Article

Repair of Geological Models Based on Multiple Material Marching Cubes

Benyu Li ^{1,2}, Deyun Zhong ^{1,2,*} and Liguan Wang ^{1,2}

¹ School of Resources and Safety Engineering, Central South University, Changsha 410083, China; 195512077@csu.edu.cn (B.L.); liguan_wang@csu.edu.cn (L.W.)

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

* Correspondence: deyizhiyun@csu.edu.cn

Abstract: In this paper, we present a multi-domain implicit surface reconstruction algorithm for geological modeling based on the labeling of voxel points. The improved algorithm sets a label for each voxel point to represent the type of its geological domain and then obtains all the voxel points in the void areas. After that, the improved algorithm modifies the labels of the voxel points in the void areas and finally reconstructs the geological models through the Multiple Material Marching Cubes (M3C) algorithm. The improved algorithm solves the problems of some unexpected overlaps and voids in geological modeling by setting and modifying the labels of the voxel points. Our key contribution is proposing a labeling processing method to repair the overlap and void defects generated in the geological modeling and realizing the improved M3C algorithm. The experimental results of some geological models show the performance of the improved method. Compared with the original method, the improved method can repair the overlap and void defects in geological modeling to ensure the raw structural adjacency relationships of the geological bodies.

Citation: Li, B.; Zhong, D.; Wang, L. Repair of Geological Models Based on Multiple Material Marching Cubes. *Mathematics* **2021**, *9*, 2207. <https://doi.org/10.3390/math9182207>

Academic Editor: Akemi Galvez Tomida

Received: 3 August 2021

Accepted: 6 September 2021

Published: 8 September 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: geological modeling; implicit surface reconstruction; marching cubes; multiple material marching cubes; model repair

1. Introduction

Using implicit functions to reconstruct smooth surfaces from given scattered data has been used in many disciplines, such as geology, mathematics, computer graphics, mining engineering, and biomedicine. In addition, the Marching Cubes (MC) [1] algorithm is a simple and most widely used implicit surface reconstruction algorithm. However, the original MC algorithm is usually applied to reconstruct a single domain model. When there is more than one domain to be modeled, multi-domain modeling technology is required. As an improvement of the MC algorithm, the M3C [2] algorithm can realize the multi-domain reconstruction of the model surface.

When geological modeling involves multiple geological models, the adjacency relationships between different models need to be considered. For the implicit modeling of the geological bodies, generally, one formation corresponds to an implicit function. Therefore, for multi-labeled implicit functions that represent multiple formations, the defects produced in the multi-domain implicit surface reconstruction need to be considered. If there are no overlaps or voids in the formation models represented by the multi-labeled implicit functions, the M3C algorithm can be used directly, and the reconstructed formation models can maintain the raw structural adjacency relationships. However, if overlaps and voids exist between the formation models represented by the multi-labeled implicit functions, overlaps and voids may occur when the model is reconstructed directly by the M3C algorithm.

If the multi-domain geological bodies are reconstructed one by one using the traditional single domain iso-surface extraction method, the common geological interfaces will be reconstructed many times, which leads to inconsistent triangulated meshes. Similarly, if

a part of the common geological interface of two formations is defined by two different implicit functions, it may also lead to inconsistent interpolation results. This leads to the fact that in practical applications, the surfaces of multiple geological models cannot maintain the raw structural adjacency relationships; that is, overlaps and voids between the models may occur. The defects of overlaps and voids affect the accuracy of the models, and the reconstructed models do not conform to the raw structural adjacency relationships. Therefore, it is necessary to deal with this problem arising in the multi-domain implicit modeling of the geological bodies.

To solve the problem, we propose a multi-domain reconstruction algorithm based on the labeling of the voxel points. The improved algorithm sets a label for each voxel point in the study area to represent the type of its geological domain. Moreover, by setting labels, the overlaps can be repaired. In addition, the improved algorithm searches for the voxel points of all void areas and repairs the voids by modifying the labels of the void points in the reconstruction of multi-domain models. When the voxel labels are set and repaired, the M3C algorithm is used to reconstruct the models to maintain the raw structural adjacency relationships.

Our key contribution is proposing a multi-domain reconstruction algorithm based on the labeling of the voxel points. The improved algorithm can repair the defects of overlaps and voids between the multi-domain models in the geological modeling. At the same time, this paper is based on the M3C algorithm and improves it so that the reconstructed multi-domain models of the geological bodies conform to the raw structural adjacency relationships. The improved algorithm is simple in logic and easy to implement. Moreover, it provides a new idea for the reconstruction of the implicit surface of the multi-domain models.

Related Works

Houlding [3] first proposed the concept of 3D geological modeling in 1993. Since then, the theory of 3D geological modeling has been continuously developed. Three-dimensional modeling methods are roughly divided into two categories, which are explicit modeling and implicit modeling [4,5]. In the explicit modeling, the contour lines are drawn manually from geological exploration data, and then the contour lines are spliced to realize a 3D reconstruction of the geological bodies [6–8]. However, in the explicit modeling, the model cannot be updated automatically. While the implicit modeling [9,10] usually uses the implicit functions to express the geological bodies, which generates geological models automatically. Implicit modeling has gradually become a research hotspot because of its high automation. In fact, the two approaches are not opposed to each other, especially in the complex geological environment; the two methods are often used together to construct the model [11].

There are many interpolation function theories. Commonly used are the inverse distance weighted (IDW) interpolation method [12,13], the Kriging interpolation method [14–16], the moving least squares (MLS) interpolation method [17–19], and the radial basis functions interpolation method [20–22]. To compare the performance of interpolation functions, Franke [23] compared dozens of different interpolation algorithms in terms of parameters and modeling accuracy. Finally, he concluded that the radial basis functions have a better fitting effect. In this paper, we will take the radial basis functions as an example to implement the improved algorithm.

Lorensen and Cline first introduced the MC algorithm [1] in 1987, which quickly became a popular method in surface reconstruction due to its simplicity and efficiency. Subsequently, many scholars have improved the MC algorithm in terms of ambiguity, approximation accuracy, and processing efficiency. Dürst [24] pointed out that the algorithm has the problem of ambiguity on the surface, and there are many solutions to the problem, such as the marching tetrahedra method [25] and Nielson's asymptotic method [26]. Natarajan [27] pointed out that even if there is no surface ambiguity, there might be ambiguity inside the voxel, which is called volume ambiguity. Natarajan introduced the concept

of saddle points to deal with the volume ambiguity, while Chernyaev [28] proposed the generalized asymptote method to solve the problem. Many scholars have improved the approximation accuracy of the MC algorithm. For example, Lopes and Brodie [29] proposed the method of adding internal points, and Cignoni et al. [30] proposed the progressive grid extraction method. The methods to improve the processing efficiency of the MC algorithm have been proposed by many scholars, such as the octree method [31] of Wilhelms and Vangelder and the progressive growth method [32] of Bajaj et al., which increase the efficiency by shortening the processing time. In contrast, Schroeder's decimation algorithm [33] improves the processing efficiency by reducing storage redundancy.

However, the MC algorithm and its improved algorithm can only reconstruct a single domain model, and the reconstruction of the multi-domain models requires the use of multi-domain reconstruction technology. Faraj et al. [34] proposed an iterative remeshing algorithm for multi-material tetrahedral meshes, which reconstructs the multi-material interfaces by developing an intermediate implicit feature complex structure. Based on the octree method, Zhang and Qian [35] use an indicator variable to identify the internal topological structure of each octree cell and resolve the problem of topology ambiguity in multi-domain reconstruction. Wu and Sullivan [2] proposed the M3C algorithm to extend the MC algorithm to a multi-domain reconstruction algorithm. Based on the M3C algorithm, we improve the M3C algorithm so that the reconstructed multi-domain models conform to the raw structural adjacency relationships of the geological bodies.

2. Overview of the Method

To reconstruct the implicit surface of the multi-labeled implicit functions, the M3C algorithm further summarizes all the basic configurations of the cube state (the connection relationship of the triangular patches) based on the MC algorithm. Finally, the M3C algorithm introduces some additional nodes to generate a topologically consistent multi-labeled triangular mesh to deal with the adjacency relationships between multiple subdomains in a single cube. Thus, the method provides a way to realize the multi-domain implicit surface reconstruction of the models. Based on the M3C algorithm, we intend to improve the reconstruction method of the multi-domain models and repair the multi-domain models by modifying the labels of voxel points. In this way, the raw structural adjacency relationships are maintained well between the multi-domain models, and there will be no overlaps or voids between the sub-domain models.

As shown in Figure 1a, there are overlaps and a void between the red and green models reconstructed by the original M3C algorithm, and the modeling priority of the red model is higher than the green. As shown in Figure 1a, the voxel points are divided into five categories, and we use different colors to distinguish different types of voxel points. Among them, the yellow and purple voxel points are ambiguous points; to be precise, the yellow points are overlapping points, and the purple points are void points. As shown in Figure 1b, the labels of the ambiguous voxel points are modified. Then the models are reconstructed using the M3C algorithm, and there will be no overlaps and voids between the reconstructed models.

Figure 1. Schematic diagram of model repair. (a) Original voxel points. (b) Repaired voxel points.

The specific methods of modifying the labels of overlapping points and void points are different. As shown in Figure 1, for the overlapping voxel points (yellow points), modifying the labels of the overlapping points as the label of the model (red model) has the highest priority. While for the void voxel points (purple points), modifying the labels of the void points as the label of the model (green model) has the lowest priority. When the labels of the voxel points are modified, there are no overlaps and voids between the two models, the reconstructed models are shown in Figure 1b.

Based on the principles above, the improved algorithm is designed. The algorithm flow chart is shown in Figure 2:

Figure 2. Overall flow chart.

Step 1: Interpolate the geological constraints constructed from the given geological sampling data and obtain the interpolated implicit functions based on the radial basis function interpolation method.

Step 2: Set a label for each voxel point to represent the type of its geological domain. Through the labeling of the voxel points, the overlapping defects between the multi-domain models are repaired.

Step 3: Search for void areas. Find the void areas, store the points of the voids, and store the labels of the geological models adjacent to the voids.

Step 4: Modify the labels of the void points. Modify the labels of the void points as the label of a specific geological model adjacent to the void.

Step 5: Use the M3C algorithm to reconstruct the multi-domain geological models.

3. The Multi-Domain Implicit Surface Reconstruction Method

3.1. Interpolate Multi-Labeled Implicit Functions

In the improved algorithm, geological bodies are placed in a voxelized space with a uniform size. To use the implicit functions to interpolate geological bodies, we establish a 3D coordinate system in the voxelized space, and each voxel point has an exact coordinate.

The geological body has a specific shape and volume, and a maximum bounding box will be artificially specified in the coordinate system during the modeling. The maximum bounding box is a 3D space containing all the geological models, in which all voxel points will participate in the algorithm. If the bounding box is too large, calculation time will be wasted; otherwise, if it is too small, part of the geological body will be missed. Users can adjust the parameters to obtain an appropriate maximum bounding box.

As shown in Figure 3, the areas in the maximum bounding box are divided into four types. The exterior area is outside the geological bodies and adjacent to the maximum

bounding box. The interior area is just inside a specific geological body (including on the surface), and the overlapping area is inside multiple geological bodies. At the same time, the void area is outside the geological bodies and is wrapped by the geological bodies.

Figure 3. According to the inside and outside relationships of the models, it can be divided into internal and external areas, and according to the ambiguity, it can be divided into overlapping and void areas.

The geological domain where the geological bodies are located is regarded as a signed implicit function field. In general, the improved algorithm agrees that the function value of the point outside the geological body is positive; conversely, the value of the point inside the geological body is negative. The relationships between the geological domain where a single formation is located and the sign of the implicit function $f(x)$ can be expressed as [36]:

$$f(x) = \text{dist}(x, x') = \begin{cases} = 0, & x \text{ is just on the interface} \\ > 0, & x \text{ is outside the geological body} \\ < 0, & x \text{ is inside the geological body} \end{cases}$$

where x is an arbitrary point in the geological domain, x' is the closest point of x on the implicit surface S , and $\text{dist}(x, x')$ is the signed distance between x and x' .

For the geological model containing l formations, a multi-labeled implicit function $F(x)$ composed of some sub-functions $f_i(x)$, $i = \{1, 2, \dots, l\}$ representing l formations can be used to express the domain type of each formation. The subscript i is used to distinguish different formations.

The functions $f_i(x)$ are the interpolation functions of l formations. The subscript i represents the modeling priority. A higher priority of the formation model corresponds to a smaller i . i starts from 1 and increases sequentially. The function $F(x)$ is used for labeling the voxel points, that is the value of $F(x)$ represents the label of the point x . The relationships between $F(x)$ and $f(x)$ can be expressed as:

$$F(x) = \min(\tilde{f}_i(x))$$

where

$$\tilde{f}_i(x) = \begin{cases} i, & f_i(x) \leq 0 \\ \infty, & f_i(x) > 0 \end{cases}$$

The label of the voxel point x is determined by the value of $F(x)$. In this paper, we agreed that any one voxel point belongs to a kind of formation. Therefore, the value of $F(x)$ corresponds to a unique label. In the overlapping area, for the point x , there are multiple stratigraphic functions with $f_i(x) \leq 0$, and thus there are multiple values of $\tilde{f}_i(x)$, but $F(x)$ takes the smallest one. While in the void area and the exterior area, the point x does not belong to any formations and the value of $F(x)$ is ∞ .

In this paper, the radial basis function is taken as an example to model the geological bodies implicitly. After using the radial basis function interpolation for the geological bodies, a set of multi-labeled implicit functions representing the domain types of the geological bodies are obtained; and the labels of the set of implicit functions have a one-to-one correspondence with the labels of the geological bodies. Then the sign of the corresponding function values can judge the internal and external relationships between the voxel points and the geological models directly.

3.2. Label the Voxel Points

The label is an essential concept of the improved algorithm. In this paper, the labels of geological bodies, interpolation functions, and voxel points are consistent. In fact, the labeling of voxel points is to first “divide” the geological bodies into voxel points and then set labels for the voxel points so that these points can be “combined” into geological bodies later. When the models are reconstructed by the M3C algorithm, the voxel points with the same label are reconstructed into the same geological body. Therefore, by modifying the labels of the ambiguous voxel points as the label of a specific geological body, the overlaps and voids can be repaired.

The priority of geological bodies in multi-domain modeling is different. The reason for the difference is related to the reliability of obtaining the geological data. The reliability is determined by geological engineers according to prior geological rules; thus it generally conforms to the actual situation. The higher the reliability of the geological model is, the higher the priority of the geological model is. Thus, if the priority of a certain geological body is higher than others, the integrity of the geological body with a higher priority should be considered in priority.

According to the sign of the function value of the voxel point p , the internal and external relationships between p and the geological bodies can be judged; and according to the value of the multi-labeled function $F(p)$, a label can be set for the voxel point p . If the voxel point p is in a geological body (including on the surface), then a label consistent with the label of the geological body is set for p . On the other hand, if the voxel point p is outside the geological bodies (p is in the exterior area or void area), then set the label of p to ∞ . If the voxel point p is inside multiple geological bodies, the label of the geological body with the highest modeling priority among these geological bodies is used as the label of p .

As shown in Figure 4, the general method to set labels for voxel points is shown. In Figure 4, the priority of the red model is higher than the green. Therefore, according to the priority, the label of the red model is set to 1, and the label of the green model is set to 2.

Figure 4. Set the labels of voxel points. (a) Labels of the general voxel points. (b) Labels of the ambiguous voxel points.

As shown in Figure 4a, the label of the red model is 1, and its interpolation function is f_1 , and the labels of all voxel points inside the red model are set to 1; the setting method of the labels for the green model is similar to the red. The labels of the voxel points in the exterior area are set to ∞ . Generally speaking, the labels of the voxel points correspond to both the labels of the geological bodies and the subscripts of the interpolation functions.

As shown in Figure 4b, for the overlapping points, the labels have been set to the label of the geological body with the highest modeling priority. For the void points, the labels have been set temporarily to ∞ .

Setting labels for the voxel points in this way can repair the overlapping defects of the multi-domain model. As shown in Figure 5, in the labeling of the voxel points, the labels of the overlapping points are set to the label of the geological body with the highest priority. As a result, the overlapping area will be classified as the geological body with the highest priority after reconstruction with the M3C algorithm. In this way, the approach ensures the integrity of the geological model with higher priority. As shown in Figure 5b,c, different modeling results are shown when the priority is reversed.

Figure 5. Strategies for repairing overlapping defects. (a) Original two geological bodies. (b) Red model with a higher priority. (c) Green model with a higher priority.

The pseudo-code of the algorithm flow for labeling the voxel points is shown in Algorithm 1 (label the voxel points).

Algorithm 1: Label the voxel points

3.3. Search for Void Voxel Points

The labeling process sets labels for all the voxel points, which repairs the defects of overlaps. However, the voids may still exist, which does not conform to the geological rules. We need to repair the voids.

3.3.1. Search for Seed Points

To find all the void areas, we need to find the seed points first. The seed point x is the first point in the void area or the exterior area obtained in the traversal process. The reason why the first point of the void area or the exterior area is selected as the seed point is that all the points inside the void area and the exterior area have the same label ∞ . The algorithm can only search out the void area and the exterior area together based on the

label ∞ but cannot find the void area separately. If a seed point is found, the corresponding area containing the seed point can be found directly by traversing. The specific steps are as follows:

- (1) The initial state of all the voxel points is set as unvisited. Assuming that a point p is obtained in the traversal process, then the state of p is modified to visited. We define a variable $Mark_p$ to represent the visited state of point p . If $Mark_p = 0$, then point p is unvisited; if $Mark_p = 1$, then p is visited.
- (2) The traversal process starts from the first voxel points in the maximum bounding box. If a point p has been visited in the traversal process, then skip p .
- (3) Before proceeding to the specific steps, we first define two arrays V_k , $L_{a,k}$, and a queue Q_k . V_k is used to store the quasi-void points, including the exterior points and the void points. $L_{a,k}$ is used to store the labels of the voids and the geological bodies adjacent to the quasi-void area. Q_k is used to perform a greedy search and obtain the quasi-void points based on the seed points.

For the unvisited point p , modify the state of p as visited, then judge whether the label of p is ∞ .

If the label of p is ∞ , store p in the array V_k ($k = 1, 2, 3 \dots$) and store the label of p in the array $L_{a,k}$ ($k = 1, 2, 3 \dots$). At the same time, p is selected as a seed point and stored in the queue Q_k ($k = 1, 2, 3 \dots$). The k here is used to record the quasi-void areas, and each quasi-void area corresponds to a specific k .

Otherwise, if the label of p is not ∞ , then skip p .

3.3.2. Search for Void Voxel Points

In this paper, a greedy search strategy is used to obtain the quasi-void areas. When the seed point is found, the corresponding quasi-void area will also be found. Generally, a quasi-void area corresponds to a specific seed point. However, the seed point is the first point with a label ∞ obtained in the traversal process. Therefore, the seed point may correspond to the void area or the exterior area; we need to distinguish the two areas.

First, the improved algorithm needs to find the seed point in the process of traversal. For the seed point p , obtain six points adjacent to p (top, bottom, left, right, front, and back). For the six adjacent points, take the points with the label ∞ as search points, and continue to obtain the six adjacent points of each search point. In the process of obtaining the six adjacent points of each search point, the points with the label ∞ among the six adjacent points are regarded as new search points. The process is to continuously obtain new search points and then continuously obtain six points adjacent to each new search point until no new search points can be obtained. At the same time, record all the search points in the process, they are the points in the quasi-void areas. Furthermore, record the labels that are not ∞ in the process, as they are the labels of the geological bodies that are adjacent to the quasi-voids. The strategy for obtaining the search points is shown in Figure 6.

Figure 6. Strategy for obtaining the search points. (a) Obtain the seed point. (b) Obtain the adjacent points of the seed point and find search points. (c) Obtain the adjacent points of each search point and find new search points.

After finding a quasi-void area, continue to search for the next seed point. In the process of searching for new seed points, skip the points obtained in the last search for the last quasi-void area. After finding a new seed point, obtain a new quasi-void area in the same way. The search for seed points and the search for the corresponding quasi-void areas are repeated until all quasi-void areas are obtained.

Based on the characteristics of the above process, the queue Q_k is used to store the search points. In the improved algorithm, the seed point is pushed into Q_k as the head element; and in subsequent operations, other search points will be pushed to the end of Q_k . The elements in Q_k are dynamic and sequential, the traversal order is from the head to the end of Q_k in turn. Each time the head element of Q_k will be searched for its six adjacent points (top, bottom, left, right, front, and back), then the element at the head of Q_k is popped from Q_k , and the new search points are pushed to the end of Q_k in the order of obtaining them. As long as there are elements in Q_k , the search for six points adjacent to the head element will continue until Q_k is empty. Because the areas inside the maximum bounding box are limited, the searching process will not go on indefinitely. When a quasi-void area is searched out, the elements in Q_k will be empty, and the corresponding search operation will be terminated.

The specific steps are as follows:

- (1) For the head element p' of Q_k (the first head element is the seed point), obtain its six adjacent points $(\tilde{p}_1, \tilde{p}_2, \tilde{p}_3, \tilde{p}_4, \tilde{p}_5, \tilde{p}_6)$, for each point $\tilde{p}_i (i = 1, 2, 3, 4, 5, 6)$, judge whether $\tilde{p}_i$ exceeds the maximum bounding box.
- (2) For the point $\tilde{p}_i$ that exceeds the maximum bounding box, store 0 in $L_{a,k}$. The purpose of this step is to determine whether the quasi-void area is a void area or an exterior area. The exterior area is adjacent to the maximum bounding box, so if there is a point of $\tilde{p}_i$ that exceeds the maximum bounding box, it means that the quasi-void area is an exterior area rather than a void area.
- (3) For the point $\tilde{p}_i$ that does not exceed the maximum bounding box, perform the following steps.

For the unvisited point $\tilde{p}_i$, modify the state of $\tilde{p}_i$ as visited. If the label of $\tilde{p}_i$ is ∞ , store $\tilde{p}_i$ in V_k and push $\tilde{p}_i$ to the end of Q_k in turn. This is to store the point $\tilde{p}_i$ and use $\tilde{p}_i$ as a new search point. If the label of $\tilde{p}_i$ is not ∞ , it means that $\tilde{p}_i$ is inside the geological body adjacent to the quasi-void area. The label of $\tilde{p}_i$ is stored in $L_{a,k}$, the purpose is to record the label of the geological body adjacent to the quasi-void area.

For the visited point $\tilde{p}_i$, if the label of $\tilde{p}_i$ is ∞ , then skip $\tilde{p}_i$. If the label of $\tilde{p}_i$ is not ∞ , then the label of $\tilde{p}_i$ is stored in $L_{a,k}$. The purpose of this is to obtain the labels of the geological bodies adjacent to the quasi-void areas.

- (4) After searching the six adjacent points of the head element p' in Q_k , pop p' from Q_k .
- (5) Perform steps (1), (2), (3), and (4) for the new head element of Q_k . Repeat the process until the elements in Q_k are empty. At this time, the points inside the quasi-void area will be obtained.
- (6) Continue to search for the next seed point, and perform steps (1), (2), (3), (4), and (5) in sequence until no new seed points can be found. At this time, the search for all quasi-void areas is completed.

The pseudo-code of the algorithm flow for searching for the quasi-void voxel points is shown in Algorithm 2 (search for the void voxel points).

Algorithm 2: Search for the Void Voxel Points

```

1: Input: Voxel points  $P_{vox}$  with labels
2: Output: The quasi-voids  $V_k$ , the label arrays  $L_{a,k}$ 
3: Initialize a queue  $Q_k$  to perform the greedy search strategy; Initialize an array  $V_k$ 
   to store the quasi-void points; Initialize an array  $L_{a,k}$  to store the labels;  $k=0$ .
4: foreach  $p \in P_{vox}$  do
5:     if  $p$  is unvisited ( $Mark_p = 0$ ) then
6:         Change the state of  $p$  to be visited ( $Mark_p = 1$ ).
7:          $k \leftarrow k + 1$ .
8:         if  $L_p = \infty$  then
9:              $p$  is selected as a seed point. Store  $p$  in  $V_k$ . And store
                $L_p$  in  $L_{a,k}$ . Push  $p$  into  $Q_k$ .
10:            do
11:                Obtain the head element  $p'$  of  $Q_k$ , and obtain the 6
                   points  $\tilde{p}_i(\tilde{p}_1, \tilde{p}_2, \tilde{p}_3, \tilde{p}_4, \tilde{p}_5, \tilde{p}_6)$  adjacent to  $p'$ .
12:                for  $i=1$  to 6 do
13:                    if  $\tilde{p}_i$  exceeds the maximum bounding box then
14:                        Store 0 in  $L_{a,k}$ .
15:                    else
16:                        if  $\tilde{p}_i$  is unvisited ( $Mark_{\tilde{p}_i} = 0$ ) then
17:                            Change the state of  $\tilde{p}_i$  to be visited
                               ( $Mark_{\tilde{p}_i} = 1$ ).
18:                            if  $L_{\tilde{p}_i} = \infty$  then
19:                                Store  $\tilde{p}_i$  in  $V_k$ , and push
                                    $\tilde{p}_i$  into  $Q_k$ .
20:                            else
21:                                Store the label of  $\tilde{p}_i$  in  $L_{a,k}$ .
22:                            end if
23:                        else
24:                            if  $L_{\tilde{p}_i} \neq \infty$  then
25:                                Store the label of  $\tilde{p}_i$  in  $L_{a,k}$ .
26:                            end if
27:                        end if
28:                    end if
29:                end for
30:                Pop  $p'$  from  $Q_k$ .
31:            until  $Q_k$  is empty
32:        end if
33:    end if
34: end for

```

3.4. Modify Voxel Point Labels

When the whole searching process is over, the elements in V_k are points of the quasi-void areas; $L_{a,k}$ stores the labels, including the label 0, ∞ and the labels of the geological bodies adjacent to the quasi-void areas. If there is a label 0 in $L_{a,k}$, it indicates that the

corresponding quasi-void area is the external area rather than the void area. The labels of the points in the external area do not need to be modified, while the labels of the points in the void area need to be modified.

The modification strategy is to modify the labels of the void points as the labels of the lowest priority of the geological body adjacent to the voids. It ensures the integrity of the geological body with the highest priority. The void areas are classified as part of the geological bodies with the lowest priority so that the voids have an exact attribution.

As shown in Figure 7, the priority of the red model is higher than the green, so the label of the red geological body is 1, and the label of the green geological body is 2. In Figure 7a, the labels of void points are ∞ , while in Figure 7b, the labels of the void points are modified as 2, which is the label with the lowest priority in the area adjacent to the void. When the labels are modified, then the M3C algorithm is used to reconstruct the models directly. The repaired models are shown in Figure 7c. As we can see, the void has been repaired.

Figure 7. Modify voxel point labels. (a) Labels of the void points. (b) Modify the labels of the void points. (c) Reconstruct the models.

The specific steps to modify the voxel point labels are as follows:

- (1) If $L_{a,k}$ contains 0, the points in V_k are in exterior areas, and the labels of the points in V_k do not need to be modified.
- (2) If $L_{a,k}$ does not contain 0, the points in V_k are in void areas. Modify the labels of the points in V_k as the largest integer (∞ is not included) in $L_{a,k}$, that is, the labels of the geological body with the lowest priority.

3.5. Reconstruct and Repair the Model

The vertices of the multi-labeled mesh may have multiple labels, and the model of a certain formation can be extracted from the multi-labeled mesh. According to the needs of users, the model of a specified formation can be extracted from the multi-labeled mesh according to a specific label.

However, after the priority order of the geological model is specified, the labels of the vertices are uniquely determined, and the algorithm will reconstruct the multi-domain model according to the established priority order. Users can specify different colors to represent the mesh of each label to distinguish different formations.

After the labels of the voxel points are set and repaired, the geological model is reconstructed using the M3C algorithm; please refer to [2] for the specific method of the M3C algorithm, which will not be further discussed here.

The pseudo-code of the algorithm flow for modifying voxel point labels and reconstruct the model is shown in Algorithm 3 (modify voxel point labels and reconstruct the model).

Algorithm 3: Modify Voxel Point Labels and Reconstruct the Model

```

1: Input: The qusai-voids  $V_k$ , the label sets  $L_{a,k}$ 
2: Output: Multi-labeled mesh without voids
3: for  $k = 1, 2, \dots$  do
4:     if there is a 0 in  $L_{a,k}$  then
5:         The points in  $V_k$  are exterior points.  $V_k$  is an exterior area.
6:     else
7:         Obtain the max label  $i$  ( $\infty$  is not included) in  $L_{a,k}$ , modify the labels
           of the void points in  $V_k$  as  $i$ .
8:     end if
9: end for
10: Extract the multi-labeled mesh based on the modified voxel points using the M3C
    method.

```

4. Results

We implemented the improved algorithm in C++ and tested it on a Windows 64-bit PC with 3.80 GHz Intel(R) Core(TM) i7-10700K and 32 GB RAM. The three experiments shown in Figures 8–10 are the results of the test. Tables 1–3 record the modeling time and the constraints of the three experiments. Among them, the voxel size is specified by the user, representing the accuracy of modeling. By changing the size, models with different accuracy can be obtained. Actually speaking, the smaller the voxel size, the more accurate the model and the greater the number of voxel points. In order to maximize the efficiency of the algorithm, an appropriate voxel size needs to be used. At the same time, the constraint points are the interpolation points used to obtain the interpolation functions.

Figure 8a,b are the appearances of the two-formation geological models. It can be seen from the appearance of the two models that there is no apparent difference in the appearance of the two models, which indicates that the improved algorithm will not significantly change the appearance of the models but only repair the overlaps and voids inside the models. Figure 8d,g shows the voids in the model reconstructed by the original M3C algorithm. As shown in Figure 8e,h, after using the improved algorithm, the voids in the geological bodies are repaired.

Through the comparison of the modeling time, it can be seen that the process of modifying the voxel labels is efficient. Table 1 shows the comparison of some parameters between the original M3C algorithm and the improved algorithm. When the constraint points and the voxel size (i.e., modeling accuracy) are the same, the improved algorithm consumes about 12% more time than the original M3C algorithm in this case.

Figure 8. Two-formation geological body.

	Formation	Color	N1	N2	N3	d_{vox} (m)	Time (s)	
							Original M3C	Repaired
1	F 1	green						
2	F 2	red	1290	695,020	8905	0.97	68.12	75.96

Table 2. The three-formation geological body statistics table. N1 is the number of the constraint points, N2 is the number of the voxel points of the models, N3 is the number of the voxel points that change their labels, and d_{vox} is the voxel size.

	Formation	Color	N1	N2	N3	d_{vox} (m)	Time (s)	
							Original M3C	Repaired
1	F 1	purple	997	1,450,800	7073	1.6	110.35	124.62
2	F 2	green						
3	F 3	red						

Table 3. The four-formation geological body statistics table. N1 is the number of the constraint points, N2 is the number of the voxel points of the models, N3 is the number of the voxel points that change their labels, and d_{vox} is the voxel size.

	Formation	Color	N1	N2	N3	d_{vox} (m)	Time (s)	
							Original M3C	Repaired
1	F 1	red	963	350,880	1989	11	27.87	32.34
2	F 2	green						
3	F 3	blue						
4	F 4	purple						

Figure 9 shows the number of voxel points for the two-formation geological body (Figure 8) when the voxel size changes. For Figure 9, as the size of the voxel points increases, the number of voxel points decreases significantly. However, the relationship is not linear. Assuming that the length of the x -axis, y -axis, and z -axis of the maximum bounding box are l_1 , l_2 , and l_3 , respectively, and the size of the voxel is d_{vox} , then the number of voxel points is $N2 = (l_1/d_{vox} + 1)(l_2/d_{vox} + 1)(l_3/d_{vox} + 1)$. It can be seen that $N2$ is in an approximate inverse cubic relationship with d_{vox} .

Figure 10 shows the computational time for the two-formation geological body (Figure 8) when the voxel size changes. Without considering the acceleration of the estimation speed, the calculation complexity of a voxel point function value is $O(n)$, where n is the number of the constraint points, and the calculation complexity of m voxel points is $O(mn)$. Assuming that the length of the x -axis, y -axis, and z -axis of the maximum bounding box are l_1 , l_2 , and l_3 , respectively, and the size of the voxel is d_{vox} , then the calculation complexity of m voxel points is $O(n(l_1/d_{vox} + 1)(l_2/d_{vox} + 1)(l_3/d_{vox} + 1))$.

Figure 11 shows the percentages of point-modified labels for the two-formation geological body (Figure 8) when the voxel size changes. It can be seen that the percentage of points modified labels is very small, which is useful to preserve the original model shape. In general, the proportion of modified voxel points becomes stable when the voxel size is small enough.

The experiment shown in Figure 12 is a three-formation model. For a three-formation geological body, the improved algorithm can still repair the void defects. It can be seen from Table 2 that when the constraint points and the voxel size are the same, the modeling time of the improved algorithm is about 13% higher than that of the original M3C algorithm.

Figure 13 is the model of a four-formation geological body. It can be seen from the model slices in Figure 13e,h that the improved algorithm fixes the voids in the model. The modeling time of the improved algorithm is about 16% higher than that of the original M3C.

The three groups of experiments from Figures 11–13 verify the multi-domain modeling effect of the improved algorithm. It can be found that when the functions of the geological bodies are defective or the common interfaces of the geological bodies are reconstructed many times, there may be overlaps and voids in the models directly reconstructed by the M3C algorithm. After using the improved algorithm, the overlaps and voids are repaired, and the interfaces of the formations are kept in good adjacency. Moreover, it can be seen from the statistics of the modeling time that the process of repairing voxel point labels is efficient.

Figure 11. The relationship between the percentages of points modified labels and the voxel size.

Figure 12. Three-formation geological body.

Figure 13. Four-formation geological body.

5. Conclusions and Discussion

Based on the M3C algorithm, this paper improves it so that the improved algorithm can deal with the defects of overlaps and voids in the process of multi-domain reconstruction of the geological models. In this paper, we label the voxel points and modify the labels of overlapping and void points, and then models reconstructed by the M3C algorithm maintain the raw structural adjacency relationships.

However, the improved algorithm still needs to be further improved. First of all, the algorithm is currently only applicable to the smooth surface reconstruction of the geological models. From the modeling results, the detailed features of the model surface cannot be guaranteed. The model is easy to remove sharp features, there are stair-stepped characteristics on the surface, and there are ring marks on the surface of the model. The stair-stepped characteristics are because the models are not smooth enough, but the models cannot be smoothed too much; otherwise, the models will change a lot. The extended marching cubes (EMC) algorithm [37], which takes sharp features into account, can be used for reference to ensure the sharp features of the models. Alternatively, the use of linear interpolation for the cubes instead of introducing intermediate nodes at the midpoint, face center, and body center of the cubes could make the model smoother, thereby reducing the stair-stepped characteristics in the modeling process.

Secondly, the processing method of the overlapping and void areas can be further optimized. The processing idea of the improved algorithm is to classify the total overlap or void areas to a certain geological body. However, for the classification method, there can be better interfaced fitting methods, such as dividing the overlaps and voids into multiple sub-regions and then discussing the attribution of each sub-region.

Finally, the modeling efficiency of the improved method needs to be improved. The improved algorithm occupies a large amount of computer memory, and the modeling efficiency of the improved algorithm needs to be improved in the future. On the one hand, it can be considered to improve the efficiency of surface reconstruction through parallel computing. On the other hand, the number of voxel points involved in the calculation can be reduced through the surface tracking method. For example, the MC algorithm based on the voxel growing method [38] can improve the efficiency of the surface reconstruction.

Author Contributions: B.L., D.Z., and L.W. conceived, designed, and performed the experiments; D.Z. analyzed the data and revised the methodology; B.L. wrote the paper; all authors discussed the results and revised the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This work was financially supported by the National Key R&D Program of China (2017YFC0602905).

Data Availability Statement: Not applicable.

Acknowledgments: Thanks to the public datasets used in this research. We also thank the reviewers for their comments and suggestions to improve the quality of the paper.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

References

1. Lorensen, W.E.; Cline, H.E. Marching cubes: A high resolution 3d surface construction algorithm. *ACM Siggraph Comput. Graph.* **1987**, *21*, 163–169. [CrossRef]
2. Wu, Z.; Sullivan, J.M., Jr. Multiple material marching cubes algorithm. *Int. J. Numer. Methods Eng.* **2010**, *58*, 189–207. [CrossRef]
3. Houlding, S.W. 3D Geoscience Modeling: 1994. Available online: <https://www.springer.com/gp/book/9783642790140> (accessed on 7 May 2021).
4. Cowan, E.J.; Beatson, R.K.; Fright, W.R.; McLennan, T.J.; Mitchell, T.J. Rapid geological modelling. In Proceedings of the Applied Structural Geology for Mineral Exploration and Mining, International Symposium, Kalgorlie, Australia, 23–25 September 2002.
5. Cowan, E.J.; Beatson, R.K.; Ross, H.J.; Fright, W.R.; McLennan, T.J.; Evans, T.R.; Carr, J.C.; Lane, R.G.; Bright, D.V.; Gillman, A.J. Practical implicit geological modelling. In Proceedings of the Fifth International Mining Geology Conference, Bendigo, Australia, 17–19 November 2003; pp. 89–99.
6. Turner, A.K. *Three-Dimensional Modeling with Geoscientific Information Systems*; Kluwer Academic Publishers: Dordrecht, The Netherlands, 1992.
7. Qiao, J.; Mao, P.; Li, Z.; Yi, J. 3D geological modeling from dem, boreholes, cross-sections and geological maps. In Proceedings of the International Conference on Geoinformatics, Shanghai, China, 24–26 June 2011; pp. 1–5.
8. Zanchi, A.; Francesca, S.; Stefano, Z.; Simone, S.; Graziano, G. 3D reconstruction of complex geological bodies: Examples from the alps. *Comput. Geosci.* **2009**, *35*, 49–69. [CrossRef]

9. Jin, X.; Tai, C.-L.; Zhang, H. Implicit modeling from polygon soup using convolution. *Vis. Comput.* **2008**, *25*, 279–288. [\[CrossRef\]](#)
10. Hornus, S.; Angelidis, A.; Cani, M.-P. Implicit modeling using subdivision curves. *Vis. Comput.* **2003**, *19*, 94–104. [\[CrossRef\]](#)
11. Guo, J.; Wang, J.; Wu, L.; Liu, C.; Li, C.; Li, F.; Lin, M.; Jessell, M.W.; Li, P.; Dai, X.; et al. Explicit-implicit-integrated 3-D geological modelling approach: A case study of the xianyan demolition volcano (Fujian, China). *Tectonophysics* **2020**, *795*, 228648. [\[CrossRef\]](#)
12. Mueller, T.G.; Dhanikonda, S.R.K.; Pusuluri, N.B.; Karathanasis, A.D.; Mathias, K.K.; Mijatovic, B.; Sears, B.G. Optimizing inverse distance weighted interpolation with cross-validation. *Soil Sci.* **2005**, *170*, 504–515. [\[CrossRef\]](#)
13. Lu, G.Y.; Wong, D.W. An adaptive inverse-distance weighting spatial interpolation technique. *Comput. Geosci.* **2008**, *34*, 1044–1055. [\[CrossRef\]](#)
14. Xiao, M.; Zhang, G.; Breilkopf, P.; Villon, P.; Zhang, W. Extended co-kriging interpolation method based on multi-fidelity data. *Appl. Math. Comput.* **2018**, *323*, 120–131. [\[CrossRef\]](#)
15. Matheron, G. Principles of geostatistics. *Econ. Geol.* **1963**, *58*, 1246–1266. [\[CrossRef\]](#)
16. Caballero, W.; Giraldo, R.; Mateu, J. A universal kriging approach for spatial functional data. *Stoch. Environ. Res. Risk Assess.* **2013**, *27*, 1553–1563. [\[CrossRef\]](#)
17. Salkauskas, P.L. Surfaces generated by moving least squares methods. *Math. Comput.* **1981**, *37*, 141–158.
18. Rath, A.K.; Chakraborty, A. Improved moving least square-based multiple dimension decomposition (mdd) technique for structural reliability analysis. *Int. J. Comput. Methods* **2020**, *18*, 2050024. [\[CrossRef\]](#)
19. Gross, B.J.; Trask, N.; Kuberry, P.; Atzberger, P.J. Meshfree methods on manifolds for hydrodynamic flows on curved surfaces: A generalized moving least-squares (gmls) approach. *J. Comput. Phys.* **2020**, *409*, 109340. [\[CrossRef\]](#)
20. Skala, V. Rbf interpolation with csrbf of large data sets. *Procedia Comput. Sci.* **2017**, *108*, 2433–2437. [\[CrossRef\]](#)
21. Park, J.; Sandberg, I. Universal approximation using radial-basis-function networks. *Neural Comput.* **2014**, *3*, 246–257. [\[CrossRef\]](#)
22. Cuomo, S.; Galletti, A.; Giunta, G.; Marcellino, L. Reconstruction of implicit curves and surfaces via rbf interpolation. *Appl. Numer. Math.* **2017**, *116*, 157–171. [\[CrossRef\]](#)
23. Franke, B.R. Scattered data interpolation: Tests of some methods. *Math. Comput.* **1982**, *38*, 181–200.
24. Durst, M.J. Letters: Additional reference to marching cubes. *Comput. Graph.* **1988**, *22*, 243. [\[CrossRef\]](#)
25. Oh, S.; Koo, B.K. Data perturbation for fewer triangles in marching tetrahedra. *Graph. Models* **2007**, *69*, 211–218. [\[CrossRef\]](#)
26. Nielson, G. In the asymptotic decider: Resolving the ambiguity in marching cubes. In Proceedings of the IEEE Visualization, San Diego, CA, USA, 22–25 October 1991.
27. Natarajan, B.K. On generating topologically consistent isosurfaces from uniform samples. *Vis. Comput.* **1994**, *11*, 52–62. [\[CrossRef\]](#)
28. Chernyaev, E.V. Marching Cubes 33: Construction of Topologically Correct Isosurfaces. 1996. Available online: <https://www.researchgate.net/publication/2651412> (accessed on 7 May 2021).
29. Lopes, A.; Brodlie, K. Accuracy in contour drawing. In *Eurographics UK*; University of Leeds: Woodhouse, UK, 1998; pp. 301–311.
30. Cignoni, P.; Ganovelli, F.; Montani, C.; Scopigno, R. Reconstruction of topologically correct and adaptive trilinear isosurfaces—Sciencedirect. *Comput. Graph.* **2000**, *24*, 399–418. [\[CrossRef\]](#)
31. Wilhelms, J.; Gelder, A.V. Octrees for faster isosurface generation. *Acm Trans. Graph.* **1990**, *24*, 57–62.
32. Bajaj, C.L.; Pascucci, V.; Schikore, D.R. Fast isocontouring for improved interactivity. In Proceedings of the 1996 Symposium on Volume Visualization, San Francisco, CA, USA, 28–29 October 1996; IEEE: Piscataway, NJ, USA, 2001; pp. 39–46.
33. Schroeder, W.J.; Zarge, J.A.; Lorensen, W.E. Decimation of triangle meshes. *Acm Siggraph Comput. Graph.* **1992**, *26*, 65–70. [\[CrossRef\]](#)
34. Faraj, N.; Thiery, J.-M.; Boubekur, T. Multi-material adaptive volume remesher. *Comput. Graph.* **2016**, *58*, 150–160. [\[CrossRef\]](#)
35. Zhang, Y.; Qian, J. Resolving topology ambiguity for multiple-material domains. *Comput. Methods Appl. Mech. Eng.* **2012**, *247*–248, 166–178. [\[CrossRef\]](#)
36. Zhong, D.Y.; Wang, L.G.; Bi, L. Implicit surface reconstruction based on generalized radial basis functions interpolant with distinct constraints. *Appl. Math. Model.* **2019**, *71*, 408–420. [\[CrossRef\]](#)
37. Congote, J.; Moreno, A.; Barandiaran, I.; Barandiaran, J.; Ruiz, O. Extending marching cubes with adaptive methods to obtain more accurate iso-surfaces. In *Computer Vision, Imaging and Computer Graphics: Theory and Applications*; Ranchordas, A.K., Pereira, J.M., Araujo, H.J., Tavares, J., Eds.; Springer: Berlin, Germany, 2010; Volume 68, p. 35.
38. Zhong, D.Y.; Zhang, J.; Wang, L.G. Fast implicit surface reconstruction for the radial basis functions interpolant. *Appl. Sci.* **2019**, *9*, 5335. [\[CrossRef\]](#)