



ARTICLE

Development of a Three-Dimensional Multiscale Octree SBFEM for Viscoelastic Problems of Heterogeneous Materials

Xu Xu¹, Xiaoteng Wang¹, Haitian Yang¹, Zhenjun Yang² and Yiqian He^{1,3,*}

¹State Key Laboratory of Structural Analysis, Optimization and CAE Software for Industrial Equipment, Department of Engineering Mechanics, Dalian University of Technology, Dalian, 116024, China

²Hubei Key Laboratory of Geotechnical and Structural Safety, School of Civil Engineering, Wuhan University, Wuhan, 430072, China

³Mechanik–Materialtheorie, Ruhr-Universität Bochum, Bochum, 44801, Germany

*Corresponding Author: Yiqian He. Email: heyiqian@dlut.edu.cn

Received: 30 November 2023 Accepted: 06 March 2024 Published: 20 May 2024

ABSTRACT

The multiscale method provides an effective approach for the numerical analysis of heterogeneous viscoelastic materials by reducing the degree of freedoms (DOFs). A basic framework of the Multiscale Scaled Boundary Finite Element Method (MsSBFEM) was presented in our previous works, but those works only addressed two-dimensional problems. In order to solve more realistic problems, a three-dimensional MsSBFEM is further developed in this article. In the proposed method, the octree SBFEM is used to deal with the three-dimensional calculation for numerical base functions to bridge small and large scales, the three-dimensional image-based analysis can be conveniently conducted in small-scale and coarse nodes can be flexibly adjusted to improve the computational accuracy. Besides, the Temporally Piecewise Adaptive Algorithm (TPAA) is used to maintain the computational accuracy of multiscale analysis by adaptive calculation in time domain. The results of numerical examples show that the proposed method can significantly reduce the DOFs for three-dimensional viscoelastic analysis with good accuracy. For instance, the DOFs can be reduced by 9021 times compared with Direct Numerical Simulation (DNS) with an average error of 1.87% in the third example, and it is very effective in dealing with three-dimensional complex microstructures directly based on images without any geometric modelling process.

KEYWORDS

Three-dimensional multiscale viscoelastic analysis; numerical base functions; octree SBFEM; image-based analysis; temporally piecewise adaptive algorithm

Nomenclature

σ	Stress
ϵ	Strain
\mathbf{F}	Body force intensities
\mathbf{u}	Displacement
\mathbf{u}^E	Nodal displacement of the coarse element



\mathbf{u}^e	Nodal displacement of the fine element
Ω_E	Solution domain
Γ_u	Displacement boundary
Γ_σ	Stress boundary
t	Time
s	Dimensionless time parameter
t_{k-1}	Initial point k -th time interval
T_k	Size of k -th time interval
m	Expanding coefficient
\mathbf{D}	Elastic tensor
E_1	Elastic modulus
E_2	Elastic modulus
ν	Poisson's ratio
η_1	Viscosity coefficient
\mathbf{N}_n	Numerical base function
\mathbf{N}_{Iz}^i	Given node displacement of coarse element in z -direction
\mathbf{G}_i	Transfer matrix
\mathbf{K}_i^e	Stiffness matrix of the i -th fine element
\mathbf{K}_G	Integral structure stiffness matrix of large-scale
\mathbf{P}_G	Large-scale load vector
\mathbf{P}_j^E	Load vector of the j -th coarse element
$A_{j=1}^R$	Assembly operator
e_r	Relative error

1 Introduction

Heterogeneous viscoelastic materials, such as concrete [1] and polymers [2], are widely used in many fields because of their excellent properties. The macroscopic responses of these heterogeneous materials are known to be significantly influenced by physical phenomena at different scales and the interaction between different scales [3].

The Direct Numerical Simulation (DNS) method, usually implemented by the Finite Element Method (FEM) [4–6], is widely used to effectively determine the mechanical responses of heterogeneous viscoelastic materials. However, the DNS method could incur high computational costs and become impractical for numerical analysis of whole structures due to the tremendous amount of computer memory and CPU time needed to mesh all heterogeneities [7].

Compared with the DNS method, the multiscale methods can effectively reduce the degree of freedoms (DOFs) of numerical analysis on a coarse-scale mesh without resolving all small-scale features [8,9]. There are a large number of multiscale methods in literature, including analytical models and computational multiscale methods. The analytical models focus on obtaining equivalent performance on a large scale, including variational principles [8–10], self-consistent methods [11–13] and Mori-Tanaka methods [14–16], etc. Although these methods work well to predict the equivalent properties of heterogeneous materials with a small volume fraction and simple geometry, most of these methods contain over-ideal assumptions [17]. To address this issue, the numerical models were developed and became more popular, including asymptotic homogenization methods [18–20], computational homogenization methods [21–24] and FE² methods [25–28]. However, these methods have some limitations including the assumption of scale separation and periodicity. The multiscale

finite element method (MsFEM) presented by Babuška et al. [29,30] is another computational multiscale method. The basic idea of the MsFEM is to construct a bridge between different scales via the numerical base functions [31–33], and the MsFEM can deal with non-periodic continuum problems. The MsFEM was firstly used to solve scalar field problems [34] and has been successfully applied in various heterogeneous multiscale analyses related to dynamic problems [35], elastic-plastic problems [36], and thermal-mechanical coupling problems [37].

To reproduce more realistic situations for heterogeneous viscoelastic analysis, it is desirable to develop three-dimensional multiscale approaches under the framework of MsFEM, as the problems in practical applications are general in 3D cases. However, most of the aforementioned works on MsFEM are limited to two-dimensional cases. Klimczak et al. presented a three-dimensional MsFEM for viscoelastic problems, in which conventional FEM is used for the three-dimensional base function in small-scale [38]. However, the calculation of numerical base functions by conventional FEM could be very time-consuming due to a large number of degrees of freedom (DOFs) when complex microstructures exist in small-scale. Besides, for the microstructures with material heterogeneity, the three-dimensional mesh generation in small-scale could be also very complex as the conforming conditions on the element/material interfaces are required to be satisfied in transition of different mesh densities [39].

The Scaled Boundary Finite Element Method (SBFEM) originally presented by Song et al. [40] is a semi-analytical method for solving partial differential equations. The SBFEM is particularly suitable for stress singularities and unbounded domain and it has been widely applied into fracture problems [41,42], dynamics problems [43,44], uncertainty analysis [45,46], optimization problems [47,48], and parallel computations [49,50]. Recently, the quadtree/octree SBFEMs have been successfully developed by virtue of the ability of SBFEM to build polygonal elements [51–53]. In our previous works [54,55], a Multiscale Scaled Boundary Finite Element Method (MsSBFEM) was proposed by integrating the advantages of quadtree SBFEM and MsFEM, in which the numerical base functions in small-scale are calculated by SBFEM instead of conventional FEM. However, these works have only addressed two-dimensional problems, which cannot accurately reflect realistic conditions in many cases. In order to apply the MsSBFEM into more realistic problems with complex geometries, materials distribution and, boundary conditions, a three-dimensional MsSBFEM for heterogeneous viscoelastic materials is further developed by extending the previous basic two-dimensional method. In the proposed method, the octree SBFEM is used to deal with the three-dimensional calculation of numerical base functions in small-scale. The Octree-SBFEM has been applied in elastic [56], elastic-plastic [57], and sound field problems [58] including the development of cutting mesh technique [59] and polyhedral mesh [60–62] to handle the complex 3D boundary interfaces. However, to the authors' best knowledge, it is the first time to develop the three-dimensional MsSBFEM with octree mesh and apply it to the multiscale viscoelastic analysis. The proposed method has the following advantages:

(1) Using the numerical base function in the framework of the MsFEM, a bridge between the small-scale and large-scale is established. In this way, the DOFs of numerical computations for three-dimensional viscoelastic problems can be significantly reduced compared with those of DNS.

(2) The numerical base functions in the proposed model are constructed via image-based analysis based on the octree SBFEM instead of the conventional FEM with twofold advantages. First, the mesh can be directly generated from three-dimensional images on a small-scale without a requiring complex geometric modeling process in conventional FEM. Second, the stiffness matrix can be directly assembled through extraction from several basic patterns of three-dimensional elements to speed up the multiscale analysis [63,64].

(3) The Temporally Piecewise Adaptive Algorithm (TPAA) is used for discretization in the time domain for three-dimensional viscoelastic analysis [65–67]. In this way, a prescribed accuracy in the time domain can be maintained by adaptively changing the recursive order without the assumption that variables remain constant or change linearly at a discretized time interval.

The paper is organized as follows. Section 2 introduces the basic framework of the MsFEM and the construction process of three-dimensional numerical base functions based on the octree SBFEM is presented. Section 3 introduces the solution of viscoelastic problems based on the TPAA and its implementation in the framework of a three-dimensional multiscale SBFEM. Section 4 demonstrates the effectiveness of the proposed method via three numerical examples, and Section 5 presents some concluding remarks and discussions of future work.

2 Key Equation of the 3D Multiscale Octree SBFEM for Elastic Problems

2.1 The Introduction on the Framework of MsFEM

The equilibrium equation with boundary conditions for elastic problems can be expressed as [66]

$$\mathbf{H}\boldsymbol{\sigma} + \mathbf{F} = \mathbf{0} \quad \text{in } \Omega_E \quad (1)$$

$$\mathbf{u} = \mathbf{u}_\Gamma \quad \text{on } \Gamma_u \quad (2)$$

$$\mathbf{p} = \mathbf{p}_\Gamma \quad \text{on } \Gamma_\sigma \quad (3)$$

where $\boldsymbol{\sigma}$, $\boldsymbol{\varepsilon}$ and \mathbf{u} represent the vectors of stress, strain, and displacement, respectively, \mathbf{F} is the vector of body force intensities, Ω_E denotes the spatial domain to be solved, and \mathbf{H} is the differential operator [65]. \mathbf{u}_Γ and \mathbf{p}_Γ are the prescribed vectors along the displacement boundary Γ_u and stress boundary Γ_σ , respectively.

The strain vector is expressed as

$$\boldsymbol{\varepsilon} = \mathbf{H}^T \mathbf{u} \quad (4)$$

The constitutive equation is

$$\boldsymbol{\sigma} = \mathbf{D} \boldsymbol{\varepsilon} \quad (5)$$

where \mathbf{D} is the elastic matrix.

According to the basic idea of the MsFEM, a three-dimensional domain can have two levels of meshes as shown in Fig. 1, i.e., the coarse elements on a large-scale and the fine elements on a small-scale. The subscript E indicates that the variable belongs to the large-scale element. The FEM equation for solving the nodal displacement on coarse elements \mathbf{u}^E is written as

$$\mathbf{K}_G \mathbf{u}^E = \mathbf{P}_G \quad (6)$$

where \mathbf{K}_G and \mathbf{P}_G are the global stiffness matrix and load vector of the large-scale assembled of all coarse elements. These can be written as

$$\mathbf{K}_G = A_{j=1}^R \mathbf{K}_j^E \quad (7)$$

$$\mathbf{P}_G = A_{j=1}^R \mathbf{P}_j^E \quad (8)$$

where $A_{j=1}^R$ is the assembly operator, \mathbf{K}_j^E and \mathbf{P}_j^E are the stiffness matrix and load vector of the j -th coarse element, respectively, and R is the total number of coarse elements.

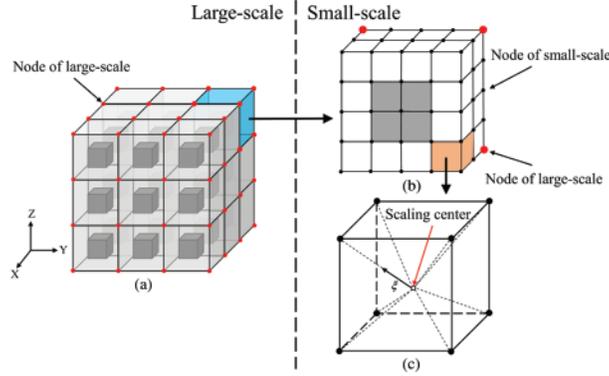


Figure 1: Meshes in 3D MsSBFEM (a) coarse grids on overall structure (b) fine grids inside the coarse element (c) octree SBFEMs

Instead of using regular numerical integration calculations, the \mathbf{K}_j^E and \mathbf{P}_j^E are obtained by a mapping relationship from the corresponding matrix or vector of fine elements inside this coarse element as [36]

$$\mathbf{K}_j^E = \sum_{i=1}^r \mathbf{G}_j^T \mathbf{K}_i^e \mathbf{G}_j \quad (9)$$

$$\mathbf{P}_j^E = \sum_{i=1}^r \mathbf{G}_j^T \mathbf{P}_i^e \quad (10)$$

where \mathbf{G}_j is the transfer matrix, r is the total number of fine elements, and the subscript e indicates that the variable belongs to the small-scale element. In this way, this elastic problem is solved on a large-scale, so the number of DOFs is significantly reduced compared to DNS in small-scale. In the following section, we introduce how to establish \mathbf{G}_j .

2.2 Octree SBFEM Based Numerical Base Functions

The three-dimensional displacement fields within the j -th coarse element can be expressed as [36]

$$\mathbf{u}_i^e = \mathbf{G}_i \mathbf{u}_j^E \quad (11)$$

where \mathbf{G}_i is the transfer matrix of the i -th fine element, and \mathbf{u}_i^e and \mathbf{u}_j^E are the nodal displacement vectors for the j -th coarse element and the i -th fine element inside this coarse element, respectively.

\mathbf{G}_i in Eq. (11) is composed of numerical base functions \mathbf{N}_n as

$$\mathbf{G}_i = \begin{bmatrix} \mathbf{N}_{1x}^i(1) & \mathbf{N}_{1y}^i(1) & \mathbf{N}_{1z}^i(1) & \dots & \mathbf{N}_{1x}^i(1) & \mathbf{N}_{1y}^i(1) & \mathbf{N}_{1z}^i(1) \\ \mathbf{N}_{1x}^i(2) & \mathbf{N}_{1y}^i(2) & \mathbf{N}_{1z}^i(2) & \dots & \mathbf{N}_{1x}^i(2) & \mathbf{N}_{1y}^i(2) & \mathbf{N}_{1z}^i(2) \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \mathbf{N}_{1x}^i(n) & \mathbf{N}_{1y}^i(n) & \mathbf{N}_{1z}^i(n) & \dots & \mathbf{N}_{1x}^i(n) & \mathbf{N}_{1y}^i(n) & \mathbf{N}_{1z}^i(n) \end{bmatrix} \quad (12)$$

where

$$\mathbf{N}_{1x}^i(n) = [N_{1xx}^i(n) \quad N_{1yx}^i(n) \quad N_{1zx}^i(n)]^T \quad (13)$$

$$\mathbf{N}_{1y}^i(n) = [N_{1xy}^i(n) \quad N_{1yy}^i(n) \quad N_{1zy}^i(n)]^T \quad (14)$$

$$\mathbf{N}_{1z}^i(n) = [N_{1xz}^i(n) \quad N_{1yz}^i(n) \quad N_{1zz}^i(n)]^T \quad (15)$$

where n and I are the numbers of nodes in the fine element and coarse element, respectively. Taking $N_{Iyz}^i(n)$ as an example, Eqs. (13)–(15) indicate that the displacement of the I -th node of the j -th coarse element in the z direction leads to the displacement of the n -th node in the y direction of the i -th fine element.

\mathbf{N}_{Ix}^i , \mathbf{N}_{Iy}^i and \mathbf{N}_{Iz}^i are solved in the same governing equations but with different boundary conditions for the fine elements shown in Fig. 1b. Taking \mathbf{N}_{Iz}^i as an example, the governing equations and boundary conditions can be written as

$$\mathbf{H}(\mathbf{D}\mathbf{H}^T\mathbf{N}_{Iz}^i) = \mathbf{0} \quad \text{in } \Omega_E \quad (16)$$

$$\mathbf{N}_{Iz}^i = \mathbb{N}_{Iz}^i \quad \text{on } \Gamma_E \quad (17)$$

$$I = 1, 2, \dots, R \quad (18)$$

where R is the nodal number of a coarse element and \mathbf{D} is the elastic matrix.

In this paper, the linear boundary conditions are considered [34] as shown in Fig. 2, which means that we calculate \mathbf{N}_{Iz}^i with $I = 1$, the boundary conditions are set as $\mathbb{N}_{1z}^i = 1$ and $\mathbb{N}_{2z}^i = \mathbb{N}_{3z}^i = \dots = \mathbb{N}_{8z}^i = 0$, and the z -direction displacement \mathbb{N}_{Iz}^i is assumed to be a linear change on the Surface 1, Surface 3, and Surface 5, as well as the lines 12, 14 and 15 on these surfaces. When constructing the numerical basis function in the z -direction, the displacements in the x -direction, y -direction and on all boundaries are fully constrained. The \mathbb{N}_{Ix}^i and \mathbb{N}_{Iy}^i in the x -direction and y -direction can be constructed similarly.

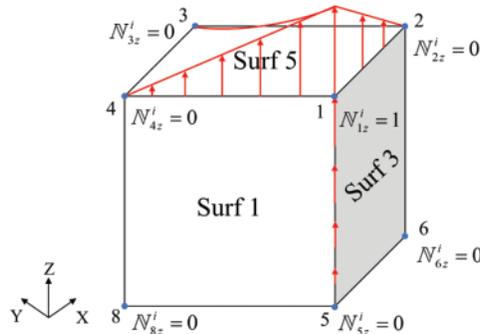


Figure 2: 3D linear boundary conditions for numerical base function of displacement field

The governing equations for the numerical base functions in Eqs. (16) and (17) are solved using the SBFEM with an octree mesh on a fine grid in the small-scale [55]. The generation of the octree mesh and the computational scheme for the three-dimensional SBFEM are introduced in the following sections and the Appendix.

2.3 Image-Based Octree Gridding Technique

In order to solve the numerical base functions introduced in Section 3.1, the octree mesh is used in the SBFEM based calculation. The octree mesh can be directly generated from 3D images in small-scale. A detailed introduction of this approach is provided in reference [63] and an example is shown in Fig. 3. Firstly, a 3D image, usually obtained from imaging techniques such as X-ray computed tomography (XCT) or produced digitally via random generation and packing of aggregates, is read to produce a 3D image matrix containing information on color intensity. Secondly, the octree decomposition is implemented to recursively divide a 3D image matrix into eight equal-sized cells at a time until all the cells satisfy the criterion of homogeneity or the minimum edge length is reached.

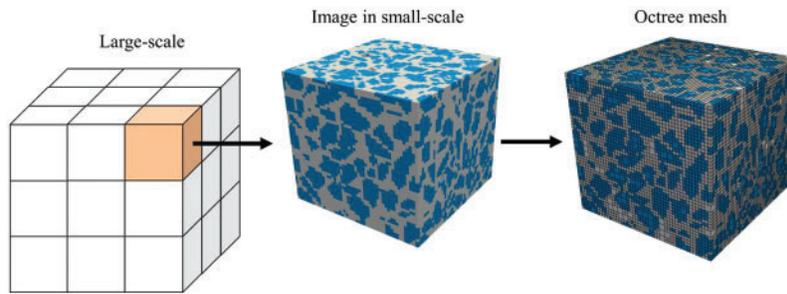


Figure 3: Image based 3D multiscale mesh generation

When a 2:1 rule is used in the process of octree gridding, a balanced decomposition can be obtained to limit the number of nodal patterns of one octree mesh. This has significant advantages, as similar nodal patterns can be precomputed because the element matrices for two octree cells with equivalent nodal configurations are proportional. In such a balanced decomposition, each element mode only depends on whether there is a hanging node in the center of each edge of the cube, and if there are four hanging nodes on any surface of the cube (as shown in Fig. 4), the node in the center of the surface will be automatically added, and it is proved that the number of total unique nodal patterns for an octree cell is $2^{12} = 4096$ [63].

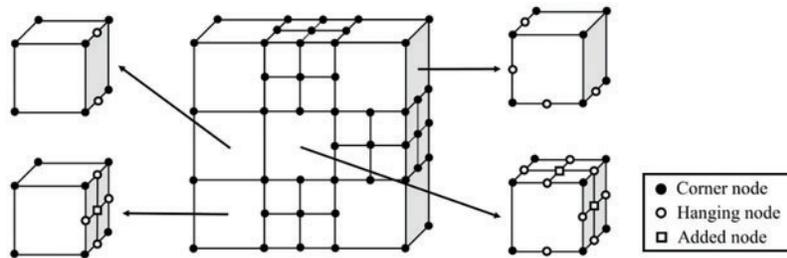


Figure 4: Several typical three-dimensional octree scaled boundary elements

3 Recursive 3D Multiscale Octree SBFEM for Viscoelastic Problems

3.1 Recursive Governing and Constitutive Equations

To handle time-dependent viscoelastic problems, the discretization in the time domain is achieved using the Temporally Piecewise Adaptive Algorithm (TPAA) [66], in which all variables are expanded in each time as follows:

$$f(t) = \sum_{m=0} f^m s^m \tag{19}$$

where the general variable $f(t)$ is used to represent $\sigma(t)$, $\epsilon(t)$, $\mathbf{F}(t)$, $\mathbf{u}(t)$, $\mathbf{u}_\Gamma(t)$, $\mathbf{p}(t)$ and $\mathbf{p}_\Gamma(t)$ in Eqs. (1)–(5), and

$$s = \frac{t - t_{k-1}}{T_k} \tag{20}$$

where t and s denote the time and dimensionless time parameters, respectively, and t_{k-1} and T_k are the initial point and size of the k_{th} time interval, respectively.

Substituting the expanded variables into Eqs. (1)–(3), we have the following governing equations for the expansion coefficients

$$\mathbf{H}\boldsymbol{\sigma}^m + \mathbf{F}^m = \mathbf{0} \quad \text{in } \Omega \quad (21)$$

$$\boldsymbol{\varepsilon}^m = \mathbf{H}^T \mathbf{u}^m \quad (22)$$

$$\mathbf{u}^m = \mathbf{u}_\Gamma^m \quad \text{on } \Gamma_u \quad (23)$$

$$\mathbf{p}^m = \mathbf{p}_\Gamma^m \quad \text{on } \Gamma_\sigma \quad (24)$$

Now, we derive the recursive constitutive relationship for $\boldsymbol{\sigma}(t)$ and $\boldsymbol{\varepsilon}(t)$. In this paper, we consider the three-parameter solid model as a typical viscoelastic constitutive model, as shown in Fig. 5. The differential form of the constitutive equation is written as [67]

$$\begin{cases} \boldsymbol{\sigma}(t) + q_1 \frac{\partial \boldsymbol{\sigma}(t)}{\partial t} = \mathbf{D} \left(q_2 \boldsymbol{\varepsilon}(t) + q_1 \frac{\partial \boldsymbol{\varepsilon}(t)}{\partial t} \right) & (t > 0) \\ \boldsymbol{\sigma}(t) = \mathbf{D} \boldsymbol{\varepsilon}(t) & (t = 0) \end{cases} \quad (25)$$

where

$$q_1 = \frac{\eta_1}{E_1 + E_2}; \quad q_2 = \frac{E_1}{E_1 + E_2} \quad (26)$$

Considering the relationship between the differentiations with respect to t and s in Eq. (20)

$$\frac{d(\cdot)}{dt} = \frac{1}{T_k} \frac{d(\cdot)}{ds} \quad (27)$$

Therefore, the time derivatives of the stress and strain can be expressed as

$$\frac{\partial \boldsymbol{\sigma}(t)}{\partial t} = \sum_{m=0} \frac{m+1}{T_k} \boldsymbol{\sigma}^{m+1} s^m \quad (28)$$

$$\frac{\partial \boldsymbol{\varepsilon}(t)}{\partial t} = \sum_{m=0} \frac{m+1}{T_k} \boldsymbol{\varepsilon}^{m+1} s^m \quad (29)$$

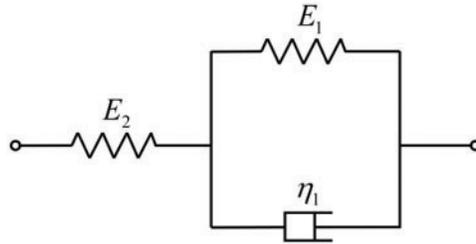


Figure 5: Three-parameter solid model

In each time interval, substituting Eqs. (28) and (29), into Eq. (25) and equating the powers of the two sides of the equation yields

$$\boldsymbol{\sigma}^m = \mathbf{D} \boldsymbol{\varepsilon}^m + \mathbf{C}^m \quad (30)$$

where

$$\mathbf{C}^m = \frac{T^k}{m} \left(\frac{E_1}{\eta_1} \mathbf{D}\boldsymbol{\varepsilon}^{m-1} - \frac{1}{q_1} \boldsymbol{\sigma}^{m-1} \right) \quad (m > 0) \quad (31)$$

At the first time interval,

$$\begin{cases} \boldsymbol{\sigma}^0 = \mathbf{D}\boldsymbol{\varepsilon}^0 \\ \mathbf{C}^0 = \mathbf{0} \end{cases} \quad (k = 1) \quad (32)$$

At the initial point of the $(k+1)$ th time interval

$$\begin{cases} \boldsymbol{\varepsilon}_{k+1}^0 = \sum_{m=0}^k \boldsymbol{\varepsilon}_k^m \\ \boldsymbol{\sigma}_{k+1}^0 = \sum_{m=0}^k \boldsymbol{\sigma}_k^m \quad (k > 1) \\ \mathbf{C}_{k+1}^0 = \sum_{m=0}^k \mathbf{C}_k^m \end{cases} \quad (33)$$

In this way, the time-dependent problems in Eqs. (21)–(25) are transferred into a series of recursive elastic problems with initial stress and strain, which will be recursively solved by the MsSBFEM introduced in the following sections.

3.2 Recursive Multiscale Octree SBFEM for Viscoelastic Problems

Using the derived MsSBFEM equations for elastic problems in Section 2, the recursive governing equations and constitutive equation for three-dimensional viscoelastic problems are naturally solved for coarse elements at large-scale [54].

$$\mathbf{K}_G (\mathbf{u}^m)^E = \mathbf{P}_G^m \quad (34)$$

where the right-term \mathbf{P}_G^m contains the term related to \mathbf{C}^m in Eq. (31) written as

$$\mathbf{P}_G^m = A_{j=1}^R \left[(\mathbf{P}\mathbf{S}_j^m)^E - (\mathbf{P}\mathbf{C}_j^m)^E \right] \quad (35)$$

where

$$(\mathbf{P}\mathbf{S}_j^m)^E = \sum_{i=1}^r \mathbf{G}_i^T (\mathbf{P}\mathbf{S}_j^m)^e \quad (36)$$

$$(\mathbf{P}\mathbf{C}_j^m)^E = \sum_{i=1}^r (\mathbf{G}_i^e)^T (\mathbf{P}\mathbf{C}_j^m)^e \quad (37)$$

where \mathbf{G}_i is the transfer matrix provided by Eq. (12) and

$$(\mathbf{P}\mathbf{S}_i^m)^e = \int_{\Gamma} [\mathbf{N}_i(s)]^e [\mathbf{F}_i^m(s)]^e ds \quad (38)$$

$$(\mathbf{P}\mathbf{C}_i^m)^e = \frac{T}{m} \left(-\frac{E_{2i}}{\eta_{1i}} \mathbf{K}_i (\mathbf{u}_i^{m-1})^e - \frac{E_{1i} + E_{2i}}{\eta_{1i}} (\mathbf{P}\mathbf{C}_i^{m-1})^e \right) \quad (39)$$

where $(\mathbf{u}_i^{m-1})^e$ is the $(m-1)$ -th order expanding coefficient of the nodal displacement of the i -th fine element.

Note that $(\mathbf{u}_i^{m-1})^e$ and $(\mathbf{P}\mathbf{C}_i^{m-1})^e$ in Eq. (39) are obtained via downscale computation from the relationship between two-scale variables using a numerical base function

$$(\mathbf{u}_i^{m-1})^e = \mathbf{G}_i (\mathbf{u}_i^{m-1})^E \quad (40)$$

$$(\mathbf{PC}_i^{m-1})^e = \mathbf{G}_i (\mathbf{PC}_i^{m-1})^E \quad (41)$$

The $(\mathbf{PC}_i^0)^e$ at the initial points of the k -th time interval is calculated as

$$(\mathbf{PC}_i^0)^e = \mathbf{0} \quad \text{if } k = 1 \quad (42)$$

$$(\mathbf{PC}_i^0)^e = \sum_m (\mathbf{PC}_i^m)^e \quad \text{if } k > 1 \quad (43)$$

At each time interval, an adaptive computation is conducted by setting the number of expansion terms m following the criteria [67]

$$\frac{\|(\mathbf{u}^m)^E s^m|_{s=1}\|_2}{\left\| \sum_{h=0}^{m-1} (\mathbf{u}^h)^E s^h \right\|_2} \leq \beta \quad (44)$$

where β is a prescribed error tolerance, and $\|\cdot\|_2$ represents the L^2 -norm.

4 Numerical Examples

Three numerical examples are provided in this section. The first example demonstrates the effectiveness of the proposed algorithm in a cubic domain with periodic inclusions. The performance of the proposed method for non-periodic inclusions is demonstrated in the second example. The third example applies the proposed method to a concrete beam with CT images of microstructures. To evaluate the accuracy of the proposed method, the relative error e_r is defined.

4.1 A Cubic Domain with Periodic Inclusions

Consider a cube domain of heterogeneous viscoelasticity under tension as shown in Fig. 6. There are 27 small cubic inclusions contained in the cube, and their material parameters are provided in Table 1. In order to investigate the proposed method for different levels of heterogeneity, two cases of volumetric ratios of inclusions κ are set with various sizes a as shown in Fig. 6c, i.e., Case I: $a = 42$, $\kappa = 28.3\%$; Case II: $a = 50$, $\kappa = 47.6\%$.

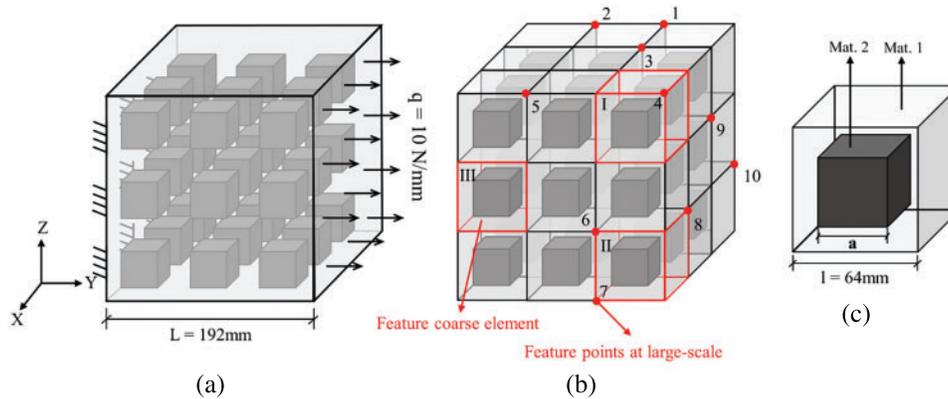


Figure 6: A cube with regular cube inclusions (a) heterogeneous material distribution and boundary conditions (b) coarse mesh (c) the material distribution in one coarse mesh

Table 1: Material parameters

Mat1			Mat2			ν
E_1 (MPa)	E_2 (MPa)	η (MPa * s)	E_1 (MPa)	E_2 (MPa)	η (MPa * s)	
2000	2000	2000	1000	1000	1000	0.3

The material distributions in large-scale and small-scale are shown in Figs. 6b and 7, respectively. In total, 27 uniform coarse elements are used, and three types of coarse elements with different nodal distributions (coarse nodes marked by red color) are used.

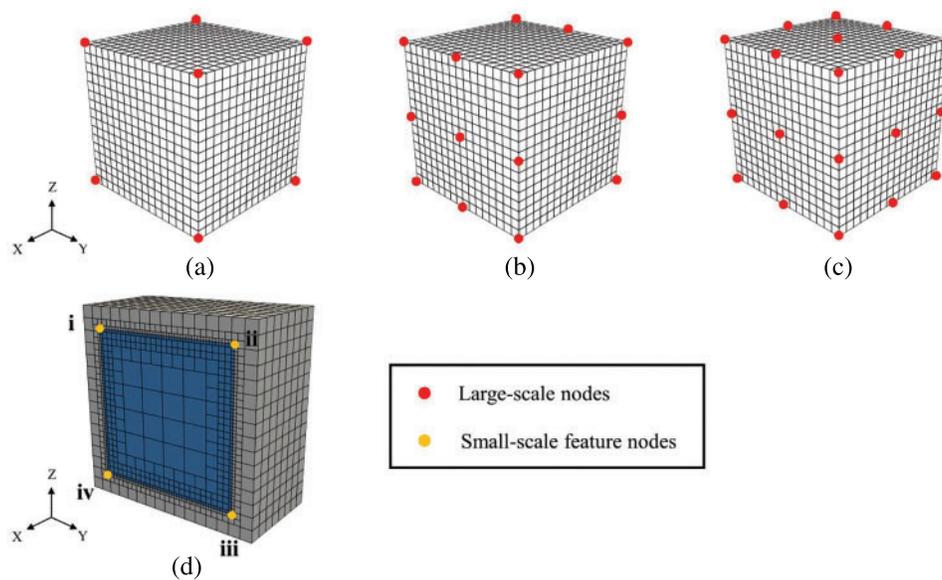


Figure 7: Coarse SBFEs with different number of nodes (a) model A (8 nodes) (b) model B (18 nodes) (c) model C (26 nodes) (d) section view of octree mesh

Firstly, the computational accuracy is verified by comparing the results of the proposed method with the reference solution from a converged Abaqus based DNS solution with its mesh in Fig. 8. Table 2 shows the results of displacement at large-scale feature point 1 (see Fig. 6b) and the results for displacements at small-scale feature point i of coarse element II (see Figs. 6b and 7d) are also provided in Table 3. The maximum relative errors of the proposed method are 0.69% and 0.30% in large-scale and small-scale, respectively, and these relative errors change very slightly when the time step is increased from 0.001 to 0.1 s.

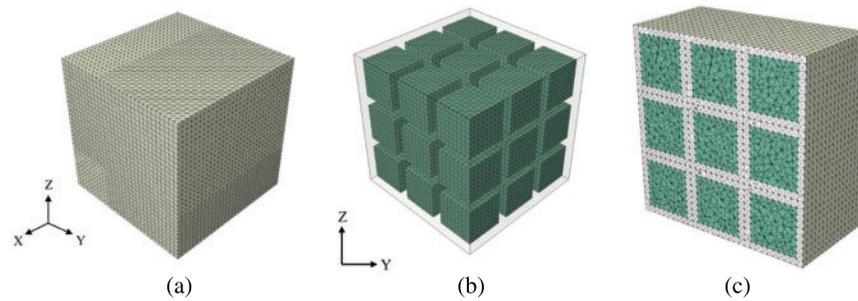


Figure 8: FE mesh of reference solution based on Abaqus (a) matrix (b) inclusions (c) section view

Table 2: e_r for displacement u_y at large-scale feature point 1 with different time steps (error tolerance $\beta = 10^{-6}$)

Time (s)	$T = 0.1$		$T = 0.01$		$T = 0.001$		Reference (mm)
	MsSBFEM (mm)	e_r (%)	MsSBFEM (mm)	e_r (%)	MsSBFEM (mm)	e_r (%)	
0.1	0.8109	0.69	0.8109	0.69	0.8110	0.68	0.8166
0.2	0.8747	0.69	0.8747	0.69	0.8748	0.68	0.8808
0.3	0.9324	0.69	0.9324	0.69	0.9325	0.68	0.9389
0.4	0.9846	0.69	0.9846	0.69	0.9847	0.68	0.9914
0.5	1.0318	0.69	1.0318	0.69	1.0319	0.68	1.0390
0.6	1.0746	0.69	1.0746	0.69	1.0746	0.68	1.0820
0.7	1.1132	0.69	1.1132	0.69	1.1133	0.68	1.1210
0.8	1.1482	0.69	1.1482	0.69	1.1483	0.68	1.1562
0.9	1.1799	0.69	1.1799	0.69	1.1799	0.69	1.1881
1.0	1.2085	0.69	1.2085	0.69	1.2086	0.69	1.2169

Table 3: e_r for displacement u_y at small-scale feature point i with different time steps (error tolerance $\beta = 10^{-6}$)

Time (s)	$T = 0.1$		$T = 0.01$		$T = 0.001$		Reference (mm)
	MsSBFEM (mm)	e_r (%)	MsSBFEM (mm)	e_r (%)	MsSBFEM (mm)	e_r (%)	
0.1	0.8593	0.30	0.8593	0.30	0.8594	0.30	0.8619
0.2	0.9269	0.30	0.9269	0.30	0.9269	0.29	0.9297
0.3	0.9880	0.30	0.9880	0.30	0.9881	0.29	0.9910
0.4	1.0433	0.30	1.0433	0.30	1.0434	0.29	1.0465
0.5	1.0934	0.30	1.0934	0.30	1.0935	0.29	1.0967
0.6	1.1387	0.30	1.1387	0.30	1.1388	0.29	1.1421
0.7	1.1797	0.30	1.1797	0.30	1.1797	0.29	1.1832
0.8	1.2167	0.30	1.2167	0.30	1.2168	0.29	1.2204
0.9	1.2503	0.30	1.2503	0.30	1.2504	0.30	1.2541
1.0	1.2807	0.30	1.2807	0.30	1.2807	0.30	1.2845

Fig. 9 shows the variation in displacement with time for large-scale feature point 2 and small-scale feature point i in coarse element II. The reference solution is provided by the Abaqus with both automatic and fixed time steps, in which the implicit integration method in the time domain is used. The comparisons of different time steps $T = 0.1, 0.01, \text{ and } 0.001 \text{ s}$ are shown in Fig. 9. When the time step of Abaqus is relatively large as $T = 0.1 \text{ s}$. Obviously, there are errors in the initial stage, but the proposed method with the same time step still achieves good results which shows that the TPA algorithm is capable of adaptively adjusting the expansion order to ensure the accuracy in the time domain. Fig. 10 shows the comparison of the expansion order under two different prescribed convergence accuracy parameters, which indicate that the TPA algorithm can also adaptively adjust the expansion order to balance the calculation accuracy and efficiency when the given convergence accuracy changes.

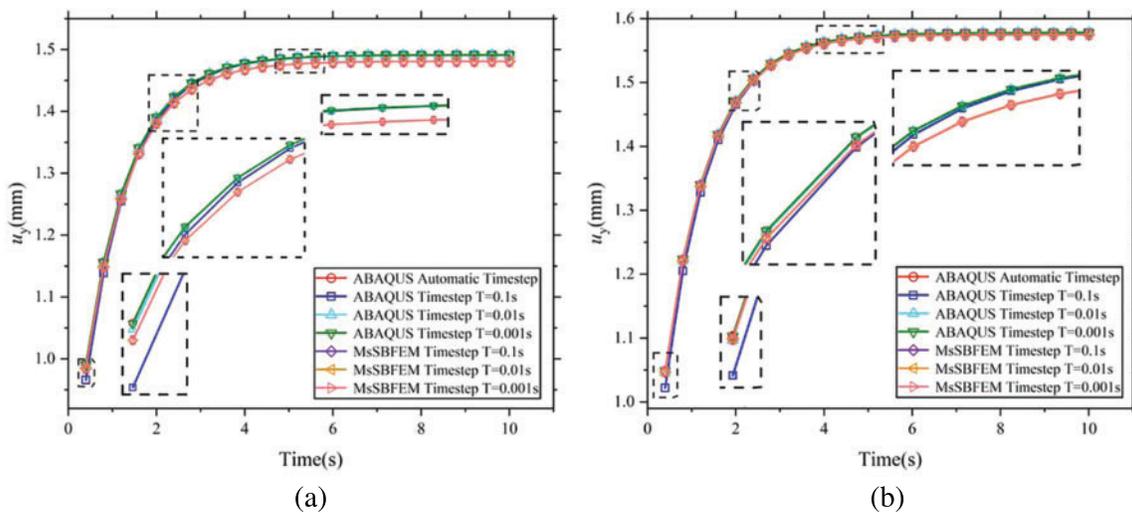


Figure 9: Displacement curves of feature points with different time steps (a) The large-scale feature point 1 (b) The small-scale feature point i in the coarse element II

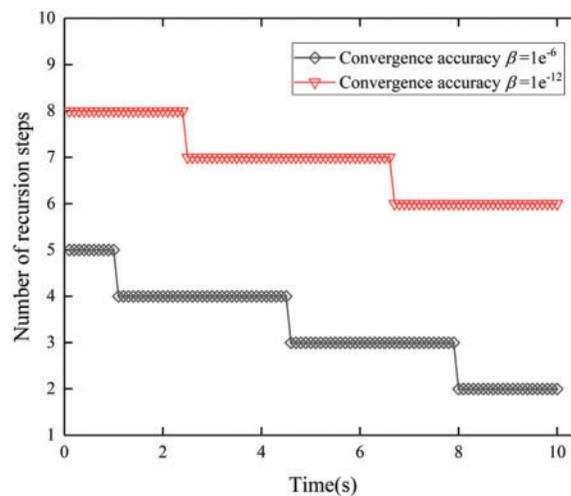


Figure 10: The variation of recursive orders with time

Secondly, the two cases of material properties in case I and case II in Table 1 are tested by using the coarse element Model A with 8 nodes, as shown in Fig. 7. Table 4 shows the results of u_y on large-scale feature nodes shown in Fig. 6b when $t = 10$ s and its relative error e_r . The calculation results show that for two cases of material properties in case I and case II, although e_r is less than 3% on nodes 4 and 10, but e_r on some other nodes are still quite large nearly 10%. We will discuss how to improve the accuracy by adjusting the types of coarse elements in the following section.

Table 4: Comparisons of u_y and e_r in large-scale with different material proportions ($t = 10$ s)

The feature points at large-scale	Case I ($\kappa = 28.3\%$)			Case II ($\kappa = 47.6\%$)		
	Reference (mm)	MsSBFEM (mm)	e_r (%)	Reference (mm)	MsSBFEM (mm)	e_r (%)
1	1.491	1.323	11.28	1.689	1.500	11.17
2	0.794	0.699	11.96	0.899	0.792	11.92
3	1.470	1.360	7.46	1.665	1.542	7.40
4	2.147	2.123	1.11	2.396	2.405	0.40
5	0.794	0.699	11.96	0.899	0.792	11.92
6	1.470	1.303	11.36	1.665	1.477	11.28
7	1.491	1.323	11.28	1.689	1.477	12.50
8	2.183	2.028	7.08	2.456	2.299	6.36
9	2.183	2.028	7.08	2.456	2.299	6.40
10	2.158	2.128	1.38	2.460	2.410	2.05

Thirdly, by comparing the computational results for the coarse element Model A (8 nodes), Model B (18 nodes), and Model C (26 nodes) as shown in Fig. 7, Tables 5 and 6 show the result of displacement u_y calculated by different models for large feature points and small-scale feature points of coarse elements I–III (see Figs. 6b and 7d) when $t = 10$ s. These results demonstrate that when the number of coarse nodes increases from 8 to 26, the computational accuracy significantly improves as the maximum error in large-scale decreases from 12.50% to 1.43%. Accordingly, the maximum error in small-scale also decreases from 11.73% to 1.54%. Notably, we increase only the number of coarse nodes here but the octree mesh used in the small-scale is not altered.

Table 5: Comparisons of u_y and e_r in large-scale with different coarse element models ($t = 10$ s)

The feature points at large-scale	Model A (8 nodes)		Model B (18 nodes)		Model C (26 nodes)		Reference (mm)
	MsSBFEM (mm)	e_r (%)	MsSBFEM (mm)	e_r (%)	MsSBFEM (mm)	e_r (%)	
Case I ($\kappa = 28.3\%$)							
1	1.323	11.28	1.480	0.75	1.481	0.69	1.491
2	0.699	11.96	0.779	2.00	0.787	0.96	0.794
3	1.360	7.46	1.457	0.85	1.460	0.66	1.470
4	2.123	1.11	2.187	1.86	2.173	1.24	2.147
5	0.699	11.96	0.779	2.00	0.787	0.96	0.794

(Continued)

Table 5 (continued)

The feature points at large-scale	Model A (8 nodes)		Model B (18 nodes)		Model C (26 nodes)		Reference (mm)
	MsSBFEM (mm)	e_r (%)	MsSBFEM (mm)	e_r (%)	MsSBFEM (mm)	e_r (%)	
6	1.303	11.36	1.460	0.65	1.460	0.66	1.470
7	1.323	11.28	1.480	0.75	1.481	0.69	1.491
8	2.028	7.08	2.190	0.32	2.183	0.01	2.183
9	2.028	7.08	2.190	0.32	2.183	0.01	2.183
10	2.128	1.38	2.196	1.79	2.177	0.90	2.158
Case II ($\kappa = 47.6\%$)							
1	1.500	11.17	1.676	0.72	1.677	0.67	1.689
2	0.792	11.92	0.882	1.96	0.891	0.93	0.899
3	1.542	7.40	1.651	0.87	1.654	0.66	1.665
4	2.405	0.40	2.440	1.83	2.427	1.30	2.396
5	0.792	11.92	0.882	1.96	0.891	0.92	0.899
6	1.477	11.28	1.655	0.63	1.654	0.66	1.665
7	1.477	12.50	1.655	1.99	1.664	1.43	1.689
8	2.299	6.36	2.476	0.84	2.462	0.26	2.456
9	2.299	6.40	2.476	0.80	2.462	0.22	2.456
10	2.410	2.05	2.483	0.93	2.453	0.28	2.460

Table 6: Comparisons of u_r and e_r in small-scale with different coarse element models ($t = 10$ s)

The feature points at small-scale	Model A (8 nodes)		Model B (18 nodes)		Model C (26 nodes)		Reference (mm)
	MsSBFEM (mm)	e_r (%)	MsSBFEM (mm)	e_r (%)	MsSBFEM (mm)	e_r (%)	
Case I ($\kappa = 28.3\%$)							
Element I (i)	1.4388	9.19	1.5728	0.73	1.5750	0.59	1.5843
Element I (ii)	1.9534	8.89	2.1024	1.94	2.1227	0.99	2.1440
Element I (iii)	1.9548	8.52	2.1018	1.64	2.1207	0.76	2.1369
Element I (iv)	1.4251	9.67	1.5667	0.69	1.5693	0.53	1.5777
Element II (i)	1.4251	9.68	1.5667	0.70	1.5693	0.54	1.5779
Element II (ii)	1.9548	8.52	2.1018	1.65	2.1207	0.76	2.1370
Element II (iii)	1.9534	8.90	2.1024	1.95	2.1227	1.00	2.1443
Element II (iv)	1.4388	9.17	1.5728	0.71	1.5750	0.57	1.5841
Element III (i)	0.0788	2.18	0.0793	2.88	0.0771	0.06	0.0771
Element III (ii)	0.5092	11.68	0.5704	1.07	0.5681	1.48	0.5766
Element III (iii)	0.5092	11.69	0.5704	1.08	0.5682	1.47	0.5766
Element III (iv)	0.0788	2.54	0.0793	3.25	0.0771	0.42	0.0768

(Continued)

Table 6 (continued)

The feature points at small-scale	Model A (8 nodes)		Model B (18 nodes)		Model C (26 nodes)		Reference (mm)
	MsSBFEM (mm)	e_r (%)	MsSBFEM (mm)	e_r (%)	MsSBFEM (mm)	e_r (%)	
Case II ($\kappa = 47.6\%$)							
Element I (i)	1.5914	9.09	1.7372	0.76	1.7402	0.59	1.7505
Element I (ii)	2.2645	7.86	2.4134	1.80	2.4418	0.65	2.4578
Element I (iii)	2.2679	7.65	2.4183	1.52	2.4449	0.44	2.4557
Element I (iv)	1.5725	9.61	1.7266	0.75	1.7293	0.59	1.7396
Element II (i)	1.5725	9.61	1.7266	0.75	1.7293	0.60	1.7397
Element II (ii)	2.2679	7.65	2.4183	1.52	2.4449	0.44	2.4557
Element II (iii)	2.2645	7.87	2.4134	1.81	2.4418	0.66	2.4579
Element II (iv)	1.5914	9.08	1.7372	0.76	1.7402	0.58	1.7504
Element III (i)	0.0563	8.40	0.0567	9.23	0.0524	1.03	0.0519
Element III (ii)	0.6203	11.73	0.6959	0.96	0.6918	1.54	0.7027
Element III (iii)	0.6203	11.71	0.6959	0.95	0.6918	1.53	0.7026

4.2 A Cube Domain with Non-Periodic Spherical Inclusions

Consider a heterogeneous cube with the boundary conditions and geometric parameters shown in Fig. 11. The cube contains 27 spherical inclusions of different sizes and the material parameters are shown in Table 7. Accordingly, 27 coarse elements are used to discretize the cube on large-scale as shown in Fig. 11b. The reference solution is provided using the convergence results of Abaqus with the mesh shown in Fig. 12. The coarse grid in large-scale and the fine grid in small-scale are shown in Figs. 11b and 13, respectively.

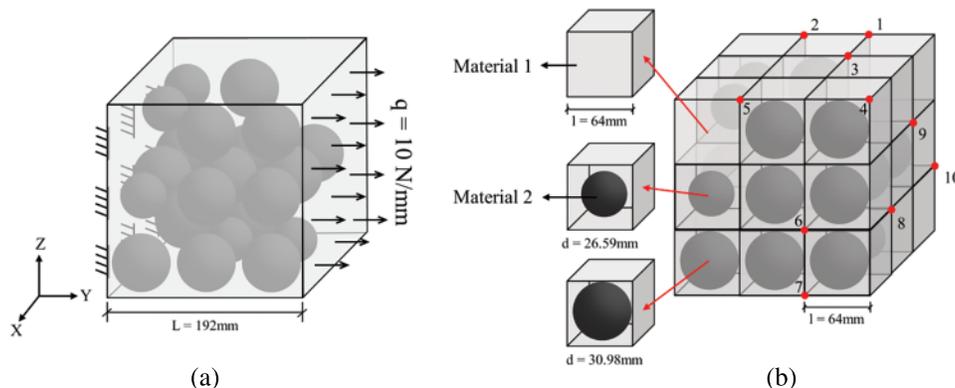


Figure 11: A cube with non-periodic spherical inclusions (a) matrix, inclusions and boundary conditions (b) coarse mesh and the material distribution

Table 7: Material parameters for the cube with aperiodic spherical inclusions

	Mat1			Mat2			ν
	E_1 (MPa)	E_2 (MPa)	η (MPa * s)	E_1 (MPa)	E_2 (MPa)	η (MPa * s)	
Case I	2000	2000	2000	1000	1000	1000	0.3
Case II	2000	2000	2000	500	500	500	

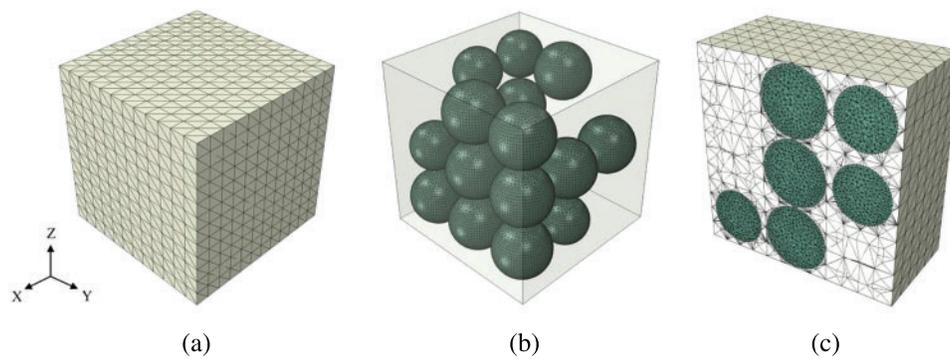


Figure 12: FE mesh in reference solution based on Abaqus (a) outer surface (b) inclusions (c) profile

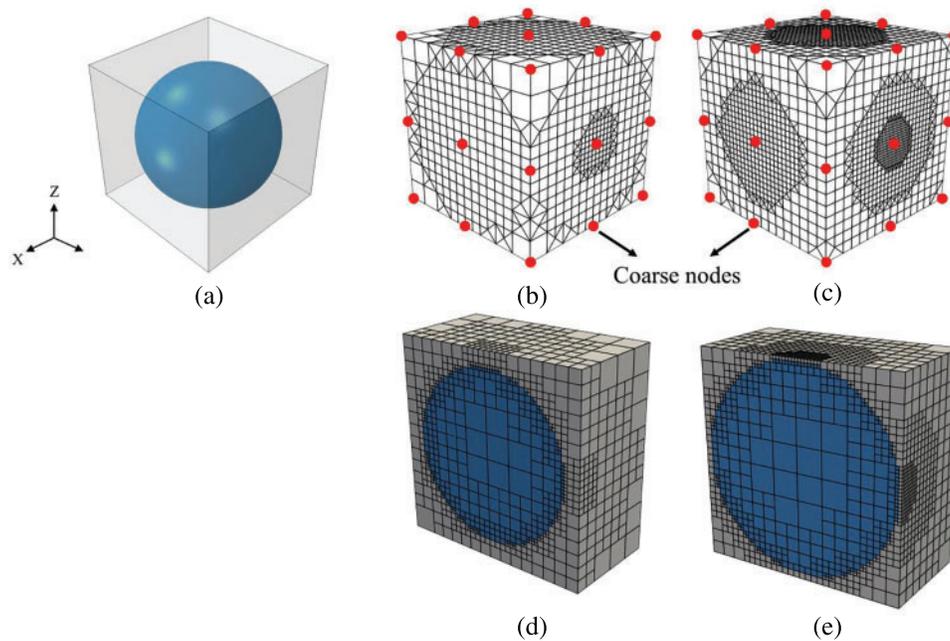


Figure 13: SBFEs with different size of spherical inclusions (a) geometric model (b) inclusion diameter $d = 26.59$ mm (c) inclusion diameter $d = 30.98$ mm (d, e) section view of octree mesh

In order to test the influence of quadtree mesh near the interface regions on calculation accuracy, we use the two sizes of quadtree grids shown in Fig. 14 to calculate four small-scale feature points in

the coarse elements. Table 8 shows that the maximum relative error is decreased from 1.61% to 1.13% as the minimal size of the elements in small-scale is refined from 2 to 1 mm.

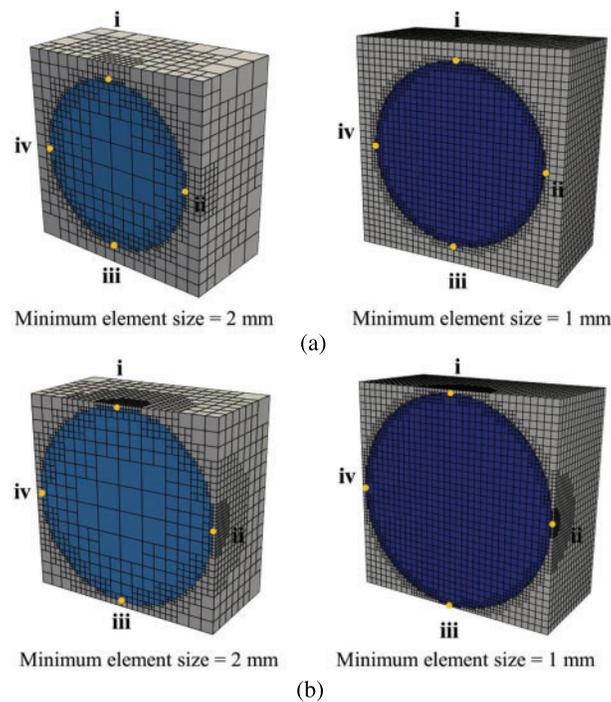


Figure 14: Two different sizes of small-scale grids (a) course element I (b) course element II

Table 8: Comparisons of u_x and e_r at small-scale with different mesh densities (Case I, $t = 10$ s)

The feature points at small-scale	Reference (mm)	Original mesh		Refined mesh	
		MsSBFEM (mm)	e_r (%)	MsSBFEM (mm)	e_r (%)
Element I (i)	1.993	1.974	0.93	1.982	0.57
Element I (ii)	2.598	2.562	1.38	2.576	0.83
Element I (iii)	2.033	2.024	0.45	2.030	0.14
Element I (iv)	1.591	1.587	0.24	1.591	0.01
Element II (i)	0.287	0.283	1.47	0.284	1.01
Element II (ii)	0.622	0.618	0.75	0.619	0.48
Element II (iii)	0.319	0.313	1.61	0.315	1.13
Element II (iv)	2.033	2.024	0.45	2.030	0.14

The accuracy of the proposed model in large scale is verified by using the two cases of material parameters (see Table 7). Table 9 shows the displacements at the feature points (shown in Fig. 11b) and the relative error e_r when the $t = 10$ s. When the difference in material parameters between the two materials increases, the maximum error of large-scale feature points only increases from 1.36% to 2.03%, showing that the proposed algorithm maintains good accuracy for varying material properties.

Table 9: Comparisons of u_r and e_r at large-scale with different material parameters ($t = 10$ s)

The feature points at large-scale	Case I			Case II		
	Reference (mm)	MsSBFEM (mm)	e_r (%)	Reference (mm)	MsSBFEM (mm)	e_r (%)
1	1.490	1.486	0.29	1.738	1.727	0.67
2	0.784	0.775	1.14	0.897	0.882	1.68
3	1.442	1.429	0.92	1.655	1.626	1.74
4	2.201	2.215	0.61	2.514	2.549	1.42
5	0.730	0.722	1.17	0.783	0.771	1.47
6	1.575	1.554	1.36	1.926	1.891	1.84
7	1.642	1.625	0.99	2.041	2.000	2.03
8	2.320	2.311	0.37	2.778	2.748	1.08
9	2.200	2.186	0.63	2.537	2.501	1.42
10	2.229	2.241	0.55	2.603	2.622	0.72

To verify the performance of the method in small-scale, we compare the displacement-time curves of several specific points in different coarse elements with two material parameters in Fig. 15. The contour of the displacement field for the entire structure at $t = 10$ s is illustrated in Fig. 16. Here the coarse element with 26 nodes (Model C) is used. The results show that the proposed method still performs well in computing the displacement values in small-scale feature points and the displacement field in the entire structure with non-periodic inclusions.

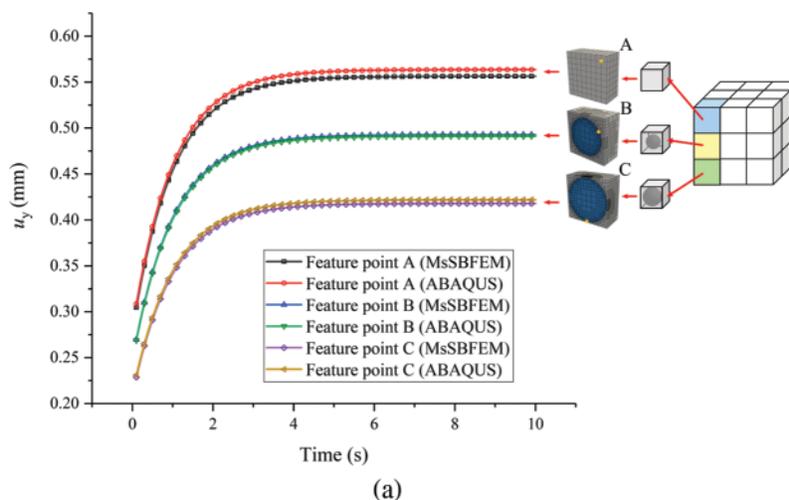


Figure 15: (Continued)

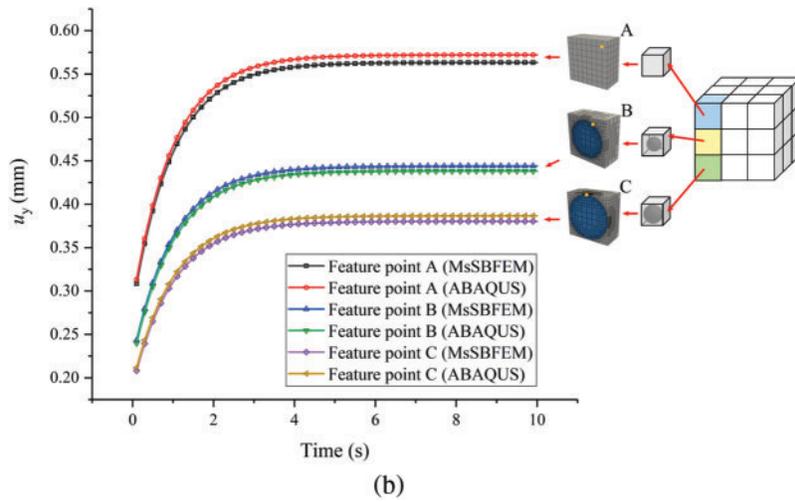


Figure 15: Displacement-time curves of different small-scale feature points (a) material parameters case I (b) material parameters case II

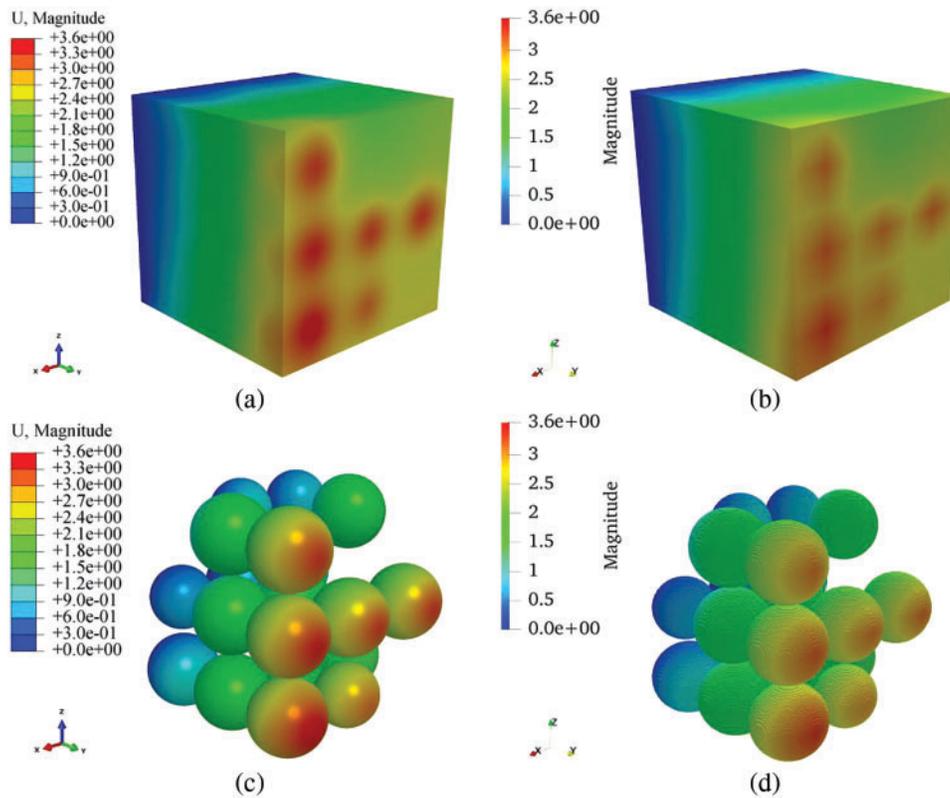


Figure 16: Displacement field cloud map of Case II (t = 10) (a) reference solution for entire structure (b) MsSBFM solution for entire structure (c) reference solution for inclusion displacements (d) MsSBFM solution for inclusions

4.3 A Concrete Beam

Consider a heterogeneous concrete beam under uniaxial tension with the boundary conditions and geometric parameters shown in Fig. 17. For simplicity, the beam is assumed to have 16 periodic subdomains for simplicity, and the CT images [68–70] of the microstructure of one subdomain are shown in Fig. 18 with the material parameters listed in Table 10. A total of 16 coarse elements are used in large-scale as shown in Fig. 17b.

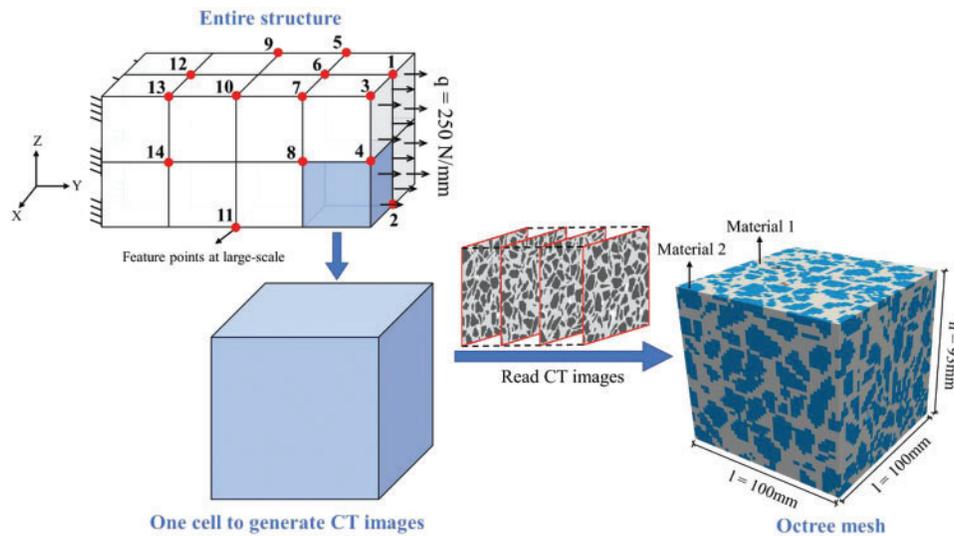


Figure 17: Image-based mesh generation in small-scale of a concrete beam

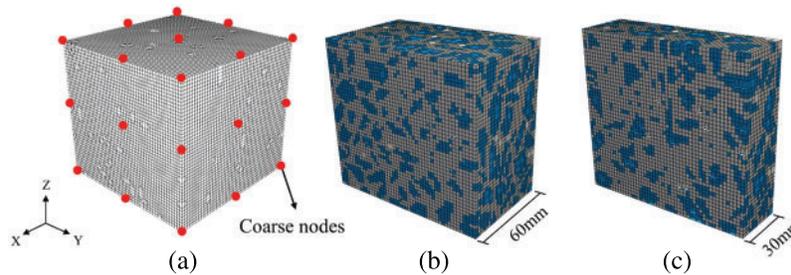


Figure 18: Coarse SBFE and its octree mesh (a) SBFEs for coarse element (b) octree mesh for material 1 (c) octree mesh for material 2

Table 10: Material parameters for the concrete beam

Mat1			Mat2			ν
E_1 (MPa)	E_2 (MPa)	η (MPa * s)	E_1 (MPa)	E_2 (MPa)	η (MPa * s)	
50000	50000	50000	25000	25000	25000	0.3

A fine grid in small-scale is shown in Fig. 18. Similar to the previous example in Section 4.1, a total of 26 nodes for each coarse element are used. This means that we use 209 coarse nodes in this example.

Table 11 shows the displacements at feature points in large-scale (shown in Fig. 17) and their relative errors e_r when the $t = 10$ s. The reference solution is provided by DNS in commercial software Abaqus with the use of 1885529 nodes as shown in Fig. 19. The average e_r of the proposed method at feature points is 1.87%, indicating that the proposed method can still obtain a good accuracy when the microstructures are very complex.

Table 11: Comparisons of u_r and e_r at large-scale ($t = 10$ s)

The feature points at large-scale	Reference (mm)	MsSBFEM (mm)	e_r (%)
1	5.709	5.592	2.05
2	5.614	5.558	1.00
3	5.693	5.605	1.55
4	5.680	5.598	1.44
5	4.258	4.157	2.36
6	4.255	4.174	1.90
7	4.264	4.193	1.66
8	4.247	4.193	1.26
9	2.801	2.741	2.13
10	2.823	2.768	1.94
11	2.802	2.745	2.05
12	1.372	1.345	2.00
13	1.430	1.395	2.46
14	1.383	1.349	2.45
Average e_r	1.87%		



Figure 19: FE mesh in reference solution for DNS by Abaqus (1885529 nodes)

Figs. 20 and 21 show the displacement-time curves of the feature points on different profiles of one specific coarse element and the contour of the displacement field of this coarse element at various times, respectively. As shown in the figure, the results of the proposed model also align well with the reference solution in small-scale.

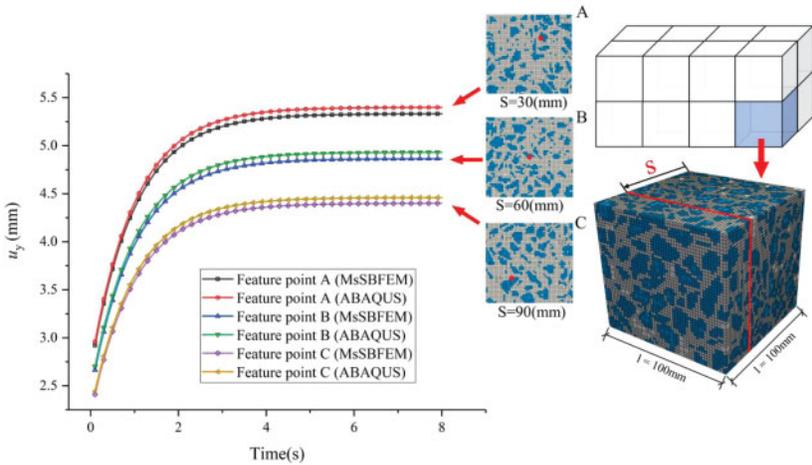


Figure 20: Displacement-time curves of different small-scale feature points in a coarse element

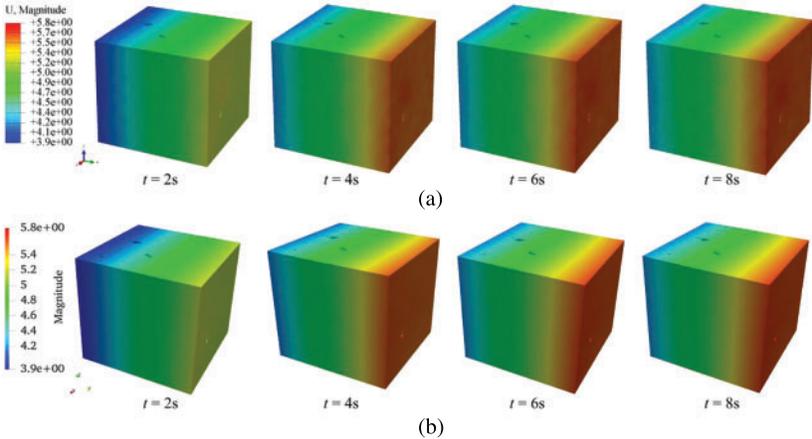


Figure 21: Displacement field cloud maps of feature coarse elements at different time steps (a) FE solution of Abaqus (b) proposed method

Table 12 shows the comparison of computational efficiency between the DNS and the MsSBFEM, in which Abaqus uses 1885529 nodes and MsSBFEM uses 209 coarse nodes to solve the problem. It is shown that the DNS solution takes 186541 s and MsSBFEM takes 12214 s, the computational efficiency has been improved by 154 times.

Table 12: Comparisons of computing time

Solution method	Computing time	
DNS	186541 s	
MsSBFEM	Construction of the numerical base function	7455 s
	Viscoelastic solution	4759 s
	Total	12214 s

5 Conclusions

By integrating the advantages of the MsSBFEM and the temporally piecewise adaptive algorithm, a new numerical algorithm is developed for multiscale analysis of three-dimensional viscoelastic problems of heterogeneous materials by extending the two-dimensional method presented in our previous work. The major merits of this study include

1. The proposed multiscale method provides an effective tool for handling three-dimensional viscoelastic analysis, that can strongly reproduce realistic situations for heterogeneous viscoelastic analysis, e.g., in the viscoelastic analysis for concrete structures with complex microstructures in Example 3.
2. The solution scale for three dimensions can be significantly reduced by the proposed multiscale model and the computational accuracy is still satisfactory. For instance, in Example 3, only 209 coarse nodes are used in the proposed method instead of 1885529 nodes in the DNS reference solution, but the average relative error of the proposed method is 1.87%.
3. Using the octree SBFEM to construct the numerical base functions, the image-based analysis can be conveniently achieved for complex three-dimensional microstructures.
4. In the time domain, the temporally piecewise adaptive algorithm ensures stable computational accuracy in both large and small scales with different time steps.
5. Based on the flexibility of the octree SBFEM, nodes can be added on large-scale without changing the mesh on small scale, which can significantly improve the calculation accuracy of the multiscale three-dimensional viscoelastic analysis.

A limitation of this work is that we currently use regular octree meshes in all of the structures; therefore, there could be jagged shapes at the interfaces, producing some errors because the smooth boundaries cannot be exactly described. Although the computational accuracy of the current model is acceptable compared with reference solutions and can also be increased by mesh refinement, a direct cutting mesh method will be further studied in order to better describe material interfaces and increase computational accuracy.

In summary, the proposed method provides an effective new approach for solving three-dimensional multiscale viscoelastic problems. This method can be extended further to solve other three-dimensional time-dependent problems, e.g., the dynamics analysis and the transient heat transfer analysis. Moreover, the extension of the proposed method to nonlinear multiscale analysis is also underway.

Acknowledgement: The authors acknowledge the advices on SBFEM from Prof. Chongmin Song from UNSW and the advices on MsFEM from Prof. Yonggang Zheng from DUT.

Funding Statement: The research leading to this paper was funded by the NSFC Grants (12072063, 11972109), Grant of State Key Laboratory of Structural Analysis for Industrial Equipment (S22403), National Key Research and Development Program of China (2020YFB1708304) and Alexander von Humboldt Foundation (1217594).

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Xu Xu, Xiaoteng Wang, Yiqian He, Haitian Yang; data collection: Zhenjun Yang, Xu Xu; analysis and interpretation of results: Xu Xu, Yiqian He; draft manuscript preparation: Xu Xu, Xiaoteng Wang, Yiqian He; draft review: Yiqian He, Zhenjun Yang, Haitian Yang. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are available from the first and corresponding authors upon reasonable request.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Wang Y, Artz T, Beel A, Shao XD, Fish J. Computational analyses of flexural behavior for ultrahigh performance fiber reinforced concrete bridge decks. *Int J Multiscale Com.* 2020;18(4):477–91. doi:10.1615/IntJMultCompEng.v18.i4.
2. Deng L, Fan S, Zhang Y, Huang ZG, Zhou HM, Jiang SF, et al. Multiscale modeling and simulation of polymer blends in injection molding: a review. *Polymers.* 2021;13(21):3783. doi:10.3390/polym13213783.
3. Sabar H, Berveiller M, Favier V, Berbenni S. A new class of micro-macro models for elastic-viscoplastic heterogeneous materials. *Int J Solids Struct.* 2002;39(12):3257–76. doi:10.1016/S0020-7683(02)00256-1.
4. Haddad H, Guessasma M, Fortin J. Heat transfer by conduction using DEM-FEM coupling method. *Comp Mater Sci.* 2014;81:339–47. doi:10.1016/j.commatsci.2013.08.033.
5. Markovic D, Ibrahimbegovic A. On micro-macro interface conditions for micro scale based FEM for inelastic behavior of heterogeneous materials. *Comput Method Appl M.* 2004;193(48–51):5503–23.
6. He Y, Guo J, Yang H, Fu Q. Numerical prediction of effective properties for heterogeneous viscoelastic materials via a temporally recursive adaptive quadtree SBFEM. *Finite Elem Anal Des.* 2020;177:103426. doi:10.1016/j.finel.2020.103426.
7. Zhang H, Wu J, Fu Z. Extended multiscale finite element method for elasto-plastic analysis of 2D periodic lattice truss materials. *Comput Mech.* 2010;45:623–35. doi:10.1007/s00466-010-0475-3.
8. Voigt W. Ueber die Beziehung zwischen den beiden Elasticitätsconstanten isotroper Körper. *Ann Phys.* 1889;274(12):573–87. doi:10.1002/andp.v274:12.
9. Reuß A. Berechnung der fließgrenze von mischkristallen auf grund der plastizitätsbedingung für einkristalle. *Zamm-Z Angew Math Me.* 1929;9(1):49–58. doi:10.1002/zamm.v9:1.
10. Hashin ZA, Shtrikman S. On some variational principles in anisotropic and nonhomogeneous elasticity. *J Mech Phys Solids.* 1962;10(4):335–42. doi:10.1016/0022-5096(62)90004-2.
11. Zecevic M, Lebensohn RA. New robust self-consistent homogenization schemes of elasto-viscoplastic polycrystals. *Int J Solids Struct.* 2020;202:434–53. doi:10.1016/j.ijsolstr.2020.05.032.
12. Wei H, Wang X, Chen C, Jiang K. An adaptive virtual element method for the polymeric self-consistent field theory. *Comput Math Appl.* 2023;141:242–54. doi:10.1016/j.camwa.2023.01.039.
13. Liu TR, Yang Y, Bacarreza OR, Tang S, Aliabadi M. An extended full field self-consistent cluster analysis framework for woven composite. *Int J Solids Struct.* 2023;281:112407. doi:10.1016/j.ijsolstr.2023.112407.
14. Mori T, Tanaka K. Average stress in matrix and average elastic energy of materials with misfitting inclusions. *Acta Metall.* 1973;21(5):571–4. doi:10.1016/0001-6160(73)90064-3.
15. Fedotov A. Mori-Tanaka experimental-analytical model for predicting engineering elastic moduli of composite materials. *Compos Part B-Eng.* 2022;232:109635. doi:10.1016/j.compositesb.2022.109635.
16. Barral M, Chatzigeorgiou G, Meraghni F, Léon R. Homogenization using modified Mori-Tanaka and TFA framework for elastoplastic-viscoelastic-viscoplastic composites: theory and numerical validation. *Int J Plasticity.* 2020;127:102632. doi:10.1016/j.ijplas.2019.11.011.
17. Zhang H, Wu JK, Lü J, Fu Z. Extended multiscale finite element method for mechanical analysis of heterogeneous materials. *Acta Metall Sin.* 2010;26(6):899–920.
18. He Z, Pindera MJ. Finite volume-based asymptotic homogenization of periodic materials under in-plane loading. *Int J Appl Mech.* 2020;87(12):121010. doi:10.1115/1.4048201.

19. He Z, Liu J, Chen Q. Higher-order asymptotic homogenization for piezoelectric composites. *Int J Solids Struct.* 2023;264:112092. doi:10.1016/j.ijsolstr.2022.112092.
20. Liu P, Liu A, Peng H, Tian L, Liu J, Lu L. Mechanical property profiles of microstructures via asymptotic homogenization. *Comput Graphics.* 2021;100:106–15. doi:10.1016/j.cag.2021.07.021.
21. Sun W, Fish J, Dhia HB. A variant of the s-version of the finite element method for concurrent multiscale coupling. *Int J Multiscale Com.* 2018;16(2):187–207. doi:10.1615/IntJMCompEng.v16.i2.
22. Sun W, Fish J, Ni P. Superposition-based concurrent multiscale approaches for poromechanics. *Int J Numer Meth Eng.* 2021;122(24):7328–53. doi:10.1002/nme.v122.24.
23. Sun W, Bao S, Zhou J, Ni P. Concurrent multiscale analysis of anti-seepage structures in embankment dam based on the nonlinear Arlequin method. *Eng Anal Bound Elem.* 2023;149:231–47. doi:10.1016/j.enganabound.2023.01.039.
24. Zhao W, Du C, Jiang S. An adaptive multiscale approach for identifying multiple flaws based on XFEM and a discrete artificial fish swarm algorithm. *Comput Method Appl M.* 2018;339:341–57. doi:10.1016/j.cma.2018.04.037.
25. Feyel F, Chaboche JL. FE2 multiscale approach for modelling the elastoviscoplastic behaviour of long fibre SiC/Ti composite materials. *Comput Method Appl M.* 2000;183(3–4):309–30.
26. Zhi J, Raju K, Tay TE, Tan VBC. Multiscale analysis of thermal problems in heterogeneous materials with Direct FE2 method. *Int J Numer Meth Eng.* 2021;122(24):7482–7503. doi:10.1002/nme.v122.24.
27. Tikarrouchine E, Benaarbia A, Chatzigeorgiou G, Meraghni F. Non-linear FE2 multiscale simulation of damage, micro and macroscopic strains in polyamide 66-woven composite structures: analysis and experimental validation. *Compos Struct.* 2021;255:112926. doi:10.1016/j.compstruct.2020.112926.
28. Jezdan G, Govindjee S, Hackl K. A variational approach to effective models for inelastic systems. *Int J Solids Struct.* 2024;286:112567.
29. Babuška I, Osborn JE. Generalized finite element methods: their performance and their relation to mixed methods. *Siam J Numer Anal.* 1983;20(3):510–36. doi:10.1137/0720034.
30. Babuška I, Caloz G, Osborn JE. Special finite element methods for a class of second order elliptic problems with rough coefficients. *Siam J Numer Anal.* 1994;31(4):945–81. doi:10.1137/0731051.
31. Coda H, Sanches R, Paccola R. Alternative multiscale material and structures modeling by the finite-element method. *Eng Comput.* 2020;38:1–19.
32. Zhelnin M, Kostina A, Plekhov O. Variational multiscale finite-element methods for a nonlinear convection-diffusion–reaction equation. *J Appl Mech Tech Phys.* 2020;61:1128–39. doi:10.1134/S0021894420070226.
33. Ye C, Dong H, Cui J. Convergence rate of multiscale finite element method for various boundary problems. *J Comput Appl Math.* 2020;374:112754. doi:10.1016/j.cam.2020.112754.
34. Zhang H, Fu Z, Wu J. Coupling multiscale finite element method for consolidation analysis of heterogeneous saturated porous media. *Adv Water Resour.* 2009;32(2):268–79. doi:10.1016/j.advwatres.2008.11.002.
35. Zhang H, Liu H, Wu J. A uniform multiscale method for 2D static and dynamic analyses of heterogeneous materials. *Int J Numer Meth Eng.* 2013;93(7):714–46. doi:10.1002/nme.v93.7.
36. Zhang H, Lu M, Zheng Y, Zhang S. General coupling extended multiscale FEM for elasto-plastic consolidation analysis of heterogeneous saturated porous media. *Int J Numer Anal Met.* 2015;39(1):63–95. doi:10.1002/nag.v39.1.
37. Zhang S, Yang D, Zhang H, Zheng Y. Coupling extended multiscale finite element method for thermoelastic analysis of heterogeneous multiphase materials. *Comput Struct.* 2013;121:32–49. doi:10.1016/j.compstruc.2013.03.001.
38. Klimczak M, Cecot W. An adaptive MsFEM for nonperiodic viscoelastic composites. *Int J Numer Meth Eng.* 2018;114(8):861–81. doi:10.1002/nme.v114.8.
39. Kabel M, Merkert D, Schneider M. Use of composite voxels in FFT-based homogenization. *Comput Method Appl M.* 2015;294:168–88. doi:10.1016/j.cma.2015.06.003.

40. Song C, Wolf JP. The scaled boundary finite-element method—alias consistent infinitesimal finite-element cell method—for elastodynamics. *Comput Method Appl M.* 1997;147(3–4):329–55.
41. Jiang S, Sun L, Ooi ET, Ghaemian M, Du C. Automatic mesoscopic fracture modelling of concrete based on enriched SBFEM space and quad-tree mesh. *Constr Build Mater.* 2022;350:128890. doi:10.1016/j.conbuildmat.2022.128890.
42. Ooi ET, Man H, Natarajan S, Song C. Adaptation of quadtree meshes in the scaled boundary finite element method for crack propagation modelling. *Eng Fract Mech.* 2015;144:101–17. doi:10.1016/j.engfracmech.2015.06.083.
43. Zhang P, Du C, Zhao W, Sun L. Dynamic crack face contact and propagation simulation based on the scaled boundary finite element method. *Comput Method Appl M.* 2021;385:114044. doi:10.1016/j.cma.2021.114044.
44. Song C. The scaled boundary finite element method in structural dynamics. *Int J Numer Meth Eng.* 2009;77(8):1139–71. doi:10.1002/nme.v77:8.
45. Dsouza SM, Varghese TM, Ooi ET, Natarajan S, Bordas SP. Treatment of multiple input uncertainties using the scaled boundary finite element method. *Appl Math Model.* 2021;99:538–54. doi:10.1016/j.apm.2021.06.021.
46. Johari A, Heydari A. Reliability analysis of seepage using an applicable procedure based on stochastic scaled boundary finite element method. *Eng Anal Bound Elem.* 2018;94:44–59. doi:10.1016/j.enganabound.2018.05.015.
47. Khajah T, Liu L, Song C, Gravenkamp H. Shape optimization of acoustic devices using the scaled boundary finite element method. *Wave Motion.* 2021;104:102732. doi:10.1016/j.wavemoti.2021.102732.
48. Zhang W, Xiao Z, Liu C, Mei Y, Youn SK, Guo X. A scaled boundary finite element based explicit topology optimization approach for three-dimensional structures. *Int J Numer Meth Eng.* 2020;121(21):4878–4900. doi:10.1002/nme.v121.21.
49. Zhang J, Zhao M, Eisenträger S, Du X, Song C. An asynchronous parallel explicit solver based on scaled boundary finite element method using octree meshes. *Comput Method Appl M.* 2022;401:115653. doi:10.1016/j.cma.2022.115653.
50. Genes MC. Dynamic analysis of large-scale SSI systems for layered unbounded media via a parallelized coupled finite-element/boundary-element/scaled boundary finite-element model. *Eng Anal Bound Elem.* 2012;36(5):845–57. doi:10.1016/j.enganabound.2011.11.013.
51. Yu B, Hu P, Saputra AA, Gu Y. The scaled boundary finite element method based on the hybrid quadtree mesh for solving transient heat conduction problems. *Appl Math Model.* 2021;89:541–71. doi:10.1016/j.apm.2020.07.035.
52. Song C, Ooi ET, Natarajan S. A review of the scaled boundary finite element method for two-dimensional linear elastic fracture mechanics. *Eng Fract Mech.* 2018;187:45–73. doi:10.1016/j.engfracmech.2017.10.016.
53. Zou D, Chen K, Kong X, Liu J. An enhanced octree polyhedral scaled boundary finite element method and its applications in structure analysis. *Eng Anal Bound Elem.* 2017;84:87–107. doi:10.1016/j.enganabound.2017.07.007.
54. Wang X, Yang H, He Y. A temporally piecewise adaptive multiscale scaled boundary finite element method to solve two-dimensional heterogeneous viscoelastic problems. *Eng Anal Bound Elem.* 2023;155:738–53. doi:10.1016/j.enganabound.2023.07.006.
55. Wang X, Yang H, He Y. A multiscale scaled boundary finite element method solving steady-state heat conduction problem with heterogeneous materials. *Numer Heat Tr B-Fund.* 2023;83(6):345–66. doi:10.1080/10407790.2022.2160850.
56. Zhang J, Eisenträger S, Zhan Y, Saputra A, Song C. Direct point-cloud-based numerical analysis using octree meshes. *Comput Struct.* 2023;289:107175. doi:10.1016/j.compstruc.2023.107175.

57. Liu L, Zhang J, Song C, He K, Saputra AA, Gao W. Automatic scaled boundary finite element method for three-dimensional elastoplastic analysis. *Int J Mech Sci.* 2020;171:105374. doi:10.1016/j.ijmecsci.2019.105374.
58. Liu L, Zhang J, Song C, Birk C, Saputra AA, Gao W. Automatic three-dimensional acoustic-structure interaction analysis using the scaled boundary finite element method. *J Comput Phys.* 2019;395:432–60. doi:10.1016/j.jcp.2019.06.033.
59. Liu Y, Saputra AA, Wang J, Tin-Loi F, Song C. Automatic polyhedral mesh generation and scaled boundary finite element analysis of STL models. *Comput Method Appl M.* 2017;313:106–32. doi:10.1016/j.cma.2016.09.038.
60. Ya S, Eisenträger S, Song C, Li J. An open-source ABAQUS implementation of the scaled boundary finite element method to study interfacial problems using polyhedral meshes. *Comput Method Appl M.* 2021;381:113766. doi:10.1016/j.cma.2021.113766.
61. Talebi H, Saputra A, Song C. Stress analysis of 3D complex geometries using the scaled boundary polyhedral finite elements. *Comput Mech.* 2016;58:697–715. doi:10.1007/s00466-016-1312-0.
62. Zhang J, Natarajan S, Ooi ET, Song C. Adaptive analysis using scaled boundary finite element method in 3D. *Comput Method Appl M.* 2020;372:113374. doi:10.1016/j.cma.2020.113374.
63. Saputra AA, Eisenträger S, Gravenkamp H, Song C. Three-dimensional image-based numerical homogenisation using octree meshes. *Comput Struct.* 2020;237:106263. doi:10.1016/j.compstruc.2020.106263.
64. Gravenkamp H, Saputra AA, Eisenträger S. Three-dimensional image-based modeling by combining SBFEM and transfinite element shape functions. *Comput Mech.* 2020;66(4):911–30. doi:10.1007/s00466-020-01884-4.
65. Wang CS, He YQ, Yang HT. A SBFEM and sensitivity analysis based algorithm for solving inverse viscoelastic problems. *Eng Anal Bound Elem.* 2019;106:588–98. doi:10.1016/j.enganabound.2019.06.014.
66. He Y, Yang H. Solving viscoelastic problems by combining SBFEM and a temporally piecewise adaptive algorithm. *Mech Time-Depend Mat.* 2017;21:481–97. doi:10.1007/s11043-017-9338-z.
67. Wang C, Long X, He Y, Yang H, Han X. An adaptive recursive SBFE algorithm for the statistical analysis of stochastic viscoelastic problems. *Comput Method Appl M.* 2022;395:114878. doi:10.1016/j.cma.2022.114878.
68. Ren W, Yang Z, Sharma R, Zhang C, Withers PJ. Two-dimensional X-ray CT image based meso-scale fracture modelling of concrete. *Eng Fract Mech.* 2015;133:24–39. doi:10.1016/j.engfracmech.2014.10.016.
69. Yang Z, Ren W, Sharma R, McDonald S, Mostafavi M, Vertyagina Y, et al. *In-situ* X-ray computed tomography characterisation of 3D fracture evolution and image-based numerical homogenisation of concrete. *Cement Concrete Comp.* 2017;75:74–83. doi:10.1016/j.cemconcomp.2016.10.001.
70. Huang Y, Yang Z, Ren W, Liu G, Zhang C. 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.* 2015;67:340–52.

Appendix A. Calculation of Stiffness Matrix of a 3D SBFE

In this paper, the octree grid is used in the small-scale solution, each octree cell is modeled as a scaled boundary cube element, and the cubic centroid O is set as the scaling center. The Cartesian coordinates within the domain $(\hat{x}, \hat{y}, \hat{z})$ are described by scaling the coordinates on the boundary (x, y, z) along the local radial coordinate ξ , as shown in Fig. A1. For the surface element on the boundary of the cube element, the local coordinate system (η, ζ) is established, based on this coordinate

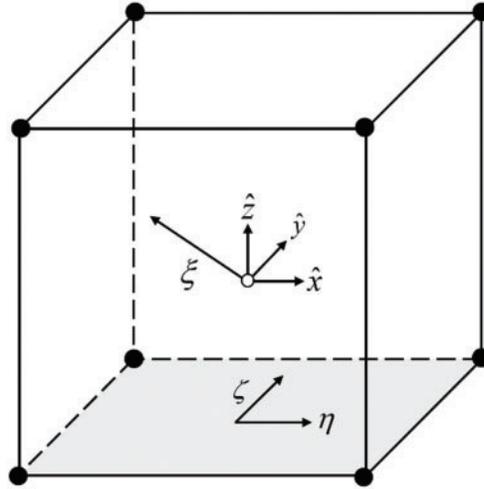


Figure A1: Scaled boundary coordinate system

system, the surface element is divided into tetrahedral element and triangular element. For local radial coordinates ξ , at the scaling center $\xi = 0$, on the boundary $\xi = 1$.

Assuming that the center of the Cartesian coordinates coincides with the scaling center of scaled boundary coordinates, the transformation relationship between the two coordinate systems is

$$\begin{cases} \hat{x}(\xi, \eta, \zeta) = \xi x(\eta, \zeta) = \xi \mathbf{N}(\eta, \zeta) \mathbf{x} \\ \hat{y}(\xi, \eta, \zeta) = \xi y(\eta, \zeta) = \xi \mathbf{N}(\eta, \zeta) \mathbf{y} \\ \hat{z}(\xi, \eta, \zeta) = \xi z(\eta, \zeta) = \xi \mathbf{N}(\eta, \zeta) \mathbf{z} \end{cases} \quad (\text{A1})$$

where $\mathbf{N}(\eta, \zeta)$ is the two-dimensional shape function and $\mathbf{x}, \mathbf{y}, \mathbf{z}$ is the node coordinate vector of each surface.

The governing differential equations of 3D elasticity problems in the absence of body forces can be formulated as

$$\mathbf{L}^T \boldsymbol{\sigma} = \mathbf{0} \quad (\text{A2})$$

where $\boldsymbol{\sigma}$ is the stress vector inside the cube element and \mathbf{L} is the linear differential operator, \mathbf{L} can be written as

$$\mathbf{L} = \mathbf{b}_1(\eta, \zeta) \frac{\partial}{\partial \xi} + \frac{1}{\xi} \left[\mathbf{b}_2(\eta, \zeta) \frac{\partial}{\partial \eta} + \mathbf{b}_3(\eta, \zeta) \frac{\partial}{\partial \zeta} \right] \quad (\text{A3})$$

where

$$\begin{cases} \mathbf{b}_1(\eta, \zeta) = \frac{1}{|\mathbf{J}|} \mathbf{G}_1 \\ \mathbf{b}_2(\eta, \zeta) = -\frac{1}{|\mathbf{J}|} (\eta \mathbf{G}_1 + \mathbf{G}_2) \\ \mathbf{b}_3(\eta, \zeta) = -\frac{1}{|\mathbf{J}|} (\zeta \mathbf{G}_1 + \mathbf{G}_3) \end{cases} \quad (\text{A4})$$

$$|\mathbf{J}| = x(y_{,\eta}z_{,\zeta} - z_{,\eta}y_{,\zeta}) + y(z_{,\eta}x_{,\zeta} - x_{,\eta}z_{,\zeta}) + z(x_{,\eta}y_{,\zeta} - y_{,\eta}x_{,\zeta}) \quad (\text{A5})$$

In Eq. (A4), $\mathbf{G}_1, \mathbf{G}_2$ and \mathbf{G}_3 are the coordinate correlation matrix of the surface element.

The displacement solution of any point in the element can be obtained by radial displacement $\mathbf{u}(\xi)$ interpolation.

$$\mathbf{u}(\xi, \eta, \zeta) = \mathbf{N}^u(\eta, \zeta) \mathbf{u}(\xi) \quad (\text{A6})$$

where $\mathbf{N}^u(\eta, \zeta)$ is the displacement shape function matrix, its specific expression is as follows:

$$\mathbf{N}^u(\eta, \zeta) = [N_1(\eta, \zeta) \mathbf{I}, N_2(\eta, \zeta) \mathbf{I}, \dots, N_n(\eta, \zeta) \mathbf{I}] \quad (\text{A7})$$

where \mathbf{I} is identity matrix and n are the total number of nodes on a surface element. Thus, the strains can be expressed using Eqs. (A3) and (A7) as

$$\boldsymbol{\varepsilon}(\xi, \eta, \zeta) = \mathbf{B}_1 \mathbf{u}(\xi)_{,\xi} + \frac{1}{\xi} \mathbf{B}_2 \mathbf{u}(\xi)_{,\xi} \quad (\text{A8})$$

where

$$\begin{cases} \mathbf{B}_1 = \mathbf{b}_1(\eta, \zeta) \mathbf{N}^u(\eta, \zeta) \\ \mathbf{B}_2 = \mathbf{b}_2(\eta, \zeta) \mathbf{N}^u(\eta, \zeta)_{,\eta} + \mathbf{b}_3(\eta, \zeta) \mathbf{N}^u(\eta, \zeta)_{,\zeta} \end{cases} \quad (\text{A9})$$

According to the principle of virtual work, the three-dimensional SBFE equation without body forces can be written as

$$\mathbf{E}_0 \xi^2 \mathbf{u}(\xi)_{,\xi\xi} + (2\mathbf{E}_0 - \mathbf{E}_1 + \mathbf{E}_1^T) \xi \mathbf{u}(\xi)_{,\xi} + (\mathbf{E}_1^T - \mathbf{E}_2) \mathbf{u}(\xi) = 0 \quad (\text{A10})$$

The coefficient matrix is as follows:

$$\begin{cases} \mathbf{E}_0 = \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{B}_1^T \mathbf{C} \mathbf{B}_1 |\mathbf{J}| d\eta d\zeta \\ \mathbf{E}_1 = \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{B}_2^T \mathbf{C} \mathbf{B}_1 |\mathbf{J}| d\eta d\zeta \\ \mathbf{E}_2 = \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{B}_2^T \mathbf{C} \mathbf{B}_2 |\mathbf{J}| d\eta d\zeta \end{cases} \quad (\text{A11})$$

where \mathbf{C} is the 6×6 elasticity matrix for three-dimensional problems. The internal node forces on each surface element can be written as

$$\mathbf{q}(\xi) = \xi (\mathbf{E}_0 \xi \mathbf{u}(\xi)_{,\xi} + \mathbf{E}_1^T \mathbf{u}(\xi)) \quad (\text{A12})$$

Transforming Eqs. (A10) and (A12) into a system of first-order ordinary differential equations

$$\xi \begin{Bmatrix} \xi^{0.5} \mathbf{u}(\xi) \\ \xi^{-0.5} \mathbf{q}(\xi) \end{Bmatrix}_{,\xi} = -\mathbf{Z} \begin{Bmatrix} \xi^{0.5} \mathbf{u}(\xi) \\ \xi^{-0.5} \mathbf{q}(\xi) \end{Bmatrix} \quad (\text{A13})$$

where \mathbf{Z} is the Hamiltonian matrix, it can be expressed as

$$\mathbf{Z} = \begin{bmatrix} \mathbf{E}_0^{-1} \mathbf{E}_1^T - 0.5\mathbf{I} & -\mathbf{E}_0^{-1} \\ -\mathbf{E}_2 + \mathbf{E}_1 \mathbf{E}_0^{-1} \mathbf{E}_1^T & -(\mathbf{E}_1 \mathbf{E}_0^{-1} - 0.5\mathbf{I}) \end{bmatrix} \quad (\text{A14})$$

The matrix \mathbf{Z} is decomposed using a Schur decomposition with block diagonalization

$$\mathbf{Z}\mathbf{V} = \mathbf{V}\mathbf{S} \quad (\text{A15})$$

where \mathbf{S} and \mathbf{V} are the real Schur form matrix and the transformation matrix, respectively. Divide the matrices \mathbf{S} and \mathbf{V} into submatrices of the same size, as shown in following equation:

$$\mathbf{S} = \begin{bmatrix} \mathbf{S}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_{22} \end{bmatrix} \quad (\text{A16a})$$

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_{11} & \mathbf{V}_{12} \\ \mathbf{V}_{21} & \mathbf{V}_{22} \end{bmatrix} \quad (\text{A16b})$$

All the eigenvalues of submatrix \mathbf{S}_{11} have negative real parts which corresponds to the solution of a bounded domain. Therefore, the general solutions for the displacements and internal nodal forces of the bounded domain are

$$\mathbf{u}(\xi) = \mathbf{V}_{11}\xi^{-\mathbf{S}_{11}-0.5\mathbf{I}}\mathbf{c} \quad (\text{A17a})$$

$$\mathbf{q}(\xi) = \mathbf{V}_{21}\xi^{-\mathbf{S}_{11}+0.5\mathbf{I}}\mathbf{c} \quad (\text{A17b})$$

where \mathbf{c} is a vector of integration constants. Consequently, the static stiffness matrix \mathbf{K} is defined as

$$\mathbf{K} = \mathbf{V}_{21}\mathbf{V}_{11}^{-1} \quad (\text{A18})$$