Dual Surface Based Approach to Block Decomposition of Solid Models

Zhihao Zheng, Rui Wang, Shuming Gao*, Yizhou Liao, Mao Ding

Abstract A high quality block structure of the solid model can support many important applications. However, automated generation of high quality block structure is still a challenging problem. In this paper, a dual surface based approach to automated and valid block decomposition of solid models is proposed. First, an optimized frame field is constructed on background tetrahedral mesh and three kinds of degenerated singularities are corrected. Then, dual loops for block decomposition are generated with the help of the optimized frame field. After that, a required dual surfaces set, whose dual surfaces can suitably separate all boundary elements of solid model and singularities of frame field, is constructed based on dual loops by min cut algorithm. Finally, a valid block structure is obtained by performing dual operations along the dual surfaces on the hex mesh generated by splitting the tetrahedral mesh of the solid model. Experimental results show the effectiveness of the proposed approach.

1 Introduction

A high quality block structure of the solid model can support many important applications. First, high quality hexahedral meshing, which is still a very challenging problem, can be easily achieved based on it. Second, a high quality block structure of a solid model plays an important role in the parameterization required for isogeometric analysis. Finally, this structure can be used to support multi-grid solvers to accelerate computations. It is due to such important applications that block decomposition of solid models has attracted more and more attention. However, automatic and high quality block decomposition for arbitrary shape is a challenging problem.

The dual operation based methods can always guarantee the topological validity of the hexahedral structure, that is, each cell of the decomposed structure is a hexahedron. Kowalski et al. [8] first obtained the initial hexahedral mesh of the model using a tet-to-hex method, then inserted the fundamental sheets into the mesh and finally extracted other sheets to obtain the final block structure. Although this method can generate a block structure for the model, it seems that the method hardly guarantees the geometric validity of block structure, that is, the final block structure cannot

Shuming Gao
State Key Laboratory of CAD&CG, Zhejiang University, Hangzhou, China e-mail: smgao@cad.zju.edu.cn

fully capture all the boundary elements of the solid model, because only the inserted fundamental sheets cannot sufficiently capture all the geometric information sometimes. To this end, Wang et al. [16] improved this method by adding curve-related sheets besides the fundamental sheets and obtaining the block structure by carefully extracting the sheets. The insertion of curve-related sheets ensures the geometric validity of the block structure while improving the quality of the final block structure. However, this method cannot deal with solid models with free-form surfaces. Gao et al. [5] obtained the base complex in the input hexahedral mesh, iteratively removed the appropriate sheets or chords, and performed geometric optimization after each deletion step, to finally obtain a conformal, non-inverted, coarse hexahedral mesh. This method can simplify the global structure of the complex hexahedral mesh robustly and efficiently, but it seems that the result is sensitive to the quality of the input hexahedral mesh.

Kowalski et al. [7] was the first to propose the frame field based method to decompose the body into blocks. The author set up the frame field by propagating the frames firstly defined along the geometric curves over the domain and smoothing the initial frame field. Then, the singular graph were extracted to obtain the block structure. However, it seems that this method does not guarantee the validity of the block structure for the models with degenerated frame fields.

Lei et al. [10] proposed a meshing algorithm based on the surface foliation theory for high-genus surfaces. This work proved the existence of a structured hexahedral mesh solution for high-genus models, but mainly considered the topological aspect. Other methods, such as skeleton driven [12] and generalized-sweeping-based [4] block decomposition, seem that are not suitable for complex model.

Campen et al. [2] proposed a method for surface quad layout construction based on dual loops. First, they made use of the method proposed by [1] to create a consistent smooth cross field. Second, they constructed admissible loops to suitably separate the singularities of the cross field. Finally, the quad layout was obtained by layout primalization.

In order to automatically and effectively generate valid block structures of solid models, in this paper we propose a dual surface based approach to block decomposition of solid models. In general, our method has the following contributions:

1. The method automatically guarantees the geometric and topological validity of the final block structure by constructing a required dual surfaces set and performing dual operations.
2. The method ensures the high quality of the every block generated by using the high quality dual surfaces whose construction is based on the dual loops and high quality frame field.

2 Preliminaries and approach overview

2.1 Preliminaries

Before describing our block decomposition method, we firstly introduce some related concepts.

2.1.1 Basic concepts related to dual space

Given a hex mesh H and a mesh edge e in H, the set of mesh edges E_s which consists of all mesh edges topologically parallel to e recursively found is called a **sheet**. E_{bs} is a set consisting of all the mesh boundary edges in E_s. If $E_{bs} \neq \emptyset$, the curve which traverses all the edges in E_{bs} is called a **dual loop**. The manifold surface which traverses all the mid points of the edges in E_s is called a **dual surface**. We call two opposite mesh edges in a mesh face topologically parallel edges. In Fig. 1a, the blue curve is a dual loop while the yellow surface is a dual surface.

The dual operations refer to the operations modifying the topology of the hex mesh. They mainly include the sheet inflation, sheet extraction and column collapse. For the details, the readers can refer to [9, 14].

2.1.2 Frame field

The frame field will be a promising tool to guide the block decomposition [7]. A 3D **frame F** is a 3-tuple $\{\mathbf{u}, \mathbf{v}, \mathbf{w}\}$, where **u**, **v**, and **w** are three unit vectors such that $\mathbf{u}.\mathbf{v} = 0, \mathbf{w} = \mathbf{u} \wedge \mathbf{v}$. In our algorithm, with a background tet mesh of the solid model, a **frame field** is a frame per cell of the tet mesh, as shown in Fig. 1c.

Between two frames F_s and F_t, there is a **matching matrix** $\Pi_{st} \in \mathscr{G}$ (the chiral cubic symmetry group). There is an orthonormal matrix $type(e,t_0)$ [6] representing the type of an oriented tetrahedral mesh edge e, where t_0 is a cell adjacent to the edge e. If $type(e,t_0) \neq I$, the e is a **singularity edge**.

A **simple singular polyline** is a set of concatenate singular mesh edges, which has the property that the number of singular edges adjacent to each end vertex is not 2 if it is not a closed polyline. To facilitate the construction of high quality dual surfaces, we divide all simple singular polylines into five catergories:

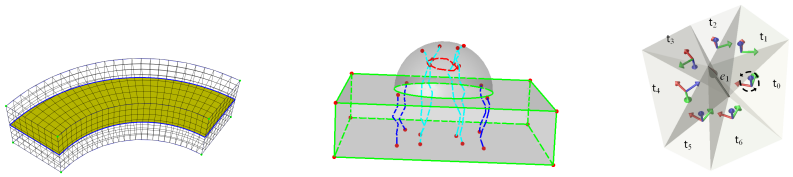
Type 1 All the singular edges of the simple singular polyline are inside the body, and only one of the polyline's endpoints is on the boundary surface, as shown by the cyan polylines in Fig. 1b;

Type 2 All the singular edges of the polyline are inside the body, while both of the two endpoints are on the boundary surface, as shown by the blue polylines in Fig. 1b;

Type 3 All the singular edges of the polyline are on the boundary surface, as shown by the green polylines in Fig. 1b;

Type 4 All of the singular edges of the polyline are inside the body, and both endpoints are not on the boundary surface. For a closed simple singular polyline, if all its singular edges are inside the volume, it is also defined as a type 4 polyline. Type 4 simple singular polylines are red polylines shown in the Fig. 1b;

Type 5 The simple singular polylines cannot be classified as one of the above four categories.



(a) (b) (c)

Fig. 1: (a)Dual loop and dual surface;(b)All simple singular polylines of test case 2 model: the cyan, blue, green and red lines are type 1,2,3 and 4 simple singular polylines, respectively;(c)A frame is associated with each cell and the edge e_1 is a singular edge with valence 3.

2.2 Overview of approach

Inspired by the work of Campen et al. [2], we solve the problem of block decomposition in the dual setting, namely, achieving the automatic and valid block decomposition of solid models based on dual surfaces and dual operations.

In order to effectively decompose a solid model into high quality blocks based on dual surfaces, three critical issues need to be addressed:

1. How to construct the dual loops required by the construction of dual surfaces toward valid block decomposition;
2. How to determine the required dual surfaces set based on dual loops;
3. How to generate the high quality dual surfaces that can ensure the high quality of every generated block.

For the first issue, the construction of dual loops is guided by the cross field for block decomposition to meet the needs. For the second issue, we determine the dual surfaces set by enabling all the dual surfaces in the set to separate each boundary elements of solid model and simple singular polylines so that the resulting block structure can be geometrically valid. If two elements are not separated by any dual surface, they will be merged into one in the final block structure. For the third issue,

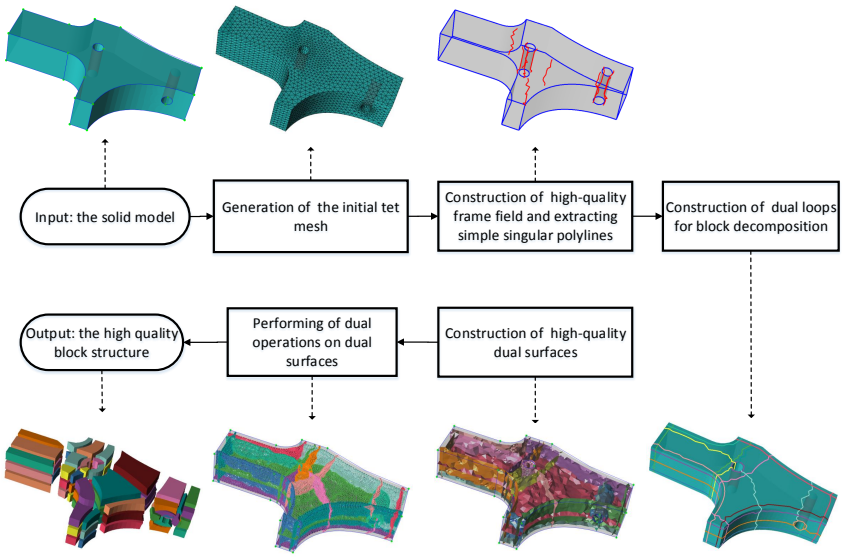


Fig. 2: Pipeline of dual surfaces based block decomposition.

we use the min cut algorithm to construct dual surfaces based on dual loops, so as to ensure the high quality of every dual surface, which leads to blocks of high quality.

The input of our approach is a solid model with boundary representation, and the output is a block structure. The following are the main steps of our method, as shown in Fig. 2:

1. A tet mesh is generated as background mesh;
2. A smooth frame field is built and three types of singularity degeneracy are corrected;
3. Dual loops for block decomposition are constructed;
4. Dual surfaces are constructed based on the dual loops and the singularity structure;
5. A block structure is generated based on the dual surfaces and the dual operations.

3 Frame field construction and singular polylines correction

To support the construction of high quality dual loops and dual surfaces, we first construct a high quality frame field of the solid model. First, we initialize the frame field by converting the cross field designed on the surface into frame field and propagating it inside the volume. Then we smooth the frame field by minimizing a non-convex object function using L-BFGS [11]. Due to limited space, we recommend

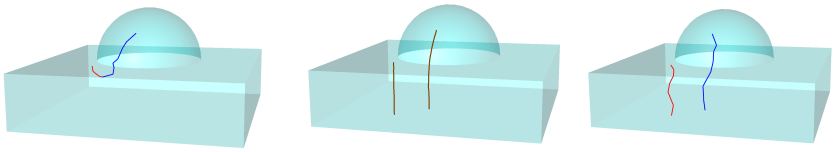
the interested readers to refer to [10]. After the frame field is smoothed, we extract simple singular polylines in the frame field and correct degenerate cases.

In order to ensure that the optimized frame field for block decomposition is as effective as possible, we automatically identify and correct possible degenerate singularities. Specifically, the following three types of degeneracy are identified and corrected: compound singularity, zigzag singularity and 3-5 simple singular polylines. For the first two types of degeneracy, we directly use the fixing strategy proposed in [6] to correct them.

3.1 Identification of 3-5 simple singular polylines

The **3-5 simple singular polyline** is the simple singular polyline whose two end edges are 3 and 5 valent singular edges respectively. As shown in Fig. 3a, the highlighted polyline is a 3-5 simple singular polyline with 3 valent end edge (in blue) and 5 valent end edge (in red). The cross field induced from the singular structure with 3-5 simple singular polylines cannot be used to construct all the dual loops for block decomposition (cf. Fig. 4a and Fig. 4c). Therefore, we have to identify and correct each 3-5 simple singular polyline.

According to the definition of 3-5 simple singular polyline, the key to identifying it is to determine the valence of the end edges in each simple singular polyline, and we determine the valence of an edge as follows: for each non-degenerate singular oriented mesh edge e, the **fixed axis** $\tilde{v}$ is one of member vectors of frame in tet t_0 such that $type(e,t_0)\tilde{v} = \tilde{v}$. If its deviation from the e is smaller(respectively, not smaller) than $\frac{\pi}{2}$ and $type(e,t_0)$ corresponds to a counterclockwise rotation by $\frac{k\pi}{2}$ along the $\tilde{v}$, then e is a $4-k$(respectively, $4+k$) valence singular edge, where $k \in \mathbb{Z}\backslash\{0\}$. In our cases, the valence of singular edges in the optimized frame field after correcting zig-zag and compound singularities is only 3 and 5, and there is theoretical possibility that higher valence of singular edges appears. According to the valence determination, the edge e_1 in Fig. 1c is a 3 valent singular edge.



(a) (b) (c)

Fig. 3: (a)A 3-5 simple singular polyline in test case 2;(b)Two related streamlines of the 3-5 simple singular polyline;(c)The 3-5 simple singular polyline is replaced with two polylines.

3.2 Correction of 3-5 simple singular polylines

According to the principle [15] that a simple singular polyline should be consistent with the direction of a frame field, we correct each 3-5 simple singular polyline based on the streamlines of the frame field as follows:

1. Tracing of two related streamlines. For each endpoint of the 3-5 simple singular polyline, one related streamline who starts from this endpoint and takes the direction of frame field closet to the directed end edge from the endpoint as the initial direction is traced to the boundary. The two related streamlines are shown in 3b.
2. Replacement of the 3-5 polyline. First, for each streamline, we find the corresponding polyline in mesh who are the shortest path connecting two endpoints of the streamline. Second, we replace the original 3-5 simple singular polyline with two polylines corresponding to two related streamlines. Two polylines are shown in 3c.

   Our strategy is heuristic and may not be able to deal with general case.

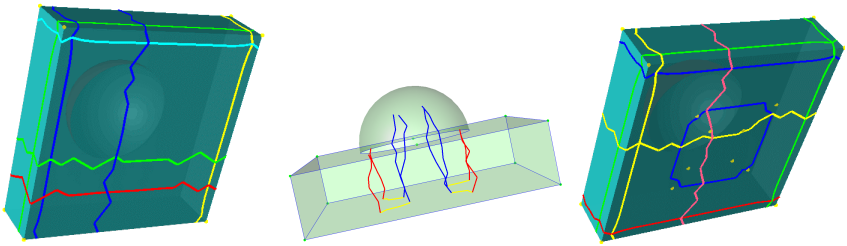
4 Construction of dual loops for block decomposition

Since it is very difficult to directly generate a high quality dual surface, we first constructs the boundary of dual surfaces, i.e., dual loops, to support the construction of the dual surfaces. In order to construct the dual loops required by the construction of dual surfaces, we need to improve the existing dual loop construction method [2]. The necessary condition for dual surfaces set to be able to separate all the simple singular polylines and all the boundary elements of solid model is that boundary loops of dual surfaces must be able to separate all the end points of singularity polylines and boundary elements of the surface. The basic idea of improvement is to construct extra dual loops based on surface cross field matching the frame field in volume. Specifically, the dual loops construction process is divided into two steps: first, the surface cross field is built based on the frame field in volume; second, dual loops are constructed to separate all the singular points and the boundary elements of the surface.

4.1 Construction of frame field based cross field

We first construct a surface cross field that matches the singularity structure in volume, and the specific process consists of the following two steps.

1. Initial construction of the surface cross field based on frame field in volume. The cross direction in each triangle is set as the non-normal direction of frame in



(a) (b) (c)

Fig. 4: (a)the dual loops are constructed based on the cross field without ajdustment;(b) the yellow polylines are the shortest paths connecting the pairs of the intersection points;(c)the dual loops are constructed based on the cross field after adjustment.

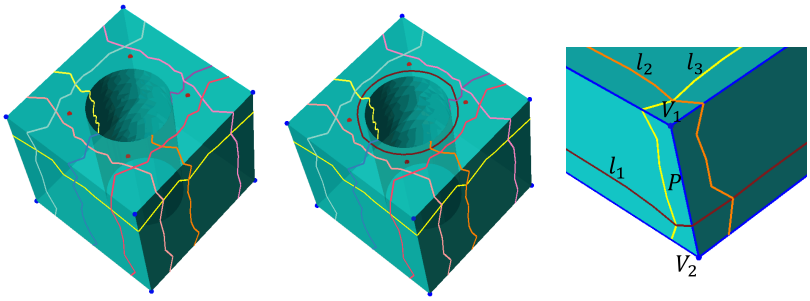
adjacent tetrahedral element; the period jump [13] between two adjacent crosses is determined by the matching matrix type between the corresponding frames.

2. Surface cross field adjustment. For each pair of simple polylines induced from a 3-5 simple singular polyline, the period jumps on shortest concatenate mesh edges connecting the new pair of intersection points between adjusted polylines and mesh boundary are adjusted so that a pair of 3-5 valence singular points meeting the requirements appears on the surface. The yellow polylines in Fig. 4b are the paths connecting the pairs of the intersection points while the loops in 4c are the dual loops constructed based on the cross field after adjustment.

4.2 Construction of dual loops for dual surfaces

We now construct the dual loops based on the cross field with the goal to separate all the target elements. In [2], only the singular points are set as the target elements to be separated. In our work, we set the singular points and the boundary elements of the surface (i.e. the geometric points and geometric edges of the solid model) as target elements. The specific steps are as follows:

1. Mesh preprocess before dual loops construction. The corresponding tetrahedron are refined so that there are at least two mesh edges connecting each pair of elements to be separated, so that the elements are separated by dual loops.
2. Construction of separation indicators (SIs) [2]. In order to separate pairs of target elements efficiently, we first construct SIs. For each pair of singularities, SIs are paths representing a homotopy class each, as [2] done. For each pair of boundary elements of the surface with other elements or singular points, SIs are shortest paths between them.
3. Construct the dual loops to cut all SIs by the greedy algorithm in [2].



(a) (b) (c)

Fig. 5: (a)The result dual loops are constructed based on SIs without boundary elements of the surface taken into consideration.(b)The result dual loops are constructed based on SIs with boundary elements of the surface taken into consideration.(c)The singular point V_1 and V_2 are all associated with the loop l_1, l_2 and l_3.

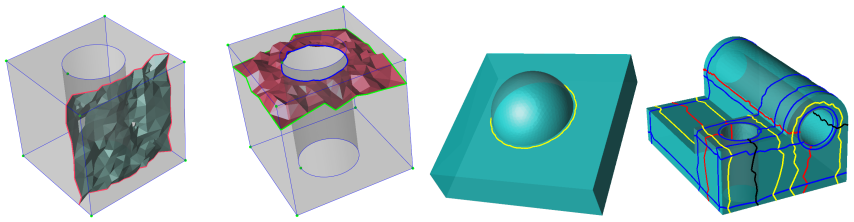
The constructed dual loops above divide the boundary surface into different regions, each containing at most one singular point. If a region contains a singular point, we associate all the dual loops that surround the region with the singular point. As shown in the Fig. 5c, the dual loops l_1, l_2 and l_3 are associated with the singular points V_1 and V_2 respectively. If all endpoints of a simple singular polyline on boundary surface are associated with a loop and located on same side of the loop, then this loop is associated with this polyline. As show in the Fig. 5c, the loops l_2 and l_3 are associated with the simple singular polyline P, respectively.

5 High quality dual surfaces construction

In this work, we decompose the solid model into a block structure based on dual surfaces. Therefore, constructing a set of high quality dual surfaces to separate all the simple singular polylines and boundary elements of the solid model is the key to accomplishing the valid block decomposition. In order to facilitate the construction of the dual surfaces, we classify all dual surfaces into two categories, closed and open. A dual surface without boundary is called a **closed dual surface** and a dual surface with boundary is called an **open dual surface**. According to the fact that the boundary of an open dual surface must be the dual loops on the boundary surface, we construct each open dual surface by first identifying its corresponding dual loops. As for the closed dual surface, it will be constructed directly according to its separation effect.

5.1 Determination and construction of open dual surfaces

In the following, we refer to the open dual surface with only one boundary loop as **simple dual surface**, as shown in Fig. 6a, and refer to the open dual surface with multiple boundary loops as **complex dual surface**, as shown in Fig. 6b. In our dual surface construction algorithm, the determination of complex dual surface requires the aid of some simple dual surfaces. Therefore, we first construct the simple dual surfaces, and then construct the complex dual surfaces.



(a) (b) (c) (d)

Fig. 6: (a)A simple dual surface.(b)A complex dual surface.(c)The yellow loop is a trivial loop but the loop is an M-loop.(d)The yellow loops are trivial loops and also S-loops. The black loops are loops in the set **B**. The red loops are non-trivial but S-loops. The blue loops are M-loops.

5.1.1 Classification of dual loops

In order to facilitate the construction of open dual surfaces, the dual loops are classified into two types:

S-loop An S-loop (short for the boundary loop to construct dual surface) is a loop that can span a required surface in the interior of the solid model by itself;

C-loop An C-loop (short for the multiple loops to construct) is a loop that can not span a required surface in the interior of the solid model by itself.

A required surface refers to the surface that can separate boundary elements of solid or/and simple singular polylines. Obviously, S-loops and C-loops are the boundary of simple and complex dual surfaces respectively.

There is a special kind of loop on surface called the **trivial loop** which alone can bound a subset of the surface; that is, a trivial loop can divide the boundary into two regions. According to our analysis, the dual loops are classified based on the following heuristic rules:

1. If a trivial loop is associated with all the endpoints of some simple singular polylines, then it is an S-loop, otherwise it is an C-loop. Though every trivial loop can

span a surface in the interior of the solid model, only the surface spanned by the trivial loop meeting the above condition can separate simple singular polylines with endpoints associated. Yellow loops in Fig. 6d are trivial loops which can span desired surfaces while the yellow loop in Fig. 6c is a trivial loop which can not span a desired surface.
2. If a nontrivial loop is the shortest loop passing through a hole, then it is an S-loop. Generally, such loop can also span a surface which can separate boundary elements or/and simple singular polylines. And we insert all such S-loops (as black loops shown in Fig. 6d) into a set **B**.
3. If a nontrivial loop which is not in **B** has no intersection with loops in **B**, then it is an S-loop (as red loops shown in Fig. 6d), otherwise it is an C-loop(as blue loops shown in Fig. 6d). Generally, such S-loop can span a surface which separates boundary elements and simple singular polylines.

5.1.2 Construction of simple dual surfaces

After identifying all the S-loops, we can construct simple dual surfaces accordingly. To obtain the block structure by dual operations, the dual surfaces need to be constructed in a hexahedral mesh. So a hex mesh is transformed from the tet mesh by using the tet-to-hex method, i.e., splitting each tetrahedron into four hexahedra. At the same time, the frame field of hex mesh inherits the frame field of tet mesh trivially.

The simple singular polylines are elements to be separated by dual surfaces, so during the dual surfaces construction the simple singular polylines are taken into consideration if the polylines are associated with the input loops. Due to the fact that the determination of the dual surfaces on the mesh is analogous to finding a cut on a graph [3], we use the min cut algorithm for input dual loop(s) to construct dual surface(s) by the following steps:

1. Determination of the source and the target hexahedra sets for min cut algorithm.
   a. Quads division on the boundary surface. The input dual loops divides the quads on the boundary surface into two sets, which are called source and target quads sets respectively.
   b. Source and target hexahedra sets determination. The hexahedra with boundary mesh face in the source(target) quads set are inserted into the source(target) hexahedra set.
   c. Source and target hexahedra sets completion. In order to take the associated simple singular polylines into consideration, the source and target hexahedra sets need to be completed. The hexahedra adjacent to the associated simple singular polylines are inserted into the source (target) hexahedron set if its boudanry endpoints belong to the source (target) quadrilateral set. The source and target hexahedra set are shown in Fig. 7b and 7c, respectively.
2. Construction of desired digraph and acquisition of dual surfaces by min cut algorithm.

a. Nodes determination. A graph is built by considering all the hexahedra as the nodes, where the nodes corresponding to source and target hexahedra are denoted as s-nodes and t-nodes, respectively.
b. Arcs determination and weight assignment. Two nodes are considered adjacent if their corresponding cells share at least one mesh face. For the s-node, a directed arc from it to each of its adjacent nodes is added; for the t-node, a directed arc from each of its adjacent nodes to it is added; for the rest pair of adjacent nodes, a pair of opposite directed arcs are added. The weight of each arcs above is set as the area of the common mesh face between corresponding cells.
c. Dual surface determination. The minimum cut is found between s- and t-nodes using the min cut algorithm, and this cut corresponds to the simple dual surface in the hex mesh.

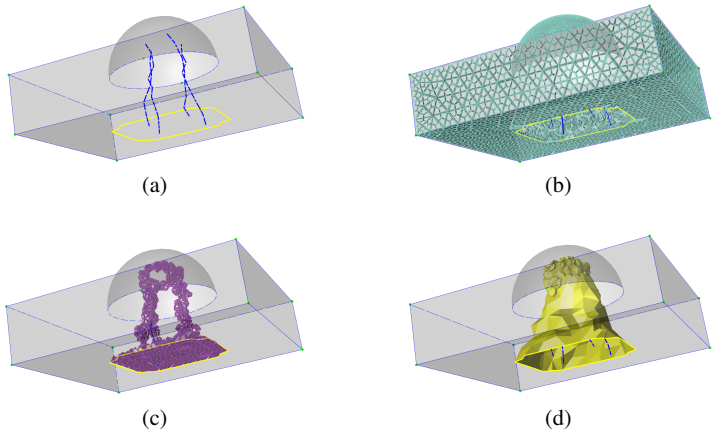


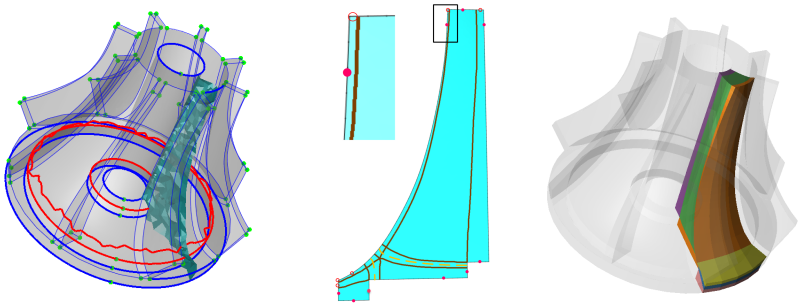
Fig. 7: Illustration of simple dual surface construction:(a)the yellow loop is associated with four type 1 simple singular polylines;(b)the hexahedra in source set;(c) the hexahedra in target set;(d)the yellow surface is constructed by min cut algorithm.

5.1.3 Determination and construction of complex dual surfaces

After all the simple dual surfaces constructed, the complex dual surfaces are constructed with the assist of existing simple dual surfaces. The intersection of two dual surfaces forms the dual curves and the intersections of the boundary loops of these dual surfaces are the endpoints of the corresponding dual curve. Therefore, we use the quad layout of the dual surfaces to determine the C-loops groups. For efficiency, only the dual surfaces spanned by the chosen dual loops and the shortest loops inter-

secting with the trivial undesired loops are considered. The following are the steps to determine the dual loops groups.

1. Extraction of the dual surfaces' quad layout based on cross field. First, the cross field on the smoothed dual surface is established by projecting the frames associated to cells on one side of surface. Second, the singular points on the surface are determined according to the intersection of the orignal dual surface and simple singular polylines, as shown in Fig. 8a. Finally, the streamlines are traced from singular points as the separatrices and quad layout is obtained, as shown in Fig. 8b.
2. Dual loops grouping. According to the quad layout, the intersecting loops at opposite side of each chord need to be grouped. Before doing this, some dual loops have to be added to balance the number of intersecting dual loops at both end of the chord. One dual loop per side of the chord is added for the chord who has no loop at neither side, except the trivial chord whose deletion will not influence the singular points configuration in quad layout, as golden dotted lines shown in Fig. 8b. If two dual loops groups intersect, they are merged into one. The Fig. 8c shows the final blocks traversed by the dual surface.



(a) (b) (c)

Fig. 8: Illustration of determining the dual loops group:(a) the target dual surface in test case 6;(b) the existing dual loops and newly added dual loops represented by solid red circles and hollow red circles in smoothed dual surface;(d)the final blocks traversed by the dual surface.

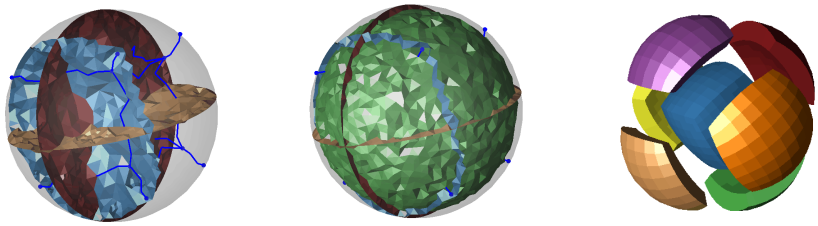
Since the uneven surface will result in undesired quad layout, we perform Laplacian smoothing to the dual surface before dual loops grouping. If the endpoints of a chord have unequal dual loops intersection numbers, we add dual loops as follows(the endpoint of the chord having larger intersection number is called the start side, and another endpoint is called the end side). First, shortest paths from each intersecting vertices on the start side to the end side are computed. Second, a dual loop for each shortest path is computed by anisotropic front propagation in [2] with the end vertex of the path chosen as the loop's start vertex. The first two steps are performed on the original tet mesh. Finally, the newly constructed dual loops are

converted into the hex mesh and replace with original dual loops on the end side. As for each chord who has no intersecting loop at neither sides, the two end vertices of the shortest path connecting the two side are chosen as start vertices to add dual loops.

After determining the dual loops groups, the complex dual surfaces are constructed by the min cut algorithm as simple dual surfaces done in Sect. 5.1.2.

5.2 Closed dual surface construction

Since construction of open dual surfaces only takes boundary dual loops and simple singular polylines into consideration, the type 4 simple singular polylines may not be sufficiently separated. Hence, closed dual surfaces must be constructed to make up for this omissions. Currently, we only consider the general situation that the type 4 polylines are not separated from the boundary surfaces. Therefore, the method first detects whether there is any type 4 polyline unseparated from the boundary surface of model and then constructs corresponding dual surface to separate the polyline and the surface if necessary. The specific algorithm is as follows:



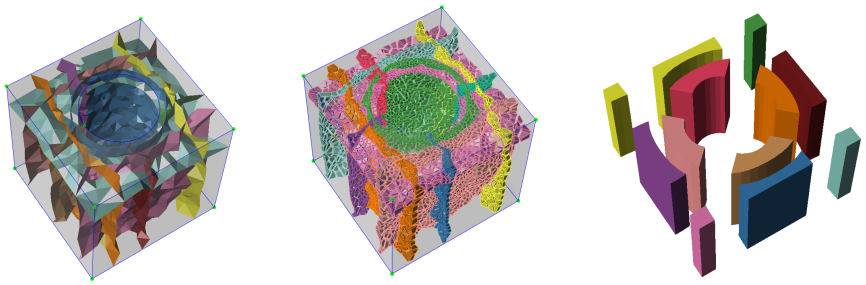
(a) (b) (c)

Fig. 9: Illustration of the closed dual surface construction:(a)12 simple singular polylines with type 4 not separated from boundary surface;(b)a closed dual surface constructed;(c)the final blocks generated.

1. Check whether a closed dual surface is needed. First, decompose the hexahedral mesh by the existing dual surfaces. Then, traverse the parts and record the boundary surface if there is a part contains both hexahedra with mesh face on this surface and the singular edges of a type 4 simple singular polyline. There are 12 polylines unseparated from the boundary surface in Fig. 9a.
2. Construct the closed dual surface if necessary. First, the hexahedra whose boundary mesh faces are in the recorded surface of solid are inserted into the source hexahedra set, while the rest hexahedra are inserted into the target set. Then, the closed dual surface is generated by min cut algorithm as described in Sect. 5.1.2. A closed dual surface is inserted in Fig. 9b and the corresponding block structure of the model is shown in Fig. 9c.

6 Block structure generation through dual operation

After the required dual surfaces set has been constructed, as shown in the Fig. 10a, the corresponding sheets are generated through sheet inflation operations applied along the dual surfaces, as shown in Fig. 10b. Sheet set extraction [16] is used to remove the remaining sheets in the current hex mesh and final block structure is obtained, as shown in Fig. 10c.



(a) (b) (c)

Fig. 10: Illustration of the block structure generation:(a)the dual surface of the model;(b)sheets inflated based on the dual surfaces;(c)the final block structure.

7 Experimental results and limitations

The proposed method has been implemented using C++ as programming language and ACIS as the geometric engine. Tet meshes are generated using commercial software Abaqus. According to the max-flow min-cut theorem, we compute the min cut of digraph by the Ford-Fulkeson method. And all the shortest paths are computed by the Dijkstra algorithm. In order to visulize the block structure and measure the scaled Jacobian value, the final block structure generated through dual operations are refined and optimized. Apart from the test case 1 which has been shown in Fig. 2, the other test cases and results are shown in Table 1. And the statistics of the results are given in Table 2.

Comparison with Kowalski et al.'s method [7]. In column 2 of Table 1, the input model, singular graph, all the dual surfaces and final block structure of test case 2 are shown. The test case 2 is a model that can not be handled by the Kowalski's method. After correcting the 3-5 simple singular polylines, a frame field without degenerate singularities can ensure the validity of the final block structure by our method, and the obtained blocks is high quality as shown in Table 2.

Comparison with Wang et al.'s method [16]. The models of test cases 3 and 4 in Table 1 all contain free-form surfaces, which can not be handled by Wang's

Table 1: The main results of test cases.

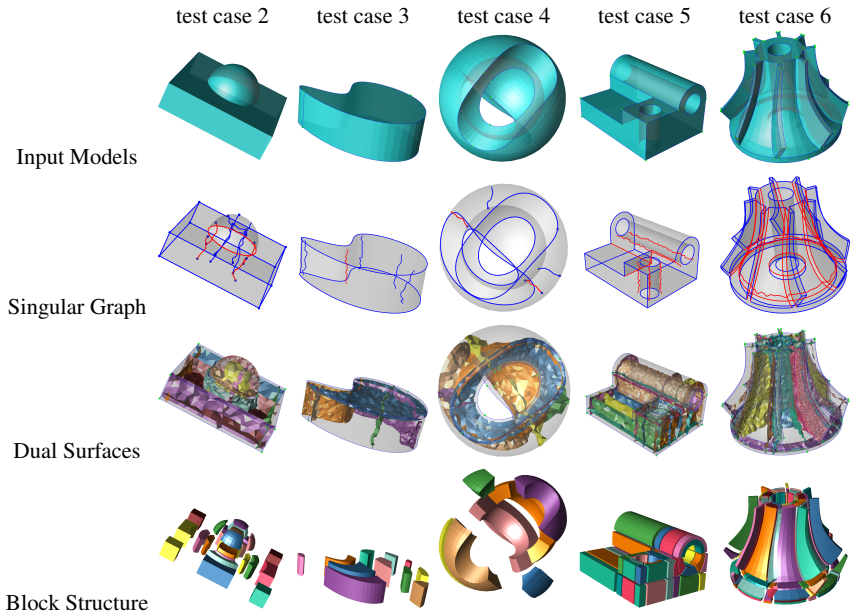


Table 2: The statistics of the results. The numbers of dual surfaces constructed, blocks of final structure, inner singular edges are in columns **Dual Surfaces, Block** and **Inner Singularity**, respectively. The fifth and sixth columns show the minimum and average scaled Jacobian values of blocks, and the last column shows the cell numbers of the refined hex mesh used.

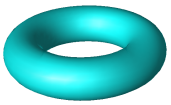
Model	Dual Surfaces	Block	Inner Singularity	Scaled Jacobian		Hex Num.
				Min	Mean	
test case 1	21	75	20	0.434	0.969	8288
test case 2	10	23	20	0.723	0.971	2432
test case 3	6	11	6	0.411	0.908	5472
test case 4	6	7	4	0.784	0.982	4710
test case 5	17	67	16	0.681	0.988	14252
test case 6	30	128	16	0.325	0.972	8800

method. Guided by the optimized frame field, the dual surfaces set are constructed, which not only ensure the geometric validity but also ensure the quality of the blocks generated.

The models of test case 1 and 5 with two through holes are decomposed into high quality blocks. In column 6 of Table 1, the input model, singular graph, some dual surfaces constrcuted and final block structure of test case 6 are shown. The test case 6 is a relatively complex model and the final block structure of it has 128 blocks.

The refined mesh's minimum scaled Jacobian value is 0.325, as shown in Table 2. We think that the reason why these low quality elements is that our geometric optimization algorithm is not good enough for this example.

Limitations: There are several limitations existing in the current method. First, there is still the possibility that our frame field optimization and degeneracy correction may fail for complex model. Second, the dual loops constructed may be not enough for block decomposition. For example, there is no dual loop on torus(as shown in the right inset). Third, our dual loops categorization algorithm is currently heuristic based.



8 Conclusion and future work

In this paper, we propose a novel approach to block decomposition of solid models. The approach automatically generates a valid block structure of a solid model based on dual surfaces and dual operations. Compared to the previous methods, our approach has the following characteristics:

1. Through constructing the required dual surfaces set based on dual loops and simple singular polylines and enabling all the dual surfaces constructed to separate every simple singular polylines and boundary elements of the solid, the geometric validity of final block structure is guaranteed. In addition, through performing the dual operations on the intermediate hexahedral mesh, the topological validity of final block structure is guaranteed.
2. By constructing dual surfaces based on dual loops for block decomposition and min cut algorithm, the dual surfaces constructed are of high quality, which furthermore ensures the high quality of the final blocks.
3. By the identifying and correcting three types of singularity degeneracy of the frame field, including the 3-5 simple singular polyline introduced in this work, the dual loops generated based on such frame field can support the construction of the necessary dual surfaces as effectively as possible.

The following shortcomings need to be overcome in our future work:

1. Our current approach deals with the three types of degenerate singular polylines that can be observed up to now. However, it is also necessary to test more complex models and theoretically make sure whether there exists any other degeneracy of the singularities in the frame field and, if so, to provide corresponding correction strategy.
2. At present our dual loops classifying algorithm is heuristic based. In the future, a more general algorithm will be studied.
3. Since the construction of optimized frame field and dual loops are all sensitive to the input tet mesh, in the future, the reasonable tet mesh discretization of the input solid model will be studied.

Acknowledgements

We thank all anonymous reviewers for their valuable comments. This research is supported by the NSF of China (Nos. 61572432).

References

1. Bommes, D., Zimmer, H., Kobbelt, L.: Mixed-integer quadrangulation. In: ACM Transactions On Graphics (TOG), vol. 28, p. 77. ACM (2009)
2. Campen, M., Bommes, D., Kobbelt, L.: Dual loops meshing: quality quad layouts on manifolds. ACM Transactions on Graphics (TOG) **31**(4), 110 (2012)
3. Chen, J., Gao, S., Wang, R., Wu, H.: An approach to achieving optimized complex sheet inflation under constraints. Computers & Graphics **59**(C), 39–56 (2016)
4. Gao, X., Martin, T., Deng, S., Cohen, E., Deng, Z., Chen, G.: Structured volume decomposition via generalized sweeping. IEEE transactions on visualization and computer graphics **22**(7), 1899–1911 (2016)
5. Gao, X., Panozzo, D., Wang, W., Deng, Z., Chen, G.: Robust structure simplification for hex re-meshing. ACM Transactions on Graphics (TOG) **36**(6), 185 (2017)
6. Jiang, T., Huang, J., Wang, Y., Tong, Y., Bao, H.: Frame field singularity correctionfor automatic hexahedralization. IEEE Transactions on Visualization and Computer Graphics **20**(8), 1189–1199 (2014)
7. Kowalski, N., Ledoux, F., Frey, P.: Smoothness driven frame field generation for hexahedral meshing. Computer-Aided Design **72**, 65–77 (2016)
8. Kowalski, N., Ledoux, F., Staten, M.L., Owen, S.J.: Fun sheet matching: towards automatic block decomposition for hexahedral meshes. Engineering with Computers **28**(3), 241–253 (2012)
9. Ledoux, F., Shepherd, J.: Topological and geometrical properties of hexahedral meshes. Engineering with Computers **26**(4), 419–432 (2010)
10. Li, Y., Liu, Y., Xu, W., Wang, W., Guo, B.: All-hex meshing using singularity-restricted field. ACM Transactions on Graphics (TOG) **31**(6), 177 (2012)
11. Liu, D.C., Nocedal, J.: On the limited memory bfgs method for large scale optimization. Mathematical programming **45**(1-3), 503–528 (1989)
12. Livesu, M., Muntoni, A., Puppo, E., Scateni, R.: Skeleton-driven adaptive hexahedral meshing of tubular shapes. In: Computer Graphics Forum, vol. 35, pp. 237–246. Wiley Online Library (2016)
13. Ray, N., Vallet, B., Li, W.C., Lévy, B.: N-symmetry direction field design. ACM Trans. Graph. **27**(2), 10:1–10:13 (2008)
14. Staten, M.L., Shepherd, J.F., Ledoux, F., Shimada, K.: Hexahedral mesh matching: Converting nonconforming hexahedraltohexahedral interfaces into conforming interfaces. International Journal for Numerical Methods in Engineering **82**(12), 1475–1509 (2009)
15. Viertel, R., Staten, M.L., Ledoux, F.: Analysis of non-meshable automatically generated frame fields. In: the 25th International Meshing Roundtable (2016)
16. Wang, R., Shen, C., Chen, J., Wu, H., Gao, S.: Sheet operation based block decomposition of solid models for hex meshing. Computer-Aided Design **85**, 123–137 (2017)