Rapid #: -25502959

TYPE:	Article CC:CCG
JOURNAL TITLE:	Mathematical geosciences
USER JOURNAL TITLE:	Mathematical geosciences
ARTICLE TITLE:	Multi-labeled Regularized Marching Tetrahedra Method for Implicit Geological Modeling
ARTICLE AUTHOR:	Sun, Hui
VOLUME:	56
ISSUE:	2
MONTH:	
YEAR:	2024-02-01
PAGES:	219-248
ISSN:	1874-8961
OCLC #:	

Mathematical Geosciences (2024) 56:219–248
https://doi.org/10.1007/s11004-023-10075-9

Multi-labeled Regularized Marching Tetrahedra Method for Implicit Geological Modeling

Hui Sun[1,2] · **Deyun Zhong**[1,2,3] · **Zhaohao Wu**[1,2] · **Liguan Wang**[1,2,3]

Received: 5 December 2022 / Accepted: 23 May 2023 / Published online: 11 July 2023

Abstract

Traditional isosurface extraction methods for a single geological interface is complex and inefficient when it comes to the surface reconstruction of multi-region geological bodies. In this paper, to deal with the problem of implicit surface reconstruction with multiple geological interfaces, we propose the multi-labeled regularized marching tetrahedra method by extending the original regularized marching tetrahedra (RMT) method. The method can directly extract isosurfaces of multi-region geological objects (such as multi-layer geobodies and multi-material geological objects). It inherits the advantages of the RMT method and can generate high-quality meshes preserving the original topology of geological interfaces through vertex clustering. The method includes three steps: (1) intersection calculation by linear interpolation; (2) mesh simplification based on the vertex clustering method; (3) triangular patch extraction using the tetrahedral label indices based on geological region labels. Different labels are utilized to represent different geological regions of sampling points. The triangulation rules and schemes for tetrahedra with different label configurations are defined to distinguish between different geological interfaces. In order to ensure that the boundary of all geological regions in geological bodies can be correctly extracted (continuous outer boundaries of geological regions and consistent model topology), we propose tetrahedral triangulation schemes suitable for multi-region modeling by introducing auxiliary points on tetrahedral faces and inside the tetrahedron. In addition, we construct a look-up table containing 75 tetrahedral configurations to simplify the process of triangular patch extraction and improve the efficiency of the algorithm. The experimental results show that the method can be effectively applied to the surface reconstruction of multi-region geological bodies.

✉ Deyun Zhong
deyun_zhong@csu.edu.cn

1 School of Resources and Safety Engineering, Central South University, Changsha 410083, China

2 Research Center of Digital Mine, Central South University, Changsha 410083, China

3 Changsha Digital Mine Co., Ltd., Changsha 410221, China

Keywords Geological modeling · Implicit modeling · Implicit surface reconstruction · Marching tetrahedra · MRMT

1 Introduction

Implicit geological modeling is a modeling method that calculates spatial implicit functions based on the interpolation of geological exploration data, and automatically generates three-dimensional surfaces of geological bodies through surface reconstruction methods. It has the advantages of a high degree of automation, smooth surfaces of the generated models, and no topological errors. Implicit modeling mainly consists of two steps: (1) computing the implicit functions of geological bodies by spatial interpolation (Macêdo et al. 2009; Yao et al. 2015) and (2) implicit surface reconstruction (Calakli and Taubin 2011; Kalra and Barr 1989; Xiao et al. 2022). There are some common interpolation methods, including the Kriging interpolation method (Boisvert et al. 2009; Che and Jia 2019; Cheng 2013; Marinoni 2003), the inverse-distance weighting interpolation method (Liu et al. 2021), the discrete smooth interpolation (DSI) method (Frank et al. 2007; Mallet 1992), and the radial basis function interpolation (RBF) method (Cuomo et al. 2017; Hillier et al. 2014; Zhong et al. 2019). Franke (1982) compared dozens of different interpolation methods in terms of accuracy, visualization, parameter sensitivity, and performance, and concluded that the RBF interpolation method has better fitting effect. In addition, the extended RBF interpolation methods can be used to construct rich geological constraints, and the corresponding implicit functions can be solved by efficient and robust numerical methods, which have been widely used.

In this paper, we focus on the surface reconstruction problem of implicit modeling with multiple geological interfaces. The implicit surface reconstruction process obtains the three-dimensional model representing the geometric shape of geological bodies by extracting isosurfaces. Unlike the isosurface extraction process of a single geological interface, the implicit surface reconstruction of multi-region geological bodies involves isosurface extraction with multiple geological interfaces, and should ensure topological consistency across different geological interfaces. Among existing surface reconstruction methods, some relevant scholars have studied the isosurface extraction methods for multi-region objects based on the marching cubes (MC) method (Masala et al. 2013; Newman and Yi 2006; Raman and Wenger 2008), the marching tetrahedra (MT) method (Bagley et al. 2016; Ciznicki et al. 2012; Guo et al. 2017, 2021), and the Delaunay triangulation (DT) method (Boissonnat et al. 2013; Chen and Xu 2004). For example, Wu and Sullivan (2003) proposed the multiple material marching cubes method (M3C) based on the marching cubes method. Cong et al. (2005) extended the marching tetrahedra method and applied it to bioluminescence tomography, but it cannot be used in geological modeling, which should ensure topological consistency across different geological interfaces. In addition, many scholars (Dobrina Boltcheva et al. 2009a, b; D. Boltcheva et al. 2009a, b; Fayolle and Pasko 2012) have studied the Delaunay triangulation method and applied it to the isosurface extraction of multi-material or multi-region objects.

The original regularized marching tetrahedra (RMT) method proposed by Treece et al. (1999) is an excellent isosurface extraction method for a single-region object. It combines the marching tetrahedra method for the isosurface extraction process and the vertex clustering method for the mesh simplification process to generate high-quality meshes and to preserve the original topology. Compared with the marching cubes method, extracting isosurfaces using the marching tetrahedra method can ensure topological consistency without additional post-processing. Tetrahedral tessellation based on the body-centered cubic lattice can be generated by subdividing each cube into five tetrahedra (Gueziec and Hummel 1995). However, the tessellation method leads to two possible subdivision schemes for a given cubic lattice, and the resulting tessellation is not very regular (there are two different tetrahedra; the ratio of the lengths between different types of edges is large). The RMT method adopts the tetrahedral tessellation scheme based on the body-centered cubic lattice (Chan 1998), as shown in Fig. 1. The scheme can divide the three-dimensional space into consecutive identical tetrahedra with no ambiguity. Additionally, these tetrahedra are closer to regular tetrahedra, and the ratio of lengths between different edges is small.

In this paper, to deal with the problem of implicit surface reconstruction with multiple geological interfaces, we propose the multi-labeled regularized marching tetrahedra (MRMT) method based on the RMT method. We set a label for each implicit function, which represents a specific geological region. Moreover, we design the tetrahedral triangulation schemes suitable for the isosurface extraction of multiple geological interfaces, and construct a look-up table of triangular patches to simplify the isosurface extraction process. The triangular patches in each tetrahedron can be

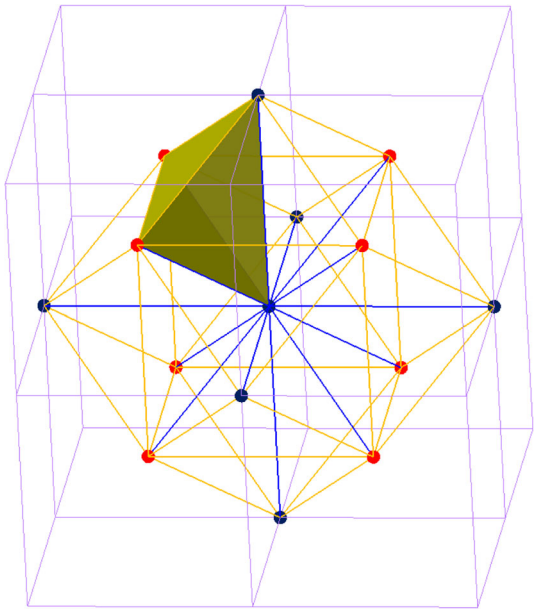


Fig. 1 Tetrahedral tessellation of three-dimensional space based on the body-centered cubic lattice

obtained by looking up the table according to the corresponding label index (determined by the labels of the four sampling points) of the tetrahedron. In addition, we improve the vertex clustering method to make it suitable for multi-region modeling, and use the surface-following (voxel growing) strategy to greatly improve the efficiency of surface reconstruction.

Our key contributions are proposing the tetrahedral triangulation schemes suitable for the isosurface extraction of multi-region geobodies and constructing the look-up table of triangular patches. We implement the MRMT algorithm which can be used to generate three-dimensional models of multi-region geological objects (or other multi-material objects) and preserve the original topology of geological interfaces. The generated meshes are smooth and have no ladder shape. Overall, the MRMT method well inherits the advantages of the RMT method and can effectively reduce the number of triangular patches through vertex clustering, thus reducing rendering requirements and generating high-quality triangular meshes.

2 Overview of the Method

Implicit geological modeling is to construct the constraint rules suitable for geological objects according to original geological data and manual interpretation data, solve the implicit functions that can represent geological bodies, and reconstruct geological surfaces. In this paper, we focus on the surface reconstruction process based on the implicit function. Each function $f(x)$ represents a geological body, and the surface of the geological body is represented by the zero-level set of the function ($\{x \in R^3 | f(x) = 0\}$). The internal domain of the geological body meets $f(x) < 0$, and the external domain of the geological body meets $f(x) > 0$. For multi-region geobodies, there will be multiple implicit functions, and each function corresponds to a geological body.

In this paper, we propose a surface reconstruction method to extract the isosurfaces of complex geological bodies (represented by multiple implicit functions) based on the multi-labeled functions. The labeling of implicit functions is realized by setting a label for each function, and different labels represent different geological regions (the internal domains of geological bodies). The function f_n with label i is denoted by f_n^i. All labels need to be determined according to the priority of the functions, where the smaller the label, the higher the priority of the function. Generally, the labels of functions are all positive, and the label of the external region of geological bodies is set to -1. When multiple surfaces intersect, the higher-priority function will be used to calculate intersections, which can reduce the influence of linear interpolation errors on model accuracy.

The surface reconstruction process of multi-region geological bodies in this paper can be briefly divided into intersection calculation, mesh simplification, and triangular patch extraction. Figure 2 shows the main flow of the MRMT method.

Intersection calculation. In the MRMT method, the three-dimensional space is divided into identical and consecutive tetrahedral voxels based on the body-centered cubic lattice, and the surface-following method is used to search for tetrahedra intersecting isosurfaces. When a tetrahedral edge intersects an isosurface, the function

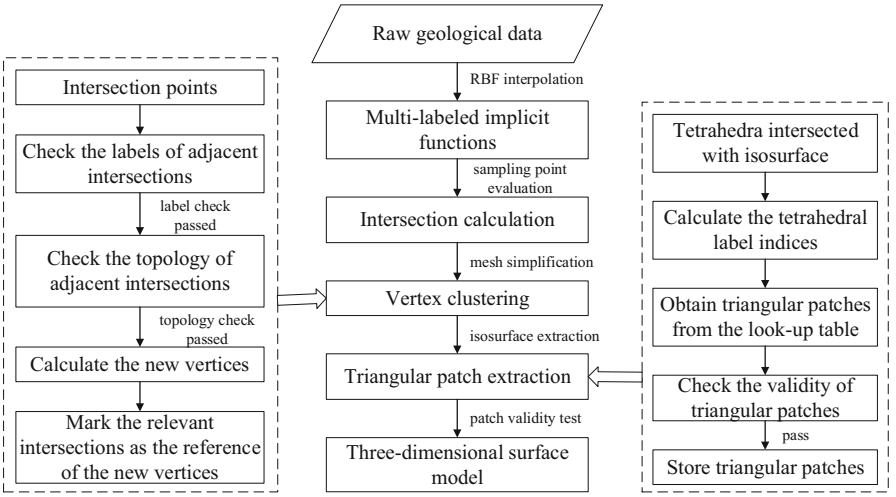


Fig. 2 Overall flow chart of the MRMT algorithm

with higher priority will be selected to calculate the intersection coordinates by linear interpolation.

Mesh simplification. To improve the mesh quality and reduce the number of triangular patches, we improve the vertex clustering method to make it suitable for multi-region modeling. It is necessary to check the topology of related vertices before clustering. The clustering operation will not be performed if it is likely to change the original topology of the associated vertices. In addition, in order to preserve the original features of the sharp and highly curved areas of geological surfaces, we use curvature and gradient weighting to calculate the positions of new vertices.

Triangular patch extraction. In order to extract the isosurface of multi-region geobodies, we propose the new tetrahedral triangulation schemes and construct a look-up table of triangular patches (see Appendix A) in this paper. We can calculate the label index of each tetrahedron by the labels of its four sampling points, and obtain the triangular patches inside a tetrahedron by looking up the table of triangular patches with the label index. Each triangular patch obtained must be validated, and those with the same vertices will be discarded.

3 Tetrahedral Triangulation Method

3.1 Definition of Terms

To avoid confusion, we use the following terms:

Sampling points: the corners of the tetrahedral lattice, used to sample the data.

Edge: a line connecting two adjacent sampling points which are contained by one tetrahedron.

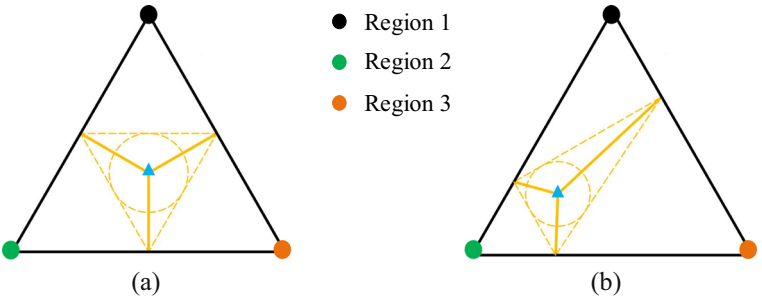


Fig. 3 The incenter of the triangles for region partitioning. The intersections in **a** are at the center of the edges, and **b** the more general case

Intersection: the point at which an edge intersects the isosurface, calculated by linear interpolation of the sampling points at each end of the edge.

Incenter of the triangle: if there are three intersections on a face of a tetrahedron, the center of the triangle formed by these intersections is called the incenter of the triangle. The position of the incenter of the triangle is calculated by the three intersections, as shown by the blue triangles in Fig. 3a, b. Generally, the incenter of the triangle is different from the face center of a tetrahedron. Figure 3b shows the more general case.

Incenter of the tetrahedron: if each face of a tetrahedron has the incenter of the triangle, the center of the inscribed sphere of the tetrahedron formed by the four incenters of the triangles is called the incenter of the tetrahedron.

Vertex: for ease of expression and distinction, we refer to all points used for triangulation, including intersections, the incenter of triangles, and the incenter of tetrahedra, as vertices.

3.2 Triangulation Rules

In order to make the marching tetrahedra method suitable for the isosurface extraction of multi-region geobodies, we must first determine the triangulation schemes of tetrahedral voxels. We should ensure that each intersection is as close as possible to a particular geological interface so as to ensure that the constructed triangular patches can approach the surfaces of geological bodies. This can be achieved by using the high-priority implicit function to calculate intersections. In addition, two constraints need to be satisfied in order for triangular patches in adjacent tetrahedra to be spatially continuous and approximate geological interfaces: region division rule and topology consistency rule.

Constraint 1: region division rule. Each unique region within a tetrahedron has one or more separating surfaces from all other regions so as to ensure that there are no holes in any tetrahedron.

Constraint 2: topology consistency rule. Each face shared by adjacent tetrahedra has the same division scheme, which ensures that the spatial topology of adjacent triangular patches is consistent.

3.3 Triangulation Schemes Based on Intersections

Each tetrahedron has four sampling points belonging to four different geological regions at most. Theoretically, we can get 256 (4^4) different tetrahedral configurations, and only five unique configurations exist by using symmetry. When constructing triangular patches with intersections, we attempted the following triangulation schemes:

When the four sampling points of a tetrahedron belong to the same geological region, the tetrahedron is located inside or outside geological bodies (we regard the external domain of geological bodies as a special geological region, and its label is marked as -1), and there is no triangular patch inside the tetrahedron.

The triangulation schemes when the four sampling points of a tetrahedron belong to two different geological regions are shown in Fig. 7a, b.

When the four sampling points of a tetrahedron belong to three different geological regions, two different triangulation schemes can be obtained if only intersections are used to construct triangular patches, as shown in Fig. 4a, b.

The triangulation scheme when the four sampling points of a tetrahedron belong to four different geological regions is shown in Fig. 4c.

In the following, we verify whether the above triangulation schemes satisfy the constraints proposed in Sect. 3.2. It is easy to see that the above triangulation schemes can ensure that different regions of each tetrahedron are completely separated by triangular patches, so the above triangulation schemes satisfy the first constraint. For the second constraint, we can judge by observing the division schemes of the common triangular faces of adjacent tetrahedra. The division scheme of each triangular face is determined by the geological regions of the three sampling points. When the three sampling points belong to two different geological regions, the face segmentation method is unique, as shown in Fig. 5a. Therefore, we only need to verify whether the segmentation method of the face with sampling points belonging to three different geological regions satisfies the second constraint. When the four sampling points of a tetrahedron belong to three different geological regions, there are two different triangulation schemes (Fig. 4a, b), so we will discuss them in two groups. The first group includes Fig. 4a and c, and the second group includes Fig. 4b, c.

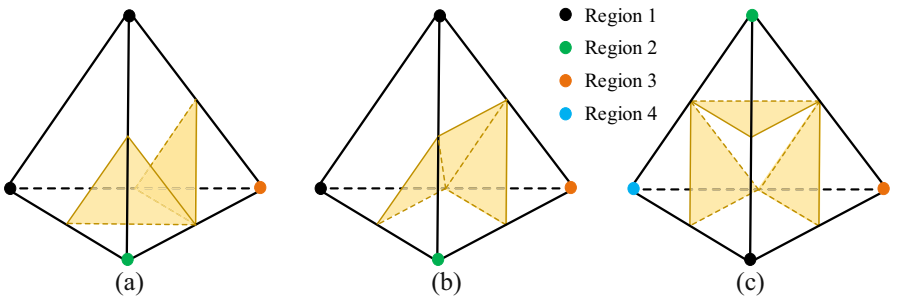


Fig. 4 The tetrahedral triangulation schemes based on intersections. **a** and **b** Two different triangulation schemes for the tetrahedron containing three different geological regions. **c** The triangulation scheme of the tetrahedron containing four different geological regions

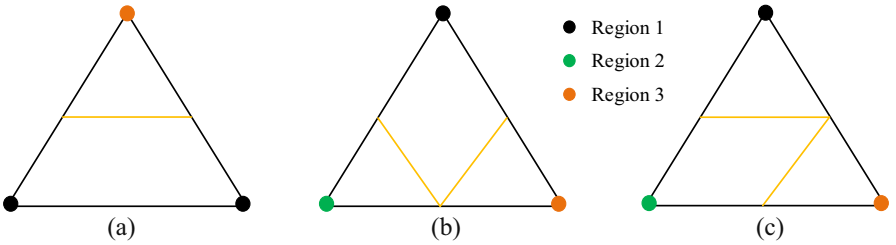


Fig. 5 The division schemes of triangular faces. The sampling points in **a** belong to two different geological regions, and the sampling points in **b** and **c** belong to three different geological regions

We first analyze the first group (Fig. 4a, c). In the tetrahedron shown in Fig. 4a, the segmentation method of triangular faces with sampling points belonging to three different geological regions is shown in Fig. 5b. But the division scheme in Fig. 5b is not spatially symmetric, and we can get three different schemes by rotating it. Therefore, the topology error shown in Fig. 6a may occur between adjacent tetrahedra. Furthermore, more topology errors may occur if the tetrahedron of Fig. 4a is adjacent to the tetrahedron of Fig. 4c or if the two tetrahedra of Fig. 4c are adjacent to each other. Therefore, the tetrahedral triangulation schemes of the first group are not feasible.

For the second group, from the tetrahedron shown in Fig. 4b, we can determine the division scheme of the triangular faces with sampling points belonging to three different geological regions, as shown in Fig. 5c. It is not difficult to see that the division scheme shown in Fig. 5c is spatially the same as that shown in Fig. 5b. If the two tetrahedra of Fig. 4b are adjacent, the topology error shown in Fig. 6b may occur. Similar to the first group, if the tetrahedron of Fig. 4c is considered, more topology errors may occur. Therefore, the tetrahedral triangulation schemes of the second group are also unfeasible.

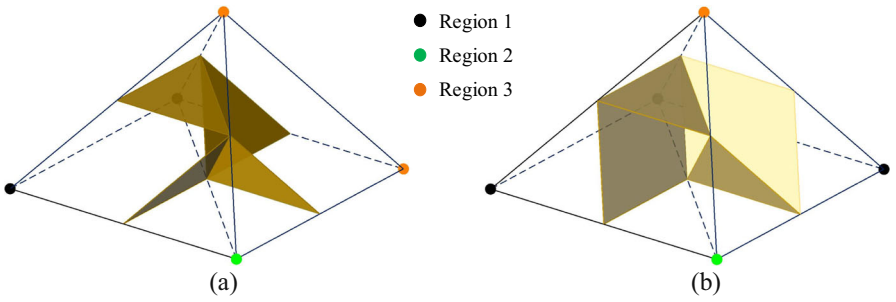


Fig. 6 Topology errors between adjacent tetrahedra with sampling points belonging to three different geological regions. **a** and **b** are two kinds of topological errors of the tetrahedral triangulation schemes shown in Fig. 4a and b, respectively

3.4 Triangulation Schemes with Auxiliary Points

Through the above attempts, we have found that it is difficult to ensure the topological consistency of adjacent tetrahedra in multi-region modeling if we construct triangular patches with intersections only. To solve the problem, we introduce the auxiliary points (including the incenter of the triangle and the incenter of the tetrahedron) and propose the new tetrahedral triangulation schemes, as shown in Fig. 7. The following are the five unique tetrahedral configurations and corresponding triangulation schemes:

For a tetrahedron with sampling points belonging to the same geological region, triangulation is not required.

For a tetrahedron with sampling points belonging to two geological regions (one point to one geological region, three points to the other), the triangulation scheme is shown in Fig. 7a.

For a tetrahedron with sampling points belonging to two geological regions (two points to one region, two points to the other), the triangulation scheme is shown in Fig. 7b.

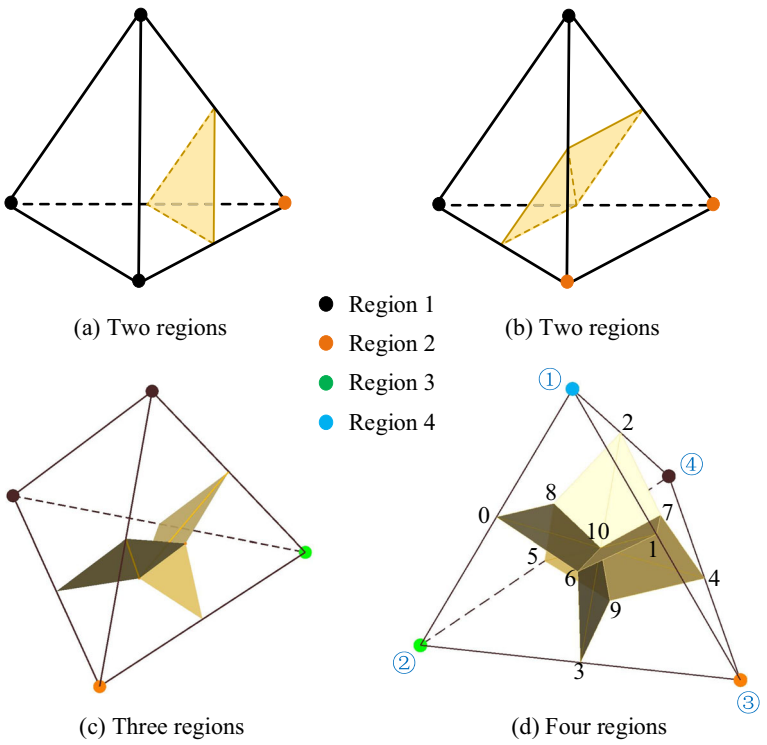


Fig. 7 The tetrahedral triangulation schemes based on intersections and auxiliary points. **a** Three sampling points belong to geological region 1, and one sampling point belongs to geological region 2; **b** two sampling points belong to geological region 1, and two sampling points belong to geological region 2; **c** four sampling points belong to three geological regions; *d* four sampling points belong to four geological regions

For a tetrahedron with sampling points belonging to three geological regions, the triangulation scheme is shown in Fig. 7c. By introducing the incenter of triangles on the two faces of the tetrahedron, five triangular patches are constructed inside the tetrahedron.

For a tetrahedron with sampling points belonging to four geological regions, the triangulation scheme is shown in Fig. 7d. Five auxiliary points are introduced on the faces of the tetrahedron and inside the tetrahedron, and 12 triangular patches are constructed inside the tetrahedron.

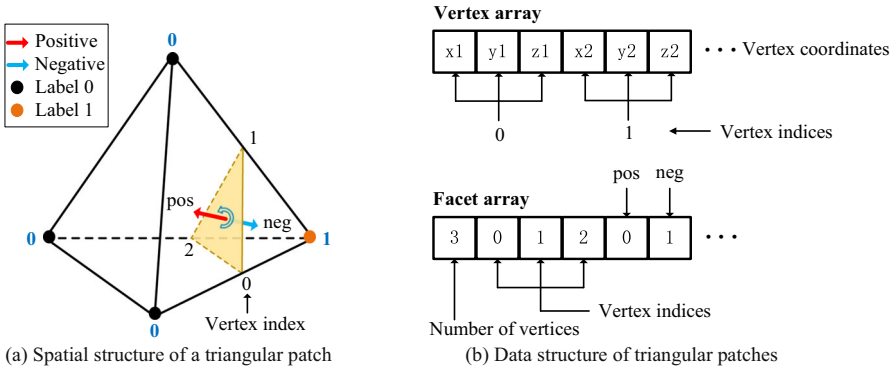
Next, we verify whether the new triangulation schemes satisfy the two constraints in Sect. 3.2. The first constraint is obviously satisfied, so we only need to verify the second constraint. By observing the four triangulation schemes in Fig. 7, we can see that there are only two different face division schemes, as shown in Figs. 3 and 5a. The division scheme of triangular faces with sampling points belonging to two different geological regions is shown in Fig. 5a. The division scheme of triangular faces with sampling points belonging to three different geological regions is shown in Fig. 3. Each division scheme can be uniquely determined by the labels of the three sampling points, which will ensure that the common faces of adjacent tetrahedra have identical division schemes. Therefore, the second constraint can be satisfied, and the new tetrahedral triangulation schemes can be applied to isosurface extraction in multi-region geological modeling.

4 Algorithm

4.1 Label Design

In order to generate the triangular meshes of multi-region geobodies, we use different labels to represent different geological regions: the smaller the label, the higher the priority of the region. We numbered the four sampling points of each tetrahedron from ① to ④, as shown in Fig. 7d. In this way, we can arrange the labels of the sampling points in each tetrahedron according to the above number to obtain the label index that can represent the configuration of the tetrahedron. In addition, we number the vertices (including intersections, the incenter of triangles, and the incenter of the tetrahedron) in each tetrahedron from 0 to 10, as shown in Fig. 7d. We use these numbers to create the look-up table of triangular patches.

Based on the above numbers, we introduce the data structure of multi-labeled triangular patches (Zhong et al. 2021), as shown in Fig. 8b. Each triangular patch in facets array contains six elements. The first element corresponds to the number of vertices of a triangular patch, which is always equal to three for triangular patches. The second to fourth elements correspond to the vertex indices of a triangular patch. The vertex array in Fig. 8b shows the data structure of vertices. The right-hand rule (Honnell 1953) is introduced to distinguish the direction of a triangular patch. The direction of the thumb is positive, and the direction opposite to the thumb is negative, as shown in Fig. 8a. The fifth element is the label of the positive region, and the sixth element is the label of the negative region.



(a) Spatial structure of a triangular patch (b) Data structure of triangular patches

Fig. 8 Spatial structure and data structure of multi-labeled triangular patches. The vertex array stores the vertex coordinates, and the facet array stores the information of each triangular patch

For each tetrahedron, its sampling points belong to four different geological regions at most; hence a tetrahedron contains four different labels at most. To reduce redundancy, we use the labels 0, 1, 2, 3 to characterize tetrahedral configurations and construct the look-up table of triangular patches. Label 0 is used when the four sampling points of a tetrahedron belong to the same geological region. Labels 0 and 1 are used when the four sampling points of a tetrahedron belong to two different geological regions. Labels 0, 1, 2 are used when the four sampling points of a tetrahedron belong to three different geological regions, and labels 0, 1, 2, 3 are used when the four sampling points of a tetrahedron belong to four different geological regions. We construct a label index for each unique tetrahedral configuration according to the above rules.

The label index of each tetrahedron can be calculated using the original labels of its sampling points. It is necessary to ensure that each label index meets the above rules and that the priority order of the labels in each label index matches the original label priority. For example, the labels of the sampling points of a tetrahedron are 5, 4, 4, 2; we can convert them to the label index 2110. The label index can be stored using a four-bit quaternary value.

4.2 Triangular Patches Table

In order to quickly obtain the triangular patches in each tetrahedron during the process of isosurface extraction, the look-up table of triangular patches has been constructed (see Appendix A). According to the definition of label index, there are 75 effective label indices. Each label index corresponds to a set of triangular patches. For example, label index 2201 corresponds to the triangular patches {4,7,9,0,1}, {1,3,9,0,2}, {1,9,7,0,2}, {2,9,5,1,2}, {2,7,9,1,2}. Note that we have omitted the first element (the number of vertices) of each triangular patch in the look-up table compared to the data structure shown in Fig. 8b.

In addition, we construct a look-up table of the patch index (see Appendix B) to simplify the search process of triangular patches. Each patch index is composed of a start index and an end index, such as the label index 2201 corresponding to the patch

index {382, 386}. In the process of isosurface extraction, we need to create a facet array to store the 488 triangular patches (the index range of the array is 0 to 487) and a patch index array to store the 256 triangular patch indices (the index range of the array is 0 to 255) in the order of label index. In this way, we can directly obtain the triangular patches in each tetrahedron through its label index. For example, the labels of a tetrahedron are 5, 5, 2, 3; we first calculate the label index of the tetrahedron (2201), then calculate the decimal value of the label index (161), and then obtain the triangular patch index: the 161st element ({382, 386}) of the patch index array. Finally, we can obtain the triangular patches from the facet array by using the start index and the end index in the triangular patch index.

4.3 Sampling Grid

Before surface reconstruction, it is necessary to define an appropriate sampling grid to determine sampling points. In this paper, we adopt the tetrahedral tessellation scheme based on the body-centered cubic lattice. The scheme can divide the three-dimensional space into consecutive identical tetrahedra, and each tetrahedron has one pair of opposite edges of two units and two pairs of $\sqrt{3}$ units. In the sampling grid, each sampling point has 14 edges connected to it, as shown in Fig. 9a. We number these edges from 0 to 13, as shown in Fig. 9b. For each sampling point, we can use a 16-bit binary number to record the intersection information of its connected edges. For example, for a sampling point whose adjacent edges (1,9,13) intersect the isosurface, the intersection information can be represented by 0x2202 (0010001000000010).

4.4 Intersection Point Calculation

After defining the sampling grid, we need to calculate intersections. Each sampling point has 14 edges connected to it, and each edge is associated with two sampling points. Therefore, when traversing sampling points, we only need to examine seven related edges for each new sampling point, as shown by the red edges in Fig. 9b. Each sampling point is surrounded by 24 tetrahedra, and each tetrahedron is associated with four sampling points. Therefore, we need to examine six tetrahedra for each new sampling point; the common faces of these tetrahedra are shown in Fig. 9c, and these tetrahedra are shown in Fig. 9d.

When searching the tetrahedra intersected with an isosurface, the labels of the four sampling points of each tetrahedron need to be calculated first. For any sampling point $p \in R^3$, the labeled function f_n^i (the label is equal to i) represents a signed distance field, and the distance from point p to the isosurface is $d = f_n^i(p)$. If $d < 0$, point p is inside the geological region represented by function f_n^i, and if $d > 0$, point p is outside the geological region. The pseudo-code of the algorithm flow for calculating the label of the sampling point is shown in Algorithm 1 (calculate the label of the sampling point).

We can judge whether there is an intersection on an edge by the labels of the sampling points associated with the edge. If the two labels are different, then there is an intersection on the edge. We can calculate the coordinate of the intersection

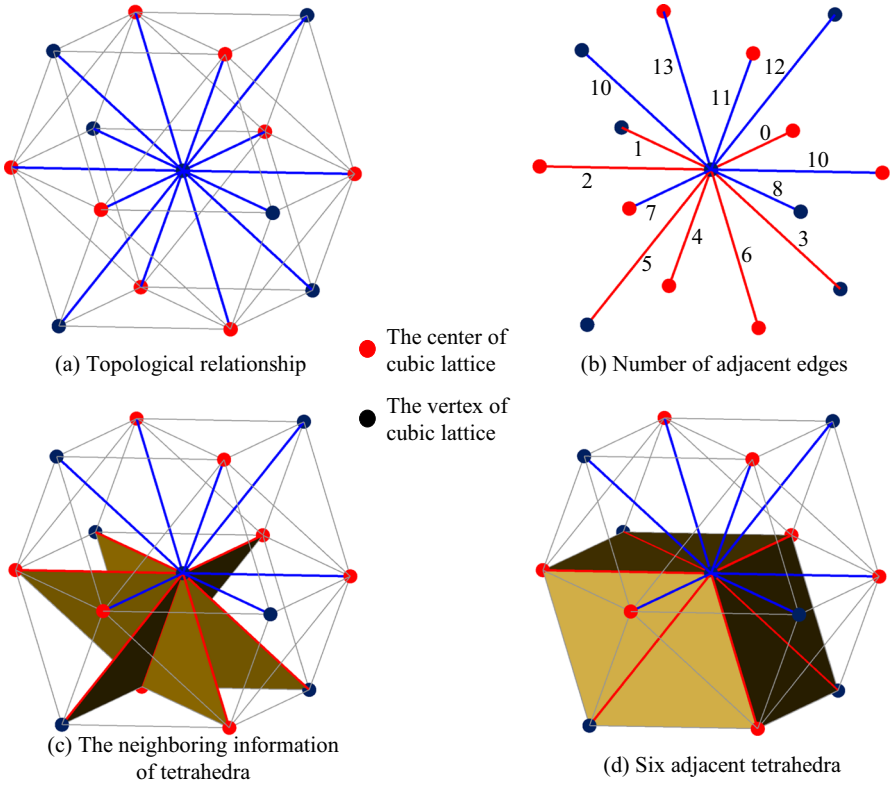


Fig. 9 Spatial topological relationships between adjacent sampling points. The red sampling points are the center of adjacent cubic lattices, and the blue sampling points are the vertices of cubic lattices

by linear interpolation and push it into the vertex array. Let M and N be the two sampling points of an edge intersected with an isosurface; M is inside the geological region represented by function f_a^1, and N is inside the geological region represented by function f_b^2. Since the label of function f_a^1 is smaller, function f_a^1 should be selected to calculate the coordinates of the intersection point. Let C be the intersection point on the edge, which can be calculated by Eq. 1

$$C = \frac{\left|f_a^1(M)\right|}{\left|f_a^1(M)\right| + \left|f_a^1(N)\right|} N + \frac{\left|f_a^1(N)\right|}{\left|f_a^1(M)\right| + \left|f_a^1(N)\right|} M. \tag{1}$$

It should be noted that the coordinates calculated by Eq. 1 have high accuracy when tetrahedral voxels are small or the surfaces of geological objects are smooth. However, if the surfaces of geological objects have large gradients or sharp corners, the coordinates will not be very precise. More accurate coordinates can be calculated using the bisection method (Corliss 1977).

In order to improve the efficiency of the surface reconstruction process, we use the surface-following method (Hartmann 1998) to search for tetrahedra intersecting

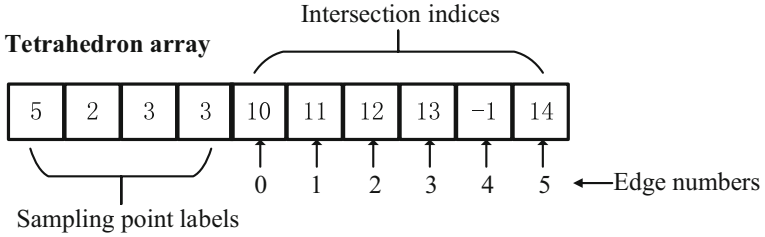


Fig. 10 The data structure of the tetrahedron intersected with an isosurface. The intersection index is one kind of vertex index, and its structure is introduced in Sect. 4.1. If there is no intersection point on an edge, we set its intersection index value to −1

isosurfaces. First, we need to determine one seed point (the point on a geological interface) for each geological interface. Points located on geological interfaces in the original geological sampling data can be used as initial seed points. In addition, we can construct some linear functions in different directions within the modeling space and calculate the intersection points of these lines with each implicit surface. These intersections can also be used as initial seed points. Then we can determine the seed tetrahedra (the tetrahedron containing at least one seed point inside) using these seed points. When using the surface-following method to search for the tetrahedra intersecting isosurfaces, an incremental queue of tetrahedra should be constructed for each geobody to store the tetrahedra to be evaluated. The evaluated sampling points will be marked to avoid repeated calculations. The surface-following process ends when the incremental queue is empty. The labels and the intersections of each tetrahedron intersected with an isosurface need to be recorded; the data structure is shown in Fig. 10.

Algorithm 1 Calculate the label of the sampling point

Input: Sampling point $\boldsymbol{p}$, the labeled function $\boldsymbol{f_n}(n = 1,2,\ldots,m)$.
Output: Sampling point label $\boldsymbol{label}$.
Initialize an empty array $\boldsymbol{labels}$.
foreach $\boldsymbol{f_n}(n = 1,2,\ldots,m)$ **do**
    if $\boldsymbol{f_n}(\boldsymbol{p}) \leq 0$ **then**
        Push the label of $\boldsymbol{f_n}$ to $\boldsymbol{labels}$.
    end if
end for
if $\boldsymbol{labels}$ is empty **then**
    $\boldsymbol{label} = -1$
else
    $\boldsymbol{label} = Min(\boldsymbol{labels})$
end if
return $\boldsymbol{label}$

4.5 Mesh Simplification

Using the original MT method to extract the isosurface results in more triangular patches than the MC method. Furthermore, these triangular patches can be exceedingly

small or thin, resulting in substantial storage and rendering requirements. In order to reduce the number of triangular patches and improve the quality of triangular meshes, we improve the vertex clustering method used in Treece et al. (1999), which can preserve the original topology of geological surfaces while clustering vertices. With the introduction of geological region labels, it can be applied to the multi-region modeling.

In this paper, auxiliary points are introduced for extracting the isosurface of multi-region geobodies. In order to preserve the original boundary between different regions, auxiliary points are not allowed to be clustered. Therefore, vertex clustering only refers to the intersections. Two conditions need to be satisfied before clustering: first, the labels of intersections to be clustered must be the same; second, the clustering process cannot change the original topology of geological surfaces.

For the first condition, we only need to check that the labels of all intersections to be clustered are the same. If there is more than one type of label, we must give up clustering. If the labels are the same, we need to check the second condition, which is to check the topology of these intersections. Several topological types that cannot be clustered were introduced by Treece et al. (1999), including closed surface (clustering the intersections will lead to the elimination of the surface), surface with a hole in it (clustering the intersections will close up the hole), and concave surface (clustering the intersections might result in flattening the surface). Note that if intersections form more than one separate surface, one new vertex is required for each surface. Each new vertex can be obtained by calculating the average of the relevant intersections, and the label of the new vertex is the same as the labels of these intersections. After new vertices are generated, the original intersections need to be marked as the reference of the new vertices. Computing the coordinates of new vertices by averaging the intersection coordinates is appropriate for smooth surfaces; however, it may alter the original features of geological surfaces for sharp corners or regions of high curvature. This can be overcome by using a data curvature estimate to weight the original intersection coordinates (Treece et al. 1999).

4.6 Triangular Patch Extraction

Classical isosurface extraction methods, such as the MC method and the MT method, utilize look-up tables to determine patch information in voxels, which greatly enhances the operational efficiency of these algorithms. Theoretically, the MC method has 256 hexahedral configurations, the MT method has 16 tetrahedral configurations, and the MRMT method has 256 (4^4) tetrahedral configurations. In contrast, the M3C method has 16,777,216 (8^8) hexahedral configurations, making it impossible to construct a look-up table for this method due to its complex and numerous configurations. At

present, the isosurface extraction process of the M3C method mainly involves classifying hexahedra according to certain features and constructing effective patch extraction rules. As a result, the isosurface extraction speed of the M3C method is relatively slow.

In this paper, we construct a look-up table of triangular patches for the MRMT method. In this section, we adopt the tetrahedral triangulation schemes (as described in Sect. 3.4) to extract triangular patches. For a tetrahedron that intersects one or more isosurfaces, we first need to calculate the label index of the tetrahedron by analyzing the labels of its sampling points. Its vertex information can be obtained by looking up Table 1 with the label index. If there are auxiliary vertices, their coordinates need to be calculated and pushed into the vertex array. Then we can obtain the triangular patches composed of vertex numbers by looking up the table (Appendix A) with the label index. The actual triangular patches can be obtained by replacing these vertex numbers with actual vertex indices. In addition, before pushing triangular patches into the triangular patch array, we need to check the vertex indices of each triangular patch. If the vertex indices of a triangular patch are different, we save the patch or discard it directly. The pseudo-code of the algorithm flow for extracting triangular patches is shown in Algorithm 2 (triangular patch extraction).

Table 1 The look-up table of a vertex mask

Label index	Mask	Label index	Mask	Label index	Mask	Label index	Mask	Label index	Mask
0001	0x34	0132	0x7ff	1012	033d	1210	0x33d	2103	0x7ff
0010	0x1a	0201	0x33d	1020	0xdf	1220	0x1b7	2110	0x1b7
0011	0x2e	0210	0x27b	1021	0x27b	1230	0x7ff	2120	0x33d
0012	0x2be	0211	0x16f	1022	0x16f	1302	0x7ff	2130	0x7ff
0021	0x2be	0212	0xdf	1023	0x7ff	1320	0x7ff	2201	0x2be
0100	0x29	0213	0x7ff	1032	0x7ff	2001	0x1b7	2210	0x2be
0101	0x1d	0221	0x1b7	1100	0x2e	2010	0xdf	2301	0x7ff
0102	0x33d	0231	0x7ff	1101	0x1a	2011	0x16f	2310	0x7ff
0110	0x33	0312	0x7ff	1102	0x2be	2012	0x27b	3012	0x7ff
0111	0x7	0321	0x7ff	1110	0x34	2013	0x7ff	3021	0x7ff
0112	0x1b7	1000	0x7	1120	0x2be	2021	0x33d	3102	0x7ff
0120	0x27b	1001	0x33	1200	0x16f	2031	0x7ff	3120	0x7ff
0121	0xdf	1002	0x1b7	1201	0x27b	2100	0x16f	3201	0x7ff
0122	0x16f	1010	0x1d	1202	0xdf	2101	0xdf	3210	0x7ff
0123	0x7ff	1011	0x29	1203	0x7ff	2102	0x27b		

Algorithm 2 Triangular patch extraction

Input: Tetrahedron intersected with isosurface $\boldsymbol{T_i}(i = 1,2,\ldots,n)$, triangular patch table $\boldsymbol{P}$, vertex mask table $\boldsymbol{M}$.
Output: Facet array $\boldsymbol{F}$.
Initialize an empty facet array $\boldsymbol{F}$.
foreach $\boldsymbol{T_i}(i = 1,2,\ldots,n)$ **do**
    Calculate the label index $\boldsymbol{li}$ of $\boldsymbol{T_i}$.
    Obtain the vertex mask $\boldsymbol{mask}$ of $\boldsymbol{T_i}$ by looking up the table $\boldsymbol{M}$ with $\boldsymbol{li}$.
    if $\boldsymbol{mask} > 0x3f$ **then**
        Calculate the auxiliary points and add them to the vertex array.
    end if
    Obtain the triangular patches $\boldsymbol{facets}$ within $\boldsymbol{T_i}$ by looking up table $\boldsymbol{P}$ with $\boldsymbol{li}$.
    foreach $\boldsymbol{facet} \in \boldsymbol{facets}$ **do**
        Replace vertices numbers in $\boldsymbol{facet}$ with vertex indices of $\boldsymbol{T_i}$.
        if the vertex indices in $\boldsymbol{facet}$ are not repeated **then**
            Replace the labels in $\boldsymbol{facet}$ with the real labels of $\boldsymbol{T_i}$.
            Push $\boldsymbol{facet}$ **to** $\boldsymbol{F}$.
        end if
    end for
end for
return $\boldsymbol{F}$

5 Experimental Results

In this paper, we have implemented the MRMT algorithm using the C++ programming language. To validate the viability of the method, we have modeled several geological bodies based on several geological sampling datasets. In addition, we conducted four control experiments using the original M3C method and the MRMT method on a Windows 64-bit PC with 3.80 GHz Intel® Core™ i7-10700 K and 32 GB RAM. In each control experiment, we controlled the cubic lattice size to be the same (the number of tetrahedral voxels is 12 times that of cubic lattices). Table 2 shows the data from the control experiments. The experimental results show that the models generated by the RMT method have fewer vertices and fewer triangular patches, and the use of the surface-following strategy in the MRMT method significantly reduces the runtime of the algorithm.

Table 2 The statistics table of four control experiments. N_R is the number of different regions, d_{vox} is the voxel size (the side length of cubic lattices), N_V is the number of vertices, and N_P is the number of triangular patches

Dataset	N_R	d_{vox} (m)	N_V		N_P		Time (s)	
			M3C	MRMT	M3C	MRMT	M3C	MRMT
Figure 11b	1	1	26,177	22,463	51,695	43,907	2.43	1.01
Figure 12d	2	1	28,514	25,895	57,028	51,790	5.23	3.42
Figure 13c	3	5	20,204	17,914	40,404	35,824	69.57	3.90
Figure 15a	4	10	140,015	114,353	280,030	228,722	1,556.46	28.66

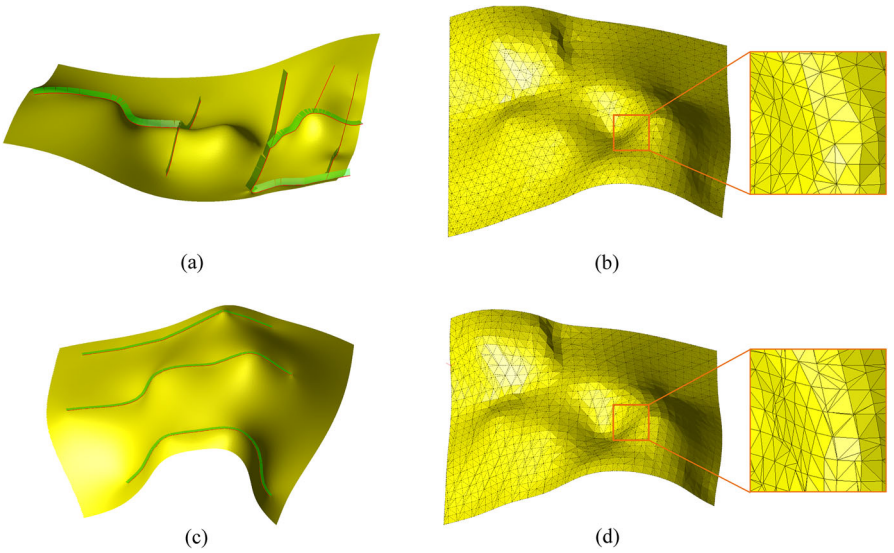


Fig. 11 Surface reconstruction results of several geological interfaces. **a–c** The triangular meshes generated by the MRMT method. **d** The triangular mesh generated by the M3C method

Figure 11a to c show the meshes of four different geological interfaces generated by the MRMT algorithm, and Fig. 11d shows the mesh generated by the M3C algorithm for the same geological surface in Fig. 11b. We have visualized the constraint lines of the meshes in Fig. 11a, c, and the edges of the meshes in Fig. 11b, d. We can see that the meshes generated by the MRMT algorithm are smooth, the ratio of the lengths between the different triangular edges is smaller, and the features of surfaces are well preserved. Comparing Fig. 11b and d, we can see that the triangular mesh generated by the MRMT method is more regular.

Figure 12 shows the generated models of three geological bodies containing two different geological regions. Figure 12d shows the model topology at the junction of two geological regions. The surface at the junction of the two geological regions is completely closed without gaps or holes. Furthermore, the junction of the two geological regions is relatively smooth and does not have a ladder shape. Figure 12c shows the surface of geological region 1 shown in Fig. 12d. The surface is smooth and continuous, except at the junction of two geological regions. Figures 13 and 14 show the reconstructed models of geological bodies containing multiple geological regions. Each geological region has a continuous and closed surface, and the geological features are well preserved.

Figure 15a shows the reconstructed model of a complex geological object containing four geological regions. At the junction of the three geological regions, each geological region is effectively separated, and there are no holes or topology errors. The profile of the geological object cut along the section in Fig. 15b is shown in Fig. 15c. It shows that there are continuous surfaces between different geological regions, and the meshes of the three regions are consistent without gaps. Figure 16a shows a model

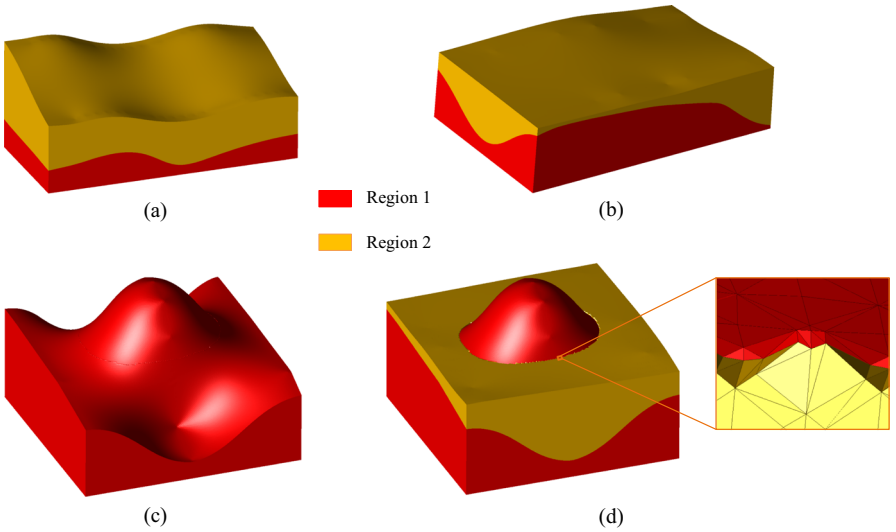


Fig. 12 Surface reconstruction results of several geological objects containing two geological regions. **a**, **b**, **d** show the surface reconstruction results of three different geological bodies, and **c** the model of geological region 1 from (**d**)

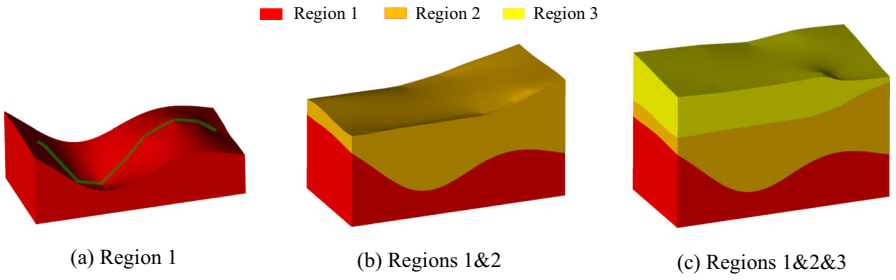


Fig. 13 The generated models of a geological object containing three geological regions. **a** The constraint lines of region 1

generated for a geological body with four intersecting geological regions. It can be seen from Fig. 16c that each geological region has a closed surface, and the junction of multiple geological regions is continuous without gaps. The results indicate that the MRMT method is suitable for surface reconstruction of multi-region geological bodies.

6 Conclusion and Discussion

The multi-labeled regularized marching tetrahedra method proposed in this paper can work well for the isosurface extraction of multi-region geological bodies. In general,

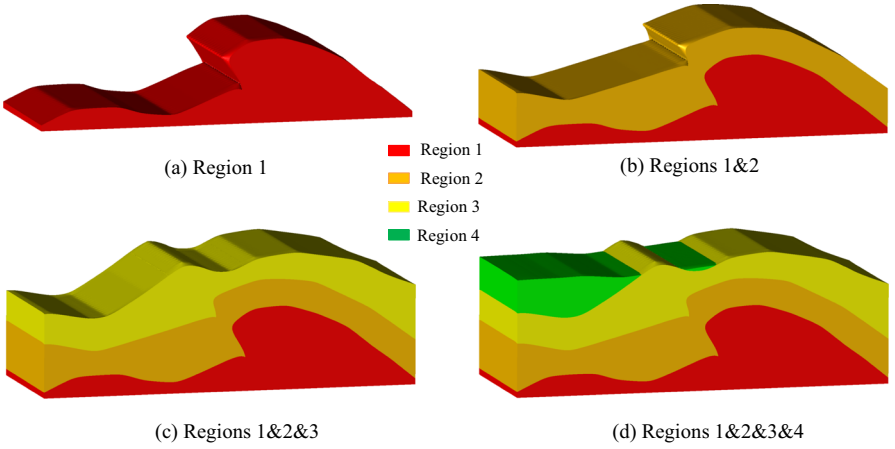


Fig. 14 Surface reconstruction results of a complex geological object containing four geological regions

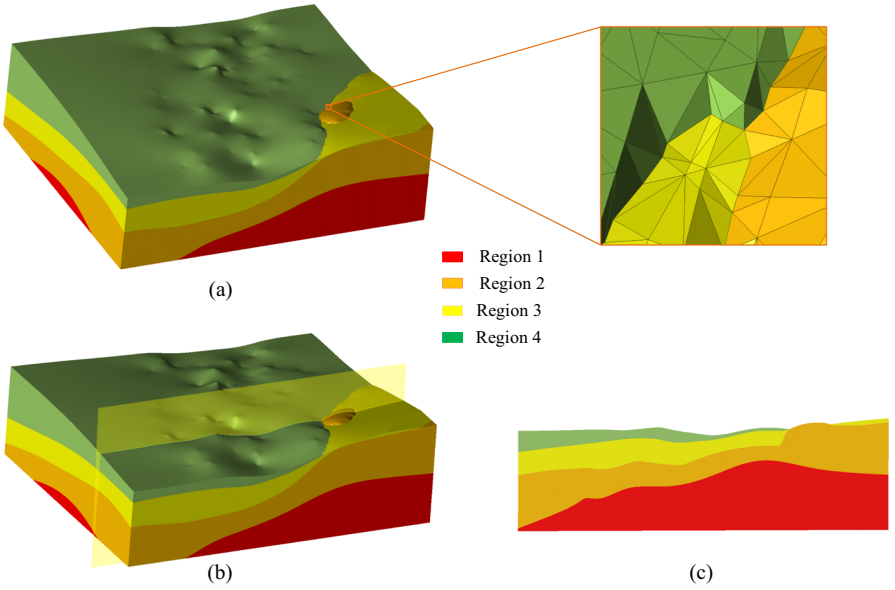


Fig. 15 Surface reconstruction results of a geological object containing four geological regions. **a** The resulting meshes of the geological object; (**c**) shows the profile cut along the section in **b**

it reduces the number of triangular patches and preserves the original topology of the geological surface. We propose a tetrahedral triangulation scheme suitable for multi-region modeling, and construct a look-up table of triangular patches based on the label index. The look-up table greatly improves the efficiency of isosurface extraction. Compared with the M3C method, the MRMT method is more efficient due to its use of the look-up table strategy. Theoretically, the method can be applied to the surface

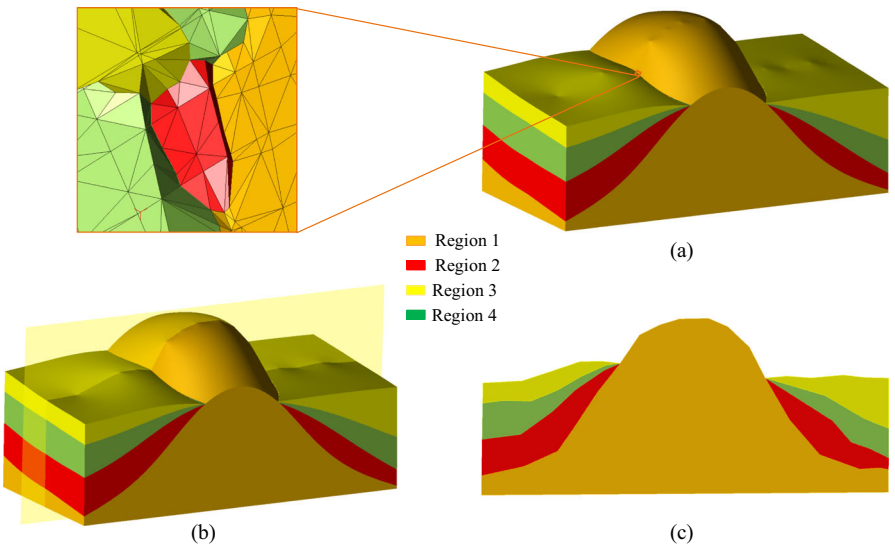


Fig. 16 Reconstructed meshes of a geological object with four intersecting geological regions. **a** The reconstructed model of the geological object, and **c** the profile cut along the section in **b**

reconstruction of geological objects containing any number of geological regions. Moreover, compared with the original M3C method, models generated by the MRMT method have no ladder shape at the junction of multiple geological regions, and the resulting meshes are smooth. In addition, the MRMT method inherits the advantages of the RMT method, and can generate high-quality triangular meshes. After the process of vertex clustering, the number of triangular patches is greatly reduced, and the efficiency and quality of model rendering are improved dramatically. The surface-following method is also utilized to efficiently search the tetrahedra intersecting the isosurface.

However, the method also has some aspects to be improved. Although the MRMT method can be used to extract the isosurface of multi-region geobodies, the auxiliary point used in this paper is not guaranteed to be located at the junction of geological interfaces, so the mesh generated at the junction of multiple regions may not be smooth. Figure 17 shows that a model has a local depression at the junction of two geological regions. An effective approach is to change the calculation method of the auxiliary point so that the auxiliary point can approximate the junction of different geological interfaces, but this comes at a certain performance cost. Another limitation of the method is that if there are some sharp geometry features on the surface of geological bodies, these features cannot be reliably recognized by judging the labels of sampling points. Therefore, the sharp features may not be well preserved unless a smaller tetrahedral lattice is used. Figure 18 shows a feasible tetrahedral subdivision method, which includes three steps: (1) Determine the tetrahedral voxels to be subdivided, and subdivide these tetrahedra by creating new vertices at the midpoints of

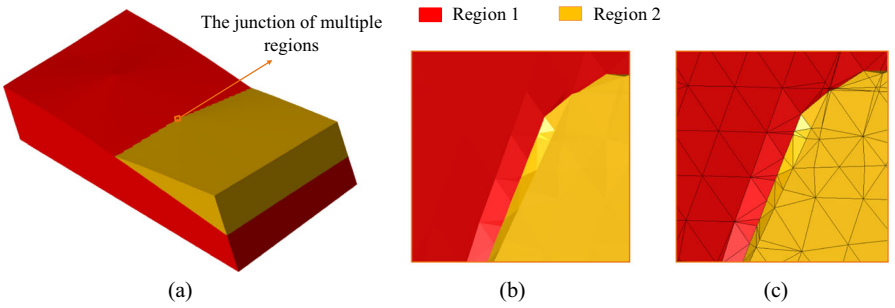


Fig. 17 Reconstructed meshes of a geological object containing two intersecting geological regions. **b** The local surface at the junction of the two geological regions, and the triangulated mesh is shown in **c**

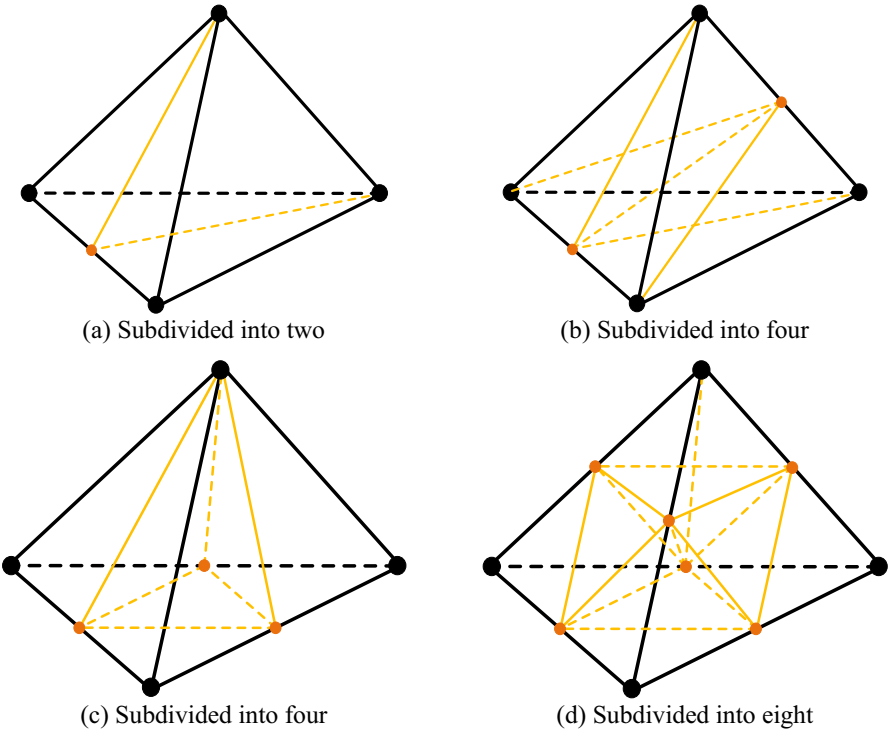


Fig. 18 The subdivision schemes of tetrahedral voxels

their edges, as shown in Fig. 18d. (2) Subdivide the tetrahedra adjacent to the subdivided tetrahedra according to the four schemes shown in Fig. 18. (If a triangular face is introduced with two vertices on its two edges, then an additional vertex should be introduced on its third edge.) (3) Repeat step 2 until no tetrahedron needs to be subdivided. In addition, the adaptive surface reconstruction method based on the octree is an important research direction in the future.

Acknowledgements We thank the reviewers for their comments and suggestions to improve the quality of the paper. This study was supported by the National Natural Science Foundation of China (52104171) and the China Postdoctoral Science Foundation (No. 2022T150740).

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Appendix A: Look-up table of triangular patches

Label index	Triangular patches											
0000												
0001	{5,4,2,0,1}											
0010	{4,3,1,0,1}											
0011	{3,1,5,0,1}	{1,2,5,0,1}										
0012	{4,7,9,1,2}	{9,3,1,0,1}	{7,9,1,0,1}	{5,9,2,0,2}	{9,7,2,0,2}							
0021	{9,7,4,1,2}	{9,3,1,0,2}	{7,9,1,0,2}	{5,9,2,0,1}	{9,7,2,0,1}							
0100	{3,5,0,0,1}											
0101	{0,3,4,0,1}	{0,4,2,0,1}										
0102	{5,9,8,1,2}	{3,9,0,0,1}	{9,8,0,0,1}	{9,4,2,0,2}	{8,9,2,0,2}							
0110	{0,1,5,0,1}	{1,4,5,0,1}										
0111	{0,1,2,0,1}											
0112	{2,8,7,0,2}	{0,1,8,0,1}	{1,7,8,0,1}	{7,8,4,1,2}	{8,5,4,1,2}							
0120	{3,6,9,1,2}	{9,5,0,0,1}	{6,9,0,0,1}	{9,6,1,0,2}	{4,9,1,0,2}							
0121	{1,7,6,0,2}	{0,6,7,0,1}	{0,7,2,0,1}	{6,7,3,1,2}	{7,4,3,1,2}							
0122	{0,6,8,0,1}	{1,8,6,0,2}	{1,2,8,0,2}	{3,6,8,1,2}	{3,8,5,1,2}							
0123	{0,10,8,0,1}	{0,6,10,0,1}	{2,8,10,0,3}	{2,10,7,0,3}	{1,10,6,0,2}	{1,7,10,0,2}	{10,9,3,1,2}	{6,10,3,1,2}	{4,7,10,2,3}	{4,10,9,2,3}	{5,10,8,1,3}	{5,9,10,1,3}
0132	{0,10,8,0,1}	{0,6,10,0,1}	{2,8,10,0,2}	{2,10,7,0,2}	{1,10,6,0,3}	{1,7,10,0,3}	{10,9,3,1,3}	{6,10,3,1,3}	{10,7,4,2,3}	{9,10,4,2,3}	{5,10,8,1,2}	{5,9,10,1,2}
0201	{8,9,5,1,2}	{3,9,0,0,2}	{9,8,0,0,2}	{9,4,2,0,1}	{8,9,2,0,1}							
0210	{9,6,3,1,2}	{9,5,0,0,2}	{6,9,0,0,2}	{9,6,1,0,1}	{4,9,1,0,1}							
0211	{0,6,8,0,2}	{1,8,6,0,1}	{1,2,8,0,1}	{8,6,3,1,2}	{5,8,3,1,2}							
0212	{1,7,6,0,1}	{0,6,7,0,2}	{0,7,2,0,2}	{3,7,6,1,2}	{3,4,7,1,2}							
0213	{0,10,8,0,2}	{0,6,10,0,2}	{2,8,10,0,3}	{2,10,7,0,3}	{1,10,6,0,1}	{1,7,10,0,1}	{3,9,10,1,2}	{3,10,6,1,2}	{4,7,10,1,3}	{4,10,9,1,3}	{5,10,8,2,3}	{5,9,10,2,3}
0221	{2,8,7,0,1}	{0,1,8,0,2}	{1,7,8,0,2}	{4,8,7,1,2}	{4,5,8,1,2}							
0231	{0,10,8,0,2}	{0,6,10,0,2}	{2,8,10,0,1}	{2,10,7,0,1}	{1,10,6,0,3}	{1,7,10,0,3}	{10,9,3,2,3}	{6,10,3,2,3}	{10,7,4,1,3}	{9,10,4,1,3}	{8,10,5,1,2}	{10,9,5,1,2}
0312	{0,10,8,0,3}	{0,6,10,0,3}	{2,8,10,0,2}	{2,10,7,0,2}	{1,10,6,0,1}	{1,7,10,0,1}	{3,9,10,1,3}	{3,10,6,1,3}	{4,7,10,1,2}	{4,10,9,1,2}	{8,10,5,2,3}	{10,9,5,2,3}
0321	{0,10,8,0,3}	{0,6,10,0,3}	{2,8,10,0,1}	{2,10,7,0,1}	{1,10,6,0,2}	{1,7,10,0,2}	{3,9,10,2,3}	{3,10,6,2,3}	{10,7,4,1,2}	{9,10,4,1,2}	{8,10,5,1,3}	{10,9,5,1,3}
1000	{2,1,0,0,1}											

Label index	Triangular patches											
1001	{5,1,0,0,1}	{5,4,1,0,1}										
1002	{2,8,7,1,2}	{8,1,0,0,1}	{8,7,1,0,1}	{7,8,4,0,2}	{8,5,4,0,2}							
1010	{4,3,0,0,1}	{2,4,0,0,1}										
1011	{0,5,3,0,1}											
1012	{5,9,8,0,2}	{0,9,3,0,1}	{0,8,9,0,1}	{9,4,2,1,2}	{8,9,2,1,2}							
1020	{1,7,6,1,2}	{7,6,0,0,1}	{2,7,0,0,1}	{6,7,3,0,2}	{7,4,3,0,2}							
1021	{3,6,9,0,2}	{0,5,9,0,1}	{0,9,6,0,1}	{9,6,1,1,2}	{4,9,1,1,2}							
1022	{8,6,0,0,1}	{1,8,6,1,2}	{1,2,8,1,2}	{3,6,8,0,2}	{3,8,5,0,2}							
1023	{8,10,0,0,1}	{10,6,0,0,1}	{2,8,10,1,3}	{2,10,7,1,3}	{1,10,6,1,2}	{1,7,10,1,2}	{10,9,3,0,2}	{6,10,3,0,2}	{4,7,10,2,3}	{4,10,9,2,3}	{5,10,8,0,3}	{5,9,10,0,3}
1032	{8,10,0,0,1}	{10,6,0,0,1}	{2,8,10,1,2}	{2,10,7,1,2}	{1,10,6,1,3}	{1,7,10,1,3}	{10,9,3,0,3}	{6,10,3,0,3}	{10,7,4,2,3}	{9,10,4,2,3}	{5,10,8,0,2}	{5,9,10,0,2}
1100	{5,1,3,0,1}	{5,2,1,0,1}										
1101	{1,3,4,0,1}											
1102	{4,7,9,0,2}	{1,3,9,0,1}	{1,9,7,0,1}	{5,9,2,1,2}	{9,7,2,1,2}							
1110	{2,4,5,0,1}											
1120	{9,7,4,0,2}	{9,3,1,1,2}	{7,9,1,1,2}	{2,9,5,0,1}	{2,7,9,0,1}							
1200	{0,6,8,1,2}	{6,8,1,0,1}	{8,2,1,0,1}	{8,6,3,0,2}	{5,8,3,0,2}							
1201	{9,6,3,0,2}	{9,5,0,1,2}	{6,9,0,1,2}	{1,6,9,0,1}	{1,9,4,0,1}							
1202	{6,7,1,0,1}	{0,6,7,1,2}	{0,7,2,1,2}	{3,7,6,0,2}	{3,4,7,0,2}							
1203	{0,10,8,1,2}	{0,6,10,1,2}	{2,8,10,1,3}	{2,10,7,1,3}	{6,10,1,0,1}	{10,7,1,0,1}	{3,9,10,0,2}	{3,10,6,0,2}	{4,7,10,0,3}	{4,10,9,0,3}	{5,10,8,2,3}	{5,9,10,2,3}
1210	{8,9,5,0,2}	{3,9,0,1,2}	{9,8,0,1,2}	{2,4,9,0,1}	{2,9,8,0,1}							
1220	{7,8,2,0,1}	{0,1,8,1,2}	{1,7,8,1,2}	{4,8,7,0,2}	{4,5,8,0,2}							
1230	{0,10,8,1,2}	{0,6,10,1,2}	{10,8,2,0,1}	{7,10,2,0,1}	{1,10,6,1,3}	{1,7,10,1,3}	{10,9,3,2,3}	{6,10,3,2,3}	{10,7,4,0,3}	{9,10,4,0,3}	{8,10,5,0,2}	{10,9,5,0,2}
1302	{0,10,8,1,3}	{0,6,10,1,3}	{2,8,10,1,2}	{2,10,7,1,2}	{6,10,1,0,1}	{10,7,1,0,1}	{3,9,10,0,3}	{3,10,6,0,3}	{4,7,10,0,2}	{4,10,9,0,2}	{8,10,5,2,3}	{10,9,5,2,3}
1320	{0,10,8,1,3}	{0,6,10,1,3}	{10,8,2,0,1}	{7,10,2,0,1}	{1,10,6,1,2}	{1,7,10,1,2}	{3,9,10,2,3}	{3,10,6,2,3}	{10,7,4,0,2}	{9,10,4,0,2}	{8,10,5,0,3}	{10,9,5,0,3}
2001	{7,8,2,1,2}	{8,1,0,0,2}	{8,7,1,0,2}	{7,8,4,0,1}	{8,5,4,0,1}							
2010	{6,7,1,1,2}	{7,6,0,0,2}	{2,7,0,0,2}	{6,7,3,0,1}	{7,4,3,0,1}							
2011	{8,6,0,0,2}	{6,8,1,1,2}	{8,2,1,1,2}	{3,6,8,0,1}	{3,8,5,0,1}							
2012	{3,6,9,0,1}	{0,5,9,0,2}	{0,9,6,0,2}	{1,6,9,1,2}	{1,9,4,1,2}							

Label index	Triangular patches											
2013	{8,10,0,0,2}	{10,6,0,0,2}	{2,8,10,2,3}	{2,10,7,2,3}	{6,10,1,1,2}	{10,7,1,1,2}	{10,9,3,0,1}	{6,10,3,0,1}	{4,7,10,1,3}	{4,10,9,1,3}	{5,10,8,0,3}	{5,9,10,0,3}
2021	{5,9,8,0,1}	{0,9,3,0,2}	{0,8,9,0,2}	{2,4,9,1,2}	{2,9,8,1,2}							
2031	{8,10,0,0,2}	{10,6,0,0,2}	{10,8,2,1,2}	{7,10,2,1,2}	{1,10,6,2,3}	{1,7,10,2,3}	{10,9,3,0,3}	{6,10,3,0,3}	{10,7,4,1,3}	{9,10,4,1,3}	{5,10,8,0,1}	{5,9,10,0,1}
2100	{8,6,0,1,2}	{6,8,1,0,2}	{8,2,1,0,2}	{8,6,3,0,1}	{5,8,3,0,1}							
2101	{6,7,1,0,2}	{7,6,0,1,2}	{2,7,0,1,2}	{3,7,6,0,1}	{3,4,7,0,1}							
2102	{9,6,3,0,1}	{0,5,9,1,2}	{0,9,6,1,2}	{1,6,9,0,2}	{1,9,4,0,2}							
2103	{8,10,0,1,2}	{10,6,0,1,2}	{2,8,10,2,3}	{2,10,7,2,3}	{6,10,1,0,2}	{10,7,1,0,2}	{3,9,10,0,1}	{3,10,6,0,1}	{4,7,10,0,3}	{4,10,9,0,3}	{5,10,8,1,3}	{5,9,10,1,3}
2110	{7,8,2,0,2}	{8,1,0,1,2}	{8,7,1,1,2}	{4,8,7,0,1}	{4,5,8,0,1}							
2120	{8,9,5,0,1}	{0,9,3,1,2}	{0,8,9,1,2}	{2,4,9,0,2}	{2,9,8,0,2}							
2130	{8,10,0,1,2}	{10,6,0,1,2}	{10,8,2,0,2}	{7,10,2,0,2}	{1,10,6,2,3}	{1,7,10,2,3}	{10,9,3,1,3}	{6,10,3,1,3}	{10,7,4,0,3}	{9,10,4,0,3}	{8,10,5,0,1}	{10,9,5,0,1}
2201	{4,7,9,0,1}	{1,3,9,0,2}	{1,9,7,0,2}	{2,9,5,1,2}	{2,7,9,1,2}							
2210	{9,7,4,0,1}	{1,3,9,1,2}	{1,9,7,1,2}	{2,9,5,0,2}	{2,7,9,0,2}							
2301	{0,10,8,2,3}	{0,6,10,2,3}	{10,8,2,1,2}	{7,10,2,1,2}	{6,10,1,0,2}	{10,7,1,0,2}	{3,9,10,0,3}	{3,10,6,0,3}	{4,7,10,0,1}	{4,10,9,0,1}	{8,10,5,1,3}	{10,9,5,1,3}
2310	{0,10,8,2,3}	{0,6,10,2,3}	{10,8,2,0,2}	{7,10,2,0,2}	{6,10,1,1,2}	{10,7,1,1,2}	{3,9,10,1,3}	{3,10,6,1,3}	{10,7,4,0,1}	{9,10,4,0,1}	{8,10,5,0,3}	{10,9,5,0,3}
3012	{8,10,0,0,3}	{10,6,0,0,3}	{10,8,2,2,3}	{7,10,2,2,3}	{6,10,1,1,3}	{10,7,1,1,3}	{10,9,3,0,1}	{6,10,3,0,1}	{4,7,10,1,2}	{4,10,9,1,2}	{5,10,8,0,2}	{5,9,10,0,2}
3021	{8,10,0,0,3}	{10,6,0,0,3}	{10,8,2,1,3}	{7,10,2,1,3}	{6,10,1,2,3}	{10,7,1,2,3}	{10,9,3,0,2}	{6,10,3,0,2}	{10,7,4,1,2}	{9,10,4,1,2}	{5,10,8,0,1}	{5,9,10,0,1}
3102	{8,10,0,1,3}	{10,6,0,1,3}	{10,8,2,2,3}	{7,10,2,2,3}	{6,10,1,0,3}	{10,7,1,0,3}	{3,9,10,0,1}	{3,10,6,0,1}	{4,7,10,0,2}	{4,10,9,0,2}	{5,10,8,1,2}	{5,9,10,1,2}
3120	{8,10,0,1,3}	{10,6,0,1,3}	{10,8,2,0,3}	{7,10,2,0,3}	{6,10,1,2,3}	{10,7,1,2,3}	{10,9,3,1,2}	{6,10,3,1,2}	{10,7,4,0,2}	{9,10,4,0,2}	{8,10,5,0,1}	{10,9,5,0,1}
3201	{8,10,0,2,3}	{10,6,0,2,3}	{10,8,2,1,3}	{7,10,2,1,3}	{6,10,1,0,3}	{10,7,1,0,3}	{3,9,10,0,2}	{3,10,6,0,2}	{4,7,10,0,1}	{4,10,9,0,1}	{8,10,5,1,2}	{10,9,5,1,2}
3210	{8,10,0,2,3}	{10,6,0,2,3}	{10,8,2,0,3}	{7,10,2,0,3}	{6,10,1,1,3}	{10,7,1,1,3}	{3,9,10,1,2}	{3,10,6,1,2}	{10,7,4,0,1}	{9,10,4,0,1}	{8,10,5,0,2}	{10,9,5,0,2}

Appendix B: Look-up table of the patch index

Label index	Patch index	Label index	Patch index	Label index	Patch index	Label index	Patch index
0000	{−1,−1}	1000	{142,142}	2000	{−1,−1}	3000	{−1,−1}
0001	{0,0}	1001	{143,144}	2001	{284,288}	3001	{−1,−1}
0002	{−1,−1}	1002	{145,149}	2002	{−1,−1}	3002	{−1,−1}
0003	{−1,−1}	1003	{−1,−1}	2003	{−1,−1}	3003	{−1,−1}
0010	{1,1}	1010	{150,151}	2010	{289,293}	3010	{−1,−1}
0011	{2,3}	1011	{152,152}	2011	{294,298}	3011	{−1,−1}
0012	{4,8}	1012	{153,157}	2012	{299,303}	3012	{416,427}
0013	{−1,−1}	1013	{−1,−1}	2013	{304,315}	3013	{−1,−1}
0020	{−1,−1}	1020	{158,162}	2020	{−1,−1}	3020	{−1,−1}
0021	{9,13}	1021	{163,167}	2021	{316,320}	3021	{428,439}
0022	{−1,−1}	1022	{168,172}	2022	{−1,−1}	3022	{−1,−1}
0023	{−1,−1}	1023	{173,184}	2023	{−1,−1}	3023	{−1,−1}
0030	{−1,−1}	1030	{−1,−1}	2030	{−1,−1}	3030	{−1,−1}
0031	{−1,−1}	1031	{−1,−1}	2031	{321,332}	3031	{−1,−1}
0032	{−1,−1}	1032	{185,196}	2032	{−1,−1}	3032	{−1,−1}
0033	{−1,−1}	1033	{−1,−1}	2033	{−1,−1}	3033	{−1,−1}
0100	{14,14}	1100	{197,198}	2100	{333,337}	3100	{−1,−1}
0101	{15,16}	1101	{199,199}	2101	{338,342}	3101	{−1,−1}
0102	{17,21}	1102	{200,204}	2102	{343,347}	3102	{440,451}
0103	{−1,−1}	1103	{−1,−1}	2103	{348,359}	3103	{−1,−1}
0110	{22,23}	1110	{205,205}	2110	{360,364}	3110	{−1,−1}
0111	{24,24}	1111	{−1,−1}	2111	{−1,−1}	3111	{−1,−1}
0112	{25,29}	1112	{−1,−1}	2112	{−1,−1}	3112	{−1,−1}
0113	{−1,−1}	1113	{−1,−1}	2113	{−1,−1}	3113	{−1,−1}
0120	{30,34}	1120	{206,210}	2120	{365,369}	3120	{452,463}
0121	{35,39}	1121	{−1,−1}	2121	{−1,−1}	3121	{−1,−1}
0122	{40,44}	1122	{−1,−1}	2122	{−1,−1}	3122	{−1,−1}
0123	{45,56}	1123	{−1,−1}	2123	{−1,−1}	3123	{−1,−1}
0130	{−1,−1}	1130	{−1,−1}	2130	{370,381}	3130	{−1,−1}

Label index	Patch index	Label index	Patch index	Label index	Patch index	Label index	Patch index
0131	{−1,−1}	1131	{−1,−1}	2131	{−1,−1}	3131	{−1,−1}
0132	{57,68}	1132	{−1,−1}	2132	{−1,−1}	3132	{−1,−1}
0133	{−1,−1}	1133	{−1,−1}	2133	{−1,−1}	3133	{−1,−1}
0200	{−1,−1}	1200	{211,215}	2200	{−1,−1}	3200	{−1,−1}
0201	{69,73}	1201	{216,220}	2201	{382,386}	3201	{464,475}
0202	{−1,−1}	1202	{221,225}	2202	{−1,−1}	3202	{−1,−1}
0203	{−1,−1}	1203	{226,237}	2203	{−1,−1}	3203	{−1,−1}
0210	{74,78}	1210	{238,242}	2210	{387,391}	3210	{476,487}
0211	{79,83}	1211	{−1,−1}	2211	{−1,−1}	3211	{−1,−1}
0212	{84,88}	1212	{−1,−1}	2212	{−1,−1}	3212	{−1,−1}
0213	{89,100}	1213	{−1,−1}	2213	{−1,−1}	3213	{−1,−1}
0220	{−1,−1}	1220	{243,247}	2220	{−1,−1}	3220	{−1,−1}
0221	{101,105}	1221	{−1,−1}	2221	{−1,−1}	3221	{−1,−1}
0222	{−1,−1}	1222	{−1,−1}	2222	{−1,−1}	3222	{−1,−1}
0223	{−1,−1}	1223	{−1,−1}	2223	{−1,−1}	3223	{−1,−1}
0230	{−1,−1}	1230	{248,259}	2230	{−1,−1}	3230	{−1,−1}
0231	{106,117}	1231	{−1,−1}	2231	{−1,−1}	3231	{−1,−1}
0232	{−1,−1}	1232	{−1,−1}	2232	{−1,−1}	3232	{−1,−1}
0233	{−1,−1}	1233	{−1,−1}	2233	{−1,−1}	3233	{−1,−1}
0300	{−1,−1}	1300	{−1,−1}	2300	{−1,−1}	3300	{−1,−1}
0301	{−1,−1}	1301	{−1,−1}	2301	{392,403}	3301	{−1,−1}
0302	{−1,−1}	1302	{260,271}	2302	{−1,−1}	3302	{−1,−1}
0303	{−1,−1}	1303	{−1,−1}	2303	{−1,−1}	3303	{−1,−1}
0310	{−1,−1}	1310	{−1,−1}	2310	{404,415}	3310	{−1,−1}
0311	{−1,−1}	1311	{−1,−1}	2311	{−1,−1}	3311	{−1,−1}
0312	{118,129}	1312	{−1,−1}	2312	{−1,−1}	3312	{−1,−1}
0313	{−1,−1}	1313	{−1,−1}	2313	{−1,−1}	3313	{−1,−1}
0320	{−1,−1}	1320	{272,283}	2320	{−1,−1}	3320	{−1,−1}
0321	{130,141}	1321	{−1,−1}	2321	{−1,−1}	3321	{−1,−1}
0322	{−1,−1}	1322	{−1,−1}	2322	{−1,−1}	3322	{−1,−1}
0323	{−1,−1}	1323	{−1,−1}	2323	{−1,−1}	3323	{−1,−1}
0330	{−1,−1}	1330	{−1,−1}	2330	{−1,−1}	3330	{−1,−1}

Label index	Patch index	Label index	Patch index	Label index	Patch index	Label index	Patch index
0331	{−1,−1}	1331	{−1,−1}	2331	{−1,−1}	3331	{−1,−1}
0332	{−1,−1}	1332	{−1,−1}	2332	{−1,−1}	3332	{−1,−1}
0333	{−1,−1}	1333	{−1,−1}	2333	{−1,−1}	3333	{−1,−1}

References

Bagley B, Sastry SP, Whitaker RT (2016) A marching-tetrahedra algorithm for feature-preserving meshing of piecewise-smooth implicit surfaces. Procedia Eng 163:162–174

Boissonnat JD, Dyer R, Ghosh A (2013) The stability of Delaunay triangulations. Int J Comput Geom Appl 23(4–5):303–333. https://doi.org/10.1142/s0218195913600078

Boisvert JB, Manchuk JG, Deutsch CV (2009) Kriging in the presence of locally varying anisotropy using non-Euclidean distances. Math Geosci 41(5):585–601. https://doi.org/10.1007/s11004-009-9229-1

Boltcheva D, Yvinec M, Boissonnat J-D (2009a) Feature preserving Delaunay mesh generation from 3D multi-material images. Comput Graph Forum 28(5):1455–1464. https://doi.org/10.1111/j.1467-8659.2009.01522.x

Boltcheva D, Yvinec M, Boissonnat JD (2009b) Mesh generation from 3D multi-material images. In: Yang GZ, Hawkes D, Rueckert D, Noble A, Taylor C (eds) 12th International conference on medical image computing and computer-assisted intervention (MICCAI2009b), vol 5762. Springer, Berlin, Heidelberg, pp 283–290. https://doi.org/10.1007/978-3-642-04271-3_35

Calakli F, Taubin G (2011) SSD: smooth signed distance surface reconstruction. Comput Graph Forum 30(7):1993–2002. https://doi.org/10.1111/j.1467-8659.2011.02058.x

Chan SL (1998) A new tetrahedral tesselation scheme for isosurface generation. Comput Graph 22(1):83–90. https://doi.org/10.1016/S0097-8493(97)00085-X

Che DF, Jia QR (2019) Three-dimensional geological modeling of coal seams using weighted kriging method and multi-source data. IEEE Access 7:118037–118045. https://doi.org/10.1109/Access.2019.2936811

Chen L, Xu JC (2004) Optimal Delaunay triangulations. J Comput Math 22(2):299–308

Cheng TP (2013) Accelerating universal kriging interpolation algorithm using CUDA-enabled GPU. Comput Geosci 54:178–183. https://doi.org/10.1016/j.cageo.2012.11.013

Ciznicki M, Kierzynka M, Kurowski K, Ludwiczak B, Napierala K, Palczynski J (2012) Efficient isosurface extraction using marching tetrahedra and histogram pyramids on multiple GPUs. Parallel Process Appl Math II 7204:343–352

Cong A, Liu Y, Kumar D, Cong W, Wang G (2005) Geometrical modeling using multiregional marching tetrahedra for bioluminescence tomography. In: Proc SPIE 5744, Medical Imaging 2005: visualization, image-guided procedures, and display, Parts 1 and 2, vol 5744. pp 756–763. https://doi.org/10.1117/12.595403

Corliss G (1977) Which root does the bisection algorithm find? SIAM Rev 19(2):325–327. https://doi.org/10.1137/1019044

Cuomo S, Galletti A, Giunta G, Marcellino L (2017) Reconstruction of implicit curves and surfaces via RBF interpolation. Appl Numer Math 116:157–171. https://doi.org/10.1016/j.apnum.2016.10.016

Fayolle PA, Pasko A (2012) Optimized surface discretization of functionally defined multi-material objects. Adv Eng Softw 45(1):301–312. https://doi.org/10.1016/j.advengsoft.2011.10.007

Frank T, Tertois AL, Mallet JL (2007) 3D-reconstruction of complex geological interfaces from irregularly distributed and noisy point data. Comput Geosci 33(7):932–943. https://doi.org/10.1016/j.cageo.2006.11.014

Franke R (1982) Scattered data interpolation: tests of some methods. Math Comput 38(157):181–200. https://doi.org/10.1090/s0025-5718-1982-0637296-4

Gueziec A, Hummel R (1995) Exploiting triangulated surface extraction using tetrahedral decomposition. IEEE Trans Visual Comput Graph 1(4):328–342. https://doi.org/10.1109/2945.485620

Guo D, Li CQ, Wu L, Yang JZ (2017) Improved marching tetrahedra algorithm based on hierarchical signed distance field and multi-scale depth map fusion for 3D reconstruction. J Vis Commun Image Represent 48:491–501. https://doi.org/10.1016/j.jvcir.2016.12.016

Guo JT, Wang XL, Wang JM, Dai XW, Wu LX, Li CL, Li FD, Liu SJ, Jessell MW (2021) Three-dimensional geological modeling and spatial analysis from geotechnical borehole data using an implicit surface and marching tetrahedra algorithm. Eng Geol 284:106047. https://doi.org/10.1016/j.enggeo.2021.106047

Hartmann E (1998) A marching method for the triangulation of surfaces. Vis Comput 14(3):95–108. https://doi.org/10.1007/s003710050126

Hillier MJ, Schetselaar EM, de Kemp EA, Perron G (2014) Three-dimensional modelling of geological surfaces using generalized interpolation with radial basis functions. Math Geosci 46(8):931–953. https://doi.org/10.1007/s11004-014-9540-3

Honnell PM (1953) A new universal right-hand rule. Electr Eng 72(4):346–349. https://doi.org/10.1109/ee.1953.6438612

Kalra D, Barr AH (1989) Guaranteed ray intersections with implicit surfaces. ACM SIGGRAPH Comput Graph 23(3):297–306. https://doi.org/10.1145/74334.74364

Liu Z, Zhang ZL, Zhou CY, Ming WH, Du ZC (2021) An adaptive inverse-distance weighting interpolation method considering spatial differentiation in 3D geological modeling. Geosciences 11(2):51. https://doi.org/10.3390/geosciences11020051

Macêdo I, Gois JP, Velho L (2009) Hermite interpolation of implicit surfaces with radial basis functions. In: 2009 XXII Brazilian symposium on computer graphics and image processing. Rio de Janeiro, Brazil, pp 1–8. https://doi.org/10.1109/Sibgrapi.2009.11

Mallet JL (1992) Discrete smooth interpolation in geometric modelling. Comput Aided Des 24(4):178–191. https://doi.org/10.1016/0010-4485(92)90054-E

Marinoni O (2003) Improving geological models using a combined ordinary-indicator kriging approach. Eng Geol 69(1–2):37–45. https://doi.org/10.1016/S0013-7952(02)00246-6

Masala GL, Golosio B, Oliva P (2013) An improved marching cube algorithm for 3D data segmentation. Comput Phys Commun 184(3):777–782. https://doi.org/10.1016/j.cpc.2012.09.030

Newman TS, Yi H (2006) A survey of the marching cubes algorithm [review]. Comput Graph 30(5):854–879. https://doi.org/10.1016/j.cag.2006.07.021

Raman S, Wenger R (2008) Quality isosurface mesh generation using an extended marching cubes lookup table. Comput Graph Forum 27(3):791–798. https://doi.org/10.1111/j.1467-8659.2008.01209.x

Treece GM, Prager RW, Gee AH (1999) Regularised marching tetrahedra: improved iso-surface extraction [article]. Comput Graph 23(4):583–598. https://doi.org/10.1016/s0097-8493(99)00076-x

Wu Z, Sullivan JM (2003) Multiple material marching cubes algorithm. Int J Numer Meth Eng 58(2):189–207. https://doi.org/10.1002/nme.775

Xiao X, Xie LW, Tang RY, Liu JA, Song P, Zhu XY, Zhao JL, Jiang C, Yang S, Wu P (2022) Improved compressive properties of lattice structure based on an implicit surface hybrid optimization design method via selective laser melting. Metals 12(9):1477. https://doi.org/10.3390/met12091477

Yao GM, Duo J, Chen CS, Shen LH (2015) Implicit local radial basis function interpolations based on function values. Appl Math Comput 265:91–107. https://doi.org/10.1016/j.amc.2015.04.107

Zhong DY, Wang LG, Bi L, Jia MT (2019) Implicit modeling of complex orebody with constraints of geological rules. Trans Nonferr Met Soc China 29(11):2392–2399. https://doi.org/10.1016/S1003-6326(19)65145-9

Zhong DY, Li BY, Shi TD, Li ZP, Wang LG, Bi L (2021) Repair of voids in multi-labeled triangular mesh. Appl Sci 11(19):9275. https://doi.org/10.3390/app11199275