

Research Paper

Mid-surface mesh abstraction for thin-walled structures based on virtual topology



Feiqi Wang^{ID}, Haidong Wang, Xiang Zhao, Qi Ran, Guidong Wang, Huan Zhang, Xin Hu, She Li,
Xiangyang Cui*

State Key Laboratory of Advanced Design and Manufacturing Technology for Vehicle, College of Mechanical and Vehicle Engineering, Hunan University, Changsha, 410082, Hunan, China

ARTICLE INFO

Keywords:

Mid-surface mesh
Virtual topology
Thin-walled structures
Maximum volume decomposition

ABSTRACT

The objective of this study is to propose a virtual topology-based mid-surface mesh abstraction method. Firstly, the complex model is decomposed into simpler volumes, and virtual topology operations are performed on unilateral faces of these simpler volumes. Subsequently, virtual topology meshes of unilateral faces are generated. With these virtual topology meshes served as initial meshes, mid-surface meshes of the simpler volumes can be effectively acquired with high accuracy by using the new-developed local mesh projection algorithm. With the combination of the mid-surface meshes of the simpler volumes, the mid-surface mesh of the original solid model can be established.

Key features of the proposed method include: (1) reducing model complexity through model decomposition, (2) enhancing initial mesh quality using virtual topology operations, (3) by utilizing local mesh projection algorithms, high-quality mid-surface meshes can be easily accessible without the need for complex mesh partitioning, compared to traditional CAT methods which accurately control the type and size of mid-surface meshes, and (4) reducing human-computer interaction in the process of converting mid-surface meshes to thin-walled structures. Finally the effectiveness and feasibility of the proposed method has been demonstrated through several cases.

1. Introduction

In the field of Computer-Aided Engineering (CAE), simulation and analysis of thin-walled structures are critical due to their widespread application in various industries, including automotive, aerospace, and civil engineering. Thin-walled structures, characterized by their small thickness compared to other dimensions, present unique challenges in modeling and simulation. Accurate representation of these structures typically requires fine meshing, which can lead to high computational costs and extensive processing time. Consequently, there is a pressing need for efficient model reduction techniques that can simplify these complex models without compromising their accuracy and fidelity.

Mid-surface abstraction, as a model reduction technique, has emerged as a promising solution for addressing the challenges associated with simulating thin-walled structures. This approach involves reducing a three-dimensional thin-walled solid model to a two-dimensional mid-surface representation. By focusing on the mid-surface, the method significantly reduces the number of elements and nodes in the mesh, thereby lowering the computational burden while maintaining essential structural characteristics and behaviors.

The motivation for this study is driven by the increasing demand for efficient and accurate simulation tools in the CAE domain. Traditional modeling approaches for thin-walled structures often result in large, complex models that are not only computationally intensive but also difficult to handle in iterative design processes. Direct mid-surface meshing offers a practical alternative, enabling engineers to achieve high-fidelity simulations with reduced computational resources.

This paper aims to explore the application of direct mid-surface meshing in the dimensional reduction of thin-walled solid models within the CAE framework. We will discuss the theoretical underpinnings of the method, including the mathematical formulation and maximum volume decomposition algorithm. Additionally, we will present case studies from various engineering applications to demonstrate the effectiveness and accuracy of this approach.

The structure of this paper is as follows: Section 2 reviews the existing model reduction techniques and their applications to thin-walled structures. Section 3 outlines the methodology of direct mid-surface meshing. Section 4 delves into the methods for mid-surface mesh

* Corresponding author.

E-mail address: cuixy@hnu.edu.cn (X. Cui).

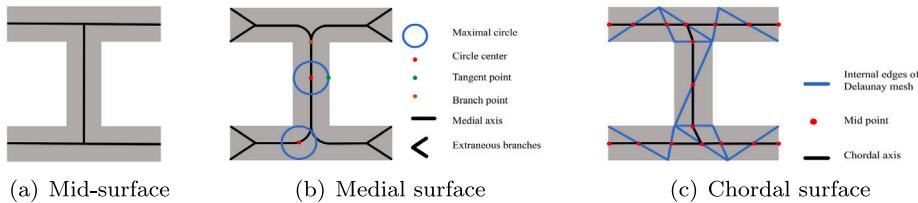


Fig. 1. The difference between Mid-surface, Medial surface and Chordal surface.

abstraction in detail. Section 5 presents the results from our case studies and discusses their implications. Finally, Section 6 concludes with a summary of findings and recommendations for future research.

By providing a comprehensive analysis of direct mid-surface meshing, this paper seeks to contribute to the advancement of efficient simulation practices in the CAE field, ultimately aiding engineers in designing and optimizing thin-walled structures with greater precision and reduced computational effort.

2. Related work

MAT method. Mid-Surface abstraction is a dimensional reduction method that is used frequently in CAE simulation and analysis. Fig. 1(a) shows the mid-surface results commonly used in CAE analysis. The MAT method is continually evolving. Blum [1] was the originator of the MAT method, and after its continuous optimization, it has become a popular tool for simplifying CAE analysis models. The fundamental concept behind the MAT method is to ascertain the central path of the most extensive circle within the 2D plane. Fig. 1(b) illustrates the Medial surface results generated using the MAT approach. Nevertheless, this approach generates numerous minuscule branches at the corners, necessitating cropping and extension in order to achieve the accurate center axis. Lee's [2] approach is capable of producing mid-surfaces for 3D models, yet it lacks the capability to address degenerate scenarios, resulting in incomplete Mid-Surface at the edges. Ramanathan's [3] approach enables the creation of more comprehensive mid-surfaces, yet necessitates alterations to the original topology generated in order to achieve the accurate topology. The MAT method was the first to be proposed for the generation of mid-surfaces for thin-walled parts, but it has obvious drawbacks. The generation of mid-surface results in the creation of numerous intricate branches. The outcomes achieved are also inferior to the mid-surface. The generation of center faces by MAT necessitates intricate topology operations, making it challenging to obtain the accurate topology.

CAT method. The mesh model based method for generating the mid-surface is an innovative approach. This allows for direct CAE mesh modeling of the mid-surface without generating a geometric mid-surface. Quadros [4] pioneered the direct generation of mesh-based mid-surface models using the CAT method. This method involves generating a single layer of tetrahedral meshes and then using the chords of the tetrahedral meshes to construct a triangular mesh. The triangular meshes are then joined together to obtain the mesh mid-surface. This method is only valid for manifold mid-surfaces. In order to be able to draw mid-surfaces for non-manifold models, Quadros [5] has optimized the previous method. The optimized algorithm is able to handle non-manifold models. This method relies too much on a single-layer tetrahedral mesh. For complex spline models with uneven thicknesses, generating a single tetrahedral mesh is time-consuming and expensive. In particular, the modeling of blends may be more complicated, and many cases need to be discussed and analyzed. The robustness of this method is poor. The quality of the meshes obtained is also poor. Fig. 1(c) illustrates the Chordal surface results generated using the CAT approach.

Face-Pair method. The face-based getting method was first proposed by Rezayat [6] and then widely used in mid-face generation. The core of this method is based on FAG(Face Adjacency Graph) to detect faces. The detected faces are utilized to generate the mid-surface. It is considered as the first method for mid-face abstraction and some basic criteria for detecting face pairs are proposed. However, these criteria are only valid for simple parts. Lee [7] improved the graph-based face pair detection method and used edges as extension rules connecting the mid-surface patches. This is probably the first method that focuses on the topological boundaries of mid-surface. However, its limitation is that face pair recognition often fails when the topology of the model is complex. Zhu [8] suggested a method based on virtual decomposition for extracting mid-surface elements. The idea of face pairs is suggested for the virtual breakdown of a thin-walled model into multiple basic virtual components, with the interconnections among these segments classified into three distinct categories. Subsequently, the mid-surface model's topological boundary edges are accurately derived from the types of connections. B-spline surface fitting remains a necessity for this technique. The technique is plagued by identical flaws as Chong's [9] method. Zhu [10] suggested a method for automated surface layering abstraction that relies on the breakdown of ribs in models with thin walls. This technique breaks down the model using the recognized rib characteristics and additionally gathers layered semantic data. Subsequently, we individually acquire intermediate surface patches for each sub-region, each with distinct semantic configurations, through a method based on offset operation and discretization. Ultimately, the creation of the intermediate surface model, encompassing hierarchical semantic data, is achieved by merging all distinct segments.

Decomposition method. Therefore, a model decomposition-based approach is proposed to deal with complex models. To deal with complex thin-walled models, a decomposition-based approach has been widely used and has become the most effective mid-surface abstraction method. Chong [9] first splits the solid model into simple sub-parts, and then detects face pairs in each part to form a mid-surface patch. This method can handle models with a mixture of thin and thick parts. However, the decomposition method can only handle models with simple planar or quadratic surfaces, and thus is not applicable to models with complex free-form surfaces. As shown in Fig. 2, Woo [11] proposed another decomposition method for mid-surface abstraction. The model is first decomposed into some basic elements, and then these elements are combined to generate the maximum volume so that the facing can be easily detected. In their approach, Boolean operations are often used, making the model unstable, especially for some models with free-form surfaces. Another problem is that each facing abstracts an intermediate surface patch. All of these patches are separate and require extension or trimming operations to put them together. This does not guarantee that the topology of the mid-surface model is correct. Robinson [12] et al. proposed a mixed dimensionality based method for analyzing thin wall models. In their method, the flat regions in the MAT model are retained while other regions (branching parts) are replaced with the original model. Then the mixed dimensional model with finite elements is generated. This method avoids topological problems caused by branching. However, it is less efficient and cannot maintain the shape of the original model. Nolan [13] et al. proposed

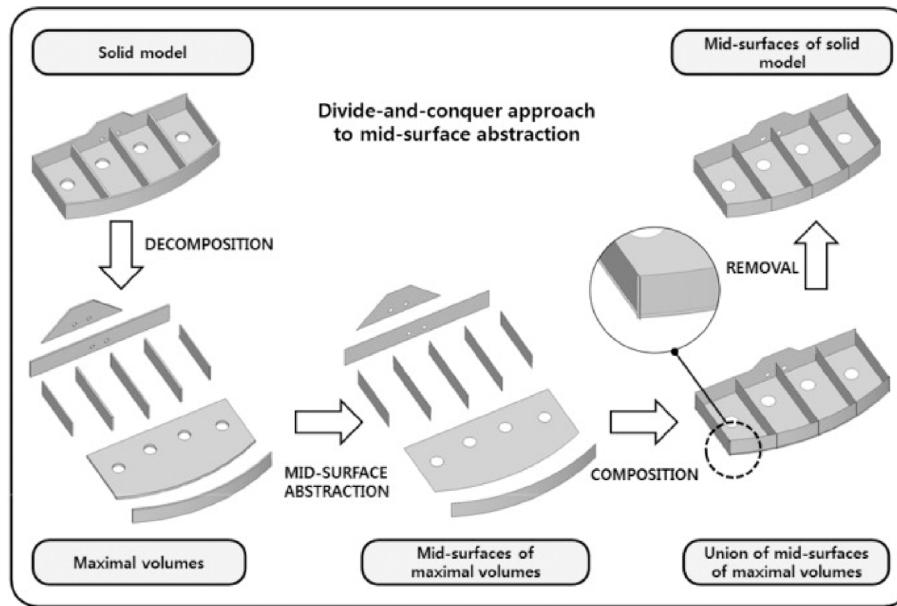


Fig. 2. The overall procedure for the divide-and-conquer approach to mid-surface abstraction.

a meshing method for reinforcing thin-walled structures, in which the original model is first decomposed into slender, thin and complex regions. Each region was converted into a different type of mesh. The slender regions were idealized as beam elements. The thin slices are used to generate triangular surface meshes. The complex regions were meshed with tetrahedral elements. Rules are then used to join these types of meshes together. Here, the abstract beam elements are recognized as thin regions and therefore play the role of connections in the model by adding constraints. However, only these constraints are not sufficient for all types of analysis applications. For example, fluid dynamics analysis requires more detail than it can provide.

At the same time, software companies have launched commercial software with mid-surface abstraction capabilities. Commercial software usually has a 'Face-Pair' function panel in the surface module, which allows the quick creation of mid-surfaces for thin-walled models from solid bodies. We tested several thin-walled models using the commercial software and experienced unsuccessful and incomplete results with mid-surface abstractions. An example of such a case is shown in Fig. 3. The mid-surface abstraction techniques used in commercial CAD systems are proprietary and not publicly disclosed, making it impossible for us to determine the exact cause of this issue. Furthermore, the commercial software is unable to directly generate a mid-surface mesh model from a thin-walled model.

3. Motivation and method overview

3.1. Motivation

While related work has proposed a number of methods for performing mid-surface abstraction for thin wall models. Many acute problems remain in real engineering applications.

(1) Boolean Operations and Surface Continuity: Earlier algorithms relied heavily on boolean operations, which significantly slowed down the mid-surface generation process. Moreover, boolean operations often introduce surface discontinuities between adjacent objects, which is problematic for applications requiring smooth, continuous surfaces, such as those in high-precision engineering.

(2) Limitations of Offset-Based Mid-Surface Generation: Although offset operations offer an efficient way to generate mid-surfaces, they are primarily suited for models containing simple geometries such as secondary surfaces and planes. This approach struggles with

thin-walled models that feature complex surfaces with varying thickness, leading to inaccurate results or even failure in capturing critical geometric details.

(3) Challenges in Finite Element Mesh Generation: Simulation analysis using the finite element method requires a discrete mesh rather than a continuous geometric model. While some methods can directly generate mid-surface meshes, they often lack efficiency and robustness, particularly when handling large-scale models or geometries with intricate details. These limitations result in poor mesh quality, which directly impacts the accuracy of the simulation results.

(4) Handling of Intersecting Features: Thin-walled models often include intersecting features, such as ribs or stiffeners, which complicate mid-surface generation. Existing methods struggle to accurately handle these intersections, leading to incomplete mid-surface or poor quality meshes. The ability to effectively manage intersecting features is critical for ensuring structural integrity and simulation accuracy.

This enumerates the acute problems with existing mid-surface abstraction methods in engineering applications. This research is motivated by the desire to propose a more complete mid-surface abstraction algorithm to generate mid-surface mesh with guaranteed efficiency and high robustness.

3.2. Method overview

The first step is to perform maximal volume decomposition on the original model, breaking down the model with complex intersecting features into simpler sub-models. The second step of mid-surface abstraction is to identify face pairs from a solid model. A face-pair is a set of two faces that forms part of the thin walls of the solid model. This automatic pairing process is critical to the success of the subsequent operations for mid-surface algorithms [11].

The third step is to create virtual topology. The virtual topology allows for extremely robust and efficient manipulation of small sharp-edged and narrow surfaces on thin walled structures. Which improves the mesh quality in the third stage.

The fourth step is the meshing of the thin-walled structure. In contrast to previous methods, our method requires only the mesh on one side of the thin-walled structure geometry. Mesh optimization based on Laplace's method after mesh refinement.

The final step is to generate mid-surface mesh of the thin-walled structure. This involves operations such as mesh clipping and merging coincident nodes.

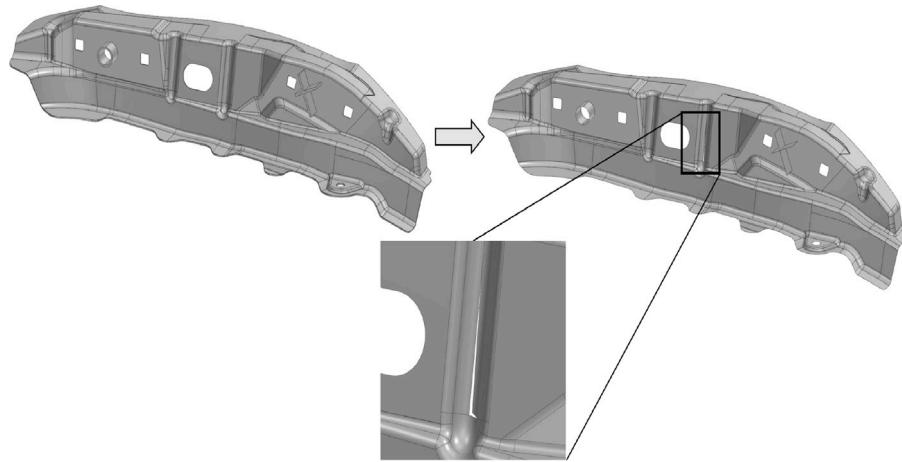


Fig. 3. The incomplete mid-surfaces obtained using commercial software.

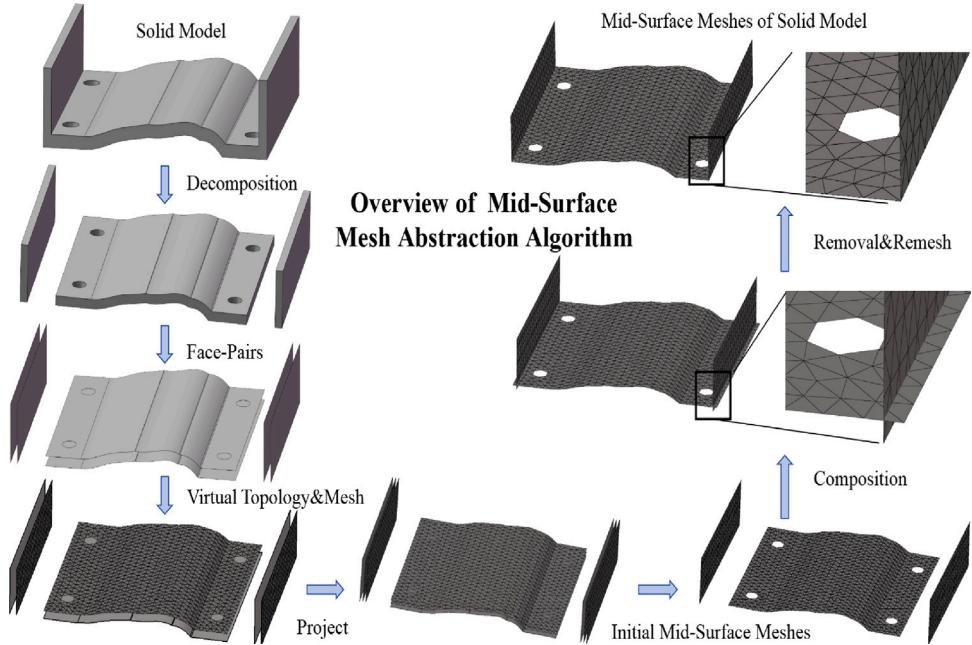


Fig. 4. Overview of mid-surface mesh abstraction algorithm.

The overall procedure for the proposed method is presented in Fig. 4.

4. Direct mid-surface mesh abstract method

4.1. Decomposition

Within the realm of manifold geometry, the term edge usually denotes a connection of limited length on a face or edge. In a three-dimensional manifold, two vertices are linked by an edge, with these vertices situated on the surface. The common edge of each manifold geometry is shared by at most two surfaces. In non-manifold geometries, the definition of edges may be more flexible, as non-manifold geometries may contain discontinuous topological structures. The common edge of each non-manifold geometry is shared by at least three surfaces. Fig. 5 illustrates the difference between non-manifold shared edges and manifold shared edges.

However, the identification and removal of end faces in thin-walled models within non-manifold geometry require more complex operations. Creating mid-surface using face-to-face methods for such models

— Non-manifold shared edge
— Manifold shared edge

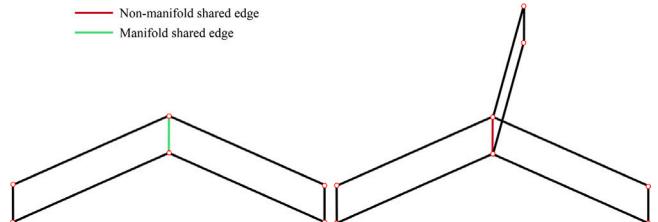


Fig. 5. Non-manifold shared edge and manifold shared edge.

is extremely challenging. In general, creating mid-surface involves offsetting the solid surface, trimming the intermediate surface to obtain intermediate surface patches, extending the intermediate surface patches, and finally stitching the patches together to form a complete intermediate surface.

Subsequently, a divide-and-conquer method for generating intermediate surfaces is proposed. Divide and conquer is a problem-solving

strategy that involves breaking down a large problem into simpler sub-problems, solving each sub-problem individually, and then combining their solutions to obtain the solution to the original problem. In feature-based modeling CAD systems, it is common to add simple entities, referred to as primitives or design features, to construct complex solid features. This is a widely recognized modeling approach. With this method, users can create complex solid models and modify them more intuitively. On the contrary, there is also research focused on employing reverse operations of the above method, decomposing complex entities into simpler ones. These simple entities do not generate non-manifold geometric features during intermediate surface abstraction. This greatly reduces the difficulty of obtaining correct intermediate surface topology. Therefore, for complex solid models, we adopt a divide-and-conquer approach to obtain simple entities [14–17]. This method first decomposes the solid model into sub-volumes known as maximal volumes. Then, for each maximal volume, end face removal operations are carried out following the steps of NFSP. Below, detailed explanations of the mentioned methods will be provided. As the solid model becomes more complex, it becomes difficult to locate and remove end faces within the solid model. More concave edges often lead to a more complex solid model. Decompose a complex solid model into simple sub-entities, ensuring that the union of all sub-models is the original complex solid. The process of finding and removing end faces for sub-models will become simpler. Many studies have already demonstrated the ability to decompose complex solid models into simple entities. As shown in Fig. 6, the maximal volume decomposition method is applied in this research. The maximal volume V of volume S , denoted as MV, should satisfy the following conditions:

- (1) $V \subset S$,
- (2) each of the halfspaces of V is a halfspace of S ,
- (3) V has no concave edges,

(4) for each maximal face of V , there exists a face of S such that the intersection of the faces is a two manifold with a boundary, and

(5) $V \subseteq A$, where A is a volume that satisfies the above conditions.

Maximal face in condition (4) is defined as the union of the faces of V that share the same halfspace. As shown in Fig. 6, the process of maximal volume decomposition mainly consists of two steps: decomposition and composition. First, the solid model S is decomposed using half-spaces, resulting in 11 cells. Then, these cells are combined. The resulting new solid V satisfies the requirements of maximal volume. Cells 1, 2, and 3 are combined to form the new solid V_1 . Cells 4, 5, and 6 are combined to form the new solid V_2 . Cells 7, 8, and 9 are combined to form the new solid V_3 . Cells 10, 11, and 12 are combined to form the new solid V_4 . V_1 to V_4 satisfy the above five conditions, thus they are the maximal volumes resulting from the decomposition of solid S .

The process for the method of maximal volume decomposition is as follows:

(1) Initial Shape Analysis: Analyze the complex shape by identifying its concave edges, vertices, and intersecting faces. For polyhedral shapes, recognize all the concave edges that need to be processed.

(2) Half-Space Division: Extend the faces corresponding to the concave edges to form half-spaces. These half-spaces are used to decompose the polyhedron into maximal convex cells (MCC). The MCCs are the largest convex volumes that can be created by intersecting these half-spaces.

(3) Recursive Decomposition: Iteratively decompose the polyhedron by intersecting the shape with different combinations of half-spaces. For each step, analyze the geometric relationships among the half-spaces to efficiently generate maximal convex cells.

(4) Combining Maximal Convex Cells: Once the maximal convex cells are generated, combine them through Boolean operations to form larger volumes.

The pseudocode for the method of maximal volume decomposition is as follows:

Due to the simple shape of the maximum volume in the solid model, which does not contain concave edges, the identification and removal of the lateral faces of the maximum volume are more straightforward. The results of the maximum volume decomposition are shown in Fig. 7.

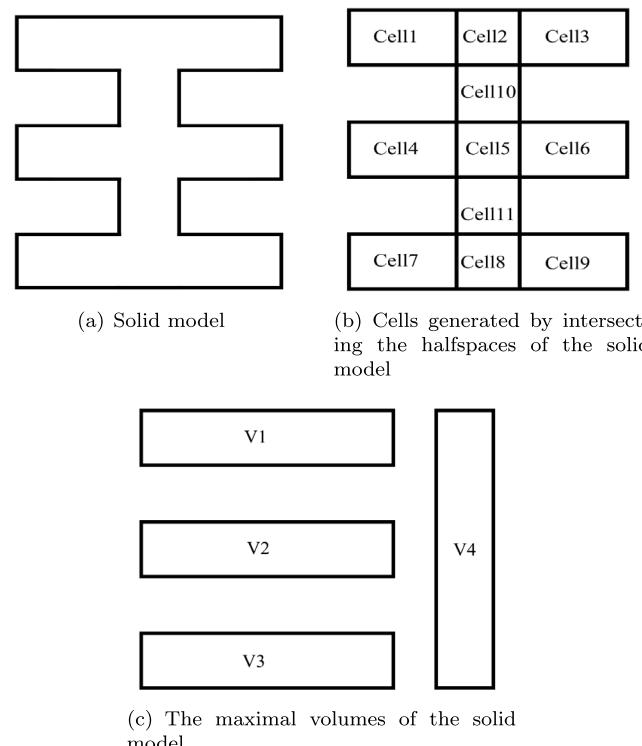


Fig. 6. Maximum volume decomposition and combination.

Algorithm 1: Maximal Volume Decomposition

```

Input: Complex geometry  $S$ 
Output: Decomposed maximal volume cells  $V_S$ 
1 Initialize an empty set  $V_S$ ;
2 Extract concave edges  $\{e_1, e_2, \dots, e_n\}$  from the geometry  $S$ ;
3 for each concave edge  $e_i$  do
4   Generate the corresponding half-space  $H_i$ ;
5   Intersect  $H_i$  with the geometry  $S$  to create multiple convex
   cells  $V_i$ ;
6   for each convex cell  $V_i$  do
7     Check if  $V_i$  satisfies the maximal convex cell (MCC)
     conditions;
8     if conditions are met then
9       | Add  $V_i$  to the set  $V_S$ ;
10      end
11    end
12 end
13 Merge adjacent convex cells in  $V_S$  to produce the final maximal
   volume decomposition;
14 return  $V_S$ ;

```

4.2. Create faces-pairs

4.2.1. The definition of faces-pairs.

The mid-surface abstraction method based on face-pairs is still the most popular method today. Thereby reducing computational and storage complexity and improving computational efficiency. There are three criteria that are commonly used in the existing methods for judging whether two faces are a valid face-pair or not: the distance criterion, the overlap criterion and direction criterion.

Distance criterion. Mid-surfaces are usually abstracted for thin-walled parts. The minimum distance H between two faces in a Face-Pair should not exceed the thickness threshold of the thin sheet member

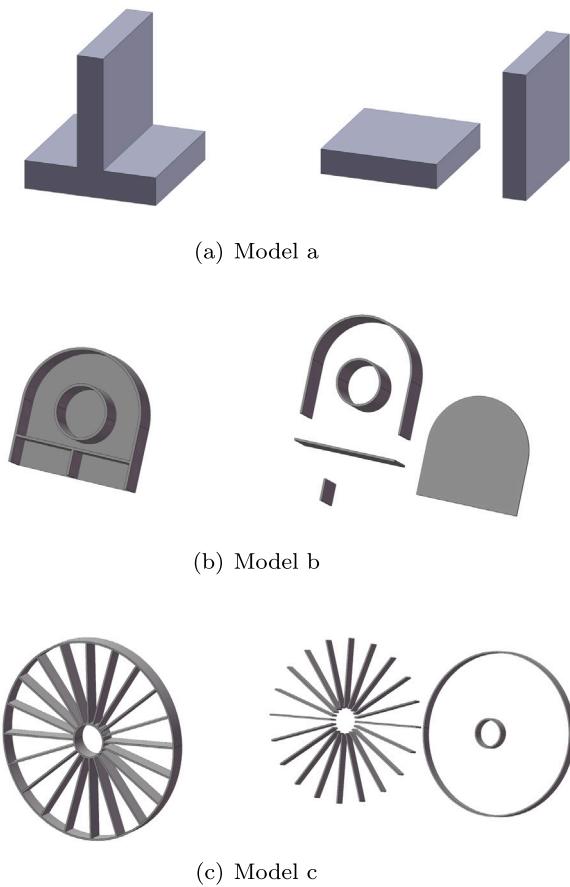


Fig. 7. Examples of maximal volume decompositions.

t_{min} , t_{min} has the potential to be either a predetermined value inputted by the user or a value acquired through automatic evaluation. Based on Eq. (1), if we choose $t_{min} = 2$, then according to Eq. (1), both the mid-surfaces in Figs. 8(a) and 8(b) are valid. However, in practical applications, the mid-surface in Fig. 8(a) is preferred because it better represents a thin-walled model.

$$H \leq t_{min} \quad (1)$$

To control the automatic selection of the mid-surface, another equation needs to be introduced [11].

$$\min(L, W)/H > X, 1.2 \leq X \leq 3. \quad (2)$$

In Fig. 8(a), the result calculated using Eq. (2) is 2, while in Fig. 8(b), the result is 0.5. Therefore, the upper and lower surfaces in Fig. 8(a) should be identified as a face-pair.

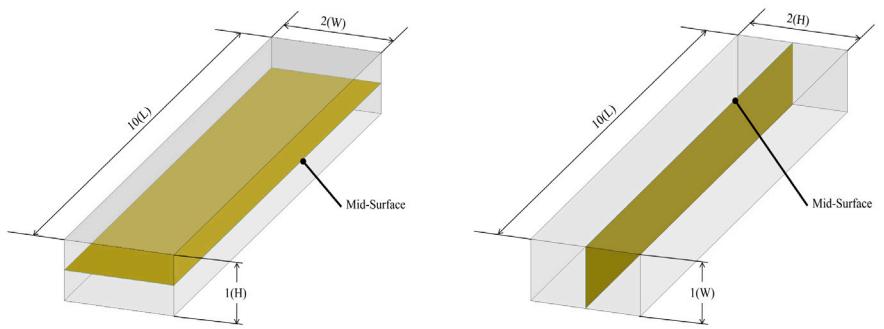
Overlap criterion. It is imperative to guarantee the presence of a shared projection area when projecting two faces in a face-pair [8]. In Fig. 9, faces A and B satisfy the distance criterion. Face A projected along the normal of face B has an overlapping region AB. Therefore, faces A and B form a valid pair. Faces A and F also satisfy the distance criterion. However, when face A is projected along the direction of face F, there is no overlapping region between them. Therefore, according to the overlapping criterion, faces A and F are not a valid pair. This criterion is generally effective, but there may be exceptions. Faces C and D satisfy the distance criterion and should be considered a valid pair. However, when face C is projected along the normal of face D, there is no overlapping region between them. Therefore, faces C and D are incorrectly estimated as an invalid pair. This issue also needs to be addressed.

By utilizing these two criteria, it is effortless to locate the face-pairs in basic thin-walled models. These face-pair models, which are thin-walled, are referred to as strictly matching. Typically, these face-pairs consist of uncomplicated surfaces like planar or cylindrical. As is shown in Fig. 10, the face-pairs AB, CD, and EF of this thin-walled model are matched one-to-one. These Face-Pairs are defined as strictly matching face-pair(SMFP).

These face-pairs features are relatively common in the machining field. Despite this, thin-walled components created through more intricate techniques such as stamping and powder metallurgy can have a great deal of intricate surfaces. These thin-walled components find extensive application in aviation, maritime vessels, automobiles, and numerous other industries. The presence of geometric model design errors, model file conversion errors, process design, and other factors often leads to the non-strict symmetry of the face-pairs in thin-walled components geometric models. As shown in Fig. 11, there are two face-pairs AB and AC. The face A is used twice. In the matching relationship of the faces, it is not one-to-one. These face-pairs are defined as non-strictly matching face-pair(NSMFP). The creation of faces in a non-strictly symmetric face-pairs presents certain obstacles. The generation of the mid-surface using the bias method will lead to ambiguity in the bias distances between A-B, and A-C, consequently causing gaps in the biased section of the B-C surface and resulting in the generation of inaccurate CAE mesh models. Zhu's [8] method also fails to work on non-strictly matching faces. When matching edge pairs, ambiguous edge pairs may arise. It would lead to incorrect mid-edge generation. The simple model depicted in Fig. 12(a) comprises only planar and cylindrical surfaces, and judgment can be made using distance criteria and overlap criteria. The complex model in Fig. 12(b) is manufactured using stamping processes. It contains numerous intricate spline and trimming surfaces, which are quite common in CAE simulation analysis models. When employing distance criteria to assess surface pairs in complex models, the lookup of variables H, L, and W in the formula becomes ambiguous. This leads to the failure of the distance criteria. Judging using the overlap criteria consumes a significant amount of computational resources. The efficiency of using the overlap criteria for judgment on complex models is extremely low. A global surface pair generation algorithm based on lateral decomposition is proposed, and this method will be described in detail below.

4.2.2. Attributed adjacency graph(AAG)

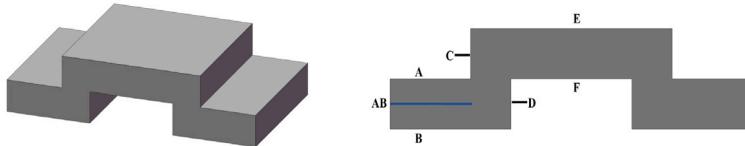
The B-Rep format is a popular way of representing 3D objects' geometry and topology in CAD. The main requirement for B-Rep is to depict entities based on their geometry and topology. The size, position, and shape of a solid, such as point coordinates, edge parameters, and face parameters, are all referred to as geometric information. Conversely, topological information pertains to the intricate interconnections among entities. B-Rep aids in comprehensively documenting the interconnections between entities. FAG offers a concise and effective depiction of the interconnections between adjacent faces in a B-Rep model. The graph structure allows for efficient traversal and analytical representation by encoding the relationships between entities, such as edges and faces. For example, if two faces have an edge in the B-Rep model, there will be an edge connecting the corresponding nodes in the FAG. FAG is a graphical representation method used to depict the topological relationships between geometric objects. It primarily describes the adjacency relationships between faces, indicating which faces are adjacent, share edges, or share vertices. FAG typically does not contain attribute information of objects but rather focuses on the topological connections between geometric objects. Based on FAG, it is straightforward to construct an Attribute Adjacency Graph (AAG), which is shown in Fig. 13, for B-Rep (Boundary Representation) structures. AAG (Attribute Adjacency Graph) is a graphical representation method used to depict attribute relationships between geometric objects. It delineates the connectivity and attribute information among geometric objects, such as the connection between edges, faces, and their shared properties. AAG not only portrays the topological relationships between objects but also encompasses their attribute information, such as dihedral angle, face type, edge type, and so forth.



(a) Mid-surface of a face-pair for thinner section

(b) Mid-surface of a face-pair for thicker section

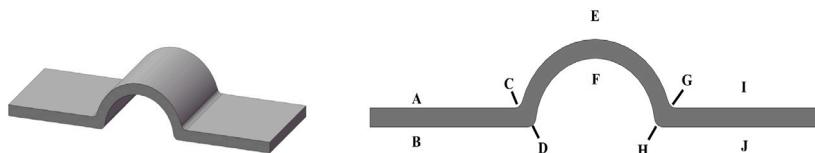
Fig. 8. An example of the distance criterion.



(a) A model that satisfy overlap criteria

(b) Front view of (a)

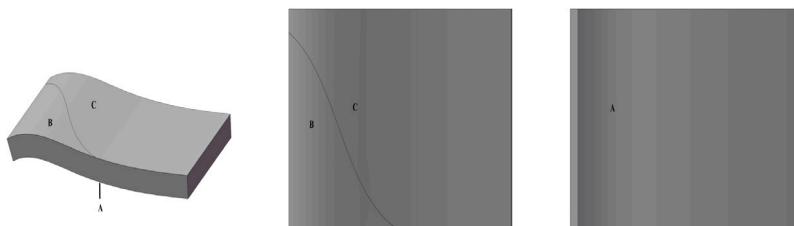
Fig. 9. An example of the overlap criterion.



(a) A model that satisfy overlap criteria

(b) Front view of (a)

Fig. 10. An example of a strictly matching face-pair model.



(a) A model that satisfy overlap criteria

(b) Top view of (a)

(c) Bottom view of (a)

Fig. 11. An example of a non-strictly matching face-pair model.

4.2.3. Lateral face finding and removal

An obvious geometric feature of thin-walled parts is the presence of distinct end faces, as shown by the orange face in Fig. 12. Numerous unique characteristics of the lateral face of thin-walled sections:

(1)The lateral face exhibits distinct geometric shapes, angled approximately 90 degrees relative to adjacent faces.

(2)The majority of the lateral faces are slender. These faces play little to no role in the mid-surface abstraction process and instead add

extra costs. The removal of all such end faces will result in numerous advantages when constructing a global face-pair.

(1)By simplifying the geometric topology, the complexity of constructing the face is reduced.

(2)It is possible to rule out the influence of incorrect geometry and topology on the final outcome.

(3)The construction of global faces diminishes the need for precisely matching faces during the mid-surface abstraction process.

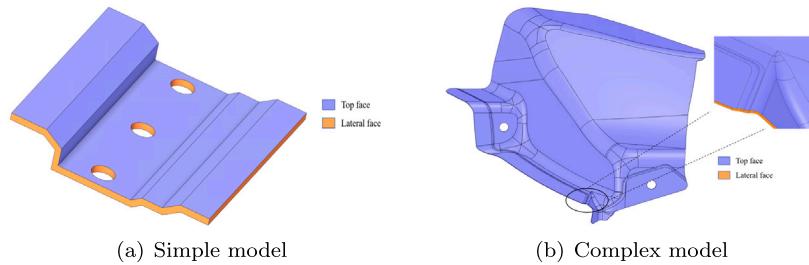


Fig. 12. Two models to describe definitions.

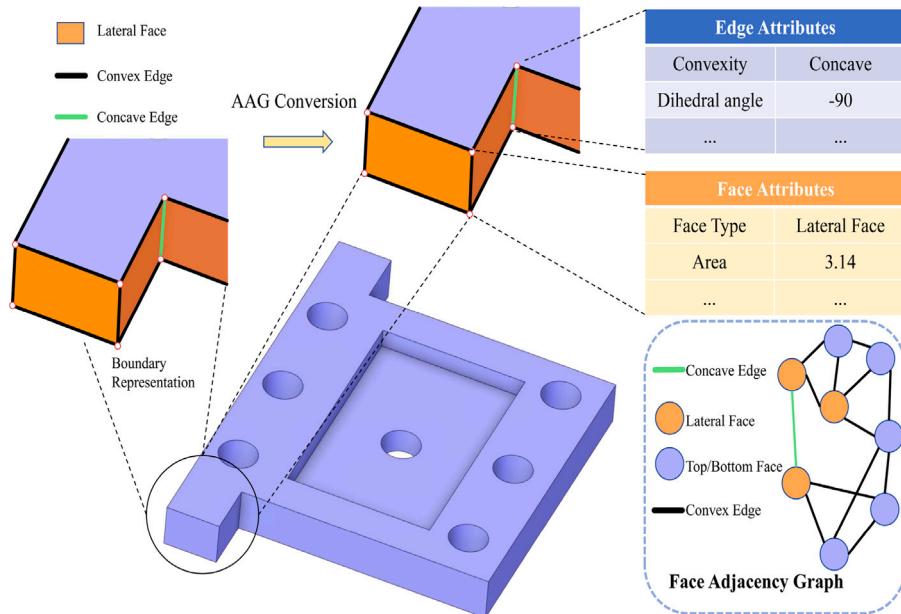


Fig. 13. The structure of geometric Attributed Adjacency Graph (AAG).

(4) By refraining from employing the ray method for facial recognition, one can enhance efficiency and minimize memory consumption.

Algorithm 2 demonstrates the process of searching lateral faces starting from the face with the maximum area of the thin-wall solid.

In thin-walled solid structures, the lateral faces are generally slender and narrow, while the largest face area on the thin-walled solid typically corresponds to the effective mid-surface. The algorithm starts by identifying the initial face, and then its adjacent faces F_{nei} are found using the attributed adjacency graph (AAG), along with the dihedral angle Ang between them. Next, the algorithm compares the angle Ang with the maximum allowed angle Ang_{max} . If $\text{Ang} \leq \text{Ang}_{max}$, F_{nei} is considered an effective face of face-pair. Otherwise, F_{nei} is classified as a lateral face. The process then recursively sets F_{nei} as the new initial face, repeating the above method until all faces have been evaluated.

4.3. Virtual topology method

4.3.1. Principle of the virtual topology method

Feature removal is a common approach for simplifying topology. The idea is to first look for features such as holes, cavities, chamfers, and chamfers in the model through the use of feature recognition, and then to analyze the local topology and use simplification algorithms for intersecting and stitching surfaces to remove features. However, not all true models can be characterized by good feature removal. In the design of thin walled parts such as molds and stamped castings, the number of small transition surfaces is large, and if those surfaces are included in the mesh as well, overall, the mesh quality is poor, primarily due to the uneven mesh, which often leads to unstable subsequent finite-element solutions.

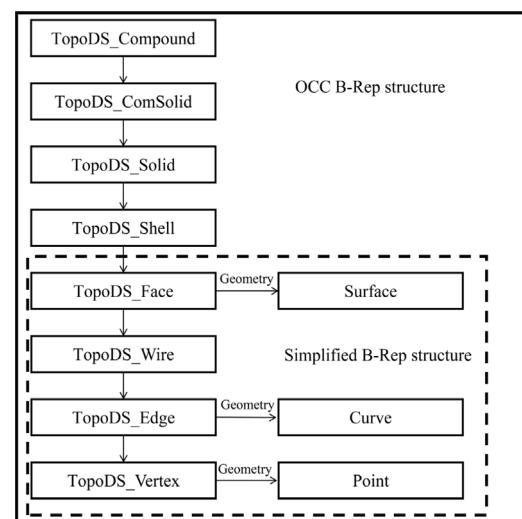


Fig. 14. B-Rep structure in OCC.

The basic structure for describing topological information in mainstream commercial CAD engines is generally based on B-Rep. The B-Rep structure described based on OpenCASCADE [18] is depicted in the following figure. As shown by the content of the solid wireframe in Fig.

14, the B-Rep structure illustrates relatively complex topological information. In this paper, our focus lies on operations concerning surface models, thus we adopt a simplified geometric-topological structure, as shown by the contents of the dotted box in Fig. 14. Within the simplified surface structure, the supporting surface employs free-form parametric surfaces, defining the geometric information of the surface. The boundaries of the surface consist of multiple curves connected end-to-end on the supporting surface. The combination of these curves is referred to as a wire.

The B-Rep structure describes the topology of the model [19,20]. On the basis of B-Rep, two new topological structures, “virtual faces” and “virtual edges” are introduced. When representing topological relationships, these two new topological objects are no different from conventional edges and faces, with the only distinction being that virtual faces and virtual edges do not have directly corresponding geometric objects. Through the structure illustrated in Fig. 15, virtual faces and virtual edges directly point to actual faces and edges. A virtual face can point to multiple actual faces, each of which has its own loops. The edges composing these loops can be actual edges or virtual edges. Similarly, a virtual edge can point to multiple actual edges, each having its own independent topological structure.

In CAE software, simplification operations for surface features are categorized into real topology operations and virtual topology operations [21,22]. For example, when merging two adjacent geometric parametric surfaces, there are significant differences between real and virtual topology operations. In real topology operations, a new representation of the geometric parametric surface needs to be formed through processes such as resampling and refitting, and its topology structure needs to be updated accordingly. In virtual topology operations, only the topology needs to be updated, and a new virtual surface is defined based on the updated topology. The geometric information

of the virtual geometric surface still depends on the old surface. Real topology operations involve a large amount of geometric computation, and this operation may not necessarily be reversible. Additionally, factors such as sampling accuracy and fitting errors can lead to geometric errors between the new and old surfaces. Virtual topology operations do not require a significant amount of geometric computation. This operation is reversible and does not introduce geometric errors.

The specific virtual topologization process is illustrated in the following figures. As shown, Fig. 15(a) represents the original state, and its corresponding B-Rep representation is shown in Fig. 15(d). From surfaces to vertices, it is a standard B-rep structure. In Fig. 15(b), surfaces F_1 and F_2 are merged to form a virtual face VF_1 . In the B-rep structure, the boundary edges of VF_1 include six edges: E_1 , E_2 , E_3 , E_4 , E_5 , and E_6 . The original edge E_7 is also removed in the virtual topological structure. In figure Fig. 15(c), virtual operations are used to merge E_1 and E_2 to form the virtual edge VE_2 . The two vertices of VE_2 are V_1 and V_3 . Similarly, E_4 and E_5 are merged to form the virtual edge VE_3 . The two vertices of VE_3 are V_4 and V_6 . The number of boundary edges of VF_4 is reduced from six to four, which are VE_2 , E_3 , VE_3 , and E_6 . The original vertices V_2 and V_5 , along with edges E_1 , E_2 , E_4 , and E_5 , are removed in the virtual topological structure.

4.3.2. Automatic virtual topology

In computer graphics, finite element analysis, and geometric modeling, the 2D aspect ratio (2DAR) is a crucial metric for evaluating the quality of triangular meshes. The aspect ratio is typically defined as the ratio of the length of the longest side to the shortest side of a triangle. For a triangle ABC with side lengths a , b , and c (where $a \leq b \leq c$), the aspect ratio 2DAR can be expressed as:

$$2DAR = \frac{c}{a} \quad (3)$$

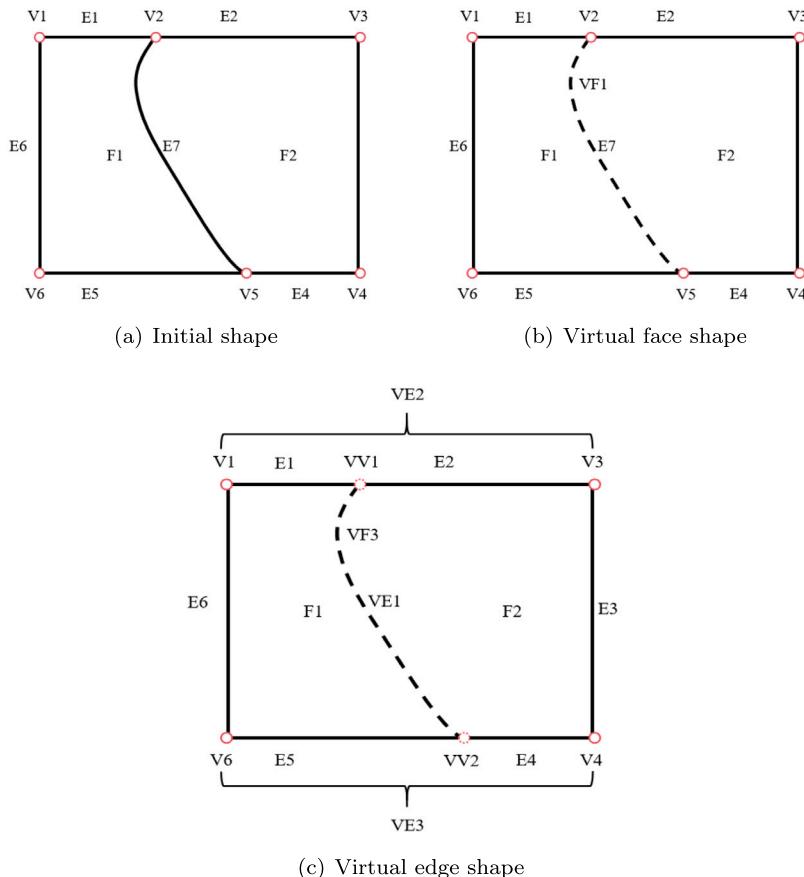


Fig. 15. Model with virtual operation.

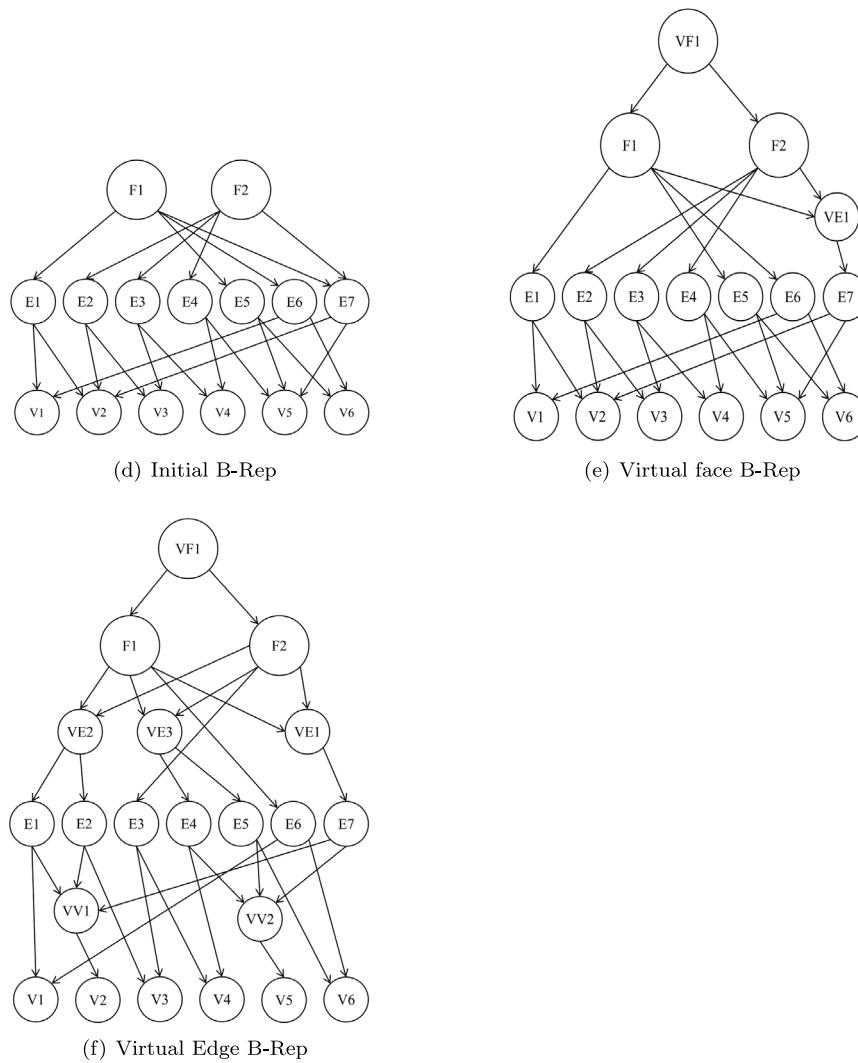


Fig. 15. (continued).

High-quality meshes should ideally consist of elements that are close to equilateral triangles, which have an aspect ratio close to 1.00. Equilateral triangles are considered the optimal shape because they distribute errors uniformly in all directions, thus enhancing the overall quality of the mesh and the reliability of numerical computations.

Complex geometric features can significantly impact mesh quality. In the face shown in Fig. 16(a), there are some short edges in the face model. When meshing this face, poor-quality elements will appear at these short edges, as illustrated in this figure. The local 2D Aspect Ratio of the mesh reaches as high as 9.95e+01. In the face shown in Fig. 16(b), there are short-distance adjacent edges. When meshing this face, poor-quality elements will appear at these adjacent edges, as depicted in this figure. The local 2D Aspect Ratio of the mesh reaches 7.02e+00. Meanwhile, the face in Fig. 16(c) contains sharp feature angles. When meshing this face, poor-quality elements will appear at these sharp feature angles, as shown in this figure. The local 2D Aspect Ratio of the mesh reaches 1.36e+01.

To effectively perform virtual topology operations, we define and control the properties of the edges by introducing a suppression mechanism based on the following three indicators:

Min Edge Length refers to the shortest edge within a topological structure. The length of an edge plays a crucial role in topology simplification, where shorter edges are often considered insignificant for feature extraction. By setting a threshold value, edges shorter than this threshold are regarded as noise or irrelevant and are thus suppressed

or removed. The formal definition is as follows:

$$L_e \leq L_{\min} \quad (4)$$

Where L_e denotes the length of edge e , and L_{\min} is the minimum length threshold.

The Edge Suppression Angle is defined as the angle between two adjacent edges. When the angle between two edges is smaller than the threshold, they are considered to have minimal impact on the topology and are likely redundant or insignificant. By controlling the edge suppression angle, we can effectively eliminate redundant connections, thereby improving the simplicity of the topological structure. The condition for suppression is as follows:

$$\theta_{e_1, e_2} \leq \theta_{\min} \quad (5)$$

Where θ_{e_1, e_2} represents the angle between edges e_1 and e_2 , and θ_{\min} is the minimum angle threshold.

Edge Adjacent Distance refers to the spatial distance between two edges. When the distance between two edges is smaller than the predefined threshold, these two edges are considered as parts of the same topological feature region, thereby reducing the complexity of their connections. This distance metric helps eliminate edges that do not significantly contribute to the structure during the simplification process. The condition is expressed by:

$$d_{e_1, e_2} \leq d_{\min} \quad (6)$$

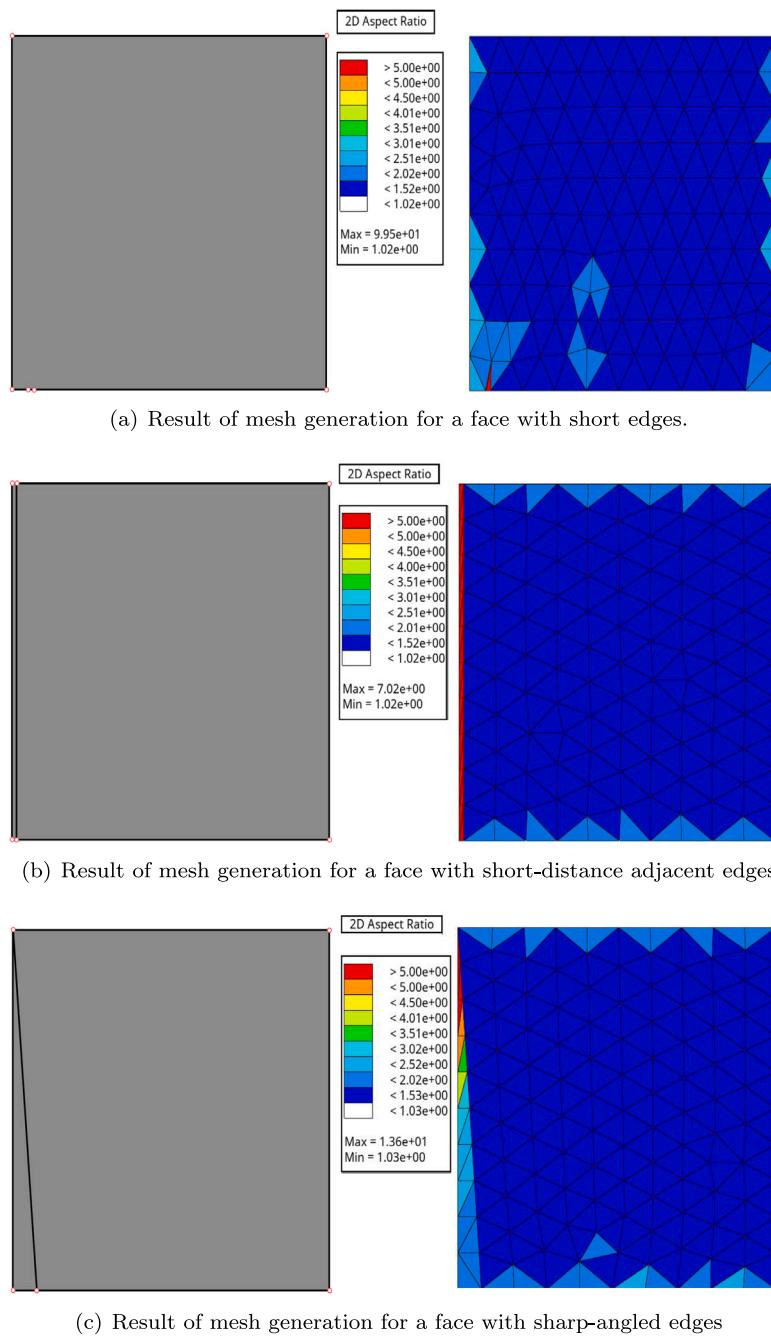


Fig. 16. Result of mesh generation for geometric features requiring automatic virtual topology operations.

Where d_{e_1, e_2} represents the distance between edges e_1 and e_2 , and d_{\min} is the minimum adjacent threshold.

The proposed method combines the above three indicators to automatically perform topology simplification. The process can be summarized as follows: (1)Adjacent Distance Suppression: For edges that are spatially too close to each other, as determined by the adjacent distance threshold, are suppressed. The adjacent faces are then merged into a parameterized surface, ensuring the topology remains compact and meaningful. (2)Angle Suppression: Edges with an angle smaller than a predefined threshold angle between adjacent edges are suppressed to eliminate redundant connections, thus simplifying the geometric structure. (3)Edge Length Suppression: Edges with a length smaller than a specified threshold are removed to enhance mesh quality and improve computational efficiency by reducing unnecessary details. The

pseudocode for the automatic virtual topological operation is shown in Algorithm 3.

4.4. Mid-surface mesh generation

4.4.1. Top face mesh generation

We use Gmsh 4.10.3 [23] as the meshing tool to generate the meshes for the top face. When generating meshes for geometries that include virtual topological structures, only the geometric entities present in the topological structure are discretized. For example, the vertices of the virtual edge VE_2 are V_1 and V_3 , with the original point V_2 being replaced by the virtual point VV_1 . At the location of the virtual point VV_1 , a mesh node will not necessarily be generated. Similarly, the boundary of the virtual face VF_1 does not include E_7 ; instead, E_7 is replaced by the virtual edge VE_1 . During mesh generation, no

Algorithm 2: Lateral face search algorithm based on AAG

Input: F_{set} : Thin-walled part faces sorted in descending order of area, Ang_{max} : The largest angle between F_c and its neighboring face; T_1 : thickness threshold provided by the user;

Output: $LFPS$: the face-pair list;

- 1 $F_{set} \leftarrow$ faces sorted by area from largest to smallest;
- 2 **for** each face $F_i \in F_{set}$ **do**
- 3 | $T \leftarrow$ shortest distance between F_i and its corresponding bottom face;
- 4 | **if** $T \leq T_1$ **then**
- 5 | | Set $F_c = F_i$ as the initial top face;
- 6 | | **break**;
- 7 | **else**
- 8 | | **continue**;
- 9 | **end**
- 10 **end**
- 11 **if** no valid face F_i is found **then**
- 12 | | Prompt user to input $F_c = F_{user}$ as the initial top face;
- 13 **end**
- 14 initial $LFPS = NULL$, F_{nei} : Neighbor face of F_c , $Angle = 0$: dihedral angle between F_c and its neighboring face;
- 15 **Function** $FindLateralFace(F_c, Ang, LFPS)$:
- 16 **for** not end of F_{nei} **do**
- 17 | | $Angle \leftarrow AAG$;
- 18 | | **if** $Angle > Ang_{max}$ **then**
- 19 | | | $FindLateralFace(F_{nei}, Ang, LFPS)$;
- 20 | | **else**
- 21 | | | Add F_{nei} to $LFPS$;
- 22 | | **end**
- 23 **end**
- 24 **end**

Algorithm 3: Automatic Virtual Topology Process for FAG Graph

Input: FAG, G_{FAG}

Output: FAG with automatic virtual topology edges, G_{FAG} with virtual edges

- 1 **Step 1: Adjacent Distance Suppression**
- 2 **for** each edge $e \in G_{FAG}$ **do**
- 3 | | **if** distance between adjacent edges of e is smaller than proximity distance threshold **then**
- 4 | | | suppress edge e
- 5 | | | merge adjacent faces into a parameter surface
- 6 **end**
- 7 **end**
- 8 **Step 2: Angle Suppression**
- 9 **for** each pair of adjacent edges $e_1, e_2 \in G_{FAG}$ **do**
- 10 | | **if** angle between e_1 and e_2 is smaller than angle threshold **then**
- 11 | | | suppress edge e_1 or e_2
- 12 | | | remove redundant connection
- 13 **end**
- 14 **end**
- 15 **Step 3: Edge Length Suppression**
- 16 **for** each edge $e \in G_{FAG}$ **do**
- 17 | | **if** length of e is smaller than edge length threshold **then**
- 18 | | | remove edge e
- 19 **end**
- 20 **end**
- 21 **return** Updated FAG with virtual topology edges

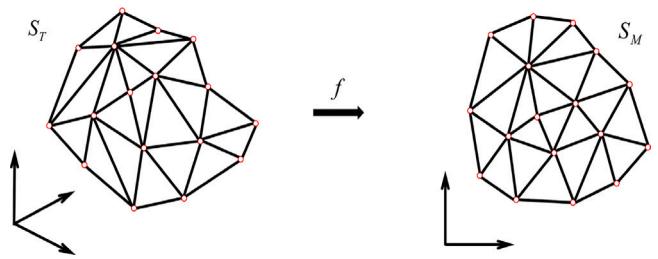


Fig. 17. Piecewise linear mapping of a triangular mesh.

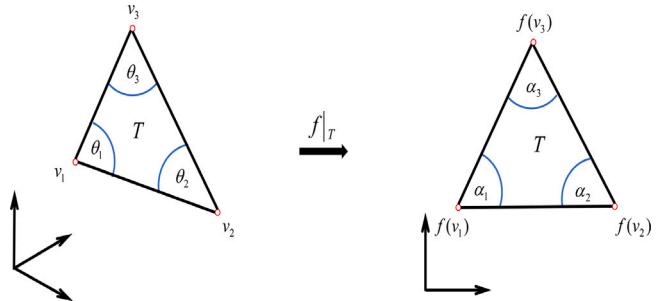


Fig. 18. Atomic map between a mesh triangle and the corresponding parameter triangle.

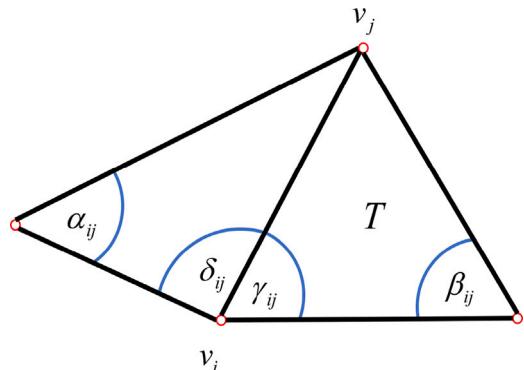


Fig. 19. Angles for the discrete harmonic map and the mean value coordinates.

mesh nodes will be created at the original boundary E_7 . Map the discretized triangular surfaces onto a parameter plane [24,25]. Reparameterize the triangular mesh on the parameter plane to generate a triangular mesh model with a unified parameter expression [26]. Map the initial mesh surface S_T onto the unit circle plane S_M . The discretized mesh surface S_T is represented piecewise-linearly by the collection of triangular meshes $T = \{T_1, \dots, T_N\}$ [26]. In the triangular mesh, adjacent triangles share the same nodes and mesh edges. The node set $V = \{V_1, \dots, V_M\}$ and the mesh edge set $E = \{E_1, \dots, E_k\}$ comprise all the nodes and mesh edges of the piecewise linear surface S_T . The boundary of the piecewise linear surface S_T is typically a closed polygon, and its boundary node set is represented as $VB = \{V_{b1}, \dots, V_{bn}\}$. Its interior node set is represented as $VI = \{V_{i1}, \dots, V_{in}\}$. As shown in Fig. 17, We need to find a suitable piecewise linear mapping function $F(x)$ to achieve a discrete harmonic mapping from the triangle mesh S_T to S_M . This process requires ensuring that each triangle remains linearly continuous during the mapping process. The expression of the piecewise linear mapping function at each node is:

$$F|_T(v_i) = F(v_i) \quad \forall v_i \in V \quad (7)$$

The discrete harmonic mapping process consists of two parts [27–30]:

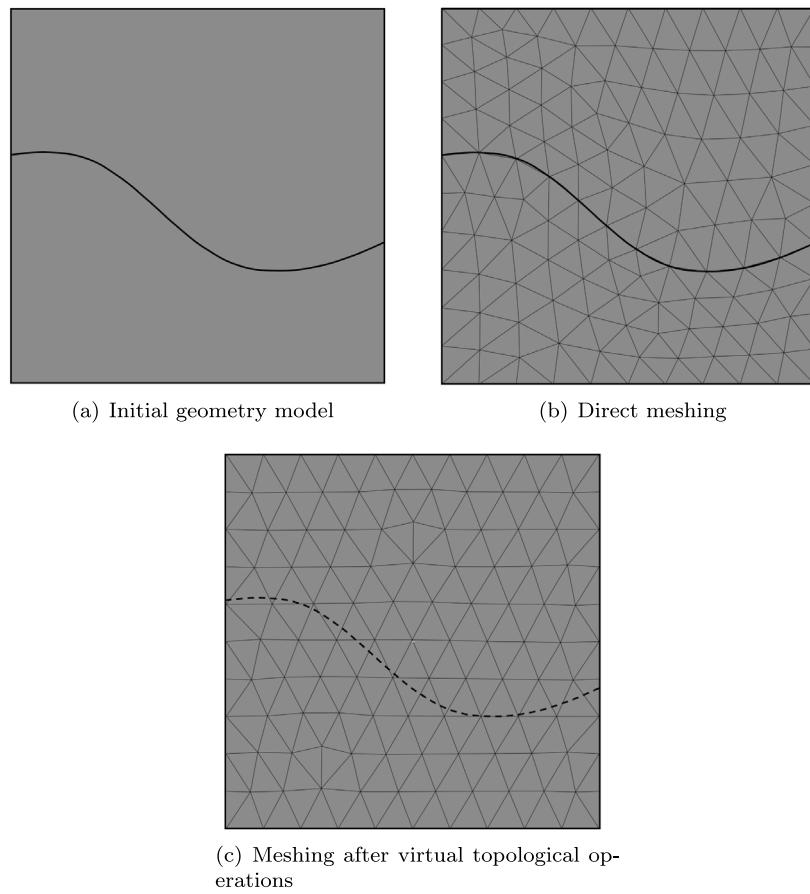


Fig. 20. Comparison of mesh generation results: Direct Meshing vs. Meshing after virtual topological operations.

(1) For the discrete surface boundary nodes V_b , let $F(v_i) = F(v_0)$, mapping all nodes on the boundary of the entire discrete mesh surface S_T to the boundary of the S_M polygon [31,32].

(2) Determine a piecewise linear function f to compute the Dirichlet minimal energy on the surface, with the boundary condition set to $f(v_i) = f(v_0)$. The expression for the Dirichlet energy of the triangle mesh is as follows:

$$E(f) = \frac{1}{2} \int_{S_T} \|\nabla S_T f\|^2 \quad (8)$$

To discretize the above formula using the finite element method, referring to Fig. 18, we compute the gradient values of the triangle mesh $t = [v_0, v_1, v_2]$ as shown in Eq. (9).

$$\begin{aligned} 2 \int_T \|\nabla T f\|^2 &= \cot \theta_3 \|f(v_0) - f(v_1)\|^2 \\ &+ \cot \theta_2 \|f(v_0) - f(v_2)\|^2 + \cot \theta_1 \|f(v_1) - f(v_2)\|^2 \end{aligned} \quad (9)$$

The minimization of the Dirichlet energy problem can be transformed into a quadratic minimization problem and simplified into a linear equation system to solve. The linear equation system to be solved is expressed as follows in Formula Eq. (10).

$$\sum_{j \in N_i} W_{ij} (f(v_j) - f(v_i)) = 0, \quad v_i \in V_I, \quad (10)$$

where

$$W_{ij} = \cot \alpha_{ij} + \cot \beta_{ij} \quad (11)$$

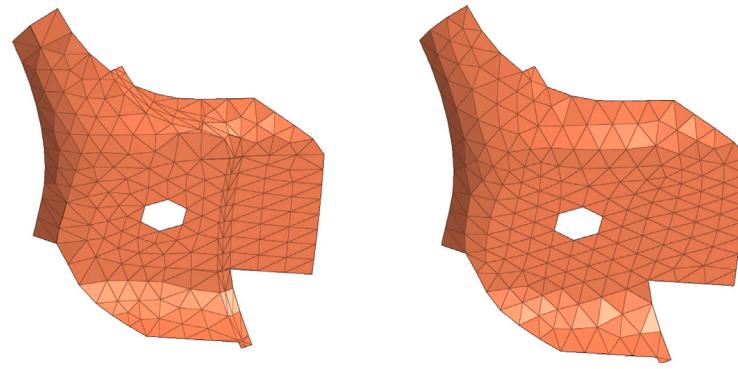
and the angles α_{ij} and β_{ij} are shown in Fig. 19. All nodes on the mesh surface have been assigned numbers. Node v_i is an interior node in the triangular mesh T , and its adjacent node set is denoted by N_i . The linear equation system is transformed into matrix form, and the

assembled coefficient matrix is symmetric positive definite. Therefore, this linear equation system has a unique solution. Based on the set boundary conditions, the parameter coordinates u and v of the interior nodes on the parametric surface S_M can be obtained through two solve operations.

Provided that the domains (for both the source domain S_T and target domain S_M) are convex, and the Rado–Kneser–Choquet theorem [30] are satisfied, harmonic mapping can be bijective. However, when the mesh satisfies the Delaunay condition, the final parametrization result will be bijective even with obtuse triangles. Since Gmsh strictly ensures that the mesh satisfies the Delaunay condition before performing harmonic mapping on the 3D mesh, a one-to-one mapping can be guaranteed during the entire parametrization process.

Through the above method, the triangular mesh on surface S_M now possesses a unified parameter space representation. Utilizing the Advancing Front Technique(AFT) combined with the Delaunay method [33,34], operations such as mesh partitioning and optimization can be performed in this parameter space. Finally, using interpolation methods, the triangular mesh on the parameter space S_M is remapped back to the surface S_T , resulting in a high-quality triangular mesh (see Fig. 20). Fig. 21 shows the results of mesh generation after direct meshing and after virtual topological operations. As shown in Fig. 21(a), figure depicts the original geometric model, which includes two faces. Fig. 21(b) shows the result of directly applying triangular mesh generation to the model in Fig. 21(a), resulting in nodes being generated on the common edge of the two faces. In figure Fig. 21(c), the model from figure Fig. 21(a) has undergone virtual topological operations before triangular mesh generation. As a result, no nodes are generated on the common edge of the two faces.

Next, the advantages of mesh generation after virtual topological operations will be demonstrated using a complex model. In Fig. 22(a),



(a) Meshing before virtual topological operations

(b) Meshing after virtual topological operations

Fig. 21. The meshing results of the two models.

the original geometric model contains numerous elongated surfaces and intricate geometric adjacency features. When a mesh size of 5 is used for triangular mesh generation, the resulting mesh model is shown in Fig. 21(a). The 2D aspect ratio of the mesh model in Fig. 21(a) was examined, and the results are presented in Fig. 23(a).

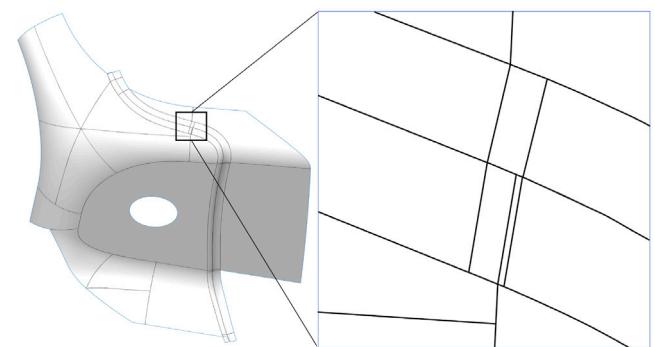
Virtual topological operations were performed on the elongated faces and complex adjacency features in the model shown in Fig. 22(a), resulting in the geometric model shown in Fig. 22(b). A mesh size of 5 was used for triangular mesh generation, and the resulting mesh model is shown in Fig. 21(b). The 2D aspect ratio of the mesh model in Fig. 21(b) was examined, and the results are presented in Fig. 23(b).

Triangles with high aspect ratios (greater than 5) typically exhibit “slender” or “flat” shapes. Such triangles can cause problems in numerical computations, such as large condition numbers, leading to instability and reduced accuracy in numerical methods. For example, in finite element analysis, elements with high aspect ratios can result in stiffness matrices with large condition numbers, adversely affecting solution accuracy. Consequently, such triangular meshes are generally considered invalid. In Fig. 23(a), the maximum 2D aspect ratio of the triangular mesh reached $4.18e + 01$, far exceeding $5.00e + 00$. The mesh model in Fig. 23(a) contains a total of 402 triangular elements, with 15 valid triangles, accounting for 4% of the total mesh. In Fig. 23(b), the maximum 2D aspect ratio of the triangular mesh is $2.15e + 00$, and all triangular elements are valid.

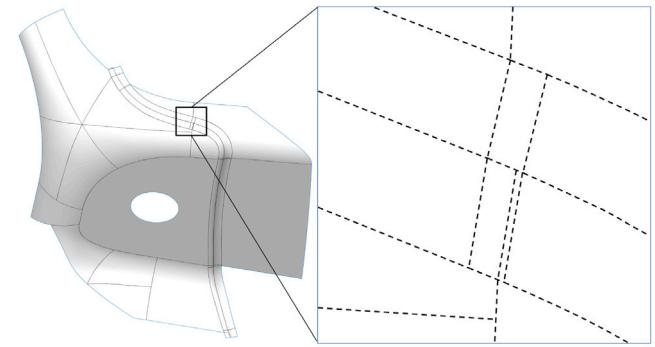
4.4.2. Projection for generating the mid-surface mesh

Once a high-quality top surface mesh is obtained, generating a high-quality mid-surface mesh becomes straightforward. The nodes of the top surface mesh are considered as initial points, and a hybrid method minimizing midpoint calculation errors is used to determine the mid-surface mesh nodes. As Fig. 24(a), assume surfaces F_1 and F_2 are on the top and bottom surface sets, respectively. V_1 is a node on F_1 . The proposed method for determining the corresponding node on F_2 is as follows:

- (1) Project V_1 onto F_2 to obtain the projection point V_3 .
 - (2) Extend the normal at V_1 in the reverse direction and find its intersection point V_2 with F_2 .
 - (3) Generate a plane through V_1 , V_2 , and V_3 . On this plane, construct the angle bisector of $\angle V_2V_1V_3$, and find its intersection point V_4 with F_2 .
 - (4) Generate the midpoint V_m of the edge V_1-V_4 , and consider V_m as a node of the mid-surface mesh.
 - (5) Use the same method to generate the remaining nodes. Construct the topology of the mid-surface mesh based on the topology of the top surface mesh, resulting in the mid-surface mesh model.
- Following the aforementioned steps, a high-quality mid-surface mesh model can be obtained. As shown in Fig. 25, an example is used to demonstrate the mid-surface mesh results produced by this method.



(a) Initial geometry model

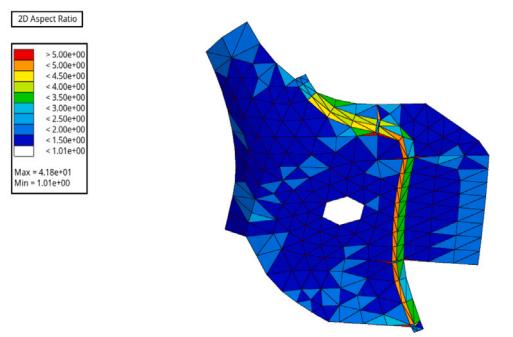


(b) Geometry model after virtual topological operations

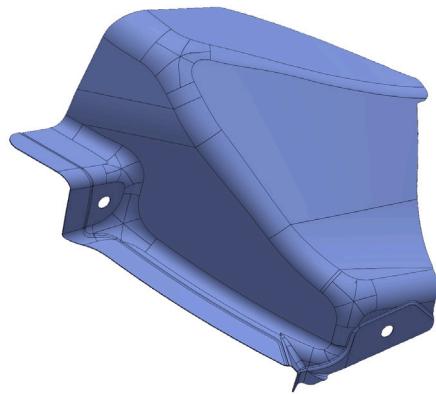
Fig. 22. Initial geometry model and Geometry model after virtual topological operations.

4.5. Composition of mid-surface meshes of maximal volumes to the mid-surface meshes of original solid model

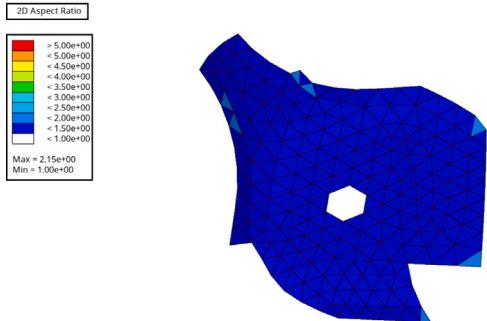
The mid-surface mesh of the original solid model is generated using the mid-surface mesh of the largest volume. However, as shown in Fig. 26, not all of the generated meshes are valid for the original model. Some invalid meshes need to be identified and adjusted. To make the mid-surface mesh valid, there should be a unique corresponding part in the original solid model for the mid-surface mesh. If a surface in the original model can correspond to a mesh region of the mid-surface mesh, then this mid-surface mesh is valid. Otherwise, the mid-surface mesh contains invalid parts that need to be adjusted. For example, in Fig. 26, the mid-surface mesh in region MF_1 corresponds to the



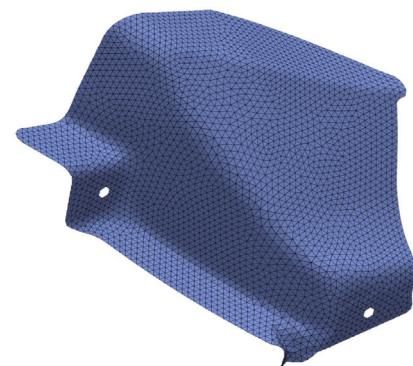
(a) Meshing after virtual topological operations



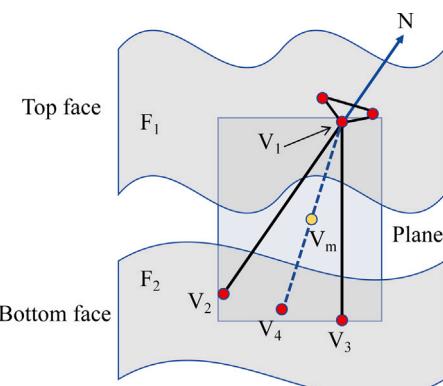
(a) Thin-wall solid



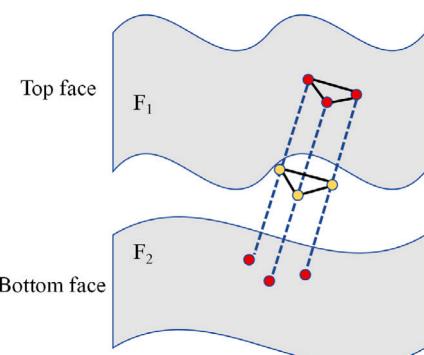
(b) Meshing after virtual topological operations



(b) Mid-surface mesh

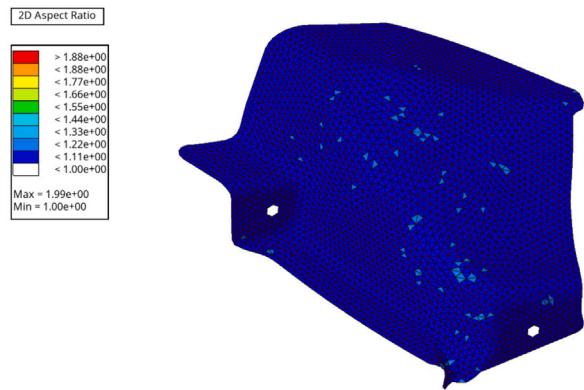


(a) A hybrid method minimizing midpoint calculation errors



(b) Mid-surface mesh generate

Fig. 24. Mid-surface mesh generation method.



(e) Mid-surface mesh quality

Fig. 25. Mid-surface mesh generation.

geometric surfaces F_1 and F_2 in the original model. Therefore, the mesh in region MF_1 is valid and does not need to be adjusted for the time being. However, the mid-surface mesh in region MF_2 cannot be found corresponding to any geometric surface in the original model, making the mesh in region MF_2 invalid. The mesh in region MF_2 needs to be removed and adjusted.

Let:

N be the set of all nodes.

T be the set of all triangular meshes.

Each triangular mesh consists of three nodes, so we can define a triangular mesh t as a set of three nodes $\{n_1, n_2, n_3\}$, where $n_1, n_2, n_3 \in$

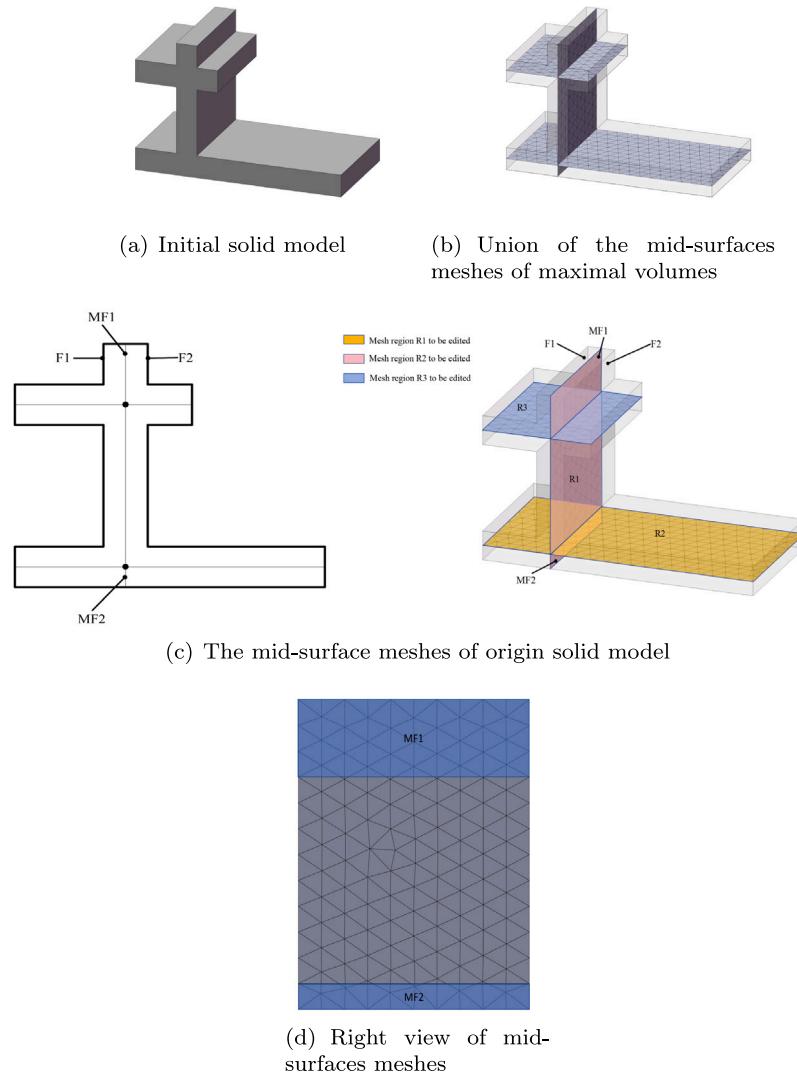


Fig. 26. Identify and adjust invalid meshes.

N . That is:

$$t = \{n_1, n_2, n_3\}$$

Thus, the set T can be expressed as:

$$T = \{t_i \mid t_i \subseteq N \text{ and } |t_i| = 3\}$$

Next, a mesh surface can be defined as a set containing multiple triangular meshes. Let M be the set of all mesh surfaces, where each mesh surface m is a subset of triangular meshes, i.e.,

$$m \subseteq T$$

Thus, the set M can be expressed as:

$$M = \{m_i \mid m_i \subseteq T\}$$

The relationship between nodes, triangular meshes, and mesh surfaces can be represented using the sets N , T , and M : N is the set of nodes.

T is the set of all triangular meshes formed by the nodes.

M is the set of all mesh surfaces formed by the triangular meshes.

If all three nodes of triangle t are located within region MF_2 , then this triangular mesh should be removed from set T . The resulting mesh after removal is shown in Fig. 27(a).

In addition, it is necessary to ensure the consistency between the generated mid-surface mesh and the original geometry. That is, at non-manifold boundaries, the mesh should be connected using shared nodes. As shown in Fig. 27(b), the intersection of region R_1 with regions R_2 and R_3 results in the intersection lines NME_1 and NME_2 , which are non-manifold edges. For the case of regions R_1 and R_2 , operations for node merging and mesh remeshing are performed. For the set of nodes in MF_2 , if the meshes connected to a node are not entirely contained within MF_2 , then node will be labeled as an initial node. The initial nodes located in the MF_2 region are projected onto NME_1 , yielding corresponding projected nodes. The specific process is illustrated in Fig. 27(c). After projection, the mesh of region R_1 is updated, as shown in Fig. 27(d). The bottom view of Fig. 26(c) provides further details. As shown in Fig. 27(e), for each projected node on NME_1 , the nearest node in region R_2 is searched and labeled as a node to be moved. The node to be moved is then merged by mapping it to the corresponding node on NME_1 , achieving node merging between regions R_1 and R_2 . The resulting merged nodes are depicted in Fig. 27(f). However, after these operations, the quality of the mesh may be affected. Therefore, a remeshing process is required for meshes of regions R_1 and R_2 . First, the mesh of region R_2 is remeshed, followed by remeshing of region R_1 . It is notable that when remeshing region R_1 , the nodes on NME_1 remain unchanged. After the remeshing process,

Table 1
Quality of the mid-surface meshes abstraction for the example parts.

	Mesh count	Max aspect	Min Jacobian	Proportion of high-quality meshes
Model A	4320	1.90	–	100.0%
Model B	4514	3.22	–	100.0%
Model C	60 545	3.26	–	100.0%
Model D	25 179	1.93	–	100.0%
Model E	12 383	3.03	–	100.0%
Model F	3564	–	0.53	99.8%
Model G	10 292	–	0.54	99.9%
Model H	9475	–	0.28	99.3%

the mesh of region R_2 is shown in Fig. 27(g). Similarly, the same operations for node merging and mesh remeshing can be applied to regions R_1 and R_3 .

After the above operations, the initial mid-surface mesh of the original solid model is obtained. However, the quality of such a mesh may be relatively low, and the generated mesh needs to be further optimized. The result after optimization is shown in Fig. 27(h).

5. Results and discussion

The method proposed in this paper is implemented on a Windows PC using C++, OpenCASCADE geometry modeling kernel, Gmsh and Visual Studio 2017. Fig. 28 shows some examples of mid-surface meshes generation. Models A and B demonstrate the results of non-manifold mid-surface meshes generation, while Models C and D illustrate the mid-surface meshes generation results for complex automotive components. The method was also tested for generating hybrid mid-surface meshes consisting of triangular and quadrilateral elements. Notably, the method shows good performance in hybrid meshes as well. Models E and F exhibit the mid-surface meshes generation results for parts with a mixture of triangular and quadrilateral meshes.

As shown in Fig. 29, we used a model highly similar to the one used by Quadros for mid-surface mesh generation. Fig. 29(a) shows the initial thin-walled solid model. As shown in Fig. 29(b), it is clear that the mid-surface mesh obtained using Quadros's method exhibits significant mesh distortion at the connections, and the local mesh quality is poor. As shown in Fig. 29(c), the mid-surface mesh model obtained using our approach handles this situation well. Table 1 presents the quality of the generated meshes. For triangular meshes, the aspect ratio is used for evaluation, with an aspect ratio greater than 5 indicating a failed mesh. In the test models, the aspect ratios of mid-surface meshes generated by this method were all less than 5. For quadrilateral meshes, the Jacobian metric is used for evaluation. The closer the Jacobian is to 1, the better the mesh quality. A Jacobian less than 0.7 is considered to significantly impact the computational accuracy. In the test models, the proportion of meshes with a Jacobian less than 0.7 in the hybrid mid-surface meshes generated by this method was 0.1%. As is well known, mid-surface abstraction is a common geometric operation in CAE model construction. However, the abstraction results of mid-surfaces in engineering are not unique, and there is currently a lack of a strict definition for mid-surfaces. This leads to robustness issues when developing mid-surface abstraction systems. For solid models, some mid-surface abstraction results can be ambiguous. As shown in Fig. 30, a model frequently discussed in mid-surface abstraction, figure b displays three possible mid-surface mesh results. All three results appear reasonable, but selecting the optimal mid-surface mesh is subjective. The method proposed in this paper can directly generate the first model, which is also more acceptable. The second model requires consideration of the interaction between the largest volume model's face-to-face and side-to-side after decomposition, which does not conform to face detection results. General face detection algorithms

cannot satisfy this situation either. It is believed that in certain cases, user intervention is necessary. Robinson [12] and Nolan [13] proposed an automated mixed-dimensional modeling method that reduces thin-walled structures to shell elements, slender structures to beam elements, while using 3D solid elements for complex regions. As shown in Fig. 31, the shell elements represent the thickness of thin-walled structures through mid-surface abstraction, the beam elements simplify slender structures by applying cross-sectional properties, and the solid elements accurately capture the stress in complex regions. By using multi-point constraints (MPC), these different dimensional elements are precisely coupled at the interfaces, reducing computational costs while maintaining accuracy and efficiency in the analysis. Therefore, to make the mid-surface abstraction method practically useful, it needs to abstract mid-surfaces from which alternative representations can be generated. Currently, this method is limited to solid models composed only of analytical surfaces. The largest volume decomposition is generated based on face expansion and cutting. However, since there is no unique method to expand free surfaces, the largest volume decomposition has not yet been achieved for solid models with free-form surfaces.

As previously mentioned, a modified projection algorithm was used for the generation of mid-surface meshes. The results demonstrated good performance, as shown in the example in Fig. 28. However, in certain special cases, this method may still fail. The cause of these failures is similar to the issue of self-intersections in offset surface. As shown in Fig. 32, when a significant surface offset is applied at regions of high curvature, self-intersections can occur. In the model shown in Fig. 33, even with the most advanced commercial software, correct mid-surface mesh generation cannot be achieved. When our method is applied, issues such as mesh flipping or mesh intersections may still arise. Nevertheless, considerable research has been devoted to addressing such challenges [35–37]. Building on this, our method can also be extended to handle the mid-surface mesh generation problem at regions of high curvature.

In CAE simulation analysis, dimensionality reduction is a commonly used method to simplify complex problems and improve computational efficiency, but its limitations should not be overlooked. First, dimensionality reduction can result in the loss of important information from the original data, potentially overlooking key features and impacting the accuracy of the analysis. Additionally, the reduced dimensions often lack clear physical meaning, making the results more difficult to interpret. There is also a risk of oversimplifying the system, which may lead to missing critical dynamics or nonlinear behaviors, particularly in complex engineering applications. Moreover, many dimensionality reduction techniques assume linear relationships between data points, whereas nonlinear relationships are often more prevalent in real-world problems, leading to less accurate outcomes. Furthermore, the results of dimensionality reduction are highly dependent on the method chosen, and different techniques can produce varying results, adding complexity to the analysis. Lastly, once data has been reduced, it can be difficult or impossible to fully recover the original information, limiting further

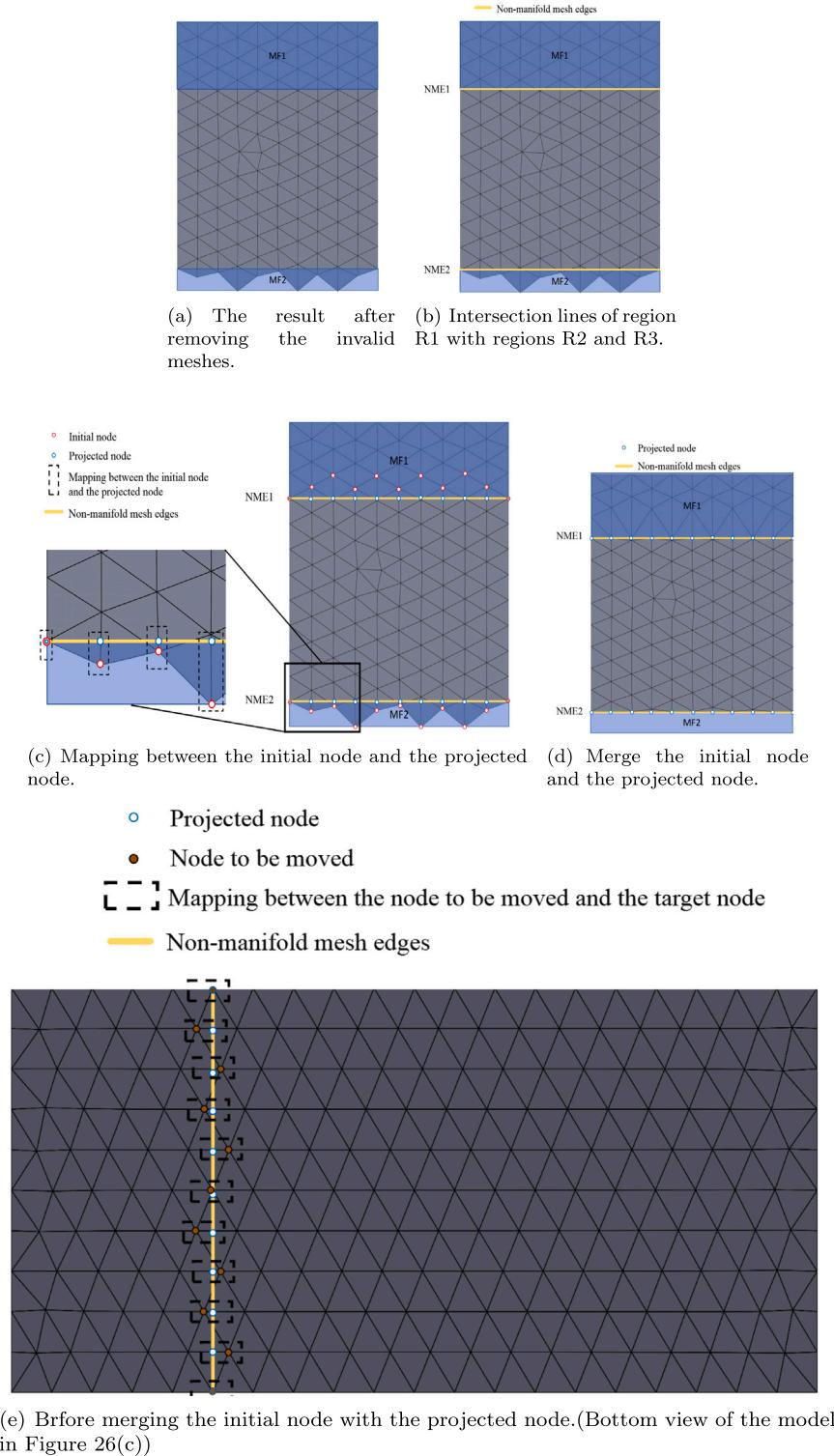
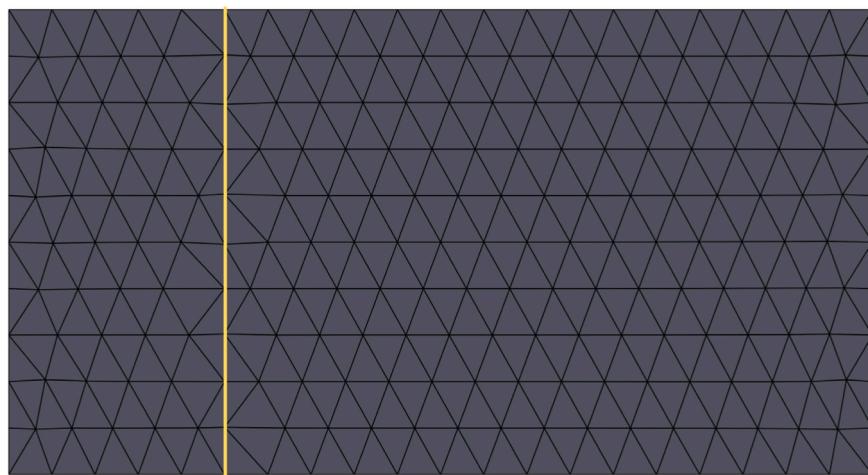
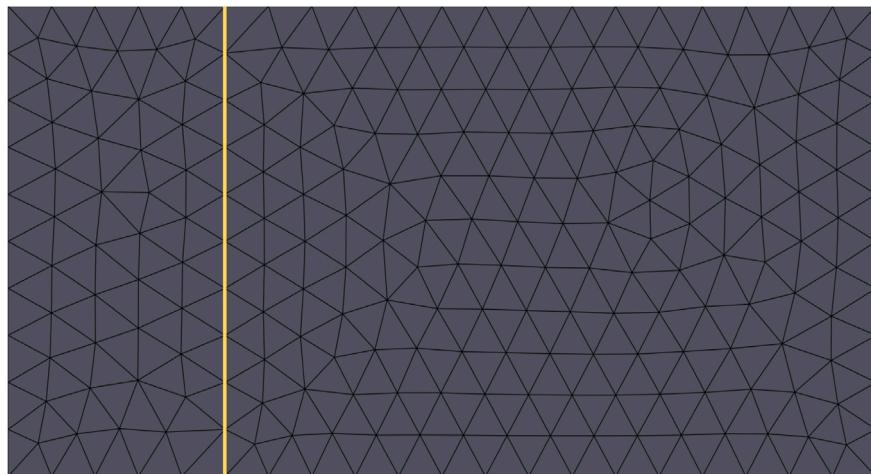


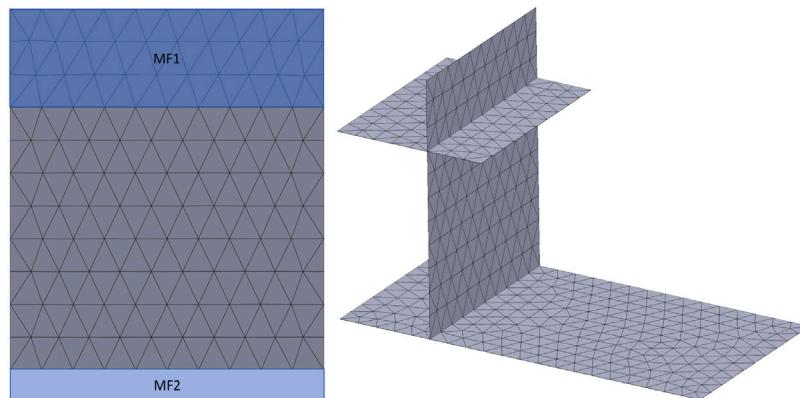
Fig. 27. Mesh editing and optimization.



(f) After merging the initial node with the projected node.(Bottom view of the model in Figure 26(c))



(g) Remesh regions R1 and R2.



(h) The result mesh.

Fig. 27. (continued).

detailed analysis. Therefore, when applying dimensionality reduction methods, it is important to carefully consider their impact on the accuracy and reliability of CAE analysis results.

6. Conclusion

This paper proposes a method for generating mid-surface meshes in thin-walled models based on virtual topology. For complex solid

thin-walled models, the model is decomposed into simple volumes, and virtual topology operations and single-layer triangular mesh divisions are performed on these simple volumes to generate mid-surface meshes. This method avoids the generation of continuous geometric mid-surfaces and directly generates the discrete mesh mid-surfaces needed for CAE analysis. Compared to previous similar methods, this

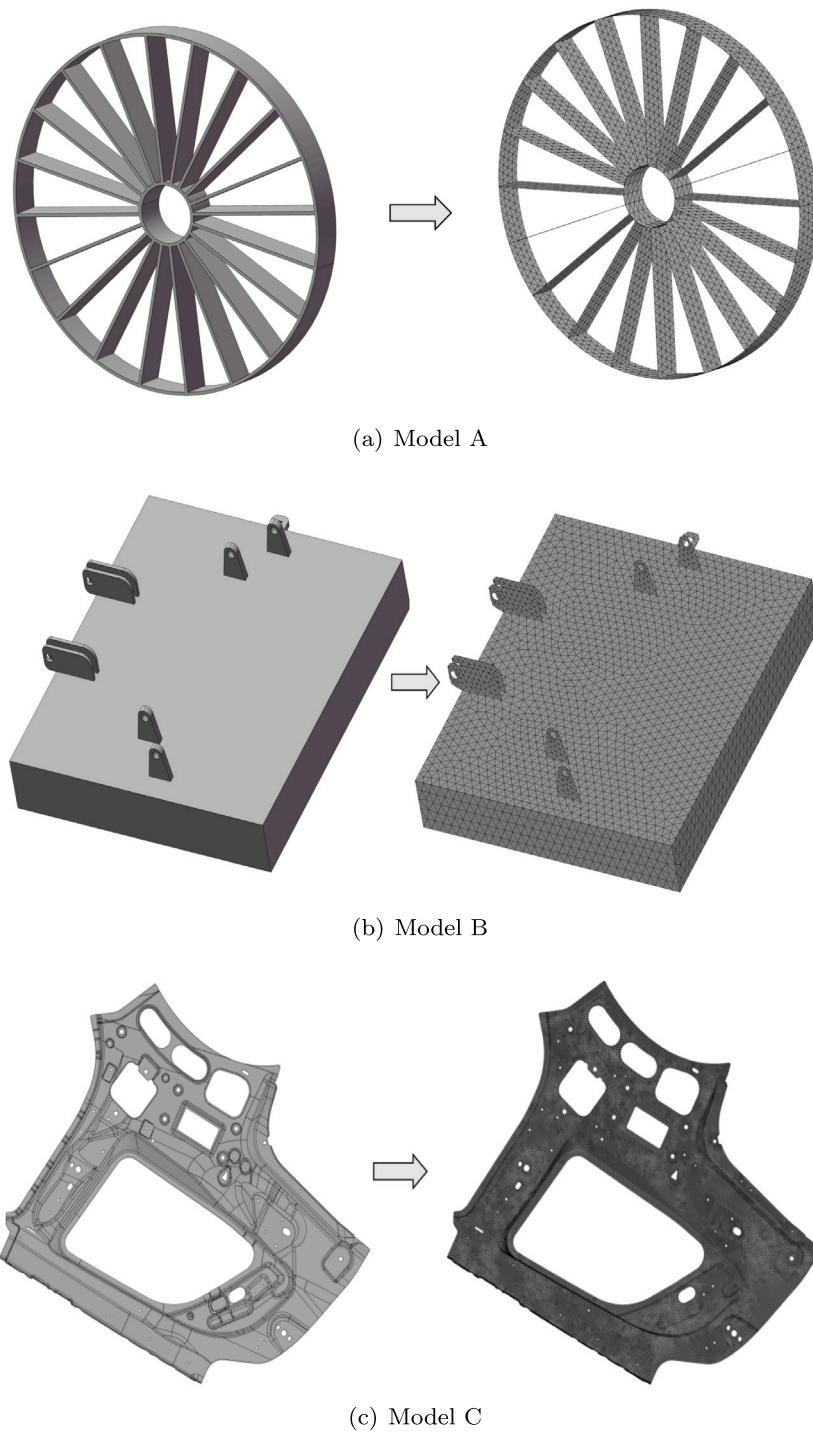
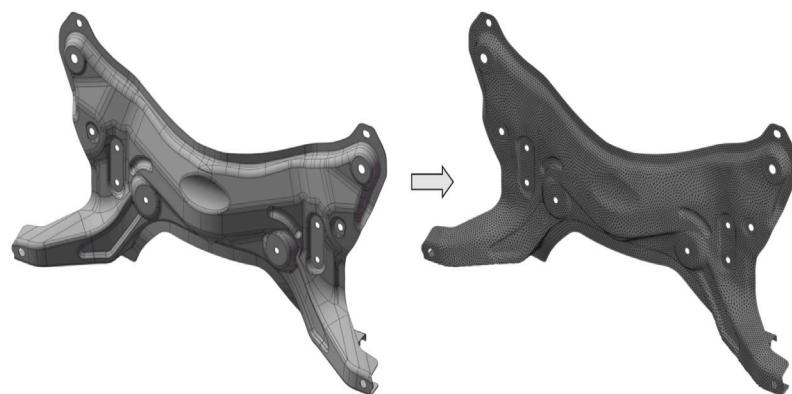


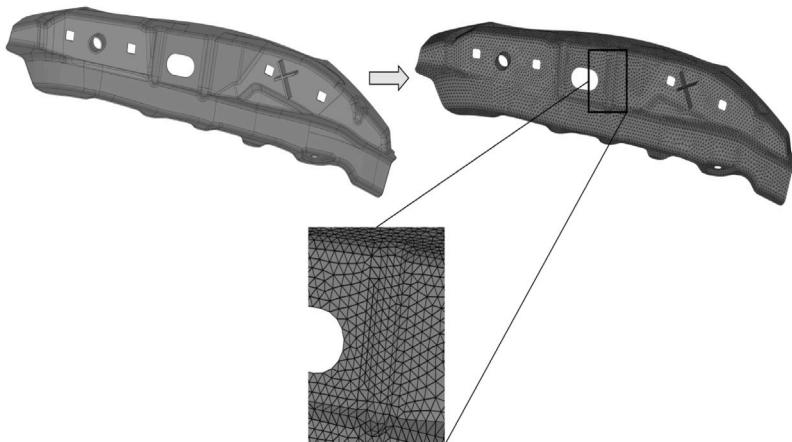
Fig. 28. Examples of mid-surface meshes abstracted by the method presented in this paper.

method allows users to choose triangular mid-surface meshes, quadrilateral mid-surface meshes, or hybrid meshes according to their needs. Furthermore, this method is more robust as it does not involve complex

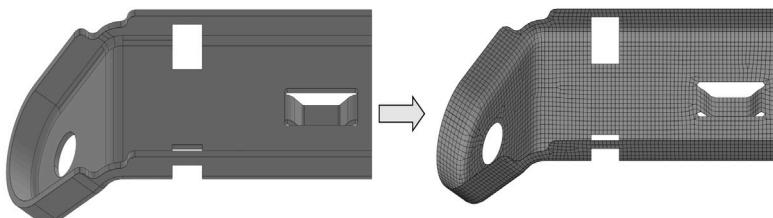
element cutting operations. This paper discusses some technical issues in previous methods and demonstrates the effectiveness of this method with research results. Overall, the method presented in this paper shows



(d) Model D



(e) Model E



(f) Model F

Fig. 28. (continued).

its potential in direct mid-surface mesh generation. It is acknowledged that in some special cases, manual intervention is required for mid-surface abstraction. As a new method, it may not be fully universal to adapt to every real industrial model, and some limitations may yet be discovered. The method proposed in this study is not the final result, and we will continue to enhance its performance in future research.

CRediT authorship contribution statement

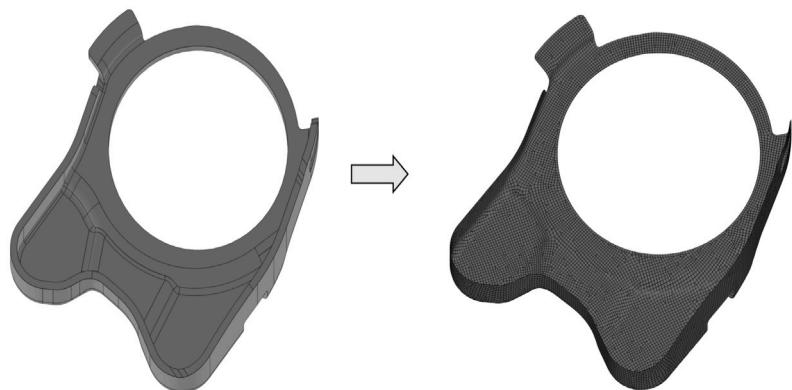
Feiqi Wang: Writing – original draft, Software, Formal analysis, Data curation, Conceptualization. **Haidong Wang:** Writing – original draft, Formal analysis. **Xiang Zhao:** Writing – review & editing. **Qi Ran:** Writing – review & editing. **Guidong Wang:** Writing – review & editing. **Huan Zhang:** Writing – review & editing. **Xin Hu:** Funding acquisition. **She Li:** Funding acquisition. **Xiangyang Cui:** Writing – review & editing.

Declaration of competing interest

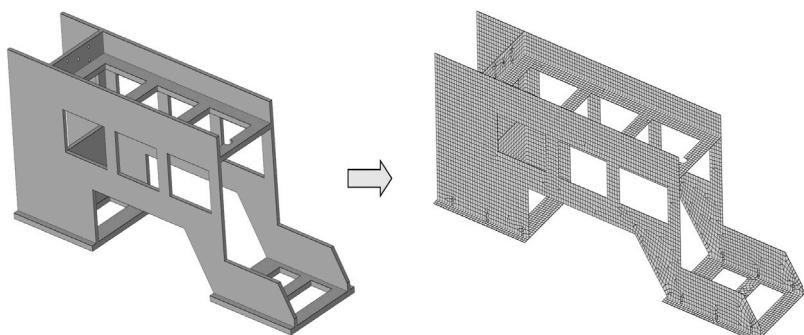
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Key R&D Program of China (2022YFB2503505), Natural Science Foundation of Hunan Province (2022JJ40031), National Natural Science Foundation of China (12302259) and China Postdoctoral Science Foundation (2022M721099).

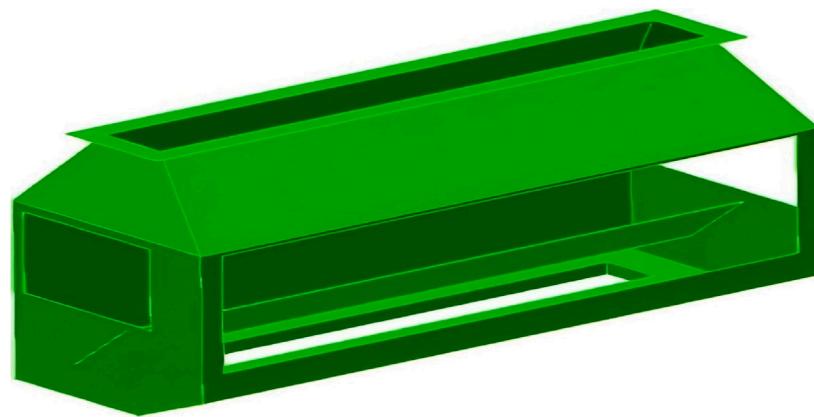


(g) Model G

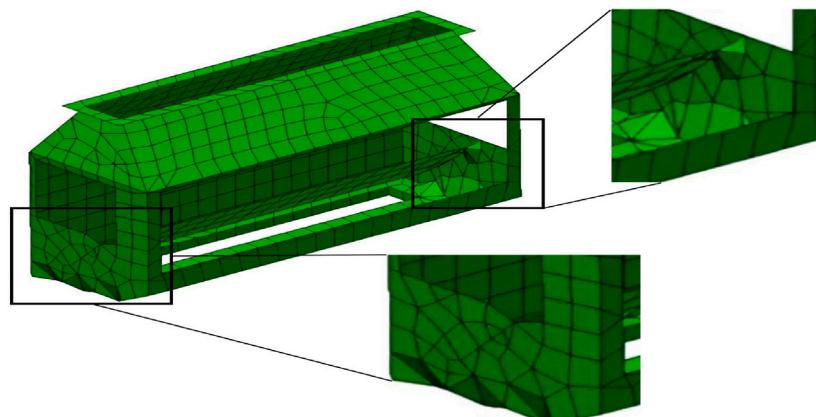


(h) Model H

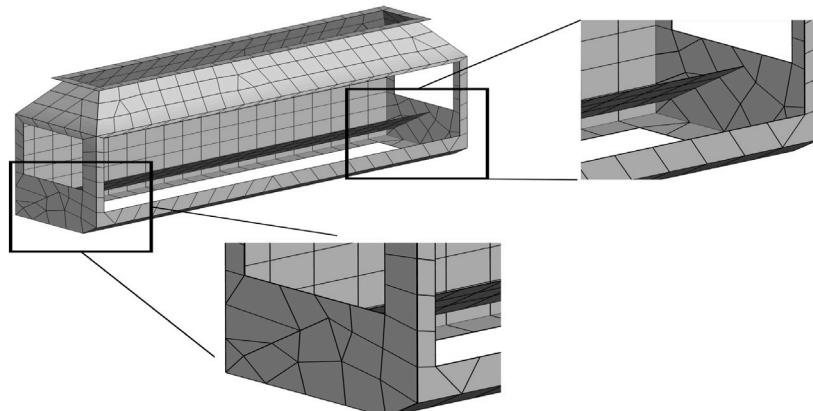
Fig. 28. (continued).



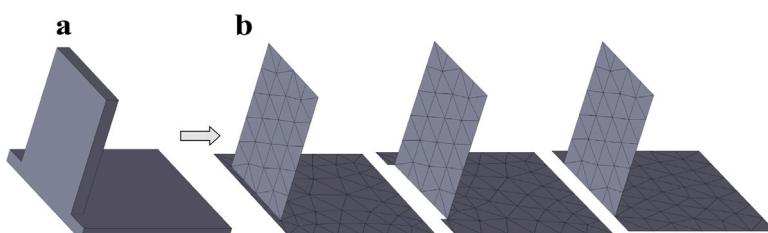
(a) Thin-walled solid model



(b) The mid-surface mesh result generated by Quadros's method



(c) The mid-surface mesh result generated by our method

Fig. 29. Comparison of mid-surface mesh results between CAT and our method.**Fig. 30.** Which mid-surface mesh is a better abstraction for the solid model?: (a) solid model, and (b) multiple possibilities of mid-surface mesh abstraction.

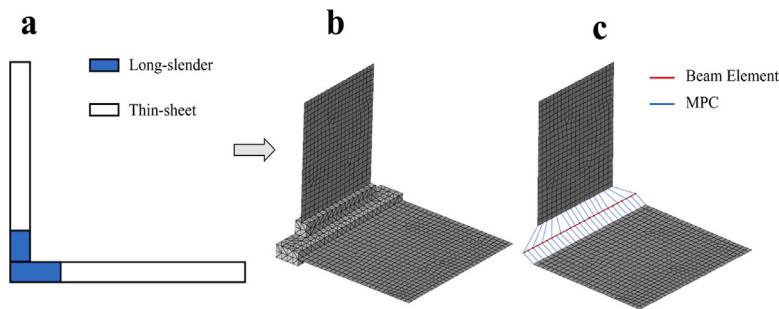


Fig. 31. Mixed dimensional analyses.

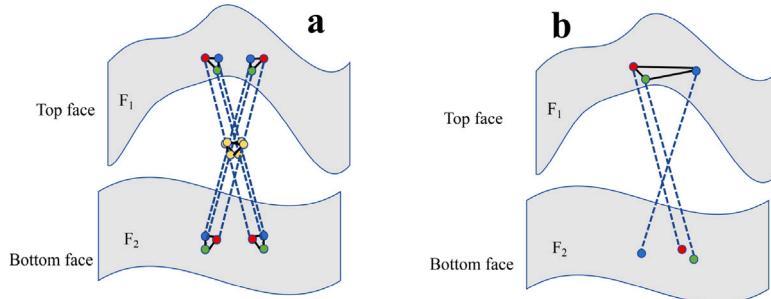


Fig. 32. Caused by excessively high local curvature of the surface: (a) mesh crossing (b) mesh inversion.

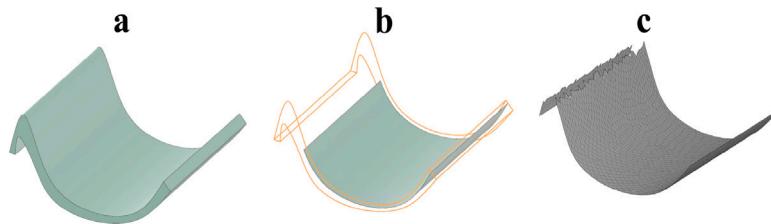


Fig. 33. (a) Thin-walled model with high curvature surfaces (b) mid-surface result of advanced commercial software (c) mid-surface mesh results of our method.

Data availability

Data will be made available on request.

References

- [1] Blum Harry. A transformation for extracting new descriptions of shape. *Model Percept Speech Vis Form* 1967;362–80.
- [2] Lee Yong-Gu, Lee Kunwoo. Computing the medial surface of a 3-D boundary representation model. *Adv Eng Softw* 1997;28(9):593–605. [http://dx.doi.org/10.1016/S0965-9978\(97\)00024-0](http://dx.doi.org/10.1016/S0965-9978(97)00024-0).
- [3] Ramanathan M, Gurumoorthy B. Generating the mid-surface of a Solid using 2D MAT of its faces. *Comput-Aided Des Appl* 2004;1(1–4):665–74. <http://dx.doi.org/10.1080/16864360.2004.10738312>.
- [4] Quadros WR, Shimada Kenji. Hex-layer: Layered all-hex mesh generation on thin section solids via chordal surface transformation.
- [5] Quadros William Roshan. An approach for extracting non-manifold mid-surfaces of thin-wall solids using chordal axis transform. *Eng Comput* 2008;24(3):305–19. <http://dx.doi.org/10.1007/s00366-008-0094-1>.
- [6] Rezayat Mohsen. Midsurface abstraction from 3D solid models: General theory and applications. *Comput-Aided Des* 1996;28(11):905–15. [http://dx.doi.org/10.1016/0010-4485\(96\)00018-8](http://dx.doi.org/10.1016/0010-4485(96)00018-8).
- [7] Lee Hanmin, Nam Yong-Youn, Park Seong-Whan. Graph-based midsurface extraction for finite element analysis. In: 2007 11th International Conference on Computer Supported Cooperative Work in Design. 2007, p. 1055–8. <http://dx.doi.org/10.1109/CSCWD.2007.4281585>.
- [8] Huawei Zhu Yusheng Liu, Li Chunguang. Mid-surface abstraction for complex thin-wall models based on virtual decomposition. *Internat J Comput Integr Manufact* 2016;29(8):821–38. <http://dx.doi.org/10.1080/0951192X.2015.1068455>.
- [9] Chong CS, Senthil Kumar A, Lee KH. Automatic solid decomposition and reduction for non-manifold geometric model generation. *Comput-Aided Des* 2004;36(13):1357–69. <http://dx.doi.org/10.1016/j.cad.2004.02.005>.
- [10] Zhu Huawei, Shao Yanli, Liu Yusheng, Zhao Jianjun. Automatic hierarchical mid-surface abstraction of thin-walled model based on rib decomposition. *Adv Eng Softw* 2016;97:60–71. <http://dx.doi.org/10.1016/j.advengsoft.2016.01.007>.
- [11] Woo Yoonhwan. Abstraction of mid-surfaces from solid models of thin-walled parts: A divide-and-conquer approach. *Comput-Aided Des* 2014;47:1–11. <http://dx.doi.org/10.1016/j.cad.2013.08.010>.
- [12] Robinson TT, Armstrong CG, Fairey R. Automated mixed dimensional modelling from 2D and 3D CAD models. *Finite Elem Anal Des* 2011;47(2):151–65. <http://dx.doi.org/10.1016/j.finel.2010.08.010>.
- [13] Nolan Declan C, Tierney Chris M, Armstrong Cecil G, Robinson Trevor T, Makem Jonathan E. Automatic dimensional reduction and meshing of stiffened thin-wall structures. *Eng Comput* 2014;30(4):689–701. <http://dx.doi.org/10.1007/s00366-013-0317-y>.
- [14] Sakurai Hiroshi. Volume decomposition and feature recognition: part I—polyhedral objects. *Comput-Aided Des* 1995;27(11):833–43. [http://dx.doi.org/10.1016/0010-4485\(95\)00007-0](http://dx.doi.org/10.1016/0010-4485(95)00007-0).
- [15] Sakurai Hiroshi, Dave Parag. Volume decomposition and feature recognition, part II: curved objects. *Comput-Aided Des* 1996;28(6):519–37. [http://dx.doi.org/10.1016/0010-4485\(95\)00067-4](http://dx.doi.org/10.1016/0010-4485(95)00067-4).
- [16] Woo Yoonhwan, Sakurai Hiroshi. Recognition of maximal features by volume decomposition. *Comput-Aided Des* 2002;34(3):195–207.
- [17] Woo Yoonhwan. Fast cell-based decomposition and applications to solid modeling. *Comput-Aided Des* 2003;35(11):969–77. [http://dx.doi.org/10.1016/S0010-4485\(02\)00144-6](http://dx.doi.org/10.1016/S0010-4485(02)00144-6).
- [18] <https://www.opencascade.com/>.
- [19] Foucault Gilles, Cuilliére Jean-Christophe, François Vincent, Léon Jean-Claude, Maranzana Roland. Adaptation of CAD model topology for finite element analysis. *Comput-Aided Des* 2008;40(2):176–96. <http://dx.doi.org/10.1016/j.cad.2007.10.009>.

- [20] Chen Jianjun, Cao Bingwan, Zheng Yao, Xie Lijun, Li Chenfeng, Xiao Zhoufang. Automatic surface repairing, defeaturing and meshing algorithms based on an extended B-rep. *Adv Eng Softw* 2015;86:55–69.
- [21] Tierney Christopher M, Sun Liang, Robinson Trevor T, Armstrong Cecil G. Using virtual topology operations to generate analysis topology. *Comput-Aided Des* 2017;85:154–67. <http://dx.doi.org/10.1016/j.cad.2016.07.015>, 24th International Meshing Roundtable Special Issue: Advances in Mesh Generation.
- [22] Sheffer Alla, Bercovier Michel, Blacker TED, Clements Jan. Virtual topology operators for meshing. *Internat J Comput Geom Appl* 2000;10(03):309–31.
- [23] <https://gmsh.info/>.
- [24] Floater Michael S. Mean value coordinates. *Comput Aided Geom Design* 2003;20(1):19–27.
- [25] Floater Michael. One-to-one piecewise linear mappings over triangulations. *Math Comp* 2003;72(242):685–96.
- [26] Floater Michael S, Hormann Kai. Surface parameterization: a tutorial and survey. *Adv Multiresolut Geom Model* 2005;157–86.
- [27] Eck Matthias, DeRose Tony, Duchamp Tom, Hoppe Hugues, Lounsbery Michael, Stuetzle Werner. Multiresolution analysis of arbitrary meshes. In: Proceedings of the 22nd annual conference on computer graphics and interactive techniques. 1995, p. 173–82.
- [28] Pinkall Ulrich, Polthier Konrad. Computing discrete minimal surfaces and their conjugates. *Exp Math* 1993;2(1):15–36.
- [29] Floater Michael S. Parametric tilings and scattered data approximation. *Int J Shape Model* 1998;4(03n04):165–82.
- [30] Remacle J-F, Geuzaine Christophe, Compere Gaëtan, Marchandise Emilie. High-quality surface remeshing using harmonic maps. *Internat J Numer Methods Engrg* 2010;83(4):403–25.
- [31] Floater Michael S. Parametrization and smooth approximation of surface triangulations. *Comput Aided Geom Design* 1997;14(3):231–50.
- [32] Hormann Kai, Greiner Günther, Campagna Swen. Hierarchical parametrization of triangulated surfaces. In: Proceedings of vision, modeling, and visualization. vol. 1999. 1999, p. 219–26.
- [33] Kucwaj Jan, Borowik Barbara. The efficiency of the 3-D delaunay triangulation combined with the advancing front method. In: 2008 international conference on signals and electronic systems. 2008, p. 479–82. <http://dx.doi.org/10.1109/ICSES.2008.4673472>.
- [34] Guo Yufei, Huang Xuhui, Ma Zhe, Hai Yongqing, Zhao Rongli, Sun Kewu. An improved advancing-front-delaunay method for triangular mesh generation. In: Advances in computer graphics: 38th computer graphics international conference, CGI 2021, virtual event, September 6–10, 2021, proceedings 38. Springer; 2021, p. 477–87.
- [35] Aomura S, Uehara T. Self-intersection of an offset surface. *Comput-Aided Des* 1990;22(7):417–21.
- [36] Pekerman Diana, Elber Gershon, Kim Myung-Soo. Self-intersection detection and elimination in freeform curves and surfaces. *Comput-Aided Des* 2008;40(2):150–9.
- [37] Hong Q Youn, Park Youngjin, Kim Myung-Soo, Elber Gershon. Trimming offset surface self-intersections around near-singular regions. *Comput Graph* 2019.