

MCMLpar and MCSLinv: A Parallel Version of MCML and an Inverse Monte Carlo Algorithm to Calculate Optical Scattering Parameters

Richelle H. Streater^{1,2}, Anne-Michelle R. Lieberson^{1,3}, Adam L. Pintar¹, and Zachary H. Levine¹

¹National Institute of Standards and Technology,
Gaithersburg, MD 20899, USA

²Colorado School of Mines,
Golden, CO 80401, USA

³Sherwood High School,
Sandy Spring, MD 20869, USA

adam.pintar@nist.gov
zlevine@nist.gov

MCMLpar DOI: <https://doi.org/10.18434/M3NM2G>

MCSLinv DOI: <https://doi.org/10.18434/M3HW9N>

Software Version: 1.0

Key words: biomedical optics; importance sampling; inverse; Monte Carlo; scattering parameters.

Accepted: September 15, 2017

Published: October 20, 2017

<https://doi.org/10.6028/jres.122.038>

1. Summary

Our recent research on Monte Carlo simulation of light propagation in biomedical phantoms has involved the implementation of an algorithm to solve for the optical attenuation and scattering coefficients of a single-layer material, or μ_a and μ_s [1]. It also involved the development of a parallel version of MCML [2], a popular light propagation model with over 2100 citations, according to ScienceDirect. The algorithm for the inverse problem involves three components: a profile log-likelihood evaluation, importance sampling, and an optimization procedure. In this article, we present two C++/OpenMP codes: MCMLpar, the parallel version of MCML, and MCSLinv, a program to solve the inverse problem for a single layer.

In **MCMLpar**, [3] a particle class is introduced to represent photons. The particles all enter an optical medium at the same point, travelling parallel to the vector normal to the surface. The particle weights are initially set to the probability of transmitting into the material, calculated by the Fresnel equation using the air and first layer indices of refraction. Each particle's path length to an interaction is calculated by sampling the exponential distribution with mean $\mu_t = \mu_a + \mu_s$. If the interaction is within the layer, the particle's direction is updated by sampling the Henyey-Greenstein distribution for the polar angle θ relative to the direction of travel. The relative azimuthal angle ϕ is sampled uniformly on $[0, 2\pi)$. The particle's

How to cite this article:

1 Streater RH, Lieberson A-MR, Pintar AL, Levine ZH
(2017) MCMLpar and MCSLinv: A Parallel Version of
MCML and an Inverse Monte Carlo Algorithm to Calculate
Optical Scattering Parameters. *J Res Natl Inst Stan* 122:38.
<https://doi.org/10.6028/jres.122.038>

weight is then multiplied by $\mu_s/(\mu_s+\mu_a)$ to account for the chance that the interaction would have been attenuated. On the other hand, if the original path length would move the particle beyond the layer, the program determines whether to transmit or reflect the particle by sampling a Bernoulli distribution based on the Fresnel equations, averaging over polarizations. A transmission event changes the particle's direction with Snell's law. After scattering, transmitting, or reflecting, the particle propagates again with a new path length, and the process repeats until the particle leaves the medium.

The particle leaves the material by transmitting outside of the sample. When this occurs, the particle weight is added to a scoring vector and a new particle is introduced. Additionally, several particles can be sent through the material at once because the code runs in parallel. If the particle's weight passes below a threshold before the particle escapes, the roulette method determines whether the particle lives or dies [2].

MCSLinv [4] uses a profile log-likelihood to compare the scoring vectors from a forward simulation using the method of MCMLpar to experimental Angle-Resolved-Scattering (ARS) data. The ARS is given by

$$ARS = \frac{P_s}{\Omega P_i}$$

where P_i is the incident power measured by the detector when no sample is present and P_s is the power scattered from the area of the sample included in the detector field of view at angles within the solid angle defined by $\Omega = A/R^2$. In addition, MCMLpar may be used to create simulated data which can be read by MCSLinv. The log-likelihood is used to construct confidence regions by identifying all parameter pairs leading to a log-likelihood value within half of a χ^2 percentile of the maximum log-likelihood value [2]. Photon interaction events are sampled with Monte Carlo in a standard way for reference values of the parameters. These are extended to a 1D grid for each type of interaction (e.g., propagation or scattering) using importance sampling. Because the sampling distributions are independent, the 2D grid is only created explicitly as the particle is scored. Calculating the likelihood for each parameter's total summed weights yields a surface of likelihood values, which is smooth and has a single peak.

The algorithm reduces the parameter search space at each iteration. The program initially runs the importance sampling simulation with a low number of particles (20 000 is typical) over a large range of parameters. The range of parameters is updated to only include values around the likelihood peak within some confidence region (99.9999 % is typical). The importance simulation runs again with twice as many particles and the smaller range of parameters. This repeats a user-defined number of times (search intervals typically converge within 7 iterations). At some point, the confidence regions are dominated by experimental error and additional Monte Carlo samples do not affect the results.

The program outputs a file with the final maximum likelihood estimates for μ_a and μ_s . It also outputs the coefficients of a paraboloidal fit for the likelihood surface, which a provided Mathematica notebook uses to graph an elliptical confidence region around the μ_a and μ_s maximum likelihood estimates.

2. Software Specifications

NIST Operating Unit(s)	Physical Measurement Laboratory, Information Technology Laboratory
Category	Monte Carlo light propagation simulations for turbid media
Targeted Users	Researchers interested in measuring optical attenuation and scattering coefficients for a material
Operating System(s)	Cross-platform. MCMLpar requires installation of SPRNG 2.0b or SPRNG 5.0 [5]. MCSLinv requires installation of SPRNG 2.0b or SPRNG 5.0 and Eigen [6]. SPRNG 2.0b is available and recommended for Windows [7], while SPRNG 5.0 is recommended for Linux.
Programming Language	C++ (at least 2011), Mathematica 10.4 for post-processing.
Inputs/Outputs	MCMLpar: One input text file, one output csv file which may be read by MCSLinv. MCSLinv: Two input text files, one output csv file. Described in README.txt
Documentation	Documentation provided in Refs. [1,3,4]
Accessibility	N/A small-scale research tool
Disclaimer	https://www.nist.gov/director/licensing

3. Methods

The software includes a Monte Carlo simulation and an inverse problem algorithm. The Monte Carlo simulation has analytic solutions when the absorption and scattering coefficients are set to almost zero, as well as in single-scattering tests. The numerical and analytic solutions were compared in each of these cases, and the results are summarized in a Mathematica notebook (which is also saved as a PDF). The Monte Carlo simulation was also compared to MCML for various sets of parameters, and completed the simulations with comparable speed and results. MCSLinV was tested with experimental data, and with a “ground truth” forward run. The code returned the “ground truth” parameter values within uncertainty, and the search region converged as expected for various levels of noise in the inputs. The experimental data trial yielded a forward ARS result that matched the real experimental data closely.

Acknowledgments

We would like to acknowledge Catherine Cooksey and Paul Lemailet for providing experimental Angle-Resolved Scattering data. We would also like to acknowledge Steven Conn for his help with the software installation.

4. References

- [1] Levine ZH, Streater RH, Lieberson A-MR, Pintar AL, Cooksey CC, Lemailet P (2017) Algorithm for rapid determination of optical scattering parameters. *Optics Express* 25(22): 26728-26746. <https://doi.org/10.1364/OE.25.026728>
- [2] Wang L, Jacques SL, Zheng L (1995) MCML—Monte Carlo modeling of light transport in multi-layered tissues. *Computer Methods and Programs in Biomedicine* 47(2):131-146. [https://doi.org/10.1016/0169-2607\(95\)01640-F](https://doi.org/10.1016/0169-2607(95)01640-F).
- [3] Streater RH, Lieberson AM, Pintar AL, Levine ZH (4 Aug 2017) *MCMLpar: Parallelized Monte Carlo: Multi Layer*. Available at <https://github.com/usnistgov/MCMLpar>. Accessed 21 September 2017.
- [4] Streater RH, Lieberson AM, Pintar AL, Levine ZH (18 September 2017) *MCSLinV: Inverse Monte Carlo Algorithm*. Available at <https://github.com/usnistgov/MCSLinV>. Accessed 21 September 2017.
- [5] Mascagni M, Srinivasan A (2000) Algorithm 806: SPRNG: a scalable library for pseudorandom number generation. *ACM Transactions on Mathematical Software* 26(3):436-461. <https://doi.org/10.1145/358407.358427>.
- [6] Guennebaud G, Jacob B (2010) *Eigen v3*. Available at <http://eigen.tuxfamily.org>. Accessed 28 July 2017.
- [7] NAADSM Development Team (2013) *SPRNG for Microsoft Windows and NAADSM*. Available at <http://www.naadsm.org/opensource/sprng>. Accessed 28 July 2017.

About the authors: *Richelle Streater is a Summer Undergraduate Research Fellow at NIST and an undergraduate student at Colorado School of Mines. Anne-Michelle Lieberson is a Summer Intern at NIST and a student at Sherwood High School. Adam Pintar is a Statistician at NIST, specializing in Bayesian and maximum likelihood methods. Zachary Levine is a Physicist at NIST and a Fellow of the American Physical Society. His work has concentrated on the interaction of light and other probes with matter. The National Institute of Standards and Technology is an agency of the U.S. Department of Commerce.*