Reply to "Comment on 'A study on tetrahedron-based inhomogeneous Monte-Carlo optical simulation'"

Haiou Shen and Ge Wang*

School of Biomedical Engineering and Sciences, Virginia Tech, Blacksburg, VA 24061, USA
**wangg@vt.edu*

Abstract: We compare the accuracy of TIM-OS and MMCM in response to the recent analysis made by Fang [Biomed. Opt. Express **2**, 1258 (2011)]. Our results show that the tetrahedron-based energy deposition algorithm used in TIM-OS is more accurate than the node-based energy deposition algorithm used in MMCM.

OCIS codes: (170.3660) Light propagation in tissues

References and links

1. H. Shen and G. Wang, "A tetrahedron-based inhomogeneous Monte Carlo optical simulator," Phys. Med. Biol. **55**(4), 947–962 (2010).
2. H. Shen and G. Wang, "A study on tetrahedron-based inhomogeneous Monte Carlo optical simulation," Biomed. Opt. Express **2**(1), 44–57 (2011).
3. J. Havel and A. Herout, "Yet faster ray-triangle intersection (using SSE4)," IEEE Trans. Vis. Comput. Graph. **16**(3), 434–438 (2010).
4. E. Alerstam, W. C. Yip Lo, T. D. Han, J. Rose, S. Andersson-Engels, and L. Lilge, "Next-generation acceleration and code optimization for light transport in turbid media using GPUs," Biomed. Opt. Express **1**(2), 658–675 (2010).
5. Q. Fang, "Comment on 'A study on tetrahedron-based inhomogeneous Monte-Carlo optical simulation'," Biomed. Opt. Express **2**, 1258–1264 (2011).

Reply

Simulation speed

In [2], we compared the latest versions of several optical Monte Carlo (MC) simulation packages with our recently developed TIM-OS [1]. Particularly, MMCM was downloaded on September 29, 2010 from its website (http://mcx.sourceforgo.net/mmc) and compiled with the best setting in the package. As shown in Dr. Fang's comment [5], he recently updated the MMCM package that now takes advantage of the SSE instructions and the Intel compiler, yielding a substantial performance gain. However, the latest MMCM still does not take the thread racing condition into account. As pointed out by Alerstam [4], thread racing may compromise data integrity. We also observed this problem in the MMCM results.

It is underlined that TIM-OS photon-tetrahedron intersection style has a less computational complexity than the Plücker-coordinate scheme used in MMCM [2,5]. When we do photon-tetrahedron intersection tests, a photon is actually inside a tetrahedron. Such a tight restriction on the position of the photon greatly reduces the computational complexity. As a result, while the Plücker-coordinate algorithm utilizes all the equations in [3], the original TIM-OS algorithm only uses the popular ray-plane intersection equation.

Simulation accuracy

Figure 1 illustrates the problem in [5]. While the solid curve shows the true value y_{truth}, $y_{mmc}(i)$ and $y_{timos}(i)$ are the values used in [5] to compare MMCM and TIM-OS. However, each $y_{timos}(i)$ datum he used had two parts: $y_{timos}(i) = (\int_{(i-1)\Delta x}^{i\Delta x} f(x)dx + \int_{i\Delta x}^{(i+1)\Delta x} f(x)dx)/2$, where $\int_{(i-1)\Delta x}^{i\Delta x} f(x)dx$ and $\int_{i\Delta x}^{(i+1)\Delta x} f(x)dx$ were the values TIM-OS estimated at the positions $(i-1/2)\Delta x$ and $(i+1/2)\Delta x$, respectively. Hence, $y_{timos}(i)$ actually was a linear interpolation of two TIM-OS results. It is not fair to compare a linearly interpolated TIM-OS result to a directly computed MMCM result.

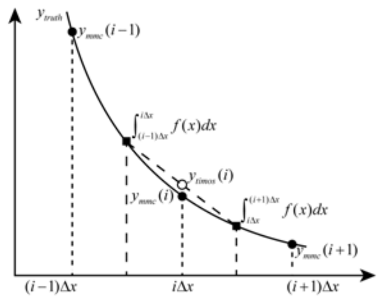

Fig. 1. Illustration of the problem in Dr. Fang's Comment.

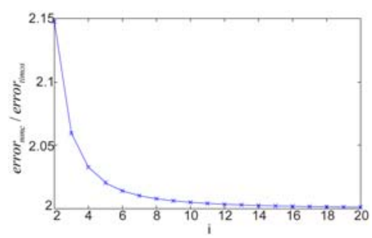

Fig. 2. Comparison of MMCM and TIM-OS in terms of the relative error.

To address this discrepancy for the problem shown in Fig. 1, we compared the results of MMCM and TIM-OS to the true value $1/(i\Delta x)$ at an arbitrarily selected point $i\Delta x$. In this case, by the meshing requirements of the two simulators, the integral range for MMCM was from $(i-1)\Delta x$ to $(i+1)\Delta x$ and the range for TIM-OS was from $(i-1/2)\Delta x$ to $(i+1/2)\Delta x$. We have

$$y_{truth} = f(x) = 1/x$$
$$y_{mmc} = (\int_{(i-1)\Delta x}^{(i+1)\Delta x} f(x)\varphi_i(x)dx)/\Delta x = ((i+1)\ln(i+1) + (i-1)\ln(i-1) - 2i\ln(i))/\Delta x$$
$$y_{timos} = (\int_{(i-1/2)\Delta x}^{(i+1/2)\Delta x} f(x)dx)/\Delta x = (\ln(i+1/2) - \ln(i-1/2))/\Delta x$$

Then, the relative errors for MMCM and TIM-OS were derived as

$$error_{mmc} = (y_{mmc} - 1/i\Delta x)i\Delta x = i(i+1)\ln((i+1)/i) - i(i-1)\ln(i/(i-1)) - 1$$
$$error_{timos} = (y_{timos} - 1/i\Delta x)i\Delta x = i(\ln(i+1/2) - \ln(i-1/2)) - 1$$

Therefore $\lim_{i->\infty} error_{mmc} / error_{timos} = 2$. Figure 2 plots $error_{mmc} / error_{timos}$ for $2 \le i \le 20$.

Furthermore, we considered a more realistic example in which a pencil beam passed through an absorbing-only media, and the intensity of the light beam would obey Beer's law along the light path. We got similar result: $\lim_{\Delta x->0} error_{mmc} / error_{timos} = 2$ and $error_{mmc} / error_{timos} > 1$ for $\Delta x > 0$. We also set up a mesh to test MMCM and TIM-OS under the above condition. Our experimental results are in an excellent agreement with the

#145724 - $15.00 USD Received 11 Apr 2011; accepted 13 Apr 2011; published 19 Apr 2011

analytical prediction. We prepared a package containing all the files for the reader to repeat the experiments, which can be downloaded from http://imaging.sbes.vt.edu/software/tim-os.

Acknowledgment

The work is partially supported by NIH R01HL098912.