Using graphics processing units to accelerate perturbation Monte Carlo simulation in a turbid medium

Fuhong Cai
Sailing He

Using graphics processing units to accelerate perturbation Monte Carlo simulation in a turbid medium

Fuhong Cai[a] and Sailing He[a,b]
[a]Zhejiang University, Centre for Optical and Electromagnetic Research, Zhejiang Key Laboratory of Sensing Technologies, Hangzhou 310058, China
[b]Royal Institute of Technology, Department of Electromagnetic Engineering, 100 44 Stockholm, Sweden

Abstract. We report a fast perturbation Monte Carlo (PMC) algorithm accelerated by graphics processing units (GPU). The two-step PMC simulation [*Opt. Lett.* **36**, 2095 (2011)] is performed by storing the seeds instead of the photon's trajectory, and thus the requirement in computer random-access memory (RAM) becomes minimal. The two-step PMC is extremely suitable for implementation onto GPU. In a standard simulation of spatially-resolved photon migration in the turbid media, the acceleration ratio between using GPU and using conventional CPU is about 1000. Furthermore, since in the two-step PMC algorithm one records the effective seeds, which is associated to the photon that reaches a region of interest in this letter, and then re-run the MC simulation based on the recorded effective seeds, radiative transfer equation (RTE) can be solved by two-step PMC not only with an arbitrary change in the absorption coefficient, but also with large change in the scattering coefficient. © *2012 Society of Photo-Optical Instrumentation Engineers (SPIE).* [DOI: 10.1117/1.JBO.17.4.040502]

The Monte Carlo (MC) method is a standard simulation method for solving radiative transfer equation (RTE).[1] Traditionally, MC simulation is time-consuming and can only treat one set of optical properties at a time. For White Monte Carlo (WMC) method,[2] which is developed from the MC method, one pre-performs a single MC simulation with a larger scattering coefficient μ_s^{max} and stores the information of the detected photons. This way, WMC can generate the simulation result in a short time with arbitrary absorption coefficient and any scattering coefficient smaller than μ_s^{max}. However, WMC when used in inverse problems is limited to infinite and semi-infinite homogeneous geometries. The Perturbation Monte Carlo (PMC) method[3] is one of the successful models suitable for

Address all correspondence to: Sailing He, Zhejiang University, Centre for Optical and Electromagnetic Research, Zhejiang Key Laboratory of Sensing Technologies, Hangzhou 310058, China. Tel: +86-571-88206525; Fax: +86-571-88206512; E-mail: sailing@kth.se.

inverse problem.[4,5] In the traditional PMC model, one traces the photon trajectory when the photon migrates in a turbid medium, which indicates the requirement of a large amount of random-access memory (RAM) space to record the information of the photon random walk in MC simulation. Therefore, the traditional PMC algorithm greatly limits its feasibility of acceleration by graphics processing unit (GPU).[6–8] In addition, the traditional PMC algorithm collapses when a large scattering coefficient change occurs in the simulation. A new algorithm is demonstrated in Ref. 9, where the PMC simulation is defined as a two-step PMC. In Ref. 9, only one detector with a detecting area of 25 mm^2 is used to record the effective seeds. In our study, in the first-step of PMC simulation, a seed of the random number generator (RNG), which is associated to a photon that reaches a region of interest (ROI), is defined as an effective seed and stored in the computer hard disk. The first step of the PMC is performed repeatedly to generate a large amount of effective seeds. In the second step of PMC by using effective seeds, we can place the detectors at arbitrary positions inside the region of interest to measure the photon weight. This way, for each detected photon, the information needs to be recorded is the seed of the RNG, which only takes 8 byte RAM in our study. Thus, this new algorithm is extremely suitable for GPU calculations.

In our study, we develop GPU (NVIDIA® GeForce™ GT 240M) based code to record the effective seeds in the first-step of PMC and then perform the GPU based MC (i.e., the second step of PMC) by using the effective seeds to measure the photon weight at detectors. Hence, we do not use the scaling relationship [i.e., Eq. (1) in Ref. 9] in our GPU based two-step PMC. The MC simulation is performed by utilizing a homogeneous sample and two-layered slab geometry. The GPU based two-step PMC is compared with a standard GPU based MC simulation in both the spatial and temporal aspects. To get the deep information of the turbid medium (i.e., the diagnosis of mammary cancer), the transmission information is usually utilized to reconstruct the optical properties. So we only consider the transmission configuration in our study.

The turbid medium we choose is a 120 mm × 120 mm× 30 mm slab. The source is set at the midpoint of the top side, which is defined as the origin in our simulation (the Cartesian coordinate is used here). The region of interest in the first step of GP2MC is a 61 mm × 61 mm square centered at [0, 0, and 30 mm]. Thirty one detectors are placed at the bottom surface, aligned from (0, 0, and 30 mm) to (30, 0, and 30 mm) at intervals of 1 mm. These detectors are inside the region of interest. The geometry of the detector is a square of size 1 mm^2. In the first step of GP2MC, a tissue that has spatial homogeneous optical absorption, scattering, and anisotropy coefficients of $\mu_a = 0.1$ (cm^{-1}), $\mu_{s0} = 100$ (cm^{-1}), and $g = 0.8$, respectively, is used to generate the effective seeds (μ_a and g remain the same in our study). For fast performance of GP2MC, 3840 threads are set to perform simultaneously during the GPU calculation. Once the GPU kernel program is running, the photon completes 2000 times random walk for each thread. The probability for one photon penetrating into the turbid medium and reaching the region of interest within 200 times of random walk is less than 10^{-5} (this result is calculated by using MC simulation). In other words, the probability

for each thread to detect more than 10 photons (i.e., the number of effective seeds) within 2000 times of random walk in the region of interest is less than 10^{-50}. Therefore, 0.3 Mbyte global device memories (i.e., $3840 \times 10 \times 8$ byte, one thread needs 10×8 byte to record up to 10 effective seeds) are more than enough to record the effective seeds in all 3840 threads and to avoid contention of global device memory. In the first step of PMC, about 4.74×10^7 photons are detected in the region of interest by injecting 2×10^9 photons. In the second step of PMC, about 3.45×10^7, 3.82×10^7, 3.98×10^7, and 4.18×10^7 photons are detected in the region of interest when the scattering coefficient is 20 cm^{-1}, 40 cm^{-1}, 60 cm^{-1}, and 80 cm^{-1}, respectively. The number of the detected photons in each of the 31 detectors (corresponding to different locations of x) is shown in Fig. 1(d) for five different values of the scattering coefficient.

Since in the first step of our GP2MC, we only record the effective seeds of the RNG, rather than the photon trajectory, and in the second step we use these effective seeds to re-generate the photon trajectory. According to the nature of the two-step PMC method, it works for a large change in the scattering coefficient. We perform the second step of GP2MC with $\mu_s = 20, 40, 60,$ and $80 \ (\text{cm}^{-1})$ to verify the hypothesis about our two-step PMC described before. The GP2MC is also performed with a two-layer slab, which is composed of a 15 mm-thick upper layer with $\mu_s = 40 \text{ cm}^{-1}$ and a 15 mm-thick bottom layer with $\mu_s = 60 \text{ cm}^{-1}$. As shown in Fig. 2(a), the plots of the transmission data obtained from GP2MC (the dotted line) and GPMC (the solid line) agree well. The relative errors are shown in the inset of Fig. 2(a). This figure reveals that the two-step PMC method is capable of treating large scattering perturbation. Besides, since the GP2MC program utilizes the
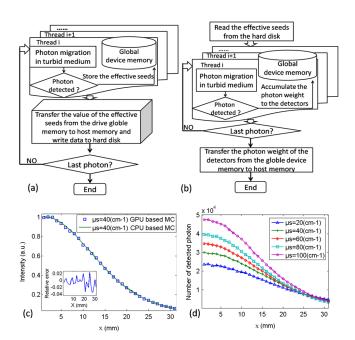

Fig. 1 The scheme of the GPU based two-step PMC. In the first step (Fig. 1(a)), the effective seeds are stored in computer hard disk; these effective seeds are used in the second step (Fig. 1(b)) to perform the MC simulation with another set of optical coefficients. (c) Comparison of transmission results at $(x, y = 0, z = 30 \text{ mm})$ obtained with CPU based MC and GPU based MC. (d) In the second step of PMC, the number of detected photon in each one of the 31 detectors (corresponding to different locations of x).
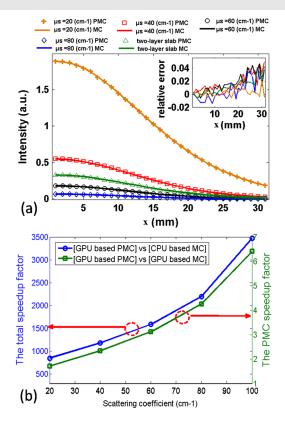

Fig. 2 (Color online) (a) Comparison of transmission results of GP2MC and GPMC at $(x, y = 0, z = 30 \text{ mm})$. (b) The speedup gain of the GP2MC in a spatially-resolved case. See the text for detail.

effective seeds, as the scattering coefficient scales from $\mu_{s0} = 100 \text{ cm}^{-1}$ to μ_s ($\mu_s = 20, 40, 60,$ and $80 \ (\text{cm}^{-1})$, $\mu_s < \mu_{s0}$) in the second step, the trajectory of the detected photon in the first step of the GP2MC will be enlarged by a factor (μ_{s0}/μ_s) in the second step of the GP2MC.[2] If one detects a photon at the position i in the first step, the same photon might be detected at another position j in the second step. The position j may be set arbitrarily inside the region of the interest defined in the first step of GP2MC. This means that the ratio of the detected versus total number of injected photons in the second step of the GP2MC can be larger than a standard GPMC in transmission configuration. Thus, the computation time of GP2MC is shorter than that of GPMC (it is worth mentioning that the computation time of the GP2MC refers to the perform time of the second step of GP2MC, since the first step of GP2MC is performed for only 'one' time to store the effective seeds in the hard disk). The speedup factor between the computation time of GPMC program and GP2MC program is shown in Fig. 2(b) (the square marked line, defined as the PMC speedup factor). One can see that the speedup factor decreases as the scattering perturbation increases. This is because that in the second-step PMC, as the scattering coefficient changes, the photon's trajectory will change from its original path. Hence, part of the photon will not arrive at the detector and does not contribute to the final results. Also, in GPMC, the ratio of detected photons to the total incident photons is increased as the scattering coefficient becomes smaller. In addition, as the absorption coefficient becomes lager, the detected photon weight gets smaller. Hence, we need to trace more number of photons (also, more effective seeds need to be recorded) to reduce the statistical error of MC program.
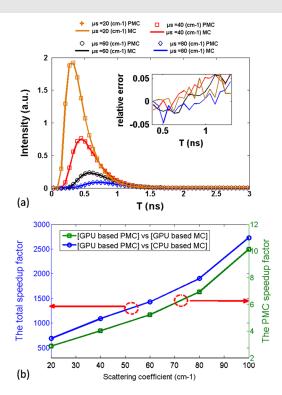
Fig. 3 (Color online) (a) Comparison of TPSF of GP2MC and GPMC at $(x = 0, y = 0, z = 30$ mm). (b) The speedup gain of the GP2MC in a time-resolved case.

Consequently, the required computation time becomes longer. The speedup factor between the computation time of GP2MC program and CPU (2.6 GHz) based MC program (the circle marked line, defined as the total speedup factor) is also shown in Fig. 2(b).

It is important to notice GP2MC's potential in time-resolved issues. Intensity versus time is defined as a temporal point spread function (TPSF).[9] To generate the TPSF, we set a point source at the origin of the coordinate system and a detector with 1 mm^2 detection area at the bottom of the turbid medium (i.e., [0, 0, and 30 mm]). The effective seeds described before are used to perform the second step of GP2MC to generate the TPSF. The plots of the TPSF in Fig. 3(a), which are obtained from the GP2MC (the dotted line) and GPMC (the solid line) program, agree well with each other. As mentioned before, the utilization of the effective seeds can increase the ratio of detected photons to the total incident photons in the second step of GP2MC. Therefore, GP2MC can generate the TPSF faster than GPMC, as shown in Fig. 3(b) (the square marked line). Furthermore, GP2MC's performance is significantly better than a standard CPU based MC simulation for calculating the TPSF (the circle marked line). The speedup gain is different between Figs. 2(b) and 3(b) as we use 31 detectors in spatially-resolve case and only one detector in time-resolved case. In addition, compared

with the spatially-resolved PMC, the time-resolved PMC requires recording the length of the photon trajectory after each random walk inside the medium.

In this letter, we transplant the PMC method to GPU, and perform the GP2MC on a personal laptop. The GP2MC achieves more than 1000 speedup factor as compared with MC simulation on CPU. Since GP2MC is based on Compute Unified Device Architecture,[6] some CPU procedures are used to read the effective seeds from the hard disc to the GPU and save the transmission data from GPU to the hard disc. These CPU procedures slow down the GP2MC and thus a GPU-based PMC can't improve the speed so much as one would expect (the speedup factor is only about six in our numerical example). In diffusive optical imaging for cancer diagnosis or brain functional monitoring, the change of the concentration of oxygenated hemoglobin (HbO) or hemoglobin (Hb) gives rise to changes in both the absorption and scattering coefficients. At the same time, the results shown in Figs. 2 and 3 clearly demonstrate that the two-step PMC can also handle RTE with a large change in the scattering coefficients. Therefore, it is feasible to apply the GP2MC as an appropriate forward solver in diffusive optical imaging to reconstruct the absorption and scattering coefficients of a turbid medium.

References

1. L. Wang, S. L. Jacques, and L. Zheng, "MCML—Monte Carlo modeling of light transport in multi-layered tissues," *Comput. Methods Programs Biomed.* **47**(2), 131–146 (1995).
2. A. Kienle and M. S. Patterson, "Determination of the optical properties of turbid media from a single Monte Carlo simulation," *Phys. Med. Biol.* **41**(10), 2221–2227 (1996).
3. A. Sassaroli et al., "Monte Carlo procedure for investigating light propagation and imaging of highly scattering media," *Appl. Opt.* **37**(31), 7392–7400 (1998).
4. C. K. Hayakawa et al., "Perturbation Monte Carlo methods to solve inverse photon migration problems in heterogeneous tissues," *Opt. Lett.* **26**(17), 1335–1337 (2001).
5. J. Chen and X. Intes, "Time-gated perturbation Monte Carlo for whole body functional imaging in small animals," *Opt. Express* **17**(22), 19566–19579 (2009).
6. E. Alerstam, T. Svensson, and S. Andersson-Engels, "Parallel computing with graphics processing units for high-speed Monte Carlo simulation of photon migration," *J. Biomed. Opt.* **13**(6), 060504 (2008).
7. 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).
8. N. Ren et al., "GPU-based Monte Carlo simulation for light propagation in complex heterogeneous tissues," *Opt. Express* **18**(7), 6811–6823 (2010).
9. A. Sassaroli, "Fast perturbation Monte Carlo method for photon migration in heterogeneous turbid media," *Opt. Lett.* **36**(11), 2095–2097 (2011).