Volumetric Decomposition via Medial Object and Pen-based User Interface for Hexahedral Mesh Generation

Jean Hsiang-Chun Lu[1], Inho Song[1], William Roshan Quadros[2], and Kenji Shimada[1]

[1] Carnegie Mellon University, Pittsburgh, PA, hsiangcl@andrew.cmu.edu, {songphd, shimada}@cmu.edu
[2] Sandia National Laboratories[†], Albuquerque, NM, wrquadr@sandia.gov

Abstract This paper describes an approach that combines the volumetric decomposition suggestions and a pen-based user interface (UI) to assist in the geometry decomposition process for hexahedral mesh generation. To generate the suggestions for decomposition, a 3D medial object (MO) is first used to recognize and group sweepable regions. Second, each sweepable region of the original model is visualized using different colors. Third, the ideal cutting regions to create cutting surfaces are highlighted. Based on the visual suggestions, users then create cutting surfaces with the pen-based UI. The models are then decomposed into sweepable sub-volumes following the MO based suggestions. The pen-based UI offers three types of tools to create cutting surfaces: (1) Freeform based tool, (2) B-REP based tool, and (3) MO based tool. The pen-based UI also selects a suitable type of tool automatically based on users input. The proposed approach has been tested on industrial CAD models and hex meshing results are presented.

Keywords: 3D medial object, pen-based user interface, hexahedral mesh generation, geometric decomposition, sweeping.

1 Introduction

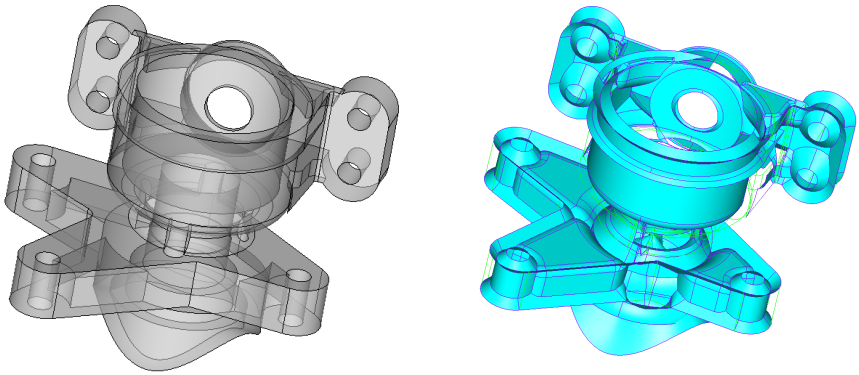
Since hex meshing is difficult for complex shapes, volumetric decomposition is routinely used for meshing industrial parts. Compared to other volumetric mesh types, a hexahedral element yields more accurate solution [1] and hence

[†]Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000

hex meshing is the only option in many simulations. Unfortunately, existing all hex meshing methods are not able to handle general shapes [2–7] or end up generating poor quality elements [8,9]. Performing volumentric decomposition reduces the complex shape into hex meshable sub-domains. High quality hex meshes can then be obtained by meshing simpler sub-domains.

Over the last decade, although volumetric decomposition has been extensively researched, fully automatic decomposition is still a challenging problem. With fully automatic approach, non-meshable sub-volumes still remain and require further manual decomposition. Also, fully automatic approach usually does not work on general solids. They work on only a special class of models such as many-to-many sweepable models. Unfortunately, manual decomposition capabilities in the current packages such as ABAQUS [10] and CUBIT [11] are highly labor-intensive and time consuming. Research has focused on providing better user interface (UI) to increase user productivity in the manual decomposition process [12, 13]; however, this work still requires user expertise and domain knowledge to determine sweepable regions and ideal cutting surfaces.

The medial object (MO) of a 3D shape is a skeleton representation that contains rich geometric information as shown in Fig. 1 and is of reduced dimension. The concept of medial object was originally proposed by Blum [14], which was called medial axis transformation (MAT). MAT combines medial axis and the radius of the inscribed circle, and has been applied to mesh generation [15].



(a) (b)

Fig. 1: (a) Original model. (b) Medial object.

In this paper, we propose an intelligent and user friendly pen-based UI that provides decomposition suggestions and automates the selection of different components to make decomposition semi-automatic. The main three steps of our approach are as follows: (1) the MO segmentation and grouping algorithm

detects the sweepable sub-regions and visualize them with different colors. (2) The user use the pen-based UI and three types of cutting surface creating tools to decompose the given model. (3) The model can be all hex meshed after conducting imprinting and merging.

One of the primary contributions of this paper is to automatically detecting sweepable sub-regions via MO. Our approach uses the 3D MO to detect sweepable sub-regions and ideal cutting regions. The sweepable sub-regions are visualized on the original geometry with different colors, and the ideal cutting regions are highlighted using a virtual volume. These MO-based decomposition suggestions are combined with the pen-based UI to make manual volumetric decomposition more precise and user friendly. Our MO-based algorithm to detect sweepable sub-regions has been tested on industrial models.

The paper is organized as follows: Section 2 discusses the related work. Section 3 covers the basic characteristics of MO and MO terminologies. Section 4 gives an overview of the automatic detection of sweepable region via MO. Section 5 presents the details of the algorithm to detect sweepable regions. Section 6 describes the three types of cutting-surface creation tools in our pen-based UI. Section 7 demonstrates the decomposition and meshing results on industrial models.

2 Related Work

2.1 Automatic Decomposition

Lu et al. [16] described an automatic approach that uses local geometric information and feature recognition algorithms to subdivide meshable regions. Their approach left the remaining pieces for further manual operations. White et al. [17] automatically decomposed multi-sweep volumes into many-to-one sweepable volumes. Shin et al. [18, 19] describe swept volume decomposition method that works for geometries that can be decomposed into linearly swept volumes. These automatic decomposition methods only work for specific types of geometries. The decomposition of general solid is not always possible, and there are usually unmeshable sub-volumes remaining.

Medial axis transformation have been used to guide the decomposition of volumes into meshable primitives [8,9,20]. Li et al. [21] use the midpoint subdivision and integer programming to create simplified meshable sub-regions. Though these methods work for general solids, poor quality elements are generated.

2.2 Manual Decomposition

CUBIT offers an Immersive Topology Environment for Meshing (ITEM) [12]. ITEM determines possible cutting positions and displays suggested cutting

surfaces for the user. However, the resulting cuts do not always produce meshable sub-volumes. It is because of the failure in recognizing complex features, and recognizing sweepable sub-regions. Lu et al. [13] proposed a pen-based UI that beautifies users freehand strokes to create precise cutting surfaces. This work requires user expertise and domain knowledge to determine the cutting position because it does not recognize sweepable regions.

2.3 MO for Mesh Generation

In the past, MO has been used in mesh generation as MO reduces the 3D meshing problem into 2D meshing on medial surface. Donaghy et al. [22] reduced model dimension for further analysis by using MO to classify several topology features such as end region of beams, concave and convex corners. Sampl [23] first meshed the medial faces, and then extruded the mesh on both side of the medial until the mesh intersects the boundary of the object. Chong et al. [24] used medial surface to reduce the complex original model to recognize features by identifying edge types,and by using classified medial information to treat different geometry combinations. In all these above approaches the goal is to generate a mesh than to decompose the domain into sweepable regions. The aim of this paper is to use MO to generate sweepable sub-domains so that high quality hex meshes can be generated using sweeping hex meshing algorithm.

3 The Medial Object

The Medial Axis Transform (MAT) – one of the most popular and mathematically well studied skeleton was initially defined by Blum [14] for biological shapes. In planar domain, MAT is defined as the locus of the centre of the maximal ball as it rolls inside an object, along with the associated radius function. The 3D equivalent is the locus of the center of maximal sphere and its associated radius (see Fig. 2(a)).

3.1 MO Terminology

- MO curve: a curve that connects two MO vertices.
- MO face: a surface bounded by MO curves.
- MO: a set of connected MO faces.
- MO patch: a set of 2-manifold MO faces connected by MO curves.
- MO group: a set of non-manifold MO patches that can be generated by extruding a set of connected 1D MO curves.
- MO segmentation: the process of generating MO patches from a MO.
- MO grouping: the process of generating MO groups from MO patches.

- Sweepable sub-volume: a sub-domain of the original volume that is sweepable, i.e., a 2.5D region of the original volume. A sweepable sub-volume consists of multiple source and target surfaces. Both a MO patch and a MO group will have a corresponding sweepable sub-volume.
- Valence: a medial curve is incident on one or more medial faces. The number of medial faces that are incident on a medial face determines the valence of a medial curve. If a medial curve is shared by two medial faces, the valence is two (CN-2); if it is shared by three medial surfaces, the valence is three (CN-3); and so on.
- Defining entity: the original model's geometric entities that the maximal sphere touches. These entities define the MO and hence are refered as defining entities As seen in Fig. 2(b), the defining entities of the medial face are two boundary surfaces on the model. The medial edge shown in Fig. 2(c) has three defining entities such as two surfaces and one edge on the original model.
- Trimmed MO: the MO without the medial faces/curves that touch the model boundary as seen in Fig. 3.

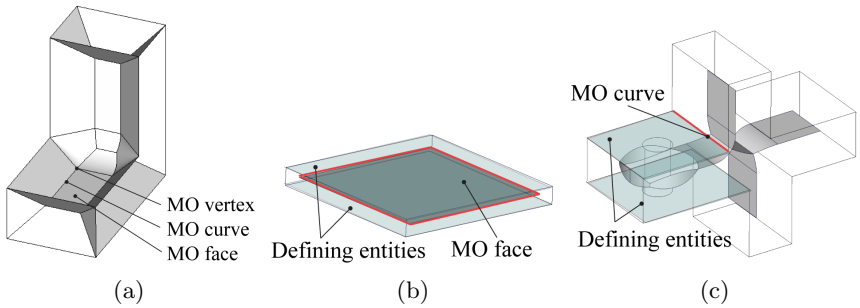
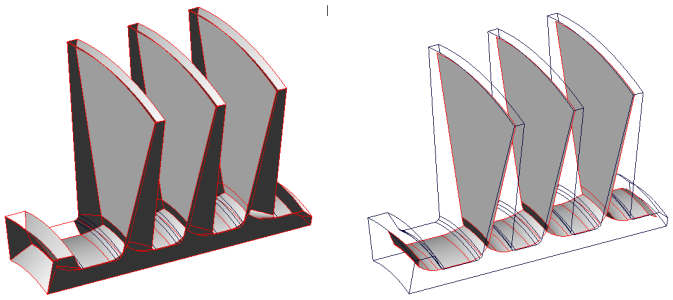


Fig. 2: (a) The MO face, MO curve and MO vertex of a L-shaped block. (b) The defining entities of a medial face. (c) The defining entities of a medial edge.

4 Automatic Detection of Sweepable Regions via MO

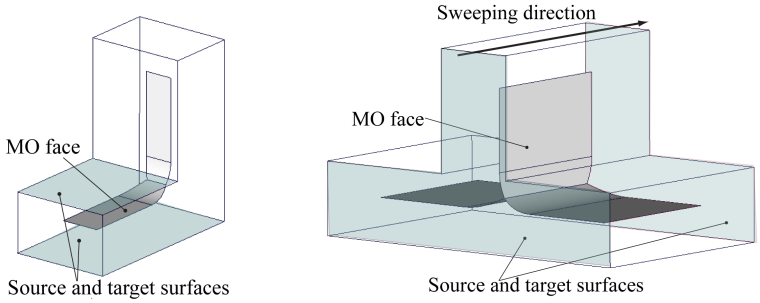
Instead of using traditional B-REP, we use MO skeleton representation in detecting sweepable regions. MO reduces the model dimension from 3D to 2D and makes the detection of sweepable regions easier. It is difficult to detect sweepable regions using B-REP based feature recognition techniques [25].

The two key concepts used in our approach are as follows: (1) A MO patch which is 2-manifold has a corresponding 3D sub-volume which is sweepable



(a) (b)

Fig. 3: (a) Original MO with the original models. (b) Trimmed MO.



(a) (b)

Fig. 4: (a) A MO face defined by its top and bottom surfaces, which are the sweep source and target surfaces of the region. (b) The volume can be made sweepable if two end surfaces are assigned as the source and target surfaces of a sweeping operation.

in 3D, and (2) a MO group which is non-manifold and is extrudable from 1D MO curves has a corresponding 3D sub-volume which is sweepable in 3D.

The following sections describe the detail of our algorithm that recognizes sweepable regions from MO. The process is shown in Fig. 5. Instead of using the original MO, the trimmed MO is used in our segmentation and grouping algorithm. Trimmed MO has also been used by Quadros et al. [26] as they are suitable for engineering purposes.

5 Details of the MO Segmentation and Grouping Algorithm

This section describes the three steps to generate the decomposition suggestions: (1) detecting sweepable regions via medial object (MO), (2) followed

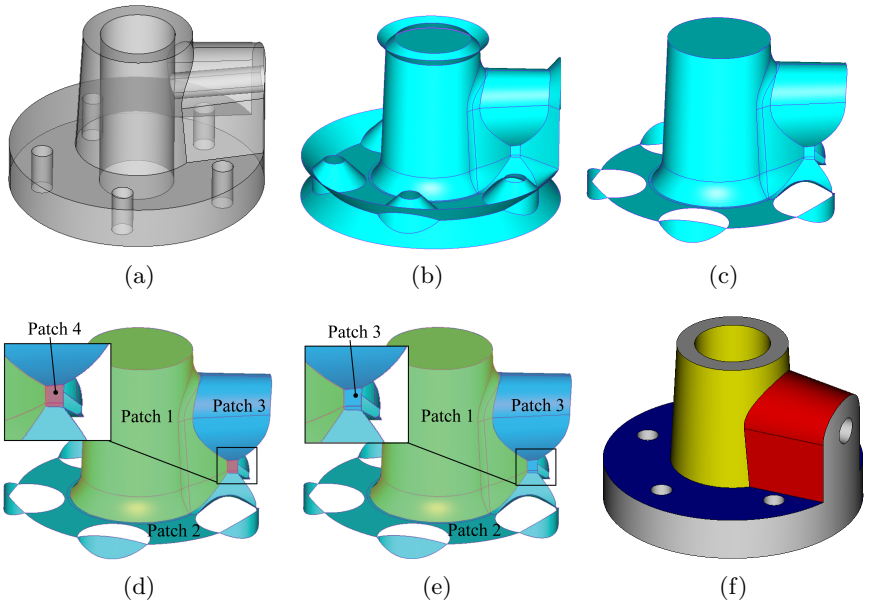


Fig. 5: (a) Original model (b) MO. (c) Trimmed MO. (d) Four 2-manifold patches. (e) Non-manifold patch obtaind by uniting patches 3 and 4. (f) Different colors are assigned to each sweepable sub-volume on the original model.

by visualizing different sweepable regions on the original model, and then (3) highlighting the ideal cutting regions if required.

5.1 Generating MO 2-manifold Patches via Segmentation

We first look at the definition of the medial face: a medial face is defined by the locus of the center of an inscribed sphere. In other words, two of the defining entities of a medial face must be the boundary surfaces of the original model. Therefore, a medial face represents a sub-volume that can be made sweepable if two defining surfaces are set as its source and target surfaces of a sweeping algorithm. As shown in Fig. 6(a), the source and target surfaces are labeled as “A” and “B”.

When another sweepable sub-volume (source and target surfaces are labeled as “C” and “D” in Fig. 6(b)) is united with the adjacent sub-volume, two medial faces are connected together through a CN-2 curve. This united volume is sweepable with source surfaces A, C and target surfaces B, D. Therefore, if medial faces are connected through a CN-2 curve, their defining entities should be grouped together to represent source and target surfaces of a united sweepable sub-volume. The MO patch made by many medial faces connected through CN-2 is always 2-manifold. The 2-manifold segmentation stops at a CN-3 curve to keep the corresponding sub-volume sweepable.

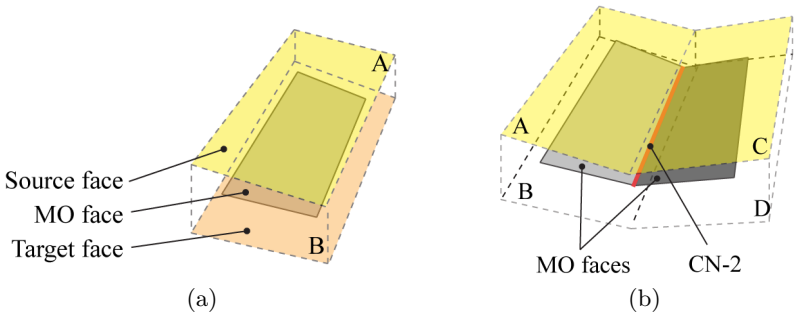


Fig. 6: A MO face represents a sweepable volume. The two defining entities are assigned as sweep source and target. (b) Two MO faces connected through a CN-2 curves represent two sweepable volumes are united. Surfaces A, C are the sweep source and surface B, D are the target.

Fig. 7 demonstrates our breadth-first-traversal MO segmentation algorithm on a MO. At the initial state (Fig. 7 (a)), we start from the largest MO face (L). In state 2, L is assigned to a group (yellow) and its neighbor MO faces are detected (Fig. 7 (b)). In state 3, the neighbor face across a CN-2 joins the current yellow patch (Fig. 7 (c)); other neighbor faces across a CN-3 remain unvisited (painted in white). The neighbors of the newly added faces are detected. In state 4, one more neighbor faces across a CN-2 joins the yellow patch (Fig. 7 (d)); the other two neighbors across a CN-3 remain unvisited. In state 5, the yellow patch has no more neighbors across CN-2 curves to visit (Fig. 7 (e)). The yellow patch is now complete. Then we start over again from the largest unvisited MO face (painted in green). One neighbor face across a CN-2 is detected and joins the green group. In state 6 (Fig. 7 (f)), the green group has no more neighbors across CN-2 s to visit. The green patch is now complete. Search for the largest unvisited MO face (painted in grey) and repeat the grouping procedure. In state 7 (Fig. 7 (g)), search for the largest unvisited MO face (paint in light blue). This MO face has no unvisited neighbor faces, and forms a stand alone patch. In the final state, only one unvisited face remains (painted in dark blue). This face forms another patch. Fig. 7(h) shows segmentation result for this trimmed MO. The algorithm is shown in Algorithm 1.

5.2 Generating MO Groups

MO faces that share the same CN-3 curve can be visualized by extruding their end curves along the CN-3 curve. The 3D sub-volumes that correspond to MO faces are also sweepable. As shown in Fig. 8, the MO is split into different sweepable patches by the CN-3 curve based on the 2-manifold segmentation as explained in previous section. Patches 1 and 2 are both sweepable along their respective sweeping directions defined by their defining entities However,

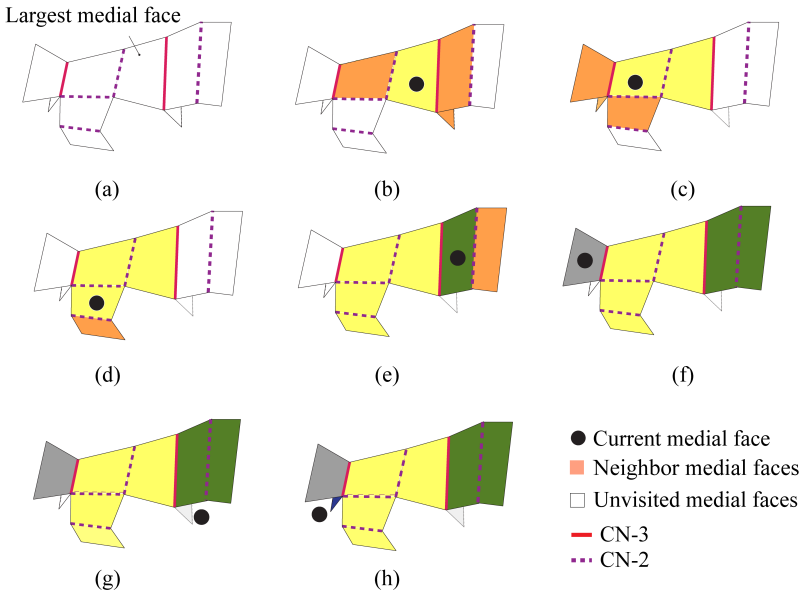


Fig. 7: A detailed MO segmentation example on a trimmed MO.

sub-volume corresponding to patches 1 and 2 is already sweepable without any decomposition if surfaces A and B are used as source and target, and swept along the CN-3 curve. Therefore, we unite these two patches together, and reduce the total number of sub-volume. If the MO patches that share the same CN-3 curve also share the same end entities as the sweep source and target, these MO patches will be grouped to represent a sweepable sub-volume. The advantage is that it reduces the number of sub-volumes and it assists in obtaining conformal mesh between the sub-volumes.

5.3 Highlight Ideal Cutting Regions

The model shown in Fig. 9(a) contains two sweepable groups but does not have any existing model entities to create a cutting surface. This happens when the defining surfaces of each sweepable MO group are overlapping. Our method detects the medial face that connects different MO groups and uses it to create a virtual volume. The overlapping portion of the virtual volume with the original models indicates the ideal cutting region, and the intersecting entities are highlighted. The user can then use the pen-based UI (Fig. 9(c)) to create a cutting surface at the highlighted region (Fig. 9(b)) using the freeform strokes. Conducting a decomposition operation in that region creates two sub-volumes that are sweepable.

Algorithm 1 MO segmentation algorithm

Input: The original model and its trimmed MO list $TrimmedMOFaceList$;
Output: MO patches;

while $TrimmedMOFaceList$ is not empty **do**
  Search for largest MO face (L) from $TrimmedMOFaceList$
  Append L to patch S_i
  Append L to $CurrentMOFaceList$
  Remove L from $TrimmedMOFaceList$
  while The number of $CurrentMOFaceList$'s neighbor MO faces ! = 0 **do**
    for each current MO face C **do**
      if the valence of C's child MO edge == 2 **then**
        for each child MO edge **do**
          Search for parent MO faces (P)
          Append P to S_i
          Append P to $CurrentMOFaceList$
          Remove P from $TrimmedMOFaceList$
        end for
      end if
      Remove C from $CurrentMOFaceList$
    end for
    Remove L from $CurrentMOFaceList$
  end while
  Remove P from $CurrentMOFaceList$
  $i++$
end while

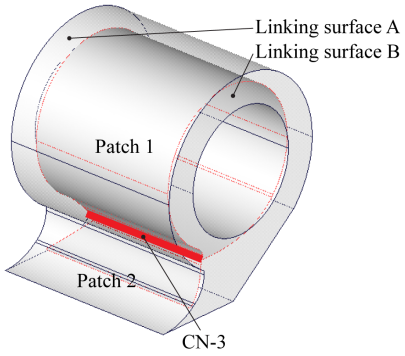


Fig. 8: A CN-3 curve splits two sweepable patches. Patches 1 and 2 share the same end MO curves and are sweepable by making linking surface A and B as source and target surfaces, repectively.

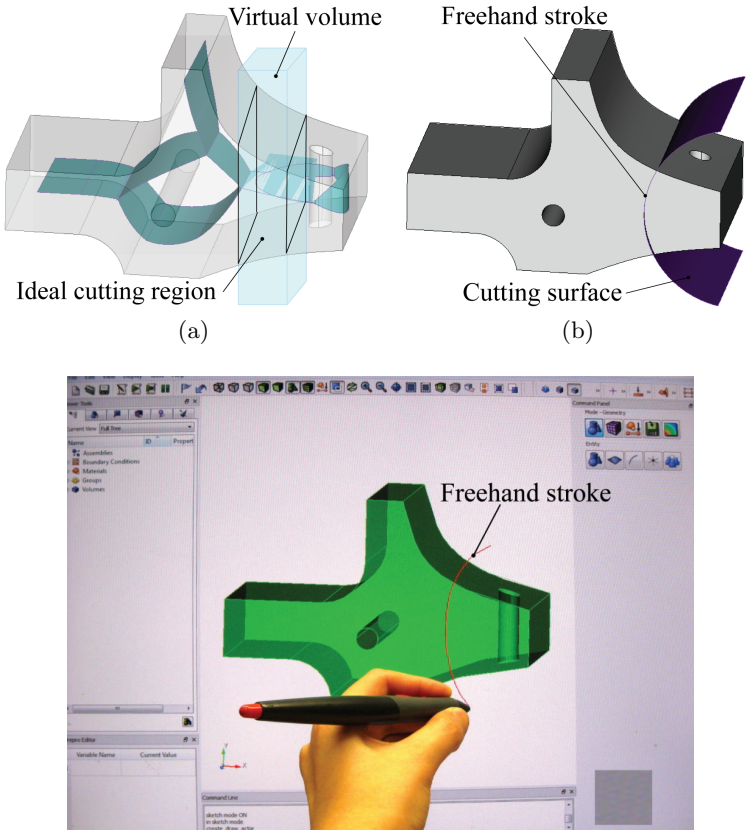


Fig. 9: (a) Highlight the ideal cutting region between two sweepable regions. (b) A cutting surface is created in the highlighted region with the pen-based UI in (c). (c) A user uses the pen-based UI to draw a freehand stroke.

6 Pen-based UI for Cutting Surface Creation Tools

This section describes the pen-based UI that enables the user to specify precise cutting surfaces with freehand strokes using one of the three tools: (1) Freeform based, (2) B-REP (boundary representation) based, or (3) MO based cutting tool. The pen-based UI intelligently selects the proper tool based on users input.

6.1 Freeform Based Cutting Tool

The pen-based UI takes user's freehand strokes as input, smoothes, beautifies, and snaps the strokes to create more precise cutting surfaces [13]. The freehand

strokes are beautified to common geometries such as lines, arcs, circles, and splines. Based on their position, strokes are also aligned to existing geometric features of the model surface. The supported alignment types include offset, overlap, perpendicular, and concentric. The mesh quality around the cutting area is improved by the alignment. Then the stroke is swept along the surface normal or the view direction to create a freeform cutting surface. This cutting surface is then used to decompose the volume into sub-volumes.

6.2 B-REP Based Cutting Tool

The B-REP based tool is able to automate the decomposition operation based on the entities selected by the user. The current implementation includes, (1) creating a cutting surface by extending a selected surface, (2) creating a cutting surface by revolving a selected surface, (3) creating a cutting surface by revolving a selected curve (Fig. 10(c)), and (4) creating a cutting surface by using a loop of bounding curves (Fig. 10(d)). The automatic selection of one of the four cutting operations is based on the number and type of selected geometric entities. If a user selects only one surface, the selected surface will be extended and the volume will be decomposed (Fig. 10(a)). If two entities are selected with the second one periodic, the first selected entity is assigned as a profile, and revolved about the axis defined by second selected entity as seen in Fig. 10(b) and (c). If one or more curves are selected, we first check if it form(s) a closed loop. If so, then the selected curve is used as a bounding curve to create a cutting surface (Fig. 10(d)).

6.3 MO Based Cutting Tool

The MO based tool uses MO to generate cutting surfaces. Fig. 11(a) shows part of the trimmed MO and a CN-3 curve. First resample the CN-3 curve and obtains the tangent point set, where the maximal spheres touch the model surfaces (Fig. 11(b), (c)). Next connect those tangent points to create polylines (Fig. 11 (d)).

Similar to the freeform based cutting tool that snaps the freehand stroke to existing entities [13], the MO based tool snaps the freehand stroke to these polylines. Then the user lofts (Fig. 11 (e)) the snapped strokes to create a cutting surface.

7 Results and Discussion

We have tested our MO assisted pen-based UI for volumetric decomposition against industrial models. These models are first imported in neutral STEP format into CUBIT. We then perform CAD cleanup operations such as CAD

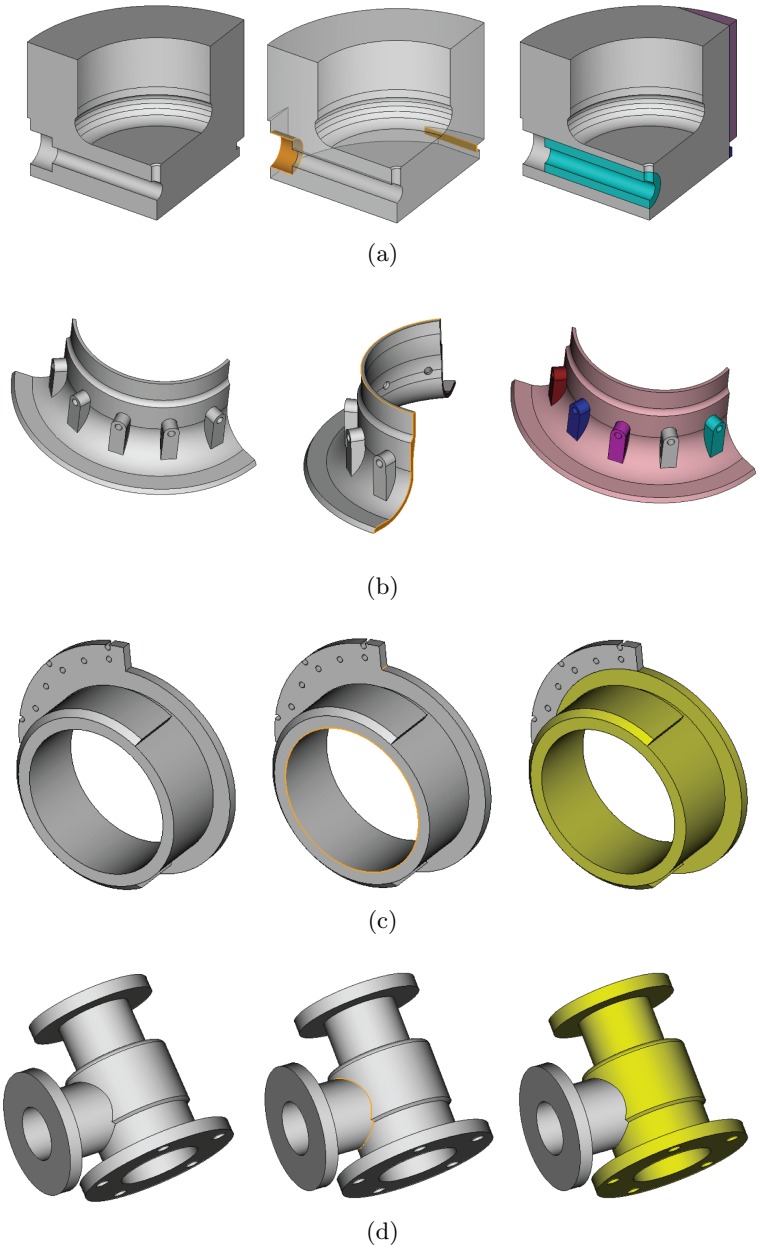


Fig. 10: The first column shows the original models. On the second column, the selected are highlighted in orange color. The last column shows the decomposition result using B-REP tool. (a) Two selected surfaces are extended. (b) A surface is revolved about the axis defined by another selected curve. (c) A curve is revolved about the axis defined by another selected curve. (d) Many selected curves from a close loop, and a cutting surface is lofted to decompose the model.

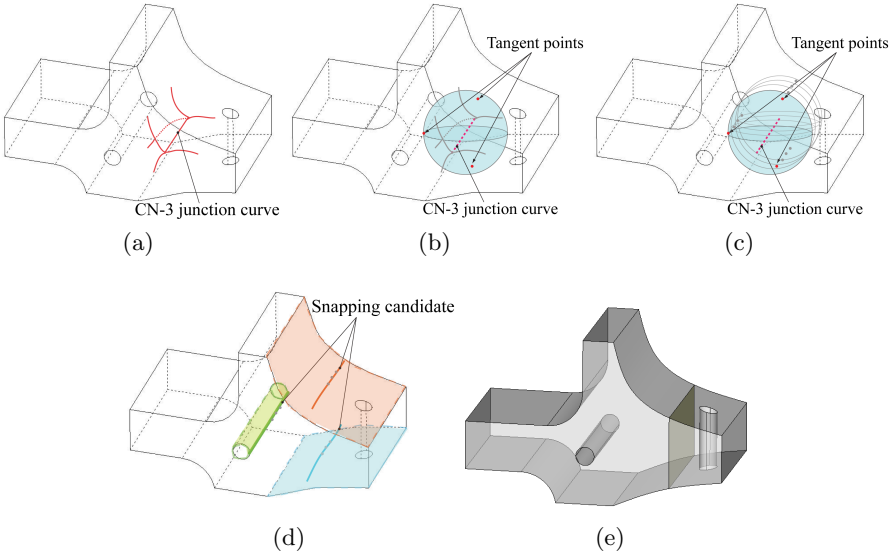


Fig. 11: (a) Part of the trimmed MO and the CN-3 junction curve. (b) The tangent points touch the model boundary. (c) The tangent points of the resampled CN-3 junction curve. (d) Create polylines by connecting tangent points. (e) Loft the two polylines to create a cutting surface.

Repair and defeaturing if needed. The volumetric decomposition is then performed using our proposed pen-based UI using MO. The volumetric decomposition step is very critical as it reduces the complex input model into simpler sub-volumes that are meshable. In order to obtain conformal mesh, we perform imprinting and merge operations. Finally, the traditional sweeping algorithms are called on the imprinted sub-volumes. Very high quality hex meshes can be obtained using the decomposition followed by the sweeping approach.

Fig. 12 shows a typical industrial model that cannot be meshed using the traditional sweeping algorithms. Trimmed MO shown in Fig. 12(b) is generated as the end branches of MAT are not useful for engineering purposes. The trimmed MO is then used to detect sweepable regions as shown in Fig. 12(c). The same figure also shows a hex mesh on the different sub-volumes using different colors. In this particular example, the model is sub-divided into ten sweepable sub-volumes. It took nine cutting surfaces to decompose this model. All the cutting surfaces are generated using one common gesture with the pen-based UI. The average mesh quality obtained on this model is 0.9404 as shown in Fig. 12(d). We are able to achieve such a high quality hex mesh because our approach decomposes the complex domain into simpler sub-volumes that can be meshed using the sweeping algorithm. Generally, the

sweeping algorithm leads to a better mesh quality than other hex meshing algorithms.

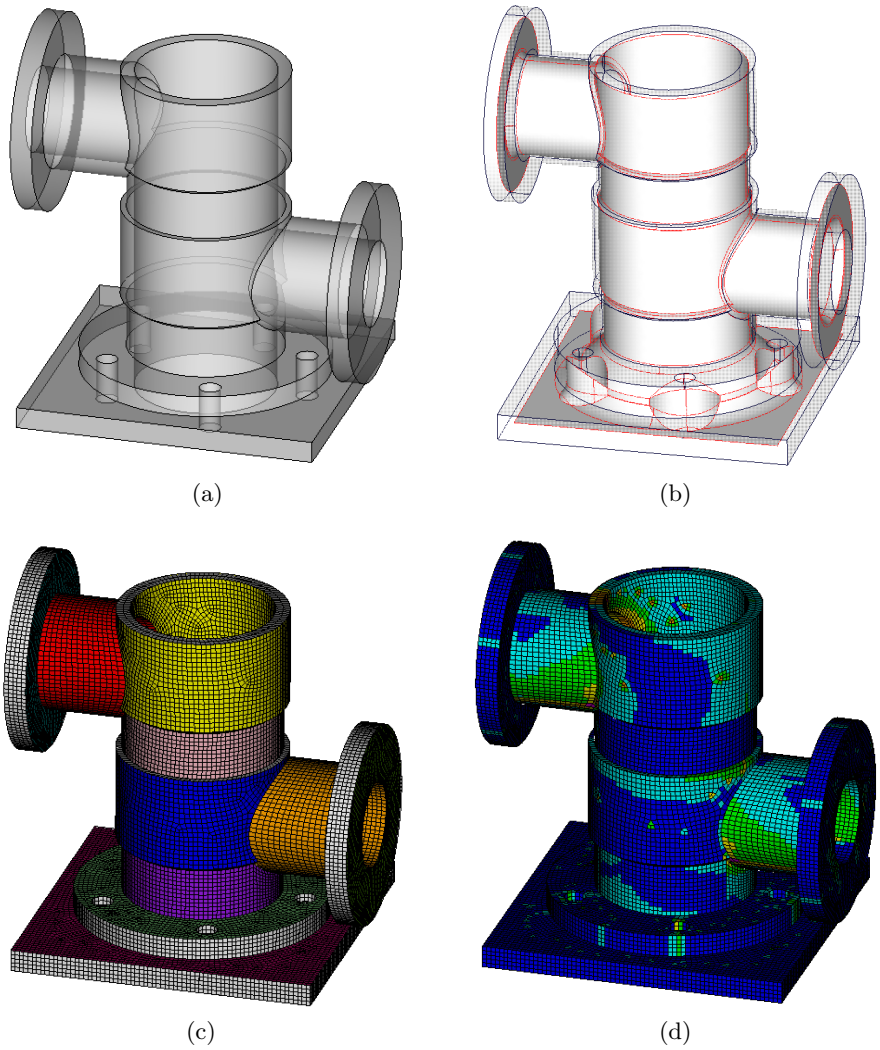


Fig. 12: (a) Original model. (b) MO with the original model boundary (black). (c) The all hex mesh on MO-based decomposition suggestion. (d) The mesh quality (Scaled Jacobian).

Fig. 13(a) shows another complex model that cannot be meshed using existing sweeping algorithms. Fig. 13(b) shows the trimmed MO that contains many non-manifold regions. Fig. 13(c) shows the sub-volumes obtained using

MO-based visual suggestion and the pen-based UI. The model is sub-divided into sixteen sweepable sub-volumes. The sweepable regions are color coded using different colors to provide visual aid to the users. Pen-based UI that provides three types of cutting tool is used to decompose the model based on the MO grouping suggestions. The system automatically selects the correct decomposition command based on the type and the number of geometric entities selected. Proposed approach uses an internal geometric kernel to perform the low level decomposition operations. Also, the imprint and merge operations are performed using the geometric kernel. The final conformal hex mesh shown in Fig. 13(d) contains 93653 elements with an average quality of 0.9345 (Scaled Jacobian).

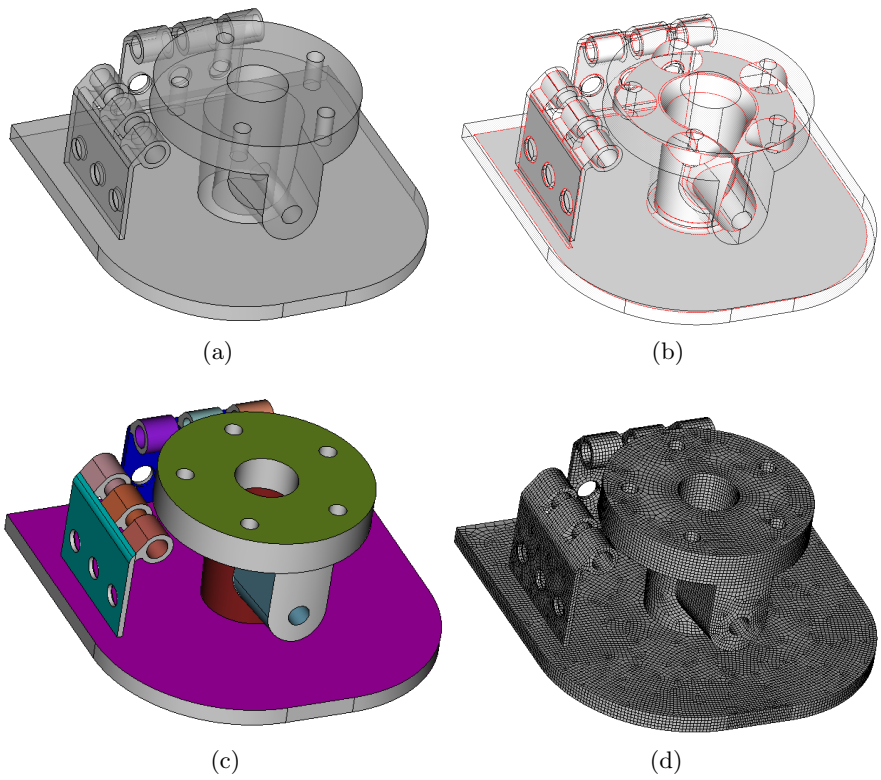


Fig. 13: (a) Original model. (b) Original model with trimmed MO. (c) The MO segmentation and grouping result. Sixteen groups are painted in different colors. (d) The all hex mesh output.

8 Conclusion

The paper presents an intelligent and user friendly pen-based UI that provides MO-based decomposition suggestions and automates the selection of difference components to make volumetric decomposition semi-automatic. One of the primary contributions of this paper is the MO segmentation and grouping algorithm that detects sweepable regions and provides visual aid for ideal cutting region. Another contribution is the pen-based UI that creates precise cutting surfaces using freeform strokes, existing B-Rep entities, and MO. In the future, cutting surface creation via MO and selection of cutting tools will be further automated. The proposed approach has been tested on industrial models by generating high quality hex meshes using sweeping algorithms.

Acknowledgement. The authors would like to thank Dr. Geoffrey Butlin, Mr. Henry Bucklow, Mr. Robin Fairey, Mr. Mark Gammon and Mr. Mike Field for assisting with the medial related work in CADFIX. The authors would also like to thank Dr. Soji Yamakawa for his valuable advise; Mr. Ved Vyas for his expertise with CUBIT; and Mr. Karthik Srinivasan for collecting MAT related articles.

References

1. S. Yamakawa, I. Gentilini, and K. Shimada, "Subdivision templates for converting a non-conformal hex-dominant mesh to a conformal hex-dominant mesh without pyramid elements," *Engineering with Computers*, vol. 27, pp. 51–65, 2011.
2. T. D. Blacker and R. J. Meyers, "Seams and wedges in plastering: A 3D hexahedral mesh generation algorithm," *Engineering with Computers*, vol. 9, pp. 83–93, 1993.
3. M. Hariya, I. Nishigaki, I. Kataoka, and Y. Hiro, "Automatic hexahedral mesh generation with feature line extraction," in *Proceeding of the 15th International Meshing Roundtable*, pp. 453–468, 2006.
4. L. Marechal, "A new approach to octree-based hexahedral meshing," in *Proceeding of the 16th International Meshing Roundtable*, pp. 209–221, 2001.
5. S. Yamakawa and K. Shimada, "HEXHOOP: modular templates for converting a hex-dominant mesh to an all-hex mesh," *Engineering with Computers*, vol. 18, pp. 211–228, 2002.
6. D. R. White and T. J. Tautges, "Automatic scheme selection for toolkit hex meshing," *International Journal for Numerical Methods in Engineering*, vol. 49, no. 1-2, pp. 127–144, 2000.
7. W. R. Quadros and K. Shimada, "Hex-layer: Layered all-hex mesh generation on thin section solids via chordal surface transformation," in *Proceeding of the 11th International Meshing Roundtable*, pp. 169–182, 2002.
8. M. A. Price, C. G. Armstrong, and M. A. Sabin, "Hexahedral mesh generation by medial surface subdivision: part I. solids with convex edges," *International Journal for Numerical Methods in Engineering*, vol. 38, no. 19, pp. 3335–3359, 1995.

9. M. A. Price and C. G. Armstrong, "Hexahedral mesh generation by medial surface subdivision: part II. solids with flat and concave edges," *International Journal for Numerical Methods in Engineering*, vol. 40, no. 1, pp. 111–136, 1997.
10. Simulia Corp., "Abaqus product description, version 6.9," in $http : //www.simulia.com/products/abaqus_fae.html$.
11. Sandia National Laboratories, "Cubit 12.1 on-line user's manual: web cutting," in $http : //cubit.sandia.gov/help - version12.1/cubithelp.htm$.
12. S. J. Owen, B. Clark, D. J. Melander, M. L. Brewer, J. Shepherd, K. G. Merkley, C. Ernst, and R. Morris, "An immersive topology environment for meshing," in *Proceeding of the 16th International Meshing Roundtable*, pp. 553–577, 2007.
13. J. H.-C. Lu, I. H. Song, W. R. Quadros, and K. Shimada, "Pen-based user interface for geometric decomposition for hexahedral mesh generation," in *Proceedings of the 19th International Meshing Roundtable*, pp. 263–278, 2010.
14. H. Blum, "A transformation for extracting new descriptors of shape," in *Models for the Perception of Speech and Visual Form*, pp. 362–380, 1967.
15. W. Quadros, "Extraction and applications of skeletons in finite element mesh generation," in *Proceeding of the 7th International Conference on Engineering Computational Technology*, 2010.
16. Y. Lu, R. Gadh, and T. J. Tautges, "Feature based hex meshing methodology: feature recognition and volume decomposition," *Computer-Aided Design*, vol. 33, no. 3, pp. 221–232, 2001.
17. D. R. White, S. Saigal, and S. J. Owen, "Ccsweep: automatic decomposition of multi-sweep volumes," *Engineering with Computers*, vol. 20, pp. 222–236, 2004.
18. B.-Y. Shih and H. Sakurai, "Automated hexahedral mesh generation by swept volume decomposition and recomposition," in *Proceeding of the 5th International Meshing Roundtable*, pp. 273–280, 1996.
19. B.-Y. Shih and H. Sakurai, "Shape recognition and shape-specific meshing for generating all hexahedral meshes," in *Proceeding of the 6th International Meshing Roundtable*, pp. 197–209, 1997.
20. P. Y. Ang and C. G. Armstrong, "Adaptive curvature-sensitive meshing of the medial axis," in *Proceeding of the 10th International Meshing Roundtable*, pp. 155–165, 2001.
21. T. S. Li, R. M. McKeag, and C. G. Armstrong, "Hexahedral meshing using midpoint subdivision and integer programming," *Computer Methods in Applied Mechanics and Engineering*, vol. 124, no. 1-2, pp. 171–193, 1995.
22. R. J. Donaghy, C. G. Armstrong, and M. A. Price, "Dimensional reduction of surface models for analysis," *Engineering with Computers*, vol. 16, pp. 24–35, 2000.
23. P. Sampl, "Semi-structured mesh generation based on medial axis," in *Proceeding of the 9th International Meshing Roundtable*, pp. 21–32, 2000.
24. C. S. Chong, A. S. Kumar, and K. H. Lee, "Automatic solid decomposition and reduction for non-manifold geometric model generation," *Computer-Aided Design*, vol. 36, no. 13, pp. 1357–1369, 2004.
25. R. Sun, S. Gao, and W. Zhao, "An approach to B-rep model simplification based on region suppression," *Computers and Graphics*, vol. 34, no. 5, pp. 556–564, 2010.
26. W. R. Quadros, K. Shimada, and S. J. Owen, "Skeleton-based computational method for the generation of a 3D finite element mesh sizing function," *Engineering with Computers*, vol. 20, pp. 249–264, 2004.