



A dual scaled boundary finite element formulation over arbitrary faceted star convex polyhedra

E. T. Ooi¹ · A. Saputra² · S. Natarajan³ · E. H. Ooi⁴ · C. Song²

Received: 18 July 2019 / Accepted: 7 March 2020 / Published online: 27 March 2020
© Springer-Verlag GmbH Germany, part of Springer Nature 2020

Abstract

A novel technique to formulate arbitrary faceted polyhedral elements in three-dimensions is presented. The formulation is applicable for arbitrary faceted polyhedra, provided that a scaling requirement is satisfied and the polyhedron facets are planar. A triangulation process can be applied to non-planar facets to generate an admissible geometry. The formulation adopts two separate scaled boundary coordinate systems with respect to: (i) a scaling centre located within a polyhedron and; (ii) a scaling centre on a polyhedron's facets. The polyhedron geometry is scaled with respect to both the scaling centres. Polygonal shape functions are derived using the scaled boundary finite element method on the polyhedron facets. The stiffness matrix of a polyhedron is obtained semi-analytically. Numerical integration is required only for the line elements that discretise the polyhedron boundaries. The new formulation passes the patch test. Application of the new formulation in computational solid mechanics is demonstrated using a few numerical benchmarks.

Keywords Scaled boundary finite element method · Polyhedra element · Shape functions · Octree

1 Introduction

The finite element method (FEM) has, since its introduction, become the most dominant approach for computational analysis in many science and engineering applications. The governing equations of the physical phenomena are formulated on standard simplex shapes such as triangles and quadrilaterals for two dimensions; and hexahedra and tetrahedra for three-dimensions, result in simple and efficient computations. However, the limitations on the shapes of the elements introduce some difficulties when modelling problems with topological changes e.g. fracture [44,55]. Often,

re-meshing is employed to generate a compatible mesh for the simulations to continue. This potentially result in ill-shaped elements, further leading to computational inaccuracies and instabilities. On another aspect of computational mechanics, the task of designing of complex components is usually made more complicated by the limited types of elements available in standard FEM libraries. It is not unusual to invest weeks of human effort to construct a suitable mesh for analysis. A possible alternative to reduce the computational burden required for mesh generation is to develop formulations that relax- the requirement of the shapes of the elements that can be used for the analysis. This can be achieved using two approaches: (i) meshless methods and (ii) polygonal (two-dimensional) or polyhedral (three-dimensional) elements.

The meshless method was initially developed to model astrophysical problems by Gingold and Monaghan [23] and Lucy [35]. Its application in computational mechanics gained popularity in the mid-1990's with the introduction of the Element Free Galerkin method [5]. Since then different variants of meshless methods have been proposed including for example, the Reproducing Kernel Particle method [33], the meshless local Petrov-Galerkin method [3], the meshfree radial point interpolation method [31], the *hp*-cloud meshless method [30] and the generalised-strain mesh-free method

✉ E. T. Ooi
e.ooi@federation.edu.au

¹ School of Science, Engineering and Information Technology, Federation University Australia, Ballarat, VIC 3350, Australia

² School of Civil and Environmental Engineering, The University of New South Wales, Sydney, NSW 2031, Australia

³ Integrated Modelling and Simulation Lab, Department of Mechanical Engineering, Indian Institute of Technology, Madras, Chennai 600036, India

⁴ School of Engineering, Monash University-Malaysia, 47500 Bandar Sunway, Selangor, Malaysia

[39]. A detailed discussion on the various formulations of meshless methods is reported by Babuška et al. [4]. Conceptually, the meshless methods formulate the governing equations over subsets of nodes. A computational mesh is not required and it is sufficient that only the information of the nodal positions are specified to define the geometry of the computational domain. Although this successfully reduces the computational burden required for pre-processing, the meshless methods introduce additional complications in the solution process such as difficulty in enforcing the Dirichlet boundary conditions, resolving regions with different materials and discontinuities and the need for high quadrature rules required to integrate the shape functions and their derivatives. Various techniques have to be used to alleviate these problems associated with the meshless methods e.g. [6,13,25,57].

The idea for the possible development of arbitrary shaped elements was conceptualised by Wachspress [53]. However, the development of arbitrary shaped elements did not gain popularity until the development of interpolants based on generalised barycentric coordinates [36], mean valued coordinates [19], natural neighbour shape functions [51] and harmonic coordinates [27]. The shape functions developed using these approaches are conforming and linearly complete, thereby guaranteeing convergence of the method in the limit where the mesh size approaches zero. The development of such class of shape functions for three-dimensional problems in the form of polyhedral elements has been recently pursued. Polyhedral element formulations based on barycentric interpolants [55], Wachspress shape functions [28] and harmonic shape functions [7] have been recently reported. Unstructured polygonal/polyhedral meshes have been shown to produce reliable solutions with superior accuracy [18]. Examples of existing approaches reported on polyhedral meshes include the smoothed finite element method (SFEM) [20], the virtual node method [42], the virtual element method (VEM) [21], Trefftz method [45] and the discontinuous Galerkin method [11] to name a few.

The VEM [15] is an alternative approach to develop polygonal/polyhedral finite elements and is seen as a generalization of finite elements over polytopes. The VEM is derived from mimetic finite differences [9]. One of the salient features of the VEM is that it does not require an explicit form of the basis functions to compute the terms in the stiffness matrix and force vector when the approximation order is linear. In the VEM, the space within an element contains certain polynomials that guarantee accuracy and certain additional functions (non-polynomial) that ensures stability. With a suitable choice of polynomial basis functions and of the number of degrees-of-freedom, the VEM is conforming and provides consistent approximations to the element strain energy for generic deformation modes. The computation of the stiffness matrix in the VEM involves computing boundary integrals and integral of polynomials over a polygonal region.

One approach is to divide the polygonal face into quadrilaterals as discussed in [21]. The other approach involves triangulating the face Aldakheel et al. [1]. The computation of the non-polynomial part that ensures stability involves sub-triangulating the polyhedral element. For more details, interested readers are referred to Aldakheel et al. [1] (c.f. Section 3.2.3). Polyhedral formulation of the VEM for three-dimensional analysis has been reported for diffusion [10], elasticity [14,21], elastoplasticity [1] and thermo-plasticity [2]. More recently, the formulation of a high-order VEM [16] has also been reported in the literature.

The SFEM [32] is an alternate approach to formulate polyhedral elements. In the SFEM, the mesh is (in general) divided into smoothing domains. The smoothing domains, through the divergence theorem, transforms the volume integral required for computing the stiffness matrix into a surface integral. Therefore, only the information on the boundaries of the subcells are required to compute the stiffness matrix. The smoothing domains may be constructed from a subset of nodes or edges over a patch of elements [29] or from cells over a patch of elements [20]. Natarajan et al. [37] also showed the equivalence of polygonal/polyhedral elements formulated from the SFEM and the VEM under specific conditions.

Other approaches that can be used to formulate polyhedral finite elements include the use of polynomial-based interpolation [44], the Voronoi cell FEM [22] and the finite volume method [12]. The polynomial-based interpolation proposed by Rashid and Selimotic [44] is formulated in terms of the physical coordinates and does not strictly satisfy the displacement compatibility condition between adjacent element interfaces. The method however, passes the patch test in the limit when the size of the element approaches zero. The Voronoi cell FEM is based on the hybrid stress formulation and requires careful care when selecting the number of coefficients for the stress interpolation to avoid spurious energy modes corresponding to a singular stiffness matrix. While the finite volume method is more commonly used in computational fluid dynamics simulations many attempts have been made to broaden its application in computational solid mechanics. The formulation of the finite volume method on polyhedral meshes [12] showed superior efficiency when compared with low order FEM approaches.

The scaled boundary finite element method (SBFEM) was developed by Song and Wolf [49]. It is a semi-analytical method that has niche applications in problems involving singularities and unbounded media, where the FEM experiences difficulties in accurately capturing the physics of these problems. The SBFEM is also sufficiently flexible in that it provides an alternative approach to formulate polygonal [40] and polyhedral elements [38,47,52]. The approaches adopted by Talebi et al. [52] and Saputra et al. [47] are similar. The facets of the polyhedral elements are defined by polygonal surfaces that are further discretised using triangular and/or

quadrilateral finite elements. The polyhedral formulation was used in conjunction with octree meshes to take advantage of the computational efficiency of the octree decomposition in mesh generation and solution process. The approach adopted by Natarajan et al. [38] follows a different technique. Polygonal shape functions based on Wachspress interpolants are employed instead of triangular and/or quadrilateral discretisation. In these approaches, numerical integration is adopted over the surfaces of the triangular, quadrilateral or polygonal elements on the polyhedral facets. An analytical integration is adopted within the polyhedral domain.

The objective of this study is to present a new technique to construct polyhedral elements based on the SBFEM. This technique, like that of Natarajan et al. [38] and Zou et al. [58], employs polygonal shape functions to discretise the facets of the polyhedrals in the mesh. The shape functions are in a form of rational polynomials and require special numerical integration over the facets to achieve a desired accuracy. In this work, instead of the Wachspress shape functions, semi-analytical polygonal shape functions are derived from the SBFEM on each facet of the polyhedron. This is achieved by introducing a second scaled boundary coordinate system on the facets of a polyhedron. The scaling of the geometry of a polyhedron is therefore performed twice: (i) with respect to the scaled boundary coordinate on the polygonal surfaces and; (ii) with respect to the scaled boundary coordinate at the scaling centre of the polyhedron. As a result, numerical integration is required only for the line elements that discretise the polyhedra boundaries. An analytical integration is adopted within the polygonal surface and within the polygonal domain. A further discretisation of the polyhedral surfaces into triangles and/or quadrilaterals e.g. in the technique employed by Talebi et al. [52] is not necessary.

This paper is organised as follows: Sect. 2, commences with the description of the developed technique, the coordinate transformation, formulation of the surface shape functions and the solution for the equilibrium condition within a polyhedron. Section 3 focuses on the analytical integration procedure required to compute the coefficient matrices in the new scaled boundary finite element formulation. Other implementation aspects of the developed method associated with the use of polyhedron and octree meshes are described in Sect. 4. Section 5 presents the application of developed method for a few numerical benchmarks for validation. The manuscript concludes in Sect. 6 summarising the major conclusions derived from this study.

2 Dual scaled boundary coordinate systems on arbitrary polyhedra

The proposed technique can be formulated on arbitrary faceted polyhedrons. Each facet is in turn an arbitrary

sided star-convex polygon that forms a plane in the three-dimensional space. For non-planar surfaces, a further triangulation step can be performed on the surface to generate an admissible polyhedron. The only restriction to the geometry of a polyhedron is that it must satisfy the scaling requirement required by the SBFEM on each of the facets of the polyhedron and also within the polyhedron. These scaling requirements require that:

1. There is a point within the polyhedron from which the entire boundary (surface) is directly visible from it. This point is defined as the *polyhedron scaling centre* $O_p(x_p, y_p, z_p)$ and it is convenient to choose this to be the geometric centre of the polyhedron.
2. For each facet that discretises the polyhedron boundary, there is a point on the plane of each facet such that the entire edge of the polygon is directly visible from it. This point is defined as the *surface scaling centre* $O_s(x_s, y_s, z_s)$.

Figure 1 shows a generic polyhedron that can be formulated from a dual scaled boundary coordinate system. A radial coordinate ξ with $0 \leq \xi \leq 1$, is defined at the polyhedron scaling centre. The radial coordinate is $\xi = 0$ at the polyhedron scaling centre O_p and $\xi = 1$ at any point on the polyhedron boundary. Together with the polyhedron scaling centre, each facet on the polyhedron forms a pyramidal sector as shown in Fig. 2. The base of the pyramid is a planar arbitrary sided polygon. A second radial coordinate ζ with $0 \leq \zeta \leq 1$ is defined at the surface scaling centre of the polygon. At the surface scaling centre O_s , $\zeta = 0$ and at the polygon boundary, $\zeta = 1$. Each edge on the polygon boundary is discretised using one-dimensional line elements having local coordinate η with $-1 \leq \eta \leq 1$ similar to the FEM.

The (ξ, η, ζ) coordinate system is shown in Fig. 2. The Cartesian coordinates $\hat{\mathbf{r}}(\eta) = [\hat{x}(\eta), \hat{y}(\eta), \hat{z}(\eta)]^T$ of a point defined on a line element of a generic tetrahedral sector shown in Fig. 2 is interpolated by shape functions as

$$[\hat{x}(\eta) \ \hat{y}(\eta) \ \hat{z}(\eta)]^T = [\mathbf{N}(\eta)\mathbf{x}_b \ \mathbf{N}(\eta)\mathbf{y}_b \ \mathbf{N}(\eta)\mathbf{z}_b]^T \quad (1)$$

where

$$\mathbf{N}(\eta) = [N_1(\eta) \ N_2(\eta) \ \dots \ N_n(\eta)] \quad (2)$$

is the shape function vector with n number of nodes and $\mathbf{x}_b^T = [x_1 \ x_2 \ \dots \ x_n]$, $\mathbf{y}_b^T = [y_1 \ y_2 \ \dots \ y_n]$ and $\mathbf{z}_b^T = [z_1 \ z_2 \ \dots \ z_n]$ are the vectors of nodal coordinates along the x -, y - and z -axes, respectively.

On each surface, the Cartesian coordinates $\bar{\mathbf{r}}(\eta, \zeta) = [\bar{x}(\eta, \zeta), \bar{y}(\eta, \zeta), \bar{z}(\eta, \zeta)]^T$ of a point on the facet of the

Fig. 1 A generic polyhedron modelled by the dual-SBFEM and radial coordinate ξ

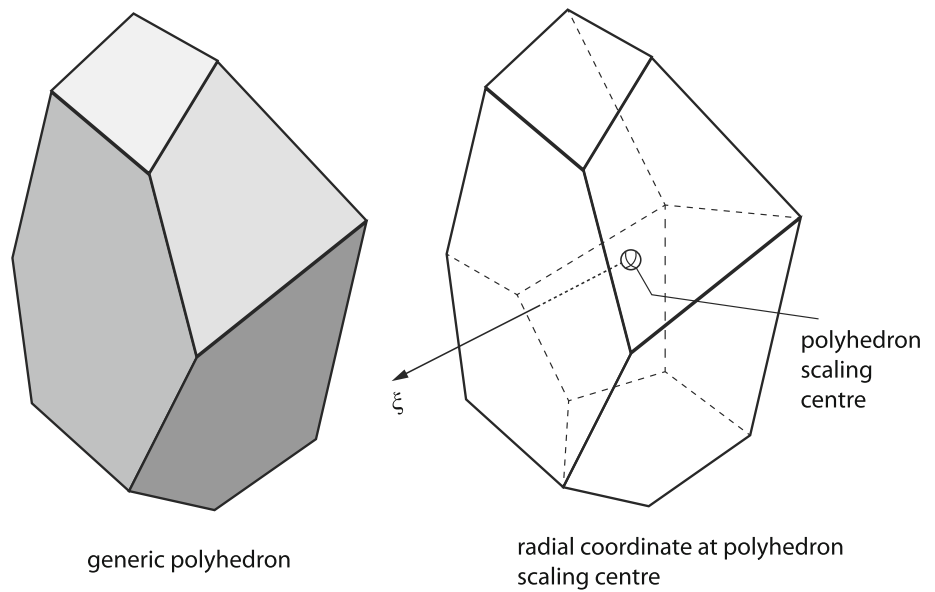
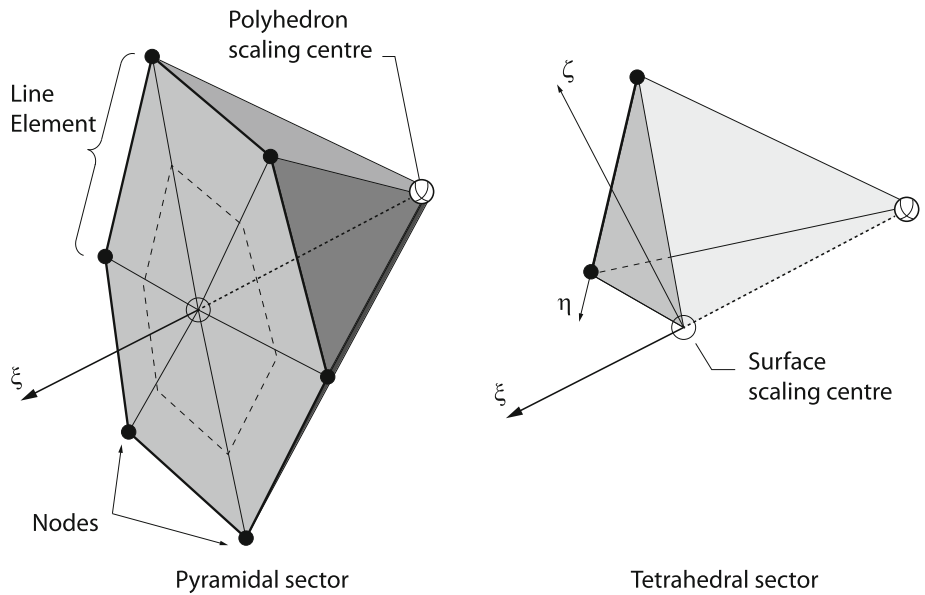


Fig. 2 Dual-SBFEM coordinate system on a pyramidal sector and on a tetrahedral sector of a polyhedron



tetrahedral sector are scaled with respect to the radial coordinate ζ as

$$\begin{Bmatrix} \bar{x}(\eta, \zeta) \\ \bar{y}(\eta, \zeta) \\ \bar{z}(\eta, \zeta) \end{Bmatrix} = (1 - \zeta) \begin{Bmatrix} x_s \\ y_s \\ z_s \end{Bmatrix} + \zeta \begin{Bmatrix} \hat{x}(\eta) \\ \hat{y}(\eta) \\ \hat{z}(\eta) \end{Bmatrix} \quad (3)$$

For a generic case where the polyhedron scaling centre is located at the origin, the Cartesian coordinate $\mathbf{r}(\xi, \eta, \zeta) = [x(\xi, \eta, \zeta), y(\xi, \eta, \zeta), z(\xi, \eta, \zeta)]^T$ of a point in the tetrahedral sector shown in Fig. 2 is obtained by scaling the coordinate $\bar{\mathbf{x}}(\eta, \zeta)$ by the radial coordinate ξ as

$$\mathbf{r}(\xi, \eta, \zeta) = \xi \bar{\mathbf{r}}(\eta, \zeta) \quad (4)$$

The coordinate transformation between the Cartesian and the scaled boundary coordinates is obtained using the chain rule of derivatives as

$$\begin{Bmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \\ \frac{\partial}{\partial \zeta} \end{Bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} & \frac{\partial z}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \eta} \\ \frac{\partial x}{\partial \zeta} & \frac{\partial y}{\partial \zeta} & \frac{\partial z}{\partial \zeta} \end{bmatrix} \begin{Bmatrix} \frac{\partial}{\partial \bar{x}} \\ \frac{\partial}{\partial \bar{y}} \\ \frac{\partial}{\partial \bar{z}} \end{Bmatrix} = \mathbf{J}(\xi, \eta, \zeta) \begin{Bmatrix} \frac{\partial}{\partial \bar{x}} \\ \frac{\partial}{\partial \bar{y}} \\ \frac{\partial}{\partial \bar{z}} \end{Bmatrix} \quad (5)$$

where $\mathbf{J}(\xi, \eta, \zeta)$ is the Jacobian matrix. Using the definitions in Eqs. (3) and (4), $\mathbf{J}(\xi, \eta, \zeta)$ can be expressed as

$$\mathbf{J}(\xi, \eta, \zeta) = \begin{bmatrix} 1 & & \\ & \xi & \\ & & \xi \end{bmatrix} \begin{bmatrix} \bar{x} & \bar{y} & \bar{z} \\ \bar{x}, \zeta & \bar{y}, \zeta & \bar{z}, \zeta \\ \bar{x}, \eta & \bar{y}, \eta & \bar{z}, \eta \end{bmatrix} \quad (6)$$

where the dependencies of \bar{x} , \bar{y} and \bar{z} on η and ζ have been dropped from Eq. (6) for convenience. The second term on the right-hand-side of Eq. (6) is defined as the *surface Jacobian*

$$\bar{\mathbf{J}}(\eta, \zeta) = \begin{bmatrix} \bar{x} & \bar{y} & \bar{z} \\ \bar{x}_{,\zeta} & \bar{y}_{,\zeta} & \bar{z}_{,\zeta} \\ \bar{x}_{,\eta} & \bar{y}_{,\eta} & \bar{z}_{,\eta} \end{bmatrix} \quad (7)$$

Using Eqs. (5), (6) and (7), the coordinate transformation can therefore be expressed as

$$\left[\frac{\partial}{\partial x} \quad \frac{\partial}{\partial y} \quad \frac{\partial}{\partial z} \right]^T = \bar{\mathbf{J}}^{-1}(\eta, \zeta) \left[\frac{\partial}{\partial \xi} \quad \frac{\partial}{\partial \eta} \quad \frac{\partial}{\partial \zeta} \right]^T \quad (8)$$

The Cartesian derivatives can be more conveniently expressed in terms of the outward normal vectors \mathbf{g}^ξ , \mathbf{g}^η and \mathbf{g}^ζ to the three surfaces (η, ζ) , (ζ, ξ) and (ξ, η) , where the coordinates ξ , η and ζ , respectively, are constant on the boundary $\xi = 1$. The outward normal vectors are, respectively

$$\mathbf{g}^\xi = \bar{\mathbf{r}}_{,\zeta} \times \bar{\mathbf{r}}_{,\eta} = \begin{bmatrix} \bar{y}_{,\zeta} \bar{z}_{,\eta} - \bar{y}_{,\eta} \bar{z}_{,\zeta} & \bar{z}_{,\zeta} \bar{x}_{,\eta} - \bar{z}_{,\eta} \bar{x}_{,\zeta} \\ \bar{x}_{,\zeta} \bar{y}_{,\eta} - \bar{x}_{,\eta} \bar{y}_{,\zeta} \end{bmatrix}^T \quad (9a)$$

$$\mathbf{g}^\eta = \bar{\mathbf{r}} \times \bar{\mathbf{r}}_{,\zeta} = \begin{bmatrix} \bar{y} \bar{z}_{,\zeta} - \bar{y}_{,\zeta} \bar{z} & \bar{z} \bar{x}_{,\zeta} - \bar{z}_{,\zeta} \bar{x} \\ \bar{x} \bar{y}_{,\zeta} - \bar{x}_{,\zeta} \bar{y} \end{bmatrix}^T \quad (9b)$$

$$\mathbf{g}^\zeta = \bar{\mathbf{r}}_{,\eta} \times \bar{\mathbf{r}} = \begin{bmatrix} \bar{y}_{,\eta} \bar{z} - \bar{y} \bar{z}_{,\eta} & \bar{z} \bar{x}_{,\eta} - \bar{z}_{,\eta} \bar{x} \\ \bar{x}_{,\eta} \bar{y} - \bar{x} \bar{y}_{,\eta} \end{bmatrix}^T \quad (9c)$$

The unit vectors of \mathbf{g}^ξ , \mathbf{g}^η and \mathbf{g}^ζ are therefore

$$\mathbf{n}^\xi = \begin{bmatrix} n_x^\xi & n_y^\xi & n_z^\xi \end{bmatrix} = \frac{\mathbf{g}^\xi}{g^\xi} \quad (10a)$$

$$\mathbf{n}^\eta = \begin{bmatrix} n_x^\eta & n_y^\eta & n_z^\eta \end{bmatrix} = \frac{\mathbf{g}^\eta}{g^\eta} \quad (10b)$$

$$\mathbf{n}^\zeta = \begin{bmatrix} n_x^\zeta & n_y^\zeta & n_z^\zeta \end{bmatrix} = \frac{\mathbf{g}^\zeta}{g^\zeta} \quad (10c)$$

where g^ξ , g^η and g^ζ are the magnitudes of the vectors \mathbf{g}^ξ , \mathbf{g}^η and \mathbf{g}^ζ , respectively. Using Eqs. (9a)–(10c), $\bar{\mathbf{J}}^{-1}(\eta, \zeta)$ can be expressed as

$$\bar{\mathbf{J}}^{-1}(\eta, \zeta) = \frac{1}{\bar{J}(\eta, \zeta)} \begin{bmatrix} g^\xi n_x^\xi & g^\zeta n_x^\zeta & g^\eta n_x^\eta \\ g^\xi n_y^\xi & g^\zeta n_y^\zeta & g^\eta n_y^\eta \\ g^\xi n_z^\xi & g^\zeta n_z^\zeta & g^\eta n_z^\eta \end{bmatrix} \quad (11)$$

where $\bar{J}(\eta, \zeta)$ is the determinant of $\bar{\mathbf{J}}(\eta, \zeta)$. Using Eq. (3), $\bar{J}(\eta, \zeta)$ can be expressed as

$$\bar{J}(\eta, \zeta) = \zeta J_b(\eta) \quad (12)$$

where

$$J_b(\eta) = x_s(\bar{y}_{,\zeta} \bar{z}_{,\eta} - \bar{y}_{,\eta} \bar{z}_{,\zeta}) + y_s(\bar{z}_{,\zeta} \bar{x}_{,\eta} - \bar{z}_{,\eta} \bar{x}_{,\zeta}) + z_s(\bar{x}_{,\zeta} \bar{y}_{,\eta} - \bar{x}_{,\eta} \bar{y}_{,\zeta}) \quad (13)$$

with

$$[\tilde{x}_{,\eta} \quad \tilde{y}_{,\eta} \quad \tilde{z}_{,\eta}]^T = [\mathbf{N}(\eta)_{,\eta} \mathbf{x}_b \quad \mathbf{N}(\eta)_{,\eta} \mathbf{y}_b \quad \mathbf{N}(\eta)_{,\eta} \mathbf{z}_b]^T \quad (14)$$

and that J_b is a function of η only. Substituting Eq. (11) into Eq. (8) results in

$$\begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{bmatrix} = \frac{g^\xi}{\bar{J}} \begin{bmatrix} n_x^\xi \\ n_y^\xi \\ n_z^\xi \end{bmatrix} \frac{\partial}{\partial \xi} + \frac{1}{\xi} \left(\frac{g^\eta}{\bar{J}} \begin{bmatrix} n_x^\eta \\ n_y^\eta \\ n_z^\eta \end{bmatrix} \frac{\partial}{\partial \eta} + \frac{g^\zeta}{\bar{J}} \begin{bmatrix} n_x^\zeta \\ n_y^\zeta \\ n_z^\zeta \end{bmatrix} \frac{\partial}{\partial \zeta} \right) \quad (15)$$

The coordinate transformation of an infinitesimal volume $d\Omega$ in the dual-scaled boundary coordinate system (ξ, η, ζ) is established from the derivatives of the position vector $\mathbf{r}(\xi, \eta, \zeta)$ defined in Eq. (5) with respect to ξ , η and ζ as

$$d\Omega = \mathbf{r}_{,\xi} \cdot (\mathbf{r}_{,\zeta} \times \mathbf{r}_{,\eta}) d\xi d\eta d\zeta = \bar{J}(\eta, \zeta) \xi^2 d\xi d\eta d\zeta \quad (16)$$

3 Polygonal surface shape functions

To proceed with the equilibrium formulation for a polyhedron, the polygonal facets on each polyhedron have to be discretised. The approach adopted in the conventional SBFEM [50,52] is to further discretise the polyhedron facets into simplex shapes e.g. triangles and quadrilaterals that can be discretised by the FEM. Standard FEM shape functions are then adopted over each simplex shape. A different approach is adopted here. Instead of a further discretisation into simplex shapes, a new set of shape functions is derived on the polygonal facets. For this purpose, local right-handed coordinate system is introduced on each polygonal facet with origin at the surface scaling centre as shown in Fig. 3. The local coordinate system (s, t, n) is defined by three orthogonal axes with unit vectors \mathbf{s} , \mathbf{t} and \mathbf{n} . The unit vector \mathbf{n} is the outward normal pointing away from the polyhedron scaling centre. The unit vectors \mathbf{s} and \mathbf{t} lie along the polygonal plane and can be chosen arbitrarily in any direction so as long as they are orthogonal. A straightforward procedure is to compute \mathbf{s} with reference to an edge on the polygon boundary. The unit vector \mathbf{t} can then be computed as

$$\mathbf{t} = \mathbf{n} \times \mathbf{s} \quad (17)$$

The Cartesian coordinates (x, y, z) of a point on the polygonal facet can be transformed into local coordinates

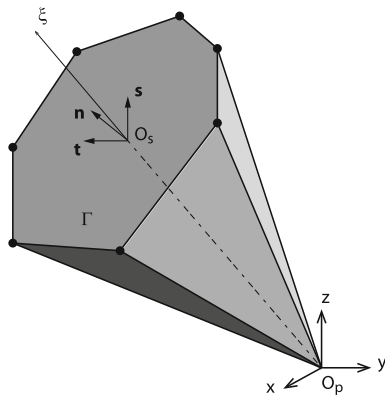


Fig. 3 Local right-handed coordinate system (s, t, n) on a polygonal facet

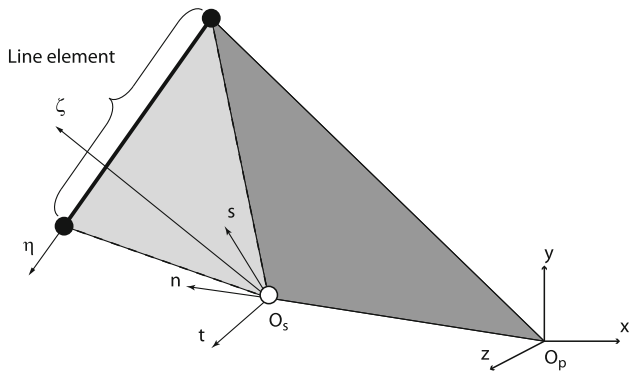


Fig. 4 Local planar coordinate system on a triangular sector bounded by a line element on the boundary

(s, t, n) by defining the 3×3 transformation matrix \mathbf{R}

$$\mathbf{R} = [\mathbf{s} \ \mathbf{t} \ \mathbf{n}] \tag{18}$$

so that

$$[s \ t \ n]^T = \mathbf{R}[x \ y \ z]^T \tag{19}$$

The two-dimensional shape functions on a polygonal facet Γ are then formulated in terms of the SBFEM surface coordinates (η, ζ) from the SBFEM solution of the Laplace equation. This involves the formulation of the SBFEM over a two-dimensional area typical in conventional SBFEM practices e.g. [48] with the coordinates (s, t) replacing the Cartesian coordinates (x, y) . This is similar to the approach of solving coupled field problems with the SBFEM where the shape functions for the (scalar) pressure field in a seepage flow problem is derived from the solution of the Laplace equation [41].

The coordinates (s, t) of a point in a triangular sector bounded by a line element on the boundary (see Fig. 4) and the surface scaling centre O_s can be expressed in terms of

the scaled boundary coordinates on the surface Γ as

$$\begin{Bmatrix} s(\eta, \zeta) \\ t(\eta, \zeta) \end{Bmatrix} = \zeta \begin{Bmatrix} \mathbf{N}(\eta)\mathbf{s}_b \\ \mathbf{N}(\eta)\mathbf{t}_b \end{Bmatrix} \tag{20}$$

where $\mathbf{N}(\eta)$ is the shape function vector described in Eq. (2) and $\mathbf{s}_b = [s_1 \ s_2 \ \dots \ s_n]^T$ and $\mathbf{t}_b = [t_1 \ t_2 \ \dots \ t_n]^T$ are the vector of nodal coordinates of the line element. It is noted here that at any point on the plane, the coordinate n is zero and is omitted in the shape function derivation.

A linear operator ∇_L is defined and expressed in terms of the local planar coordinates (s, t) as

$$\nabla_L^T = \begin{bmatrix} \frac{\partial}{\partial s} & \frac{\partial}{\partial t} \end{bmatrix} \tag{21}$$

Using the chain rule, the coordinate transformation of the derivatives in the (s, t) coordinate system into scaled boundary coordinates on the plane (η, ζ) is expressed as [56]

$$\begin{bmatrix} \frac{\partial}{\partial s} & \frac{\partial}{\partial t} \end{bmatrix}^T = \bar{\mathbf{b}}_1(\eta) \frac{\partial}{\partial \zeta} + \frac{1}{\zeta} \bar{\mathbf{b}}_2(\eta) \frac{\partial}{\partial \eta} \tag{22}$$

where

$$\bar{\mathbf{b}}_1(\eta) = \frac{1}{J_L(\eta)} \begin{bmatrix} \mathbf{N}(\eta)_{,\eta} \mathbf{t}_b \\ -\mathbf{N}(\eta)_{,\eta} \mathbf{s}_b \end{bmatrix} \tag{23a}$$

$$\bar{\mathbf{b}}_2(\eta) = \frac{1}{J_L(\eta)} \begin{bmatrix} -\mathbf{N}(\eta) \mathbf{t}_b \\ \mathbf{N}(\eta) \mathbf{s}_b \end{bmatrix} \tag{23b}$$

and $J_L(\eta)$ is the determinant of the Jacobian matrix $\bar{\mathbf{J}}_L(\eta)$, expressed as

$$\bar{\mathbf{J}}_L(\eta) = \begin{bmatrix} \mathbf{N}(\eta)\mathbf{s}_b & \mathbf{N}(\eta)\mathbf{t}_b \\ \mathbf{N}(\eta)_{,\eta} \mathbf{s}_b & \mathbf{N}(\eta)_{,\eta} \mathbf{t}_b \end{bmatrix} \tag{24}$$

Following Wolf and Song [56], an infinitesimal area $d\Gamma$ on the surface of a polygonal facet can be expressed in terms of (η, ζ) as

$$d\Gamma = J_L(\eta) \zeta d\zeta d\eta \tag{25}$$

The Laplace equation is expressed as

$$\nabla_L^T \nabla_L \theta = 0 \tag{26}$$

where θ is the potential. The potential of a point in a triangular sector bounded by a line element on the boundary and the surface scaling centre O_s is interpolated as

$$\theta(\eta, \zeta) = \mathbf{N}(\eta) \boldsymbol{\theta}_h(\zeta) \tag{27}$$

where $\boldsymbol{\theta}_h(\zeta)$ are analytical functions that satisfy the Laplace equation in the ζ direction. Using Eqs. (22) and (25), the

weak form of Eq. (26) is obtained by multiplying with an arbitrary test function $\bar{\theta}(\eta, \zeta) = \mathbf{N}(\eta)\bar{\theta}_h(\zeta)$ and integrating over Γ resulting in

$$\bar{\theta}_h^T(\zeta) \left[\int \int \mathbf{N}^T \left(\bar{\mathbf{b}}_1 \frac{\partial}{\partial \zeta} + \frac{1}{\zeta} \bar{\mathbf{b}}_2 \frac{\partial}{\partial \eta} \right) \left(\bar{\mathbf{b}}_1 \frac{\partial}{\partial \zeta} + \frac{1}{\zeta} \bar{\mathbf{b}}_2 \frac{\partial}{\partial \eta} \right) \mathbf{N} \zeta J_L \theta_h(\zeta) d\eta d\zeta \right] = 0 \tag{28}$$

where the dependence of $\bar{\mathbf{b}}_1(\eta)$, $\bar{\mathbf{b}}_2(\eta)$, $\mathbf{N}(\eta)$ and $J_L(\eta)$ on η has been dropped for convenience. As $\theta_h(\zeta)$ is arbitrary, the Laplace equation is weakly satisfied if the term inside the square brackets in Eq. (28) is zero. Equation (28) therefore, leads to a differential equation expressed in terms of $\theta_h(\zeta)$ as

$$\bar{\mathbf{E}}_0 \zeta^2 \theta_{h,\zeta\zeta} + (\bar{\mathbf{E}}_0 + \bar{\mathbf{E}}_1^T + \bar{\mathbf{E}}_1) \zeta \theta_{h,\zeta} - \bar{\mathbf{E}}_0 \theta_h(\zeta) + \bar{\mathbf{F}}_q(\zeta) = 0 \tag{29}$$

where the coefficient matrices $\bar{\mathbf{E}}_0$, $\bar{\mathbf{E}}_1$ and $\bar{\mathbf{E}}_2$ are

$$\bar{\mathbf{E}}_0 = \int_{-1}^1 \bar{\mathbf{B}}_1^T(\eta) \bar{\mathbf{B}}_1(\eta) J_L(\eta) d\eta \tag{30a}$$

$$\bar{\mathbf{E}}_1 = \int_{-1}^1 \bar{\mathbf{B}}_2^T(\eta) \bar{\mathbf{B}}_1(\eta) J_L(\eta) d\eta \tag{30b}$$

$$\bar{\mathbf{E}}_2 = \int_{-1}^1 \bar{\mathbf{B}}_2^T(\eta) \bar{\mathbf{B}}_2(\eta) J_L(\eta) d\eta \tag{30c}$$

and

$$\bar{\mathbf{B}}_1(\eta) = \bar{\mathbf{b}}_1(\eta) \mathbf{N}(\eta) \tag{31a}$$

$$\bar{\mathbf{B}}_2(\eta) = \bar{\mathbf{b}}_2(\eta) \mathbf{N}(\eta), \eta \tag{31b}$$

The coefficient matrices $\bar{\mathbf{E}}_0$, $\bar{\mathbf{E}}_1$ and $\bar{\mathbf{E}}_2$ are assembled element by element over the boundary of the polygon.

The term $\bar{\mathbf{F}}_q(\zeta)$ in Eq. (29) is

$$\bar{\mathbf{F}}_q(\zeta) = \mathbf{N}^T(\eta) \bar{\mathbf{b}}_2^T(\eta) \left(\bar{\mathbf{b}}_1 \frac{\partial}{\partial \zeta} + \frac{1}{\zeta} \bar{\mathbf{b}}_2 \frac{\partial}{\partial \eta} \right) \mathbf{N}(\eta) \theta_h(\zeta) \Big|_{-1}^1 \tag{32}$$

and represents the flux vector along a line of constant ζ . For polygons that form a closed loop such as those considered in this manuscript, $\bar{\mathbf{F}}_q(\zeta)$ vanishes and the weak form of the Laplace equation becomes

$$\bar{\mathbf{E}}_0 \zeta^2 \theta_{h,\zeta\zeta} + (\bar{\mathbf{E}}_0 + \bar{\mathbf{E}}_1^T + \bar{\mathbf{E}}_1) \zeta \theta_{h,\zeta} - \bar{\mathbf{E}}_0 \theta_h(\zeta) = 0 \tag{33}$$

Eq. (33) can be solved using an eigenvalue or a Schur decomposition by a conversion into a first order differential equation

[48,56]. This results in

$$\theta_h(\zeta) = \bar{\mathbf{V}} \zeta^{-\bar{\mathbf{S}}} \bar{\mathbf{c}} \tag{34}$$

where $\bar{\mathbf{S}}$ is eigenvalue/Schur- matrices with negative eigenvalues, which are obtained from an eigenvalue or a Schur decomposition of the Hamiltonian matrix

$$\bar{\mathbf{Z}} = \begin{bmatrix} \bar{\mathbf{E}}_0^{-1} \bar{\mathbf{E}}_1^T & -\bar{\mathbf{E}}_0^{-1} \\ -\bar{\mathbf{E}}_2 + \bar{\mathbf{E}}_1 \bar{\mathbf{E}}_0^{-1} \bar{\mathbf{E}}_1^T & -\bar{\mathbf{E}}_1 \bar{\mathbf{E}}_0^{-1} \end{bmatrix} \tag{35}$$

and $\bar{\mathbf{V}}$ is the transformation matrix with eigenvectors relevant to $\bar{\mathbf{S}}$. The integration constants $\bar{\mathbf{c}}$ in Eq. (34) depend on the potentials on the polygon boundary $\theta_b = \theta(\zeta = 1)$. They are obtained from as

$$\bar{\mathbf{c}} = \bar{\mathbf{V}}^{-1} \theta_b \tag{36}$$

Substituting Eqs. (36) and (34) into Eq. (27), the potential field of a point in the triangular sector bounded by O_s and a line element on the polygon boundary can be expressed as

$$\theta(\eta, \zeta) = \mathbf{N}(\eta) \bar{\mathbf{V}}^{(e)} \zeta^{-\bar{\mathbf{S}}} \bar{\mathbf{V}}^{-1} \theta_b \tag{37}$$

where the superscript (e) indicates that only the rows in corresponding to element (e) are required for each interpolation within a triangular sector. This notation is necessary because Eq. (37) is obtained after assembling the contributions from the line elements that form the closed loop defining the polygon. The planar shape functions $\bar{\Phi}(\eta, \zeta)$ on Γ is defined in Eq. (37) as

$$\bar{\Phi}(\eta, \zeta) = \mathbf{N}(\eta) \bar{\mathbf{V}}^{(e)} \zeta^{-\bar{\mathbf{S}}} \bar{\mathbf{V}}^{-1} \tag{38}$$

It is valid for arbitrary sided polygons on a polyhedron facet so as long as the scaling requirement on the facet is satisfied. It interpolates a scalar variable on a planar polygonal facet located on the boundary of a polyhedron.

4 Equilibrium formulation of a polyhedron

4.1 Interpolation of displacement and strain-displacement relations

In order to use the shape functions derived in Eq. (38) in a three-dimensional setting, we consider now, a generic tetrahedral sector such as that shown in Fig. 2. Following the SBFEM, the Cartesian displacement components $u(\xi, \eta, \zeta)$, $v(\xi, \eta, \zeta)$ and $w(\xi, \eta, \zeta)$ of a point in the tetrahedral sector is interpolated as

$$u(\xi, \eta, \zeta) = \bar{\Phi}(\eta, \zeta) u_h(\xi) \tag{39a}$$

$$v(\xi, \eta, \zeta) = \bar{\Phi}(\eta, \zeta)v_h(\xi) \quad (39b)$$

$$w(\xi, \eta, \zeta) = \bar{\Phi}(\eta, \zeta)w_h(\xi) \quad (39c)$$

where the shape functions in Eq. (38) have been used to interpolate the analytical displacement functions $u_h(\xi)$, $v_h(\xi)$ and $w_h(\xi)$. Comparing Eqs. (39a)–(39c), with Eq. (37), it can be inferred that the potential $\theta(\eta, \zeta)$ has the physical meaning of the Cartesian displacement components on the polygonal facet i.e. at $\xi = 1$. To consolidate the equations, the displacement field vector $\mathbf{u}(\xi, \eta, \zeta) = [u(\xi, \eta, \zeta) \ v(\xi, \eta, \zeta) \ w(\xi, \eta, \zeta)]^T$ is interpolated so that

$$\mathbf{u}(\xi, \eta, \zeta) = \Phi(\eta, \zeta)\mathbf{T}\mathbf{u}_h(\xi) \quad (40)$$

where

$$\Phi(\eta, \zeta) = \begin{bmatrix} \bar{\Phi}(\eta, \zeta) & & \\ & \bar{\Phi}(\eta, \zeta) & \\ & & \bar{\Phi}(\eta, \zeta) \end{bmatrix} \quad (41)$$

and \mathbf{T} is a $3n \times 3n$ transformation matrix with n the number of nodes on a line element on a tetrahedral sector. The transformation matrix is necessary to enable the standard Voigt notation to be adopted for the displacement functions $\mathbf{u}_h(\xi)$.

Using the structure of the planar shape functions $\bar{\Phi}(\eta, \zeta)$ in Eq. (38), the shape function $\Phi(\eta, \zeta)$ can be decomposed as

$$\Phi(\eta, \zeta) = \mathbf{N}_u(\eta)\mathbf{V}\zeta\mathbf{S}\mathbf{U} \quad (42)$$

where

$$\mathbf{N}_u(\eta) = \begin{bmatrix} \mathbf{N}(\eta) & & \\ & \mathbf{N}(\eta) & \\ & & \mathbf{N}(\eta) \end{bmatrix} \quad (43)$$

$$\mathbf{V} = \begin{bmatrix} \bar{\mathbf{V}} & & \\ & \bar{\mathbf{V}} & \\ & & \bar{\mathbf{V}} \end{bmatrix} \quad (44)$$

$$\mathbf{S} = \begin{bmatrix} -\bar{\mathbf{S}} & & \\ & -\bar{\mathbf{S}} & \\ & & -\bar{\mathbf{S}} \end{bmatrix} \quad (45)$$

$$\mathbf{U} = \begin{bmatrix} \bar{\mathbf{V}}^{-1} & & \\ & \bar{\mathbf{V}}^{-1} & \\ & & \bar{\mathbf{V}}^{-1} \end{bmatrix} \quad (46)$$

A linear differential operator ∇_u is defined as

$$\nabla_u = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 & 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} \\ 0 & \frac{\partial}{\partial y} & 0 & \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} \\ 0 & 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \end{bmatrix}^T \quad (47)$$

Using Eq. (15), ∇_u is expressed as

$$\nabla_u = \mathbf{b}_1(\eta, \zeta)\frac{\partial}{\partial \xi} + \frac{1}{\xi}\left(\mathbf{b}_2(\eta, \zeta)\frac{\partial}{\partial \eta} + \mathbf{b}_3(\eta, \zeta)\frac{\partial}{\partial \zeta}\right) \quad (48)$$

where

$$\mathbf{b}_1^T(\eta, \zeta) = \frac{g^\xi}{J} \begin{bmatrix} n_x^\xi & 0 & 0 & 0 & n_z^\xi & n_y^\xi \\ 0 & n_y^\xi & 0 & n_z^\xi & 0 & n_x^\xi \\ 0 & 0 & n_z^\xi & n_y^\xi & n_x^\xi & 0 \end{bmatrix} \quad (49a)$$

$$\mathbf{b}_2^T(\eta, \zeta) = \frac{g^\eta}{J} \begin{bmatrix} n_x^\eta & 0 & 0 & 0 & n_z^\eta & n_y^\eta \\ 0 & n_y^\eta & 0 & n_z^\eta & 0 & n_x^\eta \\ 0 & 0 & n_z^\eta & n_y^\eta & n_x^\eta & 0 \end{bmatrix} \quad (49b)$$

$$\mathbf{b}_3^T(\eta, \zeta) = \frac{g^\zeta}{J} \begin{bmatrix} n_x^\zeta & 0 & 0 & 0 & n_z^\zeta & n_y^\zeta \\ 0 & n_y^\zeta & 0 & n_z^\zeta & 0 & n_x^\zeta \\ 0 & 0 & n_z^\zeta & n_y^\zeta & n_x^\zeta & 0 \end{bmatrix} \quad (49c)$$

Furthermore, the identity

$$(\bar{J}\mathbf{b}_2(\eta, \zeta))_{,\eta} + (\bar{J}\mathbf{b}_3(\eta, \zeta))_{,\zeta} = -2\bar{J}\mathbf{b}_1(\eta, \zeta) \quad (50)$$

can be established from Eqs. (9a)–(9c) and Eqs. (49a)–(49c).

Using the definitions of \mathbf{g}^ξ , \mathbf{g}^η and \mathbf{g}^ζ in Eqs. (9a)–(9c), the Jacobian on the boundary in Eq. (12) and the surface coordinates in Eq. (3), $\mathbf{b}_1(\eta, \zeta)$, $\mathbf{b}_2(\eta, \zeta)$ and $\mathbf{b}_3(\eta, \zeta)$ can be decomposed as

$$\mathbf{b}_1(\eta, \zeta) = \frac{1}{J_b}\mathbf{b}_{1a}(\eta) \quad (51a)$$

$$\mathbf{b}_2(\eta, \zeta) = \frac{1}{\zeta J_b}\mathbf{b}_{2a}(\eta) \quad (51b)$$

$$\mathbf{b}_3(\eta, \zeta) = \frac{1}{J_b}\mathbf{b}_{3a}(\eta) + \frac{\zeta}{J_b}\mathbf{b}_{3b}(\eta) \quad (51c)$$

where the matrices $\mathbf{b}_{1a}(\eta)$, $\mathbf{b}_{2a}(\eta)$, $\mathbf{b}_{3a}(\eta)$ and $\mathbf{b}_{3b}(\eta)$ are functions of η only with

$$I = \int \xi \left[\oint_{\Gamma^\xi} (\mathbf{w}^T \bar{\mathbf{J}} \mathbf{b}_2^T - \mathbf{w}^T \bar{\mathbf{J}} \mathbf{b}_3^T) \sigma d\Gamma - \int \int ((\mathbf{w}^T \bar{\mathbf{J}} \mathbf{b}_2^T)_{,\eta} + (\mathbf{w}^T \bar{\mathbf{J}} \mathbf{b}_3^T)_{,\zeta}) \sigma d\eta d\zeta \right] d\xi \tag{57}$$

$$\mathbf{b}_{1a}^T = \begin{bmatrix} \bar{y}_{,\zeta} \tilde{z} - \tilde{y} \bar{z}_{,\zeta} & 0 & 0 & 0 & \bar{x}_{,\zeta} \tilde{y} - \tilde{x} \bar{y}_{,\zeta} & \bar{z}_{,\zeta} \tilde{x} - \tilde{z} \bar{x}_{,\zeta} \\ 0 & \bar{z}_{,\zeta} \tilde{x} - \tilde{z} \bar{x}_{,\zeta} & 0 & \bar{x}_{,\zeta} \tilde{y} - \tilde{x} \bar{y}_{,\zeta} & 0 & \bar{y}_{,\zeta} \tilde{z} - \tilde{y} \bar{z}_{,\zeta} \\ 0 & 0 & \bar{x}_{,\zeta} \tilde{y} - \tilde{x} \bar{y}_{,\zeta} & \bar{z}_{,\zeta} \tilde{x} - \tilde{z} \bar{x}_{,\zeta} & \bar{y}_{,\zeta} \tilde{z} - \tilde{y} \bar{z}_{,\zeta} & 0 \end{bmatrix} \tag{52a}$$

$$\mathbf{b}_{2a}^T = \begin{bmatrix} y_s \bar{z}_{,\zeta} - \bar{y}_{,\zeta} z_s & 0 & 0 & 0 & x_s \bar{y}_{,\zeta} - \bar{x}_{,\zeta} y_s & z_s \bar{x}_{,\zeta} - \bar{z}_{,\zeta} x_s \\ 0 & z_s \bar{x}_{,\zeta} - \bar{z}_{,\zeta} x_s & 0 & x_s \bar{y}_{,\zeta} - \bar{x}_{,\zeta} y_s & 0 & y_s \bar{z}_{,\zeta} - \bar{y}_{,\zeta} z_s \\ 0 & 0 & x_s \bar{y}_{,\zeta} - \bar{x}_{,\zeta} y_s & z_s \bar{x}_{,\zeta} - \bar{z}_{,\zeta} x_s & y_s \bar{z}_{,\zeta} - \bar{y}_{,\zeta} z_s & 0 \end{bmatrix} \tag{52b}$$

$$\mathbf{b}_{3a}^T = \begin{bmatrix} \tilde{y} z_s - y_s \tilde{z} & 0 & 0 & \tilde{x} y_s - x_s \tilde{y} & \tilde{z} x_s - z_s \tilde{x} \\ 0 & \tilde{z} x_s - z_s \tilde{x} & 0 & \tilde{x} y_s - x_s \tilde{y} & 0 & \tilde{y} z_s - y_s \tilde{z} \\ 0 & 0 & \tilde{x} y_s - x_s \tilde{y} & \tilde{z} x_s - z_s \tilde{x} & \tilde{y} z_s - y_s \tilde{z} & 0 \end{bmatrix} \tag{52c}$$

$$\mathbf{b}_{3b}^T = \begin{bmatrix} \tilde{y} \bar{z}_{,\zeta} - \bar{y}_{,\zeta} \tilde{z} & 0 & 0 & \tilde{x} \bar{y}_{,\zeta} - \bar{x}_{,\zeta} \tilde{y} & \tilde{z} \bar{x}_{,\zeta} - \bar{z}_{,\zeta} \tilde{x} \\ 0 & \tilde{z} \bar{x}_{,\zeta} - \bar{z}_{,\zeta} \tilde{x} & 0 & \tilde{x} \bar{y}_{,\zeta} - \bar{x}_{,\zeta} \tilde{y} & 0 & \tilde{y} \bar{z}_{,\zeta} - \bar{y}_{,\zeta} \tilde{z} \\ 0 & 0 & \tilde{x} \bar{y}_{,\zeta} - \bar{x}_{,\zeta} \tilde{y} & \tilde{z} \bar{x}_{,\zeta} - \bar{z}_{,\zeta} \tilde{x} & \tilde{y} \bar{z}_{,\zeta} - \bar{y}_{,\zeta} \tilde{z} & 0 \end{bmatrix} \tag{52d}$$

4.2 Discretisation of equilibrium equation

The condition of static equilibrium of a generic polyhedron is expressed

$$\nabla_u^T \sigma + \mathbf{f}_b = \mathbf{0} \tag{53}$$

with boundary conditions

$$\mathbf{u} = \bar{\mathbf{u}} \text{ on } \Gamma_u \tag{54}$$

$$\sigma \cdot \mathbf{n} = \bar{\mathbf{t}} \text{ on } \Gamma_t \tag{55}$$

where σ is the stress field, \mathbf{f}_b is the body load intensity, Γ_u is the portion of the boundary with prescribed displacements $\bar{\mathbf{u}}$, Γ_t is the portion of the boundary with prescribed tractions $\bar{\mathbf{t}}$ and \mathbf{n} is the outward unit normal on Γ_t . The corresponding weak form of Eq. (53) is formulated using the method of weighted residuals. Multiplying Eq. (53) with a weight function $\mathbf{w}(\xi, \eta, \zeta)$, using Eq. (48) and integrating over the domain $d\Omega$ results in

$$\int \mathbf{w}^T \mathbf{b}_1^T \sigma_{,\xi} d\Omega + \int \mathbf{w}^T \xi^{-1} (\mathbf{b}_2^T \sigma_{,\eta} + \mathbf{b}_3^T \sigma_{,\zeta}) d\Omega + \int \mathbf{w}^T \mathbf{f}_b d\Omega = 0 \tag{56}$$

where the dependency of \mathbf{w} and σ on ξ, η and ζ ; and $\mathbf{b}_1, \mathbf{b}_2$ and \mathbf{b}_3 on η and ζ have been dropped for convenience. The second term in Eq. (56) is examined. Substituting Eq. (16) for $d\Omega$ and integrating by parts results in

where Γ^ξ is the surface with constant ξ . Using the identity in Eq. (50) results in

$$I = \int \xi \left[\oint_{\Gamma^\xi} (\mathbf{w}^T \bar{\mathbf{J}} \mathbf{b}_2^T - \mathbf{w}^T \bar{\mathbf{J}} \mathbf{b}_3^T) \sigma d\Gamma - \int \int (-2\mathbf{w}^T \bar{\mathbf{J}} \mathbf{b}_1^T + \mathbf{w}_{,\eta}^T \bar{\mathbf{J}} \mathbf{b}_2^T + \mathbf{w}_{,\zeta}^T \bar{\mathbf{J}} \mathbf{b}_3^T) \sigma d\eta d\zeta \right] d\xi \tag{58}$$

Substituting Eq. (58) into Eq. (56) and rearranging the terms results in

$$\int \left[\xi^2 \int \int \mathbf{w}^T \bar{\mathbf{J}} \mathbf{b}_1^T \sigma_{,\xi} d\eta d\zeta - \xi \int \int (-2\mathbf{w}^T \bar{\mathbf{J}} \mathbf{b}_1^T + \mathbf{w}_{,\eta}^T \bar{\mathbf{J}} \mathbf{b}_2^T + \mathbf{w}_{,\zeta}^T \bar{\mathbf{J}} \mathbf{b}_3^T) \sigma d\eta d\zeta + \dots \right. \\ \left. \xi^2 \int \int \mathbf{w}^T \bar{\mathbf{J}} \mathbf{f}_b d\eta d\zeta + \xi \oint_{\Gamma^\xi} (\mathbf{w}^T \bar{\mathbf{J}} \mathbf{b}_2^T - \mathbf{w}^T \bar{\mathbf{J}} \mathbf{b}_3^T) \sigma d\Gamma \right] d\xi = 0 \tag{59}$$

The last term on the left-hand-side of Eq. (59) represents the surface tractions acting along a line of constant ξ . For polyhedrons that form a close loop, the contributions from the individual tetrahedral sectors over the polyhedron equilibrate and this term vanishes when it is assembled over all the entire polyhedron. Therefore, the equilibrium condition of a polyhedron reduces to

$$\xi^2 \int \int \mathbf{w}^T \bar{\mathbf{J}} \mathbf{b}_1^T \sigma_{,\xi} d\eta d\zeta - \xi \int \int (-2\mathbf{w}^T \bar{\mathbf{J}} \mathbf{b}_1^T + \mathbf{w}_{,\eta}^T \bar{\mathbf{J}} \mathbf{b}_2^T + \mathbf{w}_{,\zeta}^T \bar{\mathbf{J}} \mathbf{b}_3^T) \sigma d\eta d\zeta + \dots \\ \xi^2 \int \int \mathbf{w}^T \bar{\mathbf{J}} \mathbf{f}_b d\eta d\zeta = 0 \tag{60}$$

Using Hooke’s law $\sigma = \mathbf{D}\epsilon$ and the strain displacement relation $\epsilon = \nabla_u \mathbf{u}$, the stress field is expressed as

$$\sigma(\xi, \eta, \zeta) = \mathbf{D}(\mathbf{B}_1(\eta, \zeta)\mathbf{T}\mathbf{u}_h(\xi),_{\xi} + \xi^{-1}\mathbf{B}_2(\eta, \zeta)\mathbf{T}\mathbf{u}_h(\xi)) \tag{61}$$

where

$$\mathbf{B}_1(\eta, \zeta) = \mathbf{b}_1(\eta, \zeta)\Phi(\eta, \zeta) \tag{62a}$$

$$\mathbf{B}_2(\eta, \zeta) = \mathbf{b}_2(\eta, \zeta)\Phi(\eta, \zeta),_{\eta} + \mathbf{b}_3(\eta, \zeta)\Phi(\eta, \zeta),_{\zeta} \tag{62b}$$

The weighting function is selected so that it is interpolated by the same shape functions as the displacement field $\mathbf{u}(\xi, \eta, \zeta)$

$$\mathbf{w}(\xi, \eta, \zeta) = \Phi(\eta, \zeta)\mathbf{T}\mathbf{w}_h(\xi) \tag{63}$$

Substituting Eqs. (61)–(63) into Eq. (60) and invoking the condition that the equilibrium has to be satisfied for arbitrary $\mathbf{w}_h(\xi)$ results in

$$\mathbf{E}_0 \xi^2 \mathbf{u}_h(\xi),_{\xi\xi} + (2\mathbf{E}_0 - \mathbf{E}_1 + \mathbf{E}_1^T) \xi \mathbf{u}_h(\xi),_{\xi} + (\mathbf{E}_1^T - \mathbf{E}_2) \mathbf{u}_h(\xi) + \xi^2 \mathbf{F}_b = \mathbf{0} \tag{64}$$

where the coefficient matrices \mathbf{E}_0 , \mathbf{E}_1 and \mathbf{E}_2 are

$$\mathbf{E}_0 = \mathbf{T}^T \int_0^1 \int_{-1}^1 \mathbf{B}_1^T(\eta, \zeta) \mathbf{D} \mathbf{B}_1(\eta, \zeta) \bar{J}(\eta, \zeta) d\eta d\zeta \mathbf{T} \tag{65a}$$

$$\mathbf{E}_1 = \mathbf{T}^T \int_0^1 \int_{-1}^1 \mathbf{B}_2^T(\eta, \zeta) \mathbf{D} \mathbf{B}_1(\eta, \zeta) \bar{J}(\eta, \zeta) d\eta d\zeta \mathbf{T} \tag{65b}$$

$$\mathbf{E}_2 = \mathbf{T}^T \int_0^1 \int_{-1}^1 \mathbf{B}_2^T(\eta, \zeta) \mathbf{D} \mathbf{B}_2(\eta, \zeta) \bar{J}(\eta, \zeta) d\eta d\zeta \mathbf{T} \tag{65c}$$

and

$$\mathbf{F}_b = \mathbf{T}^T \int_0^1 \int_{-1}^1 \Phi^T(\eta, \zeta) \bar{J} \mathbf{f}_b d\eta d\zeta \tag{66}$$

It is worth noting here that Eq. (64) is similar to the scaled boundary finite element equation in displacements [49] but with different coefficient matrices \mathbf{E}_0 , \mathbf{E}_1 and \mathbf{E}_2 . The coefficient matrices \mathbf{E}_0 , \mathbf{E}_1 and \mathbf{E}_2 can be integrated numerically in the η direction using standard Gauss-Lobatto quadrature along the line elements on the boundaries of a polyhedron. Along the ζ direction, analytical integration is employed.

In each of the coefficient matrices \mathbf{E}_0 , \mathbf{E}_1 and \mathbf{E}_2 , substituting Eqs. (62a) and (62b) and using the definition of $\Phi(\eta, \zeta)$ in Eq. (42) results in

$$\mathbf{E}_0 = \mathbf{T}^T \mathbf{U}^T \int_0^1 \zeta^{\mathbf{S}^T + \mathbf{I}} \mathbf{Y}_{00} \zeta^{\mathbf{S}} d\zeta \mathbf{U} \mathbf{T} \tag{67a}$$

$$\mathbf{E}_1 = \mathbf{T}^T \mathbf{U}^T \left(\int_0^1 \zeta^{\mathbf{S}^T + \mathbf{I}} \mathbf{Y}_{10} \zeta^{\mathbf{S}} d\zeta + \int_0^1 \zeta^{\mathbf{S}^T} \mathbf{Y}_{11} \zeta^{\mathbf{S}} d\zeta \right) \mathbf{U} \mathbf{T} \tag{67b}$$

$$\mathbf{E}_2 = \mathbf{T}^T \mathbf{U}^T \left(\int_0^1 \zeta^{\mathbf{S}^T + \mathbf{I}} \mathbf{Y}_{20} \zeta^{\mathbf{S}} d\zeta + \int_0^1 \zeta^{\mathbf{S}^T} \mathbf{Y}_{21} \zeta^{\mathbf{S}} d\zeta + \int_0^1 \zeta^{\mathbf{S}^T - \mathbf{I}} \mathbf{Y}_{22} \zeta^{\mathbf{S}} d\zeta \right) \mathbf{U} \mathbf{T} \tag{67c}$$

where

$$\mathbf{Y}_{00} = \mathbf{V}^T \int_{-1}^1 \mathbf{N}_u^T \mathbf{b}_{1a}^T \mathbf{D} \mathbf{b}_{1a} \mathbf{N}_u J_b^{-1} d\eta \mathbf{V} \tag{68a}$$

$$\mathbf{Y}_{10} = \mathbf{S}^T \mathbf{V}^T \int_{-1}^1 \mathbf{N}_u^T \mathbf{b}_{3b}^T \mathbf{D} \mathbf{b}_{1a} \mathbf{N}_u J_b^{-1} d\eta \mathbf{V} \tag{68b}$$

$$\mathbf{Y}_{20} = \mathbf{S}^T \mathbf{V}^T \int_{-1}^1 \mathbf{N}_u^T \mathbf{b}_{3b}^T \mathbf{D} \mathbf{b}_{3b} \mathbf{N}_u J_b^{-1} d\eta \mathbf{V} \mathbf{S} \tag{68c}$$

$$\mathbf{Y}_{11} = \mathbf{S}^T \mathbf{V}^T \int_{-1}^1 \mathbf{N}_u^T \mathbf{b}_{3a}^T \mathbf{D} \mathbf{b}_{1a} \mathbf{N}_u J_b^{-1} d\eta \mathbf{V} + \mathbf{V}^T \int_{-1}^1 \mathbf{N}_{u,\eta}^T \mathbf{b}_{2a}^T \mathbf{D} \mathbf{b}_{1a} \mathbf{N}_u J_b^{-1} d\eta \mathbf{V} \tag{68d}$$

$$\mathbf{Y}_{21} = \mathbf{V}^T \int_{-1}^1 \mathbf{N}_{u,\eta}^T \mathbf{b}_{2a}^T \mathbf{D} \mathbf{b}_{3b} \mathbf{N}_u J_b^{-1} d\eta \mathbf{V} \mathbf{S} + \mathbf{S}^T \mathbf{V}^T \int_{-1}^1 \mathbf{N}_u^T \mathbf{b}_{3b}^T \mathbf{D} \mathbf{b}_{2a} \mathbf{N}_{u,\eta} J_b^{-1} d\eta \mathbf{V} + \mathbf{S}^T \mathbf{V}^T \int_{-1}^1 \mathbf{N}_u^T \mathbf{b}_{3a}^T \mathbf{D} \mathbf{b}_{3b} \mathbf{N}_u J_b^{-1} d\eta \mathbf{V} \mathbf{S} + \mathbf{S}^T \mathbf{V}^T \int_{-1}^1 \mathbf{N}_u^T \mathbf{b}_{3b}^T \mathbf{D} \mathbf{b}_{3a} \mathbf{N}_u J_b^{-1} d\eta \mathbf{V} \mathbf{S} \tag{68e}$$

$$\mathbf{Y}_{22} = \mathbf{V}^T \int_{-1}^1 \mathbf{N}_{u,\eta}^T \mathbf{b}_{2a}^T \mathbf{D} \mathbf{b}_{2a} \mathbf{N}_{u,\eta} J_b^{-1} d\eta \mathbf{V} \mathbf{S} + \mathbf{S}^T \mathbf{V}^T \int_{-1}^1 \mathbf{N}_u^T \mathbf{b}_{3a}^T \mathbf{D} \mathbf{b}_{3a} \mathbf{N}_u J_b^{-1} d\eta \mathbf{V} \mathbf{S} + \mathbf{V}^T \int_{-1}^1 \mathbf{N}_{u,\eta}^T \mathbf{b}_{2a}^T \mathbf{D} \mathbf{b}_{3a} \mathbf{N}_u J_b^{-1} d\eta \mathbf{V} \mathbf{S} + \mathbf{S}^T \mathbf{V}^T \int_{-1}^1 \mathbf{N}_u^T \mathbf{b}_{3a}^T \mathbf{D} \mathbf{b}_{2a} \mathbf{N}_{u,\eta} J_b^{-1} d\eta \mathbf{V} \tag{68f}$$

are functions of η only and can be integrated numerically.

The integrals with respect to ζ in Eqs. (67a)–(67c) can be computed by introducing the matrices

$$\mathbf{X}_{i0} = \int_0^1 \zeta^{\mathbf{S}^T + \mathbf{I}} \mathbf{Y}_{i0} \zeta^{\mathbf{S}} d\zeta \quad \text{for } i = 0, 1, 2 \tag{69a}$$

$$\mathbf{X}_{i1} = \int_0^1 \zeta^{\mathbf{S}^T} \mathbf{Y}_{i1} \zeta^{\mathbf{S}} d\zeta \quad \text{for } i = 1, 2 \tag{69b}$$

$$\mathbf{X}_{i2} = \int_0^1 \zeta^{\mathbf{S}^T - \mathbf{I}} \mathbf{Y}_{i2} \zeta^{\mathbf{S}} d\zeta \quad \text{for } i = 2 \tag{69c}$$

The matrices \mathbf{X}_{00} , \mathbf{X}_{10} , \mathbf{X}_{20} , \mathbf{X}_{11} , \mathbf{X}_{21} and \mathbf{X}_{22} are obtained by integrating Eqs. (69a)–(69c) analytically and solving the

resulting Lyapunov equations

$$(\mathbf{S} + \mathbf{I})^T \mathbf{X}_{i0} + \mathbf{X}_{i0}(\mathbf{S} + \mathbf{I}) = \mathbf{Y}_{i0} \quad \text{for } i = 0, 1, 2 \tag{70a}$$

$$(\mathbf{S} + 0.5\mathbf{I})^T \mathbf{X}_{i1} + \mathbf{X}_{i1}(\mathbf{S} + 0.5\mathbf{I}) = \mathbf{Y}_{i1} \quad \text{for } i = 1, 2 \tag{70b}$$

$$\mathbf{S}^T \mathbf{X}_{i2} + \mathbf{X}_{i2}\mathbf{S} = \mathbf{Y}_{i2} \quad \text{for } i = 2 \tag{70c}$$

The coefficient matrices \mathbf{E}_0 , \mathbf{E}_1 and \mathbf{E}_2 are then computed as

$$\mathbf{E}_0 = \mathbf{T}^T \mathbf{U}^T \mathbf{X}_{00} \mathbf{U} \mathbf{T} \tag{71a}$$

$$\mathbf{E}_1 = \mathbf{T}^T \mathbf{U}^T (\mathbf{X}_{10} + \mathbf{X}_{11}) \mathbf{U} \mathbf{T} \tag{71b}$$

$$\mathbf{E}_2 = \mathbf{T}^T \mathbf{U}^T (\mathbf{X}_{20} + \mathbf{X}_{21} + \mathbf{X}_{22}) \mathbf{U} \mathbf{T} \tag{71c}$$

They can be assembled for each polyhedron, considering the contributions from each of the tetrahedral sectors that form the polygonal facets on the polyhedron boundary.

4.3 Solution for vanishing body loads

The static stiffness matrix is obtained from the solution of Eq. (64) without the presence of body loads

$$\mathbf{E}_0 \xi^2 \mathbf{u}_h(\xi)_{,\xi\xi} + (2\mathbf{E}_0 - \mathbf{E}_1 + \mathbf{E}_1^T) \xi \mathbf{u}_h(\xi)_{,\xi} + (\mathbf{E}_1^T - \mathbf{E}_2) \mathbf{u}_h(\xi) = \mathbf{0} \tag{72}$$

This second order differential equation is first transformed to a first order differential equation with twice the number of unknowns [50]

$$\xi \begin{Bmatrix} \xi^{0.5} \mathbf{u}_h(\xi) \\ \xi^{-0.5} \mathbf{q}_h(\xi) \end{Bmatrix}_{,\xi} = -\mathbf{Z} \begin{Bmatrix} \mathbf{u}_h(\xi) \\ \mathbf{q}_h(\xi) \end{Bmatrix} \tag{73}$$

where \mathbf{Z} is the Hamiltonian matrix

$$\mathbf{Z} = \begin{bmatrix} \mathbf{E}_0^{-1} \mathbf{E}_1^T - 0.5\mathbf{I} & -\mathbf{E}_0^{-1} \\ -\mathbf{E}_2 + \mathbf{E}_1 \mathbf{E}_0^{-1} \mathbf{E}_1^T & -(\mathbf{E}_1 \mathbf{E}_0 - 0.5\mathbf{I}) \end{bmatrix} \tag{74}$$

with identity matrix \mathbf{I} and $\mathbf{q}_h(\xi)$ is the internal nodal force functions along the radial line ξ

$$\mathbf{q}_h(\xi) = \mathbf{E}_0 \xi^2 \mathbf{u}_h(\xi)_{,\xi} + \mathbf{E}_1^T \xi \mathbf{u}_h(\xi) \tag{75}$$

The Hamiltonian matrix can be decomposed into pairs of eigenvalues using either an eigenvalue decomposition or a Schur decomposition as [48]

$$\mathbf{Z}\Psi = \Psi\Lambda \tag{76}$$

where Λ is a diagonal or block diagonal matrix (depending on the decomposition of \mathbf{Z} i.e. eigenvalue or Schur) containing the eigenvalues of \mathbf{Z} . The transformation matrix Ψ

contains the eigenvectors of Λ . The matrices Λ and Ψ can be conformably partitioned as

$$\Lambda = \begin{bmatrix} -\Lambda_n & * \\ & -\Lambda_p \end{bmatrix} \tag{77}$$

$$\Psi = \begin{bmatrix} \Psi_n^{(u)} & \Psi_p^{(u)} \\ \Psi_n^{(q)} & \Psi_p^{(q)} \end{bmatrix} \tag{78}$$

The solution of Eq. (73) is therefore

$$\begin{Bmatrix} \mathbf{u}_h(\xi) \\ \mathbf{q}_h(\xi) \end{Bmatrix} = \begin{bmatrix} \Psi_n^{(u)} & \Psi_p^{(u)} \\ \Psi_n^{(q)} & \Psi_p^{(q)} \end{bmatrix} \xi \begin{bmatrix} -\Lambda_n - 0.5\mathbf{I} & * \\ & -\Lambda_p + 0.5\mathbf{I} \end{bmatrix} \begin{Bmatrix} \mathbf{c}_n \\ \mathbf{c}_p \end{Bmatrix} \tag{79}$$

with integration constants \mathbf{c}_n and \mathbf{c}_p . For bounded polyhedrons such as those considered in this study, the condition of finiteness of the displacement at $\xi = 0$ is satisfied by setting $\mathbf{c}_p = 0$. This leads to

$$\mathbf{u}_h(\xi) = \Psi_n^{(u)} \xi^{-\Lambda_n - 0.5\mathbf{I}} \mathbf{c}_n \tag{80}$$

$$\mathbf{q}_h(\xi) = \Psi_n^{(q)} \xi^{-\Lambda_n + 0.5\mathbf{I}} \mathbf{c}_n \tag{81}$$

The integration constants \mathbf{c}_n are determined from the nodal displacements $\mathbf{u}_b = \mathbf{u}_h(\xi = 1)$ of a polyhedron as

$$\mathbf{c}_n = (\Psi_n^{(u)})^{-1} \mathbf{u}_b \tag{82}$$

The stiffness matrix of a polyhedron \mathbf{K}_{pol} is obtained from Eqs. (80) and (81) as

$$\mathbf{K}_{pol} = \Psi_n^{(q)} (\Psi_n^{(u)})^{-1} \tag{83}$$

The complete displacement field in a tetrahedral sector as shown in Fig. 2 is obtained by substituting Eq. (80) into Eq. (40) resulting in

$$\mathbf{u}(\xi, \eta, \zeta) = \Phi(\eta, \zeta) \mathbf{T} \mathbf{u}_h(\xi) \tag{84}$$

Accordingly, the stress field can be expressed as

$$\sigma(\xi, \eta, \zeta) = \mathbf{D} \Psi_\epsilon(\eta, \zeta) \xi^{-\Lambda_n - 1.5\mathbf{I}} \mathbf{c}_n \tag{85}$$

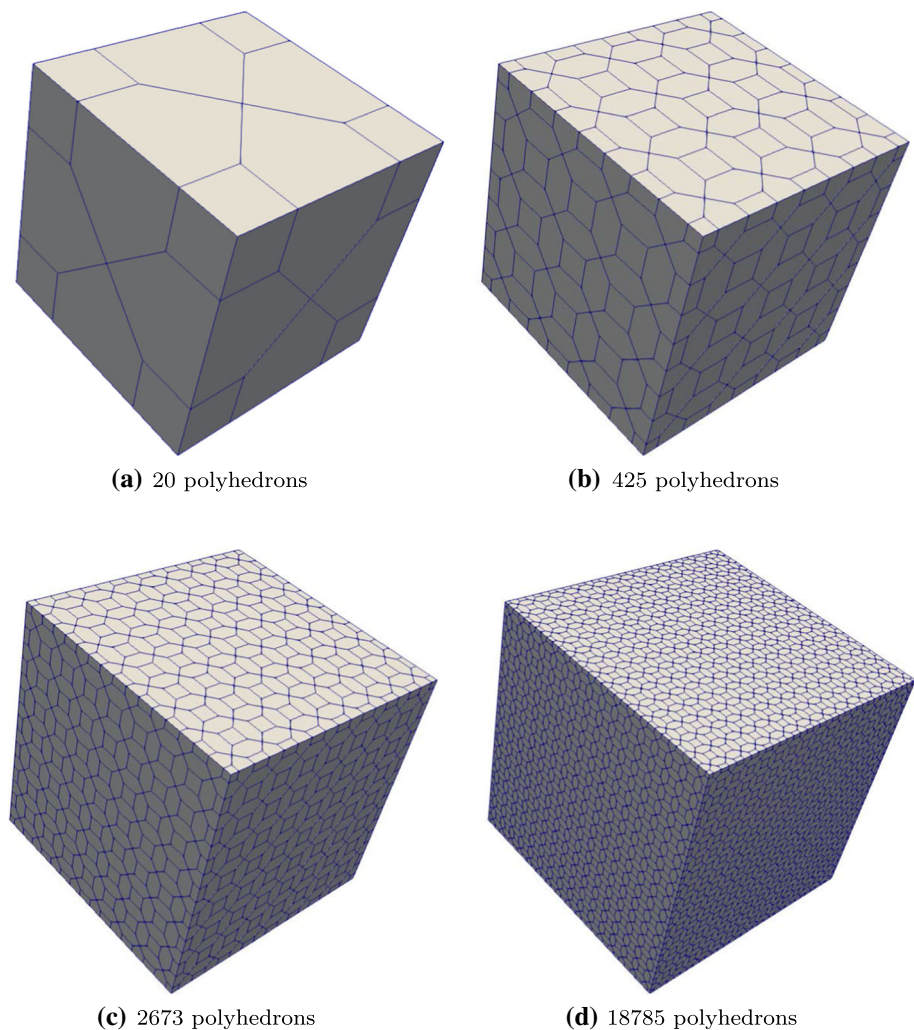
where the strain mode $\Psi_\epsilon(\eta, \zeta)$ is

$$\Psi_\epsilon(\eta, \zeta) = \mathbf{B}_1(\eta, \zeta) \mathbf{T} \Psi_n^{(u)} (-\Lambda_n - 0.5\mathbf{I}) + \mathbf{B}_2(\eta, \zeta) \mathbf{T} \Psi_n^{(u)} \tag{86}$$

Remark

Although the formulation is presented with reference to arbitrary faceted polyhedra, it is directly applicable to octree

Fig. 5 Polyhedron meshes for three-dimensional patch test



meshes, which can be viewed as a special case of arbitrary faceted polyhedra. In a three-dimensional analysis, discretisation using octree meshes has the advantage of rapid adaptive modelling, particularly for problems exhibiting complex geometries [58]. The flexibility to assume arbitrary shaped facets removes the numerical difficulty introduced by the hanging nodes, which would otherwise require additional treatment in conventional methods such as the FEM. Like the approach advocated by Zou et al. [58], the formulation also improves over the octree-type decomposition of Saputra et al. [47] and Liu et al. [34] by eliminating the need of introducing additional nodes and faces on the octree cells. Moreover, the use of semi-analytical integration over the facets of the polyhedra in this work eliminates the need for special integration rules associated with the application of irrational functions-based interpolants as in the work of Natarajan et al. [38] and Zou et al. [58]. The use of octree meshes also lead to additional advantages with regard to the computation of the stiffness matrices by exploiting the limited number of cell types in an octree decomposition [34,47,58].

5 Numerical examples

This section presents the application of the formulation to a few numerical benchmarks for the purposes of validation. The majority of the numerical examples employ a combination of octree and polyhedron meshes for the purpose of discretisation.

5.1 Patch test

A unit cuboid ($1\text{ m} \times 1\text{ m} \times 1\text{ m}$) is considered for the patch tests. Two types of patch tests are performed: (1) a constant strain patch test and (2) a high order patch test. Figure 5 shows the polyhedral meshes that are used for the patch tests.

5.1.1 Constant strain patch test

A constant strain patch test is considered first. The mesh in Fig. 5a is adopted. The material properties are: Young's modulus $E = 1 \times 10^3$ kPa and Poisson's ratio $\nu = 0.25$. The

Table 1 Displacements at interior nodes for three-dimensional patch test

Node	Analytical			Present		
	<i>u</i>	<i>v</i>	<i>w</i>	<i>u</i>	<i>v</i>	<i>w</i>
1(0.75, 0.25, 0.25)	0.1000	0.0750	0.0750	0.1000	0.0750	0.0750
2(0.25, 0.25, 0.75)	0.0750	0.0750	0.1000	0.0750	0.0750	0.1000
5(0.75, 0.75, 0.25)	0.1250	0.1250	0.1000	0.1250	0.1250	0.1000
6(0.75, 0.75, 0.75)	0.1500	0.1500	0.1500	0.1500	0.1500	0.1500
7(0.75, 0.25, 0.75)	0.1250	0.1000	0.1250	0.1250	0.1000	0.1250
8(0.50, 0.50, 0.75)	0.1125	0.1125	0.1250	0.1125	0.1125	0.1250
9(0.25, 0.75, 0.25)	0.0750	0.1000	0.0750	0.0750	0.1000	0.0750
10(0.50, 0.50, 0.25)	0.0875	0.0875	0.0750	0.0875	0.0875	0.0750
11(0.25, 0.25, 0.25)	0.0500	0.0500	0.0500	0.0500	0.0500	0.0500
13(0.25, 0.75, 0.75)	0.1000	0.1250	0.1250	0.1000	0.1250	0.1250

objective of the patch test is to examine the convergence of the developed method when subjected to a linear displacement field imposed along the boundaries of the cuboid. The boundary conditions are prescribed at the exterior nodes according to

$$\begin{aligned}
 u &= 5(2x + y + z) \times 10^{-2} \text{ m} \\
 v &= 5(x + 2y + z) \times 10^{-2} \text{ m} \\
 w &= 5(x + y + 2z) \times 10^{-2} \text{ m}
 \end{aligned}$$

Table 1 compares the computed displacements at the interior nodes with the analytical solution. It is observed that the exact values can be reproduced to machine precision and therefore the proposed technique passes the patch test. This is an indication that the convergence of the proposed technique is ensured with mesh refinement.

5.1.2 High order patch test

A high order patch test is considered next. A consistent displacement field is derived from the solution of the Navier equations of equilibrium following the approach outlined in Rajendran et al. [43]. Displacement boundary conditions are prescribed at the exterior nodes according to

$$\begin{aligned}
 u &= 3x^2 - 8xy + y^2 \text{ m} \\
 v &= 3y^2 - 8yz + z^2 \text{ m} \\
 w &= 3z^2 - 8xz + x^2 \text{ m}
 \end{aligned}$$

The Young’s modulus $E = 8.25 \text{ kPa}$ and the Poisson’s ratio $\nu = 0.375$. As only linear shape functions are adopted to discretize the line elements on the boundary of the polyhedron, the scaled boundary finite element formulation developed in this paper is not high order complete. A convergence study is therefore performed using the meshes shown in Fig. 5 Amongst the available approaches over polytopes,

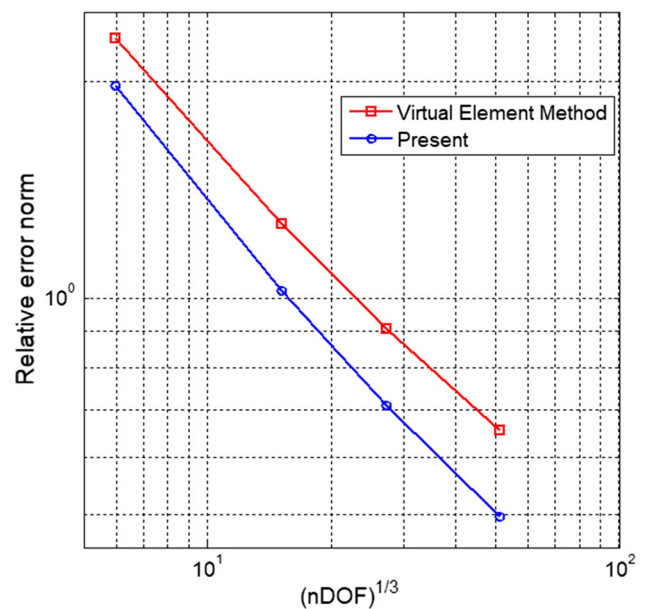


Fig. 6 Plots of convergence of norm of error of displacements for high order patch test

in this study, the accuracy and the convergence behaviours of the scaled boundary finite element formulation is compared to that of the VEM.

Figure 6 shows the convergence plot of the displacement error norm defined as $\|\mathbf{u}_{numer} - \mathbf{u}_{anal}\|_{L_2}$. It is observed from Fig. 6 that for the same mesh, the SBFEM is slightly more accurate than the VEM. The convergence rate of the SBFEM is 0.6443. The convergence rate of the VEM is 0.5871. The improved convergence and accuracy may come at the expense of the additional effort required in the eigenvalue solution required to compute the surface shape functions and the stiffness matrix over each polyhedron. However, this computational aspect may be improved by adopting octree meshes in the computations.

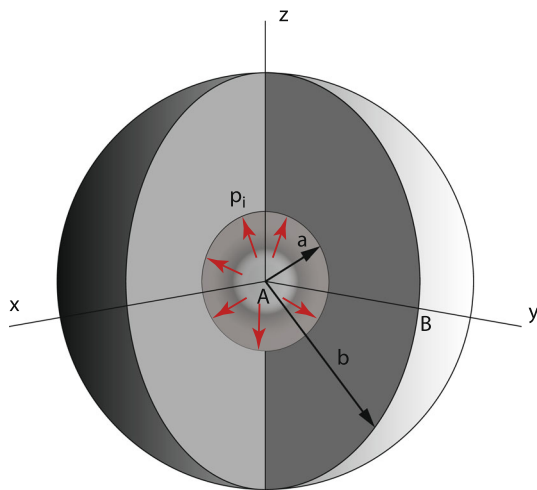


Fig. 7 Pressurised hollow sphere

5.2 Pressurised hollow sphere

The problem of a pressurised hollow sphere shown in Fig. 7 is considered. The analytical solution of the displacements and stresses are reported by Bower [8] in spherical coordinates (r, θ, ϕ) as

$$u_r(r, \theta, \phi) = \frac{p_i a^3}{2E(b^3 - a^3)r^2} \{2(1 - 2\nu)r^3 + (1 + \nu)b^3\}$$

$$u_\theta(r, \theta, \phi) = u_\phi(r, \theta, \phi) = 0$$

$$\sigma_{rr}(r, \theta, \phi) = \frac{p_i a^3 (r^3 - b^3)}{(b^3 - a^3)r^3}$$

$$\sigma_{\theta\theta}(r, \theta, \phi) = \sigma_{\phi\phi}(r, \theta, \phi) = \frac{p_i a^3 (2r^3 + b^3)}{2(b^3 - a^3)r^3}$$

$$\sigma_{r\theta}(r, \theta, \phi) = \sigma_{r\phi}(r, \theta, \phi) = \sigma_{\theta\phi}(r, \theta, \phi) = 0$$

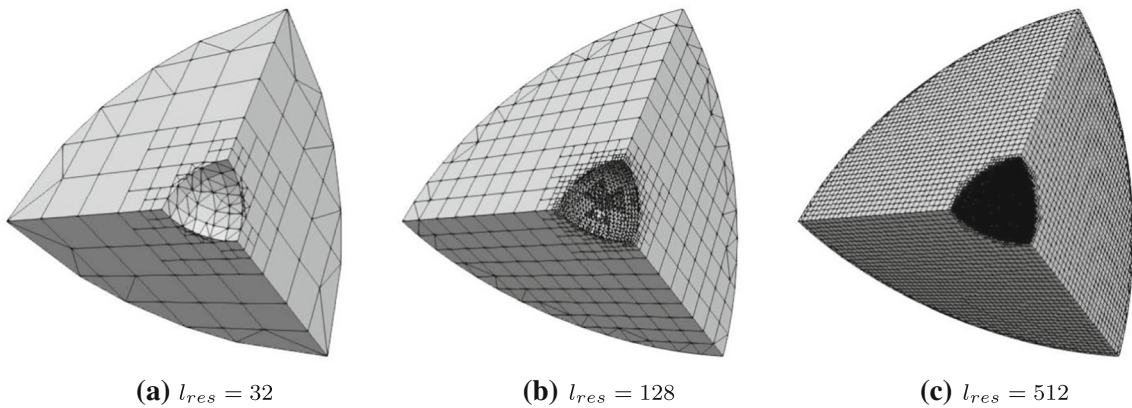


Fig. 8 Hybrid octree-polyhedron meshes of an octant of the sphere

Fig. 9 Plots of radial displacement component u_r along line AB and L_2 - and H_1 - norms for pressurised hollow sphere

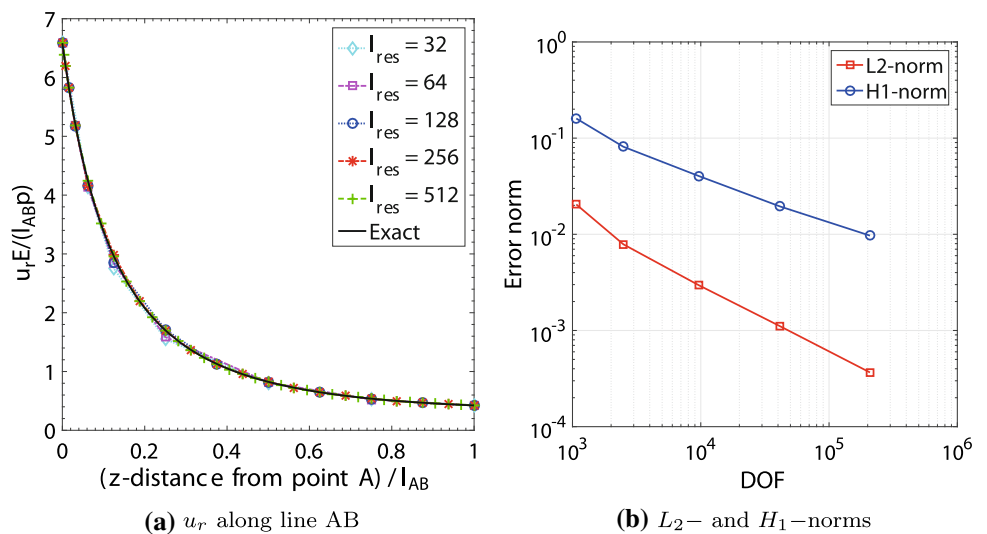
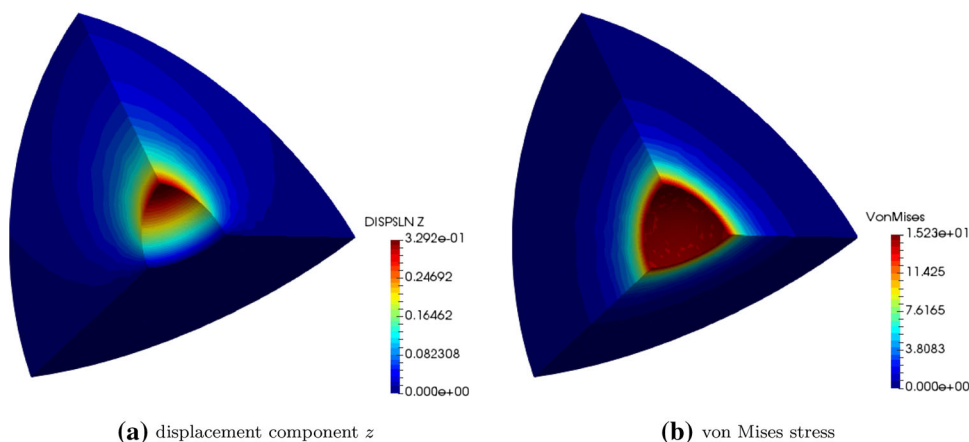


Fig. 10 Contour plot of magnitude of displacement and von Mises stress of pressurised hollow sphere



where p_i is the internal pressure, a is the internal radius, b is the external radius, E is the Young’s modulus and ν is the Poisson’s ratio. The parameters selected for validation are: $E = 200$ GPa, $\nu = 0.3$, $a = 10$ mm, $b = 50$ mm and $p_i = 6$ GPa.

Considering the symmetry of the geometry and the loads, only an octant of the sphere is modelled. The sphere is discretised using a combination of octree cells and arbitrary faceted polyhedrons. On the internal and external surfaces of the sphere, boundary conditions are specified according to the analytical solution. Additionally, symmetric boundary conditions are prescribed on the planes of symmetry. The mesh generation algorithm is augmented from Saputra et al. [47] and Liu et al. [34]. The key feature in this development is the elimination of the triangulation process required for the octree cells containing hanging nodes. A bounding box of $80 \times 80 \times 80$ mm³ is used to construct the mesh. During mesh generation the ratio of the maximum to the minimum size of the octree cells ($\frac{h_{max}}{h_{min}}$) is kept constant at 8. The density of the mesh is controlled by specifying h_{res} , which scales the physical size of the sphere to an octree cell as $h_{min} = \frac{b}{l_{res}}$ mm. Along the curved boundaries of the sphere, the surfaces are triangulated in order to maintain the planarity condition required by the surfaces of the polyhedron and also to better represent the curvature of the sphere. Figure 8 shows the mesh of one octant of the sphere for $l_{res} = 32$, $l_{res} = 128$ and $l_{res} = 512$.

Figure 9a compares the computed radial displacement component u_r along the line AB shown in Fig. 7 for the five meshes generated with the analytical solution. It can be observed from Fig. 9a that the computed displacements agree well with the analytical solution. The convergence of the L_2 - and H_1 -norms computed are shown in Fig. 9b. It is observed that both the error norms decrease as the mesh density increases. The convergence rates for the L_2 - and H_1 -norms are -0.742 and -0.521 , respectively. The theoretical convergence rate using uniform mesh refinement, which are $-2/3$ and $-1/3$, respectively. Figure 10 shows the con-

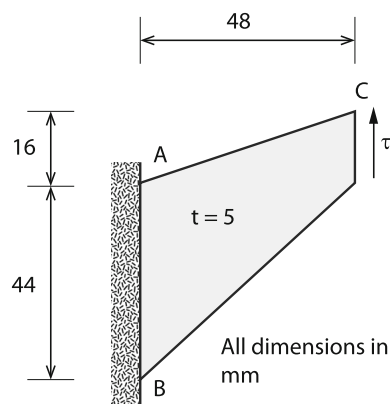


Fig. 11 Cook’s membrane problem

tour plots of the displacement magnitudes and the von Mises stress of the hollow sphere. As is expected, the contour of the von Mises stress exhibits radial symmetry, which is reflective of the symmetric geometry and boundary conditions of the problem.

5.3 Cook’s membrane

The Cook’s membrane shown in Fig. 11 is investigated as a three-dimensional problem. The material properties of the membrane are: Young’s modulus $E = 1000$ Nmm⁻² and Poisson’s ratio $\nu = 0.33$. A vertical distributed load $\tau = 10$ Nmm⁻² is applied on the right edge of the membrane. The membrane is constrained from motion on its left edge. A hybrid octree polyhedron mesh of the membrane is generated starting from a pure octree mesh. The density of the meshes is controlled by specifying the resolution h_{res} used during voxel generation. The three meshes adopted in this study are shown in Fig. 12.

Table 2 shows the y -displacement component of point C shown in Fig. 11 at the mid z -plane of the membrane. When computed displacements exhibited a converging trend as the mesh density is increased. The results compare well

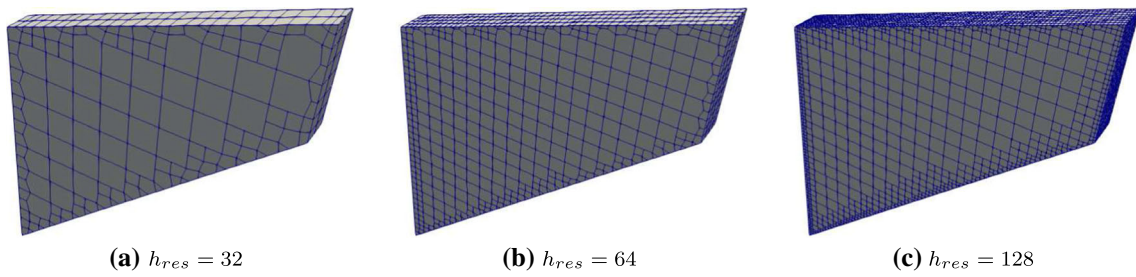


Fig. 12 Hybrid octree polyhedron meshes used for Cook’s membrane problem

Table 2 y –displacement component at point C

h_{res}	Elements	Nodes	DOF	y –displacement (mm)
32	273	642	1926	3.9058
64	1394	2552	7656	3.9620
128	5576	9392	28176	3.9826

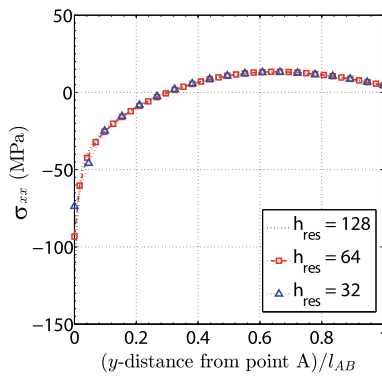


Fig. 13 Distribution of σ_{xx} along line AB of Cook’s membrane

with the reference solution obtained from the commercial FEM software ABAQUS using a very fine mesh consisting of 66700 brick elements which was 4.00856 mm.

Figure 13 shows the distribution of the stress component σ_{xx} along line AB of the Cook’s membrane. Similar to the displacements, the computed stresses also show a conver-

Fig. 14 Contour plots of displacement magnitude and stress σ_{xx} of Cook’s membrane

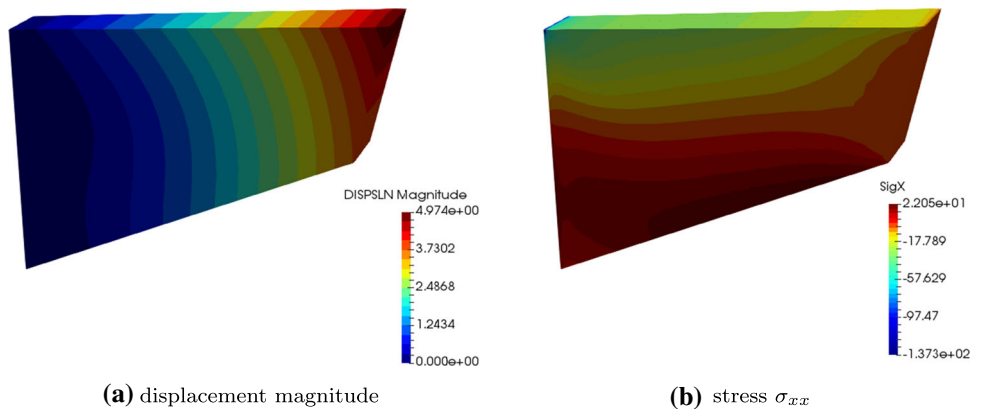
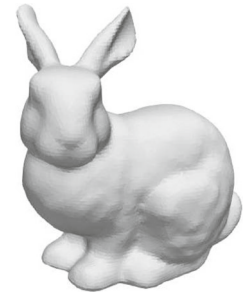


Fig. 15 STL model of a sculpture of a rabbit



gent trend as the results computed from the coarser meshes approach those of the finer. Figure 14 shows the contour plots of the magnitude of displacement and the stress component σ_{xx} of the membrane. As is expected, the displacements are highest in magnitude in the vicinity of the applied traction. The membrane exhibits a region of tensile stress, which is maximum near mid-length of the membrane as is expected due to the nature of the load τ in the problem.

5.4 STL model of sculpture of rabbit under self-weight

The STL model of a sculpture of a rabbit shown in Fig. 15 is considered. The model is bounded by a cuboid with dimensions 100 cm \times 98.7 cm \times 76.9 cm. The sculpture is assumed to be fixed at its base and undergoes deformation due to its self-weight. The material properties of the sculpture are: Young’s modulus $E = 17$ GPa, Poisson’s ratio $\nu = 0.15$ and

Fig. 16 Hybrid-octree-polyhedron mesh of sculpture of rabbit

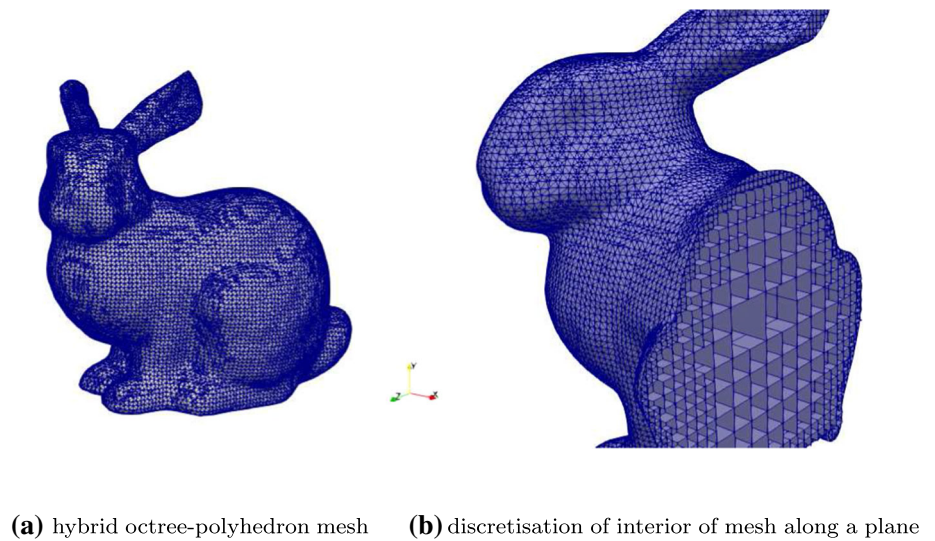
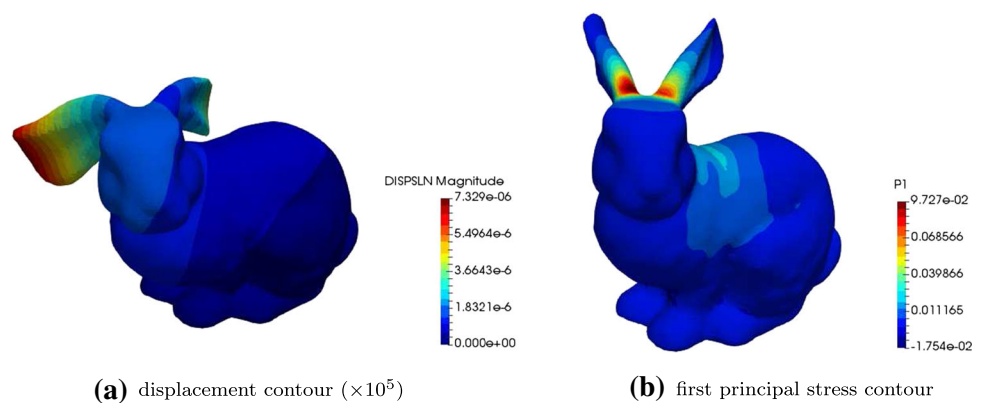


Fig. 17 Contour plots of sculpture of rabbit deforming under self weight



density 2400 kgm^{-3} . The gravitational acceleration is taken to be $g = 9.81 \text{ ms}^{-2}$.

The model is meshed with a resolution of $h_{res} = 128$. The ratio of the maximum and minimum cell size is $h_{max} : h_{min} = 8 : 1$. The corresponding mesh of the sculpture is shown in Fig. 16. It can be observed from Fig. 16a that along the exterior of the model, the discretisation is of the form of polyhedrons with arbitrary number of faces. Within the interior and away from the boundary of the sculpture as is observed in Fig. 16b, the majority of the cells in the mesh belong to the generic cell types in a balanced octree discretisation [58].

Figure 17 shows the contour plots of the displacement magnitude and first principal stress of the sculpture of the rabbit. The tips of the ears of the sculpture were observed to have the largest deformation and are an expected result as the ears form a cantilever-like structure supported at the head of the rabbit. Regions of high magnitude of tensile stresses were observed at the head of the sculpture near the base of the ears as is shown in Fig. 17b and is a consequence of the bending-like deformation of both ears under self-weight.

5.5 Image-based analysis of concrete mesostructure subjected to uniaxial tension

Figure 18a shows 3D image of part of a concrete mesostructure generated from images obtained from an X-ray computed tomography scan analysed by Huang et al. [26]. The aggregates in the meso-structure are represented in black whereas the mortar is represented in grey. The image is of size $64 \times 64 \times 64$ voxels. The size of each voxel is $0.08 \text{ mm} \times 0.08 \text{ mm} \times 0.08 \text{ mm}$. A mesh consisting of purely octree cells is used to discretize the image. The ratio of the maximum and minimum cell size adopted for the discretization is $h_{max} : h_{min} = 8:1$. Figure 18b shows the octree mesh of the meso-structure, which comprises of 46187 cells and 71402 nodes. It is noticeable in Fig. 18b that the mesh is made up entirely of octree cells without the need for further triangularization like in the work of Saputra et al. [47] and Talebi et al. [52].

A stress analysis is carried out to simulate a uniaxial tensile strain of 1×10^{-5} by prescribing the displacement components on the top and bottom surfaces of the meso-structure (in the z -direction). The material properties assumed for

Fig. 18 Mesh of concrete meso-structure

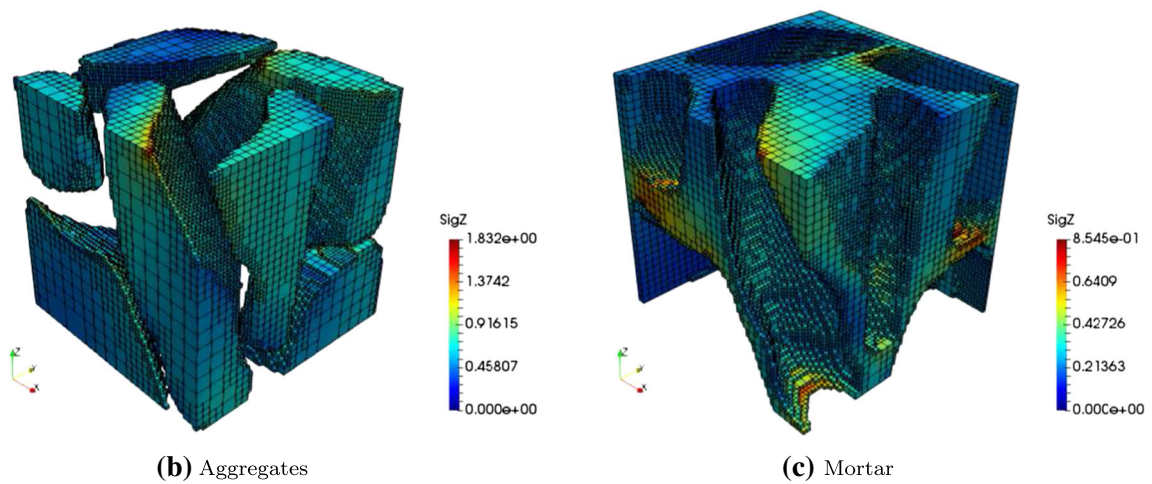
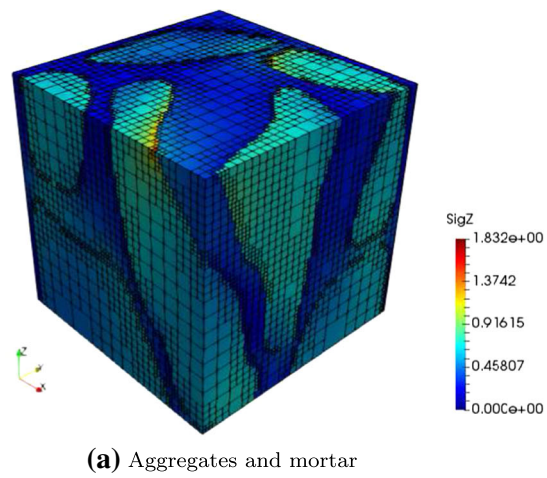
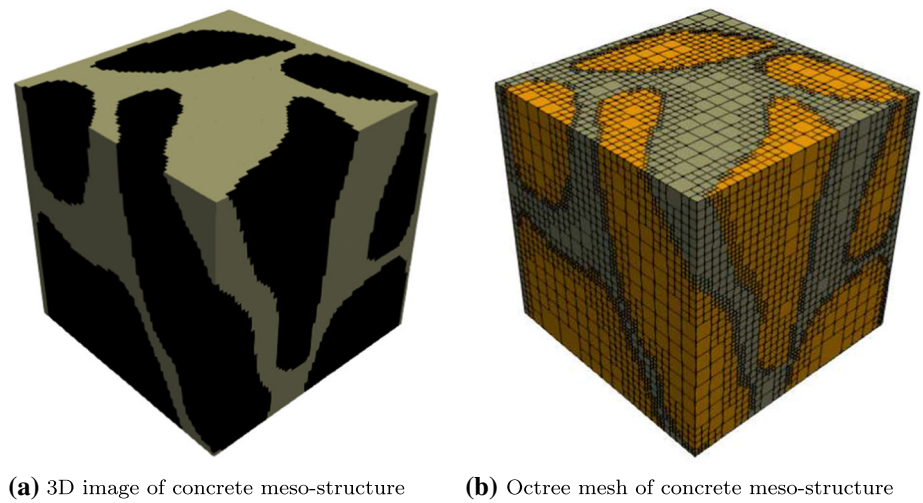
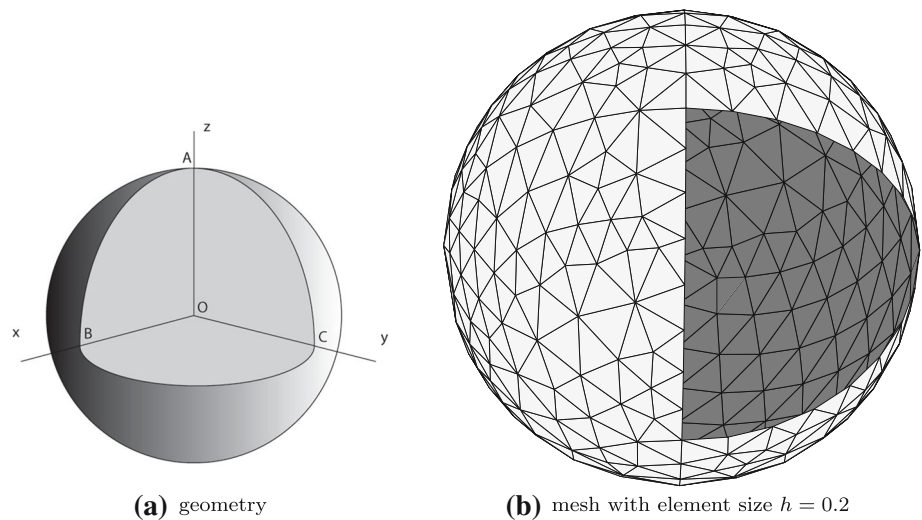


Fig. 19 Stress contour plot of concrete meso -structure

Fig. 20 Fichera’s corner



the analysis are as follows. The Young’s modulus of aggregate $E_a = 70$ GPa. The Young’s modulus of mortar $E_m = 25$ GPa. The Poisson’s ratio is assumed to be $\nu = 0.2$ for both the aggregate and the mortar. Figure 19a shows the contour plot of the stress σ_{zz} of the concrete meso-structure. Fig. 19b, c show the σ_{zz} -contour plot for the aggregate and mortar phases separately. Regions of high stresses can be observed in the vicinity of the interfaces between the aggregates and the mortar where the dissimilarity of the material properties between both phases generates stress concentrations. This reflects the observations of the response of concrete meso-structure made in the literature (e.g. Wang et al. [54]; Rodrigues et al. [46]) where cracks tend to nucleate along the aggregate-mortar interfaces due to the high stress concentration in these regions.

5.6 Three-dimensional stress singularity: Fichera’s corner

The three-dimensional stress singularity problem defined by Fichera’s corner shown in Fig. 20 is considered. The surfaces of the corner defined by three mutually orthogonal planes OAB , OAC and OBC are stress-free. The corner is embedded at the centre of a sphere of radius $r = 1$. The sphere is assumed to be homogeneous with Poisson’s ratio $\nu = 0.3$ and Young’s modulus $E = 1$.

The sphere is modelled using only one polyhedron with scaling centre at O . The orthogonal planes are not discretized. The surface of the sphere is discretized by lines forming triangular sectors of size h as shown in Fig. 20b. Three mesh sizes are considered i.e. $h = 0.2$, $h = 0.1$ and $h = 0.05$.

The order of the mode-I stress singularity defined as $([\lambda^R] - 1)$ is computed using the developed formulation [24]. Table 3 compares the computed results. The stress singular-

Table 3 Three-dimensional stress singularities for Fichera’s corner

Method	Number of nodes	$([\lambda^R] - 1)$
FEM [17]	N/A	-0.238329
SBFEM [24]	N/A	-0.230313
Present $h = 0.2$	337	-0.226018
Present $h = 0.1$	1611	-0.232455
Present $h = 0.05$	4518	-0.235890

ities computed using the FEM [17] and the original SBFEM [24] are reported for comparison. It is observed that the computed stress singularities are close to the results reported in the literature. The developed technique therefore, preserves the advantages of the SBFEM in computing the order of stress singularities inherent in the original SBFEM.

6 Conclusions

A novel SBFEM formulation has been developed for three-dimensional analysis. The formulation adopts a novel approach in which planar shape functions for a scalar field are derived over the facets of each polyhedron from the semi-analytical solution of the Laplace equation in a transformed two-dimensional coordinate system. The resulting shape functions are then adopted in the SBFEM in a formulation for elasto-static analyses. The subsequent processes in the formulation are similar to standard SBFEM procedures with the exception that the isoparametric (or Wachspress [38,58]) shape functions are replaced by the semi-analytical scaled boundary shape functions. The resulting coefficient matrices can be integrated in both the scaling directions ζ and ξ . Numerical integration is required only along the edges of each polyhedron. This approach removes the need to further

partition the faces of the octree cells into triangles [34,47] or to adopt special integration techniques associated with the rational polygonal interpolants over the surface [58].

The developed formulation was validated using six numerical benchmarks. A discretisation that adopts a combination of octree and polyhedron cells was adopted. The combined use of octree and polyhedron cells leads to significant reduction in the number of cells in a mesh due to the efficiency of the octree decomposition in transitioning across different scales [34,47,58]. It was shown that the developed formulation can pass the patch test. The convergence behaviour of the formulation was also demonstrated for a case where analytical solution is available. Two additional numerical examples show the application of the formulation in conjunction with STL models and image-based analyses.

References

- Aldakheel F, Hudobivnik B, Wriggers P (2019a) A low order 3D virtual element formulation for finite elasto-plastic deformations. *Comput Mech* 63:253–269
- Aldakheel F, Hudobivnik B, Wriggers P (2019b) Virtual elements for the finite thermo-plasticity problems. *Comput Mech* 64:1347–1360
- Atluri SN, Zhu T (1998) A new Meshless Local Petrov–Galerkin (MLPG) approach in computational mechanics. *Comput Mech* 22(2):117–127
- Babúska I, Banerjee U, Osborn JE (2003) Survey of meshless and generalized finite element methods: a unified approach. *Acta Numer* 12:1–125
- Belytschko T, Gu YY, Gu L (1994) Element-free Galerkin methods. *Int J Numer Methods Eng* 37(2):229–256
- Belytschko T, Lu YY, Gu L (1995) Crack propagation by element-free Galerkin methods. *Eng Fract Mech* 51(2):295–315
- Bishop JE (2014) A displacement-based finite element formulation for general polyhedra using harmonic shape functions. *Int J Numer Methods Eng* 97:1–31
- Bower AF (2009) *Appl Mech Solids*. CRC Press, Boca Raton
- Brezzi F, Lipnikov K, Simoncini V (2005) A family of mimetic finite difference methods on polygonal and polyhedral meshes. *Math Models Methods Appl Sci* 15(10):1533–1551
- Brezzi F, Lipnikov K, Shashkov M, Simoncini V (2007) A new discretization methodology for diffusion problems on generalized polyhedral meshes. *Comput Methods Appl Mech Eng* 196:3682–3692
- Cangiani A, Dong Z, Georgoulis EH, Houston P (2017) *hp-version discontinuous galerkin methods on polygonal and polyhedral meshes*. Springer, Cham
- Cardiff P, Tuković Ž, Jasak H, Ivanković A (2016) A block-coupled Finite Volume method for linear elasticity and unstructured meshes. *Comput Struct* 175:100–122
- Chen JS, Wu CT, Yoon S, You Y (2001) A stabilized conforming nodal integration for Galerkin mesh-free methods. *Int J Numer Methods Eng* 50:435–466
- Chi H, da Veiga LB, Paulino GH (2017) Some basic formulations of the virtual element method (VEM) for finite deformations. *Comput Methods Appl Mech Eng* 318:148–192
- da Veiga LB, Brezzi F, Cangiani A, Manzini G, Marini LD, Russo A (2013) Basic principles of virtual element methods. *Math Models Methods Appl Sci* 23(1):199–214
- da Veiga LB, Dassi F, Russo A (2017) High-order virtual element method on polyhedral meshes. *Comput Math Appl* 74:1110–1122
- Dimitrov A, Andra H, Schnack E (2001) Efficient computation of order and mode of corner singularities in 3D-elasticity. *Int J Numer Methods Eng* 52:805–827
- Dvorak P (2006) New element lops time off CFD simulations. *Mach Des* 78:154–155
- Floater M (2003) Mean value coordinates. *Comput Aided Geom Des* 20:19–27
- Francis A, Ortiz-Bernardin A, Bordas SPA, Natarajan S (2017) Linear smoothed polygonal and polyhedral finite elements. *Int J Numer Methods Eng* 109:1263–1288
- Gain AL, Talischi C, Paulino GH (2014) On the virtual element method for three-dimensional linear elasticity problems on arbitrary polyhedral meshes. *Comput Methods Appl Mech Eng* 282:132–160
- Ghosh S, Moorthy S (2004) Three dimensional Voronoi cell finite element model for microstructures with ellipsoidal heterogeneities. *Comput Mech* 34:510–531
- Gingold RA, Monaghan JJ (1977) Smoothed particle hydrodynamics—theory and application to non-spherical stars. *Mon Not R Astron Soc* 181:375–389
- Goswami S, Becker W (2012) Computation of 3-D stress singularities for multiple crack and crack intersections by the scaled boundary finite element method. *Comput Mech* 175:13–25
- Hillman M, Chen JS (2016) An accelerated, convergent, and stable nodal integration in Galerkin meshfree methods for linear and nonlinear mechanics. *Int J Numer Methods Eng* 107(7):603–630
- Huang Y, Yang Z, Ren W, Liu G, Zhang C (2015) 3D meso-scale fracture modelling and validation of concrete based on in-situ X-ray computed tomography images using damage plasticity model. *Int J Solids Struct* 67–68:340–352
- Joshi P, Meyer M, Rose T, De, Green B, Sanocki T (2007) Harmonic coordinates for character articulation. *ACM Trans Graph* 26(3), 71
- Kraus M, Steinmann P (2012) Finite element formulation for 3D convex polyhedra in nonlinear continuum mechanics. *Comput Assist Methods Eng Sci* 19:121–134
- Lee C, Kim H, Im S (2016) Polyhedral elements by means of node/edge-based smoothed finite element method. *Int J Numer Methods Eng* 110:1069–1100. <https://doi.org/10.1002/nme.5449>
- Liszka TJ, Duarte CAM, Tworzydło WW (1996) hp-Meshless cloud method. *Comput Methods Appl Mech Eng* 139:263–288
- Liu GR, Gu YT (2001) A point interpolation method for two-dimensional solids. *Int J Numer Methods Eng* 50:937–951
- Liu GR, Nguyen TT, Dai KY, Lam KY (2007) Theoretical aspects of the smoothed finite element method (SFEM). *Int J Numer Methods Eng* 71:902–930
- Liu WK, Jun S, Zhang YF (1995) Reproducing kernel particle methods. *Int J Numer Methods Fluids* 20(8–9):1081–1106
- Liu Y, Saputra AA, Wang J, Tin-Loi F, Song C (2017) Automatic polyhedral mesh generation and scaled boundary finite element analysis of STL model. *Comput Methods Appl Mech Eng* 313:106–132
- Lucy LB (1977) A numerical approach to the testing of the fission hypothesis. *Astron J* 82(12):1013–1024
- Meyer M, Lee H, Barr A, Desbrun M (2002) Generalized barycentric coordinates on irregular polygons. *J Graph Tools* 7:13–22
- Natarajan S, Bordas SPA, Ooi ET (2015) Virtual and smoothed finite elements: a connection and its application to polygonal/polyhedral finite element methods. *Int J Numer Methods Eng* 104:1173–1199
- Natarajan S, Ooi ET, Saputra A, Song C (2017) A scaled boundary finite element formulation over arbitrary faceted star convex polyhedra. *Eng Anal Bound Elem* 80:218–229
- Oliveira T, Portela A (2016) Weak-form collocation—a local meshless method in linear elasticity. *Eng Anal Bound Elem* 73:144–160

40. Ooi ET, Song C, Tin-Loi F, Yang ZJ (2012) Polygon scaled boundary finite elements for crack propagation modelling. *Int J Numer Methods Eng* 91(3):319–342
41. Ooi ET, Song C, Natarajan S (2018) A scaled boundary finite element method for poroelasticity. *Int J Numer Methods Eng* 114:905–929
42. Perumal L, Fadel MI (2014) New polyhedral elements based on virtual node method for solid mechanics and heat transfer applications. *Appl Mech Mater* 493:367–371
43. Rajendran S, Ooi ET, Yeo JH (2007) Mesh-distortion immunity assessment of QUAD8 elements by strong-form patch tests. *Commun Numer Methods Eng* 23:157–168
44. Rashid MM, Selimotic M (2006) A three-dimensional finite element method with arbitrary polyhedral elements. *Int J Numer Methods Eng* 67:226–252
45. Rjasanow S, Weiber S (2014) FEM with Trefftz trial functions on polyhedral elements. *J Comput Appl Math* 263:202–217
46. Rodrigues EA, Manzoli OL Jr, Bitencourt LAG, Bittencourt TN (2016) 2D mesoscale concrete model for concrete based on the use of interface element with high aspect ratio. *Int J Solids Struct* 94–95:112–124
47. Saputra A, Talebi H, Tran D, Birk C, Song C (2017) Automatic image-based stress analysis by the scaled boundary finite element method. *Int J Numer Methods Eng* 109(5):697–738
48. Song C (2004) A matrix function solution for the scaled boundary finite-element equation in statics. *Comput Methods Appl Mech Eng* 193(23–26):2325–2356
49. Song C, Wolf JP (1997) The scaled boundary finite element method—alias consistent infinitesimal finite element cell method for elastodynamics. *Comput Methods Appl Mech Eng* 147:329–355
50. Song C, Wolf JP (1999) Body loads in scaled boundary finite-element method. *Comput Methods Appl Mech Eng* 180(1–2):117–135
51. Sukumar N, Tabarraei A (2004) Conforming polygonal finite elements. *Int J Numer Methods Eng* 61(12):2045–2066
52. Talebi H, Saputra A, Song C (2016) Stress analysis of 3D complex geometries using the scaled boundary polyhedral finite elements. *Comput Mech* 58(4):697–715
53. Wachspress E (1975) A rational finite element basis. Academic Press, New York
54. Wang XF, Yang ZJ, Yates JR, Jivkov AP, Zhang C (2015) Monte carlo simulations of mesoscale fracture modelling of concrete with random aggregates and pores. *Constr Build Mater* 75:35–45
55. Wicke M, Botsch M, Gross M, Zurich ETH (2007) A finite element method on convex polyhedra. *EuroGraphics* 6(3):355–364
56. Wolf JP, Song C (2000) The scaled boundary finite-element method: a primer—derivations. *Comput Struct* 78(1–3):191–210
57. Wu CKC, Plesha ME (2002) Essential boundary condition enforcement in meshless methods: boundary flux collocation method. *Int J Numer Methods Eng* 53:499–514
58. Zou D, Chen K, Kong X, Liu J (2017) An enhanced octree polyhedral scaled boundary finite element method and its application in structure analysis. *Eng Anal Bound Elem* 84:87–107

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.