A new grid-based midsurface generation algorithm

Christoph Schinko[1] and Torsten Ullrich[2]

Fraunhofer Austria Research GmbH & Graz University of Technology, Austria
[1]christoph.schinko@fraunhofer.at, [2]torsten.ullrich@fraunhofer.at

Abstract

A midsurface consists of surface elements (2D) representing three-dimensional, thin solids whose local thickness is small compared to its other dimensions. These simplified representations allow a significant acceleration of almost all physical simulations – a prerequisite for rapid prototyping in the context of 3D additive manufacturing. In this article we present a new algorithm that can extract a midsurface of a thin solid using a grid-based approach: The first step builds a binary 3D grid and marks each voxel that contains some part of the input surface geometry. The second step interprets the binary grid as a density field and applies a kernel to it. In the third step, the vertices of the original input data are moved within the density field towards the climax. Finally, vertices which approach each other are clustered and merged. Having merged all clusters of vertices, the resulting two-sided, incident surfaces are merged to one-sided surfaces. The result is a midsurface. The algorithm has been designed to run fully automatic and the presented proof of concept delivers convincing results.

Introduction

Dimension reduction transformation is widely used in Computer-Aided Engineering (CAE) analysis as it substantially reduces the compexity of physical simulations. Computer-Aided Design (CAD) models are therefore transformed into a representative surface known as “Midsurface” [1]. It is a simplified, idealized 2D-surface lying (mostly) midway of a thin-walled CAD model and mimicking its shape. Midsurfaces are part of the family of medial objects representing simplified versions of thin-walled, solid input objects. The goal of a midsurface is to reduce the complexity of simulations by switching from 3D volume representations to 2D surface representations. As a consequence the midsurface definition depends (i) on the input and (ii) on the output of a subsequent analysis:

Definition: The *midsurface* of a closed, thin-walled object is a set of surfaces which reveal in an CAE analysis the same results (within a δ-tolerance) as the closed, thin-walled input model.

Therefore, the correctness of any midsurface transformation can only be examined within a given context. Nevertheless, the simplified representation of a midsurface reduces the number of features of an object – according to its definition without significant differences in the simulation results. Due to less degrees of freedom (DOF) the simplified model allows a considerable acceleration of almost all physical simulations; thus, it saves time and resources. Consequently, the automatic extraction of reasonable midsurfaces is an important field of research. This article presents a new grid-based approach to solve this problem.

Related Work

The creation of midsurface geometry is an area of extensive research. Still, a lot of manual labor is required to create reduced dimensional models from solid geometries, both in terms of computational geometry and surface reconstruction. Naturally, related work can be found in these two categories.

Top-Down

Many approaches of midsurface construction work on boundary representations [12] in a top-down manner; i.e. a complex part is split into small parts, which can be transformed into a midsurface representation more easily. The first step of these approaches is usually a decomposition step according to semantic interrelationships [14]. Due to the inherent difficulty of finding a general solution to the midsurface generation problem, Y. H. Woo and C. U. Choo propose a divide and conquer approach [15]. The work of Nolan et al. presents an approach to automatically create mixed dimensional meshes [9]. It uses feature detection to cluster a surface into three categories: long-slender regions, thin-sheet regions, and complex regions. The state-of-the-art approach based on decomposition [4], on identification of corresponding faces [2], and on a Divide-and-Conquer strategy [3] has been developed by Y. H. Kulkarni: "Development of Algorithms for Generating Connected Midsurfaces using Feature Information in Thin-Walled Parts" [1].

Bottom-Up

Due to the nature of our approach being grid-based, this Section focuses on a subgroup of so-called thinning approaches. The idea behind these approaches is to iteratively shrink an object locally by offsetting its boundary towards its interior. This process is stopped, when the interior vanishes and just a (potentially non-manifold) surface structure remains. Work on thinning approaches dates back to the late seventies: P. Kwok and V. Ranjan give a good overview of the most important thinning algorithms up to 1991 [5]. All presented algorithms work on voxel data and eventually result in a skeleton rather than a surface. S. Prohaska and H.-C. Hege propose a robust, noise resistant criterion for the characterization of plane-like skeletons [11]. Their algorithm is based on a distance map and the geodesic distance along the object's boundary. After obtaining a skeleton of lines and surfaces, the focus is on obtaining a surface representation for expressive rendering of complex structures. In contrast to the already mentioned algorithms that return skeletons, the work by T.-C. Lee and R. L. Kashyap on thinning algorithms is concerned with parallel thinning for extracting medial surfaces and medial axes of binary voxel data [6]. Another thinning algorithm for extracting medial surfaces is presented by C.-M. Ma and S.-Y. Wan [7]. It defines two sets of voxels on the gridthat are iteratively reduced. It is proven to preserve connectivity. Another thinning approach for medial surface extraction is presented by K. Palágyi and G. Németh [10]. The proposed algorithms rely on sufficient conditions for parallel, topology-preserving reduction operators. A novel thinning scheme using iteration-level endpoint checking is proposed in follow-up work [8].

Midsurface Generation

The new grid-based approach consists of four steps, which are illustrated in Figure 1.

Rasterization Step

The new algorithm can be applied to triangle meshes only. This prerequisite is not a limitation as most CAD systems offer the possibility to export the tessellated geometry (e.g. in STL format). Besides the tessellated geometry the algorithm needs two additional values: ε and δ, which are the smallest distance and the greatest distance between two opposite surfaces that should be merged to one midsurface. All other subsequent parameters are set automatically and do not need to be modified.The first step builds a binary 3D grid and marks each voxel that has a non-empty intersection with the input surface geometry. If the grid size α, i.e. the edge length of each cube in the voxel grid, is not specified, the algorithm uses the default value $\alpha = \frac{1}{2}\varepsilon$.

Voxel Kernel

The second step interprets the binary grid as a density field with the values 0.0, if a voxel does not contain any geometry, and 1.0 otherwise. The main idea of the second step is to spread the density values using a kernel. The overlaps of the kernels at the position of the midsurface result in significantly higher values compared to their neighborhood. The algorithm uses a uniform distribution over a 3D sphere with radius κ. The implementation simply sums for each voxel (i, j, k) the values of all surrounding voxels (r, s, t) with an Euclidean distance less than κ between the voxels' centers. The default value is $\kappa = \frac{1}{2}\delta$.

Figure 1: *The new midsurface generation algorithm samples the input geometry (top row, left) to a binary 3D grid (top row, middle). Then the binary grid is interpreted as a density field and applies a kernel to it (top row, right; cut for visibility). Afterwards, the vertices of the original input data are moved within the density field towards the climax (bottom row, left). Finally, vertices which approach each other are clustered and merged. The result is a midsurface. Input and output geometry are shown in the bottom-right rendering (cut for visibility).*

Vertex Climax

In the third step, the vertices of the original input tessellation data are moved within the density field generated in the previous steps. The start position of a moving vertex is the center of its corresponding voxel. Then each vertex moves to the center of a neighboring voxel with the highest density until no improvement concerning density values is possible or the length of the movement path exceeds the greatest distance δ between two opposite surfaces that should be merged to one midsurface.

Vertex Clustering & Topological Cleaning

Having moved all vertices, the resulting set of vertices is clustered using a mean-shift clustering algorithm with a window size ω with default value δ. The result is already a two-sided midsurface. If a one-sided midsurface representation is needed, a dense resampling of the two-sided midsurface and a reconstruction according to Shawn and Watson [13] returns the desired representation.

Conclusion

A midsurface allows a significant acceleration of almost all physical simulations. With a few exceptions, these midsurfaces cannot be generated fully automatically. The acceleration of the subsequent simulation currently justifies even the manual creation of midsurfaces, so that manual re-engineering is still the first choice in many engineering offices.

This article directly targets the interplay of theory and practice: we present a new algorithm to extract a midsurface of a thin solid automatically using a grid-based approach. The first concept already delivers convincing results. The algorithm has been designed to run fully automatic. This is an important property for automated tests and simulations in a rapid prototyping scenario.

Acknowledgments

The authors would like to thank Ulrich Krispel and Andreas Riffnaller-Schiefer for fruitful discussions; Peer Bertram and Martin Schwarz for their expert knowledge in construction and simulation. Furthermore, the authors gratefully acknowledge the support of the Austrian Research Promotion Agency, (Forschungsförderungsgesellschaft) for the research project (K-Projekt) "Advanced Engineering Design Automation (AEDA)".

References

[1] Kulkarni, Y. H. *Development of algorithms for generating connected midsurfaces using feature information in thin walled parts*. PhD Thesis at Savitribai Phule Pune University, 2016.

[2] Kulkarni, Y. H., and Kale, M. Formulating midsurface using shape transformations of form features. *Proceedings of All India Manufacturing Technology, Design and Research 5* (2014), 981–985.

[3] Kulkarni, Y. H., Sahasrabudhe, A., and Kale, M. Dimension-reduction technique for polygons. *International Journal of Computer-Aided Engineering and Technology 9* (2017), 1–17.

[4] Kulkarni, Y. H., Sahasrabudhe, A., and Kale, M. Leveraging feature generalization and decomposition to compute a well-connected midsurface. *Engineering with Computers 33* (2017), 159–170.

[5] Kwok, P. C. K., and Ranjan, V. A survey of 3-d thinning algorithms. *Vision Interface 6* (1991), 13–20.

[6] Lee, T.-C., and Kashyap, R. L. Building skeleton models via 3-d medial surface/axis thinning algorithms. *Graphical Models and Image Processing (CVGIP) 56* (1994), 462–478.

[7] Ma, C.-M., and Wan, S.-Y. A medial-surface oriented 3-d two-subfield thinning algorithm. *Pattern Recognition Letters 22* (2001), 1439–1446.

[8] Németh, G., Kardos, P., and Palágyi, K. Topology preserving 3d thinning algorithms using four and eight subfields. *Image Analysis and Recognition 6111* (2010), 316–325.

[9] Nolan, D. C., Tierney, C. M., Armstrong, C. G., Robinson, T. T., and Makem, J. E. Automatic dimensional reduction and meshing of stiffened thin-wall structures. *Engineering with Computers 30* (2014), 689–701.

[10] Palágyi, K., and Németh, G. Fully parallel 3d thinning algorithms based on sufficient conditions for topology preservation. *Discrete Geometry for Computer Imagery (DGCI) 5810 (LNCS)* (2009), 481–492.

[11] Prohaska, S., and Hege, H.-C. Fast visualization of plane-like structures in voxel data. *IEEE Visualization 13* (2002), 29–36.

[12] Schinko, C., Riffnaller-Schiefer, A., Krispel, U., Eggeling, E., and Ullrich, T. State-of-the-art overview on 3d model representations and transformations in the context of computer-aided design. *International Journal on Advances in Software 3 & 4* (2017), 446–458.

[13] Shawn, M., and Watson, J.-P. Non-manifold surface reconstruction from high-dimensional point cloud data. *Computational Geometry 44* (2011), 427–441.

[14] Sun, L., Tierney, C. M., Armstrong, C. G., and Robinson, T. T. Automatic decomposition of complex thin walled cad models for hexahedral dominant meshing. *Procedia Engineering 163* (2016), 225–237.

[15] Woo, Y.-H., and Choo, C.-U. Automatic generation of mid-surfaces of solid models by maximal volume decomposition. *Transactions of the Society of CAD/CAM Engineers 14* (2009), 297–305.