Towards Distributed 3D Adaptive Anisotropic Image-to-mesh Conversion For Biomedical Applications

Kevin Garner, **Fotios Drakopoulos, PhD., and Nikos Chrisochoides, PhD.**
Center for Real-time Computing (CRTC), Department of Computer Science, Old Dominion University, Norfolk, VA 23529

Motivation

Image-to-Mesh (I2M) conversion algorithms are pertinent when using the Finite Element (FE) method to provide a quantitative analysis of patient-specific images. Moreover, the tessellation of complex vascular geometries is required for Computational Fluid Dynamics (CFD) simulations of hemodynamics in cerebral aneurysms. An I2M software pipeline that meets the time and quality requirements of CFD simulations holds significant impact for how medical surgical simulations can improve precision medicine in healthcare. There are two categories for which surgical simulations can be classified: predictive and interactive. Patient medical images, such as MRIs and CTs, are provided as input to both categories of simulations. Predictive simulations are used to predict and optimize the outcome of an intervention by using patient-specific pre-operative image data. Large-scale mesh generation becomes integral in providing high geometric, topologic, and material fidelity while meeting CFD time constraints. Interactive simulations reduce healthcare costs by training surgical residents using virtual environments, allowing them to attain needed skills without imposing risks on real patients. Real-time mesh generation is required in order to meet time constraints imposed by end-to-end computations in near-haptic rates. Fidelity of the anatomic modeling is compromised if the mesh generation algorithms are solely focused on performance. The I2M software must be capable of maximizing element quality while maintaining good performance.

Method

Presented is a new Image-to-Mesh conversion software pipeline that focuses on satisfying both the time and qualitative constraints required by CFD simulations of hemodynamics in cerebral aneurysms. The pipeline is made up of modules that are executed in the following sequence:

1. CBC3D [1] is a sequential I2M method that generates an adaptive multi-tissue isotropic tetrahedral mesh from a multi-labeled segmented image. It initially constructs an adaptive Body-Centered Cubic (BCC) mesh and then deforms the mesh surface to its corresponding physical image boundary in order to improve mesh fidelity and smoothness.
2. In order to achieve parallel mesh adaptation and leverage the concurrency offered by emerging high-performance computing (HPC) architectures, the surface is extracted from the CBC3D-generated mesh and is given as input to a distributed memory mesh generation method. In this distributed method, meshing functionality is separated from performance aspects by utilizing a separate entity for each - CDT3D for mesh generation and PREMA for parallel runtime support. This "separation of concerns" ideology is fundamental to the design of a scalable framework we term the Telescopic Approach [2]. The Telescopic Approach provides a layout of multiple memory hierarchies within an exascale architecture and how different meshing kernels can be utilized at each level to achieve maximum concurrency. CDT3D is a shared-memory code specifically built to operate within the lowest level of the hierarchy, exploiting fine-grain parallelism at the core and node levels. The distributed memory method serves to exploit coarse-grain parallelism at the node level.

One capability of CDT3D is to generate boundary-conforming isotropic tetrahedral meshes with element sizes defined by a point distribution function. The pipeline for isotropic mesh generation can be divided into three steps: initial grid construction, grid refinement, and grid quality improvement [3]. CDT3D also offers metric-based anisotropic mesh adaptation, where the metric can be derived from analytic or discrete fields, and can be combined with Computer-Aided Design (CAD)-based information to accomplish adaptation [4].

Message passing and data migration within the distributed memory method is handled by utilizing the Parallel Runtime Environment for Multicore Applications (PREMA) system, which serves to alleviate the burden of work scheduling and load balancing [5]. This system provides constructs which enable asynchronous message passing between encapsulations of data, work load balancing, and migration capabilities all within a globally addressable namespace.

Given its scalable design and performance in stability, the shared memory CDT3D was abstracted as a library to be used in the refinement of individual subdomains in the distributed memory method (DMCDT3D). A high-level overview of the distributed memory method is shown in Figure 1. It essentially includes six steps, the latter three of which are executed in a loop until the entire mesh satisfies qualitative criteria. These steps include: initial coarse mesh generation, decomposition, interior refinement/adaptation of all subdomains (with frozen subdomain boundaries to maintain conformity), interface shift (movement of subdomain boundary elements from some subdomains into the interior of colored subdomains), interior refinement/adaptation of colored subdomains, and a topology update of subdomain adjacency. Although Figure 1 specifies adaptation for an anisotropic metric field, the method is also applicable to isotropic grid generation for a specified distribution function.

```
DMCDT3D (M_i, m, N, O)
Input:  M_i is the initial mesh
        m is the target metric
        N is the number of subdomains
        O is a list of mesh operations to perform
Output: Adapted mesh that conforms to the anisotropic metric field m
    INITIAL_COARSE_MESH(M_i, m, O)
    Perform data decomposition to create subdomains S_1…S_N
    Distribute subdomains S_1…S_N to processes
    Adapt interior elements of all subdomains
    ADAPT_SUBDOMAIN(S_1…S_N, m, O)
    Adapt low quality interface elements amongst subdomains
    while (S_1…S_N mesh quality is NOT satisfied)
        Color subdomains between those that will send or receive interface data
        S_S1…S_Sn, S_R1…S_Rk := COLOR_SUBDOMAINS(S_1…S_N)
        INTERFACE_SHIFT(S_1…S_N)
        ADAPT_SUBDOMAIN(S_R1…S_Rk, m, O)
        UPDATE_TOPOLOGY(S_1…S_N)
    endwhile
    TERMINATE()
```

Figure 1: High-level Algorithm of Distributed Memory CDT3D Implementation

Results

The quality of the distributed memory method's generated volume meshes are compared to those generated by CBC3D to evaluate if its meshes fall within operational limits of analytical CFD software. Case 1 is an isotropic image of an aneurysm with the following dimensions: image spacing of 1.00 x 1.00 x 1.00 mm^3 and image size of 512 x 512 x 508 voxels^3. This image was obtained from the Neurosurgery Department at Stony Brook University. Case 2 is an anisotropic image of a brain with a tumor with the following dimensions: image spacing of 0.48 x 0.48 x 1.00 mm^3 and image size of 384 x 512 x 176 voxels^3. This case was obtained from Kitware [6]. The distributed memory method was executed with a decomposition of 64 subdomains using 64 cores for refinement across 2 nodes. Data were collected on Old Dominion University's Wahab cluster using dual socket nodes that each featured two Intel Xeon Gold 6148 CPUs @ 2.40 GHz and 384 GB of memory. Figure 2 shows the meshes generated by the distributed method for cases 1 and 2. Figure 3 shows the dihedral angle statistics comparisons between the distributed method's generated meshes and the original meshes generated by CBC3D, for cases 1 and 2. For case 1, the minimum and maximum angles within the distributed method's mesh are 13.24 and 158.75 degrees, respectively. The minimum and maximum angles within CBC3D's mesh are 5.07 and 171.73 degrees, respectively. For case 2, the minimum and maximum angles within the distributed method's mesh are 5.83 and 169.69 degrees, respectively. The minimum and maximum angles within CBC3D's mesh are 8.89 and 166.67 degrees, respectively.

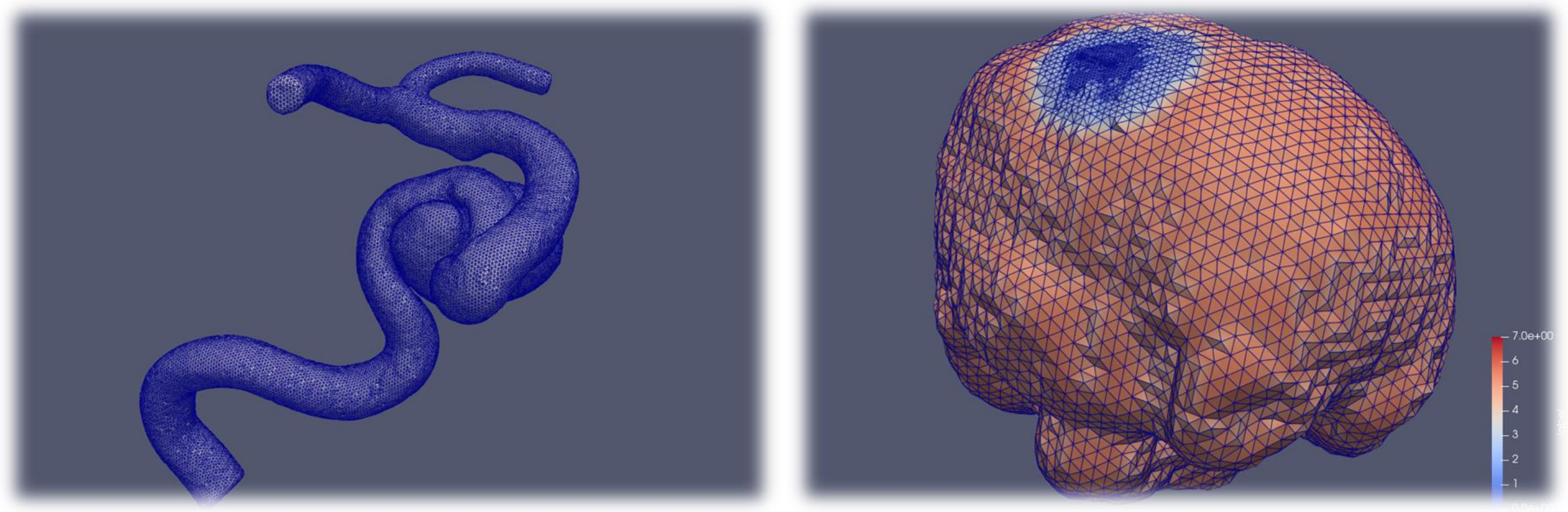

Figure 2: Cases 1 and 2 – aneurysm and brain (with tumor) meshes generated by DMCDT3D

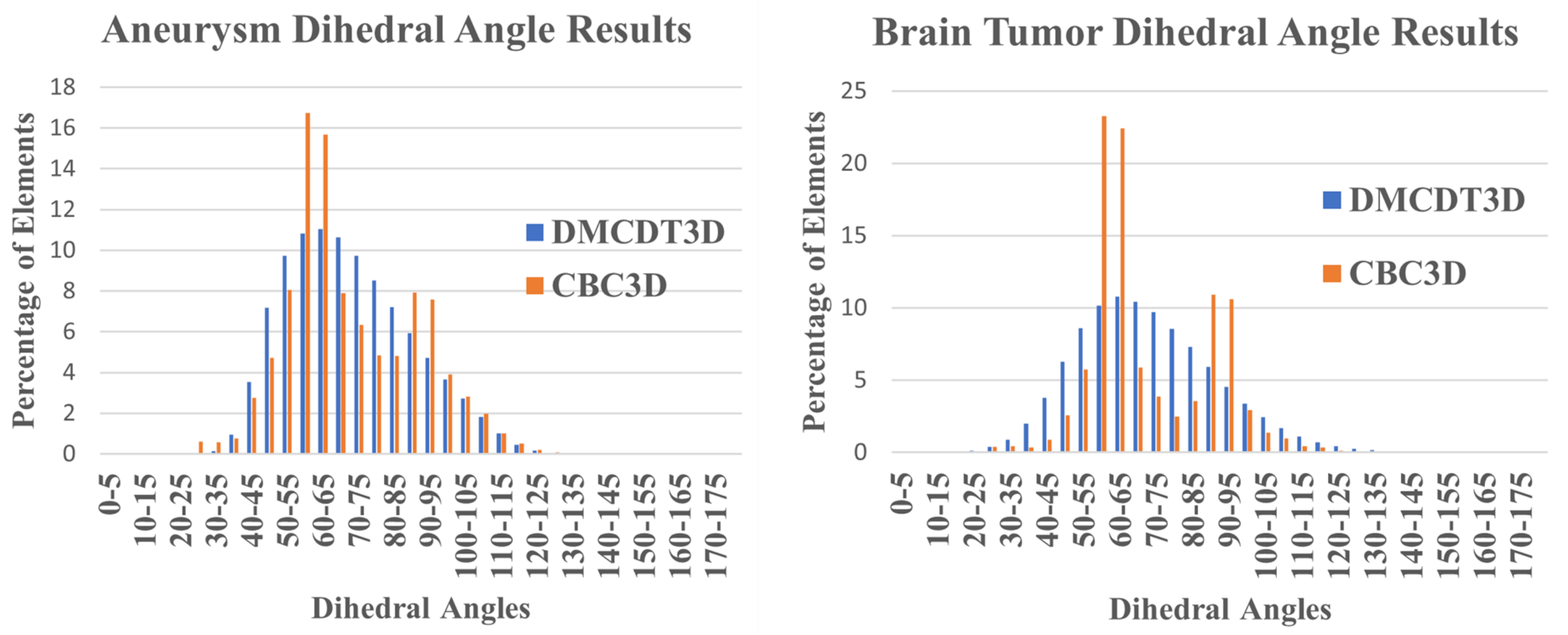

Figure 3: Dihedral Angle Statistics for the Aneurysm (left) and the brain (right) meshes generated by CBC3D and DMCDT3D

Conclusion/Future Work

Presented is a new Image-to-Mesh (I2M) conversion software pipeline that focuses on the tessellation of complex vascular geometries required for Computational Fluid Dynamics (CFD) simulations of hemodynamics in cerebral aneurysms. The pipeline utilizes a sequential I2M method called CBC3D in order to extract a high quality smooth surface which is given as input to a distributed memory method designed to target large-scale mesh generation. Preliminary results show that the distributed memory method generates meshes that are qualitatively comparable to those generated by CBC3D and that fall within the operational limits of analytical CFD software. This project's completion is expected to yield an adaptive anisotropic parallel mesh generation pipeline that will satisfy both time and quality constraints required by CFD simulations. The meshes generated in this work are isotropic, where the distribution function for the distributed method meshes was calculated based on the surface meshes provided as input by CBC3D. Figure 4 shows an anisotropic mesh of an aerospace case generated by CDT3D where the mesh was adapted to conform to a specific metric tensor field. Future work includes the use of anisotropic metric tensors to accomplish adaptation of blood flow cases. In this work, the distributed memory method only executes with one iteration of the interface shift and interior adaptation of colored subdomains. The topology update of subdomain adjacency will also be completed to allow for multiple iterations of subdomain boundary element movement/adaptation, improving overall output mesh quality. Currently, a centralized master/work model is used for communication within the distributed memory method. A decentralized model will be implemented in the future as well, improving potential scalability.

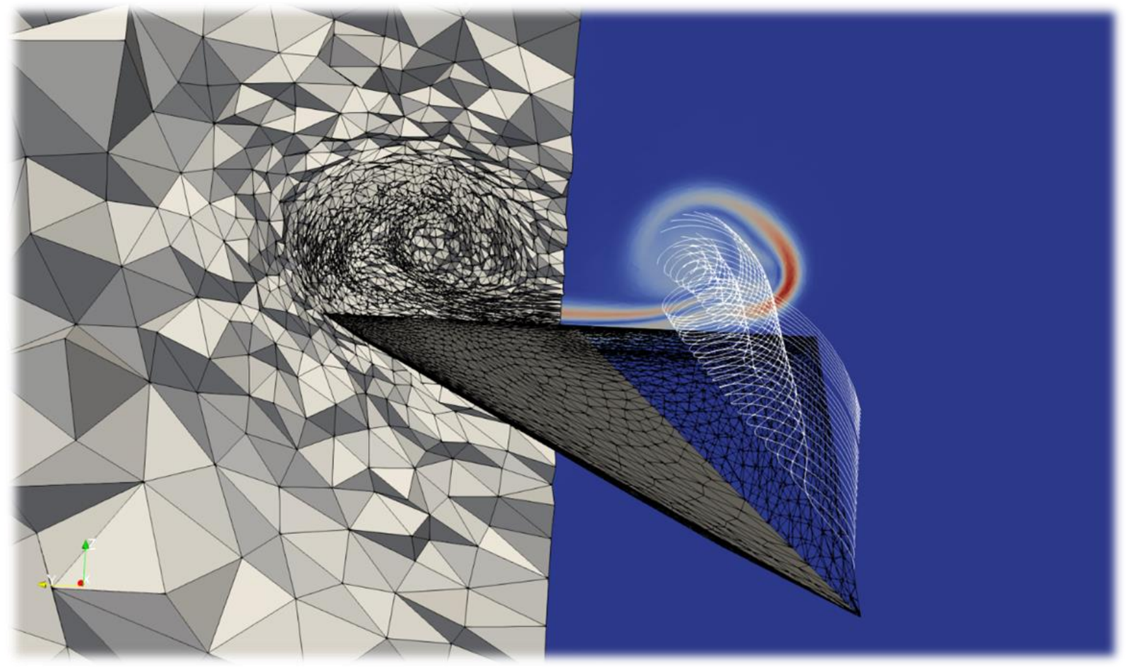

Figure 4: Anisotropic metric-adapted mesh to a laminar flow over a delta wing, generated by CDT3D [courtesy of Christos Tsolakis]

References

[1] Drakopoulos, F., Ortiz, R., Enquobahrie, A., Sasaki-Adams, D., & Chrisochoides, N. Tetrahedral image-to-mesh conversion software for anatomic modeling of arteriovenous malformations, Procedia Engineering, vol. 124, 2015, pp. 278-290. doi: 10.1016/j.proeng.2015.10.139

[2] Chrisochoides, N., "Telescopic Approach for Extreme-Scale Parallel Mesh Generation for CFD Applications," 46th AIAA Fluid Dynamics Conference, Washington D.C., USA, 2016. https://doi.org/10.2514/6.2016-3181, AIAA 2016-3181.

[3] Drakopoulos, F., Tsolakis, C., and Chrisochoides, N., "Fine-grained Speculative Topological Transformation Scheme for Local Reconnection Methods," AIAA Journal, Vol. 57, 2019, pp. 4007–4018. https://doi.org/10.2514/1.J057657.

[4] Tsolakis, C., Chrisochoides, N., Park, M. A., Loseille, A., and Michal, T., "Parallel Anisotropic Unstructured Grid Adaptation," AIAA Journal, Vol. 59, 2021, pp. 4764–4776. https://doi.org/10.2514/1.J060270

[5] Thomadakis, P., Chrisochoides, N. Toward runtime support for unstructured and dynamic exascale-era applications. J Supercomput 79, 9245–9272 (2023). https://doi.org/10.1007/s11227-022-05023-z

[6] Kitware Inc., https://www.kitware.com (accessed Jul. 18, 2023).

Acknowledgments

We would like to thank Christos Tsolakis and Polykarpos Thomadakis for their previous work with CDT3D and PREMA, respectively, upon which this work now builds. Research reported in this publication was supported in part by the Richard T. Cheng Endowment and the National Institute of General Medical Sciences of the National Institutes of Health under Award Number 1T32GM140911-03. The content is solely the authors' responsibility and does not necessarily represent the official views of the National Institutes of Health.

G-RISE Program is funded by NIH grant number: 1T32GM140911