Parallelized Monte Carlo software to efficiently simulate the light propagation in arbitrarily shaped objects and aligned scattering media

Christian Johannes Zoller
Ansgar Hohmann
Florian Foschum
Simeon Geiger
Martin Geiger
Thomas Peter Ertl
Alwin Kienle

Parallelized Monte Carlo software to efficiently simulate the light propagation in arbitrarily shaped objects and aligned scattering media

Christian Johannes Zoller,[a,*] **Ansgar Hohmann,**[a] **Florian Foschum,**[a] **Simeon Geiger,**[a]
Martin Geiger,[b] **Thomas Peter Ertl,**[c] **and Alwin Kienle**[a]
[a]Ulm university, Institute for Laser Technologies in Medicine and Metrology, Ulm, Germany
[b]Ulm university, Department of Orthodontics, Ulm, Germany
[c]Dentsply Sirona, Hanau, Germany

Abstract. A GPU-based Monte Carlo software (MCtet) was developed to calculate the light propagation in arbitrarily shaped objects, like a human tooth, represented by a tetrahedral mesh. A unique feature of MCtet is a concept to realize different kinds of light-sources illuminating the complex-shaped surface of an object, for which no preprocessing step is needed. With this concept, it is also possible to consider photons leaving a turbid media and reentering again in case of a concave object. The correct implementation was shown by comparison with five other Monte Carlo software packages. A hundredfold acceleration compared with central processing units-based programs was found. MCtet can simulate anisotropic light propagation, e.g., by accounting for scattering at cylindrical structures. The important influence of the anisotropic light propagation, caused, e.g., by the tubules in human dentin, is shown for the transmission spectrum through a tooth. It was found that the sensitivity to a change in the oxygen saturation inside the pulp for transmission spectra is much larger if the tubules are considered. Another "light guiding" effect based on a combination of a low scattering and a high refractive index in enamel is described. © *2018 Society of Photo-Optical Instrumentation Engineers (SPIE)* [DOI: 10.1117/1.JBO.23.6.065004]

Keywords: Monte Carlo simulation; parallelization; ray tracing; anisotropic light propagation.

Paper 180097RR received Feb. 15, 2018; accepted for publication Jun. 1, 2018; published online Jun. 22, 2018.

1 Introduction

Noninvasive investigations of biological tissue with light have numerous applications in modern diagnostics, ranging from pulse oximetry to optical imaging techniques based on (auto) fluorescence, photoacoustic, or diffuse optical tomography.[1–4] By utilizing diffuse reflectance spectroscopy, useful information about the skin physiology, morphology, and composition can be obtained.[5,6] In the dental field, a variety of subjective diagnostics tools are used.[7] Optical tools, such as the pulse oximetry or caries detection based on fluorescence, were adapted and tested for noninvasive, objective diagnostics.[8,9] Their feasibility was shown, for example, for pulse oximetry, by correlating the oxygen saturation signal with a subjective diagnostic tool.[7,9,10] For the development of these and other powerful diagnostic methods, an understanding of the light propagation in turbid media is fundamental. For this reason, it is of special interest to be able to simulate the light propagation in arbitrarily shaped three-dimensional (3-D) objects, such that no simplification to simple geometrical structures has to be made.[11] Monte Carlo simulations have previously been used in the dental field to gain a better understanding of the light propagation. Applications are, for example, root canal sterilization, tissue ablation, and curing of dental restoration materials.[12–15]

The propagation of electromagnetic waves in a medium can be accurately described by Maxwell's equations. Analytical solutions are found for simple geometries, such as spheres or cylinders, whereas numerical methods, such as the finite-difference time-domain method (FDTD) or the more efficient born series, are used to solve Maxwell's equations for more complex structures.[16–19] Even though these numerical tools are a very accurate way to describe the light propagation in turbid media, they have some restrictions. By now only media with a size of tens of micrometers can efficiently be solved by the algorithms, due to the immense memory and computational power needed. For larger volumes, the so-called radiative transport equation (RTE) is usually applied.[20] Within this theory, a medium is characterized by the following optical parameters: the absorption coefficient μ_a, the scattering coefficient μ_s, the refractive index n, and the scattering phase function $p(\theta)$. A commonly used scattering phase function is the Henyey–Greenstein (HG) phase function, which is rotationally symmetrical and can therefore be characterized by a single parameter, g, the so-called anisotropy factor.[21] The RTE has recently been solved analytically for simple geometries, such as the semi-infinite space or a layered medium including fluorescence.[22,23] However, the RTE cannot be solved analytically for arbitrarily shaped objects, as it would be needed for several real-life applications. A numerical way to solve the RTE up to, in principle, any desired accuracy is the Monte Carlo simulation, where many photon packages propagate on random paths through a turbid medium, based on its optical properties.[24,25] Physical

*Address all correspondence to: Christian Johannes Zoller, E-mail: christian.zoller@ilm-ulm.de

laws and their related probability functions are used to generate each photon path separately and statistically independent of each other. Over the last few years, several Monte Carlo software packages have been released to simulate the light propagation in layered media and also in complex-shaped objects.[26–29] Due to several advantages, like its high flexibility or its easy and straightforward way to implement, the Monte Carlo method has been referred to as the gold standard to numerically solve the RTE.

The main drawback of the Monte Carlo method is its potentially huge simulation time for accurate results due to its statistical nature. To reduce these problems, several software-based solutions, such as variance reduction methods or rescaling of single simulations, were developed.[30–32] As all photon paths are generated independently of each other, the algorithm can be parallelized well. Up to a thousandfold hardware-based acceleration has been established using graphics processing units (GPUs) instead of the central processing units (CPUs) for layered media.[33,34] In addition, parallelized Monte Carlo tools for the GPU have been presented to simulate the light propagation in arbitrarily shaped objects.[35–37] A known problem for high-performing Monte Carlo programs is an efficient way to realize all kinds of light sources for arbitrarily shaped surfaces. Shin and Kwon[38] found a solution by preselecting all possible tetrahedra, where a photon can potentially start according to a defined light source. During the simulation, they only have to search for the starting tetrahedron among the preselected ones. Yao et al.[39] described a way by adding additional tetrahedra connecting the light source and detector with the surface of the object. The tetrahedra are set to be transparent, such that they do not affect the path of the photons. A disadvantage of both methods is the preprocessing step, which can potentially become complex for fractal surfaces. The second method requires additional tetrahedra and therefore increases the needed memory as well as the total simulation time.

In this paper, a GPU-based Monte Carlo program is presented to simulate the light propagation in arbitrarily shaped objects, such as a tooth. A feature of this Monte Carlo program is an efficient way to realize all kinds of light-sources illuminating the surface of an arbitrarily shaped object. Unlike other works, no extra preprocessing steps for the light source are needed, in addition the generation of the simulation mesh itself, which can be achieved, for example, using TetGen.[40] Additionally, photons leaving and reentering a concave object can be easily considered with the proposed method. A hundredfold acceleration compared with CPU-based Monte Carlo simulations was found for typical optical properties of biological tissue. To simulate aligned tissue, the program is designed to efficiently consider anisotropic light propagation, e.g., caused by cylinder scattering in addition to rotationally symmetrical phase functions. The design of the program allows the user to define the orientation and concentration of the cylinders for each tetrahedron individually.

In this study, the developed software is used to calculate transmission spectra through a human tooth, where the dominant anisotropic light propagation caused by the tubules in the dentin has been considered. Analogue simulations, where the phase functions caused by cylinder scattering are replaced by the rotationally symmetrical HG phase function, were done. By comparing the results, the importance of the tubules for the sensitivity of the transmission spectrum with respect to a change of the oxygen saturation in the pulp is shown.

## 2 Methods

2.1 Monte Carlo Software Package

The developed Monte Carlo software (MCtet) uses a tetrahedral mesh to represent the objects, which has several benefits compared with voxels or layered arrangement. Arbitrary shapes can be taken into account adaptively by refining the mesh in regions of high curvature on the one hand and taking large elements for less structured regions on the other hand. This leads to a well-balanced state of accuracy and memory demand. In this program, each tetrahedron is expressed by four integers, linking the four nodes to the matrix, where the coordinates of all nodes are saved. Additionally for each of a tetrahedron's four faces, the connected neighbor tetrahedra, the normal vectors, and the total reflectance angles are stored to accelerate the simulation speed. Different optical properties can be assigned to each tetrahedron individually by addressing a material number to each element. A flow chart of the MCtet is shown in Fig. 1.

Some functions of the Monte Carlo simulation are similar to those described by Wang et al.,[26] like the calculation of the step length, the survival roulette, the decision if a photon is reflected at a refractive index mismatch, and the calculation of a direction for the HG phase function. The functions needed due to the tetrahedral representation of the objects, most importantly the intersection test of a photon with the tetrahedron borders, are comparable with the functions described by Shen and Wang,[27,28] except for the following improvement. Shen and Wang checked all four faces if the ray intersects the plane of the face. By temporarily storing the face number, where a photon entered a tetrahedron, only three faces need to be checked for the subsequent intersection test. All parts of the program concerning the light propagation of a photon inside the diffuse medium by the Monte Carlo algorithm can be found inside the red box in Fig. 1.

The parts inside the red box are implemented with OpenCl.[41] Therefore, the program can be executed in parallel on GPUs and CPUs of different vendors compared with a CUDA-implementation, for which the software would be limited to NVIDIA-GPUs. Nested OpenMP is used to spread the work load across multiple GPUs and to use several CPU cores for the ray tracing per GPU. As the program is implemented with OpenCl, it is also possible to spread the work load of the Monte Carlo algorithm among CPU cores and GPUs at the same time. In most cases, this is not done for two reasons. First, the performance of Monte Carlo simulation for light propagation on a GPU is up to thousandfold faster compared with a CPU.[33] Second, the CPU cores are used in MCtet to handle the light source as described in the following chapter.

2.2 Simulating Arbitrary Light Sources

In this paper, a concept is presented where neither a preprocessing step is needed nor additional tetrahedra to realize almost any desired light source. Additionally the method enables to track photons leaving a concave object, traveling through air and potentially reenter again the scattering medium at a different position. The elements in the red box of Fig. 1 are the basic Monte Carlo functions running on the GPU, which were expanded for more challenging tasks, such as anisotropic light propagation. Light sources as well as photons leaving the object are dealt by the elements outside the red box
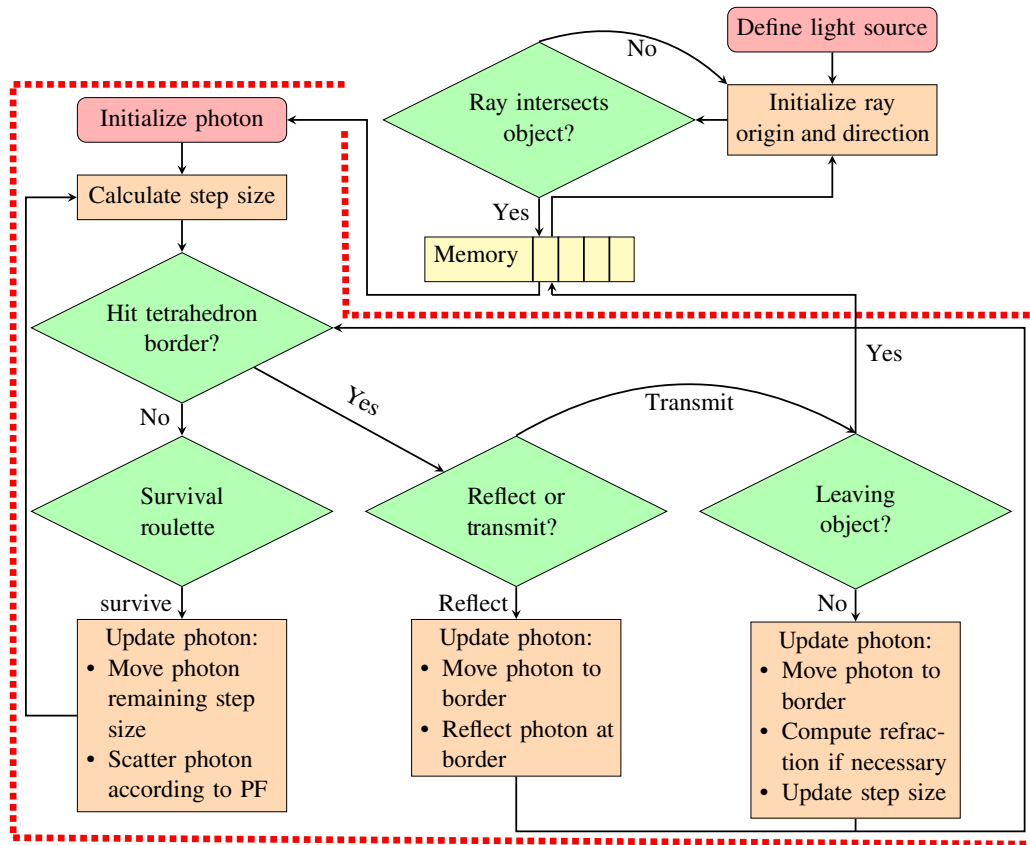
Fig. 1 Flow chart of the Monte Carlo software package. The parts inside the red box handle photons propagating inside the turbid medium and are executed on the GPU. The parts outside the red box treat the photons traveling through air and are executed on the CPU based on the ray-tracing kernel Embree. Both parts are running in parallel. (PF, phase function)

based on the fast ray-tracing kernel Embree on the CPU.[42] Embree is a collection of high-performance ray-tracing kernels developed at Intel. The target users of Embree are graphics application engineers, who want to improve the performance of their applications by leveraging the optimized ray-tracing kernels of Embree.[43] The basic idea is to use a highly optimized ray-tracing kernel for photons traveling in air where they do not change their direction and use a parallelized Monte Carlo algorithm for photons propagating in turbid media. Thus, the whole computational power of a system, the GPU as well as the CPU, is leveraged at the same time.

In a first step, the CPU is handling the light source in a way that it computes the first intersection point (IP) of the photon with the object. Therefore, the Embree function *rtcIntersect* is used to calculate the IP of a ray with an object. A ray is defined by an origin and direction.

```
RTCRay ray; // init embree ray
ray.org = RayOrg(); // function to set
   starting position of the ray
ray.dir = RayDir(); // function to set
   starting direction of the ray
```

There are two situations to define the starting position and direction of a ray. They are either determined by the desired light source or are set to the position and direction of a photon

leaving the object. Next the IP with the surface of the object is calculated, if the ray hits the object.

```
rtcIntersect(scene, ray); // get inter-
   section point of the ray with the object
```

The "scene" is the description of the simulation object by its surface triangles. If a photon is changing from air to turbid medium or vice versa its position and direction needs to be transferred between CPU and GPU. A detailed explanation about the parallelization of the GPU and the CPU can be found in Table 1, where steps 3 + 4 are repeated consecutively until the desired number of photons is completed. A single IP is defined by its position, direction, and the index of the first intersected tetrahedron, resulting in 7×4-bytes of memory and a total memory consumption of only 28 MB, if $N = 1 \times 10^6$ IPs are used and the data are stored in single precision. The positions and directions of photons leaving the object can be saved in the same memory as their starting positions and directions, as shown in Table 1, and no additional memory is needed because at most the number of started photons can be reflected, if no photon is absorbed. Furthermore, the communication is run asynchronously, in parallel with the GPU code, at the cost of a higher needed memory $2 \times$ IPs (56 MB). In general, the memory consumption on the GPU as well as the time needed to transfer the data between CPU and GPU cause only minor restrictions for modern graphics cards as they typically have up to 6 GB of memory and a bandwidth of a few GB/s.

Table 1 Parallelization of the CPU and GPU. The brackets $[\ldots]$ symbolize the GPU-memory used at the current step (see "memory" in Fig. 1). An IP is defined by the position, the direction, and the index of the tetrahedron where a ray hits the object. A reflected photon (RP) is defined by the current position and direction a photon is leaving the turbid medium. Tasks at the same step are running in parallel and the synchronization is done at the end of each task.

Step	CPU	GPU
1	• Calculate N IPs with desired light source	
	• Copy N IPs to GPU $[0 \ldots N-1]$	
2	• Calculate N IPs from desired light source	• Propagate $P1$ ($P1 < N$) photons $[0 \ldots P1]$
	• Copy N IPs to GPU $[N \ldots N + N - 1]$	• Save $R1$ ($R1 \leq P1 < N$) RPs $[0 \ldots R1]$
3	• Load $R1$ RPs from GPU $[0 \ldots R1]$	• Propagate $P2$ ($P2 < N$) photons $[N \ldots N + P2]$
	• Check if RPs reenter in the object	• Save $R2$ ($R2 \leq P2 < N$) RPs $[N \ldots N + R2]$
	• Calculate the remaining IPs with desired light source until P1 IPs are found	
	• Copy $P1$ IPs to GPU $[0 \ldots P1]$	
4	• Load $R2$ RPs from GPU $[N \ldots N + R2]$	• Propagate $P1$ ($P1 < N$) photons $[0 \ldots P1]$
	• Check if RPs reenter in the object	• Save $R1$ ($R1 \leq P1 < N$) RPs $[0 \ldots R1]$
	• Calculate the remaining IPs with desired light source until P2 IPs are found	
	• Copy $P2$ IPs to GPU $[N \ldots N + P2]$	

2.3 Simulation Setup

A human tooth morphology can mainly be separated in three different regions, the enamel, the dentin, and the pulp, as shown in Fig. 2(a).[44] Enamel is the outer most regions containing ~2.3% of water by weight and the rest mainly consists of the mineral hydroxylapatite.[45] For scattering in enamel, the HG phase function was applied. The biggest part of the tooth by volume is the dentin located under the enamel. Dentin contains ~10% of water by weight, 70% of hydroxylapatite by weight and the remaining parts are organic materials, mainly collagen
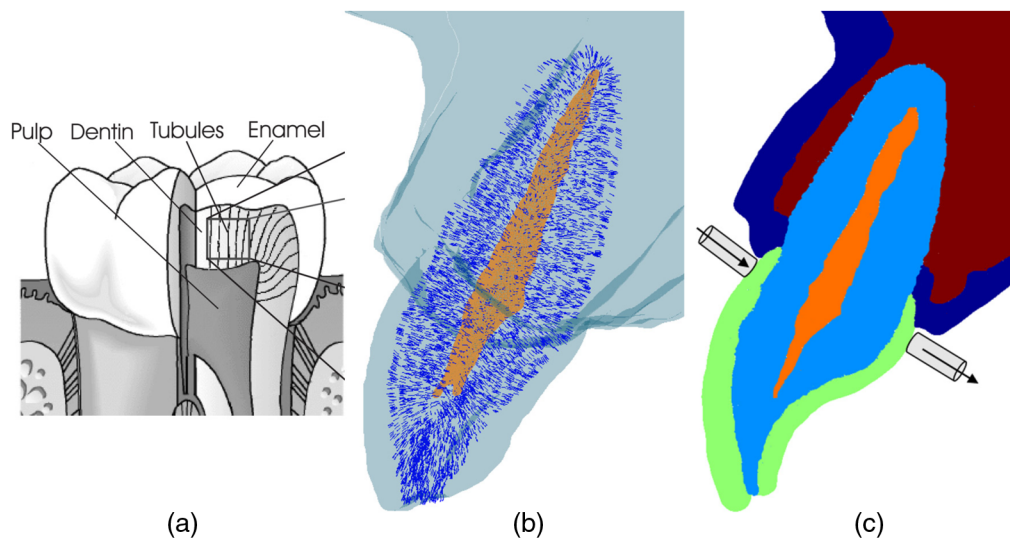
Fig. 2 (a) Human tooth morphology with its different components and the schematic orientation of the tubules extending from the DEJ to the pulp.[15] (b) Illustration of the tubules' orientation (blue lines) for every 20th tetrahedron in dentin of the used simulation object. The solid orange area marks the pulp. (c) Cross section of the used tetrahedral tooth mesh and the optical fibers to demonstrate the Monte Carlo simulation setup. The different colors indicate the different materials (dark blue: gingiva, light blue: dentin, green: enamel, orange: pulp, red: bone).

and elastin.[45] The assumed reduced scattering coefficient μ_s' for enamel and the absorption coefficient μ_a for enamel and dentin were chosen depending on the wavelength, comparable with ones reported previously.[46–48] The refractive index n was set constant to 1.633 and to 1.54 for enamel and dentin, respectively. The developed software MCtet is able to consider the important anisotropic light-propagation behavior of dentin caused by the tubules. Tubules are small pipes, embedded in the dentin material, with an assumed radius of 1 μm, which are filled with water, and a typical concentration of $c_A = 4.5 \times 10^4 \text{ mm}^{-2}$ was chosen.[49] The tubules form an aligned structure that extends from the dental enamel junction (DEJ) to the pulp as shown in Fig. 2(a) and serves as supply channel for the tooth.[50] In Fig. 2(b), the defined orientations of the tubules in the simulation object for every 20th dentin-tetrahedron are illustrated. Tubules are the main scatterers in dentin and cause a high-scattering coefficient, such that the scattering coefficient of dentin is significantly higher than the scattering coefficient of enamel, whereas the absorption coefficient is moderately higher. For both materials, the scattering is much higher than the absorption, causing the typical white color of a tooth. The pulp consists of living connective tissue located at the center of a tooth, which is supplied by blood via the tooth root. Typical values for the optical properties of connective tissue were used for the pulp.[51,52] The absorption coefficient of the pulp is varied in different simulations, based on a defined concentration of oxygenated hemoglobin and deoxygenated hemoglobin as the main absorber of blood in this spectral range.[53]

For all of the following simulations, the same incisor tooth model (left maxillary central incisor, tooth 21) was used and a cross section is shown in Fig. 2(c). The tetrahedral tooth mesh was generated at the Department of Orthodontics at Ulm University based on cone beam-computed tomography of the specimen.[54,55] The model consists of 1.9×10^5 nodes and 9.0×10^5 tetrahedra. Five different materials were separated for the simulation. The tooth itself is consisting of three materials: enamel, dentin, and the pulp as described before. The whole tooth is grounded in bone, which was assumed to be a highly scattering, low absorbing medium, and has no significant influence on the presented results. On top of the bone, a 1.5-mm-thick gingiva tissue layer is present. For the gingiva, the same optical properties as for the pulp (connective tissue) were assumed except of the absorption, which was fixed to oxygenated hemoglobin with a concentration of $3 \text{gHbO}_2/\text{L}$. An overview of the used optical properties is given in Table 2. MCtet was used to calculate transmission spectra through a tooth, considering the anisotropic light propagation caused by the tubules. For each of the investigated cases (different
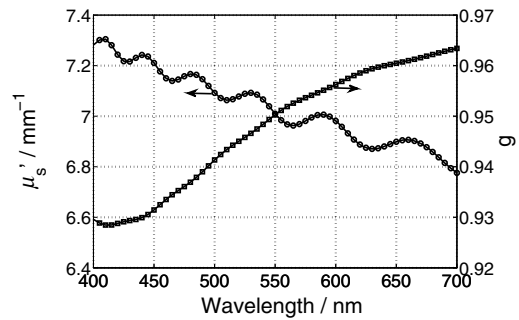
Fig. 3 Optical properties of dentin assuming that the tubules are randomly oriented calculated with Eqs. (3) and (4). These properties were used for the HG case.

absorptions in the pulp), an analogue simulation has been made to demonstrate the influence of the tubules on the light propagation, where all parameters were kept constant but the scattering in dentin was changed. Each analogue simulation calculates a second transmission spectrum assuming only rotationally symmetrical scattering in dentin, represented by the HG phase function. These simulations are called "HG case," whereas the simulations considering the anisotropic light propagation caused by the tubules are called "cylinder case." To estimate the scattering coefficient and anisotropy factor of dentin for the HG case, it is assumed that the tubules would be randomly oriented. In Sec. 2.4, the needed equations are derived to calculate the scattering coefficient and anisotropy factor of dentin. The results are shown in Fig. 3. For the illumination and detection, an optical fiber with an NA of 0.22 and diameter of 365 μm was modeled. This source type can easily be realized by the concept presented in this paper. The detection fiber was placed lingually and the illumination fiber was placed labially. Both fibers were placed on the tooth as close to the gingiva as possible without overlapping.

2.4 Anisotropic Light Propagation

The anisotropic light propagation behavior caused by aligned, e.g., fibrous, media has been reported for different materials, such as wood or hair.[56,57] In the dental field also anisotropic light propagation appears, caused by tubules in the dentin.[15,49] A detailed description of the human tooth morphology can be found in Sec. 2.3.

To correctly describe the light propagation in a tooth, a scattering phase function obtained by scattering light at a cylinder is used in addition to a rotationally symmetric phase function, in this case the HG phase function, for the Monte Carlo simulation.

Table 2 Assumed optical properties for the different materials of the tooth model from 400 to 700 nm.[46–53]

Material	μ_a/mm^{-1}	μ_s'/mm^{-1}	g	n
Enamel	0.06 to 0.02	0.8 to 0.2	0.9	1.633
Dentin cylinder case	0.2 to 0.1	Depending on tubules		1.54
Dentin HG case	0.2 to 0.1	See Fig. 3	See Fig. 3	1.54
Pulp/Gingiva	Depending on $\text{Hb}(\text{O}_2)/\text{L}$	4 to 1.5	0.77	1.39
Bone	0.001	4.2 to 3.3	0.85	1.55
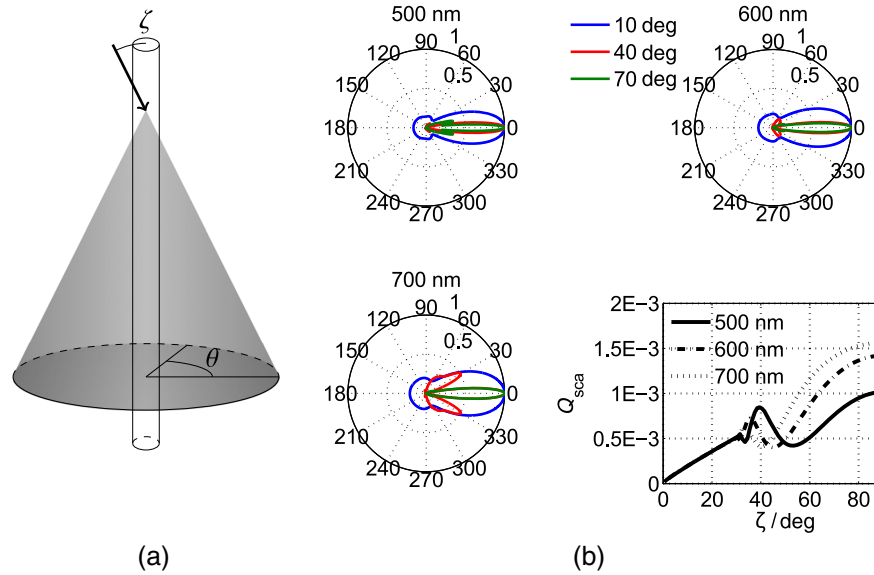
Fig. 4 (a) Scattering cone for one incident angle ζ. (b) Cylindrical scattering phase functions $p(\lambda, \zeta, \theta)$ on the scattering cone for $\lambda = 500$, 600, and 700 nm, for $\zeta = 10$ deg, 40 deg, and 70 deg, as well as the scattering efficiency $Q_{\mathrm{sca}}(\lambda, \zeta)$ for these wavelengths.

If a photon in the simulation is scattered by a cylinder at a certain incident angle ζ, its direction is restricted to be on the so-called scattering cone, as shown in Fig. 4(a). For an extreme angle $\zeta \to 90$ deg, all scattered photons directions will be in the same plane, defined by the incident direction and a vector perpendicular to the orientation of the cylinder. For another extreme angle $\zeta \to 0$ deg, the photons effectively do not change their direction as the scattering cone is extremely narrow and is pointing in forward direction. The effect, for $\zeta \to 0$ deg, that photons are scattered in the direction of the cylinders if they travel parallelly to the elongation of the cylinders is the reason for the so-called light guiding effect described in Sec. 3.2. For the Monte Carlo simulation, the probabilities for the scattering angles $p(\lambda, \zeta, \theta)$ on the scattering cone in Fig. 4(a), as well as the scattering efficiency $Q_{\mathrm{sca}}(\lambda, \zeta)$, are needed for all possible incident angles ζ for the desired wavelength λ. The scattering efficiency is defined as $Q_{\mathrm{sca}(\lambda, \zeta)} = \frac{C_{\mathrm{sca}(\lambda, \zeta)}}{G}$, where $C_{\mathrm{sca}(\lambda, \zeta)}$ is the scattering cross section and G is the particle cross-sectional area projected onto a plane perpendicular to the incident beam, in this case $G = 2r$ where r is the radius of the cylinder.[58] These values are obtained by solving Maxwell's equations for an incident electromagnetic plane wave scattered by an infinitely extended cylinder.[58]

Resulting values are exemplary shown in Fig. 4(b) for three wavelengths for the case of tubules, where a radius $r = 1$ μm, a refractive index $n_i = 1.33$ for the cylinders, and $n_o = 1.54$ for the surrounding medium was assumed. Calculations are not done during the simulation but in a preprocessing step, where $p(\lambda, \zeta, \theta)$ is tabulated in a cumulative way to efficiently obtain a direction. An example how to draw a scattering angle θ on the scattering cone with a random number χ is shown in Fig. 5. In MCtet, the orientation of the cylindrical scatterers as well as the concentration c_A can be defined for each tetrahedron individually. Hence, the light propagation during the simulation is highly depending on a photon's direction relative to the orientation of the cylinders inside the current tetrahedron. In addition to the scattering phase function, also the scattering coefficient $\mu_s(\lambda)$ has to be adjusted.
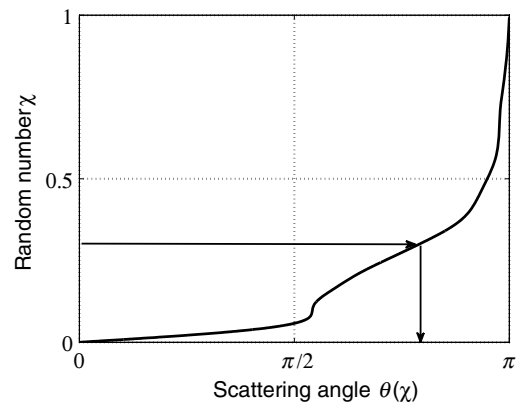
Fig. 5 Cumulative scattering phase function, calculated for the scattering of a plane wave at an infinitely large cylinder, for $\lambda = 700$ nm and an incident angle $\zeta = 40$ deg, to directly draw a scattering angle θ based on a random number.

The scattering coefficient

$$\mu_s(\lambda) = \mu_s^{\mathrm{cyl}}(\lambda, \zeta) + \mu_s^{\mathrm{hg}}(\lambda), \tag{1}$$

is the sum of the direction-independent scattering coefficient $\mu_s^{\mathrm{hg}}(\lambda)$ of the HG phase function and the direction-dependent scattering coefficient of the cylinder scattering $\mu_s^{\mathrm{cyl}}(\lambda, \zeta)$

$$\mu_s^{\mathrm{cyl}}(\lambda, \zeta) = Q_{\mathrm{sca}}(\lambda, \zeta) 2rc_A, \tag{2}$$

where r is the radius of the cylinder and $Q_{\mathrm{sca}}(\lambda, \zeta)$ is the scattering efficiency depending on ζ, as shown in Fig. 4(b) for three different wavelengths. The $\mu_s^{\mathrm{cyl}}(\lambda, \zeta)$ of dentin typically is ten up to hundred times higher than $\mu_s^{\mathrm{hg}}(\lambda)$, whereas the differences in $\mu_s'(\lambda)$ are not that high. For the Monte Carlo simulation at a certain wavelength λ, the scattering coefficient μ_s must be updated every time the direction of a photon changes within the same
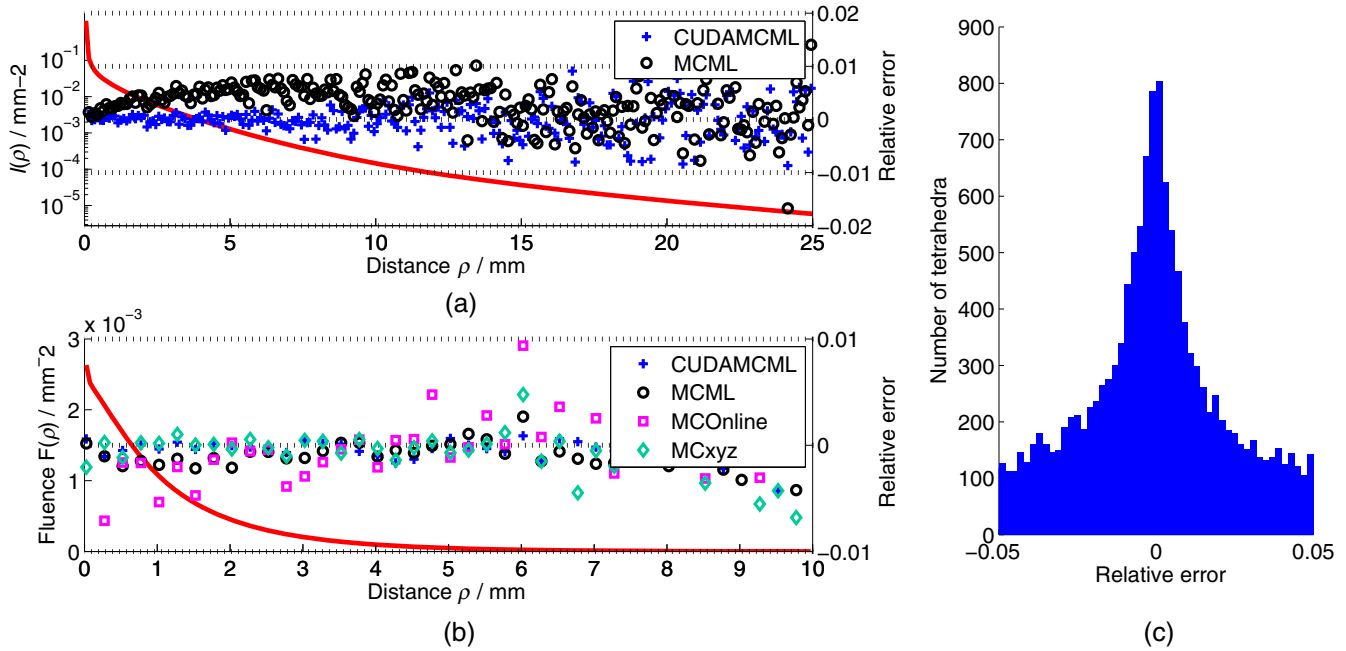
Fig. 6 (a) Spatially resolved reflectance $I(\rho)$ (red curve) and the relative errors compared with CUDAMCML and MCML for a three layered medium. (b) Spatially resolved fluence $F(\rho)$ (red curve) at a depth $z = 2$ mm and the relative errors compared with CUDAMCML, MCML, MCOnline, and MCxyz for a single slab. (c) Histogram of the relative error of the fluence comparison per tetrahedron with timOS for a cube.

tetrahedron or the photons are entering a tetrahedron. With a changing μ_s also, the $\mu_t = \mu_a + \mu_s$ is changing and consequently the step length has to be adjusted. Thus, every *Update photon* box in Fig. 1 is extended by a bullet point update scattering efficiency + adjust step size. One would have to do the same for the absorption coefficient, but the imaginary part of the refractive index of the cylinders n_i was set to zero for solving Maxwell's equations. Hence, the absorption efficiency for all angles ζ is zero and the absorption coefficient does not depend on the direction of the photon.

At each scattering event, either HG scattering or cylinder scattering is happening. Based on a random number $\chi \in [0,1]$ scattering at a cylinder happens if

$$\frac{\mu_s^{hg}}{\mu_s} = \frac{\mu_s^{hg}}{\mu_s^{cyl}(\zeta) + \mu_s^{hg}} < \chi,$$

otherwise the HG phase function will be used.

In Sec. 3, the influence of a cylindrical phase function is shown for a tooth model by comparing simulations where the cylindrical scattering is considered with simulations where only HG scattering is assumed. On comparison, the corresponding optical properties for the HG phase function are needed. Therefore, it is assumed that all cylinders are oriented randomly in space, instead of being aligned. Then, a scattering coefficient

$$\mu_s^{hg}(\lambda) = \int_0^{\pi/2} Q_{sca}(\lambda, \zeta) 2 r c_A \sin(\zeta) d\zeta, \quad (3)$$

and the related g-factor, with its definition

$$g(\lambda) = \frac{1}{4\pi} \int_{4\pi} (\vec{s} \cdot \vec{s}') p(\lambda, \vec{s}, \vec{s}') d\omega$$

$$= 2 \int_0^{\pi/2} \int_0^{2\pi} [\sin(\zeta)^2 \cos(\theta)$$

$$+ \cos(\zeta)^2] p(\lambda, \zeta, \theta) \sin(\zeta) d\theta \, d\zeta, \quad (4)$$

can be calculated by integrating the scattering efficiency $Q_{sca}(\lambda, \zeta)$ and scattering-phase function $p(\lambda, \zeta, \theta)$ over the entire solid angle.

3 Numerical Results

3.1 Validation

To check the correct implementation of the developed Monte Carlo software, comparisons with the software packages MCML, CUDAMCML, MCOnline, MCxyz, and timOS were done.[26,27,33,59,60] All comparisons were performed on an Intel Core i7-950 (3.07 GHz) with eight CPU cores using a Microsoft Windows 7 system. The used graphics card was an NVIDIA GeForce GTX 480. The simulations for the MCML, MCxyz, and timOS software packages were performed on one CPU core and were supposed to scale linearly with the number of additionally used cores. The calculations done by CUDAMCML were performed exclusively on the GPU. Results with MCOnline were generated using the online user interface at Ref. 61. Simulations with MCtet were performed on two CPU cores and the GPU simultaneously. MCtet is designed to spread the work across multiple GPUs, decreasing the simulation time linearly with the number of used GPUs. For the following tests, the relative errors were computed as the fraction $\frac{x_0 - x}{x}$, where x_0 is the respective results computed by MCTet

and x is the corresponding results computed by CUDAMCML, MCML, MCOnline, MCxyz, and TimOS. Comparisons with these five software have been performed for three different problem sets.

The first case proofs the correct implementation of the feature allowing photons to leave and reenter a concave object. Therefore, the spatially resolved reflectance for a three-layered slab, with a thickness of 4 mm for each layer, was calculated. The optical properties of the first and the last layer are set to $\mu_a = 0.01$ mm^{-1}, $\mu_s = 5$ mm^{-1}, $g = 0.8$, and $n = 1.4$, whereas the middle layer as well as the surrounding medium were set to be air. Please note that the second layer needs no representation for the MCtet software but must be explicitly defined as transparent for all other programs. To simulate 4×10^8 photons, MCML needed 3.6×10^4 s, CUDAMCML needed 27.71 s, and MCtet took 103.44 s, respectively. The spatially resolved reflectance calculated by MCtet as well as the relative errors compared with the other programs are shown in Fig. 6(a).

For the second test case, the spatially resolved fluence at a depth $z = 2$ mm for a single slab with a thickness of 4 mm was computed and compared with CUDAMCML, MCML, MCOnline, and MCxyz. The optical properties of the slab are set to $\mu_a = 0.01$ mm^{-1}, $\mu_s = 5$ mm^{-1}, $g = 0.8$, and $n = 1.0$. To simulate 4×10^8 photons, MCML needed 9.4×10^3 s,

CUDAMCML needed 83.77 s, MCxyz needed 2.3×10^4 s, and MCtet took 433.76 s, respectively. For MCOnline, it was not possible to set the number of photons or measure the time needed for the simulation. The spatially resolved fluence calculated by MCtet as well as the relative errors compared with the other programs are shown in Fig. 6(b).

For the first two test cases, a very good agreement with CUDAMCML and MCxyz was found and a deviation when comparing with MCML. Similar deviations with the MCML software have been reported previously by Shen and Wang,[27] where they explained it by the bias of the used pseudorandom number generator. The comparison with MCOnline showed a good agreement, but a simulation with more photons would be needed for a better discussion.

For the third test case, the fluence inside a cube ($2 \times 10 \times 10$ mm) was calculated and compared with the software package timOS. The cube is represented by 6172 nodes and 17954 tetrahedra and has the same optical properties as the layers in the slab described in first test case. The fluence was calculated and compared per tetrahedron. A histogram of the relative errors shown in Fig. 6(c) gives a good agreement of the two simulations.

The comparisons show a correct implementation of MCtet, also for the introduced method to consider photons leaving and
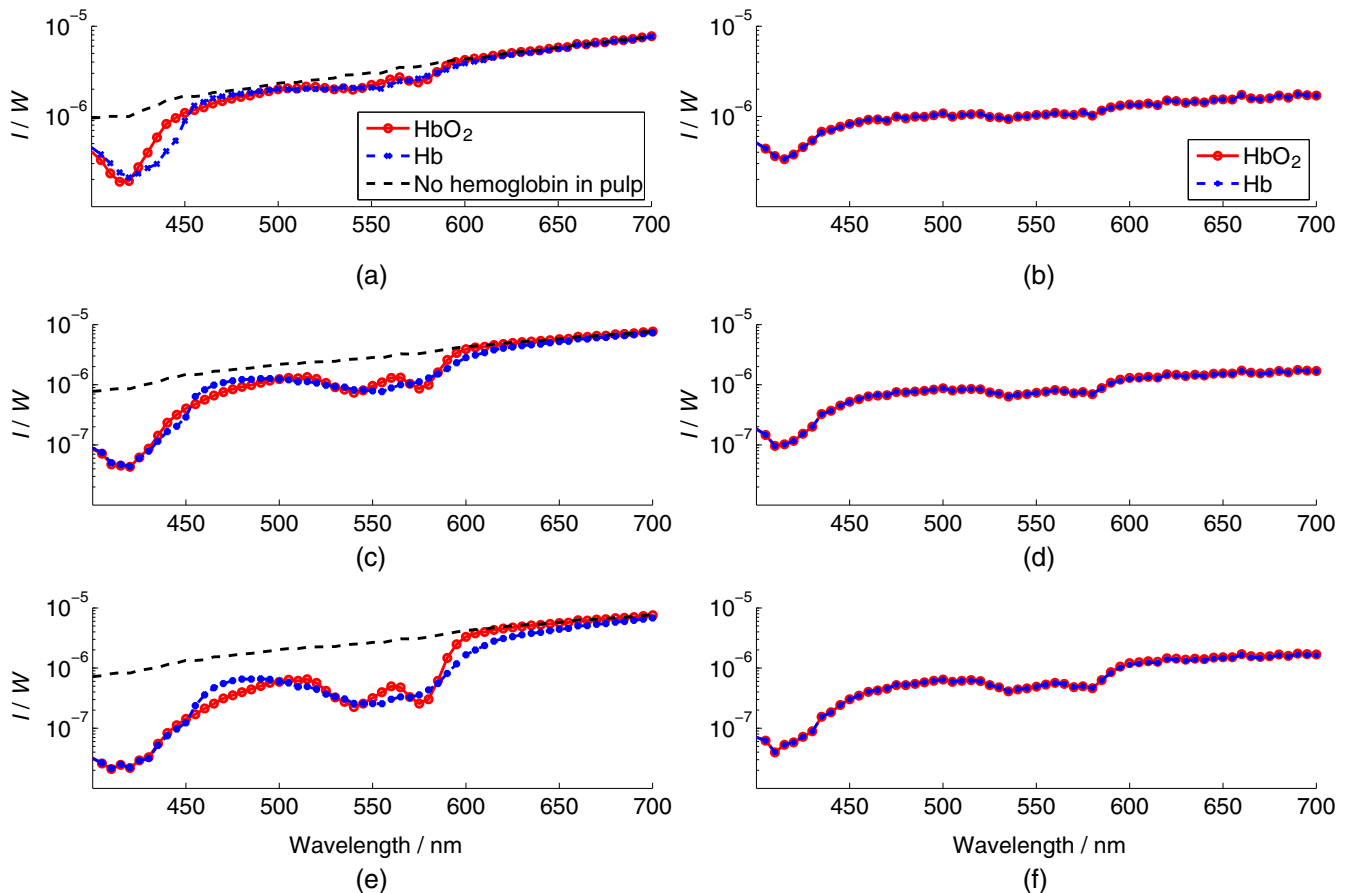
Fig. 7 Transmission spectra for different scattering types in the dentin as well as different concentrations of oxygenated and deoxygenated hemoglobin in the pulp. The left column shows the transmission spectra for the cylinder case, the right column shows the result for the HG case. For the first row, the hemoglobin concentration was set to 0.75 g Hb(O$_2$)/L, for the second row to 3 g Hb(O$_2$)/L, and for the third row to 7.5 g Hb(O$_2$)/L. The black dotted line for the cylinder case indicates a case where 0 g Hb(O$_2$)/L was assumed for the pulp. This spectrum not shown for the HG case as it is identical with the other lines. The total illuminated power was set to 1 W per wavelength.

reentering a concave object. The GPU-based CUDAMCML program has a four up to five times better performance compared with MCTet due to the restriction to layered media and therefore its high specialization. In particular, CUDAMCML does not have to check for the IP with the tetrahedron borders. Comparisons of the simulation times indicate a superior speed improvement of the GPU-implementation compared with the programs running on CPU.

3.2 Monte Carlo Simulation Results

Using the simulation setup as described in Sec. 2.3, a change of the oxygen saturation, from 100% to 0%, in the pulp was simulated for different concentrations of hemoglobin in the pulp. The concentrations were assumed to be 0.75 g $Hb(O_2)/L$, 3 g $Hb(O_2)/L$, and 7.5 g $Hb(O_2)/L$, respectively. For a concentration of 150 g $Hb(O_2)/L$ for whole blood, the used values relate to a volume concentration of 0.5%, 2%, and 5% of blood in the pulp.[62] It was assumed that the change in concentration only affects the absorption coefficient and not the other optical properties. The transmission spectra for the cylinder and HG case as well as for varying hemoglobin concentrations are shown in Fig. 7.

For the cylinder case, the transmission spectra of the two hemoglobin types are getting more distinguishable for higher concentrations of hemoglobin in the pulp for a spectral range between 500 and 600 nm and less distinguishable for the spectral range between 400 and 450 nm. This finding is caused by the large difference of the hemoglobin absorption of the two spectral ranges. In contrast, for the HG case, the hemoglobin types cannot be distinguished in their transmission spectra. This behavior can be explained by the light-guiding effect of the tubules described by Kienle et al.[15] It was found that light is guided along the orientation of the tubules due to scattering when illuminating them approximately parallel. Although the illumination is pointed on the enamel, which is scattering and therefore blurs the incident direction, a main direction of the light is still conserved when hitting the dentin. This main direction is pointing toward the pulp, approximately parallel to the preferred direction of the tubules.[50] Hence, the light-guiding effect is still dominant and major portions of the incident light are guided to the pulp. This can be seen by looking at the fluence, showed for a cross section in the incident plane for 570 nm in Fig. 8. A clearly higher intensity on a straight line toward the pulp can be seen for the cylinder case in Fig. 8(a) (top) compared with the HG case in Fig. 8(a) (bottom). The same effect holds for the parts of the light passing through the pulp and are likely to be guided toward the detection fiber, which can be seen in Fig. 8(b) (top). For the HG case, the direction-independent scattering coefficient is very high and no light guiding effect is presented in dentin. Therefore, less photons get to the pulp and reach the detection fiber, causing that the
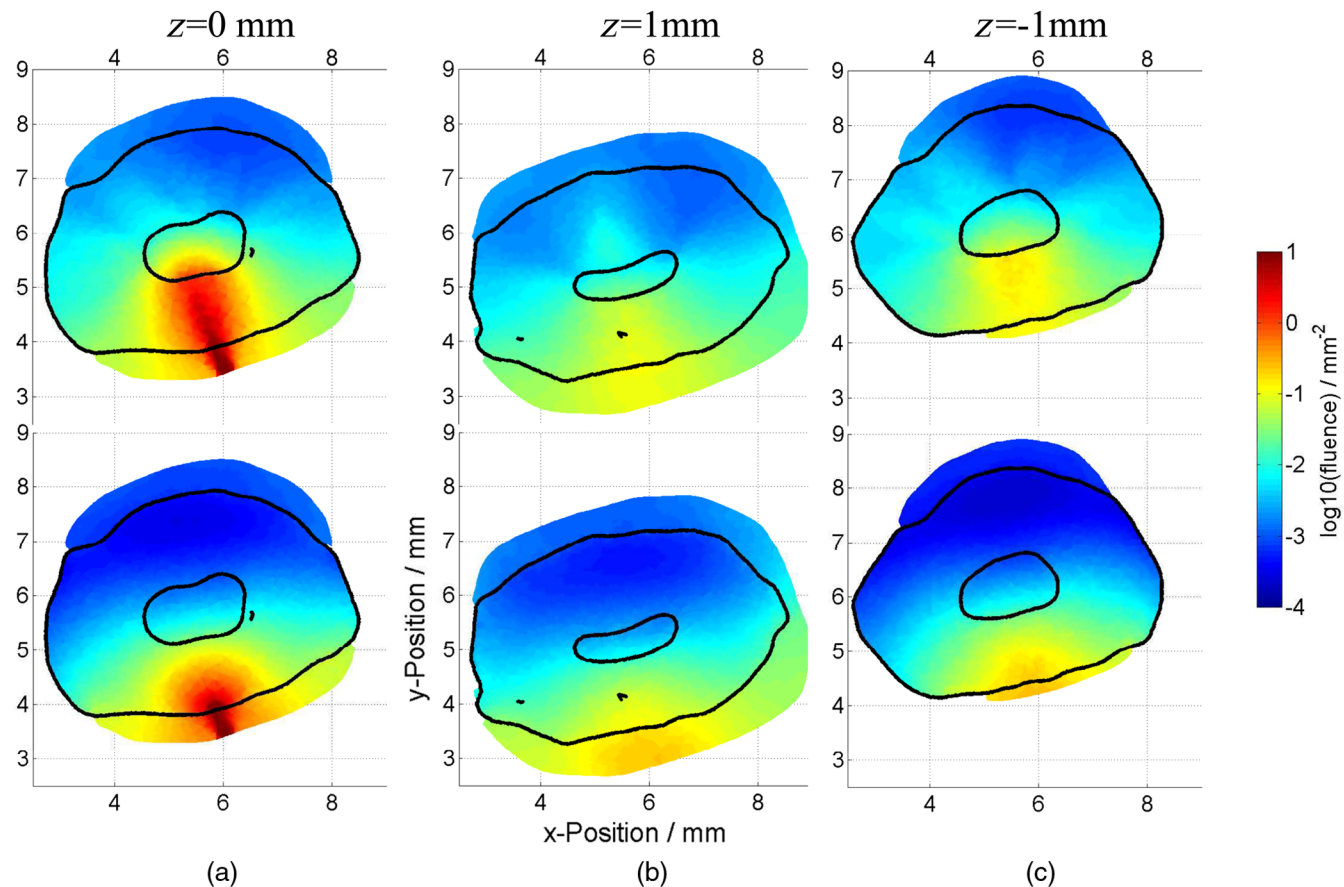
Fig. 8 Fluence compared for cylinder case (top) and HG case (bottom) for 570 nm and an oxygenated hemoglobin concentration of 3 g HbO_2/L in the pulp. (a) $z = 0$ mm denotes the cross section through the tooth in the illumination-detection plane. (b) $z = 1$ mm and (c) $z = -1$ mm denotes a shift of the $z = 0$ mm plane toward the tooth incisal edge and the tooth root, respectively. The outlined material is dentin, in the middle the pulp is located, and the surrounding material is enamel.

transmission spectra are insensitive to a change of the oxygen saturation. This direction-independent, high scattering is also the reason why the transmitted intensity can partially be higher for the HG case compared with the cylinder case, in spectral regions where the pulp is strongly absorbing. First, the light is not guided toward this highly absorbing part of the tooth. Second, the scattering in enamel is lower but the refractive index is higher compared with dentin. The combination of the low scattering and relative high refractive index mismatch results in another "light guiding" effect in the enamel. Comparable with an optical fiber, light is guided around the tooth inside the enamel to the detection fiber, although the enamel is not completely covering the dentin in the incident plane at the used tooth model [see Fig. 2(c)]. The effect can be seen by looking at a plane located parallelly to the incident plane $z = 1$ mm toward the incisal edge [Fig. 8(b)], where the enamel is starting to completely surround the dentin. By comparing the two fluence plots in Fig. 8(b), it can be seen that the fluence in the enamel is higher for the HG case, because more light is scattered back to the enamel from the dentin compared with the cylinder case. Light that is scattered back to the enamel is then guided inside the enamel around the dentin due to the refractive index mismatch and low scattering. The same effect is observed for the case where cylindrical scattering is assumed but less pronounced because a smaller fraction of the light is scattered back from the dentin into the enamel.

The transmission spectra in Fig. 7 for the cylinder case show the typical feature of oxygenated hemoglobin, two dips at 542 and 577 nm and for deoxygenated hemoglobin a single dip at 556 nm, respectively. Interestingly for the HG case only, if at all, a double dip can be observed independently on the type of hemoglobin in the pulp. The double dip is getting more pronounced for an increase of the HbO_2 as well as an increase of the Hb concentrations in the pulp. In addition to the pulp, the only tissue containing blood is the gingiva. Thus, this evolving double peak must be caused by the gingiva as the absorption and scattering coefficient of the other materials are smooth for this spectral range. This behavior can be explained by the described light guiding effect in the enamel, which is causing a higher fluence in the enamel. Therefore, more light can interact with the gingiva, which is partially covering the enamel [see Fig. 2(c)], and retrieve to the detection fiber. To clarify this, further simulations have been done where the absorber in the gingiva was changed to deoxygenated hemoglobin with the same concentration 3 g Hb/L. In these simulations, the results for the cylinder case remained almost unchanged, where the oxygen saturation of the pulp could still be distinguished in the transmission spectra in terms of a single and double dip. The transmission spectra for the HG case are still nonsensitive to the change of the hemoglobin type but showed the feature of deoxygenated hemoglobin, a single dip at 556 nm, induced by the gingiva.

4 Discussion

One potential drawback of the presented combination of a Monte Carlo simulation running on the GPU and a ray-tracing algorithm running on the CPU is the constant memory transfer between CPU and GPU. Assuming typical optical properties of biological tissue in the red wavelength range, e.g., $\mu_a = 0.01$ mm^{-1}, $\mu_s' = 1$ mm^{-1}, the time needed to propagate all photons in the turbid media on the GPU is in general longer than the time needed to calculate the fresh IPs on the CPU and copy them to the GPU. Hence, the simulation speed is neither limited by the calculations
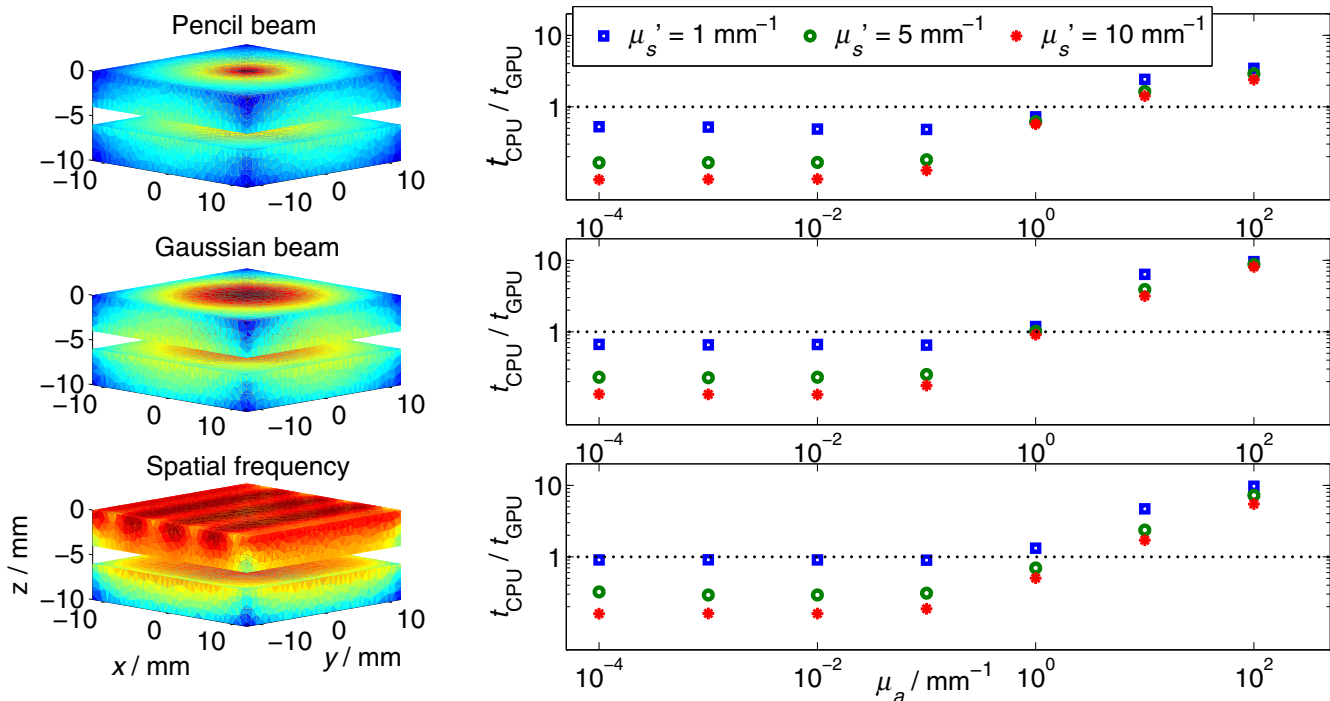
Fig. 9 Comparison of the total time spent on the CPU (t_{CPU}) versus the total time spent on the GPU (t_{GPU}) depending on three different light source types, three different reduced scattering coefficients $\mu_s' \in \{1, 5, 10\}$ mm^{-1}, and seven different absorption coefficients $\mu_a \in \{10^{-4}, 10^{-3}, \ldots, 10^1, 10^2\}$ mm^{-1}, where g was kept constant at 0.8 and n at 1.4. The object consists of 2×10^5 tetrahedra, the threshold for the Russian roulette was set to 1×10^{-4}, and the survival chance to 0.1.

of the IPs nor by the GPU-CPU transfer process for typical optical properties of biological tissue. For simulations where only a few scattering events occur before a photon is absorbed or reflected, for example, for high absorption ($\mu_a \gg \mu_s'$) or an optically thin layer, this method might reach its limits. In these scenarios, the CPU is the limiting factor where the GPU has to wait for starting positions and directions. To investigate the method's limits, the total time needed on the CPU (t_{CPU}) for the ray-tracing part is compared with the total time spent for the Monte Carlo part on the GPU (t_{GPU}) in Fig. 9. The results were generated using an NVIDIA GeForce GTX 480 and two out of eight cores of an i7-950 (3.07 GHz), where also four and eight cores were tested, which resulted on an almost linear decrease of t_{CPU}. In general, the whole software is limited by the CPU when the fraction $\tau = \frac{t_{\mathrm{CPU}}}{t_{\mathrm{GPU}}} > 1$. The different light source types represent different computation complexities to generate a starting position on the CPU. The pencil beam has the lowest computational cost, the Gaussian beam has a medium cost and the spatial frequency has the highest cost as it is based on rejection sampling. In Fig. 9 it is shown that an increasing complexity for the light source results in an increase of τ for different μ_s', μ_a combinations. A higher absorption coefficient leads to a higher probability that a photon is absorbed at each scattering event. Therefore, more photons can be propagated on the GPU within a certain time frame and consequently leading to an increasing τ for a constant scattering coefficient. The same argumentation holds if the absorption coefficient is kept constant and the scattering coefficient is increased. In this case, one photon survives on average more scattering events and less photons can be propagated on the GPU within the same time frame leading to a decreasing τ.

In previously described Monte Carlo packages that can consider anisotropic light propagation caused by cylindrical scatterers, the orientation of the cylinders is often defined by an analytical formulation, for example all cylinders are aligned parallel to one axis.[15,63] In contrast, MCtet was designed such that the user can define the concentration as well as the orientation of the cylindrical scatterers for each tetrahedron individually. Thus, the scattering coefficient as well as the step size have to be updated every time the photon is entering a tetrahedron or is changing its direction. A drawback, therefore, is an increased computational complexity and the need of slow memory access by the global memory as the cylindrical scattering phase function is too big to fit in a faster memory on the GPU. Although this is done in a very efficient way, for example, by saving the phase function in a cumulative way (see Fig. 5), the updates slow down the simulation.

5 Conclusion

A MCtet was developed to calculate the light propagation in arbitrarily shaped objects, such as a tooth. Similar to other software packages, MCtet uses the superior performance of a GPU compared with a CPU for this task. OpenCl was used for the implementation so that the workload can be spread among GPUs as well as CPUs of different vendors at the same time. A concept is introduced to easily realize different light-sources illuminating an arbitrarily shaped surface. This concept additionally allows the software to consider photons that leave a concave object and potentially reenter at a different location. It is based on treating photons traveling through air with the fast ray-tracing kernel Embree on the CPU and the photons propagating in turbid media on the GPU. Both tasks run parallel so that the whole computational power of a computer is leveraged. A key

improvement for the performance presented in this paper is the explicitly described, efficient way to parallelize the CPU and GPU. Efficiency is hereby concerning the low memory overhead as well as the asynchronous way to transfer data between the CPU and GPU. Therefore, the core MC simulation on the GPU is in general neither restricted by the CPU-GPU communication nor by the memory demand.

A correct implementation of MCtet was shown by comparison with five other Monte Carlo programs. A speed improvement compared with CPU based Monte Carlo programs was found as expected. MCtet showed a hundredfold speed improvement compared with MCML. Compared with the GPU based implementation of MCML (CUDAMCML), a four up to five times slower speed was recorded, which can be explained by the high specification and therefore restriction of CUDAMCML to layered media.

MCtet is also capable to efficiently simulate anisotropic light propagation caused by aligned structures, like tubules in a human tooth. In this paper, transmission spectra through a human incisor tooth were simulated, where a morphological correct tooth mesh, based on cone beam computed tomography of a specimen, was used. The important influence of the tubules on the propagation of light in aligned scattering media was showed by comparing simulated transmission spectra through a human incisor tooth for a case, where the tubules were taken into account and a corresponding case where only HG scattering was present. It was found that a significant sensitivity of the transmission spectrum to a change of oxygen saturation is only present for the case where the anisotropic light propagation in dentin is considered. This finding was explained by the light-guiding effect caused by the tubules in the dentin as it can be seen in the calculated fluence. Hence, it is essential to take the tubules into account when simulating light propagation in a tooth, for example, to understand and optimize pulse oximetry. Another, less pronounced, light-guiding effect in enamel was described. This effect is explained by a combination of the low scattering and high refractive index of enamel. The result of this effect is a higher fluence in enamel, whereby the light is guided around the tooth.

Disclosures

Acknowledgments

References

1. T. Aoyagi, "Pulse oximetry: its invention, theory, and future," *J. Anesth.* **17**(4), 259–266 (2003).
2. B. Schulz et al., "Experimental fluorescence tomography of tissues with noncontact measurements," *IEEE Trans. Med. Imaging* **23**(4), 492–500 (2004).
3. L. V. Wang and S. Hu, "Photoacoustic tomography: in vivo imaging from organelles to organs," *Science* **335**(6075), 1458–1462 (2012).
4. D. A. Boas et al., "Imaging the body with diffuse optical tomography," *IEEE Signal Process. Mag.* **18**(6), 57–75 (2001).
5. G. Zonios et al., "Skin melanin, hemoglobin, and light scattering properties can be quantitatively assessed in vivo using diffuse reflectance spectroscopy," *J. Invest. Dermatol.* **117**(6), 1452–1457 (2001).
6. H. Arimoto et al., "Depth profile of diffuse reflectance near infrared spectroscopy for measurement of water content in skin," *Skin Res. Technol.* **11**(1), 27–35 (2005).

7. V. Gopikrishna, K. Tinagupta, and D. Kandaswamy, "Comparison of electrical, thermal, and pulse oximetry methods for assessing pulp vitality in recently traumatized teeth," *J. Endod.* **33**(5), 531–535 (2007).

8. A. Lussi, R. Hibst, and R. Paulus, "DIAGNOdent: an optical method for caries detection," *J. Dent. Res.* **83**(Suppl. 1), 80–83 (2004).

9. J. M. Schnettler and J. A. Wallace, "Pulse oximetry as a diagnostic tool of pulpal vitality," *J. Endod.* **17**(10), 488–490 (1991).

10. A. Munshi, A. Hegde, and S. Radhakrishnan, "Pulse oximetry: a diagnostic instrument in pulpal vitality testing," *J. Clin. Pediatr. Dent.* **26**(2), 141–145 (2003).

11. I. V. Meglinsky and S. J. Matcher, "Modelling the sampling volume for skin blood oxygenation measurements," *Med. Biol. Eng. Comput.* **39**(1), 44–50 (2001).

12. A. Hohmann and H. J. Foth, "Simulations of light propagation and heat conduction during laser root canal sterilization," in *The Finite Element Method in Biomedical Engineering, Biomechanics and Related Fields, FEM-Workshop 2005*, Ulm University, pp. 101–107 (2005).

13. W. Seka et al., "Light deposition in dental hard tissue and simulated thermal response," *J. Dent. Res.* **74**, 1086–1092 (1995).

14. Y. Chen, J. L. Ferracane, and S. A. Prahl, "A pilot study of a simple photon migration model for predicting depth of cure in dental composite," *Dent. Mater.* **21**(11), 1075–1086 (2005).

15. A. Kienle et al., "Light propagation in dentin: influence of microstructure on anisotropy," *Phys. Med. Biol.* **48**(2), N7 (2002).

16. J. W. Wiscombe, "Improved Mie scattering algorithms," *Appl. Opt.* **19**(9), 1505–1509 (1980).

17. R. J. Wait, "Scattering of a plane wave from a circular dielectric cylinder at oblique incidence," *Can. J. Phys.* **33**(5), 189–195 (1955).

18. A. Taflove and S. C. Hagnesss, *Computational Electrodynamics: The Finite-Difference Time-Domain Method*, Artech House, Norwood (2005).

19. B. Krueger, T. Brenner, and A. Kienle, "Solution of the inhomogeneous Maxwell's equations using a Born series," *Opt. Express* **25**(21), 25165–25182 (2017).

20. S. Chandrasekhar, *Radiative Transfer*, Courier Corporation, North Chelmsford (2013).

21. L. G. Henyey and J. L. Greenstein, "Diffuse radiation in the galaxy," *Astrophys. J.* **93**, 70–83 (1941).

22. A. Liemert and A. Kienle, "Exact and efficient solution of the radiative transport equation for the semi-infinite medium," *Sci. Rep.* **3**, 1–7 (2013).

23. A. Liemert, D. Reitzle, and A. Kienle, "Analytical solutions of the radiative transport equation for turbid and fluorescent layered media," *Sci. Rep.* **7**, 1–9 (2017).

24. A. S. Prahl et al., "A Monte Carlo model of light propagation in tissue," *Dosim. Laser Radiat. Med. Biol.* **5**, 102–111 (1989).

25. T. S. Flock et al., "Monte Carlo modeling of light propagation in highly scattering tissues. I. Model predictions and comparison with diffusion theory," *IEEE Trans. Biomed. Eng.* **36**(12), 1162–1168 (1989).

26. L. Wang, S. L. Jacques, and L. Zheng, "MCML-Monte Carlo modeling of light transport in multi-layered tissues," *Comput. Methods Progr. Biomed.* **47**(2), 131–146 (1995).

27. H. Shen and G. Wang, "A tetrahedron-based inhomogeneous Monte Carlo optical simulator," *Phys. Med. Biol.* **55**(4), 947–962 (2010).

28. H. Shen and G. Wang, "A study on tetrahedron-based inhomogeneous Monte Carlo optical simulation," *Biomed. Opt. Express* **2**(1), 44–57 (2011).

29. Q. Fang, "Mesh-based Monte Carlo method using fast ray-tracing in Pluecker coordinates," *Biomed. Opt. Express* **1**(1), 165–175 (2010).

30. J. N. Bixler et al., "Methods for variance reduction in Monte Carlo simulations," *Proc. SPIE* **9706**, 970612 (2016).

31. O. Yang and B. Choi, "Accelerated rescaling of single Monte Carlo simulation runs with the graphics processing unit (GPU)," *Biomed. Opt. Express* **4**(11), 2667–2672 (2013).

32. 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).

33. 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).

34. E. Alerstam et al., "Next-generation acceleration and code optimization for light transport in turbid media using GPUs," *Biomed. Opt. Express* **1**(2), 658–675 (2010).

35. R. Watte et al., "Modeling the propagation of light in realistic tissue structures with MMC-fpf: a meshed Monte Carlo method with free phase function," *Opt. Express* **23**(13), 17467–17486 (2015).

36. 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).

37. N. Ren et al., "GPU-based Monte Carlo simulation for light propagation in complex heterogeneous tissues," *Opt. Express* **18**(7), 6811–6823 (2010).

38. Y. Shin and H. Kwon, "Mesh-based Monte Carlo method for fibre-optic optogenetic neural stimulation with direct photon flux recording strategy," *Phys. Med. Biol.* **61**(6), 2265–2282 (2016).

39. R. Yao, X. Intes, and Q. Fang, "Generalized mesh-based Monte Carlo for wide-field illumination and detection via mesh retessellation," *Biomed. Opt. Express* **7**(1), 171–184 (2016).

40. H. Si, "TetGen, a Delaunay-based quality tetrahedral mesh generator," *ACM Trans. Math. Software* **41**(2), 1–36 (2015)

41. J. E. Stone, D. Gohara, and G. Shi, "OpenCL: a parallel programming standard for heterogeneous computing systems," *Comput. Sci. Eng.* **12**(3), 66–73 (2010).

42. I. Wald et al., "Embree: a kernel framework for efficient CPU ray tracing," *ACM Trans. Graphics* **33**(4), 1–8 (2014).

43. Intel Corporation, "Embree overview," https://embree.github.io/ (2018).

44. S. J. Nelson, *Wheeler's Dental Anatomy, Physiology and Occlusion*, 10th ed., Elsevier Saunders (2015).

45. A. R. Helfer, S. Melnick, and H. Schilder, "Determination of the moisture content of vital and pulpless teeth," *Oral Surg. Oral Med. Oral Pathol.* **34**(4), 661–670 (1972)

46. D. Fried et al., "Nature of light scattering in dental enamel and dentin at visible and near-infrared wavelengths," *Appl. Opt.* **34**(7), 1278–1285 (1995).

47. D. Spitzer and J. J. Ten Bosch, "The absorption and scattering of light in bovine and human dental enamel," *Calcif. Tissue Res.* **17**(2), 129–137 (1975).

48. J. Zijp, "Optical properties of dental hard tissues," Doctoral dissertation, Universtity of Groningen (2001).

49. J. R. Zijp and J. Jaap, "Theoretical model for the scattering of light by dentin and comparison with measurements," *Appl. Opt.* **32**(4), 411–415 (1993).

50. P. Zaslansky, S. Zabler, and P. Fratzl, "3D variations in human crown dentin tubule orientation: a phase-contrast microtomography study," *Dent. Mater.* **26**(1), e1–e10 (2010).

51. S. L. Jacques, "Corrigendum: optical properties of biological tissues: a review," *Phys. Med. Biol.* **58**(14), 5007–5008 (2013).

52. A. N. Bashkatov, E. A. Genina, and V. V. Tuchin, "Optical properties of skin, subcutaneous, and muscle tissues: a review," *J. Innovative Opt. Health Sci.* **4**(1), 9–38 (2011).

53. A. Roggan et al., "Optical properties of circulating human blood in the wavelength range 400–2500 nm," *J. Biomed. Opt.* **4**(1), 36–47 (1999).

54. R. Clement et al., "Quasi-automatic 3D finite element model generation for individual single-rooted teeth and periodontal ligament," *Comput. Methods Progr. Biomed.* **73**(2), 135–144 (2004).

55. A. Boryor et al., "A downloadable meshed human canine tooth model with PDL and bone for finite element simulations," *Dent. Mater.* **25**(9), e57–e62 (2009).

56. A. Kienle et al., "Light propagation in dry and wet softwood," *Opt. Express* **16**(13), 9895–9906 (2008).

57. S. R. Marschner et al., "Light scattering from human hair fibers," *ACM Trans. Graphics* **22**(3), 780 (2003).

58. C. F. Bohren and D. R. Huffman, *Absorption and Scattering of Light by Small Particles*, John Wiley & Sons (2008).

59. S. Jacques, "MCxyz," https://omlc.org/software/mc/mcxyz/index.html (2018).

60. A. Doronin and I. Meglinski, "Online object oriented Monte Carlo computational tool for the needs of biomedical optics," *Biomed. Opt. Express* **2**(9), 2461–2469 (2011).

61. Biophotonics group Oulu, "MCOnline," www.lighttransport.net (2018).

62. S. Jacques and S. Prahl, "OMLC," https://omlc.org/index.html (2018).

63. T. Yun et al., "Monte Carlo simulation of polarized photon scattering in anisotropic media," *Opt. Express* **17**(19), 16590–16602 (2009).

Biographies for the authors are not available.