Published in final edited form as:

Proc Int Conf Biomed Eng Inform. 2010 October 16; 2010: 472–476. doi:10.1109/BMEI.2010.5639253.

Quality Multi-domain Meshing for Volumetric Data

Qin Zhang[CVC],

Institute for Computational Engineering and Sciences, the University of Texas, Austin, TX 78712

Bharadwaj Subramanian[CVC],

Institute for Computational Engineering and Sciences, the University of Texas, Austin, TX 78712

Guoliang Xu[LSEC], and

Institute of Computational Mathematics, Academy of Mathematics and System Sciences, Chinese Academy of Sciences, Beijing, 100190

Chandrajit L. Bajaj[CVC]

Institute for Computational Engineering and Sciences, Department of Computer Science, the University of Texas, Austin, TX 78712

Qin Zhang: zqyork@ices.utexas.edu; Bharadwaj Subramanian: bharadwaj.subramanian@gmail.com; Guoliang Xu: xuguo@lsec.cc.ac.cn; Chandrajit L. Bajaj: bajaj@cs.utexas.edu

Abstract

Multi-domain meshing from volumetric data is of great importance in many fields like medicine, biology and geology. This paper proposes a new approach to produce a high quality mesh with separated multiple domains. A point cloud is generated from a preliminary mesh representing the boundary between different domains from the discrete volumetric representation used as input. A higher-order level-set method is employed to produce a quality sub-mesh from this point cloud and geometric flow is used as smoothing mechanism. A new approach to detect and curate intersections within an assembly of these 2-manifold sub-meshes by utilizing the intermediate volumetric representation is developed. The separation between sub-meshes can be controlled by the user using a gap threshold parameter. The resulting high quality multi-domain mesh is free from self- and inter-domain intersections and can be further utilized in finite element and boundary element computations. The proposed pipeline has been efficiently implemented and sample meshes have been provided for visualization.

I. INTRODUCTION

The problem of generating of multi-domain non-intersecting meshes from volumetric data is of great interest in the biological [1], chemical, medical [2] fields. For instance, in CT imaging or MRI of the human body, the domain of interest often consists of regions of different functionality. In such cases, it is desirable to segment and individually mesh these regions for the purpose of visualization and analysis. Figure I.1 gives an example of such a case where an MRI of the brain has been segmented into 41 domains and meshed individually. Each color represents a single brain region and a small gap exists between different regions. Figure I.1. (A) shows the original volumetric data, figure (B) shows the multi-domain surface and figure (C) shows a cut-view of figure (B) and zooming the part in the green box. We can clearly see small gaps between different sub-meshes.

The problem of generating a manifold mesh from volumetric data is an isosurface discretization problem. Two widely used methods for producing a single domain mesh are the Marching Cubes (MC) and the Dual Contouring methods, due to Lorensen and Cline [3] and Ju et al. [4] respectively.

In this paper, a new multi-domain meshing pipeline is proposed for producing a high quality mesh composed of an assembly of multiple non-intersecting 2-manifold sub-meshes. The sub-meshes do not share a common boundary, and the separation between them can be controlled by a parameter known as the gap threshold. A mesh separation algorithm guarantees that there are no intersections or osculations between any two sub-meshes. Hence, the produced mesh is of a high quality and may be directly used in scientific computation and in other application areas like surgical planning. The above pipeline has been implemented and sample meshes have been generated and illustrated in the following pages.

The reminder of this paper is organized as follows. In Section II, the problem is formulated and an overall sketch of the pipeline is given. This is followed by the details for several steps of the pipeline with examples for demonstrating and verifying its practicality and effectiveness in Section III. Section IV summarizes the results. Finally, Section V concludes the paper and presents some future research topics.

II. SKETCH OF MULTI-DOMAIN MESHING ALGORITHM

The problem considered in this paper can be formulated as follows. Given a volumetric density map, expressed by a trivariate function as

$$\begin{aligned} f: \Omega \subset \mathbb{R}^3 &\rightarrow \mathbb{R} \\ (x_i, y_j, z_k) &\rightarrow f(x_i, y_j, z_k), i, j, k=1, \dots, n. \end{aligned}$$

The goal is to partition Ω into multiple domains Ω_i , such that $\Omega = \bigcup \Omega_i$ and for each domain of interest Ω_i , to produce a mesh $M_i (i = 1, \dots, N)$ such that $M_i \cap_{i \neq j} M_j = \emptyset$.

The index $i (i = 1, \dots, N)$ is referred to as the domain id of Ω_i . The volume corresponding to domain Ω_i is V_i and the isovalue for mesh M_i from volume V_i is c_i . Note that mesh M_i may be composed of multiple components. Since the mesh M_i is operated upon in several steps along the pipeline, the same symbol will be used through this paper to refer to the mesh; if an ambiguity arises with notation, new notation will be introduced as necessary.

The complete pipeline is outlined in Figure II.1, where the gray boxes represents the data being operated upon or obtained from the previous step, and the yellow boxes summarize the approach or technique used. And the step in the yellow box with green boundary is the main technique which will be detailed in next section.

III. ALGORITHM DETAILS

A. Segmentation and Classification

Current multi-seeded fast marching methods are used to segment 3D density maps into individual domains of interest [5] The basic idea of this method is that a contour, initialized from a seed point chosen a priori, is allowed to grow until a certain stopping condition is reached. Each voxel is assigned with a value called “time”, which is initially set to zero for seed points and infinity for all other voxels. The voxel on the marching contour with minimal time value is removed from the contour iteratively and the time values of its neighbors are updated according to the equation

$$\|\nabla T(\mathbf{x})\|F(\mathbf{x})=1,$$

where $F(\mathbf{x})$ is called the speed function, and is usually determined by the gradients of the density map. The neighbors are then inserted into the contour if they are being updated for the first time. In order to achieve effective results, the three key problems in this approach which need to be solved are the formulation of the speed function, the initialization of the seed points and determination of the stopping criterion.

The density map is then classified into several different domains after segmenting it into multiple components of interest. Multiple components are classified as belonging to the same domain if their density maps are nearly identical after symmetry, rotation and reflection are taken into consideration. Prior knowledge from the field of study is also greatly useful for classification. After classification, a multi-domain volumetric map is obtained:

$$\begin{aligned} g: \mathbb{R}^3 &\rightarrow \mathbb{N} \\ (x_i, y_j, z_k) &\rightarrow g(x_i, y_j, z_k), \end{aligned}$$

here $\mathbb{N}$ is the set of natural numbers.

B. Point Cloud Generation

This step consists of partition surface extraction and preliminary smoothing and point cloud generation steps.

1) Partition surface extraction—In this step, partition surfaces are produced from the multi-domain map produced from the previous segmentation and classification step. A partition surface is a surface whose interior includes voxels from a single domain only. In order to produce a partition surface for a domain with id i , a cube (or cell) with each vertex with domain id i is taken and a dual cube with six faces with face id i is produced. If two dual cubes have common faces with the same face id, say i , then two cubes are combined into one bigger cuboid, and so on. If two dual cubes have common faces with different face id, say i and j ($i \neq j$), two different surfaces are produced. Note that the produced surfaces may not be manifold since they may include non-manifold vertices and/or edges.

2) Preliminary smoothing—The obtained partition surfaces may not be smooth in general, and a HLS reconstruction may result in a mesh with excessive, unnecessary features. Furthermore, it was found experimentally that a preliminary smoothing of the partition mesh results in a final multi-material mesh with much lesser inter-mesh intersections. Hence, geometric flow is used to perform a preliminary smoothing step. Geometric PDEs, such as the mean curvature flow, the surface flow and Willmore flow have been extensively used in surface and image processing [6]. In the pipeline described in this paper, surface diffusion flow with a tangential movement modification is chosen for smoothing:

$$\frac{\partial \mathbf{x}}{\partial t} = \Delta H \mathbf{n} + v \mathbf{t}, \quad (\text{III.1})$$

where $\mathbf{n}$, $\mathbf{t}$ are the normal and tangent vectors of vertex $\mathbf{x}$, H is the mean curvature, and Δ is the Laplace-Beltrami operator, v is the velocity in the tangential direction and t is a marching parameter.

Each term of the above expression (III.1) is discretized over the partition mesh, including the mean curvature H and the Laplace-Beltrami operator [7], and the equation is solved over the mesh using the forward Euler method.

3) Point cloud generation—Although the partition surfaces have been smoothed, it is very difficult to reconstruct a smooth surface since in general a partition surface is non-manifold. Hence, a different approach is used where a point cloud is produced to reconstruct a smooth manifold surface. In order to generate the point cloud, each triangle of the partition surface is subdivided into four triangles if the length of any edge of the triangle is greater than a threshold.

C. Higher-Order Level Set Method Based Surface Reconstruction

The produce of smooth surfaces from a point cloud is a challenging task and large amount of research has been done in the past on this problem. In [8], a higher-order level-set method is proposed to construct smooth surfaces for molecules and scattered data sets. The reconstructed smooth surface is represented by the level set of a tri-cubic B-spline function. Several important issues related to this approach here are initial surface construction, thin shell technique, adaptive reinitialization of the signed distance function, conversion of piecewise tri-linear function to tri-cubic B-spline and iso-contouring methods.

D. Intersection Detection and Curation

Following the surface reconstruction from a point cloud (subsection III-C), a volume V_i and a mesh M_i with prescribed isovalue c_i for each domain Ω_i are obtained. Although a small isovalue may be selected to prevent intersections between different meshes, doing so may result in large separations between sub-meshes which is generally not acceptable. On the other hand, if surfaces with a larger isovalue are reconstructed from the volumes, intersections between different domains may appear. Hence, there is a trade-off between intersections and the extent of separation between sub-meshes. A balanced way of selecting a suitable isovalue is by taking into account the initial mesh obtained after separation but keep no serious intersections between different meshes. It was found experimentally that this approach produces a quality mesh with correct topology and acceptable gaps between sub-meshes.

1) Intersection detection—In order to prevent intersection between surfaces, the presence of an intersection between any two meshes must be checked. The proposed algorithm checks whether vertices belonging to one mesh exist within the interior of another mesh and vice versa. This intersection detection test is valid since it is known that a point in the volume exists in the interior of the mesh representing the isosurface if and only if the function value at that point is less than the isovalue of the mesh. The following example explains this more clearly.

Suppose there are two meshes M_1 with P_1 vertices and M_2 with P_2 vertices from their corresponding volumes V_1 and V_2 with isovalue c_1 and c_2 . For the vertex $\mathbf{v}_{1i}$ from mesh M_1 , if $f(\mathbf{v}_{1i}) < c_2$, then it must be in the interior of mesh M_2 . If $f(\mathbf{v}_{1i}) > c_2$ for $\mathbf{v}_{1i}$, $i = 1, \dots, P_1$ in M_1 , there is no intersection between M_1 and M_2 . The same is for the vertices in M_2 with mesh M_1 . The total number of vertex $\mathbf{v}_{1i}$ which exist in the interior of M_2 are denoted by p_{12} and the total number of vertex $\mathbf{v}_{2i}$ which exist in the interior of M_1 is denoted by p_{21} .

2) Intersection curation—Suppose M_1 and M_2 have intersections with p_{12} vertices of M_1 in M_2 and p_{21} vertices of M_2 in M_1 . In order to obtain meshes M'_1 and M'_2 without intersections, it is required that $p'_{12}=p'_{21}=0$. Although the mesh M_1 is extracted from volume

V_1 with isovalue c_1 , the function value $f(\mathbf{v}_{1i})$ is not identical to c_1 and depends on the isosurface extraction method used. For instance, if the improved primal contouring is used the error between the function value and isovalue is small. On the other hand, if a dual contouring method is used, the error is larger since the vertices of the mesh from dual contouring approach is just a minimizer which is not guaranteed to lie on the isosurface. Hence, a remedial step is carried out where the vertices of each mesh are projected back to their corresponding isosurface within an small error tolerance ϵ so that $f(\mathbf{v}_{ji}) = c_j \pm \epsilon$.

For vertex $\mathbf{v}_{2i}(i = 1, \dots, p_{21})$ of mesh M_2 , the following partial differential equation is then used to move it outside of the mesh M_1 ,

$$\frac{\partial \mathbf{v}'_{2i}}{\partial t} = \tilde{d}_{21i} * \mathbf{n}_{21i}, \quad (\text{III.2})$$

where $\mathbf{v}'_{2i}$ is the new position of vertex $\mathbf{v}_{2i}$, $\tilde{d}_{21i}$ is an improved version of the distance d_{21i} between vertex $\mathbf{v}_{2i}$ with the isosurface from volume V_1 with isovalue c_1 . $\mathbf{n}_{21i}$ is the normal of vertex $\mathbf{v}_{2i}$ calculated from volume V_1 . The explicit form for $\tilde{d}_{21i}$ is given by

$$\tilde{d}_{21i} = (1 + \delta) d_{21i},$$

where $\delta > 0$ is called gap measure which should be than ϵ . d_{21i} is the distance between vertex $\mathbf{v}_{2i}$ with the isosurface from volume V_1 with isovalue c_1 and it can be calculated as

$d_{21i} = \frac{|f(\mathbf{v}_{2i})|}{\|\nabla f(\mathbf{v}_{2i})\|}$. Generally, the bigger δ is, the wider the gap. Taking the error of normal computation and speed into consideration, the Euler forward method is used to solve this equation. The process is iterated until $|f(\mathbf{v}'_{2i}) - (1 + \delta)d_{21i}| < \epsilon$, where $\epsilon > 0$ is small.

We show a 3D example in Figure III.1, where figure (A) shows three meshes after domain partition and smoothing. The thin shell in figure (A) is shown in figure (B) and (C) as two different views, and we can see clearly the intersection of the red and yellow meshes. Figure III.1. D shows the result after reconstruction from point clouds and intersection detection and curation steps. Figures (E) and (F) show two different views for the thin shell in figure (D). One can also see that the small piece in figure (A) is eliminated.

IV. RESULTS

Our pipeline has been implemented on a 8-core Sun Workstation running Ubuntu 9.04 using the C++ language. The algorithm was tested on the P22, PSV, Brain and Neuron datasets. For space limit, we show the segmentation result of brain and part result of PSV.

V. CONCLUSIONS AND FUTURE WORK

In this paper, a new pipeline for generating quality mesh from volumetric data is proposed. The mesh is composed of many 2-manifold non-intersecting sub-meshes. There is a small gap between adjacent sub-meshes and which may be controlled by a prescribed threshold. An efficient intersection detection and curation technique has also been presented. Finally, the implemented pipeline has been described and numerical experiments performed using the pipeline have been illustrated. We recently noticed that the authors of [9] have proposed another interesting approach to solve a similar problem.

Future work in this area includes improvements such as code speedup and parallelization, adaptive meshing and extension to 3D volumetric meshes.

Acknowledgments

Qin Zhang would like to thank the support from NIH contracts R01-GM074258, R01-EB004873 and a grant from the UT-Portugal COLAB grant and NSFC grant 60773165 and National Key Basic Research Project of China (2004CB318000). Guoliang Xu would like to thank the support from NSFC grant 60773165 and National Key Basic Research Project of China (2004CB318000). Chandrajit Bajaj is grateful for the support from NSF grant CNS-0540033 and NIH contracts R01-GM074258, R01-EB004873 and a grant from the UT-Portugal COLAB grant.

REFERENCES

1. Zhang Y, Hughes TJR, Bajaj CL. An automatic 3D mesh generation method for domains with multiple materials. *Comput. Methods Appl. Mech. Engrg.* 2009; vol. 199(no. 5-):405–415.
2. Wu Z, M SJ Jr. Multiple material marching cubes algorithm. *International Journal for Numerical Methods in Engineering.* 2003; vol. 58:189–207.
3. Lorensen, WE.; Cline, HE. Marching cubes: A high resolution 3D surface construction algorithm. *SIGGRAPH '87: Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*; 1987. p. 163-169.
4. Ju, T.; Losasso, F.; Schaefer, S.; Warren, J. Dual contouring of Hermite data. *SIGGRAPH '02: Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques*; 2002. p. 339-346.
5. Yu Z, Bajaj CL. Automatic ultrastructure segmentation of reconstructed Cryo-EM maps of icosahedral viruses. *IEEE Transactions on Image Processing: Special Issue on Molecular and Cellular Bioimaging.* 2005; vol. 14(no. 9):1324–1337.
6. Xu G, Pan Q, Bajaj CL. Discrete surface modelling using partial differential equations. *Comput. Aided Geom. Des.* 2006; vol. 23(no. 2):125–145. [PubMed: 19830268]
7. Xu G. Discrete Laplace-Beltrami operators and their convergence. *Comput. Aided Geom. Des.* 2004; vol. 21(no. 8):767–784.
8. Bajaj C, Xu G, Zhang Q. Smooth surface construction via a higher-order level-set method. *Institute of Computational Engineering and Sciences, the University of Texas at Austin, Tech. Rep.* 2006 Jun 18.
9. Edwards, J.; Bajaj, C. Topologically correct reconstruction of tortuous contour forests. *Proceedings of the ACM Symposium on Solid and Physical Modeling*; 2010. accepted

Fig I.1. Multi-domain meshing of human brain. Figure (A) shows the original data. Figure (B) shows the multi-domain meshing result. Cut-view and zooming are shown in figure (C).

Fig II.1.
Multi-domain meshing pipeline

Fig III.1.

3D intersection curation. Figure (A) shows the partition mesh after geometric flow smoothing. (B) and (C) show two different views for the thin shell in figure (A). (D) shows the result after point clouds reconstruction by higher-order level-set method and intersection detection and curation. (E) and (F) show two different views for the thin shell in figure (D).