Octree-based Implementation of the Monte Carlo Algorithm for Solving Three-Dimensional Radiative Transport Problems on an Unstructured Mesh

Tim Bashford, Kelvin E Donne

School of Applied Computing
UWTSD
Swansea, UK.
tim.bashford@uwtsd.ac.uk, kelvin.donne@uwtsd.ac.uk

***Abstract* - Computational modelling of light-tissue interaction problems typically requires the determination of the volumetric deposition of incident low-energy photons and then subsequently incorporating this distribution in a time-dependent thermal calculation. In a clinical context the starting point is often an arbitrary 3D geometry approximated by a tetrahedral mesh. This paper considers the challenges of achieving a computationally efficient approach to solving the initial radiative problem.**

Keywords - tissue optics, Monte Carlo, BEM, parallelisation

I. INTRODUCTION

Clinical procedures sometimes require tissue to be illuminated by low energy photons from a laser or an intense pulse light flash-lamp [1]. The absorption of low energy photons can subsequently lead to photo-chemical, photo-thermal and photomechanical effects with increasing light intensity. The wavelength and the exposure time are also significant in determining the tissue response. In the paper we will consider tissue to be a turbid medium where the only interactions are absorption or scattering. These processes are characterized by μa and μs, which are the reciprocals of the mean free paths for absorption and scattering. Photons in tissue are predominantly scattered anisotropically, with strong forward bias. This is usually represented by the Henyey-Greenstein phase function [2], p(θ) and illustrated in Figure 1:

$$p(\theta) = \frac{1 - g^2}{(1 + g^2 - 2g\cos\theta)^{3/2}} \quad (1)$$

where g is the anisotropy factor, typically 0.8 in tissue.

II. LITERATURE REVIEW

Determination of the volumetric distribution of photons requires a solution to the Radiative Transport Equation (RTE) [4,5]. Although deterministic solution methods and approximations such as diffusion theory exist, the Monte Carlo numerical method has gained traction as the method of choice for tissue-optics modelling. Through a probabilistic approach, the Monte Carlo method models multiple instances of the same measurable unit to obtain a numerical result. By taking multiple instances of a photon's trajectory through tissue, it is possible to create a computational model to simulate the absorption and scattering events for a large number of incident photons. This method will then produce a photon distribution throughout the tissue equivalent to solving the deterministic Boltzmann Radiative Transfer Equation.

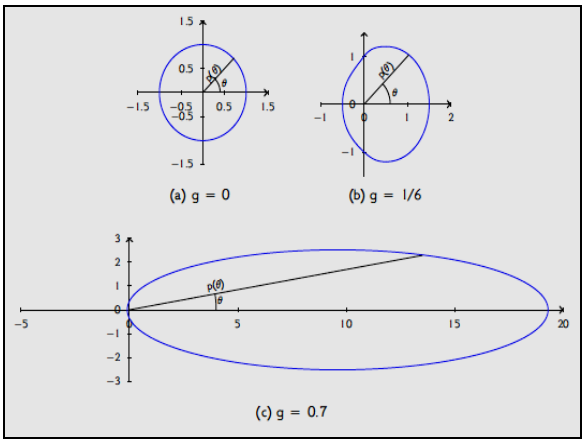

Figure 1: Henyey–Greenstein phase function for varying values of anisotropy factor g. [3]

Due to the probabilistic nature of the model, an inevitable relationship is formed between the simulated particle count and the accuracy of the result. The extent to which this is true varies significantly by the degree to which random number generation features in the model; a simulation which is only nominally impacted by random elements requires a smaller particle count, while for a simulation on which random numbers make a significant impact the opposite is true.
The Monte Carlo method is well suited to the simulation of laser light interacting with tissue by modeling photon propagation for probability density; multiple instances of a photon following a random walk will, in sufficient numbers, result in a numerically valid model of photon deposition as

an alternative to solving the radiative transport equation. While other methods such as diffusion theory can be faster and easier to implement than the Monte Carlo method, they inevitably lose accuracy significantly for conditions where fluence rate or gradient are non-linear. A typical photon trajectory is illustrated in Figure 2 [3,4,5], where the pathlength between events is determined by a random number RND:

$$pathlength = -\frac{\log(RND)}{\mu t} \quad (2)$$

where 0<RND<1 and μt= μa + μs.

The scattering angle for the 'spin' part of each event is similarly determined from the Henyey-Greenstein phase function.

At each interaction event a proportion of the photon's energy μa/μt is absorbed and the scattered photon travels on with reduced energy to the next event, until either the photon's energy reduces below a prescribed value or the photon exits the domain.

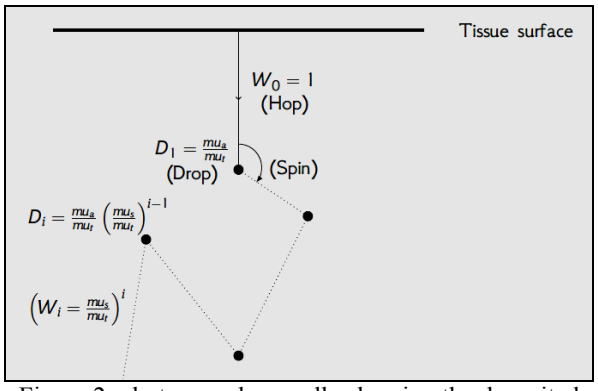

Figure 2: photon random walk, showing the deposited energy, Di, and the propagated energy, Wi, at ith event

A. *Challenges for Implementing Monte-Carlo Algorithm for Real-World Problems*

Creation of the modified Monte Carlo method required consideration of a number of computational and physical problems. Literature [5] pertaining to three-dimensional Monte Carlo photon transport problems have focused almost exclusively on regular geometry. While regular geometry (ubiquitously collinear along the Cartesian axes) is sufficient for many problems, there is an inherent inaccuracy introduced between model and physical processes; simply put, physical structures are not constructed from and cannot be modelled with cubes.

The Monte Carlo method is inevitably reliant upon volume geometry to both dictate homogenous scattering and attenuation coefficients and to act as a storage boundary for photon deposition. It is, therefore, imperative to select an alternative approach to volume geometry for an implementation of the Monte Carlo method which hopes to accurately model human tissue.

Ideal for the modelling of physical problem is an unstructured mesh. There are two geometries often utilised in an unstructured mesh; tetrahedrons and hexahedrons. While there is a great deal of literature comparing the two, the relative merits of each pertain to meshes where the structure significantly modifies the output, such as in the finite element method or computational fluid dynamics modelling where hexahedral meshes are generally considered superior [6,7] in part because of the higher strain deformation requirements of tetrahedral meshes.

Literature recommendations pertaining to selection of tetrahedral or hexahedral meshes are, however, immaterial when considering geometry for Monte Carlo method which does not utilise the geometry beyond selecting a bounding deposition area. There were two primary factors for selection of a mesh type to be utilised with the Monte Carlo method: ease of use and computational cost.

The primary consideration as pertains to ease of use is how accessible each mesh type is for an end user. Ideally, a user would be able to import geometry in the form of a medical imaging file obtained through a real-world scan, at which point the underlying method would be irrelevant.

Access to such scans, however, is not a guarantee, indeed there can often be significant difficulty obtaining real-world scans to match problems to be modelled, ultimately requiring the creation or modification of meshes in Computer Aided Design (CAD) packages.

A compiled list of mesh generators by Schneiders [8] asserts that of 148 free and commercial mesh generators, 67 are capable of tetrahedral mesh generation while a mere 13 include hexahedral mesh generation functionality, meaning that from this perspective, a tetrahedral mesh is far more practical for an end user.

A further consideration is the computational efficiency of eliciting the location of a simulated photon based on a relative position in Cartesian space. The Monte Carlo method provides only the location of an exact point for a photon, which during the "drop" step must be mapped to the volume geometry. This is a trivial process for regular meshes collinear with each axis, solved by Pythagoras' theorem and a knowledge of the width of each volume element for each homogenous region. The implication of this is that calculation of a point in an arbitrary geometry mesh requires a point-in-geometry test to locate the container object for deposition.

Assuming comparison of a mesh containing a specific number of tetrahedrons with another containing an identical number of hexahedrons, computational efficiency resolves to a single point in geometry test for tetrahedrons against hexahedrons. The usual method for point in geometry testing for a hexahedron is subdivision of a single hexahedron into four tetrahedrons then performing point in geometry testing for these.

As such, inevitably, a hexahedron is four times more computationally expensive to test for no benefit to accuracy; utilising a tetrahedral mesh with four times the number of elements is equally efficient, but provides a significantly more accurate output. With tetrahedral meshes appearing more practical in the majority of circumstances, and more computationally efficient for processing, a tetrahedral mesh approach was selected.

A literature review of published work on Monte Carlo modelling of photon transport in turbid media indicated an opportunity to develop a novel approach to photon tracking in a turbid medium on an unstructured mesh. A review of computational tissue optics by Zhu & Liu [9] summarised recent developments in 3D MC and acceleration methods including parallelisation techniques. The well-established MCML code by Wang et al [10] is voxel based and therefore has limitations in modelling complex curvilinear geometries. Models based on unstructured meshes have been developed by Fang [11,12] and Shen & Wang [13] use optimised ray-tracing techniques often based on Plücker coordinates. To the authors' knowledge there are no published 3D Monte Carlo light-tissue models based on unstructured meshes that implement domain decomposition. This approach might be advantageous for very large meshes. This paper describes the implementation of a parallelised Octree algorithm which has been tested for geometries represented by $>10^6$ tetrahedral elements.

III. PROPOSED OCTREE-BASED MONTE CARLO METHOD

The classic three-dimensional Monte Carlo implementation remains largely applicable to the version created for arbitrary geometry simulation. Notably, each photon remains at a single point in Cartesian space, with little initial impact on the "hop" and "spin" steps (until a boundary is reached, as described later), however, the "drop" step requires a lookup of which tetrahedron is to be found at the point at which the photon currently rests. Unlike the solution for regular geometry, there is no trivial method of eliciting in which tetrahedron to deposit a given fraction of photon weight.

The resolution selected was one of trivial elimination through domain decomposition; performing computationally cheap testing which would immediately discount the vast majority of tetrahedrons, removing the need for random, expensive testing.

An octree is a recursive tree structure [14] in which each node has eight 'child' nodes (Figure 4). Within computer graphics, each of these eight nodes is usually axis aligned to the scene, and subdivided into equal parts by dimension. This is applied recursively, meaning that the head node will contain the entire graphical scene, the first level down will contain one eighth per node, the second level down will contain one sixty-fourth, continuing recursively as required.

In many cases, geometry is discretized to Cartesian geometry in this final structure, to improve computational efficiency. It is important to note that when the subdivision of a scene occurs, that subdivision is always co-linear to each axis and always subdivides the parent oct exactly in half. The implication of this is that traversing from a parent oct to a child oct only requires three floating point comparisons; one for each axis for the point being tested. This means that, for an evenly distributed scene, 7/8ths of the scene can be trivially and recursively excluded until few enough objects remain that computationally expensive point-in-geometry comparison is acceptable.

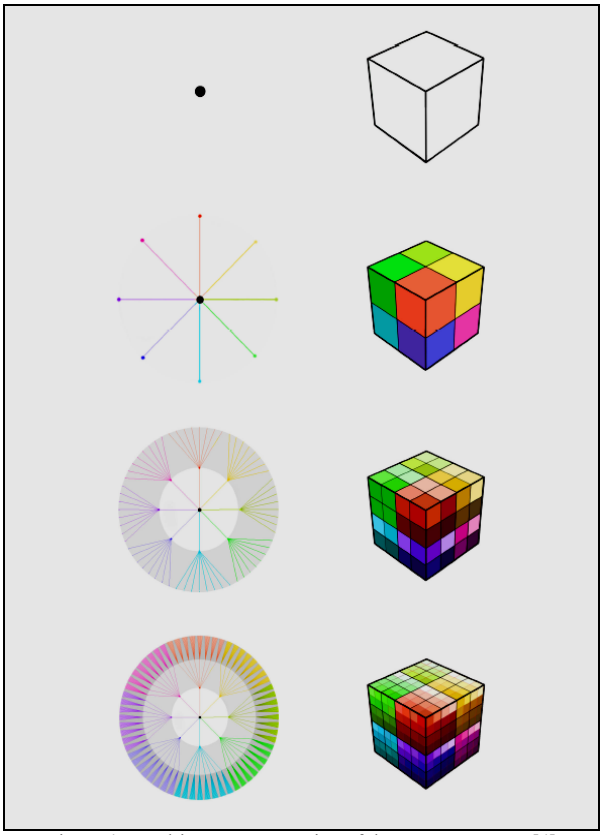

Figure 4: Dendrimer representation of the octree structure [4]

The octree structure ultimately significantly mitigates the computational issues presented by conversion of the photon transport Monte Carlo method from regular to true arbitrary geometry. First and foremost, it permits a comparatively computationally cheap method for mapping Cartesian geometry (on which the method operates) to the arbitrary geometry of the mesh.

In an attempt to reduce the sequential processing of the model, thus reducing the portion which falls under Amdahl's law, a parallel octree construction algorithm was created specific to this problem. While a true linear speed increase relative to parallel processing power is not achievable, it is possible to achieve a degree of performance increase which is extremely close.

Disregarding the mesh as a whole, a given node in an octree can act as the head of the tree for those nodes

beneath it, i.e. each branch node it is an entirely self-containing hierarchical structure. A node one level down from the head will always encompass one eighth of the mesh, while one further level down will always encompass one sixty-fourth, increasing in powers of eight.

An approach of trivial mesh subdivision was therefore taken, calculating a "working level" of the tree to ensure that each parallel process is assigned at least one node. The operational level is calculated by dividing the cpu core count by 8 until the value is at most 1.0, with the number of subdivisions being the level. If, for example, there are 60 cores available, the tree will be subdivided at the second layer (64 nodes), while if there are 65 then it will be subdivided at the third (512 nodes). Cores are then evenly assigned to each subtree, with each core performing point-in-geometry tests on each point (which delimit the geometry) in every tetrahedron, to assess which geometric objects fall into their subtree.

For example, if processing on 16 cores, the algorithm will begin at the second level of the tree structure, effectively building the first two levels of the tree, then decomposing the octree into its 64 constituent subtrees. Each of these subtrees is allocated to the available cores, in this example such that each core is allocated 4 subtrees. If the number of cores utilized were not divisible by 8, some cores will be assigned one more subtree than others.

Ultimately, all subtrees are allocated to a core. Each core may then inspect the list of tetrahedrons, and perform computationally cheap tests to ascertain whether that geometry should fall within the assigned subtree(s). From this point, the build process continues recursively, as it would if processed in serial, with geometry added to each subtree as a constituent of the octree as a whole. Due to the structure of an octree, this process can be performed, requiring no additional memory over an octree constructed in serial.

This technique, however, has two potential inadequacies. Firstly, assuming a mesh is of similar granularity throughout (an assumption which certainly cannot be made, but will be addressed later), if the hardware performing the calculations does not provision a number of cores which is a power of eight, inevitably one or more cores will be required to undertake more processing than others, leaving cores idle.

Take, for example, a case of a 63 core cluster; this will operate at level 3, with each of the 63 cores processing one of the 64 subtrees. Once complete, however, one subtree remains which can only be processed on a single core, vastly increasing the serial portion of the program. That is to say that the technique, as described so far will not see a benefit beyond the number of nodes at the level of operation divided by two, i.e. at level 3, there is no calculable difference between 32 and 63 cores in terms of time taken; only an increase to 64 will see a time improvement.

Furthermore, meshes may vary significantly in granularity. A mesh may be very fine in one area, yet very coarse in another; the fine mesh will inevitably take considerably longer than the coarse mesh to process, expecting an almost linear increase for the number of tetrahedrons in a given octet.

Through experimentation it was discovered that pre-calculation of time taken in itself was time consuming enough not to be worthwhile. Instead, an approach was taken whereby processing is assigned trivially as detailed above until a core has completed processing. This core, rather than sitting idle until the program completes will interrogate other cores to check the processing tasks they have remaining, and especially whether the computational load of sharing the remaining tasks is worthwhile; if there are only a few seconds left, the time taken to swap data in memory, or even over a network may entirely negate the benefits of parallel processing, or even prolong processing time.

If another core has work which will take more than a few seconds to complete, that work will be subdivided and shared; the parent octet will be reassigned a level down, with the eight component octets that result shared among the two available cores. When more cores become free, they are then capable of taking control of processing one octet each until every core is processing, or a further subdivision is needed. Through this process, the recursive algorithm is capable of keeping as many cores as a system has processing until the octree is entirely constructed. This also removes any issues of granularity differences in the mesh, as significantly uneven initial distribution will be automatically rebalanced as cores complete processing of more coarse subtrees.

IV. RESULTS

A. Timings

In order to assess whether the theoretical speed benefits offered by the mapped octree Monte Carlo method were significant, or even worthwhile taking into account the front-loaded computation required, a small case study was conducted across a variety of meshes for comparison.

Table I summarizes each mesh and Table II shows the comparison between the implemented Octree Monte Carlo method and a simple linear look-up approach.

TABLE I. SEVEN DIFFERENT MESH DENSITIES FOR TESTING OCTREE METHOD

Mesh	Surface Elements	Volume(tet) elements
1	390	1881
2	944	2,724
3	1,374	1,940
4	4,800	21,540
5	1,920	138,793
6	1200,00	1,285,036
7	1,371,90	1,926,250

TABLE II. TIMING COMPARISONS BETWEEN OCTREE AND NON-OCTREE MONTE CARLO CALCULATIONS

Input		Octree (minutes)			Non-octree (minutes)
Mesh	Photon Count	Build	MC	Total	MC
1	1e05	0.1	0.04	0.14	0.1
3	1e05	0.2	0.05	0.25	0.2
6	1e05	6.0	0.4	6.4	44.2
7	1e05	8.2	0.8	9.0	899.1
1	1e07	0.1	0.3	0.4	7.8
2	1e07	0.2	0.4	0.6	11.3
3	1e07	0.2	0.3	0.5	8.9
4	1e07	1.0	0.8	1.8	95.6
5	1e07	3.1	1.3	4.4	244.0
6	1e07	6.0	5.5	11.5	2097.4
7	1e07	8.2	8.7	16.9	3811.4

It is apparent from the timing results in Table 2 that the octree-enabled technique outperforms the standard linear lookup technique for all but the smallest meshes, i.e. meshes 1 and 3 for photon count 1e5.

This marginal performance benefit is gained because of the front-loaded octree construction time; in run time, the octree method performs equal to or better, even for small meshes and photon counts.

B. Benchmarking

Figure 4 summarizes the benchmarking tests for the 3D Octree Monte Carlo method for a 1 cm cube, where the Monte Carlo method is checked against the analytical solution for the diffusion approximation to the Radiative Transport Equation given by Morse and Feshbach [15]. The graphs show photon absorption against depth along the central axis of the cube.

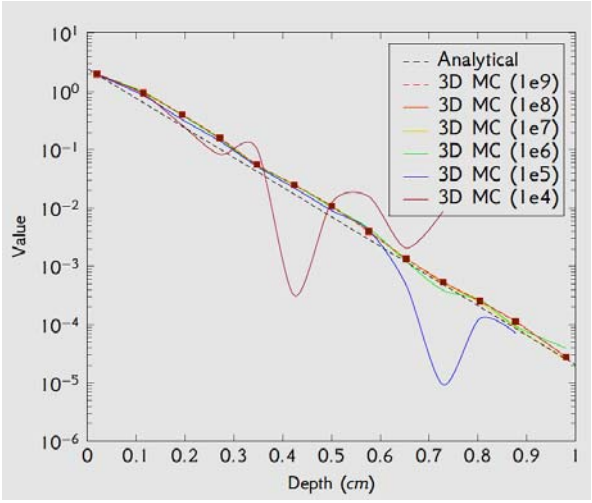

Figure 4: 3D Monte Carlo runs for increasing photon count

V. CONCLUSIONS

The 3D Monte Carlo model has been verified and, with 1e9 photons is considered acceptable for modelling clinical light-tissue problems. The analytical solution solves the diffusion approximation to the full radiative transport equation and therefore a precise comparison with the 3D Monte Carlo, especially close to the incident surface should not be expected. However, the results show very good behavior and accuracy, with the parallelized Octree implementation facilitating very good calculation times for large, complex 3D tetrahedral meshes.

REFERENCES

[1] M. J. C. Van Gemert, Steven L. Jacques, H. J. C. M. Sterenborg, and W.M.Star. "Skin Optics." Biomedical Engineering, IEEE Transactions on, 36(12):1146–1154, Dec 1989. ISSN 0018-9294. doi: 10.1109/10.42108

[2] L. G. Henyey and J. L Greenstein. "Diffuse radiation in the galaxy." The Astrophysical Journal, 93:70–83, 1941.

[3] G.M. Daniel. "Optical and thermal transport modelling of laser-tissue interaction." PhD thesis, UWTSD, March 2002.

[4] T. Bashford, "Parallelisation techniques for the numerical solution of radiative and thermal transport in tissue photonics", PhD Thesis, UWTSD, January 2017

[5] A. J. Welch & M. J. C. van Gemert (eds), "Optical-Thermal Response of Laser-Irradiated Tissue" 2nd edn, Springer, 2011

[6] X.Bourdin, X. Trosseille, P. Petit, and P. Beillas. "Comparison of tetrahedral and hexahedral meshes for organ finite element modeling:an application to kidney impact." Proceedings of the 20th International Technical Conference on the Enhanced Safety of Vehicles, 2007.M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[7] S. E. Benzley, E. Perry, K.Merkley, B. Clark, and G. Sjaardema. " comparison of all hexagonal and all tetrahedral finite element meshes for elastic and elasto-plastic analysis. "Proceedings, 4th International Meshing Roundtable, pages 179–191, 1995.

[8] R. Schneiders,"Mesh Generation and Grid Generation on the Web", www.robertschneiders.de/meshgeneration/meshgeneration.html . Accessed on March 9th 2019

[9] C. Zhu and Q. Liu, "Review of Monte Carlo modeling of light transport in tissues", Journal of Biomedical Optics,18(5):1-8, 2013

[10] L. Wang, S.L. Jacques and L. Zheng, "MCML- Monte Carlo modeling of light transport in multi-layered tissues", Computer Methods and Programs in Biomedicine, 47:131-146 (1995)

[11] Q. Fang,"Mesh-based Monte Carlo method using fast ray-tracing in Plücker coordinates", Biomedical Optics Express, 1(1):165-175 (2010)

[12] R. Yao,1 X. Intes,1 and Q. Fang, "Generalized mesh-based Monte Carlo for widefield illumination and detection via mesh retessellation", Biomedical Optics Express, 7(1):171-184 (2016)

[13] H. Shen and G. Wang,"A study of tetrahedron-based inhomogeneous Monte Carlo optical simulation", Biomedical Optics Express 2(1):44-57 (2011)

[14] K. Ogawa and S. Maeda, "A Monte Carlo method using octree structure in photon and electron transport", Nuclear Science, IEEE Transactions on, 42(6):2322–2326, 1995. ISSN 0018-9499. doi: 10.1109/23.489435.

[15] P. M. Morse and H. Feshbach. Methods of Theoretical Physics, Parts 1 and 2. McGraw-Hill, 2005. ISBN 9780976202103.