A Dual-Mesh Monte Carlo Algorithm Using a Coarse Tetrahedral Mesh and Voxel Output

Shijie Yan[1], **Anh Phong Tran**[2], **Qianqian Fang**[3,1*]

[1]*Dept. of Electrical and Computer Engineering,* [2]*Dept. of Chemical Engineering, and* [3]*Dept. of Bioengineering, Northeastern University, 360 Huntington Ave, Boston, MA 02115*

**q.fang@neu.edu*

Abstract: We report a new technique to accelerate mesh-based Monte Carlo (MMC) photon transport simulations by using a coarsely tessellated tetrahedral mesh to represent tissue boundaries and a voxelated space to store output fluence.

OCIS codes: (170.3660) Light propagation in tissues; (170.5280) Photon migration; (170.7050) Turbid media.

1. Introduction

The Monte Carlo method (MC) offers accurate solutions to the radiative transport equation (RTE) for an extended range of media properties, and has been widely recognized as the gold-standard for biophotonics modeling in human tissues [1]. Over the the past decade, active research in MC simulations has led to dramatically improved simulation speed [2] and enhanced ability to model complex domains. The first reported MC algorithm for photon transport, MCML [1], was designed to handle only multi-layered media. This algorithm was later generalized to handle arbitrarily heterogeneous domain by using a voxelated space [2,3]. Since then, a number of memory efficient MC algorithms have been proposed, including the (triangular) surface-based MC [4] and (tetrahedral) mesh-based MC [5, 6].

Both the surface and mesh-based MCs can efficiently represent curved boundaries without uniform grid refinement, however, surface-based MC requires extensive ray-triangle intersection testing for every photon movement. This challenge was resolved in MMC because the tetrahedral element enclosing the photon is always known - as a result, only 2.5 ray-triangle intersection tests per photon movement are needed on average.

In both reported voxel- and mesh-based MCs, a single grid is used for the purposes of 1) facilitating ray-tracing computation and 2) storing of the output quantities. One should recognize that photon trajectories are independent of the volume discretization strategy. In other words, given the same random number sequence, a photon should follow the same path whether a voxelated grid, a surface model or a tetrahedral mesh is used for discretization.

With the above observation, we propose a dual-mesh MC (DMMC) algorithm to utilize a tetrahedral mesh for ray-tracing calculations and shape representation, and use an independent voxelated grid for output quantity storage. Because the density of the tetrahedral mesh is no longer restricted by the desired resolution of the output, a coarse tessellation of the sub-surface nodes is sufficient, leading to significantly improved computational speed.

2. Methods

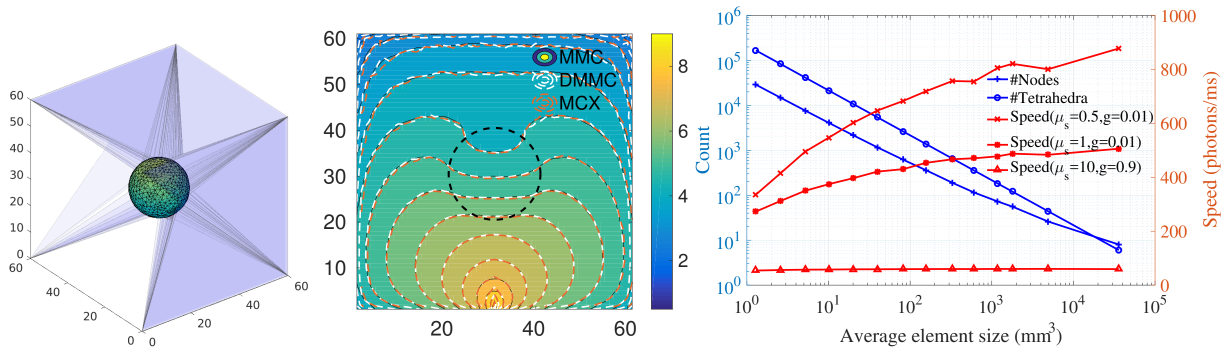
Similar to conventional MMC, we first create the triangular surface meshes to accurately represent tissue boundaries. These surfaces can be produced using the ε-sampling algorithm [7] or tessellation of a piece-wise-linear complex [8]. A key step in DMMC preprocessing is to use TetGen to create a coarsely tessellated mesh combining all input surfaces without adding Steiner nodes [8]. The cross-cut view of a sample coarse mesh is shown in Fig. 1(a).

In DMMC, the mesh in which the fluence is accumulated is implicitly defined by a uniform grid, similarly to the voxel-based MC. For every photon movement, the photon energy loss is stored within the voxels that the photon traverses. Because the photon paths are delineated at the tetrahedral/surface boundaries, the tissue boundaries remain honored in DMMC. The combination of storage in a grid and decreased mesh density for ray-tracing allows for significant speed improvements without a reduction in accuracy.

3. Results

We first validate the DMMC algorithm by comparing with MMC with a single fine mesh. A heterogeneous domain with a 10 mm radius sphere ($\mu_a = 0.002\,\text{mm}^{-1}$, $\mu_s = 1\,\text{mm}^{-1}$, $g = 0.01$ and $n = 1.37$) centered at a $60 \times 60 \times 60$ cubic domain ($\mu_a = 0.05\,\text{mm}^{-1}$, $\mu_s = 5\,\text{mm}^{-1}$, $g = 0.9$ and $n = 1.37$). In the case of MMC, a dense mesh (64,042 nodes, 376,990 elements), similar to [6] Fig. 3a, is used. For DMMC, a coarsely tessellated mesh (801 nodes, 4,064 elements), shown in Fig. 1(a), is used, containing the same sphere surface nodes (N=793) as the MMC mesh. DMMC records the fluence in a uniform $61 \times 61 \times 61$ grid with 1 mm resolution. the contour plots of the fluence along the source plane ($y = 30$) are compared in Fig. 1(b).

Next, we quantify the speed improvement by comparing DMMC using different tetrahedral mesh densities. A uniform cubic domain ($\mu_a = 0.005\,\text{mm}^{-1}$, $\mu_s = 1\,\text{mm}^{-1}$, $g = 0.01$ and $n = 1$) is simulated with 10^7 photons using an Intel



(a) (b) (c)

Fig. 1. (a) A sample coarse mesh for ray-tracing in dual-mesh MC (DMMC), (b) comparison of DMMC with MMC and voxel-based MC (MCX), and (c) speed vs mesh density for DMMC.

i7-7700 CPU. A total of 14 meshes are created, with the finest mesh containing 28,946 nodes and 167,438 tetrahedra and the coarsest mesh with only 8 nodes and 6 tetrahedra. DMMC simulations are performed and the simulation speed is calculated. The node/elem counts and the simulation speeds (in photons/ms) are plotted in Fig. 1(c) against the average element size. We repeat the above tests for two other scattering settings: 1) $\mu_s = 0.5\,\text{mm}^{-1}/g = 0.01$ and 2) $\mu_s = 10\,\text{mm}^{-1}/g = 0.9$. All simulations accumulate fluence in a $61 \times 61 \times 61$ grid with 1 mm^3 voxels.

4. Conclusion

From the plots in Fig. 1(b), although using only a fraction of nodes and elements, DMMC produced a solution that matches excellently with that of MMC. From Fig. 1(c), it is clear that DMMC gains more computational speed from using a coarse mesh for ray-tracing calculation. A nearly 2x speed improvement was observed for $\mu_s = 1\,\text{mm}^{-1}/g = 0.01$; this speed enhancement is improved to nearly 3x when reducing μ_s by half. Nonetheless, when the scattering coefficient is high, such improvement diminishes. This is because when the mean-free-path of the photon is significantly smaller than the average size of the tetrahedral elements, the number of ray-tetrahedral intersection tests is primarily due to the scattering events instead of the density of the tetrahedral mesh. DMMC can bring significant speedup when the mean-free-path of photons is larger than the feature sizes of the tissue structures.

In summary, we described an efficient MMC algorithm by using separate meshes for ray-tracing and output storage. Our DMMC code can be downloaded from https://github.com/fangq/mmc/tree/dualmesh.

5. Acknowledgement

This research is supported by the National Institutes of Health (NIH) grants# R01-GM114365 and R01-CA204443.

References

1. L. Wang, et al., "MCML-Monte Carlo modeling of light transport in multilayered tissues," Comput. Meth. Prog. Bio. **47(2)**, 131-146 (1995).
2. Q. Fang. and D. A. Boas, "Monte Carlo Simulation of Photon Migration in 3D Turbid Media Accelerated by Graphics Processing Units," Opt. Express **17(22)**, 20178-20190 (2009).
3. D. Boas, et al., "Three dimensional Monte Carlo code for photon migration through complex heterogeneous media including the adult human head," Opt. Express **10(3)**, 159-170 (2002).
4. E. Margallo-Balbs and P. J. French, "Shape based Monte Carlo code for light transport in complex heterogeneous tissues," Opt. Express **15**, 14086-14098 (2007).
5. H. Shen, G. Wang, "A tetrahedron-based inhomogeneous Monte Carlo optical simulator," Phys. Med. Biol. **55**, 947-962 (2010).
6. Q. Fang, "Mesh-based Monte Carlo method using fast ray-tracing in Plücker coordinates," Biomed. Opt. Express **1(1)**, 165-175 (2010).
7. L. Rineau and M. Yvinec. "3D Surface Mesh Generation," in CGAL Editorial Board, editor, *CGAL User and Reference Manual.* 3.4 edition (2008)
8. H. Si, "TetGen, a Delaunay-Based Quality Tetrahedral Mesh Generator," AMC Trans. Math. Software, **41(2)**, 11 (2015).