We are IntechOpen, the world's leading publisher of Open Access books
Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Chapter

A High-Order Finite Volume Method for 3D Elastic Modelling on Unstructured Meshes

Wensheng Zhang

Abstract

In this chapter, a new efficient high-order finite volume method for 3D elastic modelling on unstructured meshes is developed. The stencil for the high-order polynomial reconstruction is generated by subdividing the relative coarse tetrahedrons. The reconstruction on the stencil is performed by using cell-averaged quantities represented by the hierarchical orthonormal basis functions. Unlike the traditional high-order finite volume method, the new method has a very local property like the discontinuous Galerkin method. Furthermore, it can be written as an inner-split computational scheme which is beneficial to reducing computational amount. The reconstruction matrix is invertible and remains unchanged for all tetrahedrons, and thus it can be pre-computed and stored before time evolution. These special advantages facilitate the parallelization and high-order computations. The high-order accuracy in time is obtained by the Runge-Kutta method. Numerical computations including a 3D real model with complex topography demonstrate the effectiveness and good adaptability to complex topography.

Keywords: numerical solutions, computational seismology, 3D elastic wave, wave propagation, high-order finite volume method, unstructured meshes

1. Introduction

Wave propagation based on wave equations has important applications in geophysics. It is usually used as a powerful tool to detect the structures of reservoir. Thus solving wave equations efficiently and accurately is always an important research topic. There are several types of numerical methods to solve wave equations, for example, the finite difference (FD) method [1, 2], the pseudo-spectral (PS) method [3, 4], the finite element (FE) method [5–9], the spectral element (SE) method [10–14], the discontinuous Galerkin (DG) method [15–18], and the finite volume (FV) method [19–22]. Each numerical method has its own inherent advantages and disadvantages. For example, the FD method is efficient and relatively easy to implement, but the inherent restriction of using regular meshes limits its application to complex topography. The FE method has good adaptability to complex topography, but it has huge computational cost. In this chapter, the FV method is the key consideration.

In order to simulate wave propagation on unstructured meshes efficiently, the FV method is a good choice due to its high computational efficiency and good

adaptability to complex geometry. In this chapter an efficient FV method for 3D elastic wave simulation on unstructured meshes is developed. It incorporates some nice features from the DG and FV methods [15–17, 19, 20, 23] and the spectral FV (SFV) method [24–26]. In our method, the computational domain is first meshed with relative coarse tetrahedral elements in 3D or triangle elements in 2D. Then, each element is further divided as a collection of finer subelements to form a stencil. The high-order polynomial reconstruction is performed on this stencil by using local cell-averaged values on the finer elements. The resulting reconstruction matrix on all coarse elements remains unchanged, and it can be pre-computed before time evolution. Moreover, the method can be written as an inner-split computational scheme. These two advantages of our method are very beneficial to enhancing the parallelization and reducing computational cost.

The rest of this chapter is organized as follows. In Section 2, the theory is described in detail. In Section 3, numerical results are given to illustrate the effectiveness of our method. Finally, the conclusion is given in Section 4.

2. Theory

2.1 The governing equation

The three-dimensional (3D) elastic wave equation with external sources in velocity-stress formulation can be written as the following system [1, 15]:

$$
\begin{cases}
\dfrac{\partial \sigma_{xx}}{\partial t} - (\lambda + 2\mu)\dfrac{\partial u}{\partial x} - \lambda\dfrac{\partial v}{\partial y} - \lambda\dfrac{\partial w}{\partial z} = g_1, \\[2mm]
\dfrac{\partial \sigma_{yy}}{\partial t} - \lambda\dfrac{\partial u}{\partial x} - (\lambda + 2\mu)\dfrac{\partial v}{\partial y} - \lambda\dfrac{\partial w}{\partial z} = g_2, \\[2mm]
\dfrac{\partial \sigma_{zz}}{\partial t} - \lambda\dfrac{\partial u}{\partial x} - \lambda\dfrac{\partial v}{\partial y} - (\lambda + 2\mu)\dfrac{\partial w}{\partial z} = g_3, \\[2mm]
\dfrac{\partial \sigma_{xy}}{\partial t} - \mu\left(\dfrac{\partial v}{\partial x} + \dfrac{\partial u}{\partial y}\right) = g_4, \\[2mm]
\dfrac{\partial \sigma_{yz}}{\partial t} - \mu\left(\dfrac{\partial v}{\partial z} + \dfrac{\partial w}{\partial y}\right) = g_5, \\[2mm]
\dfrac{\partial \sigma_{xz}}{\partial t} - \mu\left(\dfrac{\partial u}{\partial z} + \dfrac{\partial w}{\partial x}\right) = g_6, \\[2mm]
\rho\dfrac{\partial u}{\partial t} - \dfrac{\partial \sigma_{xx}}{\partial x} - \dfrac{\partial \sigma_{xy}}{\partial y} - \dfrac{\partial \sigma_{xz}}{\partial z} = \rho g_7, \\[2mm]
\rho\dfrac{\partial v}{\partial t} - \dfrac{\partial \sigma_{xy}}{\partial x} - \dfrac{\partial \sigma_{yy}}{\partial y} - \dfrac{\partial \sigma_{yz}}{\partial z} = \rho g_8, \\[2mm]
\rho\dfrac{\partial w}{\partial t} - \dfrac{\partial \sigma_{xz}}{\partial x} - \dfrac{\partial \sigma_{yz}}{\partial y} - \dfrac{\partial \sigma_{zz}}{\partial z} = \rho g_9,
\end{cases}
\tag{1}
$$

where u, v, and w are the wavefield of particle velocities in x, y, and z directions, respectively; λ and μ are the Lamé coefficients and ρ is the density; $g_i(x,y,z,t)$ are the known sources; σ_{xx}, σ_{yy}, and σ_{zz} are the normal stress components while σ_{xy}, σ_{xz}, and σ_{yz} are the shear stresses. For the convenient of discussion, we rewrite Eq. (1) as the following compact form:

$$\frac{\partial \mathbf{u}}{\partial t} + A\frac{\partial \mathbf{u}}{\partial x} + B\frac{\partial \mathbf{u}}{\partial y} + C\frac{\partial \mathbf{u}}{\partial z} = \mathbf{g}, \tag{2}$$

where $\mathbf{g} = (g_1, \cdots, g_9)^T$, $\mathbf{u} = (\sigma_{xx}, \sigma_{yy}, \sigma_{zz}, \sigma_{xy}, \sigma_{yz}, \sigma_{xz}, u, v, w)^T$, and the matrices A, B, and C are all 9×9 matrices and can be obtained obviously [27].

The propagation velocities of the elastic waves are determined by the eigenvalues s_i of matrices A, B, and C and are given by

$$s_1 = -v_p, \quad s_2 = -v_s, \quad s_3 = -v_s, \quad s_4 = s_5 = s_6 = 0, \quad s_7 = v_s, \quad s_8 = v_s, \quad s_9 = v_p, \tag{3}$$

where

$$v_p = \sqrt{\frac{\lambda + 2\mu}{\rho}}, \quad v_s = \sqrt{\frac{\mu}{\rho}} \tag{4}$$

are the velocities of the compression (P) wave and the shear (S) wave velocities, respectively.

2.2 The generation of a stencil

Suppose that the 3D computational domain Ω is meshed by N_E conforming tetrahedral elements $T^{(m)}$:

$$\Omega = \bigcup_{m=1}^{N_E} T^{(m)}. \tag{5}$$

In practical computations, the integrals in the FV scheme on physical tetrahedral element $T^{(m)}$ are usually changed to be computed on its reference element. **Figure 1** shows a physical tetrahedron $T^{(m)}$ in the physical system, and $x - y - z$ is transformed into a reference element T_E in the reference system $\xi - \eta - \zeta$. Let (x_i, y_i, z_i) for $i = 1, 2, 3, 4$ be the coordinates of physical element $T^{(m)}$. The transformations between $x - y - z$ system and $\xi - \eta - \zeta$ system will be given in the final
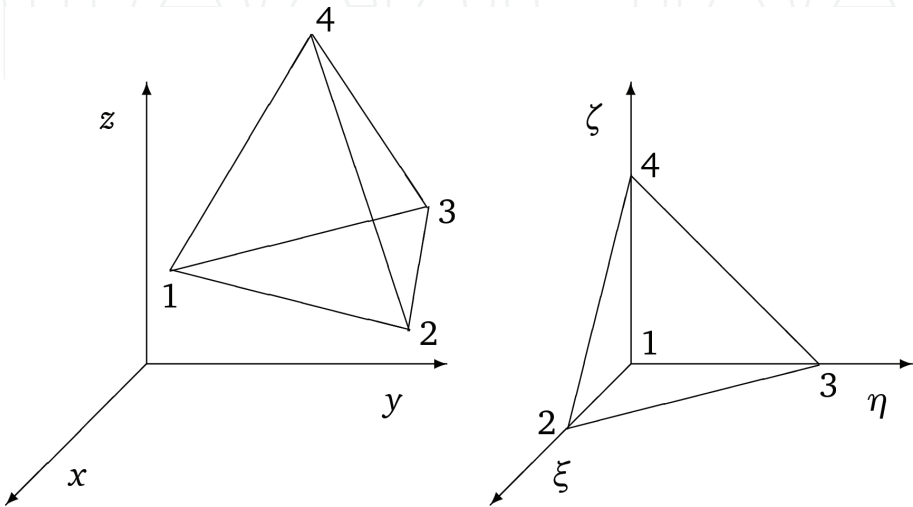

Figure 1.
The physical element $T^{(m)}$ (left) in the physical coordinate system $x - y - z$ is transformed into a reference element T_E (right) in the reference coordinate system $\xi - \eta - \zeta$.

subsection of Section 2. For convenience, let $\mathbf{x} = (x, y, z)$ and $\boldsymbol{\xi} = (\xi, \eta, \zeta)$. And denote the transformation from $\xi - \eta - \zeta$ system to $x - y - z$ system by

$$\mathbf{x} = \mathbf{x}\left(T^{(m)}, \boldsymbol{\xi}\right), \tag{6}$$

and its corresponding inverse transformation by

$$\boldsymbol{\xi} = \boldsymbol{\xi}\left(T^{(m)}, \mathbf{x}\right). \tag{7}$$

The detailed expressions of the transformations (6) and (7) will be given in Section 2.5.

Inside each T_E the solutions of Eq. (2) are approximated numerically by using a linear combination of polynomial basis functions $\phi_l(\xi, \eta, \zeta)$ and the time-dependent coefficients $\hat{\mathbf{w}}_l^{(m)}(t)$:

$$\mathbf{u}^{(m)}(\xi, \eta, \zeta, t) = \sum_{l=1}^{N_p} \hat{\mathbf{w}}_l^{(m)}(t)\phi_l(\xi, \eta, \zeta), \tag{8}$$

where N_p is the degree of freedom of a complete polynomial.

In order to construct a high-order polynomial, we need to choose a stencil. Traditionally, the elements being adjacent to the element $T^{(m)}$ are selected to form a stencil. In [20] three types of stencils, i.e., the central stencil, the primary sector stencil, and the reverse stencil, are investigated. These stencils usually choose $2N$ neighbors for the 3D reconstruction. Here N is the degree of a complete polynomial. Due to geometrical issues, the reconstruction matrix resulting from these stencils may be not invertible. This may happen when all elements are aligned in a straight line [20]. In the following, we propose to partition $T^{(m)}$ or in fact its corresponding reference element T_E into finer subelements to form a stencil. The subdivision algorithm guarantees the number of subelements is greater than the degrees of freedom of a complete polynomial. Moreover, this algorithm is easy to implement especially in 3D and for all elements whether they are internal or boundary elements.

Let N_e be the number of subelements in $T^{(m)}$ after subdividing. For a complete polynomial of degree N in 3D, a reconstruction requires at least N_p subelements, where

$$N_p = (N+1)(N+2)(N+3)/6. \tag{9}$$

In our algorithm, we guarantee N_e is always greater than N_p. As shown in **Figure 2**, we divide each edge of the reference element T_E into M uniform segments. Thus we have $N_e := M^3$ tetrahedral subelements in T_E. Note that a small subcubic in T_E consists of six tetrahedrons. With the transformations of Eqs. (6) and (7), we denote all subelements in $T^{(m)}$ for a fixed m by $T^{(m(k))}$ for $k = 1, \cdots, N_e$. In **Table 1**, the degree of a complete polynomial N and its corresponding degrees of freedom N_p are listed. Correspondingly, the number of M and N_e are also listed in **Table 1**. This algorithm for generating the stencil is easily implemented for all coarse tetrahedrons. Moreover, the reconstruction matrix resulting from this stencil is always invertible and remains unchanged for all elements $T^{(m)}$ for $m = 1, \cdots, N_E$. Note that the reconstruction matrix may be not invertible if all elements are aligned on a straight line [15]. However, this will not happen here for our algorithm.
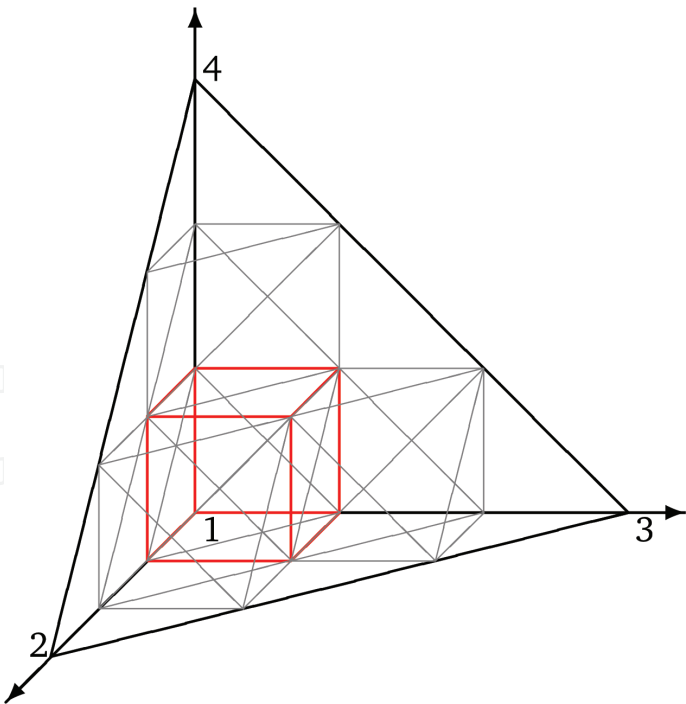
Figure 2.
The stencil obtained by subdividing the reference element T_E into $M^3 = 3^3$ tetrahedral subelements, where $M = 3$ is the number of uniform segments on each edge of T_E. Note that a small subcubic (red) in T_E consists of six tetrahedrons.

N	1	2	3	4
N_p	4	10	20	35
M	2	3	3	4
N_e	8	27	27	64

Table 1.
The degree of a complete polynomial N and its corresponding degrees of freedom N_p are listed. Correspondingly, the number of uniform segments M on each edge and the number of subelements N_e are also listed.

2.3 The high-order polynomial reconstruction

The high-order polynomial is reconstructed in each element $T^{(m)}$ or T_E. For the stencil designed above, we have

$$T^{(m)} = \bigcup_{k=1}^{N_e} T^{(m(k))}, \qquad (10)$$

where $k = 1, \cdots, N_e$ is the index for subelements in $T^{(m)}$. The FV method will use the cell-averaged quantities, i.e.,

$$\overline{\mathbf{u}}^{(m(k))} = \frac{1}{|T^{(m(k))}|} \int_{T^{(m(k))}} \mathbf{u}^{(m)}(\mathbf{x})dV, \quad k = 1, \cdots, N_e, \qquad (11)$$

to reconstruct a high-order polynomial, where $|T^{(m(k))}|$ represents the volume of the subelement $T^{(m(k))}$. The time variable t in $\mathbf{u}^{(m)}$ is omitted for discussion convenience. The reconstruction requires integral conservation for $\mathbf{u}^{(m)}$ in each subelement $T^{(m(k))}$, i.e.,

$$\int_{T^{(m(k))}} \mathbf{u}^{(m)}\left(\mathbf{x}\left(T^{(m)}, \boldsymbol{\xi}\right)\right) dV = |T^{(m(k))}| \overline{\mathbf{u}}^{(m(k))}, \tag{12}$$

$$\forall T^{(m(k))} \subset T^{(m)}, \quad k = 1, \cdots, N_e.$$

To solve the reconstruction problem, inspired by the DG method [15–17, 23, 28, 29], we use hierarchical orthogonal basis functions. The basis functions $\phi_l(\xi, \eta, \zeta)$ of a complete polynomial of degree N ($N = 1, 2, 3, 4$) in the reference coordinate system can be found in [27]. We remark that the basis functions are orthonormal and satisfy the following property:

$$\int_{T_E} \phi_l(\xi, \eta, \zeta) d\xi d\eta d\zeta = \begin{cases} \dfrac{\sqrt{6}}{6}, & l = 1, \\[2mm] 0, & l \neq 1. \end{cases} \tag{13}$$

Transforming equation (12) in the physical coordinate system $x - y - z$ into the reference coordinate system $\xi - \eta - \zeta$ and noticing Eq. (8), we obtain

$$\sum_{l=1}^{N_p} \left(\int_{\tilde{T}^{(m(k))}} \phi_l(\xi, \eta, \zeta) d\xi d\eta d\zeta \right) \hat{\mathbf{w}}_l^{(m)} = |\tilde{T}^{(m(k))}| \overline{\mathbf{u}}^{(m(k))}, \tag{14}$$

$$\forall \tilde{T}(m(k)) \subset \tilde{T}(m) = T_E, \quad k = 1, \cdots, N_e,$$

where $\tilde{T}(m)$ is in fact the reference element T_E and $\tilde{T}(m(k))$ is the transformed element corresponding to the subelement $T^{(m(k))}$.

The integration in Eq. (14) over $\tilde{T}(m(k))$ in $\boldsymbol{\xi}$ system can be computed efficiently if it is performed over its reference element in a second reference system $\tilde{\boldsymbol{\xi}}$. Denote the transformation from $\tilde{\boldsymbol{\xi}}$ to $\boldsymbol{\xi}$ and its inverse by $\boldsymbol{\xi} = \boldsymbol{\xi}\left(\tilde{T}(m(k)), \tilde{\boldsymbol{\xi}}\right)$ and $\tilde{\boldsymbol{\xi}} = \tilde{\boldsymbol{\xi}}\left(\tilde{T}(m(k)), \boldsymbol{\xi}\right)$, respectively. Transforming Eq. (14) into $\tilde{\boldsymbol{\xi}}$ system and rewriting the result as a compact form, we have

$$G\hat{\mathbf{w}} = \overline{\mathbf{u}}, \tag{15}$$

where G is the $N_e \times N_p$ matrix with entries G_{kl} given by

$$G_{kl} = \frac{1}{|T_E|} \left(\int_{T_E} \phi_l\left(\boldsymbol{\xi}\left(\tilde{T}^{(m(k))}, \tilde{\boldsymbol{\xi}}\right)\right) d\tilde{\xi} d\tilde{\eta} d\tilde{\zeta} \right), \qquad k = 1, \cdots, N_e; \ l = 1, \cdots, N_p, \tag{16}$$

and

$$\overline{\mathbf{u}} := \left(\overline{\mathbf{u}}^{(m(1))}, , \overline{\mathbf{u}}^{(m(2))}, \cdots, \overline{\mathbf{u}}^{(m(N_e))} \right)^T, \qquad \hat{\mathbf{w}} := \left(\hat{\mathbf{w}}_1^{(m)}, \hat{\mathbf{w}}_2^{(m)}, \cdots, \hat{\mathbf{w}}_{N_p}^{(m)} \right)^T. \tag{17}$$

We need at least N_p subelements in the stencil since the reconstructed number of degrees of freedom is N_p. As listed in **Table 1**, N_e subelements are used to form the stencil. Note that N_e is definitely larger than N_p, which is helpful to improve the reconstruction robustness [20, 21]. Thus Eq. (15) is an overdetermined problem. We use the constrained least squared technique to solve it.

From the orthogonality of basis functions and the property of Eq. (13), we remark that Eq. (15) is subject to the following constraint condition [27]:

$$\sqrt{6}\hat{\mathbf{w}}_1^{(m)} = \sum_{k=1}^{N_e} \frac{\overline{\mathbf{u}}^{(m(k))}}{N_e}. \tag{18}$$

With the constraint, Eq. (15) is solved by the Lagrange multiplier method [19, 20, 27]. And the system can be written as

$$\begin{pmatrix} 2G^T G & -R^T \\ R & 0 \end{pmatrix} \begin{pmatrix} \hat{\mathbf{w}} \\ \lambda_p \end{pmatrix} = \begin{pmatrix} 2G^T \overline{\mathbf{u}} \\ \tilde{R}\overline{\mathbf{u}} \end{pmatrix}, \tag{19}$$

where λ_p is the Lagrangian multiplier and both R and $\tilde{R}$ are $1 \times N_e$ matrices:

$$R = \left(\sqrt{6}, 0, \cdots, 0\right), \qquad \tilde{R} = \left(\frac{1}{N_e}, \cdots, \frac{1}{N_e}\right). \tag{20}$$

The coefficient matrix on the left-hand side of Eq. (19) is the so-called reconstruction matrix [19, 20].

2.4 The spatial discrete formulation

We now derive the semi-discrete finite volume scheme based on Eqs. (2) and (8). Integrating over each subelement $T^{(m(k))}$ on both sides of Eq. (2), we have

$$\int_{T^{(m(k))}} \frac{\partial \mathbf{u}}{\partial t} dV + \int_{T^{(m(k))}} \left(A\frac{\partial \mathbf{u}}{\partial x} + B\frac{\partial \mathbf{u}}{\partial y} + C\frac{\partial \mathbf{u}}{\partial z}\right) dV = 0, \qquad k = 1, \cdots, N_e. \tag{21}$$

Using Eq. (8) and integration by parts yield

$$\int_{T^{(m(k))}} \frac{\partial \mathbf{u}}{\partial t} dV + \int_{\partial T^{(m(k))}} \mathbf{F}^h dS = 0, \tag{22}$$

where dS denotes the infinitesimal element in the face integral and $\mathbf{F}^h$ is the numerical flux, and we adopt the widely used Godunov flux [15, 19, 20, 23]

$$\mathbf{F}^h = \frac{1}{2}T\left(A^{(m(k))} + |A^{(m(k))}|\right)T^{-1}\sum_{l=1}^{N_p}\hat{\mathbf{w}}_l^{(m)}\phi_l^{(m)}$$

$$+ \frac{1}{2}T\left(A^{(m(k))} - |A^{(m(k))}|\right)T^{-1}\sum_{l=1}^{N_p}\hat{\mathbf{w}}_l^{(m_j)}\phi_l^{(m_j)}, \tag{23}$$

where m_j is the index number of coarse tetrahedral element neighboring subelement $T^{(m(k))}$. The notation $|A^{(m(k))}|$ denotes applying the absolute value operator of the eigenvalues given in Eq. (3), i.e.,

$$|A^{(m(k))}| = R|\Lambda|R^{-1}, \qquad |\Lambda| = \text{diag}(|s_1|, \cdots, |s_9|), \tag{24}$$

where R is the matrix and its columns are made up of the eigenvectors associated with eigenvalues in Eq. (3), i.e.,

$$R = \begin{pmatrix} \lambda + 2\mu & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda + 2\mu \\ \lambda & 0 & 0 & 0 & 1 & 0 & 0 & 0 & \lambda \\ \lambda & 0 & 0 & 0 & 0 & 1 & 0 & 0 & \lambda \\ 0 & \mu & 0 & 0 & 0 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \mu & 0 & 0 & 0 & \mu & 0 & 0 \\ v_p & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -v_p \\ 0 & v_s & 0 & 0 & 0 & 0 & 0 & -v_s & 0 \\ 0 & 0 & v_s & 0 & 0 & 0 & -v_s & 0 & 0 \end{pmatrix}. \tag{25}$$

And T is the rotation matrix given by

$$T = \begin{pmatrix} n_x^2 & s_x^2 & t_x^2 & 2n_x s_x & 2s_x t_x & 2n_x t_x \\ n_y^2 & s_y^2 & t_y^2 & 2n_y s_y & 2s_y t_y & 2n_y t_y \\ n_z^2 & s_z^2 & t_z^2 & 2n_z s_z & 2s_z t_z & 2n_z t_z \\ n_y n_x & s_y s_x & t_y t_x & n_y s_x + n_x s_y & s_y t_x + s_x t_y & n_y t_x + n_x t_y \\ n_z n_y & s_z s_y & t_z t_y & n_z s_y + n_y s_z & s_z t_y + s_y t_z & n_z t_y + n_y t_z \\ n_z n_x & s_z s_x & t_z t_x & n_z s_x + n_x s_z & s_z t_x + s_x t_z & n_z t_x + n_x t_z \end{pmatrix}, \tag{26}$$

where (n_x, n_y, n_z) is the normal vector of the face and (s_x, s_y, s_z) and (t_x, t_y, t_z) are the two tangential vectors. T^{-1} denotes the inverse of T.

Inserting Eqs. (23) into (22) and rewriting the result into a splitting form of easy computation in the reference system ξ, we have

$$\frac{\partial}{\partial t} \overline{\mathbf{u}}^{(m(k))} |T^{(m(k))}| + \sum_{j=1}^{4} \mathbf{F}_j^h = 0 \tag{27}$$

with

$$\mathbf{F}_j^h = T^j A^{(m(k))} \left(T^j\right)^{-1} |S_j| \sum_{l=1}^{N_p} F_l^{-,j} \hat{\mathbf{w}}_l^{(m)}, \qquad m = m_j, \tag{28}$$

and

$$\mathbf{F}_j^h = \frac{1}{2} T^j \left(A^{(m(k))} + |A^{(m(k))}| \right) \left(T^j\right)^{-1} |S_j| \sum_{l=1}^{N_p} F_l^{-,j} \hat{\mathbf{w}}^{(m)}$$
$$+ \frac{1}{2} T^j \left(A^{(m(k))} - |A^{(m(k))}| \right) \left(T^j\right)^{-1} |S_j| \sum_{l=1}^{N_p} F_l^{+,i,p} \hat{\mathbf{w}}_l^{(m_j)}, \qquad m \neq m_j, \tag{29}$$

where S_j is the area of the j-th ($j = 1, 2, 3, 4$) face of subelement $T^{(m(k))}$. $F_l^{-,j}$ and $F_l^{+,i,p}$ are the left flux matrix and the right state flux matrix, respectively, which are given by

$$F_l^{-,j} = \int_{\partial(T_E)_j} \phi_l\left(\xi\left(\tilde{T}(m(j)), \tilde{\xi}(j)(\chi, \tau)\right)\right) d\chi d\tau, \quad j = 1, 2, 3, 4, \tag{30}$$

$$F_l^{+,i,p} = \int_{\partial(T_E)_j} \phi_l\left(\boldsymbol{\xi}\left(\tilde{T}(m(i)), \tilde{\boldsymbol{\xi}}(i)\left(\tilde{\chi}^{(p)}, \tilde{\tau}^{(p)}\right)\right)\right) d\chi d\tau, \quad i=1,2,3,4; \quad p=1,2,3.$$

$$(31)$$

where χ and τ are the face parameters. The transformation of the face parameters χ and τ to the face parameters $\tilde{\chi}$ and $\tilde{\tau}$ in the neighbor tetrahedron depends on the orientation of the neighbor face with respect to the local face of the considered tetrahedron. And the mapping is given in **Table 2**. For a given tetrahedral mesh with the known indices i and p, there are only 4 of 12 possible matrices $F^{+,i,p}$ per element [15, 20]. Comparing with the traditional FV method, the method with the splitting form described above has much less computations of face integrations. Note that only our proposed FV method can be written as a splitting form. Theoretical analysis shows our method can save about half computational time under the condition of the same number of elements [27].

2.5 The time discretization

Equation (27) is in fact a semi-discrete ordinary differential equation (ODE) system. In order to solve it formally, we denote the spatial semi-discrete part in Eq. (27) by a linear operator L. Then Eq. (27) can be written as a concise ODE form:

$$\frac{d\mathbf{u}}{dt} = L(\mathbf{u}, t).$$

$$(32)$$

Traditionally, the classic fourth-order explicit RK (ERK) method

$$
\begin{aligned}
\mathbf{k}^{(1)} &= L(\mathbf{u}^n, , t^n), \\
\mathbf{k}^{(2)} &= L\left(\mathbf{u}^n + \frac{1}{2}\Delta t \mathbf{k}^{(1)}, t^n + \frac{1}{2}\Delta t\right), \\
\mathbf{k}^{(3)} &= L\left(\mathbf{u}^n + \frac{1}{2}\Delta t \mathbf{k}^{(2)}, t^n + \frac{1}{2}\Delta t\right), \\
\mathbf{k}^{(4)} &= L\left(\mathbf{u}^n + \Delta t \mathbf{k}^{(3)}, t^n + \Delta t\right), \\
\mathbf{u}^{n+1} &= \mathbf{u}^n + \frac{1}{6}\Delta t\left(\mathbf{k}^{(1)} + 2\mathbf{k}^{(2)} + 2\mathbf{k}^{(3)} + \mathbf{k}^{(4)}\right)
\end{aligned}
$$

$$(33)$$

can be applied to advance $\mathbf{u}$ from $\mathbf{u}^n$ to $\mathbf{u}^{n+1}$. Here Δt is the time step. Now we use the low-storage version of ERK (LSERK) to solve Eq. (32):

$$
\begin{aligned}
\mathbf{u}^{(0)} &= \mathbf{u}^n, \\
&\begin{cases}
\mathbf{k}^{(i)} = a_i \mathbf{k}^{(i-1)} + \Delta t L\left(\mathbf{p}^{(i-1)}, t^n + c_i \Delta t\right), \\
\mathbf{p}^{(i)} = \mathbf{p}^{(i-1)} + b_i \mathbf{k}^{(i)}, \qquad i=1,\cdots,5,
\end{cases} \\
\mathbf{u}^{(n+1)} &= \mathbf{p}^{(5)}.
\end{aligned}
$$

$$(34)$$

p	1	2	3
$\tilde{\chi}$	τ	$1-\chi-\tau$	χ
$\tilde{\tau}$	χ	τ	$1-\chi-\tau$

Table 2.
Transformation of the face parameters χ and τ to the face parameters $\tilde{\chi}$ and $\tilde{\tau}$.

As we can see the LSERK only requires one additional storage level, while ERK has four. The coefficients required in Eq. (34) are listed in **Table 3** [30].

As to the stability condition, it is controlled by the Courant-Friedrichs-Lewy (CFL) condition [15, 19];

$$\Delta t \leq \frac{1}{2N+1} \frac{h_{\min}}{v_p}, \tag{35}$$

where v_p is the P wave velocity and $h_{\min}$ is the minimum diameter of the circumcircles of tetrahedral elements. This condition is a necessary condition for discrete stability, and a bit more restrictive form is actually used in numerical computations.

The absorbing boundary conditions (ABCs) in computations are required as the computational domain is finite. There are two typical ABCs to be adopted here. One is flux type ABCs [16, 19]. That is to say, the following numerical flux in Eq. (23) at all tetrahedral faces that coincide with domain boundary

$$\mathbf{F}^h = \frac{1}{2} T\left(A^{(m(k))} + |A^{(m(k))}|\right) T^{-1} \sum_{l=1}^{N_p} \hat{\mathbf{w}}_l^{(m)} \phi_l^{(m)}, \tag{36}$$

which allows only for outgoing waves and is equivalent to the first order ABCs. Though the absorbing effects of this method vary the angles of incidence, it is still effective in many cases [19]. The advantage of this type ABCs is that it merged into the FVM framework naturally and there is almost no additional computational cost. Another type is the perfectly matched layer (PML) technique originally developed by [31], which is very popular in recent more 10 years.

2.6 Coordinate transformation

The transformation between different coordinate systems is frequently used. For ease of reading, we present the formulations here. Let (x_i, y_i, z_i) for $i = 1, 2, 3, 4$ be the coordinates of a physical element. The transformation from $\xi - \eta - \zeta$ system to $x - y - z$ system is defined by

$$\begin{cases} x = x_1 + (x_2 - x_1)\xi + (x_3 - x_1)\eta + (x_4 - x_1)\zeta, \\ y = y_1 + (y_2 - y_1)\xi + (y_3 - y_1)\eta + (y_4 - y_1)\zeta, \\ z = z_1 + (z_2 - z_1)\xi + (z_3 - z_1)\eta + (z_4 - z_1)\zeta, \end{cases} \tag{37}$$

then the transformation from $x - y - z$ system to $\xi - \eta - \zeta$ system can be solved for ξ, η and ζ from Eq. (37) by the Cramer ruler, i.e.,

i	a_i	b_i	c_i
1	0	0.1496590219992291	0
2	−0.4178904744998519	0.3792103129996273	0.1496590219992291
3	−1.1921516946426769	0.8229550293869817	0.3704009573642048
4	−1.6977846924715279	0.6994504559491221	0.6222557631344432
5	−1.514834442571558	0.1530572479681520	0.9582821306746903

Table 3.
Coefficients for the low-storage five-stage fourth-order ERK method.

$$\xi = \frac{|J_1|}{|J|}, \qquad \eta = \frac{|J_2|}{|J|}, \qquad \zeta = \frac{|J_3|}{|J|}, \tag{38}$$

where

$$|J_1| = \begin{vmatrix} x - x_1 & x_3 - x_1 & x_4 - x_1 \\ y - y_1 & y_3 - y_1 & y_4 - y_1 \\ z - z_1 & z_3 - z_1 & z_4 - z_1 \end{vmatrix}, \qquad |J_2| = \begin{vmatrix} x_2 - x_1 & x - x_1 & x_4 - x_1 \\ y_2 - y_1 & y - y_1 & y_4 - y_1 \\ z_2 - z_1 & z - z_1 & z_4 - z_1 \end{vmatrix}, \tag{39}$$

$$|J_3| = \begin{vmatrix} x_2 - x_1 & x_3 - x_1 & x - x_1 \\ y_2 - y_1 & y_3 - y_1 & y - y_1 \\ z_2 - z_1 & z_3 - z_1 & z - z_1 \end{vmatrix}, \qquad |J| = \begin{vmatrix} x_2 - x_1 & x_3 - x_1 & x_4 - x_1 \\ y_2 - y_1 & y_3 - y_1 & y_4 - y_1 \\ z_2 - z_1 & z_3 - z_1 & z_4 - z_1 \end{vmatrix}. \tag{40}$$

Note that J is the determinant of the Jacobian matrix of the transformation being equal to six times the volume of the tetrahedron element $T^{(m)}$.

The coordinate transformation from the second reference coordinate $\tilde{\xi} - \tilde{\eta} - \tilde{\zeta}$ to $\xi - \eta - \zeta$ system is defined by

$$\begin{cases} \xi = \xi_1 + (\xi_2 - \xi_1)\tilde{\xi} + (\xi_3 - \xi_1)\tilde{\eta} + (\xi_4 - \xi_1)\tilde{\zeta}, \\ \eta = \eta_1 + (\eta_2 - \eta_1)\tilde{\xi} + (\eta_3 - \eta_1)\tilde{\eta} + (\eta_4 - \eta_1)\tilde{\zeta}, \\ \zeta = \zeta_1 + (\zeta_2 - \zeta_1)\tilde{\xi} + (\zeta_3 - \zeta_1)\tilde{\eta} + (\zeta_4 - \zeta_1)\tilde{\zeta}, . \end{cases} \tag{41}$$

then the transform from $\xi - \eta - \zeta$ system to $\tilde{\xi} - \tilde{\eta} - \tilde{\zeta}$ system can be solved for $\tilde{\xi} - \tilde{\eta} - \tilde{\zeta}$ from Eq. (41) by Cramer ruler similarly. Denote

$$|\tilde{J}| = \begin{vmatrix} \xi_2 - \xi_1 & \xi_3 - \xi_1 & \xi_4 - \xi_1 \\ \eta_2 - \eta_1 & \eta_3 - \eta_1 & \eta_4 - \eta_1 \\ \zeta_2 - \zeta_1 & \zeta_3 - \zeta_1 & \zeta_4 - \zeta_1 \end{vmatrix}, \tag{42}$$

which is the determinant of the Jacobian matrix of the transformation being equal to six times the volume of the subelement $\tilde{T}(m(k))$ for $k = 1, \cdots, N_e$. In Eqs. (41) and (42), (ξ_i, η_i, ζ_i) for $i = 1, 2, 3, 4$, denote the vertex coordinates of $\tilde{T}(m(k))$ in $\xi - \eta - \zeta$ system.

3. Numerical computations

In this section we give three numerical examples to illustrate the performance of the developed method above. The convergence test of the proposed method can be found in [27]. Though the method is developed for the 3D case, it can be simplified to 2D without essential difficulty. The principle is the same. The first example is a test for a 2D model with uneven topography. The other two examples are for two 3D models.

Example 1. The first example is a two-layered model with the inclined interface shown in **Figure 3a**. The range of the model is $x \in [-1.6km, 1.6km]$ and $z \in [-1.6km, 1.8km]$. The surface of the model is uneven to imitate the real topography. The v_p and v_s velocities are 3000 m/s and 2000 m/s in the upper layer and 2400 m/s and 1600 m/s in the lower layer, respectively. The densities ρ are 2200 kg/m^3 and 1800 kg/m^3 in the upper and lower layer, respectively. **Figure 3b** is the coarser triangular meshes for this model. A coarser version of the mesh is
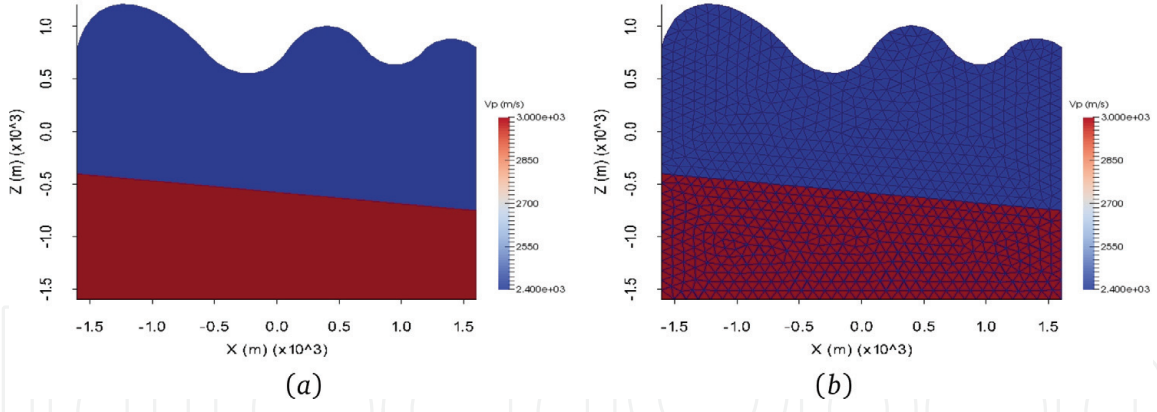
Figure 3.
A two-layered model with curved surface topography (a) and the triangular meshes (b).

shown here as the finest mesh in computations cannot be seen clearly. The triangular meshes can fit the curve topography very well. Note that none triangular element crosses the interface. In computations the P_4 polynomial reconstruction is applied. The computational domain is meshed by 113472 coarse elements. Each coarse element is subdivided into 25 subelements further. So there are 2,836,800 fine elements totally. The time step is $\Delta t = 5 \times 10^{-5}$ s. The source is located at $(x, z) = (0, 0.2km)$ with time history

$$f(t) = -2\alpha(t - t_0)e^{-\alpha(t-t_0)^2}, \qquad t_0 = 0.08, \qquad \alpha = \left(\pi f_0\right)^2, \tag{43}$$

where $f_0 = 20$ Hz is the main frequency. In order to simulate point source excitation, a spatial local distribution function defined by

$$G(\mathbf{x}) = \begin{cases} \exp\left(-7\|\mathbf{x} - \mathbf{x}_0\|_2^2 / r_0^2\right), & \|\mathbf{x} - \mathbf{x}_0\|_2^2 \leqslant r_0^2, \\ 0, & \|\mathbf{x} - \mathbf{x}_0\|_2^2 > r_0^2, \end{cases} \tag{44}$$

is applied, where $\mathbf{x}_0 = \left(x_0, y_0, z_0\right)$ are positions of the source center. The source is added to the u component; that is to say, all source terms except g_7 in Eq. (1) are all zero. **Figure 4** is the snapshots of u and v components at propagation time 0.25 s. **Figure 5** is the snapshots of u and v components at propagation time 0.30 s. We can see the P wave and S wave propagate toward out of the model. The reflected and transmitted waves due to the tilted physical interface are also very clear. These are the expected physical phenomena of wave propagation in elastic media.
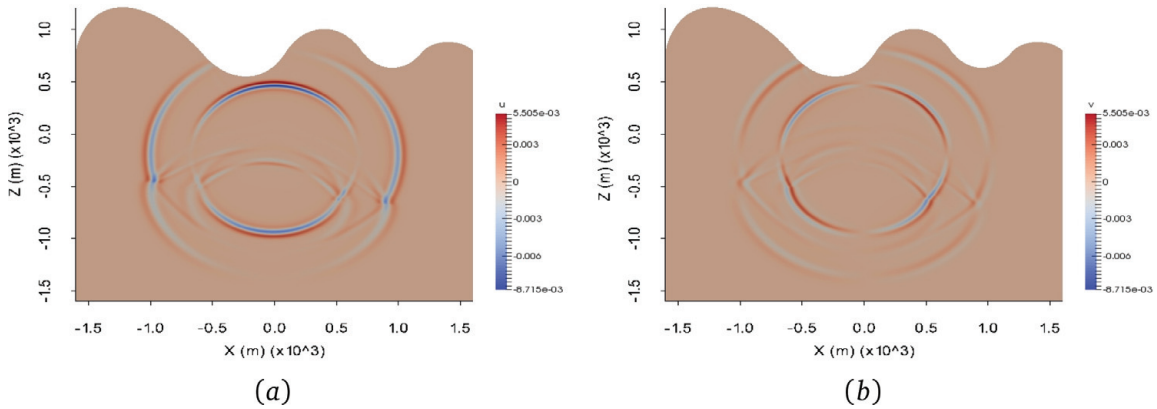

Figure 4.
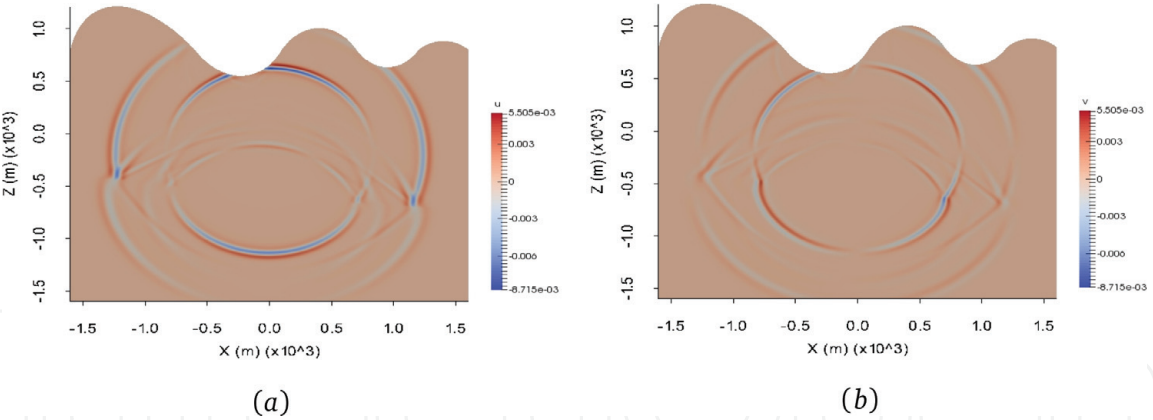*Snapshots of u component (a) and v component (b) at propagation time 0.25s.*

Figure 5.
Snapshots of u component (a) and v component (b) at propagation time 0.30s.

Example 2. The second example is a cuboid model. The physical size of the model is $(x, y, z) \in [0, 2km] \times [0, 2km] \times [0, 1km]$. The model and its unstructured tetrahedral meshes are shown in **Figure 6**. There are totally 836,612 coarse tetrahedrons to mesh the model. A coarser mesh is shown as the actual mesh in computations is too fine to see clearly. Each coarse tetrahedron is subdivided into $N_e = 27$ subelements as we adopt P_3 polynomial reconstruction. The parameters for λ, μ, and ρ are 10^9 Pa, 10^9 Pa, and 1000 kg/m^3. The time step in computations is 10^{-4} s. The source is located in the center of the model with time history given by

$$f(t) = \sin{(40\pi t)}e^{-100t^2}. \tag{45}$$

It is applied to the u component. The 3D snapshots of u, v, and w components at propagation time 0.42 s are shown in **Figure 7**. From these figures, we can clearly see two types of waves, i.e., the compressive wave and the shear wave. The splitting PML in nonconvolutional form is adopted here [32], and the boundary reflections are absorbed obviously and effectively. The message passing interface (MPI) parallelization based on spatial domain decomposition is applied. The CPU time for extrapolation 1000 time steps is about 33, 310 s with 128 processors each with 2.6 GHz main frequency.
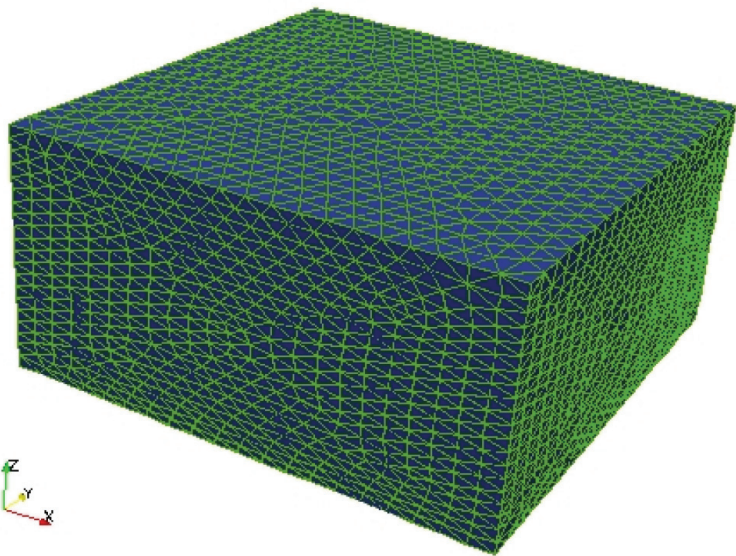

Figure 6.
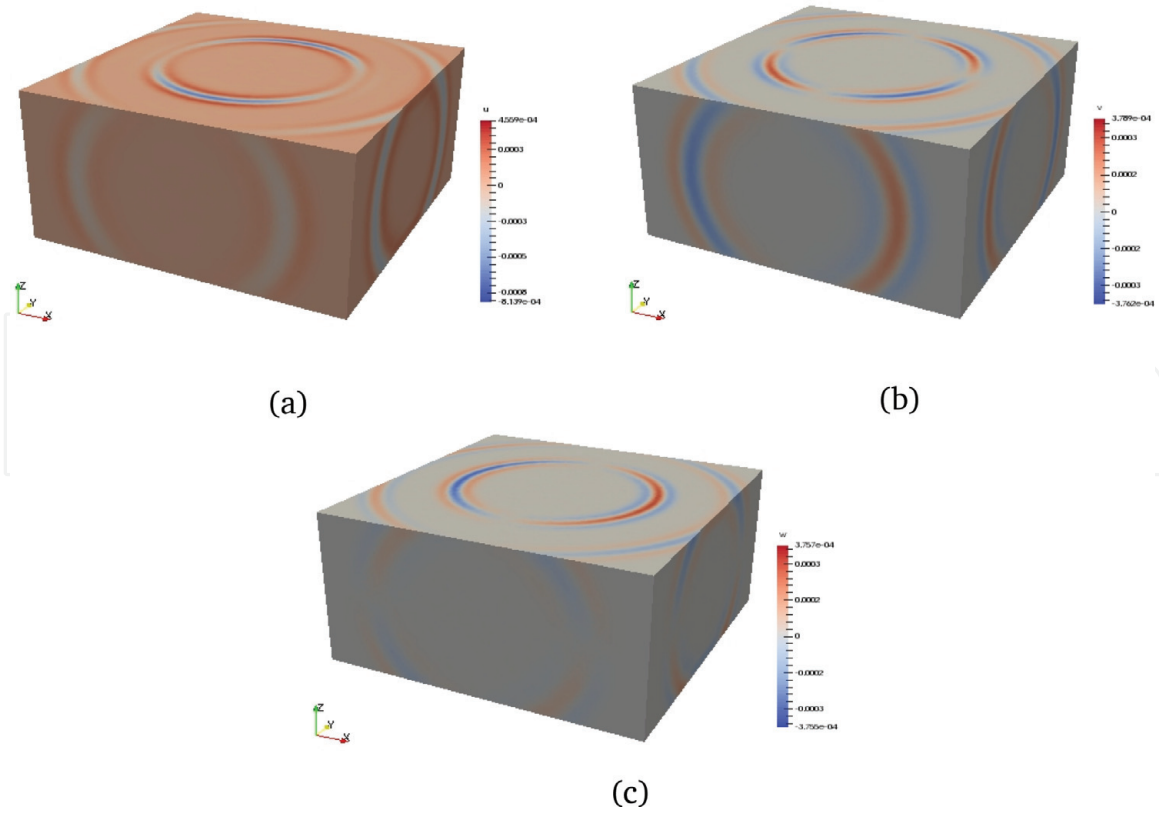*A cubic model and its unstructured tetrahedral meshes.*

Figure 7.
The 3D snapshots of u component (a), v component (b), and w component (c) at propagation time 0.42 s in a cuboid model. The source is located in the center of the model.

Example 3. The third example is a real geological model in China. As shown in **Figure 8a**, it has a very complex topography. The physical scope of the model is $x \in [0, 2.0km]$, $y \in [0, 3.5km]$, and $z \in [0, 1.1km]$. The corresponding 3D mesh is shown in **Figure 8b**. A coarser version of the mesh is given as the actual mesh in computations is too fine to see clearly in the figure. The model is meshed with 210,701 relative coarse tetrahedral elements. Each coarse tetrahedron is subdivided into $N_e = 64$ subelements as we adopt P_4 polynomial reconstruction, and thus there are 13,484,864 fine elements totally. The time step Δt is 10^{-4} s. The source is situated at $(x_0, y_0, z_0) = (750m, 1300m, 300m)$ with the same time history in Eq. (45). The media velocities of v_p and v_s are $v_p = 3000$ m/s and $v_s = 2000$ m/s. The MPI parallelization based on spatial domain decomposition is applied. The
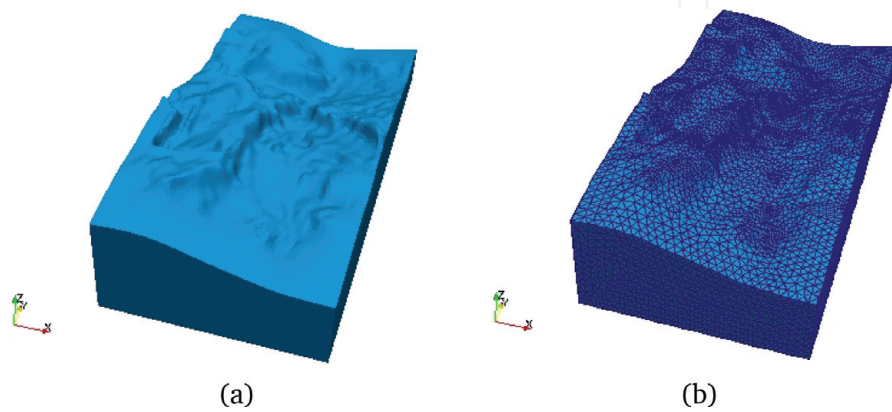

Figure 8.
A real 3D model with complex topography. (a) model and (b) unstructured tetrahedral meshes.
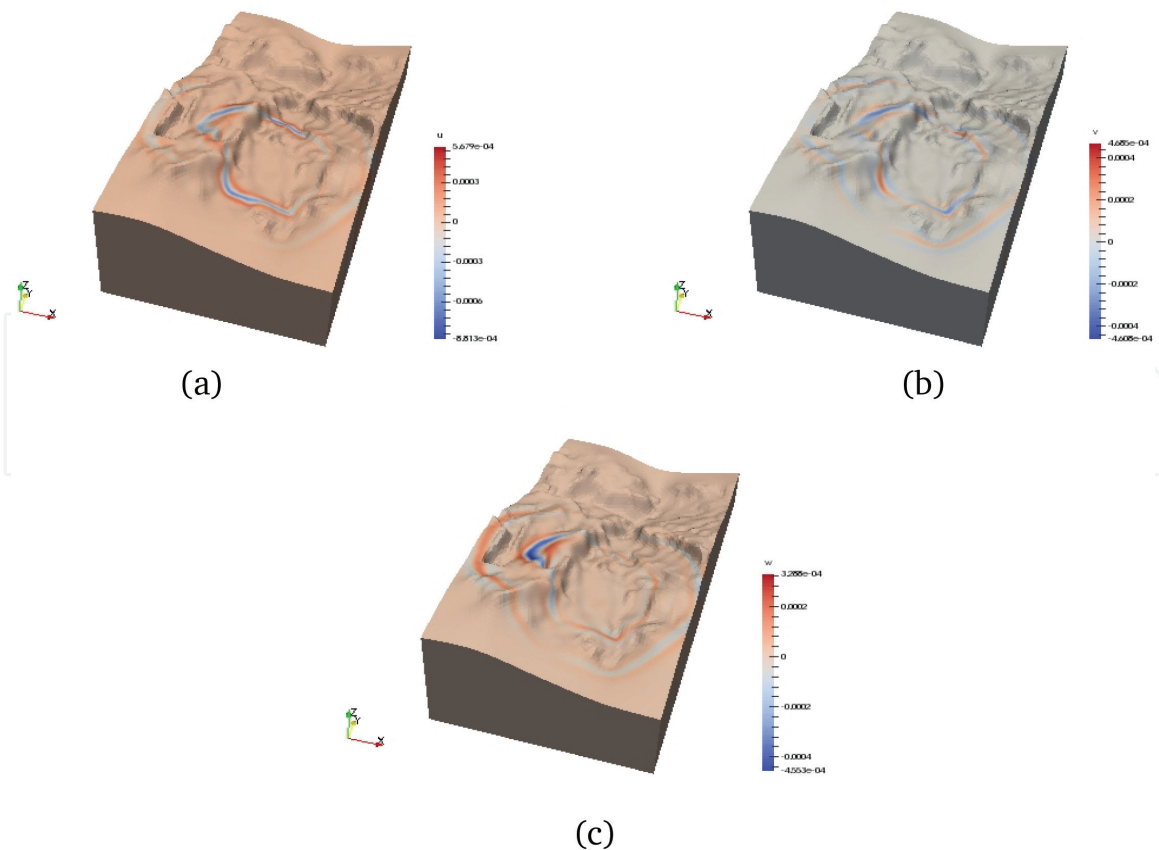
Figure 9.
3D snapshots of u, v, and w components at propagation time 0.80 s in a real 3D model. The results are obtained by the method in this chapter with P_4 reconstruction. (a) u component, (b) v component, (c) w component.

nonconvolutional splitting PML [32] is adopted. The 3D snapshots of u, v, and w components at propagation time 0.80 s are shown in **Figure 9**. The CPU time for extrapolation 10,000 time steps is 100, 449 s with 256 processors each with 2.6 GHz main frequency. From **Figure 9**, we can see clearly the propagation of P wave and S wave.

4. Conclusions

A new efficient high-order finite volume method for the 3D elastic wave simulation on unstructured meshes has been developed. It combines the advantages of the DG method and the traditional FV method. It adapts irregular topography very well. The reconstruction stencil is generated by refining each coarse tetrahedron which can be implemented effectively for all tetrahedrons whether they are internal or boundary elements. The hierarchical orthogonal basis functions are exploited to perform the high-order polynomial reconstruction on the stencil. The resulting reconstruction matrix remains unchanged for all tetrahedrons and can be precomputed and stored before time evolution. The method preserves a very local property like the DG method, while it has high computational efficiency like the FV method. These advantages facilitate 3D large-scale parallel computations. Numerical computations including a 3D real physical model show its good performance. The method also can be expected to solve other linear hyperbolic equations without essential difficulty.

Acknowledgements

Author details

Wensheng Zhang

1 LSEC, ICMSEC, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, China

2 School of Mathematics Sciences, University of Chinese Academy Sciences, Beijing, China

*Address all correspondence to: zws@lsec.cc.ac.cn

IntechOpen

References

[1] Minkoff SE. Spatial parallelism of a 3D finite difference velocity-stress elastic wave propagation code. SIAM Journal on Scientific Computing. 2002;**24**:1-19. DOI: 10.1137/S1064827501390960

[2] Virieux J. P-SV wave propagation in heterogeneous media: Velocity-stress finite-difference method. Geophysics. 1986;**51**:889-901. DOI: 10.1190/1.1442147

[3] Klin P, Priolo E, Seriani G. Numerical simulation of seismic wave propagation in realistic 3-D geo-models with a Fourier pseudo spectral method. Geophysical Journal International. 2010;**183**:905-922. DOI: 10.1111/j.1365-246X.2010.04763.x

[4] Zhang W. Stability conditions for wave simulation in 3-D anisotropic media with the pseudospectral method. Communications in Computational Physics. 2012;**12**:703-720. DOI: 10.4208/cicp.120610.090911a

[5] Bécache E, Joly P, Tsogka C. A new family of mixed finite elements for the linear elastodynamic problem. SIAM Journal on Numerical Analysis. 2002;**39**:2109-2132. DOI: 10.2307/4101053

[6] Cohen G, Joly P, Roberts JE, Tordjman N. Higher order triangular finite elements with mass lumping for the wave equations. SIAM Journal on Numerical Analysis. 2001;**38**:2047-2078. DOI: 10.1137/s0036142997329554

[7] Cohen G, Fauqueux S. Mixed finite elements with mass-lumping for the transient wave equation. Journal of Computational Acoustics. 2000;**8**:171-188. DOI: 10.1142/s0218396x0000011x

[8] Cohen G, Fauqueux S. Mixed spectral finite elements for the linear elasticity system in unbounded domains. SIAM Journal on Scientific Computing. 2005;**26**:864-884. DOI: 10.1137/S1064827502407457

[9] Zhang W, Chung E, Wang C. Stability for imposing absorbing boundary conditions in the finite element simulation of acoustic wave propagation. Journal of Computational Mathematics. 2014;**32**:1-20. DOI: 10.4208/jcm.1310-m3942

[10] Dubiner M. Spectral methods on triangles and other domains. Journal of Scientific Computing. 1991;**6**:345-390. DOI: 10.1007/BF01060030

[11] Komatitsch D, Tromp J. Introduction to the spectral element method for three-dimensional seismic wave propagation. Geophysical Journal International. 1999;**139**:806-822. DOI: 10.1046/j.1365-246x.1999.00967.x

[12] Komatitsch D, Martin R, Tromp J, Taylor MA, Wingate BA. Wave propagation in 2-D elastic media using a spectral element method with triangles and quadrangles. Journal of Computational Acoustics. 2001;**9**:703-718. DOI: 10.1142/S0218396X01000796

[13] Komatitsch D, Tromp J. Spectral-element simulations of global seismic wave propagation—I. Validation. Geophysical Journal International. 2002;**149**:390-412. DOI: 10.1046/j.1365-246X.2002.01653.x

[14] Seriani G. 3-D large-scale wave propagation modeling by spectral-element method on Cray T3E multiprocessor. Computer Methods in Applied Mechanics and Engineering. 1998;**164**:235-247. DOI: 10.1016/S0045-7825(98)00057-7

[15] Dumbser M, Käser M. An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes-II: The three-dimensional isotropic case.

Geophysical Journal International. 2006;**167**:319-336. DOI: 10.1111/j.1365-246X.2006.03120.x

[16] Käser M, Dumbser M. An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes. I: The two-dimensional isotropic case with external source terms. Geophysical Journal International. 2006;**166**:855-877. DOI: 10.1111/j.1365-246X.2006.03051.x

[17] Käser M, Dumbser M, Puente J, Igel H. An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes—III: Viscoelastic attenuation. Geophysical Journal International. 2007;**168**:224-242. DOI: 10.1111/j.1365-246X.2006.03193.x

[18] Ye R, de Hoop MV, Petrovitch CL, Pyrak-Nolte LJ, Wilcox LC. A discontinuous Galerkin method with a modified penalty flux for the propagation and scattering of acousto-elastic waves. Geophysical Journal International. 2016;**205**:1267-1289. DOI: 10.1093/gji/ggw070

[19] Dumbser M, Käser M, de la Puente J. Arbitrary high-order finite volume schemes for seismic wave propagation on unstructured meshes in 2D and 3D. Geophysical Journal International. 2007;**171**:665-694. DOI: 10.1111/j.1365-246X.2007.03421.x

[20] Dumbser M, Käser M. Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems. Journal of Computational Physics. 2007;**221**:693-723. DOI: 10.1016/j.jcp.2006.06.043

[21] Käser M, Iske A. ADER schemes on adaptive triangular meshes for scalar conservation laws. Journal of Computational Physics. 2005;**205**:486-508. DOI: 10.1016/j.jcp.2004.11.015

[22] Zhang W, Zhuang Y, Chung ET. A new spectral finite volume method for

elastic modelling on unstructured meshes. Geophysical Journal International. 2016;**206**:292-307. DOI: 10.1093/gji/ggw148

[23] Dumbser M, Käser M, Toro EF. An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes—V: Local time stepping and p-adaptivity. Geophysical Journal International. 2007;**171**:695-717. DOI: 10.1111/j.1365-246X.2007.03427.x

[24] Liu Y, Vinokur M, Wang ZJ. Spectral (finite) volume method for conservation laws on unstructured grids. V: Extension to three-dimensional systems. Journal of Computational Physics. 2006;**212**:454-472. DOI: 10.1016/j.jcp.2003.09.012

[25] Wang ZJ. Spectral (finite) volume method for conservation laws on unstructured grids: Basic formulation. Journal of Computational Physics. 2002;**178**:210-251. DOI: 10.1006/jcph.2002.7041

[26] Wang ZJ, Liu Y. Spectral (finite) volume method for conservation laws on unstructured grids. II: Extension to two-dimensional scalar equation. Journal of Computational Physics. 2002;**179**:665-697. DOI: 10.1006/jcph.2002.7082

[27] Zhang W, Zhuang Y, Zhang L. A new high-order finite volume method for 3D elastic wave simulation on unstructured meshes. Journal of Computational Physics. 2017;**340**:534-555. DOI: 10.1016/j.jcp.2017.03.050

[28] Puente J, Käser M, Dumbser M, Igel H. An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes-IV: Anisotropy. Geophysical Journal International. 2007;**169**:1210-1228. DOI: 10.1111/j.1365-246X.2007.03381.x

[29] Puente J, Dumbser M, Käser M, Igel H. Discontinuous Galerkin method for

propagation in poroelastic media. Geophysics. 2008;**73**:T77-T97. DOI: 10.1190/1.2965027

[30] Hesthaven JS, Warburton T. Nodal Discontinuous Galerkin Methods. New York: Springer-Verlag; 2008. 502 p. DOI: 10.1007/978-0-387-72067-8

[31] Bérenger JP. A perfectly matched layer for the absorption of electromagnetic waves. Journal of Computational Physics. 1994;**114**: 185-200. DOI: 10.1006/jcph.1996.0181

[32] Collino F, Tsogka C. Application of the perfectly matched absorbing layer model to the linear elastodynamic problem in anisotropic heterogeneous media. Geophysics. 2001;**66**:294-307. DOI: 10.1190/1.1444908