

# An enhanced octree polyhedral scaled boundary finite element method and its applications in structure analysis



Degao Zou<sup>a,b</sup>, Kai Chen<sup>a,b,\*</sup>, Xianjing Kong<sup>a,b</sup>, Jingmao Liu<sup>a,b</sup>

<sup>a</sup> The State Key Laboratory of Coastal and Offshore Engineering, Dalian University of Technology, Dalian, Liaoning 116024, China

<sup>b</sup> School of Hydraulic Engineering, Dalian University of Technology, Dalian, Liaoning 116024, China

## ARTICLE INFO

### Keywords:

Octree polyhedron  
Scaled boundary finite element method  
Polygonal shape function  
Swift modelling  
Three-dimensional

## ABSTRACT

In this paper, an enhanced octree polyhedral scaled boundary finite element method (SBFEM) is proposed in which arbitrary convex polygon (pentagon, hexagon, heptagon, octagon etc.) can be directly served as boundary face elements. The presented method overcomes the existing SBFEM's limitation that boundary face is strictly restricted to be a quadrangle or triangle. The conforming shape functions are constructed using a polygon mean-value interpolation scheme for polyhedral face. A highly efficient octree mesh generation technology is introduced to accelerate the progress of pre-treatment, wherein the mesh information can be directly used in the enhanced SBFEM. The accuracy of the proposed method is first verified using a beam under shear and torsion load. Another three more complicated geometries including a nuclear power plant structure, as well as two sculptures named Terra-Cotta Warriors and Sioux Falls Church are presented to demonstrate the application and robustness of the proposed method. The new method possesses appealing versatility and offers a swift adaptive capacity in mesh generation, which can provide a powerful technique for the simulation of complex geometries, rapid-design analysis and multi-scale problems.

© 2017 Elsevier Ltd. All rights reserved.

## 1. Introduction

The finite element method (FEM) is a powerful computational technique used in numerical simulation that has been extensively applied to structural analysis since it was first proposed. Moreover, a wide variety of improvements and alterations to this method are continually emerging. The three-dimensional finite elements are typically tetrahedrons (four vertices and four faces) and hexahedrons (eight vertices and six faces). The hexahedron can degenerate into shapes with fewer vertices, such as pentahedrons (five or six vertices and five faces), which are sufficient for many applications. However, there is a growing need for more general polyhedral shapes with increasing geometric complexity to include shapes that have an arbitrary number of vertices and faces.

Polyhedral element shapes can provide more flexibility for meshing geometrically complex structures, which will enable a rapid design-to-analysis paradigm. Finite volume methods based on polyhedral cells have reached a level of maturity in fluid dynamic simulations, as evidenced by their availability and use in commercial software [1,2]. Mimetic finite difference (MFD) methods capable of handling general three-dimensional meshes have also been a topic of active research and have been successfully applied to diffusion, elasticity, and fluid flow problems [3–7]. However, the extension of the FEM to this field has

been relatively slow despite the availability of special interpolation functions. This progress has primarily been slow because these interpolants are subject to restrictions on the admissible element geometry (e.g., convexity and maximum valence count) and can be sensitive to geometric degeneracies. More importantly, calculating these functions and their gradients are often prohibitively expensive. The numerical evaluation of weak form integrals with sufficient accuracy poses yet another challenge due to the non-polynomial nature of these functions as well as the arbitrary integration domain [8].

Polyhedral finite element formulations have only recently been proposed in the literature [9,10–13]. We highlight several works that have aimed to overcome these barriers. In his ground-breaking theoretical work that has drawn widespread attention from subsequent researchers, Wachpress [14] proposed polygonal rational shape functions for plane problems. However, this work was not extended to polyhedrons until 1996 [15] and was then extended to only polyhedrons with triangular faces. Similarly, Wicke et al. [11] developed a formulation for convex polyhedrons using mean-value coordinates where the faces were restricted to triangles. The polyhedral finite elements [10,13,16] were developed using the shape functions derived from meshless methods, and Idelsohn et al. [10] used non-Sibsonian coordinates that required a certain Voronoi construction within an element. To achieve numerical integration, the polyhedron was first subdivided into tetrahedrons, and

\* Corresponding author.

E-mail address: [chenkai@mail.dlut.edu.cn](mailto:chenkai@mail.dlut.edu.cn) (K. Chen).

tetrahedral subdivision was performed using the face centroids and the centroid of the polyhedron. Then, element integration was performed using standard integration rules for tetrahedrons [17]. Milbradt and Pick [13] developed a formulation for both convex and non-convex polyhedrons using natural-element coordinates to construct the shape function. Ghosh [18] developed the Voronoi-cell FEM using a stress-based finite element approach. In this formulation, the shape functions were constructed using a polynomial basis optimised for smoothness. More recently, researchers have focused on the virtual element method (VEM), which has addressed some of the aforementioned challenges encountered by finite element schemes [19–22].

The scaled boundary finite element method (SBFEM) was developed by Wolf and Song in the mid-1990s [23,24] and has been increasingly used for numerical simulations of structures. For example, Goswami [25], Hell [26], and Saputra et al. [27] used SBFEM to conduct three-dimensional crack analysis. Man et al. [28] used this method to simulate plate bending, and Lin et al. [29] conducted a sloshing analysis of liquid storage tanks. An alternative hydrodynamic pressure method for a high concrete-faced rockfill dam has also been proposed [30]. Birk and Behnke [31] modelled dynamic soil-structure interactions in layered soil using a modified SBFEM. This method was also used to simulate electrostatic problems [32,33], short-crested wave interactions [34] and the time-domain analyses of the layered soil [35]. Novel consistent analysis approach for uniform beams [36], nonlinear analysis application in two-dimensional geotechnical structures [37] and error study of Westergaard's approximation using SBFEM [38] are emerged, too. Discretisation is conducted only at the boundary surface, and each face can be treated as a sub-element in each volume element, thus making the SBFEM easy to formulate for a polyhedral element. More recently, Liu et al. [39] developed a rapid automatic polyhedral mesh generation and scaled boundary finite element analysis. In this research, the octree mesh technique was adopted, and the two-dimensional isoparametric element was utilised for each face element. However, each face is still restricted to being either a triangle or quadrangle, and additional efforts are required to manage the hanging nodes, which may exceed the capacity of the method to manage general polyhedrons. This method will have greater influence if the hanging nodes are free from addressing.

In 1975, Wachspress [40] developed polygonal finite elements with an arbitrary number of sides. Subsequently, Meyer et al. [41] and Floater et al. [42] generalised these elements to arbitrarily shaped polygons using the concept of barycentric and mean-value coordinates. Voronoi diagrams and natural neighbour shape functions were considered to develop conforming polygon elements [43]. Recently, an alternative method using the maximum entropy approach [44] to formulate polygonal shape functions was also studied [45,46]. In summary, polygonal shape functions have reached a level of maturity that provides a powerful technique for formulating a general polyhedral element using SBFEM.

Based on previous studies, a more distinct three-dimensional polyhedral scaled boundary finite element method (PSBFEM3D) is proposed. The shape function of the boundary face elements is constructed using the mean-value coordinates instead of an isoparametric element. Subsequently, the solution process is identical to that of the SBFEM, which is described in detail in Section 3. The proposed method is easy to formulate and implement in a program. More importantly, the difficulties in calculating the shape functions and their gradients are significantly decreased compared with polyhedral FEM. In addition, this method can be seamlessly combined with efficient octree meshing, which is more advantageous for rapid adaptive modelling. The PSBFEM3D offers appealing universality and is efficient for managing a cross-scale or multi-scale mesh. This method may have significant potential for use in practical applications.

The remainder of this paper is organised as follows. A brief theoretical derivation of the mean-value shape function is introduced in Section 2. Section 3 describes the scaled boundary polyhedron formulation for a polyhedral element. The development platform of the

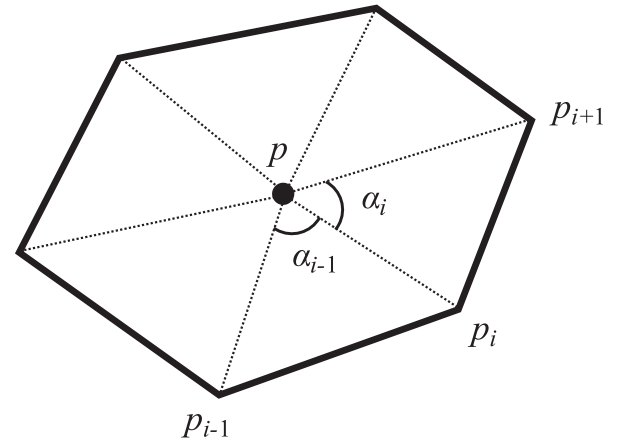


Fig. 1. Mean-value coordinates.

proposed algorithm is introduced in Section 4. The reliability of the procedure is validated using four numerical examples in Section 5. Section 6 summarises the major conclusions that can be drawn from this study.

## 2. Mean-value shape functions on polygons

The mean-value shape function is a type of polygon function in which the interpolant is simpler and computationally attractive. The desirable properties for finite element interpolants indicated in reference [43] are satisfied, and the proof was presented in that previous study. Here, we briefly discuss the formulation of the mean-value shape functions on general polygon elements, and the reader can refer to [43] for a more detailed description. The linearly precise mean-value coordinate is written as [47]

$$N_i(\mathbf{x}) = \frac{w_i(\mathbf{x})}{\sum_{j=1}^n w_j(\mathbf{x})} \quad (2.1)$$

$$w_i(\mathbf{x}) = \frac{\tan(\alpha_{i-1}/2) + \tan(\alpha_i/2)}{\|\mathbf{x} - \mathbf{x}_i\|} \quad (2.2)$$

$$\tan(\alpha_i/2) = \frac{\sin \alpha_i}{1 + \cos \alpha_i} \quad (2.3)$$

where  $w_i(\mathbf{x})$  is the mean-value weight function,  $\|\mathbf{x} - \mathbf{x}_i\|$  is the Euclidean distance between  $p$  and  $p_i$  (Fig. 1), point  $p$  is set as the geometric centre, and  $n$  is the number of vertices. The mean-value shape functions can be readily computed by substituting Eqs. (2.3) and (2.2) into (2.1). Notably, the formulation is also applicable to a non-convex polygon.

Typically, the mean-value shape functions are derived in the physical  $x$ -coordinate itself. To improve the SBFEM and to simplify the integral, a technique to construct conforming approximations on polygons is described using mean-value shape functions. Similar to isoparametric elements, the shape function is defined on a canonical element with local coordinates  $\xi \equiv (\xi_1, \xi_2) \in \mathbf{R}_0$ . The canonical elements are given for a triangle, quadrangle, pentagon and hexagon and are illustrated in Fig. 2. The nodes lie on the same circumcircle in each case, and the geometric centre is located in the centre of the circumcircle; hence, all vertices of the polygon are natural neighbours for any point in  $\mathbf{R}_0$ . The vertical coordinates of an  $n$ -gon are expressed as  $(\cos 2\pi/n, \sin 2\pi/n)$ ,  $(\cos 4\pi/n, \sin 4\pi/n), \dots$ , and  $(1, 0)$ . Then, the shape function can first be defined using the reference coordinate system. Subsequently, each arbitrary polygon can be transformed into corresponding canonical elements using the isoparametric mapping function  $\mathbf{N}$ . The mapping for a pentagonal element is illustrated in Fig. 3.

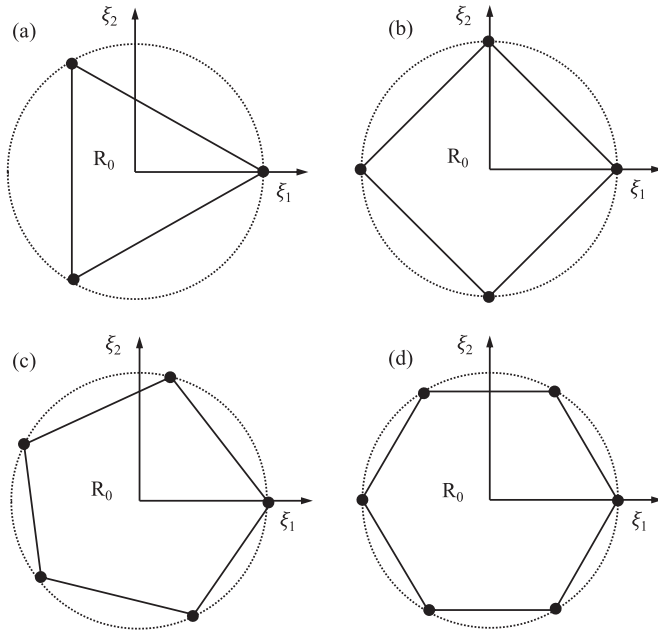


Fig. 2. Canonical elements: (a) triangle, (b) quadrangle, (c) pentagon, and (d) hexagon.

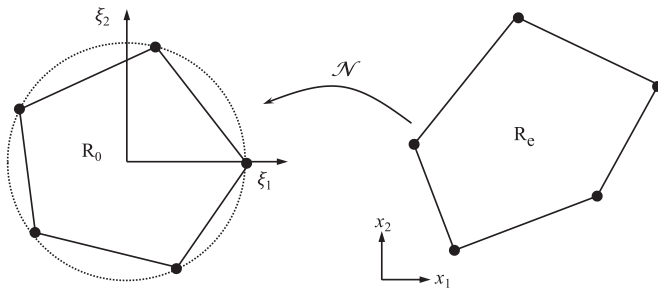


Fig. 3. Isoparametric mapping.

### 3. Formulation of the polyhedral scaled boundary finite element method

#### 3.1. Selection of element types for the discretisation boundary

The SBFEM inherits the concept of discretisation from FEM. Similarly, the plane isoparametric element is adopted for boundary elements in the existing SBFEM. Only two options are available for the element: a triangle or quadrangle. This element library may be sufficient for many application, but there is still room for improvement for a more general, efficient and robust method.

The discretisation is only conducted on the boundary in SBFEM. Thus, for a three-dimensional problem, only plane element interpolant functions are required. Additionally, the polygonal elements have been developed for decades, which provide more flexibility for the shape of elements. These two methods have been evolving in parallel and not previously been combined. Generally, interdisciplinary studies bring about unexpected superiority. Thus, this paper explores combining these two highly complementary techniques. The boundary interpolation is accomplished using polygon mean-value shape functions, which are pre-calculated using an external code and imported into the main program as constants. Subsequently, the polyhedron shape functions are constructed using SBFEM.

Compared with the existing method, the modified SBFEM offers a more abundant element library, which is illustrated in Fig. 5. This proposed method is more universal and more efficient; for instance, when the number of element sides exceeds four, extra effort is needed to sat-

isfy the demands of the traditional method (the shape must be a triangle or quadrangle). Therefore, more boundary face elements and nodes are generated, increasing the computational burden. A representative drawing is presented in Fig. 4.

Obviously, the polygonal boundary elements can be used as conveniently as the conventional plane triangle and quadrangle elements, and the cumbersome aspects of the existing SBFEM are eliminated in the modified SBFEM. Moreover, using the modified SBFEM would improve the computational efficiency for a large-scale simulation, as the faces of the boundary elements and the degrees of freedom (DOFs) could be decreased significantly. A typical comparison is shown in Table 1, considering polygonal boundary elements containing hanging nodes.

#### 3.2. Polygonal shape function and its derivative for boundary elements

A considerable amount of literature has been published on improving the SBFEM over the past several years. Repetitive formulations are omitted in this paper, and only a few vital equations are discussed. Here, the shape functions and matching derivatives are introduced. First, the *similarity* is addressed in the context of a solid element. A three-dimensional element with discretised boundary surfaces, as shown in Fig. 6, is used as an example. A similarity centre, denoted  $O$ , is selected at the geometric centre, and the entire boundary must be directly visible from this point.

In contrast to the existing SBFEM, for each face element, the domain is interpolated using a polygonal shape function  $N$  formulated with local coordinates  $(\xi_1, \xi_2)$ , as discussed in Section 2.  $(\xi_1, \xi_2)$  can be called the scaled boundary coordinates. Then, the coordinates can be written as

$$\begin{aligned} x(\xi_1, \xi_2) &= N(\xi_1, \xi_2)\mathbf{x} \\ y(\xi_1, \xi_2) &= N(\xi_1, \xi_2)\mathbf{y} \\ z(\xi_1, \xi_2) &= N(\xi_1, \xi_2)\mathbf{z} \end{aligned} \quad (3.1)$$

where  $\mathbf{x}$ ,  $\mathbf{y}$ , and  $\mathbf{z}$  are the coordinate vectors of the nodes on the boundary. The geometry of a pyramid can be described by scaling the boundary with a dimensionless radial coordinate  $\xi$  connecting the scaling centre  $O(\hat{x}_0, \hat{y}_0, \hat{z}_0)$  and the boundary nodes. The radial coordinate  $\xi$  varies from 0 at the scaling centre to 1 at the boundary. Therefore, the scaled outer image of the boundary is obtained by scaling the Cartesian coordinates of the boundary with the same scaling factor  $\xi$ , which can be expressed as

$$\begin{aligned} \hat{x}(\xi, \xi_1, \xi_2) &= \xi N(\xi_1, \xi_2)\mathbf{x} + \hat{x}_0 \\ \hat{y}(\xi, \xi_1, \xi_2) &= \xi N(\xi_1, \xi_2)\mathbf{y} + \hat{y}_0 \\ \hat{z}(\xi, \xi_1, \xi_2) &= \xi N(\xi_1, \xi_2)\mathbf{z} + \hat{z}_0 \end{aligned} \quad (3.2)$$

$(\xi, \xi_1, \xi_2)$  are called the scaled boundary coordinates in a three-dimensional domain. The change in coordinates from  $(\hat{x}, \hat{y}, \hat{z})$  to  $(\xi, \xi_1, \xi_2)$  is called the *scaled boundary transformation*.

The isoparametric mapping  $x(\xi) = \sum_{i=1}^n N_i(\xi)x_i$  was adopted, and a standard procedure was applied to accomplish the transformation, as in the FEM. To obtain the shape function derivatives in the physical coordinate system  $\mathbf{x} \equiv (x_1, x_2, \dots, x_n)$ , the Jacobian matrix of the transformation  $\mathbf{J}_m = \partial \mathbf{x} / \partial \xi$  is first required. The derivatives can be formulated as  $\nabla N_i = \mathbf{J}_m^{-1} \nabla_{\xi} N_i^m$ , where  $\nabla_{\xi}$  denotes the gradient with respect to the reference coordinates. Finally, the Jacobian matrix and determinant on the face element can be expressed as

$$[\mathbf{J}_m(\xi_1, \xi_2)] = \begin{bmatrix} x(\xi_1, \xi_2) & y(\xi_1, \xi_2) & z(\xi_1, \xi_2) \\ x(\xi_1, \xi_2)_{,\xi_1} & y(\xi_1, \xi_2)_{,\xi_1} & z(\xi_1, \xi_2)_{,\xi_1} \\ x(\xi_1, \xi_2)_{,\xi_2} & y(\xi_1, \xi_2)_{,\xi_2} & z(\xi_1, \xi_2)_{,\xi_2} \end{bmatrix} \quad (3.3a)$$

$$|\mathbf{J}_m(\xi_1, \xi_2)| = x(y_{,\xi_1} z_{,\xi_2} - z_{,\xi_1} y_{,\xi_2}) + y(z_{,\xi_1} x_{,\xi_2} - x_{,\xi_1} z_{,\xi_2}) + z(x_{,\xi_1} y_{,\xi_2} - y_{,\xi_1} x_{,\xi_2}) \quad (3.3b)$$

where the derivatives can be expressed as

$$\begin{aligned} x(\xi_1, \xi_2)_{,\xi_1} &= N(\xi_1, \xi_2)_{,\xi_1} \mathbf{x} \\ y(\xi_1, \xi_2)_{,\xi_1} &= N(\xi_1, \xi_2)_{,\xi_1} \mathbf{y} \\ z(\xi_1, \xi_2)_{,\xi_1} &= N(\xi_1, \xi_2)_{,\xi_1} \mathbf{z} \end{aligned} \quad (3.4a)$$

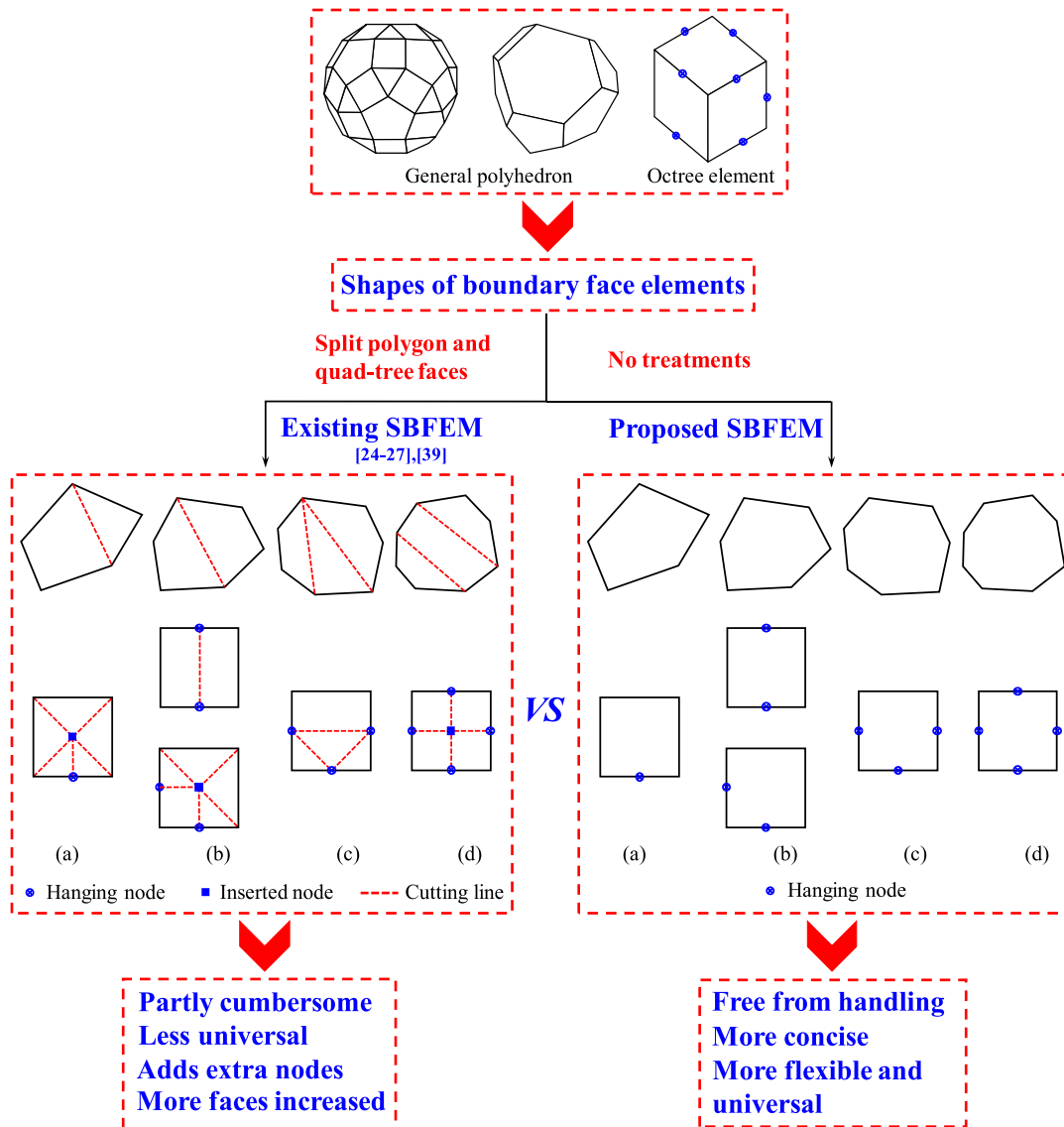


Fig. 4. Comparison of the representative strategies for managing polygonal boundary elements: (a) pentagon, (b) hexagon, (c) heptagon, and (d) octagon.

Table 1

Comparison of faces and nodes in the polygonal boundary element (with hanging nodes) in two methods.

Content	Method	Pentagon	Hexagon	Heptagon	Octagon
Faces	Existing SBFEM	5	2 or 5	4	4
	Modified SBFEM	1	1	1	1
Increase in faces	–	4	1 or 4	3	3
Nodes	Existing SBFEM	6	6 or 7	7	9
	Modified SBFEM	5	6	7	8
Increase in nodes	–	1	0 or 1	0	0

$$\begin{aligned} x(\xi_1, \xi_2)_{,\xi_2} &= \mathbf{N}(\xi_1, \xi_2)_{,\xi_2} \mathbf{x} \\ y(\xi_1, \xi_2)_{,\xi_2} &= \mathbf{N}(\xi_1, \xi_2)_{,\xi_2} \mathbf{y} \\ z(\xi_1, \xi_2)_{,\xi_2} &= \mathbf{N}(\xi_1, \xi_2)_{,\xi_2} \mathbf{z} \end{aligned} \quad (3.4b)$$

$$\mathbf{L} \mathbf{D} \mathbf{u} = \begin{bmatrix} \mathbf{N}(\xi_1, \xi_2) \\ \mathbf{N}(\xi_1, \xi_2)_{,\xi_1} \\ \mathbf{N}(\xi_1, \xi_2)_{,\xi_2} \end{bmatrix} \quad (3.6)$$

The Jacobian matrix can be alternatively expressed as

$$[\mathbf{J}_m(\xi_1, \xi_2)] = \begin{bmatrix} \mathbf{N}(\xi_1, \xi_2) \\ \mathbf{N}(\xi_1, \xi_2)_{,\xi_1} \\ \mathbf{N}(\xi_1, \xi_2)_{,\xi_2} \end{bmatrix} \begin{bmatrix} \mathbf{x} & \mathbf{y} & \mathbf{z} \end{bmatrix} \quad (3.5)$$

where  $\mathbf{L} \mathbf{D} \mathbf{u}$  denotes the gradients of the shape functions with respect to the reference coordinates and is written as

### 3.3. Nodal displacement functions in the radial direction

One of the appealing characteristics of the SBFEM is that the solution is analytical in the radial direction. To this end, a radial nodal displacement function  $\mathbf{u}(\xi)$  is introduced along the radial lines, where  $\xi$  varies from 0 to 1 from the scaling centre to the boundary. The nodal displacements on the boundary can be expressed as  $\{\mathbf{u}(\xi = 1)\} = \mathbf{u}_b$ , which can be



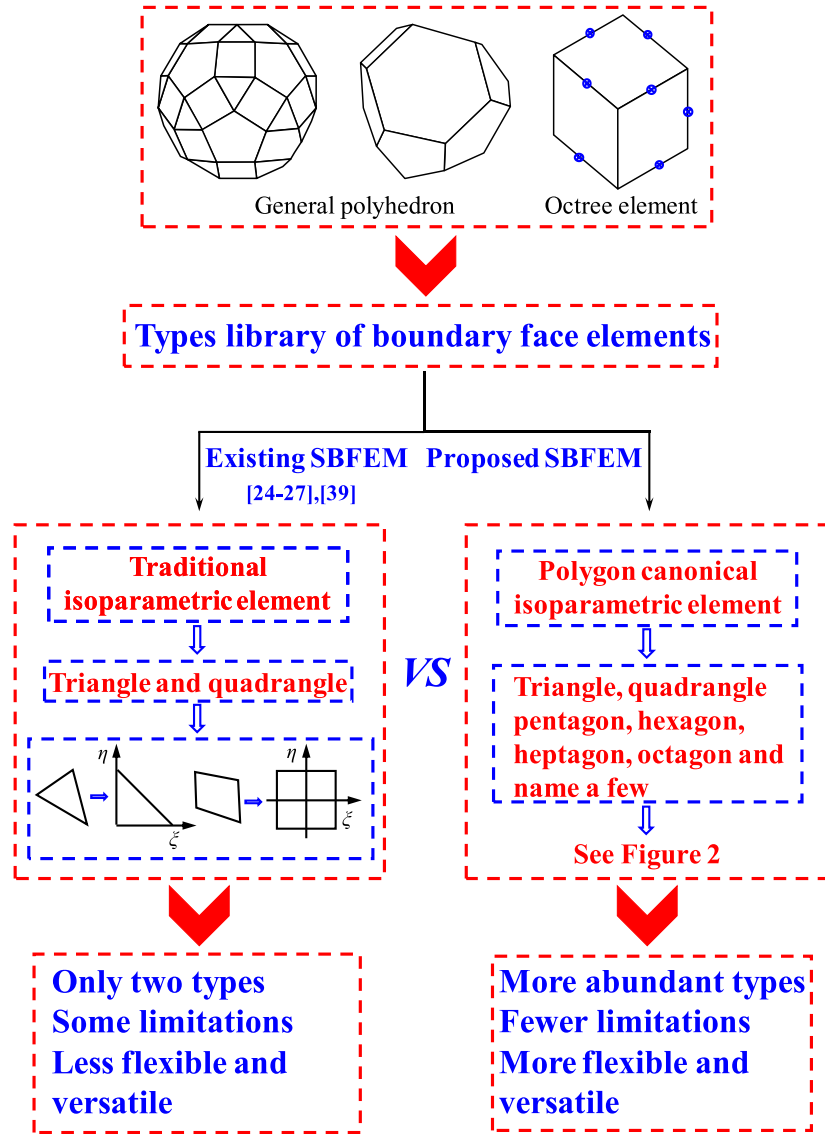


Fig. 5. The comparison of the element library for the discretisation boundary.

evaluated from the equilibrium equation. Each direction of the displacement components is also defined in the original Cartesian coordinates. The mean-value polygon displacement model is used to interpolate the boundary displacement.

For three-dimensional problems, the displacement components at a point  $(\xi, \xi_1, \xi_2)$  inside a pyramid are interpolated from the displacement function  $u(\xi)$ , which is written as shown in Eq. (3.13), where  $I$  is the  $3 \times 3$  identity matrix and  $N_i$  ( $i=1,2,3,\dots,n$ ) is the nodal interpolated function in a canonical isoparametric element. The expression of the polygonal shape function is extremely complex; therefore, the description is not expanded here.

$$u(\xi, \xi_1, \xi_2) = N^u(\xi_1, \xi_2)u(\xi) \quad (3.7a)$$

$$N^u(\xi_1, \xi_2) = [N_1 I, N_2 I, N_3 I, N_4 I, \dots, N_n I] \quad (3.7b)$$

### 3.4. Polyhedral element shape functions based on SBFEM

The variational principle is introduced to derive the scaled boundary finite element equation in the nodal displacement function  $u(\xi)$ . According to reference [20], the equilibrium equation can be expressed as

$$E_0 \xi^2 u(\xi)_{,\xi\xi} + (2E_0 + E_1^T - E_1) \xi u(\xi)_{,\xi} + (E_1^T - E_2) u(\xi) + F(\xi) = 0 \quad (3.8)$$

where the coefficient matrices  $E_0, E_1$ , and  $E_2$  are intermediate variables that depend on only the geometry and material properties of the domain and are evaluated on the face elements and assembled over the discretised polyhedron boundary.  $F(\xi)$  is a load vector, and only the contributions from body loads are considered in this paper. Then, the equation can be rewritten as

$$E_0 \xi^2 u(\xi)_{,\xi\xi} + (2E_0 + E_1^T - E_1) \xi u(\xi)_{,\xi} + (E_1^T - E_2) u(\xi) = 0 \quad (3.9)$$

and the coefficient matrices and mass matrix can be expressed as

$$E_0 = \int_{-1}^{+1} \int_{-1}^{+1} B_1^T D B_1 |J_m| d\xi_1 d\xi_2 \quad (3.10a)$$

$$E_1 = \int_{-1}^{+1} \int_{-1}^{+1} B_2^T D B_1 |J_m| d\xi_1 d\xi_2 \quad (3.10b)$$

$$E_2 = \int_{-1}^{+1} \int_{-1}^{+1} B_2^T D B_2 |J_m| d\xi_1 d\xi_2 \quad (3.10c)$$

$$M_0 = \int_{-1}^{+1} \int_{-1}^{+1} \rho N^T N |J_m| d\xi_1 d\xi_2 \quad (3.10d)$$

$$B_1(\xi_1, \xi_2) = b_1(\xi_1, \xi_2) N^u(\xi_1, \xi_2) \quad (3.10e)$$

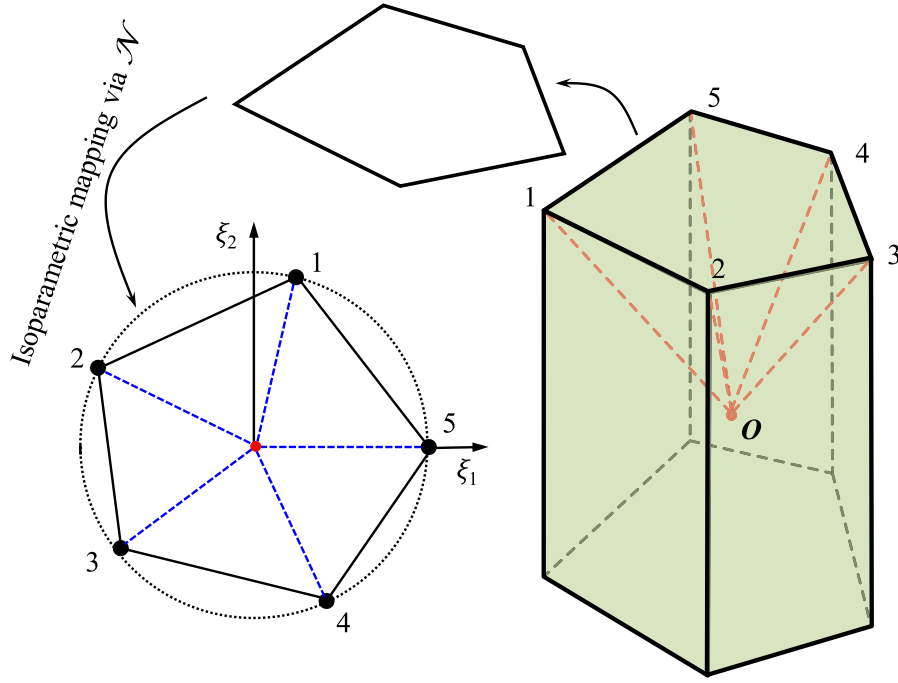


Fig. 6. Scaling centre  $O$  and isoparametric mapping in a three-dimensional polyhedral scaled boundary element.

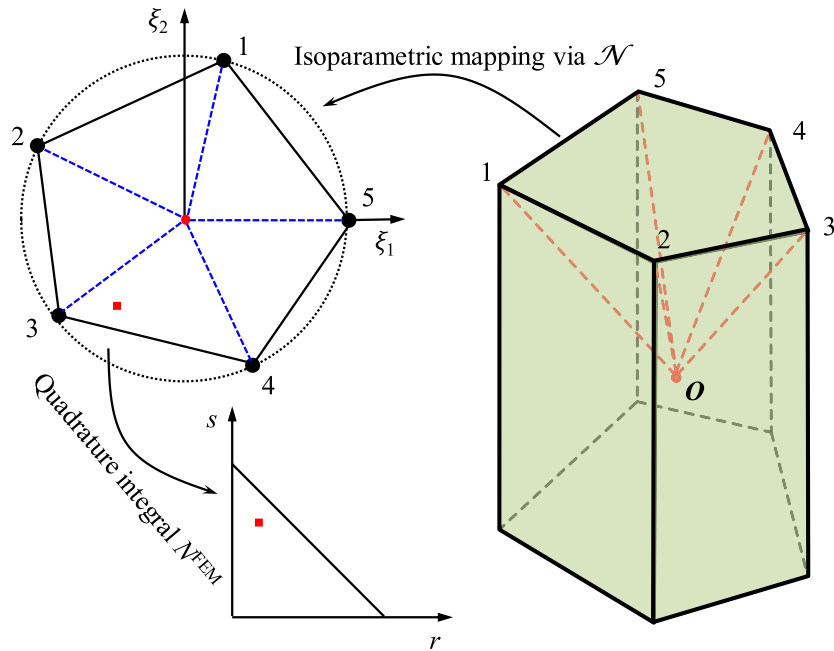


Fig. 7. Numerical integration in a three-dimensional polyhedral scaled boundary element.

$$\mathbf{B}_2(\xi_1, \xi_2) = \mathbf{b}_2(\xi_1, \xi_2) \mathbf{N}^u(\xi_1, \xi_2)_{,\xi_1} + \mathbf{b}_3(\xi_1, \xi_2) \mathbf{N}^u(\xi_1, \xi_2)_{,\xi_2} \quad (3.10f)$$

where  $\mathbf{B}_1$  and  $\mathbf{B}_2$  are strain-displacement transition matrices derived from the polygonal shape function  $\mathbf{N}$  and Jacobian matrix on the boundary,  $\mathbf{b}_i$  ( $i = 1, 2, 3$ ) can be referred in literature [48] and  $\mathbf{D}$  is the constitutive matrix.

The numerical integration can be performed by sub-dividing the canonical element into triangles and using the standard quadrature rules on each triangle. The divided sub-triangles are used solely for numerical integration. The diagram is illustrated in Fig. 7, where  $\mathbf{N}^{\text{FEM}}$  is the linear triangle shape function, as in the FEM.

Eq. (3.9) is a linear second-order matrix-form ordinary differential equation (ODE), where the nodal displacement function  $\mathbf{u}(\xi)$  can be analytically derived in the radial direction. The detailed procedure for this solution is described in reference [24]. In order to solve the SBFEM equation, a variable  $\mathbf{X}(\xi)$  which incorporate the nodal displacement function  $\mathbf{u}(\xi)$  and the nodal force function  $\mathbf{q}(\xi)$  is introduced.

$$\mathbf{X}(\xi) = \begin{Bmatrix} \xi^{0.5} \mathbf{u}(\xi) \\ \xi^{-0.5} \mathbf{q}(\xi) \end{Bmatrix} \quad (3.11)$$

The operation yields the first-order ODE

$$\xi \mathbf{X}(\xi)_{,\xi} = -\mathbf{Z} \mathbf{X}(\xi) \quad (3.12)$$

**Table 2**

Mean-value coordinates of the evaluated points (4-digit precision).

Coordinates	1	2	3	4	5
$\xi_1$	0.7182	0.2182	0.3727	0.0712	−0.0833
$\xi_2$	0.1585	0.1585	0.6340	0.7320	0.2565
Coordinates	6	7	8	9	10
$\xi_1$	−0.4878	−0.6742	−0.2697	−0.6742	−0.4878
$\xi_2$	0.5504	0.2939	0.0000	−0.2939	−0.5504
Coordinates	11	12	13	14	15
$\xi_1$	−0.0833	0.0712	0.3727	0.2182	0.7182
$\xi_2$	−0.2565	−0.7320	−0.6340	−0.1585	−0.1585

where  $\mathbf{Z}$  is a Hamiltonian matrix, which is formulated with the coefficient matrices  $\mathbf{E}_i (i=0,1,2)$  and the identity matrix  $\mathbf{I}$ .

$$\mathbf{Z} = \begin{bmatrix} \mathbf{E}_0^{-1} \mathbf{E}_1^T - 0.5 \mathbf{I} & -\mathbf{E}_0^{-1} \\ -\mathbf{E}_2 + \mathbf{E}_1 \mathbf{E}_0^{-1} \mathbf{E}_1^T & -(\mathbf{E}_1 \mathbf{E}_0^{-1} - 0.5 \mathbf{I}) \end{bmatrix} \quad (3.13)$$

Using eigenvalue decomposition, the matrix  $\mathbf{Z}$  can be decoupled into pairs of eigenvalues  $\lambda_i$  and  $-\lambda_i$ , and the standard eigenvalue problem necessary for a polyhedron is formulated as

$$\mathbf{Z} \begin{bmatrix} \psi_u \\ \psi_q \end{bmatrix} = \begin{bmatrix} \psi_u \\ \psi_q \end{bmatrix} \mathbf{S}_n \quad (3.14)$$

where the  $\mathbf{S}_n$  is a diagonal matrix and the entries are composed of the real parts of the eigenvalues. The parameters  $\psi_u$  and  $\psi_q$  are modal displacements and forces, respectively, depending on the DOFs of the surface elements. For a bounded polyhedron, the interpolation function in the radial direction and the internal force vector can be expressed as

$$\mathbf{u}(\xi) = \psi_u \xi^{-S_n-0.5} \psi_u^{-1} \mathbf{u}_b \quad (3.15a)$$

$$\mathbf{q}(\xi) = \psi_q \xi^{-S_n-0.5} \psi_u^{-1} \mathbf{u}_b \quad (3.15b)$$

Finally, the displacement field  $\mathbf{u}(\xi, \xi_1, \xi_2)$  for the polyhedral element is evaluated by substituting Eq. (3.15a) into Eq. (3.7a), and the formulation can be expressed in terms of  $\mathbf{u}_b$ :

$$\mathbf{u}(\xi, \xi_1, \xi_2) = (N^u(\xi_1, \xi_2) \psi_u \xi^{-(0.5+S_n)} \psi_u^{-1}) \mathbf{u}_b \quad (3.16)$$

Therefore, the polyhedral element shape function  $\Phi(\xi, \xi_1, \xi_2)$  formulated for the SBFEM is extracted as

$$\Phi(\xi, \xi_1, \xi_2) = N^u(\xi_1, \xi_2) \psi_u \xi^{-(0.5+S_n)} \psi_u^{-1} \quad (3.17)$$

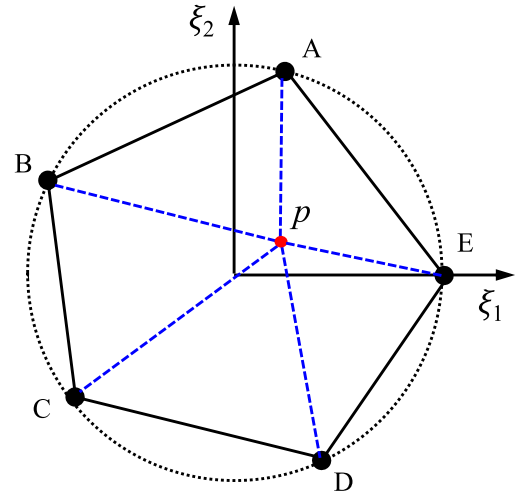
Additionally, the stiffness matrix  $\mathbf{K}$  can be derived from Eq. (3.15) and is formulated as

$$\mathbf{K} = \psi_q \psi_u^{-1} \quad (3.18)$$

The polyhedral element shape functions can be readily computed once the polygonal shape function is known. More importantly, all that is needed is the calculated value of the polygonal shape function and the corresponding derivatives, which are calculated at the integral points instead of using the expression, making the computation simpler.

In this paper, the value of the polygonal shape function  $\mathbf{N}$  and the corresponding derivatives  $\mathbf{LDu}$  were calculated using an external MATLAB code, which significantly simplified the expression of the polyhedral element shape functions  $\Phi(\xi, \xi_1, \xi_2)$ , the stiffness matrix  $\mathbf{K}$ , and the coefficient matrix used in the SBFEM. Therefore, the presented polyhedral element does not require a complex integration expression and the computational efficiency would be enhanced compared with other polyhedral FEMs.

Here, a few calculated values in the shape functions and derivatives  $\mathbf{LDu}$  are mentioned to provide the reader with a general understanding. A convex five-node canonical element was adopted as an example with the following nodal coordinates (4-digit precision): (0.3090, 0.9511), (−0.8090, 0.5878), (−0.8090, −0.5878), (0.3090, −0.9511) and (1, 0). The mean-value coordinates of all the evaluated points are given in Table 2, and a representative point  $p$  with coordinates (0.2182, 0.1585)

**Fig. 8.** Mean-value coordinates at point  $p$ .**Table 3**

Value of the mean-value shape function at the evaluated points (4-digit precision).

Points	A	B	C	D	E
1	0.2200	0.0268	0.0239	0.0551	0.6741
2	0.2936	0.1501	0.1126	0.1501	0.2936
3	0.6741	0.0551	0.0239	0.0268	0.2200
4	0.6741	0.2200	0.0268	0.0239	0.0551
5	0.2936	0.2936	0.1501	0.1126	0.1501
6	0.2200	0.6741	0.0551	0.0239	0.0268
7	0.0157	0.7283	0.2393	0.0089	0.0079
8	0.1501	0.2936	0.2936	0.1501	0.1126
9	0.0089	0.2393	0.7283	0.0157	0.0079
10	0.0239	0.0551	0.6741	0.2200	0.0268
11	0.1126	0.1501	0.2936	0.2936	0.1501
12	0.0239	0.0268	0.2200	0.6741	0.0551
13	0.0268	0.0239	0.0551	0.6741	0.2200
14	0.1501	0.1126	0.1501	0.2936	0.2936
15	0.0551	0.0239	0.0268	0.2200	0.6741

is plotted in Fig. 8. Then, the domain is triangulated with  $p$  as an internal point. The polygonal shape functions and derivatives can be computed by substituting these known variables, and the results are listed in Tables 3 and 4.

### 3.5. Strain and stress fields

The radial analytical displacement function  $\mathbf{u}(\xi)$  can be obtained from the aforementioned derivation. By combining the strain field formulation provided by Wolf [24], the strain field of a polyhedron can be written as

$$\epsilon(\xi, \xi_1, \xi_2) = (\mathbf{B}_1(\xi_1, \xi_2) \psi_u (\mathbf{S}_n - 0.5) \xi^{-(1.5+S_n)} \psi_u^{-1}) \mathbf{u}_b + \left( \frac{1}{\xi} \mathbf{B}_2(\xi_1, \xi_2) \psi_u \xi^{-(0.5+S_n)} \psi_u^{-1} \right) \mathbf{u}_b \quad (3.19)$$

Using Hooke's law, the stress field  $\sigma$  can be written as

$$\sigma = \mathbf{D} \epsilon \quad (3.20)$$

where  $\mathbf{D}$  is the elastic constitutive matrix. Substituting Eq. (3.19) (which is formulated for the strain field  $\epsilon$ ), the stress field can be immediately obtained.

## 4. Scaled boundary polyhedron procedure

The PSBFEM3D is implemented using object-oriented programming with Visual C++ based on the Windows program GEODYNA [49]; the

**Table 4**  
Values of the derivatives at the evaluated points (4-digit precision).

Points	Gradients	A	B	C	D	E	
1	<b>LDu</b>	$N_i$	0.2200	0.0268	0.0239	0.0551	0.6741
		$\partial N_i/\partial \xi_1$	-0.3064	-0.1686	-0.1444	-0.3214	0.9408
		$\partial N_i/\partial \xi_2$	0.7238	-0.0724	-0.0809	-0.3224	-0.2480
2	<b>LDu</b>	$N_i$	0.2936	0.1501	0.1126	0.1501	0.2936
		$\partial N_i/\partial \xi_1$	0.0408	-0.3421	-0.2110	-0.0402	0.5524
		$\partial N_i/\partial \xi_2$	0.5676	0.0689	-0.1533	-0.3466	-0.1366
3	<b>LDu</b>	$N_i$	0.6741	0.0551	0.0239	0.0268	0.2200
		$\partial N_i/\partial \xi_1$	0.0548	-0.4060	-0.1216	-0.1209	0.5936
		$\partial N_i/\partial \xi_2$	0.9714	-0.2060	-0.1123	-0.1380	-0.5151
4	<b>LDu</b>	$N_i$	0.6741	0.2200	0.0268	0.0239	0.0551
		$\partial N_i/\partial \xi_1$	0.5266	-0.7830	0.0168	0.0323	0.2073
		$\partial N_i/\partial \xi_2$	0.8181	-0.0678	-0.1827	-0.1623	-0.4053
5	<b>LDu</b>	$N_i$	0.2936	0.2936	0.1501	0.1126	0.1501
		$\partial N_i/\partial \xi_1$	0.3006	-0.5272	-0.1712	0.0806	0.3172
		$\partial N_i/\partial \xi_2$	0.4832	0.2142	-0.3040	-0.2480	-0.1453
6	<b>LDu</b>	$N_i$	0.2200	0.6741	0.0551	0.0239	0.0268
		$\partial N_i/\partial \xi_1$	0.6733	-0.9069	0.0705	0.0693	0.0938
		$\partial N_i/\partial \xi_2$	0.4054	0.3523	-0.4498	-0.1503	-0.1577
7	<b>LDu</b>	$N_i$	0.0157	0.7283	0.2393	0.0089	0.0079
		$\partial N_i/\partial \xi_1$	0.3925	-0.5415	-0.2399	0.2061	0.1829
		$\partial N_i/\partial \xi_2$	0.0154	0.8293	-0.8316	-0.0095	-0.0037
8	<b>LDu</b>	$N_i$	0.1501	0.2936	0.2936	0.1501	0.1126
		$\partial N_i/\partial \xi_1$	0.2362	-0.3666	-0.3666	0.2362	0.2608
		$\partial N_i/\partial \xi_2$	0.2568	0.4352	-0.4352	-0.2568	0.0000
9	<b>LDu</b>	$N_i$	0.0089	0.2393	0.7283	0.0157	0.0079
		$\partial N_i/\partial \xi_1$	0.2061	-0.2399	-0.5415	0.3925	0.1829
		$\partial N_i/\partial \xi_2$	0.0095	0.8316	-0.8293	-0.0154	0.0037
10	<b>LDu</b>	$N_i$	0.0239	0.0551	0.6741	0.2200	0.0268
		$\partial N_i/\partial \xi_1$	0.0693	0.0705	-0.9069	0.6733	0.0938
		$\partial N_i/\partial \xi_2$	0.1503	0.4498	-0.3523	-0.4054	0.1577
11	<b>LDu</b>	$N_i$	0.1126	0.1501	0.2936	0.2936	0.1501
		$\partial N_i/\partial \xi_1$	0.0806	-0.1712	-0.5272	0.3006	0.3172
		$\partial N_i/\partial \xi_2$	0.2480	0.3040	-0.2142	-0.4832	0.1453
12	<b>LDu</b>	$N_i$	0.0239	0.0268	0.2200	0.6741	0.0551
		$\partial N_i/\partial \xi_1$	0.0323	0.0168	-0.7830	0.5266	0.2073
		$\partial N_i/\partial \xi_2$	0.1623	0.1827	0.0678	-0.8181	0.4053
13	<b>LDu</b>	$N_i$	0.0268	0.0239	0.0551	0.6741	0.2200
		$\partial N_i/\partial \xi_1$	-0.1209	-0.1216	-0.4060	0.0548	0.5936
		$\partial N_i/\partial \xi_2$	0.1380	0.1123	0.2060	-0.9714	0.5151
14	<b>LDu</b>	$N_i$	0.1501	0.1126	0.1501	0.2936	0.2936
		$\partial N_i/\partial \xi_1$	-0.0402	-0.2110	-0.3421	0.0408	0.5524
		$\partial N_i/\partial \xi_2$	0.3466	0.1533	-0.0689	-0.5676	0.1366
15	<b>LDu</b>	$N_i$	0.0551	0.0239	0.0268	0.2200	0.6741
		$\partial N_i/\partial \xi_1$	-0.3214	-0.1444	-0.1686	-0.3064	0.9408
		$\partial N_i/\partial \xi_2$	0.3224	0.0809	0.0724	-0.7238	0.2480

PSBFEM3D program was developed by the first author. Multi-core parallel and GPU technology were used in the GEODYNA program, which provides the computational capacity to handle large-scale analysis, including millions of DOFs.

## 5. Numerical example

This section describes the numerical simulations used to evaluate the performance of polyhedral discretisation using the coupled SBFEM. This section is divided into three subsections. First, a shear-loaded beam is investigated using general polyhedral elements. Second, a beam with pure torsion forces is similarly investigated. Third, an octree grid is investigated through repeated cantilever numerical studies, and three complicated structural simulations are performed.

### 5.1. Shear-loaded beam

A cantilever beam loaded in shear was studied to evaluate the performance of PSBFEM3D; the domain  $R$  for this problem was

$(-1, 1) \times (-1, 1) \times (0, L)$ , as illustrated in Fig. 9. The load was constant tractions given by  $S = [0, -F, 0]^T$  on the free surface. To compare the performance of our method with findings from the literature, we used the expressions for stresses available in Barber [50], which are written as

$$\sigma_{11} = \sigma_{22} = \sigma_{12} = 0 \quad (5.1a)$$

$$\sigma_{33} = \frac{3F}{4}xz \quad (5.1b)$$

$$\sigma_{31} = \frac{3Fv}{2\pi^2(1+v)} \sum_{n=1}^{\infty} \frac{(-1)^n}{n^2 \cosh(n\pi)} \sin(n\pi x) \sinh(n\pi y) \quad (5.1c)$$

$$\sigma_{23} = \frac{3F(1-y^2)}{8} + \frac{Fv(3x^2-1)}{8(1+v)} - \frac{3Fv}{2\pi^2(1+v)} \sum_{n=1}^{\infty} \frac{(-1)^n}{n^2 \cosh(n\pi)} \cos(n\pi x) \cosh(n\pi y) \quad (5.1d)$$

The displacement fields corresponding to these stresses up to rigid body motion are

$$u_1 = -\frac{3Fv}{4E}xyz \quad (5.2a)$$

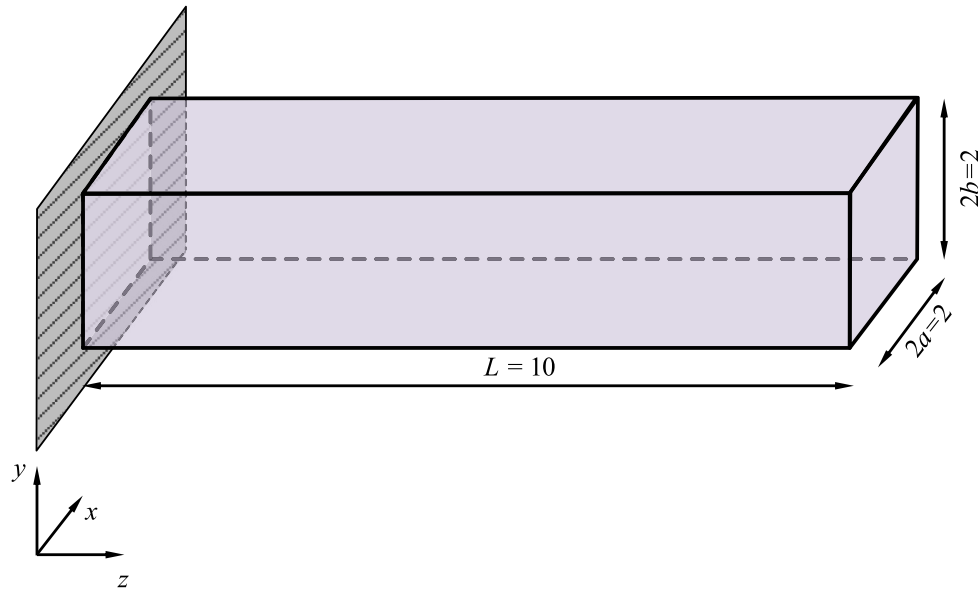


Fig. 9. Shear-loaded beam of length  $L$  with a rectangular cross section of width  $2a$  and height  $2b$ .

**Table 5**  
Relative error for a beam under a shear load at different mesh levels (3-digit precision).

Mesh	Elements	Nodes	Faces	Displacement(m)	Relative error ( $ u-u^* /u^*$ )
Mesh (a)	91	440	528	1.037	0.028
Mesh (b)	236	1232	1465	1.024	0.015
Mesh (c)	367	1845	2209	1.015	0.006
Mesh (d)	842	4394	5233	1.010	0.001

**Table 6**  
Relative error for a beam under torsion at different mesh levels (3-digit precision).

Mesh	Elements	Nodes	Faces	Displacement(m)	Relative error ( $ u-u^* /u^*$ )
Mesh (a)	91	440	528	-0.114	0.092
Mesh (b)	236	1232	1465	-0.108	0.036
Mesh (c)	367	1845	2209	-0.105	0.014
Mesh (d)	842	4394	5233	-0.104	0.004

$$u_2 = \frac{F}{8E} [3vz(x^2 - y^2) - z^3] \quad (5.2b)$$

$$u_3 = \frac{F}{8E} [3yz^2 + vy(y^2 - 3x^2)] + \frac{2(1+\nu)}{E} z(y) \quad (5.2c)$$

where  $z(y)$  is the anti-derivative of  $\sigma_{23}$  with respect to  $y$ . In the present numerical study, the length of the beam was 10, the shear load was taken as  $F=0.1$ , and the material was taken to be homogeneous and isotropic with a Young's modulus of  $E=25$  and a Poisson's ratio of  $\nu=0.3$ .

The beam is fixed at one end, where the coordinate  $z$  is zero. The loading condition due to an end shear load is considered to capture the universality of the proposed method. The mesh type is an arbitrary polyhedral mesh, and four levels of mesh refinement are considered for the grid, as shown in Fig. 10. Fig. 11 shows a cutaway view and a few representative polyhedral elements of one grid level. The number of elements, nodes and faces are listed in Table 5.

Fig. 12 shows the displacements in the  $y$ -direction for the four levels of mesh refinement. The deformation pattern was consistent for all four mesh levels. At the finest mesh level, the displacements were closest to the theoretical asymptotic value. The relative errors of the displacements in the  $y$ -direction are listed in Table 3, where  $u$  is the calculated value and  $u^*$  is the theoretical asymptotic value. The error decreased with an increasing number of elements. This result demonstrates that the presented method is reasonable and verifies its accuracy.

## 5.2. Pure torsion

To investigate the applicability of the PSBFEM3D for simulating elastic torsion problems, numerical studies were conducted considering the deformation of the same beam in a torsion state. The analytical displacements for this problem up to rigid body motion are given by [50]

$$u_1 = -\beta yz \quad (5.3a)$$

$$u_2 = -\beta zx \quad (5.3b)$$

$$u_3 = \beta \left( xy + \sum_{n=1}^{\infty} \frac{32(-1)^n}{\pi^3(2n-1)^3 \cosh[(2n-1)\pi/2]} \times \sin[(2n-1)\pi x/2] \sinh[(2n-1)\pi y/2] \right) \quad (5.3c)$$

where  $\beta$  is the twist per unit length, which is proportional to the applied torque. In this numerical study,  $\beta=0.1$ , and the displacement boundary conditions were applied to the face at  $z=0$ .

Fig. 13 shows the beam deformation and displacements in the  $y$ -direction generated under the torsional load for different mesh levels.

The relative errors of the displacements in the  $y$ -direction are listed in Table 6; this table shows that the error decreased with an increasing number of elements.

Compared with standard elements, which are widely used in computational fluid dynamics, the polyhedral elements are more flexible for modelling and offer many advantages. However, these cells are clearly not allowed in the existing SBFEM.

## 5.3. Octree grid investigation

The octree is a hierarchical tree-like data structure used to quickly generate a high-quality structural mesh. Due to the limited number of patterns in the octree decomposition, the stiffness and mass matrices of these cubic octants can be pre-computed and stored for quick retrieval. The stiffness and mass matrices of two similar cubic octants with different sizes are simply scalar multiples. This is a powerful technique for a rapid, adaptive design-to-analysis simulation for engineering structures. The readers are referred to the literature [27,39] for more detailed descriptions.



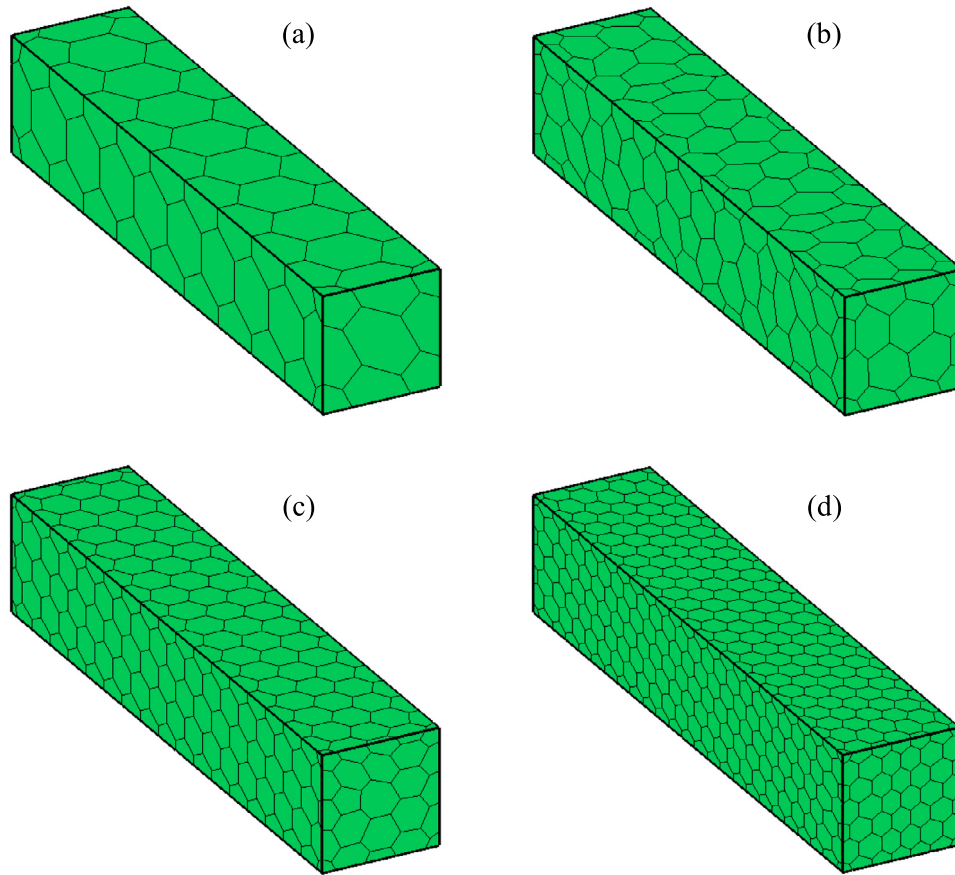


Fig. 10. Four levels of polyhedral grid refinement.

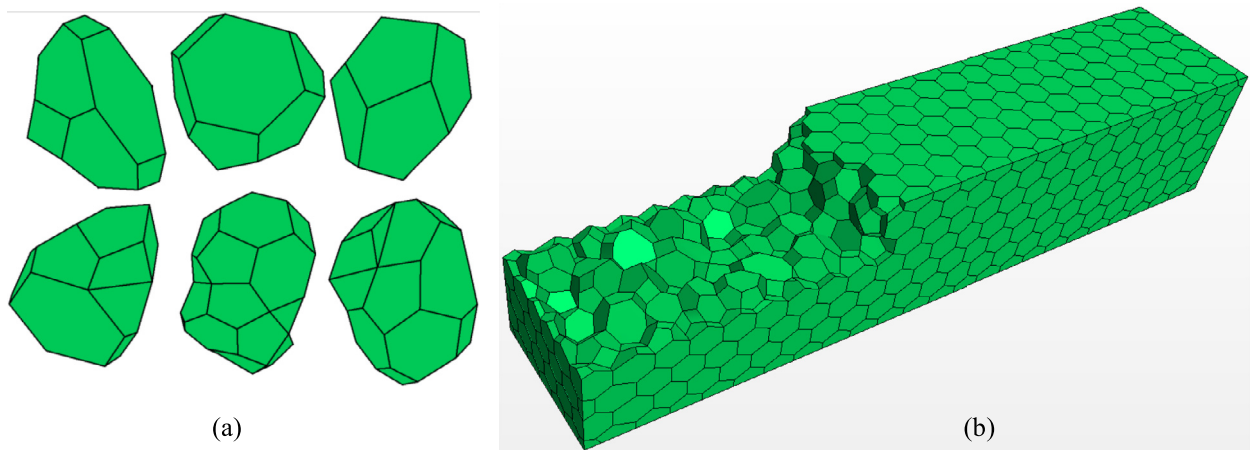


Fig. 11. One level of a polyhedral grid: (a) some representative polyhedrons and (b) a cutaway view.

### 5.3.1. Cantilever beam

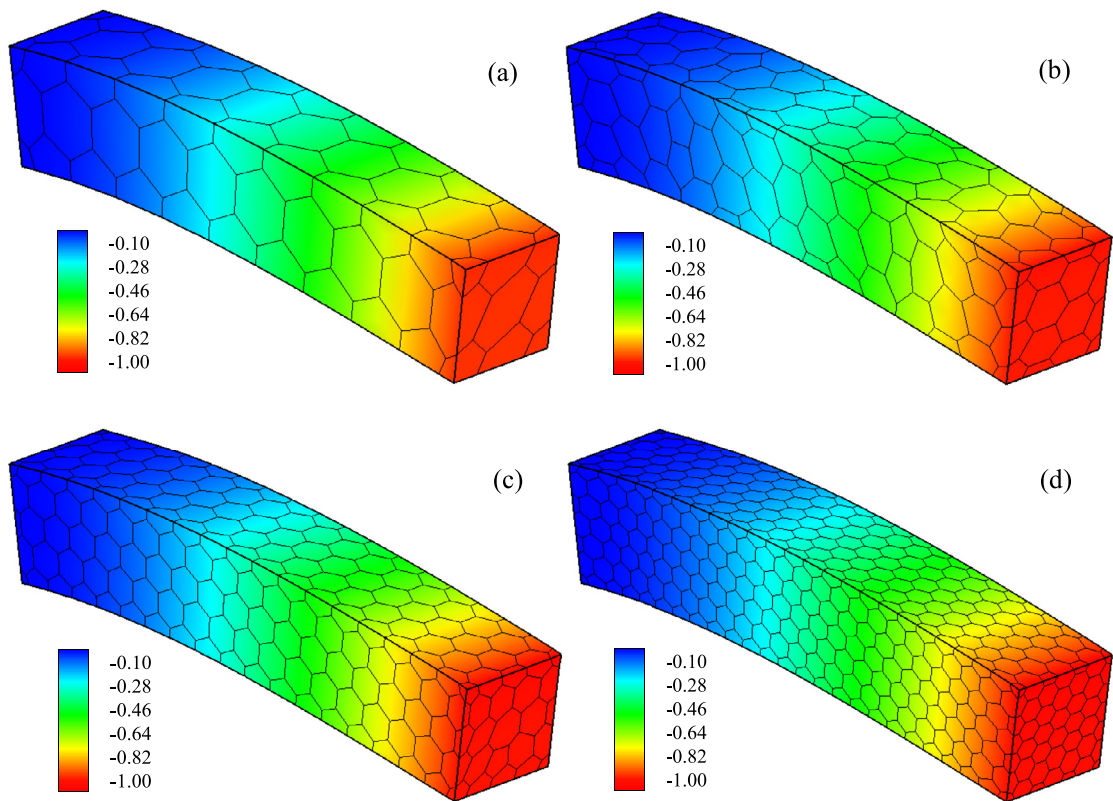
Mean-value polygonal shape functions are also applicable to an octree mesh with hanging nodes. In this section, the applicability of the octree mesh is verified by repeating the previous simulations. A similar study was conducted considering elements with hanging nodes, which lead to a total of 2,056 elements, 3,023 nodes, and 7,100 faces. The global discretisation is illustrated in Fig. 14, and a corresponding cut-away view is shown in Fig. 15.

Fig. 16 shows a high level of agreement, indicating that the relative error was 0.003 in the shear-loaded bending test and 0.004 in the pure torsion state. Thus, the proposed method performs well in the simulation using an octree mesh. In addition, the boundary surfaces with hanging

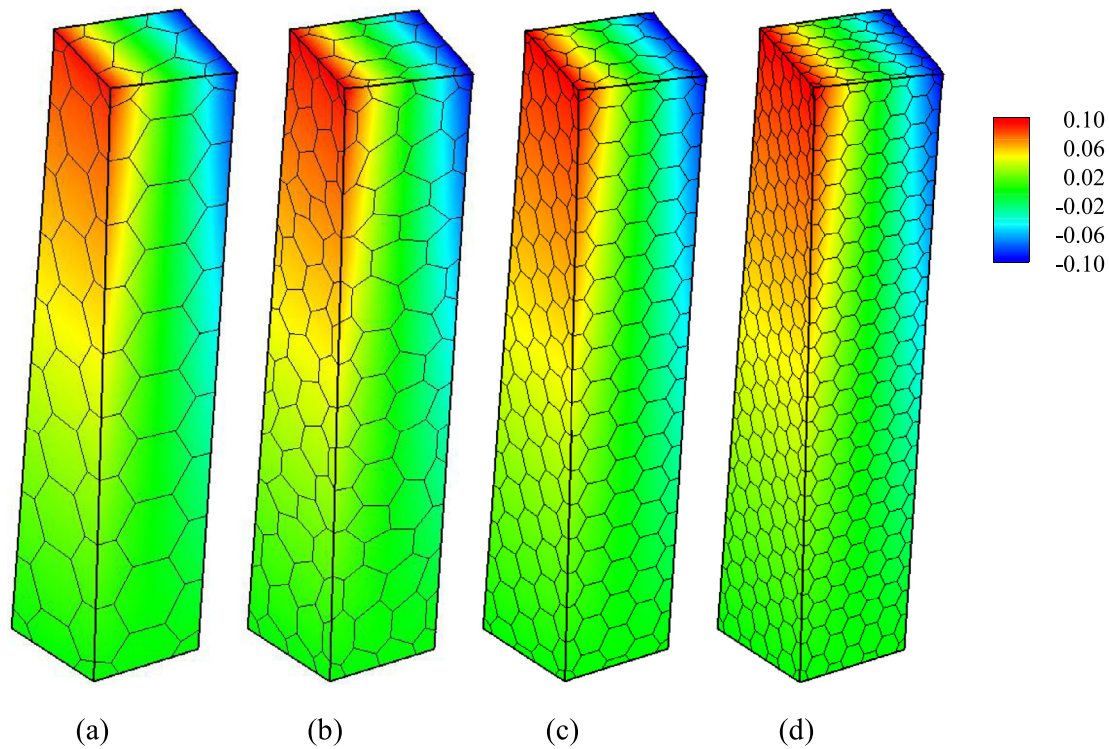
nodes do not need to be addressed because an arbitrary number of vertices are allowed in the face elements. Therefore, the proposed method is more advantageous than the existing SBFEM in a rapid adaptive analysis using octree mesh.

### 5.3.2. Sculptures of Terra-Cotta Warriors

The next two examples are sculptures that are fixed on the bottom, for which only the self-weight is considered. The first example is a 3D Studio Max model of the Terra-Cotta Warriors sculptures that was found online [51]. The material properties of clay are considered to be the following: mass density of  $\rho = 2350 \text{ kg/m}^3$ , Young's modulus of  $E = 0.05 \text{ GPa}$  and Poisson's ratio of  $\nu = 0.3$ .



**Fig. 12.** Shear-loaded beam with a deformed shape. The colours indicate the magnitude of the displacement in the y-direction. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).



**Fig. 13.** Pure beam torsion with a deformed shape. The colours indicate the magnitude of the displacement in the y-direction. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

**Table 7**

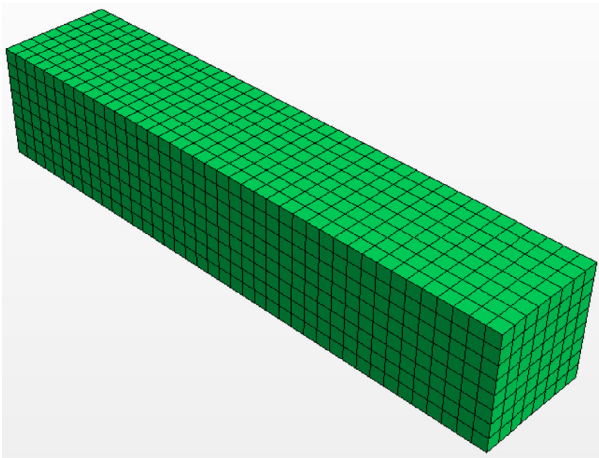
The element information statistics of the Terra-Cotta Warriors model using the existing SBFEM.

Shapes of boundary face elements	Faces	Additional required nodes	Additional required faces
Triangle	56,340	–	–
Quadrangle	440,260	–	–
Pentagon	36,833	36,833	277,028
Hexagon	10,384	10,384	77,685
Heptagon	148	0	767
Octagon	19	19	86
Sum	–	47,236	355,566

**Table 8**

A final element information comparison of this model using the two methods.

Items	Presented SBFEM	VS	Existing SBFEM	Reduction	Reduction percentage (%)
Elements	161,383		161,383	0	0
Nodes	215,186		262,422	47,236	18.00
Faces	543,984		899,550	355,566	39.53
DOF	645,558		787,266	141,708	18.00

**Fig. 14.** Global discretisation of the beam using an octree grid.

The model can be easily discretised using an octree generator, as illustrated in Fig. 17. The sculpture dimensions are  $260 \times 1200 \times 295 \text{ mm}^3$ . The maximum and minimum element sizes are set as 32 mm and 2 mm, respectively. The time spent on this discretisation is only 52.12 s. The element size transition inside the octree mesh is shown in Fig. 18, and a partial detail is given. Compared with the

existing SBFEM, no treatments are required for the hanging nodes, and a contrast diagram is presented.

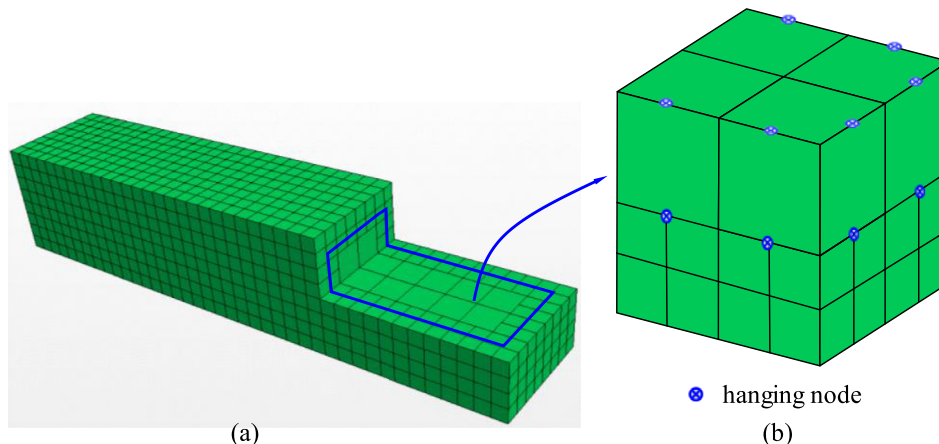
The comparison in Table 1 shows that the modified SBFEM offers more advantages and superior performance than the existing SBFEM. Table 7 presents a statistical table of the discretised model. As shown in this table, the vast majority of the boundary faces are quadrangles, indicating that most of the elements are hexahedral or cubic; thus, high-quality elements can be guaranteed.

As previously mentioned, additional measures need to be taken for the boundary faces in the existing SBFEM when the element sides exceed four, resulting in additional DOFs and faces. Figs. 19 and 20 present schematic diagrams of both methods, and the modified method is shown to be more concise and superior. Table 8 shows that the model using the existing SBFEM has 141,708 DOFs and 355,566 faces. These values can be decreased by using the method presented in this paper, which somewhat improves the computational efficiency, especially for large-scale problems.

Additionally, the displacements along the x-, y- and z-directions are displayed in Fig. 21, where the largest displacement in the y-direction occurs in the back-left mode, indicating that the deformation pattern is retroverted along the left side as this part is hanging without a fulcrum.

### 5.3.3. Sculpture of Sioux Falls Church

The second example is a 3D printing model of Sioux Falls Church, which was downloaded from [52]. The dimensions of the church are  $1567 \times 2100 \times 2030 \text{ mm}^3$ . The maximum and minimum element sizes are selected as 100 mm and 32 mm, respectively, and 116.17 s is needed

**Fig. 15.** Representative octree meshing grid: (a) cutaway view and (b) partial detail with a hanging node.



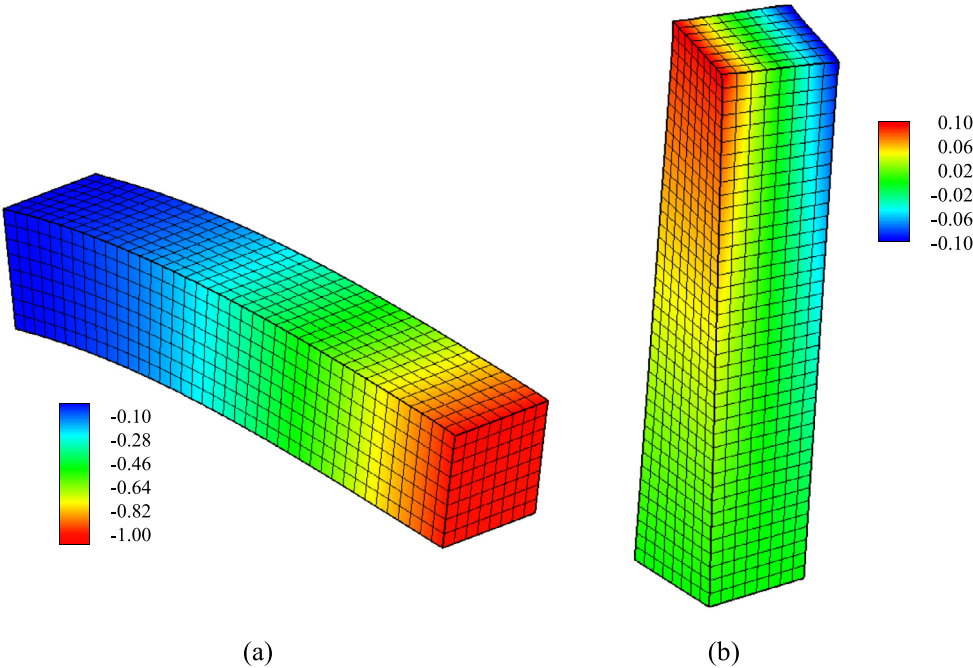


Fig. 16. Deformed shape with the octree meshing grid: (a) in a shear-loaded state and (b) in a pure torsion state.



Fig. 17. The octree polyhedral mesh for the Terra-Cotta Warriors model.

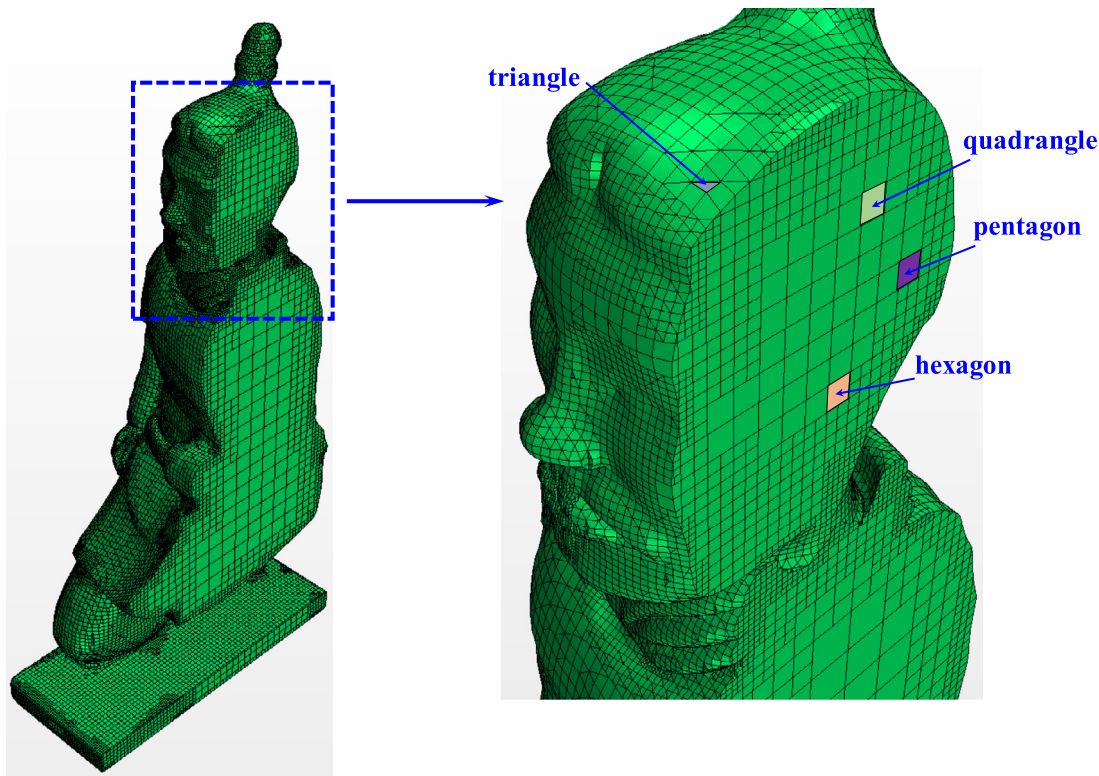


Fig. 18. The sectional drawing and partial detail of the model with polyhedral mesh.

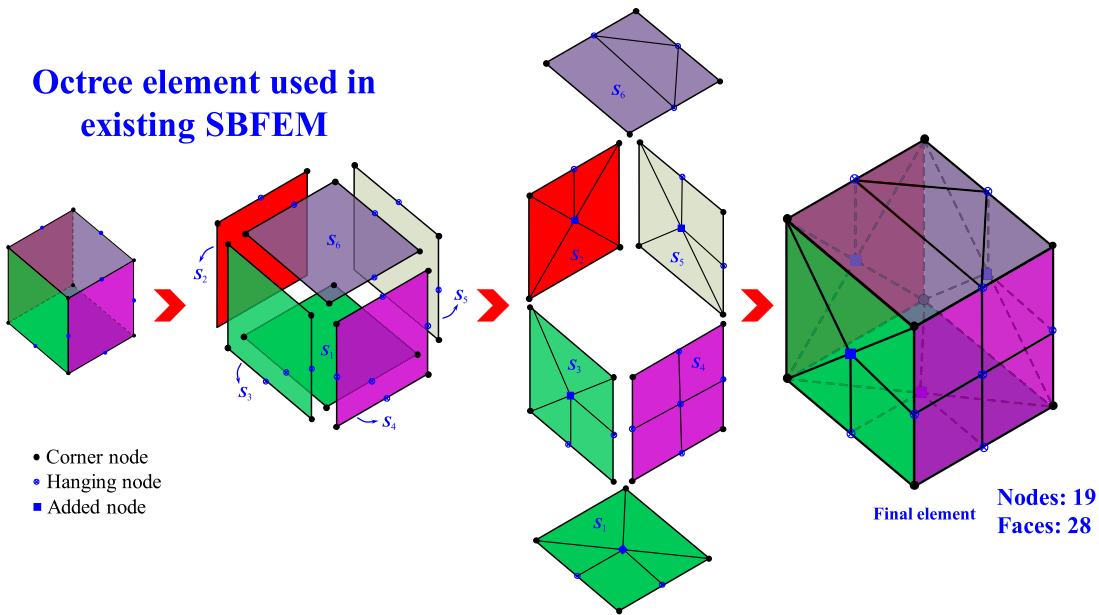


Fig. 19. The schematic diagram of an octree element with hanging nodes used in the existing SBFEM.

**Table 9**  
The element information statistics for the Sioux Falls Church model using the existing SBFEM.

Shapes of boundary face elements	Faces	Additional required nodes	Additional required faces
Triangle	39,043	–	–
Quadrangle	855,312	–	–
Pentagon	80,777	80,777	625,869
Hexagon	12,081	12,081	85,792
Heptagon	361	0	1,783
Octagon	44	44	207
Sum	–	92,902	713,653



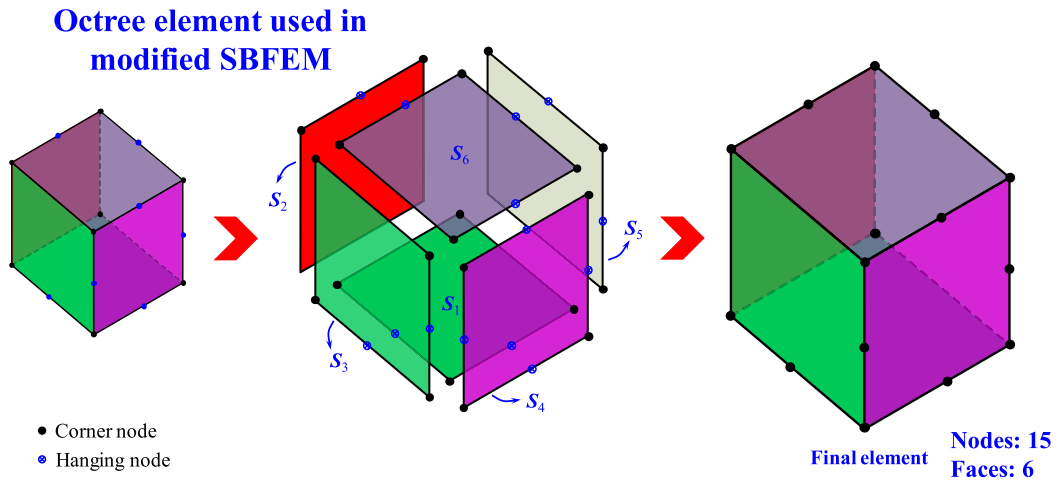


Fig. 20. The schematic diagram of an octree element with hanging nodes used in the modified SBFEM.

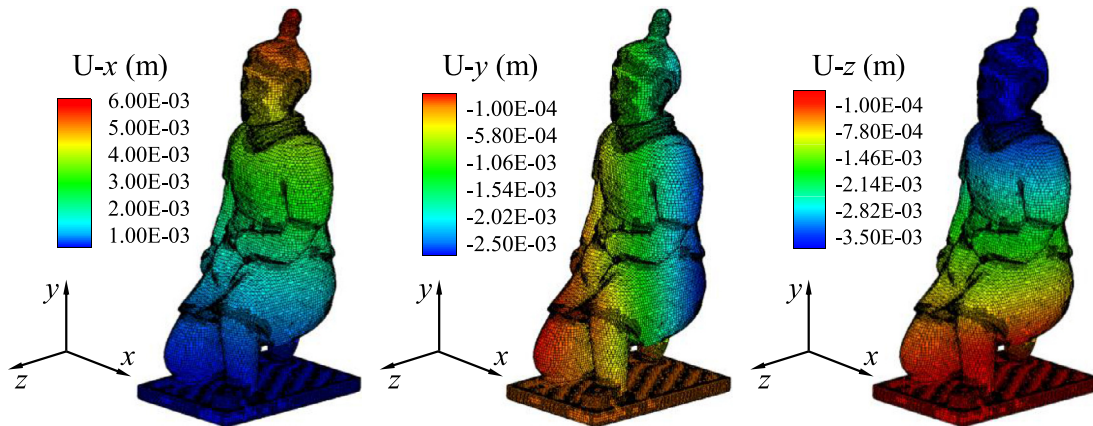


Fig. 21. The distribution of the displacements along three directions in the overall model.

Table 10

A final element information comparison of this model using the two methods.

Items	Presented SBFEM	VS	Existing SBFEM	Reduction	Reduction percentage (%)
Elements	286,322		286,322	0	0
Nodes	413,605		506,507	92,902	18.34
Faces	987,618		1,701,269	713,651	41.95
DOF	1,240,815		1,519,521	278,706	18.34

to generate all of the octree meshing. The detailed statistical and comparison data are listed in Tables 9 and 10. Clearly, the DOFs and boundary face elements can be greatly reduced by using the method proposed in this paper. The displacement along the y-direction of the model is presented in Fig. 22. This distribution is symmetrical in the x-direction and asymmetrical in the z-direction, matching the symmetry of the geometric shape.

Fig. 23 presents the entire distribution of normal stress along the y-direction using a deformed octree mesh with a deformation scale factor of 500. The maximum compressive stress appears at the bottom of the structure, which is in accordance with the general cognitive law.

To present the inner stress state, two slices with an x-coordinate of 0.4167 m and a z-coordinate of 1.6 m are introduced, as illustrated in Figs. 24 and 25, respectively. Due to the existence of non-uniform settlement in this model, the tensile stress emerges along the x-direction and z-direction for part of the roof, as shown in the two above illustrations.

#### 5.3.4. Nuclear power plant structure

The last example is a simulation of a nuclear power plant structure. As an introductory application, this paper considers only the elastic response. This example is subjected to self-weight, and only the concrete structure and a square foundation are considered in this simulation. The material properties of concrete are considered to be the following: mass density of  $\rho = 2600 \text{ kg/m}^3$ , Young's modulus of  $E = 34.5 \text{ GPa}$  and Poisson's ratio of  $\nu = 0.25$ , and the properties of soil are  $\rho = 1890 \text{ kg/m}^3$ ,  $E = 1.12 \text{ GPa}$  and Poisson's ratio of  $\nu = 0.4$ . The overall geometry is shown in Fig. 26, and interested readers can refer to the literature [53] for a more detailed description of this model. The maximum and minimum element dimensions are selected as 24 m and 0.38 m, respectively, and the total model takes 137.95 s to perform the mesh generation. A cutaway view and the major structure meshing are presented in Fig. 27, and exhaustive statistical information for the grid is given in Tables 11 and 12. As shown in these tables, using the proposed SBFEM can significantly decrease the global DOFs and boundary faces compared with the current SBFEM.

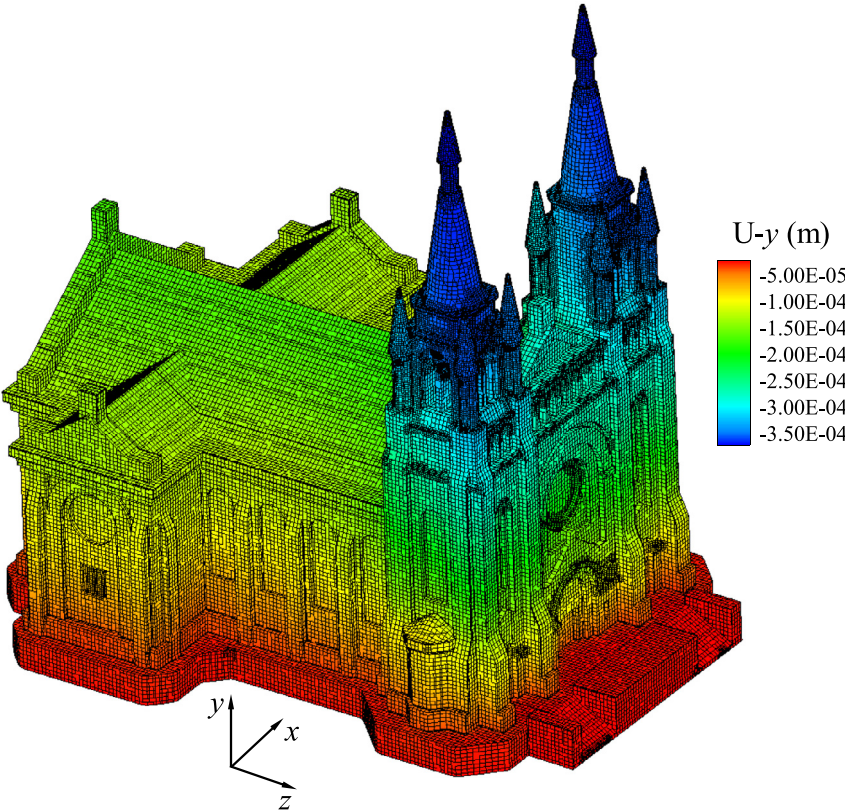


Fig. 22. The displacement along the y-direction of the whole model with the octree grid.

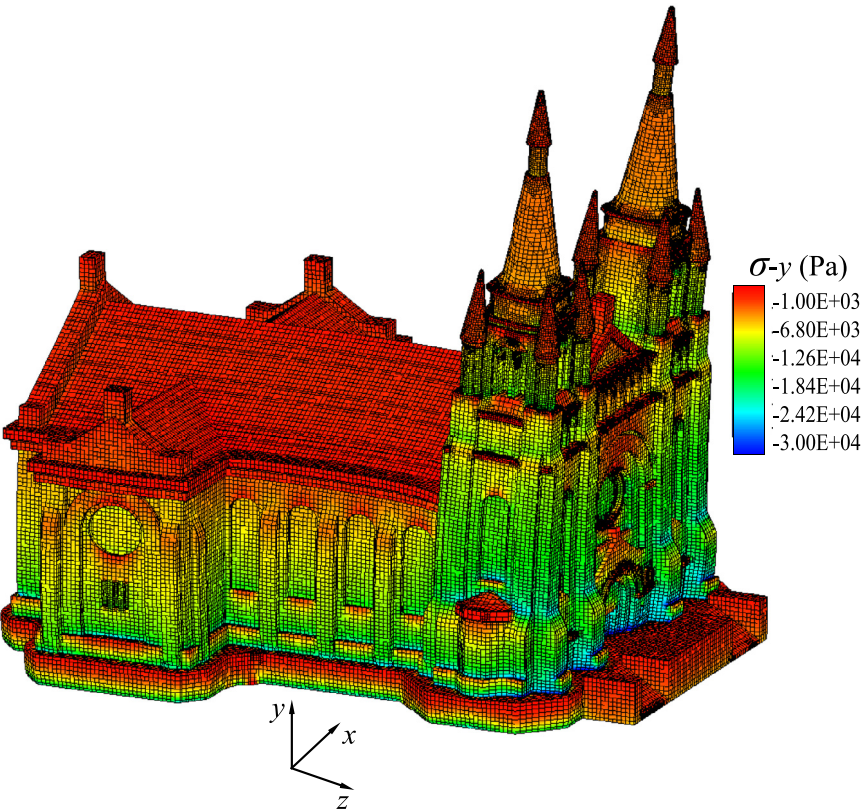


Fig. 23. The normal stress along the y-direction of the whole model with the deformed octree grid (the minus sign denotes compressive stress, and the deformation scale factor is 500).

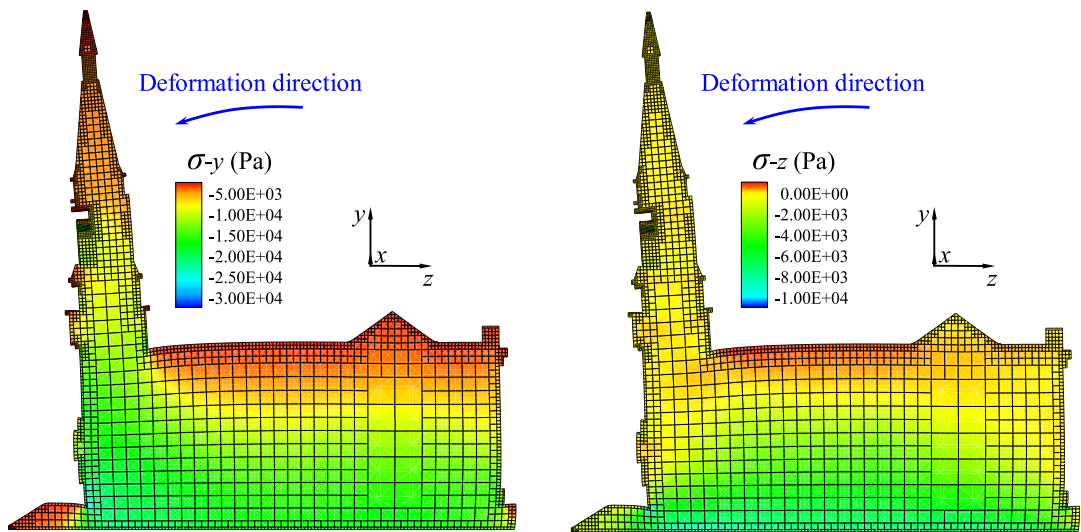


Fig. 24. The distribution of normal stress at the yz-slices with an x-coordinate of 0.4167 (the minus sign denotes compressive stress, and the deformation scale factor is 500).

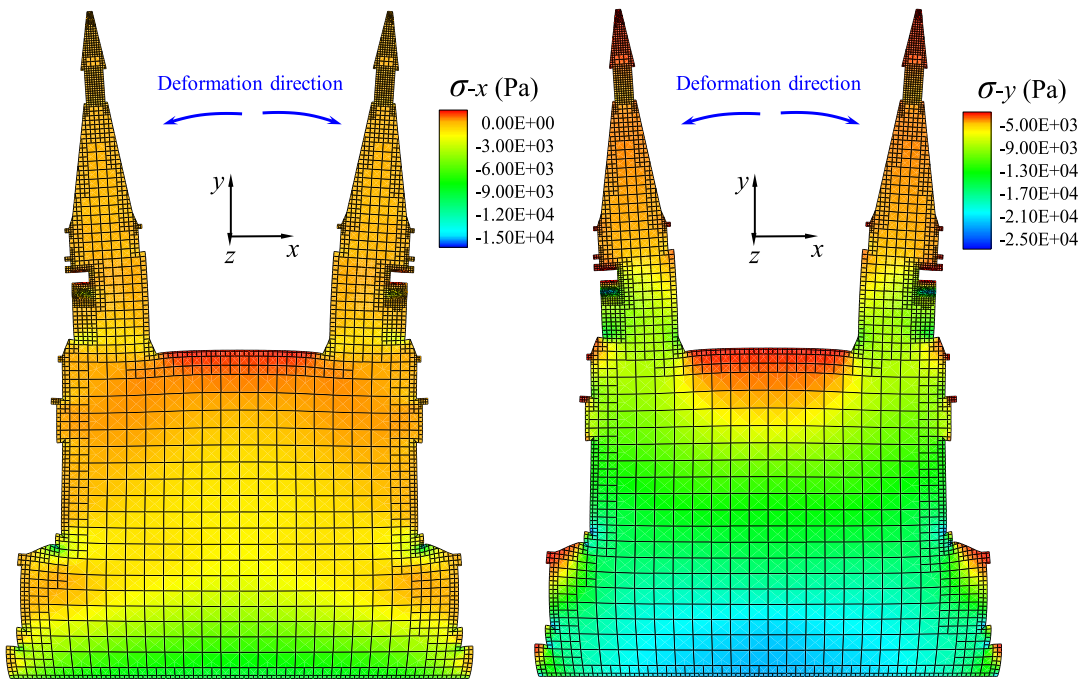


Fig. 25. The distribution of normal stress at the xy-slices with a z-coordinate of 1.6 (the minus sign denotes compressive stress, and the deformation scale factor is 500).

**Table 11**  
The element information statistics of the nuclear power plant model using the existing SBFEM.

Shapes of boundary face elements	Faces	Additional required nodes	Additional required faces
Triangle	82,277	–	–
Quadrangle	876,568	–	–
Pentagon	29,543	29,543	208,942
Hexagon	2,635	2,635	18,497
Heptagon	8	0	25
Octagon	0	0	0
Sum	–	32,178	490,348

**Table 12**  
A final element information comparison of this model using the two methods.

Items	Presented SBFEM	VS	Existing SBFEM	Reduction	Reduction percentage (%)
Elements	304,491		304,491	0	0
Nodes	376,359		408,537	32,178	7.88
Faces	991,031		1,040,379	490,348	47.13
DOF	1,129,077		1,225,611	96,534	7.88



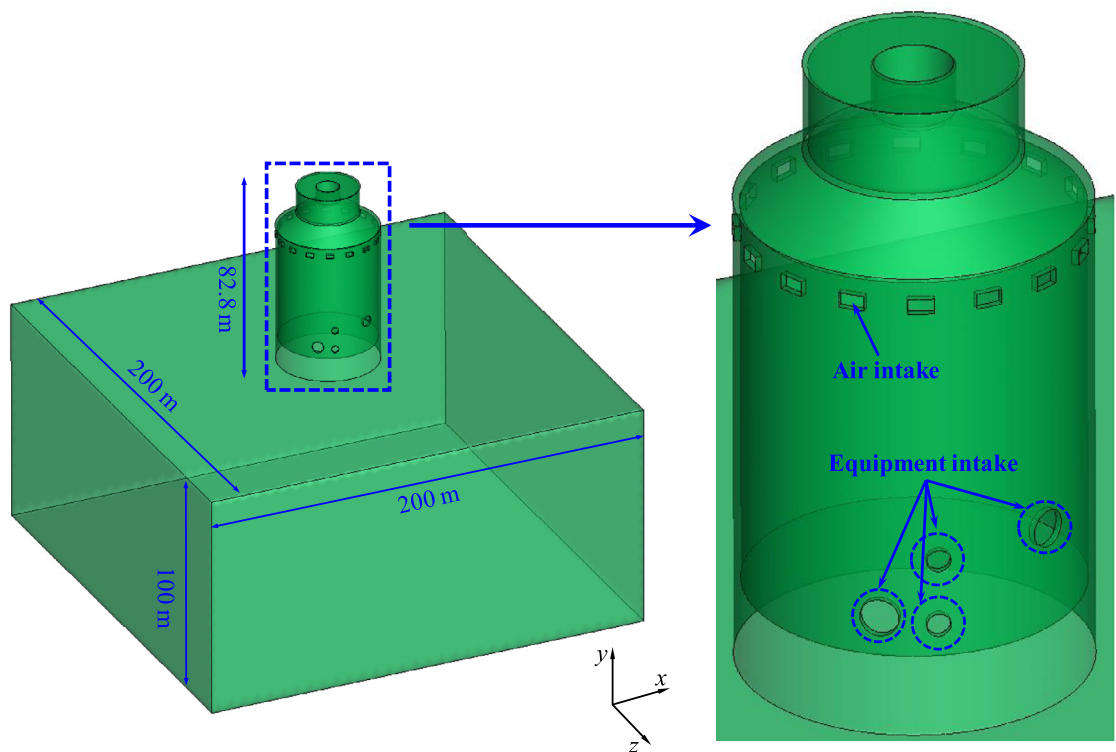


Fig. 26. Geometry of the nuclear power plant structure.

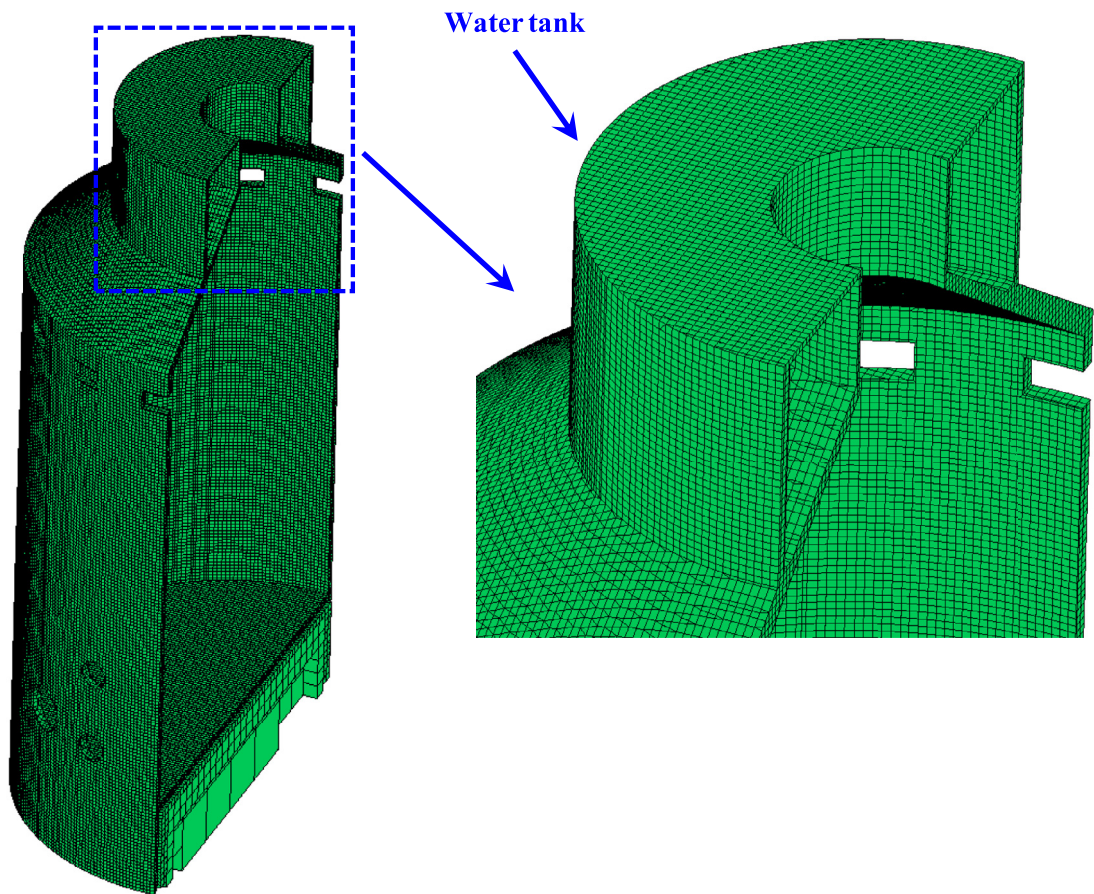


Fig. 27. Mesh inside the containment vessel and a detail of the water tank.

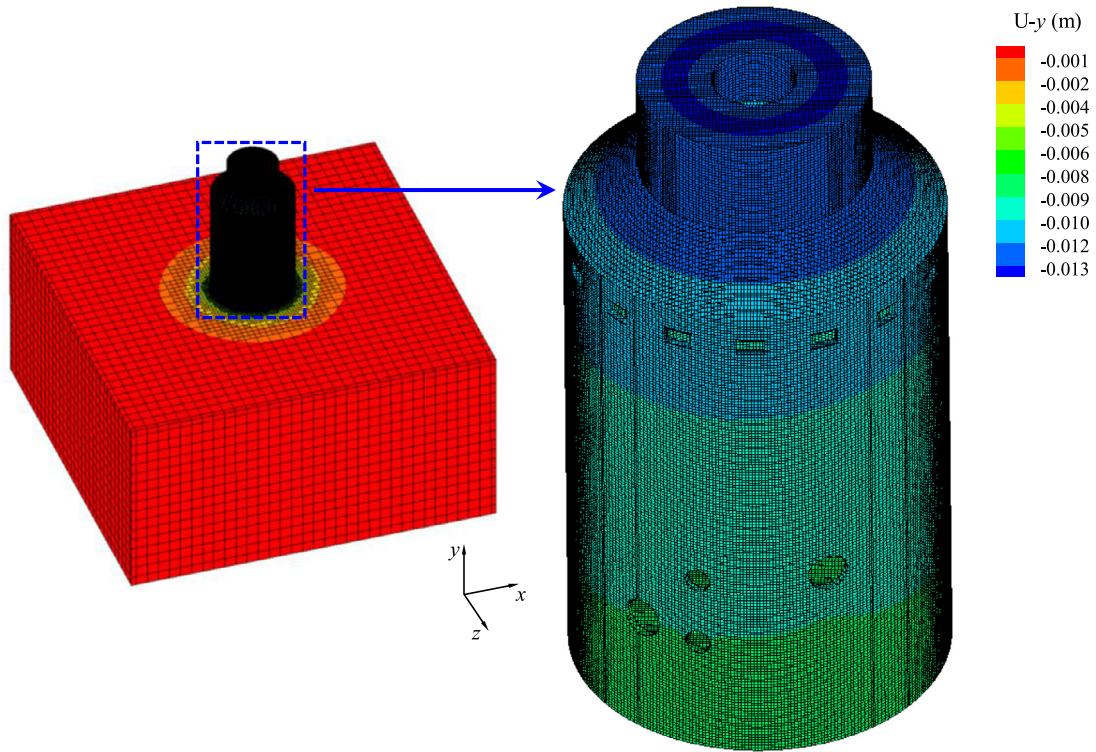


Fig. 28. The displacement along the y-direction for the whole model with the octree grid.

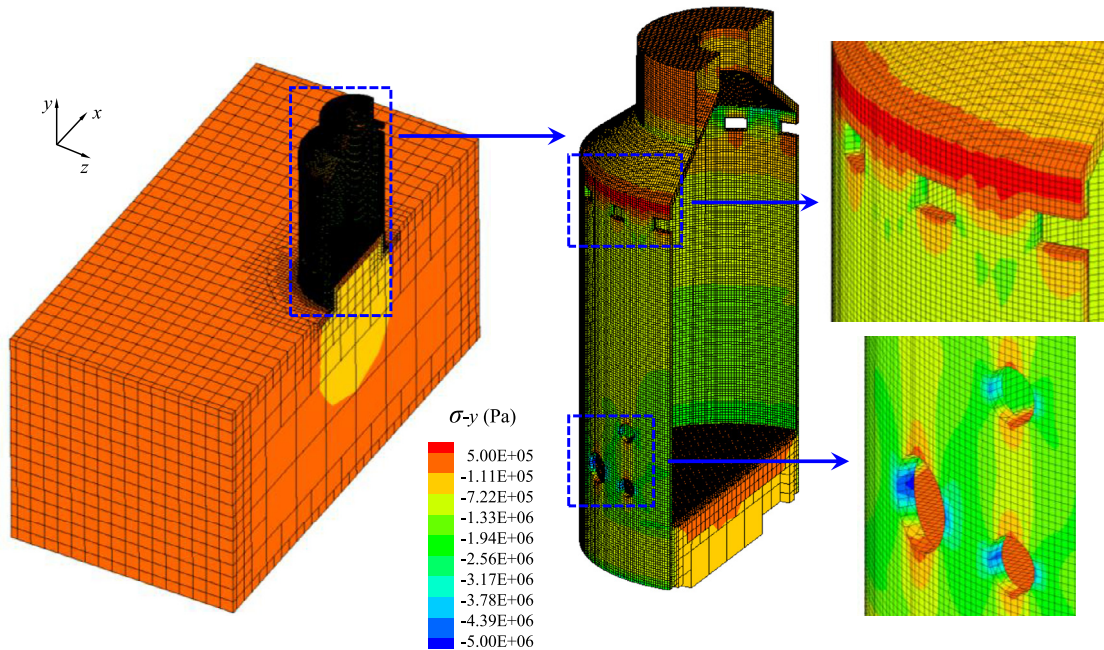


Fig. 29. The normal stress along the y-direction for half of the model with the octree grid.

Fig. 28 plots the displacement in the y-direction for the model, and the distribution is incompletely uniformly hierarchical at the equipment intake.

To visualise the results inside the domain, Fig. 29 plots the normal stress in the y-direction for the whole model. In addition, a corresponding deformed illustration is shown in Fig. 30. As shown in these figures, the structure located around the air intake is relatively weak, leading to partial tensile stress around the intake. Similarly, a stress concentration

usually occurs at the equipment intake, where the maximum compressive stress is observed.

## 6. Conclusions

This paper proposed a distinctive universal octree polyhedral scaled boundary finite element based on mean-value polygonal shape functions. This modelling approach adopts the SBFEM to accomplish the



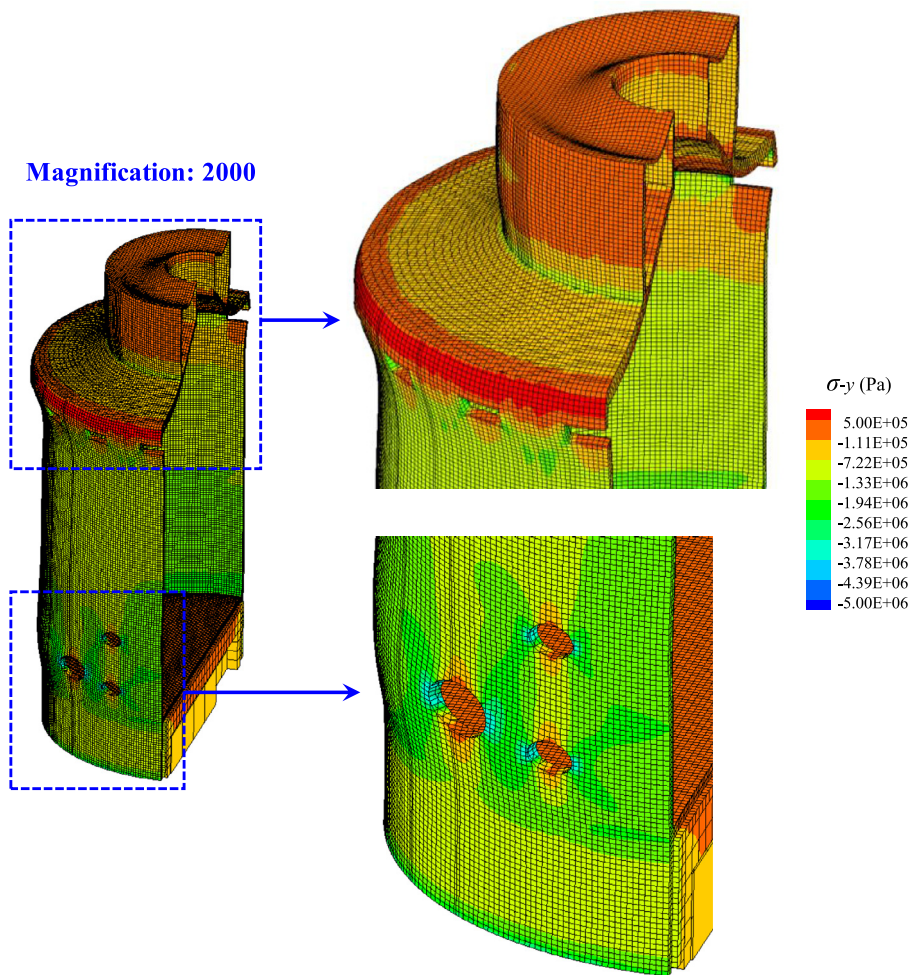


Fig. 30. The distribution of normal stress in the y-direction for the deformed mesh (the minus sign denotes compressive stress, and the deformation scale factor is 2000).

Gauss integration and assigns mean-value polygon functions to construct the shape function for the face elements. In this manner, the advantageous phenomenological nature of the SBFEM and flexible polyhedrons is exploited. Polygon face elements can be directly considered in each sub-cell without requiring additional treatments. This approach is first verified as being capable of predicting the performance of a beam under a shear load and under pure torsion. The simulations compare reasonably well with the available analytical solutions. Additionally, this method can be seamlessly combined with efficient octree meshing, which is more advantageous in a rapid adaptive modelling process. The three practical example applications, including two sculptures and a nuclear power plant structure, use octree discretisation technology to demonstrate that the proposed method can be applied to more complicated geometries. The three primary conclusions from the study are summarised below:

1. Although polyhedral elements are permissible in the existing SBFEM, the faces are restricted to only quadrangles or triangles, which could be relatively time consuming for complex geometries. Moreover, more nodes and boundary faces are required when polygon faces arise in a sub-cell. The proposed method accommodates more general polyhedral shapes, which completely eliminates the requirement for additional treatments and makes the method more general, succinct and flexible.
2. As arbitrary face element geometries are permitted, the enhanced method is completely free from addressing the hanging nodes on the boundary surfaces in rapid, adaptive simulations using the oc-

tree mesh. The proposed method is more advantageous and powerful in rapid-design analysis, multi-scale problems and adaptive local refinement simulations based on the octree generator.

3. When formulating the polyhedrons using the SBFEM framework, only polygonal shape functions are allowed for the boundary surface elements. Therefore, the mean-value polygon functions and gradients can be pre-computed for the Gauss points. Subsequently, the functions can be integrated into the PSBFEM3D program, which considers these functions to be constant variables. Therefore, the extremely complex formulation of the functions and their gradients is no longer required, which significantly simplifies the integration of the weak form of the governing equation and is less computationally intense.

In this paper, polyhedral elements are primarily demonstrated in the context of linear elasticity. It is necessary and meaningful for this approach to be expanded to nonlinear material fields, which should be an interesting direction for future work.

### Acknowledgements

This work was supported by National Key R&D Program of China (2017YFC0404900) and the National Natural Science Foundation of China (grant nos. 51421064, 51779034). This financial support is gratefully acknowledged.

## References

- [1] ANSYSFluent. <http://www.ansys.com>.
- [2] CD-adapco. [www.cd-adapco.com](http://www.cd-adapco.com).
- [3] Brezzi F, Lipnikov K, Simoncini V. A family of mimetic finite difference methods on polygonal and polyhedral meshes. *Math Models Methods Appl Sci* 2005;15(10):1533–51.
- [4] Brezzi F, Lipnikov K, Shashkov M, Simoncini V. A new discretization methodology for diffusion problems on generalized polyhedral meshes[J]. *Computer Methods in Applied Mechanics and Engineering* 2007;196(37):3682–92.
- [5] Brezzi F, Buffa A, Lipnikov K. Mimetic finite differences for elliptic problems. *ESAIM* 2009;43(2):277–95.
- [6] Da Veiga LB, Lipnikov K, Manzini G. Error analysis for a mimetic discretization of the steady Stokes problem on polyhedral meshes. *SIAM J Numer Anal* 2010;48(4):1419–43.
- [7] Da Veiga LB. A mimetic discretization method for linear elasticity. *ESAIM* 2010;44(2):231–50.
- [8] Talisch C, Paulino GH. Addressing integration error for polygonal finite elements through polynomial projections: a patch test connection. *Math Models Methods Appl Sci* 2014;24(08):1701–27.
- [9] Rashid MM, Selimotic M. A three-dimensional finite element method with arbitrary polyhedral elements. *Int J Numer Methods Eng* 2006;67(2):226–52.
- [10] Idelsohn S R, Onate E, Calvo N, Del Pin F. The meshless finite element method[J]. *International Journal for Numerical Methods in Engineering* 2003;58(6):893–912.
- [11] Wicke M, Botsch M, Gross M. A finite element method on convex polyhedra. *Comput Graphics Forum* 2007;26(3):355–64.
- [12] Martin S, Kaufmann P, Botsch M, Wicke M, Gross M. Polyhedral finite elements using harmonic basis functions[C]//*Computer Graphics Forum*. Blackwell Publishing Ltd 2008;27(5):1521–9.
- [13] Milbradt P, Pick T. Polytope finite elements. *Int J Numer Methods Eng* 2008;73(12):1811–35.
- [14] Wachspress EL. A rational finite element basis. Elsevier; 1975.
- [15] Warren J. Barycentric coordinates for convex polytopes. *Adv Comput Math* 1996;6:97–108.
- [16] Bishop JE. Simulating the pervasive fracture of materials and structures using randomly close packed Voronoi tessellations. *Comput Mech* 2009;44(4):455–71.
- [17] Yu J. Symmetric Gaussian quadrature formulae for tetrahedral regions. *Comput Methods Appl Mech Eng* 1984;43(3):349–53.
- [18] Ghosh S. The voronoi cell finite element method. New York: CRC Press; 2011.
- [19] Beirão da Veiga L, Brezzi F, Cangiani A, Manzini G, Marini L D, Russo A. Basic principles of virtual element methods[J]. *Mathematical Models and Methods in Applied Sciences* 2013;23(01):199–214.
- [20] Da Veiga LB, Brezzi F, Marini LD. Virtual elements for linear elasticity problems. *SIAM J Numer Anal* 2013;51(2):794–812.
- [21] Brezzi F, Marini LD. Virtual element methods for plate bending problems. *Comput Methods Appl Mech Eng* 2013;253:455–62.
- [22] Natarajan S, Bordas S, Ooi ET. Virtual and smoothed finite elements: a connection and its application to polygonal/polyhedral finite element methods. *Int J Numer Methods Eng* 2015;104(13):1173–99.
- [23] Wolf JP, Song CM. Finite-element modelling of unbounded media. Chichester: Wiley; 1996.
- [24] Wolf JP, Schanz M. The scaled boundary finite-element method. Chichester, UK: Wiley; 2004.
- [25] Goswami S, Becker W. Computation of 3-D stress singularities for multiple cracks and crack intersections by the scaled boundary finite element method. *Int J Fract* 2012;175(1):13–25.
- [26] Hell S, Becker W. The scaled boundary finite element method for the analysis of 3D crack interaction. *J Comput Sci* 2015;9:76–81.
- [27] Saputra AA, Birk C, Song C. Computation of three-dimensional fracture parameters at interface cracks and notches by the scaled boundary finite element method. *Eng Fract Mech* 2015;148:213–42.
- [28] Man H, Song C, Gao W, Tin-Loi F. A unified 3D-based technique for plate bending analysis using scaled boundary finite element method[J]. *International Journal for Numerical Methods in Engineering* 2012;91(5):491–515.
- [29] Lin G, Liu J, Li J, Hu Z. A scaled boundary finite element approach for sloshing analysis of liquid storage tanks. *Eng Anal Boundary Elem* 2015;56:70–80.
- [30] Xu H, Zou D, Kong X, Hu Z. Study on the effects of hydrodynamic pressure on the dynamic stresses in slabs of high CFRD based on the scaled boundary finite-element method[J]. *Soil Dynamics and Earthquake Engineering* 2016;88:223–36.
- [31] Birk C, Behnke R. A modified scaled boundary finite element method for three-dimensional dynamic soil-structure interaction in layered soil. *Int J Numer Methods Eng* 2012;89(3):371–402.
- [32] Liu J, Lin G. A scaled boundary finite element method applied to electrostatic problems. *Eng Anal Boundary Elem* 2012;36(12):1721–32.
- [33] Liu J, Zhang P, Lin G, Wang W, Lu S. Solutions for the magneto-electro-elastic plate using the scaled boundary finite element method[J]. *Engineering Analysis with Boundary Elements* 2016;68:103–14.
- [34] Liu J, Lin G, Li J. Short-crested waves interaction with a concentric cylindrical structure with double-layered perforated walls. *Ocean Eng* 2012;40:76–90.
- [35] Lu S, Liu J, Lin G, Wang W. Time-domain analyses of the layered soil by the modified scaled boundary finite element method[J]. *Structural Engineering and Mechanics* 2015;55(5):1055–86.
- [36] Li J, Shi Z, Ning S. A two-dimensional consistent approach for static and dynamic analyses of uniform beams. *Eng Anal Boundary Elem* 2017;82:1–16.
- [37] Chen K, Zou D, Kong X, Chan A, Hu Z. A novel nonlinear solution for the polygon scaled boundary finite element method and its application to geotechnical structures[J]. *Computers and Geotechnics* 2017;82:201–10.
- [38] Xu H, Zou D, Kong X, Su X. Error study of Westergaard's approximation in seismic analysis of high concrete-faced rockfill dams based on SBFEM[J]. *Soil Dynamics and Earthquake Engineering* 2017;94:88–91.
- [39] Liu Y, Saputra AA, Wang J, Tin-Loi F, Song C. Automatic polyhedral mesh generation and scaled boundary finite element analysis of STL models[J]. *Computer Methods in Applied Mechanics and Engineering* 2017;313:106–32.
- [40] Wachspress EL. A rational finite element basis. New York: Academic Press; 1975.
- [41] Meyer M, Barr A, Lee H, Desbrun M. Generalized barycentric coordinates on irregular polygons[J]. *Journal of Graphics Tools* 2002;7(1):13–22.
- [42] Floater MS, Kós G, Reimers M. Mean value coordinates in 3D. *Comput Aided Geom Des* 2005;22(7):623–31.
- [43] Sukumar N, Tabarraei A. Conforming polygon finite elements. *Int J Numer Methods Eng* 2004;61:2045–66.
- [44] Jaynes ET. Information theory and statistical mechanics. *Phys Rev* 1957;106(4):620–30.
- [45] Sukumar N. Construction of polygonal interpolants: a maximum entropy approach. *Int J Numer Methods Eng* 2004;61(12):2159–81.
- [46] Sukumar N. Quadratic maximum-entropy serendipity shape functions for arbitrary planar polygons. *Comput Methods Appl Mech Eng* 2013;263(263):27–41.
- [47] Floater MS. Mean value coordinates. *Comput Aided Geom Des* 2003;20(1):19–27.
- [48] Chen K, Zou D, Kong X. A nonlinear approach for the three-dimensional polyhedron scaled boundary finite element method and its verification using Koyna gravity dam. *Soil Dyn Earthquake Eng* 2017;96:1–12.
- [49] Degao Z, Xianjing K, Bin X. User manual for geotechnical dynamic nonlinear analysis. Dalian: Institute of Earthquake Engineering, Dalian University of Technology; 2005.
- [50] Barber JR. Elasticity. 3rd edition. Springer; 2010.
- [51] <http://www.3d66.com/reshtml/9092/9092.html>.
- [52] <http://www.3ddayin.net/soft/jianzhumoxing/28883.html>.
- [53] Zhao C, Chen J, Xu Q. FSI effects and seismic performance evaluation of water storage tank of AP1000 subjected to earthquake loading. *Nucl Eng Des* 2014;280:372–88.