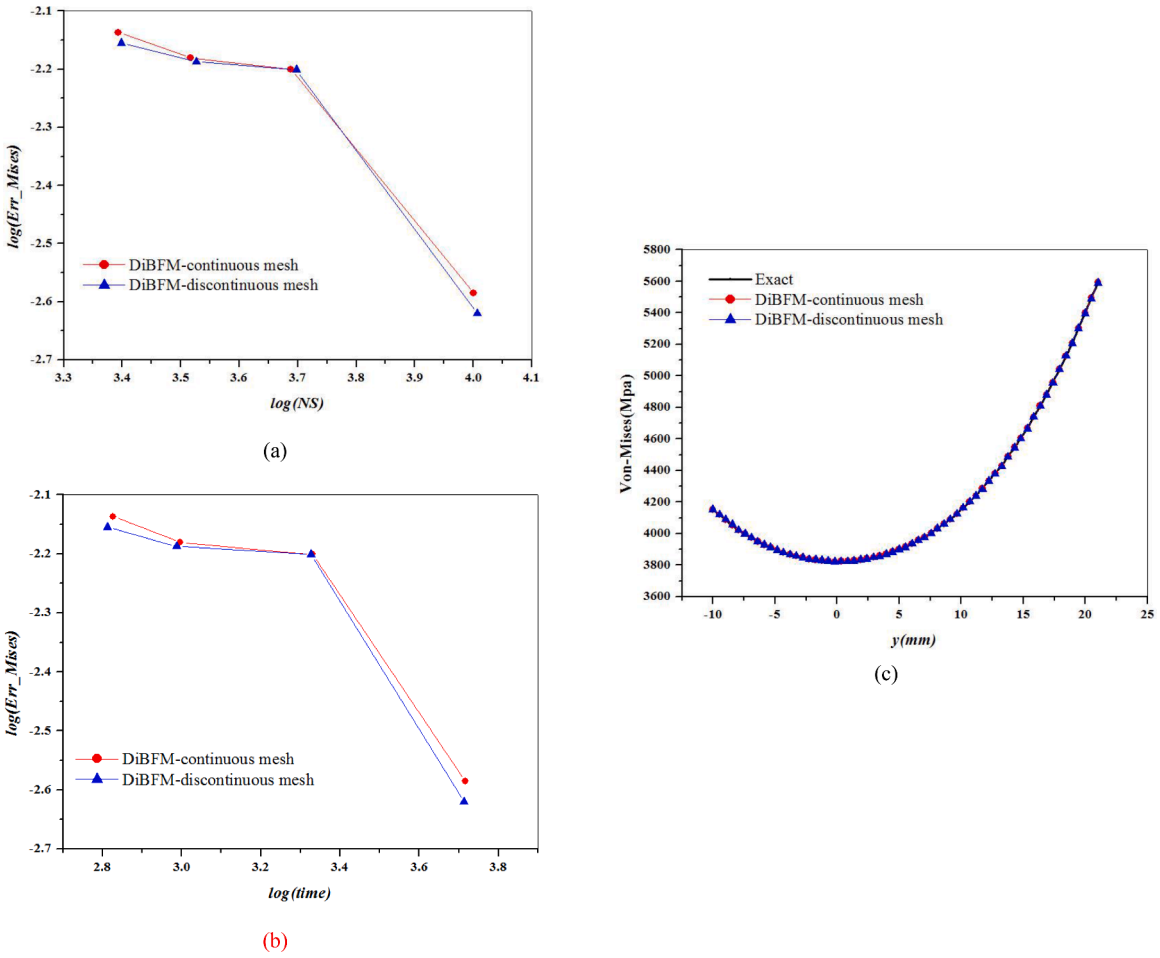


Fig. 13. Numerical comparison: (a) relative errors for Von-mises, (b) computational efficiency and (c) accuracy for Von-mises along line AB.

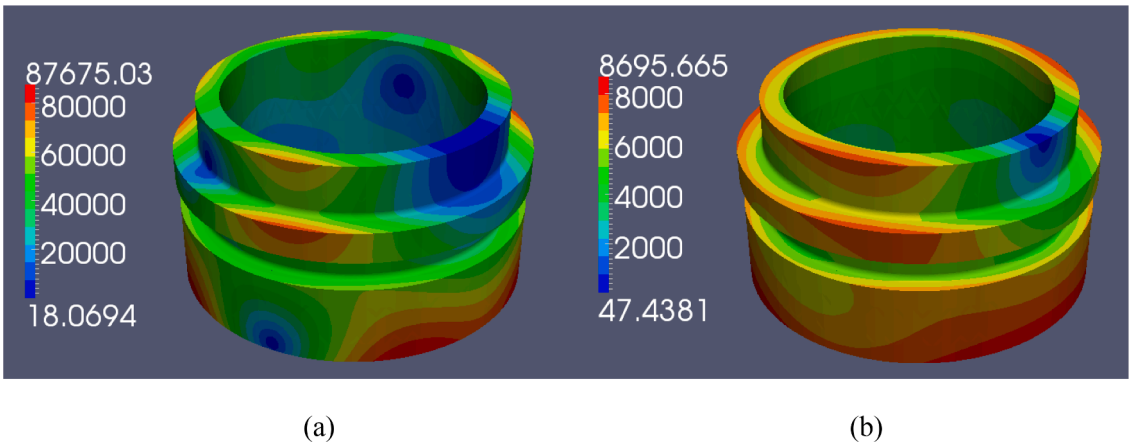


Fig. 14. Contour plots of DiBFM by discontinuous meshes with 4,551 elements, 4,551 source nodes: (a) displacement (b) Von-Mises stress.

and computed solutions, respectively.

In the following examples, unless otherwise mentioned, $\bar{t}$and $\bar{u}$represent the prescribed traction and displacement conditions. "*Err_Mises*", "*NE*", "*NS*" "*NN*" and "*time*" denote the relative errors for Von-Mises stress, the number of elements, source points, total nodes in FEM and CPU time, respectively.

5.1. *Bearing support problem*

This example refers to a quadratic displacement field problem, and the bearing support is presented to verify the precision and the reliability of our algorithm. Geometric dimensioning is shown in Fig. 7. Displacement boundary conditions (see Eq. (33)) are imposed on the bottom boundary faces with the red dotted line, and traction boundary conditions are prescribed at remaining boundary faces. The analytical solution is shown as follow:

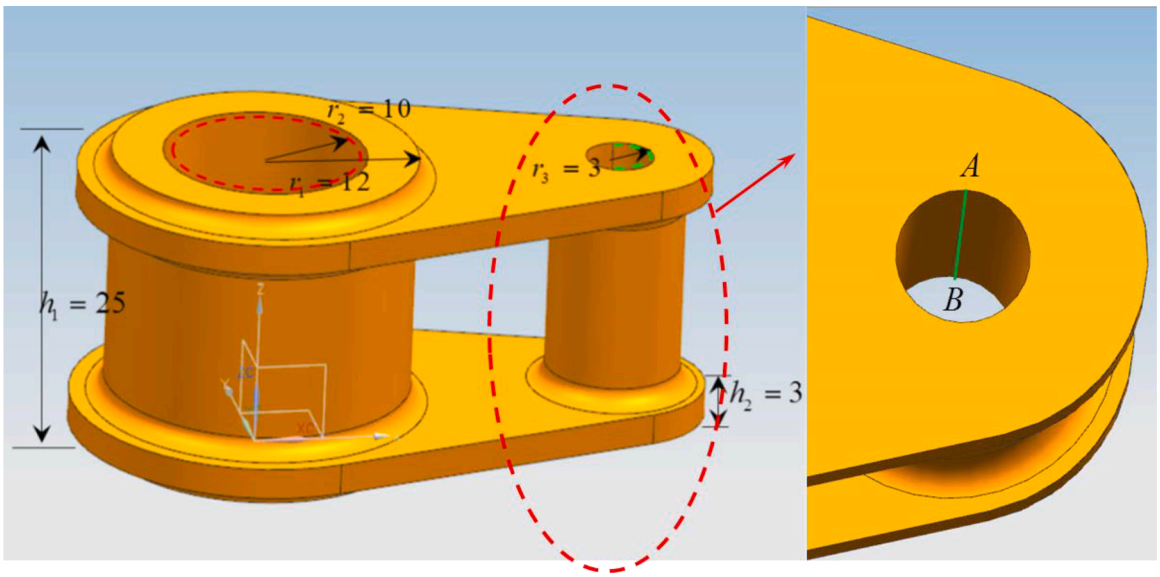


Fig. 15. Geometric model of adapting piece

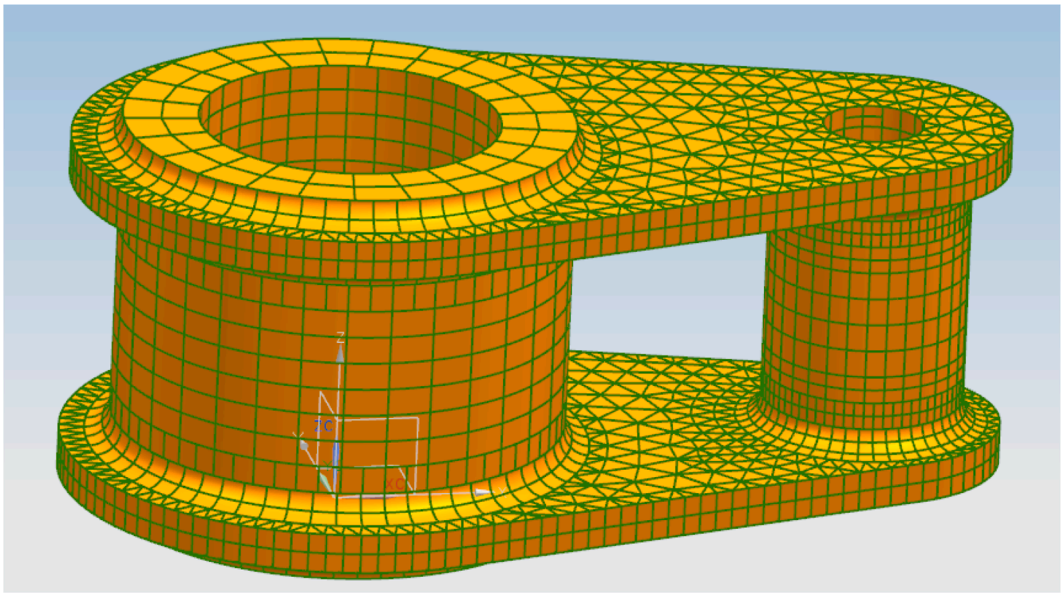


Fig. 16. Discontinuous meshes for adapting piece

Table 4
Numerical results of DiBFM with continuous mesh, DiBFM with discontinuous mesh and FEM for adapting piece.

DiBFM-continuous mesh	NE	4,590	6,684	9,339	12,256
	NS	4,590	6,684	9,339	12,256
	Max_Mises (Mpa)	3.15	3.24	3.44	3.32
DiBFM-discontinuous mesh	NE	4,452	6,546	9,218	12,142
	NS	4,452	6,546	9,218	12,142
	Max_Mises (Mpa)	3.14	3.22	3.44	3.32
FEM	NE	60,312	88,264	149,236	210,373
	NN	116,213	171,786	268,687	310,240
	Max_Mises (Mpa)	3.28	3.31	3.31	3.32

$$\begin{cases} U_1 = -2{x_1}^2 + 3{x_2}^2 + 3{x_3}^2 \\ U_2 = 3{x_1}^2 - 2{x_2}^2 + 3{x_3}^2 \\ U_3 = 3{x_1}^2 + 3{x_2}^2 - 2{x_3}^2 \end{cases}, \tag{33}$$

The physical variables are approximated by dual interpolation constant elements in DiBFM with continuous mesh and DiBFM with discontinuous mesh. Young's modulus E=1.0Mpa and Poisson's ratio ν=0.25, are prescribed. The discontinuous meshes are shown in Fig. 8.

Table 2 summarizes the relative errors of Von-Mises along the line AB and CPU consumption, and Fig. 9 (a), (b) give the contrast between DiBFM with continuous meshes and discontinuous meshes in accuracy and computational efficiency. Information above illustrates that no matter accuracy or computational efficiency improves gradually with the increasing numbers of source points, and DiBFM with discontinuous mesh possess equivalent computational efficiency comparing with continuous meshes in the same accuracy; except that, the convergence of the new method can be demonstrated.

Numerical solutions by DiBFM with two kinds of meshes and the

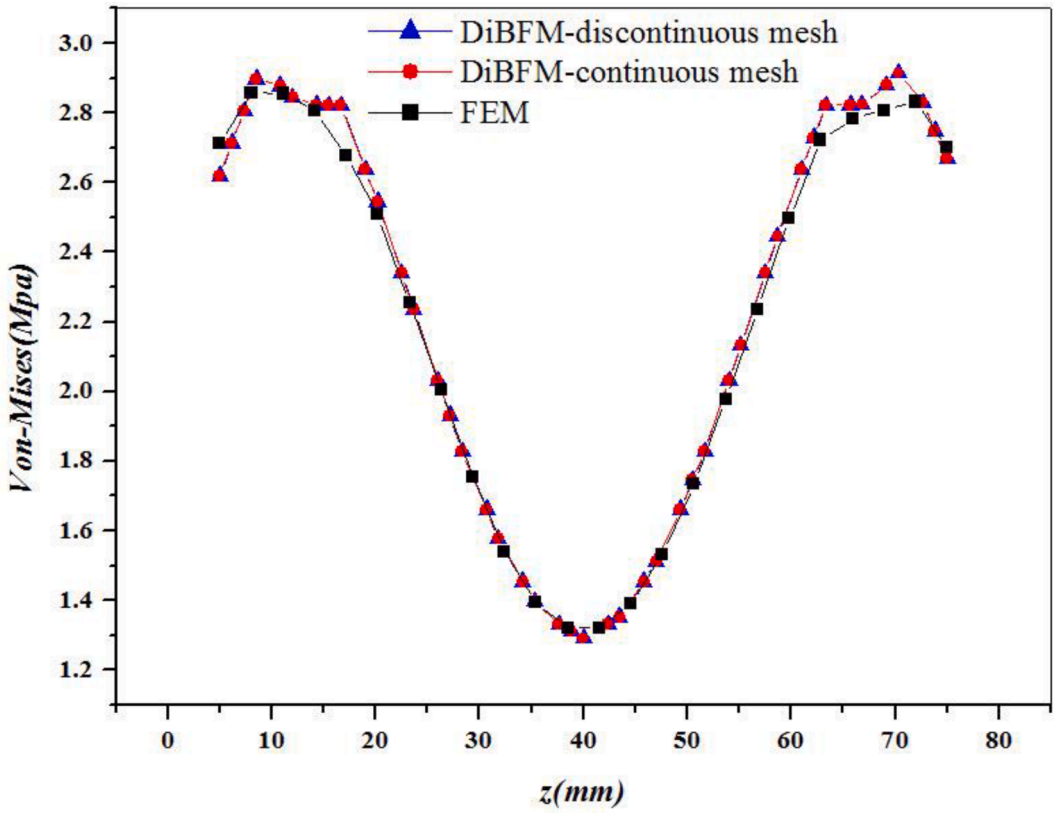


Fig. 17. Comparison of Von-Mises stress along line AB

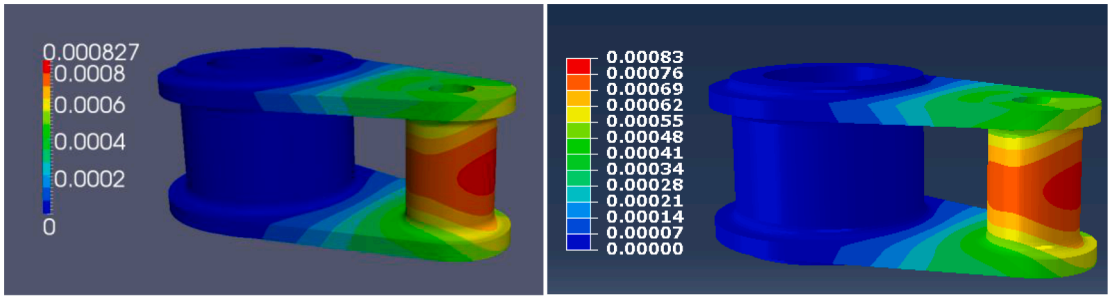


Fig. 18. Contour plots of displacement: (a) DiBFM: discontinuous meshes with 4,984 elements, 4,984 source nodes, and (b) FEM: with 210,373 elements, 310,240 nodes.

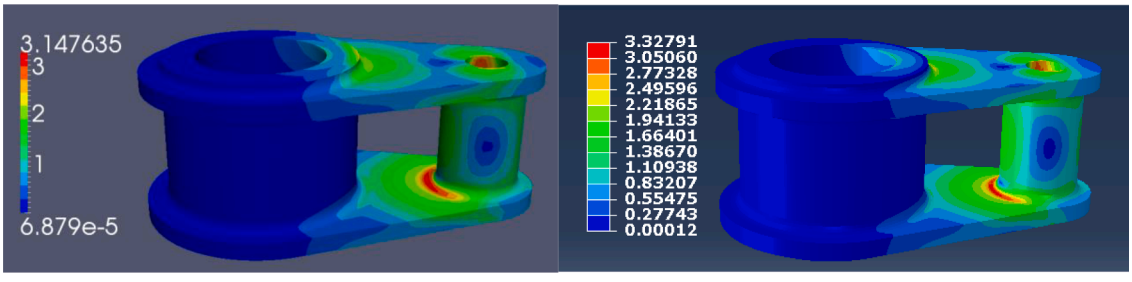


Fig. 19. Contour plots of Von-Mises stress: (a) DiBFM: discontinuous meshes with 4,984 elements, 4,984 source nodes, and (b) FEM: with 210,373 elements, 310,240 nodes.

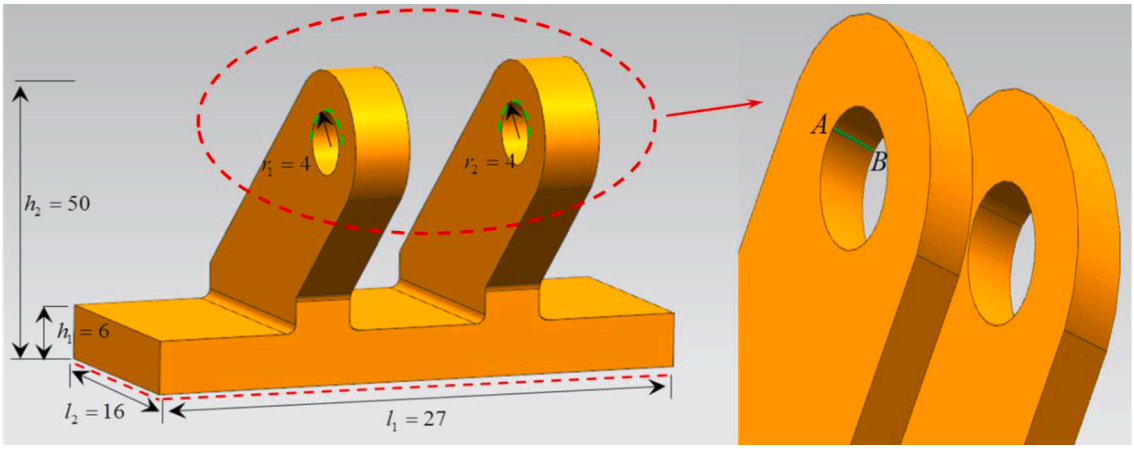


Fig. 20. Geometric model of angle connector

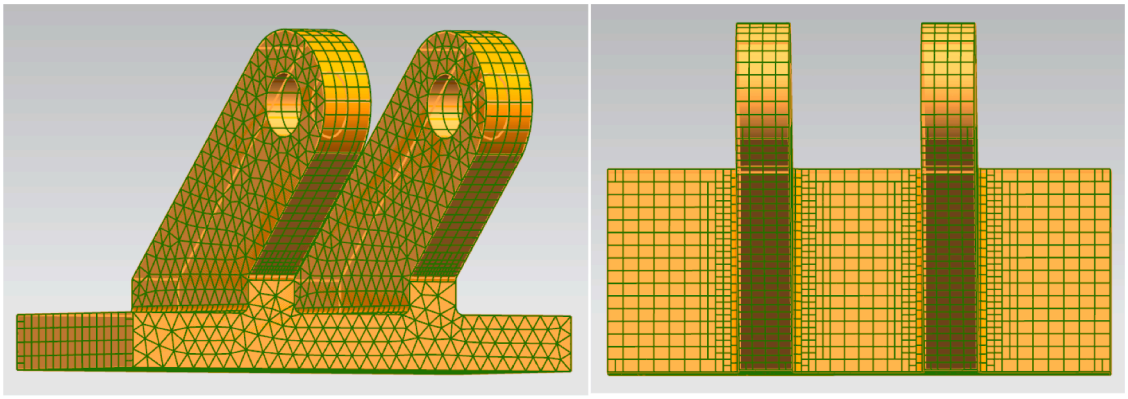


Fig. 21. Discontinuous meshes for angle connector: (a) oblique view, (b) vertical view.

Table 5
Numerical results of DiBFM with continuous mesh, DiBFM with discontinuous mesh and FEM for angle connector.

DiBFM-continuous mesh	NE	2,940	4,669	6,870	7,427
	NS	2,940	4,669	6,870	7,427
	Max_Mises (Mpa)	2.27	2.22	2.25	2.22
DiBFM-discontinuous mesh	NE	2,834	4,896	6,784	7,506
	NS	2,834	4,896	6,784	7,506
	Max_Mises (Mpa)	2.26	2.22	2.25	2.22
FEM	NE	64,563	102,343	184,562	223,433
	NN	122,762	212,436	286,421	323,127
	Max_Mises (Mpa)	2.26	2.27	2.27	2.28

exact Von-Mises along AB are plotted in Fig. 9 (c). Displacement and Von-Mises calculated by discontinuous meshes with 4,216 source points are described in Fig. 10. The contour plots demonstrate that analytical solution can be reproduced by our method, accurately. Through the numerical results, the reliability of our algorithm can be illustrated, sufficiently.

5.2. Shaft sleeve problem

The following example studies a cubic polynomial solution problem. A shaft sleeve is employed to illustrate the validity and the convergence of the discontinuous meshes algorithm in solving high-order analytical field problem. Displacement boundary conditions (see Eq. (34)) are imposed on the outer surface with the red dotted lines, and traction boundary conditions are prescribed on the remaining boundary faces. The dimensions of the shaft sleeve are shown in Fig. 11. The cubic polynomial solution is given by:

$$\begin{cases} U_1 = {x_2}^3 - {x_3}^3 - 3(x_2 - x_3){x_1}^2 \\ U_2 = {x_3}^3 - {x_1}^3 - 3(x_3 - x_1){x_2}^2 \\ U_3 = {x_1}^3 - {x_2}^3 - 3(x_1 - x_2){x_3}^2 \end{cases} \tag{34}$$

The same material parameters and interpolation elements as in Section 5.1 are adopted. Discretization of the geometric surface by discontinuous meshes is shown in Fig. 12.

Relative Von-Mises errors and CPU time along the given line AB are listed in a form of table as well in Table 3, and line charts concerning the accuracy and computational efficiency by DiBFM with different mesh method are given Fig. 13 (a)-(c). It is hardly to use the same number of meshes in these two methods, so similar mesh in number is adopted. From the table and figure, it is not difficult to see that DiBFM with discontinuous meshes can come up to the robust algorithm (DiBFM with continuous meshes) in accuracy and computational cost, and the convergence of the proposed method can be testified by Table 3 and Fig. 13 (a), (b) once again. The smooth contour plots in Fig. 14 demonstrate that higher-order analytical field can also be simulated through our method with 4,878 source points, accurately. Simultaneously, it shows that the discontinuous meshes work very well in complicated geometric structure.

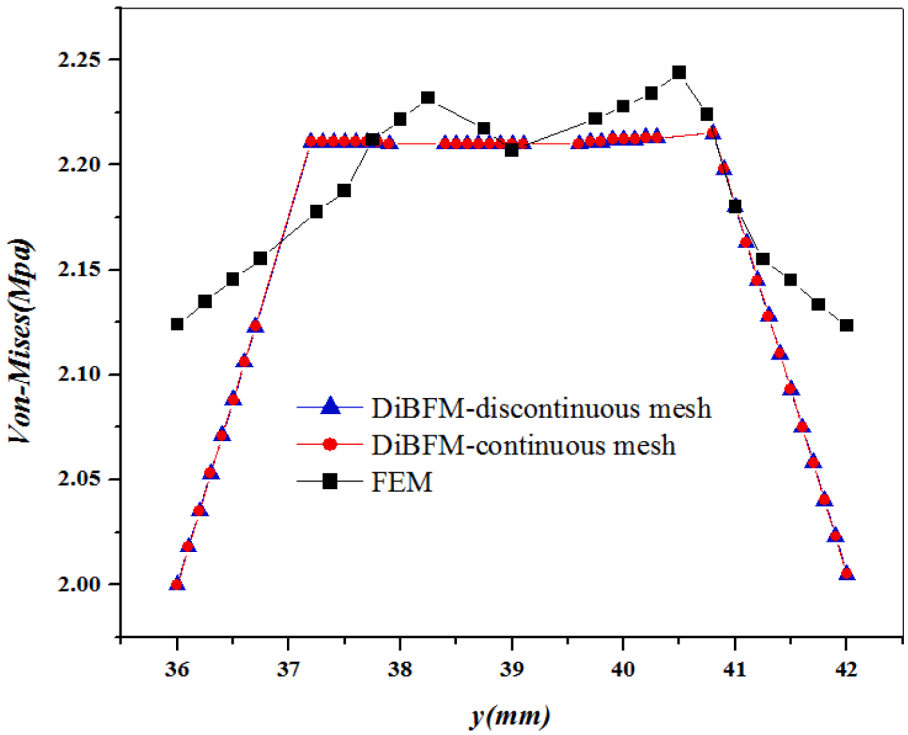


Fig. 22. Comparison of Von-Mises stress along line AB

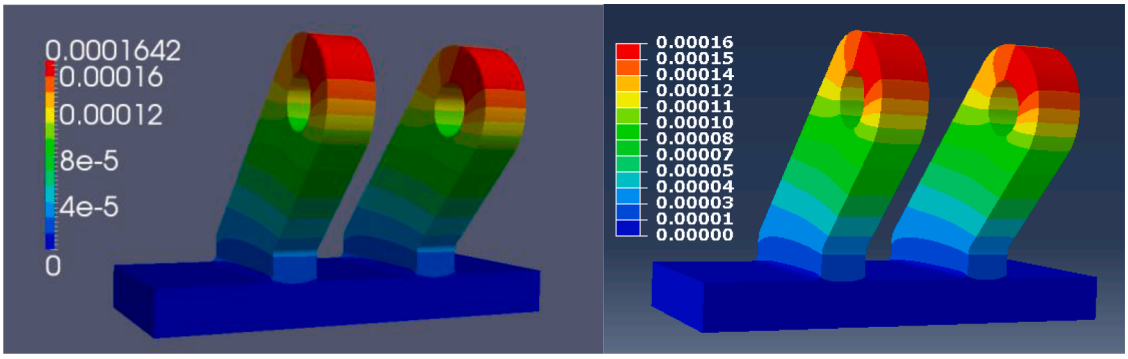


Fig. 23. Contour plots of displacement: (a) DiBFM: discontinuous meshes with 4,896 elements, 4,896 source nodes, and (b) FEM: with 223,433 elements, 323,127 nodes.

5.3. Adapting piece problem

To illustrate the performance of the proposed method in engineering background problems, in this part, an adapting piece structure is employed and the geometric sizes are given in Fig. 15. Mixed boundary condition is specified: where $\overline{u} = 0$is imposed on the inner face of the big cylinder with the red dotted line, and $\overline{t} = 1.0Mpa$ is prescribed on the half inner face of the small cylinder with the green dotted line. The physical variables are approximated by constant elements, while by quadratic elements in FEM. Young's modulus E=*2.00E05Mpa* and Poisson's ratio v=0.25 are considered. Corresponding discontinuous meshes are shown in Fig. 16.

Table 4 lists the max Von-Mises by DiBFM based on two mesh methods and FEM. Numerical data above illustrate that there is no obvious fluctuation in max Von-Mises with the increasing number of elements, and the convergence of the new method can be verified in real engineering problem. The line charts in Fig. 17 give the Vom-Mises along line AB. The contour plots in Figs. 18 and 19 present the displacement and Vom-Mises by DiBFM using discontinuous mesh with 6,546 source points and FEM with 310,240 nodes, respectively. From the comparison, we can conclude that our numerical result has same variation tendency with the FEM. The practicability and reliability of the discontinuous meshes for solving real-life problem can be demonstrated.

5.4. Angle connector problem

The last geometric model (see Fig. 20) is investigated to illustrate the feasibility of our algorithms. Simultaneously, it can testify the practicability of the discontinuous meshes in more complex case. The size of the angle connector is described in Fig. 20. The same boundary condition in Section 5.3 is considered: $\overline{u} = 0$ is specified on the bottom face with the red dotted line, and $\overline{t} = 1.0Mpa$ is applied on the half of the inner face of circular holes with the green dotted lines. The same interpolation elements and material parameters shown in Section 5.3 are

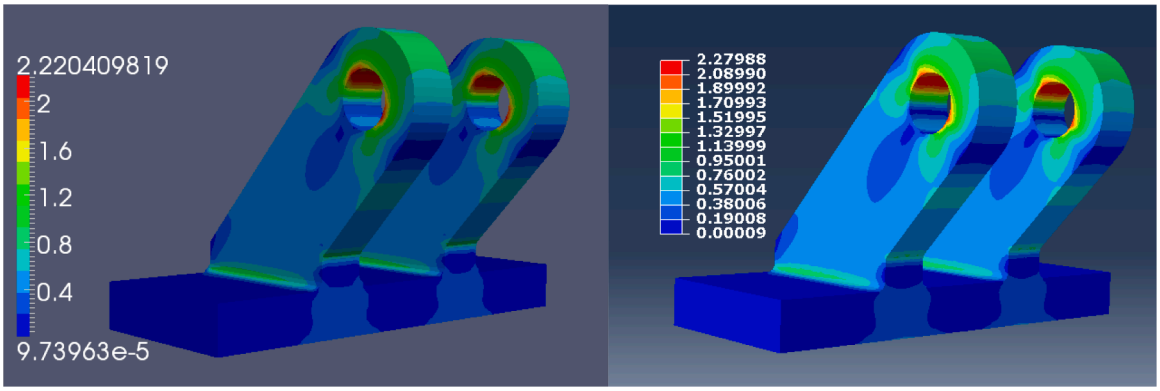


Fig. 24. Contour plots of Von-Mises stress: (a) DiBFM: discontinuous meshes with 4,896 elements, 19,548 nodes, and (b) FEM: with 223,433 elements, 323,127 nodes.

used in this example. Discontinuous meshes of this structure are presented in Fig. 21.

From the Table 5, convergence of DiBFM under discontinuous meshes can be demonstrated further. Simultaneously, the line charts in Fig. 22 illustrates that the local results by DiBFM along line AB are more considerable than FEM. Numerical solutions (see Figs. 23 and 24) by FEM with 223,433 elements, 323,127 nodes are for comparison purpose. This comparison can demonstrate that our method is feasible for solving more complicated structures. Excellent performance of the discontinuous meshes can also be revealed.

6. Conclusions and discussions

Using the discontinuous meshes to analyze the 3D elasticity problems has been implemented successfully in this research. The dual interpolation elements and the traditional continuous elements were employed to simulate the physical variables. A Binary-tree scheme was proposed to obtain the discontinuous meshes. This method can effectively minimize the difficulty of mesh division and avoid geometric repairing or model simplification for geometric models with small features and defects. For hang points in discontinuous meshes, the improved HMLS approximation and the compatibility conditions were used to construct the second-layer interpolation. This interpolation method offers an effective measure for handling hang points and guarantees the successful application of the discontinuous meshes in solving arbitrarily complicated structures. Moreover, it provides the possibility for the full-automatic CAE analysis and promotes the process of CAD/CAE integration. All the considerable numerical examples results have demonstrated that the presented methods are applicable and reliable when solving geometries structures with different engineering backgrounds.

At present, the proposed method was only used to solve elastostatic problems with a single domain. In future work, this method may be used to solve multiple domain problems, and the binary-tree-based mesh generation method can be considered to apply to parallelization of mesh division. Extension of the addressed method to solve 3D multiple domain problems and non-linear problems such as 3D contact problems is in progress.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by National Natural Science Foundation of China under grant numbers 11772125 and 11472102.

References

[1] Tristano J R, Owen S J, Canann S A. Advancing front surface mesh generation in parametric space using a riemannian surface definition. In: Proceedings of 7th International Meshing Roundtable; 1998. p. 429–45.
[2] Lan T S, Lo S H. Finite element mesh generation over analytical curved surfaces. Comput Struct 1996;59(2):301–9.
[3] Baehmann Peggy L. Robust, geometrically based, automatic two-dimensional mesh generation. Int J Numer Methods Eng 1987;24(6):1043–78.
[4] Borouchaki H, George P L, Hecht F, et al. Delaunay mesh generation governed by metric specifications, Part I, algorithms. Finite Elem Anal Des 1997;25(1-2):61–83.
[5] Baldwin K H, Schreyer H L. Automatic generation of quadrilateral elements by a conformal mapping. Eng Comput 1985;2(3):187–94.
[6] Yerry M, Shephard M. A modified quadtree approach to finite element mesh generation. IEEE Comput Graphics Appl 1983;3(1):39–46.
[7] Ang WT, Yun BI. A complex variable boundary element method for axisymmetric heat conduction in a nonhomogeneous solid. Appl Math Comput 2011;218:2225–36.
[8] Wen PH, Aliabadi MH, Rooke DP. Cracks in three dimensions: A dynamic dual boundary element analysis. Comput Meth Appl Mech Eng 1998;167:139–51.
[9] Schnack E, Chen HB. A multi-variable non-singular BEM in 2D elasticity. Eur J Mech A Solids 2001;20:645–59.
[10] Telles JCF, Brebbia CA. On the application of the boundary element method to plasticity. Appl Math Modell 1979;3:466–70.
[11] Manolis GD, Banerjee PK. Conforming versus non-conforming boundary elements in three-dimensional elastostatics. Int J Numer Method Eng 1986;23:1885–904.
[12] Florez WF, Power H. Comparison between continuous and discontinuous boundary elements in the multidomain dual reciprocity method for the solution of the two-dimensional Navier–Stokes equations. Eng Anal Bound Elem 2001;25:57–69.
[13] Parreira P. On the accuracy of continuous and discontinuous boundary elements. Eng Analy 1988;5:205–11.
[14] Yerry M, Shephard M. A modified quadtree approach to finite element mesh generation. IEEE Comput Graph Applica 1983;3(1):39–46.
[15] Baehmann P L, Wittchen S L, Shephard M S, et al. Robust, geometrically based, automatic two-dimensional mesh generation. Int J Numer Methods Eng 1987;24:1043–78.
[16] Yiu K F C, Greaves D M, Cruz S, et al. Quadtree grid generation: Information handling, boundary fitting and CFD applications. Comput Fluid 1996;25(8):759–69.
[17] Lancaster P, Salkauskas K. Surfaces generated by moving least squares methods. Math Comput 1981;37(155):141–58.
[18] Zhang JM, Lin WC, Dong YQ, et al. A double-layer interpolation method for BIE implementation for potential problems. Appl Math Modell 2017;51:250–69.
[19] Zhang J, Lin W, Dong Y. A dual interpolation boundary face method for elasticity problems. Eur J Mech A Solid 2019;73:500–11.
[20] Zhang J, Shu X, Trevelyan J, et al. A solution approach for contact problems based on the dual interpolation boundary face method. Appl Math Modell 2019;70(6):643–58.
[21] Zhang Jianming, Chi Baotao, Singh Krishna M, Zhong YuDong, Ju Chuanming. A binary-tree element subdivision method for evaluation of nearly singular domain integrals with continuous or discontinuous kernel. J Comput Appl Math 2019;362:22–40.

[22] Zhang Jianming, Qin Xianyun, Han Xu, Li Guangyao. A boundary face method for potential problems in three dimensions. Int J Numer Method Eng 2009;80:320–37.
[23] Zhang J M, He R, Chi B T, et al. A dual interpolation boundary face method with Hermite-type approximation for potential problems. Appl Math Modell 2020;81:457–72.
[24] Zhang Jianming, He Rui, Lin Weicheng, Yang Le, Chi Baotao, Ju Chuanming, et al. A dual interpolation boundary face method with Hermite-type approximation for elasticity problems. Eur J Mech/A Solid 2020;82:104005.
[25] Mirzaei D, Schaback R, Dehghan M. On generalized moving least squares and diffuse derivatives. IMA J Numer Anal 2012;32:983–1000.
[26] Li XL, Li SL. On the stability of the moving least squares approximation and the element-free Galerkin method. Comput Math Appl 2016;72:1515–31.
[27] Li G, Aluru NR. Boundary cloud method: a combined scattered point/boundary integral approach for boundary-only analysis. Comput Meth Appl Mech Eng 2002;191:2337–70.
[28] Li G, Aluru NR. A boundary cloud method with a cloud-by-cloud polynomial basis. Eng Anal Bound Elem 2003;27:57–71.
[29] Zhang Jianming, Ju Chuanming, Divo Eduardo, Zhong Yudong, Chi Baotao. A binary-tree subdivision method for evaluation of singular integrals in 3D BEM. Eng Anal Boundary Elem 2019;103:80–93.
[30] Liu Y. Fast multipole boundary element method: theory and applications in engineering. Cambridge university press; 2009.