



LUND UNIVERSITY

Multi--Scattering: Computational light transport in turbid media

Jönsson, Joakim

2021

Document Version:

Publisher's PDF, also known as Version of record

[Link to publication](#)

Citation for published version (APA):

Jönsson, J. (2021). *Multi--Scattering: Computational light transport in turbid media*. [Doctoral Thesis (compilation), Combustion Physics]. Department of Physics, Lund University.

Total number of authors:

1

General rights

Unless other specific re-use rights are stated the following general rights apply:

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Read more about Creative commons licenses: <https://creativecommons.org/licenses/>

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

LUND UNIVERSITY

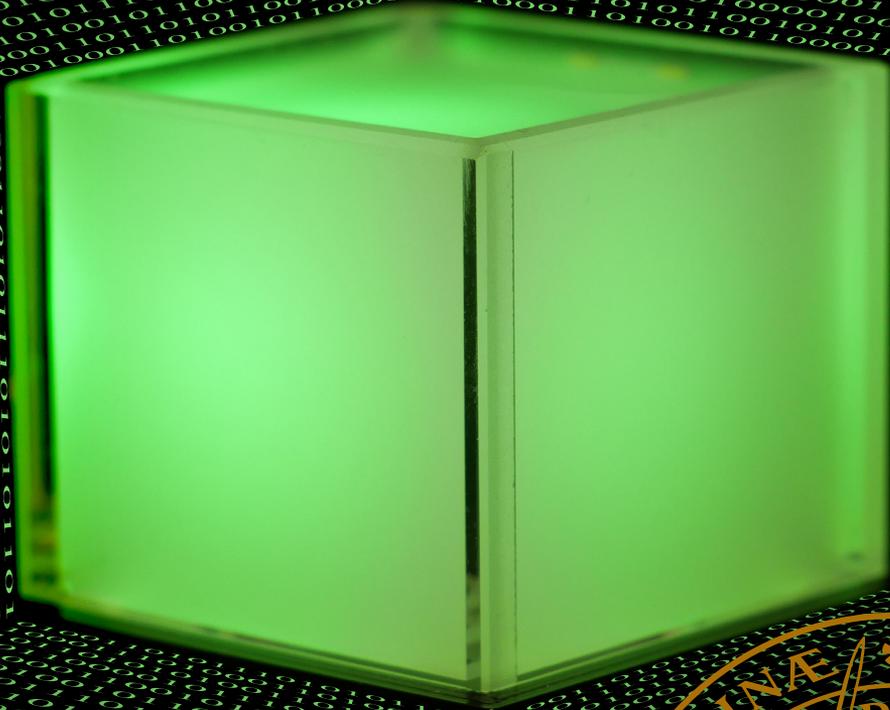
PO Box 117
221 00 Lund
+46 46-222 00 00

Multi-Scattering

Computational light transport in turbid media

JOAKIM JÖNSSON

COMBUSTION PHYSICS | DEPARTMENT OF PHYSICS | LUND UNIVERSITY



Multi-Scattering

Multi-Scattering

Computational light transport in turbid media

by Joakim Jönsson



LUND
UNIVERSITY

Thesis for the degree of Doctor of Philosophy
Thesis main advisor: Dr. Edouard Berrocal
Thesis co-advisor: Dr. Elias Kristensson
Faculty opponent: Prof. Matthias Hullin

To be presented, with the permission of the Faculty of Engineering of Lund University, for public criticism in the Rydberg lecture hall (Rydbergsalen) at the Department of Physics on Thursday, the 17th of June 2021 at 13:15.

Organization LUND UNIVERSITY Division of Combustion Physics Department of Physics Box 118, SE-221 00 LUND Sweden		Document name DOCTORAL THESIS	
		Date of disputation 2021-06-17	
Author(s) Joakim Jönsson		Sponsoring organization ERC Starting Grant nb. 638546	
Title and subtitle <i>Multi-Scattering: Computational light transport in turbid media</i>			
Abstract <p>This thesis presents and describes the development of an online freely accessible software called <i>Multi-Scattering</i> for the computational modeling of light propagation in scattering and absorbing media. The model is based on the use of the Monte Carlo method, where billions of photon packets are being launched and tracked through simulated cubic volumes. The software also includes features for modeling image formation by inserting a virtual collecting lens and a detection matrix which simulate a camera objective and a sensor array respectively. In addition, the Lorenz-Mie theory is integrated to generate the scattering phase functions from spherical particles. The model has been accelerated by means of general-purpose computing on graphics processing units, reducing the computation time by a factor up to 200x in comparison with a single CPU thread. By using four graphic cards on a single computer, the simulation speed increases by a factor of 800x. With an anisotropy factor $g = 0.86$, the transport path of one billion photons can be computed in 10 seconds for optical depth $OD = 10$ and in 20 minutes for $OD = 500$.</p> <p>The simulations are running from a computer server at Lund University, allowing researchers to login and use it freely without any need for programming skills or specific software/hardware installations. There are countless types of scattering media in which this model can be used to predict photon transport, including medical tissues, blood samples, clouds, smoke, fog, turbid liquids, spray systems, <i>etc.</i> In this thesis, the software has been used for a variety of scattering situations and to simulate photon transport: 1) inside a portion of a human head, 2) within atomizing spray systems, 3) in controlled aqueous dispersion of polystyren spheres, 4) for time-of-flight measurements in intralipid solutions and 5) for Diffuse Correlation Spectroscopy applications. Finally, the numerical results have been validated by rigorously comparing the simulated results with experimental data. The user interface for both setting-up a simulation and displaying the corresponding results is found at: https://multi-scattering.com</p>			
Key words Monte Carlo simulations, GPU, CUDA, particulate scattering, scattering in tissues			
Classification system and/or index terms (if any)			
Supplementary bibliographical information		Language English	
ISSN and key title 1102-8718		ISBN 978-91-7895-897-9 (print) 978-91-7895-898-6 (pdf)	
Recipient's notes		Number of pages 204	Price
		Security classification	

I, the undersigned, being the copyright owner of the abstract of the above-mentioned dissertation, hereby grant to all reference sources the permission to publish and disseminate the abstract of the above-mentioned dissertation.

Signature



Date 2021-05-06

Multi-Scattering
Computational light transport
in turbid media

by Joakim Jönsson



LUND
UNIVERSITY

Cover illustration front: A photo of a scattering medium by Joakim Jönsson

Funding information: This thesis work was financially supported by the H2020 European Research Council through the ERC Starting Grant nb. 638546 titled “Spray-Imaging”.

© Joakim Jönsson 2021

Faculty of Engineering, Division of Combustion Physics
Department of Physics

ISBN: 978-91-7895-897-9 (print)

ISBN: 978-91-7895-898-6 (pdf)

ISSN: 1102-8718

ISRN: LUTFD2/TFCP-233-SE

Printed in Sweden by Media-Tryck, Lund University, Lund 2021



Media-Tryck is a Nordic Swan Ecolabel certified provider of printed material. Read more about our environmental work at www.mediatryck.lu.se

MADE IN SWEDEN 

Super Bien Joue Ed !!!
Multi-Scattering

Contents

List of publications	iii
Acknowledgements	v
Popular summary	vii
Populärvetenskaplig sammanfattning	ix
Multi-Scattering: Computational light transport in turbid media	1
1 Introduction	3
2 Monte Carlo simulation of photon transport	9
2.1 General description	9
2.2 Radiative Transfer Equation	12
2.3 History of Monte Carlo simulation for light propagation in scattering media	13
3 Definition and terminology	17
3.1 Conversion from photon counts to irradiance	17
3.2 Extinction, scattering and absorption	18
3.3 Scattering phase function and anisotropy factor	20
4 Multi-Scattering: Successive steps and algorithm	23
4.1 General simulation algorithm	23
4.2 Launching photon packets	24
4.3 Determination of the free path length	26
4.4 New photon weight	27
4.5 New photon direction	27
4.6 Photon crossing voxels and exiting the scattering volume	29
4.7 Photon detection and image formation	30
4.8 Libraries used in <i>Multi-Scattering</i>	34
5 Accelerated computing with modern hardware	37
5.1 CUDA and modern GPU's architecture	37
5.2 Optimizing the code for running on the GPU	39
5.3 Generation of random numbers	40
5.4 Optimized light source matrix	41

5.5	Accelerated simulation speed	43
6	Experimental validation	47
6.1	Comparison for the light intensity distribution	47
6.1.1	Experimental method	47
6.1.2	Simulation method	49
6.1.3	Results	52
6.2	Comparison for time-of-flight in tissue-like phantoms	56
6.2.1	Methodology	56
6.2.2	Results	58
7	Conclusions and Outlook	61
	References	63
	Summary of papers	69
	Author contributions	69
	Paper I: <i>Multi-Scattering</i> software: part I: online accelerated Monte Carlo simulation of light transport through scattering media	69
	Paper II: <i>Multi-Scattering</i> software: part II: experimental validation for the light intensity distribution	70
	Paper III: Comparison between two-phase and one-phase SLIPI for instantaneous imaging of transient sprays	70
	Paper IV: Simulation of light scattering and imaging of spray systems using the open-access software “Multi-Scattering”	71
	Paper V: Fast and sensitive diffuse correlation spectroscopy with highly parallelized single photon detection	71
	Paper VI: <i>Multi-Scattering</i> software: part III: Comparison with experimental results for photon time-of-flight in tissue mimicking phantoms”	71
	Paper I: <i>Multi-Scattering</i> software: part I: online accelerated Monte Carlo simulation of light transport through scattering media	73
	Paper II: <i>Multi-Scattering</i> software: part II: experimental validation for the light intensity distribution	103
	Paper IV: Comparison between two-phase and one-phase SLIPI for instantaneous imaging of transient sprays	123
	Paper IV: Simulation of light scattering and imaging of spray systems using the open-access software “Multi-Scattering”	143
	Paper V: Fast and sensitive diffuse correlation spectroscopy with highly parallelized single photon detection	153
	Paper VI: <i>Multi-Scattering</i> software: part III: Comparison with experimental results for photon time-of-flight in tissue mimicking phantoms”	175

List of publications

- I ***Multi-Scattering* software: part I: online accelerated Monte Carlo simulation of light transport through scattering media**
J. Jönsson, E. Berrocal
Optics Express 28, 37612-37638 (2020).
<https://doi.org/10.1364/OE.404005>
- II ***Multi-Scattering* software: part II: experimental validation for the light intensity distribution**
D. Frantz, J. Jönsson and E. Berrocal
Optics Express, (2021). (*to be submitted*)
- III **Comparison between two-phase and one-phase SLIPI for instantaneous imaging of transient sprays**
Y. N. Mishra, E. Kristensson, M. Koegl, J. Jönsson, L. Zigan and E. Berrocal,
Experiments in Fluids 58, 110 (2017).
<https://doi.org/10.1007/s00348-017-2396-9>
- IV **Simulation of light scattering and imaging of spray systems using the open-access software “Multi-Scattering”**
J. Jönsson, E. Berrocal
Proceeding of the 29th ILASS-Europe, Paris - France, (2019).
<https://ilass19.sciencesconf.org/246310>
- V **Fast and sensitive diffuse correlation spectroscopy with highly parallelized single photon detection**
W. Liu, R. Qian, S. Xu, P. C. Konda, M. Harfouche, D. Borycki, J. Jönsson, E. Berrocal, C. L. Cooke, H. Wang, Q. Dai and R. Horstmeyer
APL Photonics 6, 026106 (2021).
<https://doi.org/10.1063/5.0031225>
- VI ***Multi-Scattering* software: part III: Comparison with experimental results for photon time-of-flight in tissue mimicking phantoms”**
B. Angeli, J. Jönsson, C. Xu, N. Reistad and E. Berrocal
Optics Express, (2021). (*manuscript in preparation*)

All papers are reproduced with permission of their respective publishers.

Acknowledgements

My journey as a PhD student is coming to an end and I would therefore like to take this opportunity to share some of my gratitude towards my colleagues, my friends and my family.

First of all, I would like to thank both *Edouard Berrocal*, my main supervisor and *Elias Kristensson*, my co-supervisor. They have both helped me in my work by sharing their endless amount of inspiration with me. Thank you Edouard for all the patience, knowledge and support and to help me grow as a scientist. Without your assistance and attention to detail I would never have been able to reach this far. Through the years we have made so many small adjustments and changes, but in the end, the results have always been perfect. Thank you Elias for reading through my thesis and all the encouraging and helpful comments.

Having a position as a PhD student have allowed me to meet many new friends from almost an equal amount of countries. To experience their cultures through them has been a great pleasure during my time at the division. I would like to thank all my colleagues at the division for bringing so much friendliness to our workplace. Sitting here writing this text I know I will not be able to mention all of the persons I have had the honor to meet but be assured, I appreciated getting to know each and every one of you. Of course I would like to extend a big thanks to *Christoffer Pichler* whom I “almost” shared an office with. I enjoyed all our talks during breaks discussing everything from books, programming and politics. Through our time at the division I have come to see you as one of my best friends. I would also like to thank both *Christian Brackmann* and *Samuel Jansson* whom I actually shared an office with. I thank both of you for the time and to Samuel I would like to add: I think we managed to clear out some of the junk that was in the office but I am pretty sure some of it is still there even to this day.

I would like to extend a special thank you to *Panagiota Stamatoglou* and *Alexios Matamis* for their hospitality and for the endless supply of happiness at the division. A special thank you to *Xin Liu*, *Haisol Kim*, *Manu Mannazhi* and *Arman Subash* for bringing your hospitality and sharing your friendship with me and the people around you. I would like to thank *Dina Hot* for bringing the division together and *Yupan Bao* for the occasional short chat in the corridors.

I would like to thank *Miaoxin Gong*, *Ali Hosseinnia*, *Wubin Weng*, *Chengdong Kong* and many, many more. As I wrote earlier this list could be extended much longer but for now I will finish by mentioning some of the people that are part of the foundation of the division. I would like to thank *Minna Ramkull* and *Cecilia Bille* for the amazing amount of work you do to keep the division rolling. I would like to thank *Marcus*

Aldén for sharing your home with us. Not only the division itself but also by extending your hospitality to your private home before each summer. I would like to thank *Per-Erik Bengtsson* for the endless inspiration and chats about exercises (with the perfect amount of competitiveness mixed in).

I would like to thank my many users of *Multi-Scattering*. I would especially like to thank my two most devoted users: *David Frantz* and *Brian Angeli*. Thank you for bringing me all the extra work of implementing new features and fixing all the odd bugs you found. David, your help with the computer exercises we supervised was invaluable to me and greatly appreciated. I would like to thank *Yogeshwar Nath Mishra* for his hospitality and for bringing *Multi-Scattering* one step forward. I would like to thank *Adrian Roth* for your valuable contributions. I would like to thank *Shiqi Xu* and *Roarke Horstmeyer* for giving me the opportunity of pushing *Multi-Scattering* into new realms of possibilities.

A special thank you to the reader reading this sentence. Thank you for your interest in the topic of light transport and my thesis.

I would like to thank my family and relatives for bringing me this far in life. I would especially like to thank my brother for sharing his recipe of appreciating life and for all the fun moments we shared together.

Finally I would like to thank *Mari Hara* for sharing a great adventure together with me.

Popular summary

Humans have always been fascinated by the idea of seeing through different materials and obstacles. Seeing through turbid media such as fog, mist or clouds (see Fig. 1) can, for example help people orienting themselves and avoid dangers, as well as enabling clear communications. Clouds can for instance hide the sky and flying objects (e.g. a plane), conceal the ground when perceived from above (e.g. viewed from a satellite) and shroud mountains. Mist and fog can limit the view of a driver and can easily disturb the flow of traffic. While those natural phenomena reduce visibility on a cloudy day or in foggy weather conditions, they can also be found at much smaller scales and in completely different places. This is why the study of light scattering is important for a broad range of fields, from industrial to medical applications. Those applications are not only limited to the concept of visibility. Some of them are focusing on the measurement of quantities (sizing droplets and particles) while others are aiming at detecting and treating diseases (finding and destroying malignant tumors).

Turbid media are challenging to study due to unwanted effects caused by multiple scattering. The analysis of blurred images, the laser probing of atmospheric layers, the development of new optical diagnostics of fuel sprays and the improvement of optical imaging of human tissues can only be performed if light propagation through such complex media is understood. This can be done through the numerical modeling of light propagation using computer simulations.

In past few decades great leaps in computer hardware have been made. This has led to a vast increase in excess computational power. As a result it has spurred interest among researchers in the development of computer methods and tools to tackle more and more complex problems. Light scattering is a typical example of such topic.

In this thesis a new computational model is presented, which is able to simulate the propagation of billions of photons through turbid scattering media. The software is called *Multi-Scattering* and it is capable of image formation, how long light is propagating over time. It can help also help quantifying measurement errors. It has been developed to be as user friendly and versatile as possible. It is based on the Monte Carlo method where the trajectory of each photon is governed by a series of random events. The software has been validated by comparison with experimental results and it is an open access tool that can be used online by anyone: <https://multi-scattering.com>

Finally this computational tool has been applied for: 1) Simulating light scattering in spray systems used in combustion. 2) Verifying and improving the performance of optical instruments such as Structured Laser Planar Imaging (SLIPI). 3) Detecting blood flow in human heads. 4) Quantifying the energy deposition for detecting and treating tumors by tracking photons inside tissues.

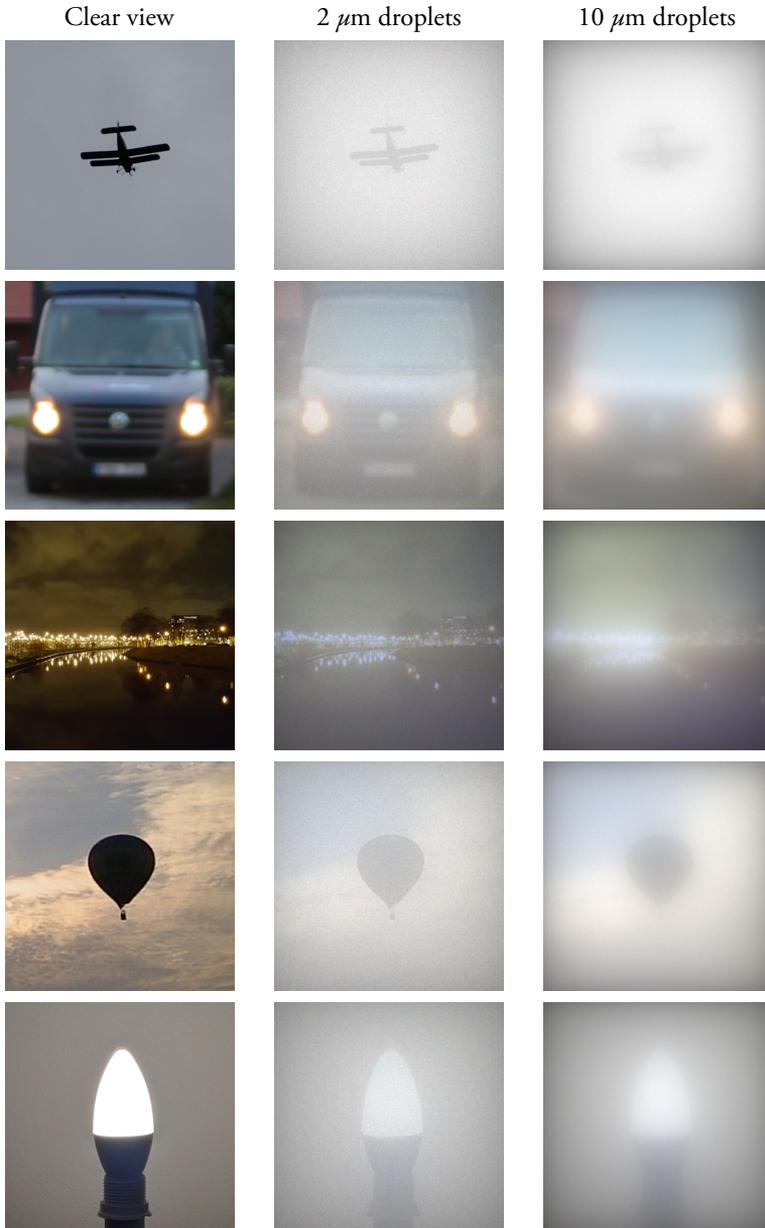


Figure 1: Examples of simulation results from *Multi-Scattering* for different situations. The clear view images on the left have been used as the input light source in the software. Then, the simulation adds a cloud of water droplets in front of each light source, causing a reduction in visibility. Two simulation configurations are shown here: One considering small droplets ($2\ \mu\text{m}$) in the center and one considering larger droplets ($10\ \mu\text{m}$) on the right. From those simulated results, it is deduced that at equal probability of light/droplet interaction, larger droplets are responsible for stronger blurring effects.

Populärvetenskaplig sammanfattning

Människan har alltid fascinerats av tanken på att kunna se genom olika material och föremål. Möjligheten att kunna se genom något grumligt så som dimma, dis, eller moln (se Fig. 1) kan till exempel hjälpa människor att orientera sig samt förhindra olyckor och det möjliggör tydlig och klar kommunikation. Moln kan till exempel dölja himlen och flygande föremål (till exempel flygplan), dölja marken när den ses från ovan (till exempel från en satellit) och bilda ett molntäcke över en bergstopp. Dimma och dis kan begränsade sikten för en förare och orsaka störningar i trafiken. De naturliga fenomen som orsakar den reducerade fria sikten på en molnig dag eller under disiga väderförhållanden kan även hittas på helt andra platser och förutsättningar. Det är därför som ljusets spridning är viktigt att studera inom många områden i allt från industriella till medicinska tillämpningar. Dessa tillämpningar behöver nödvändigtvis inte vara bundet till visuell information. Några tillämpningar är inriktade på att mäta kvantiteter (storleksbestämma droppar eller partiklar) och andra tillämpningsområden har som mål att lokalisera och behandla sjukdomar (hitta och avlägsna elakartade tumörer).

Att studera grumligheten i ett ljusspridande prov är problematisk på grund av de oönskade effekter som orsakas av upprepad ljusspridning. Analysen av suddiga bilder, lasersondering av atmosfäriska skikt, utvecklingen av ny optisk mätutrustning för bränslesprayer och förbättring av optisk utrustning för mänskliga vävnader kan endast möjliggöras om ljusets spridning genom sådana komplexa miljöer förstås. Detta kan åstadkommas genom numeriska beräkningsmodeller av ljuset spridning i form av datorsimuleringar.

Under de senaste decennierna har det gjorts flera stora framsteg inom datorhårdvara. Detta har lett till en kraftig ökning av överskott av beräkningskraft. Följaktligen så har forskningsintresset ökat för att utveckla och förbättra datormetoder och programvara för att på så vis kunna hantera mer och mer avancerad problematik. Att studera ljusets spridning är ett typiskt exempel på ett sådant användningsområde.

I denna avhandling presenteras en ny beräkningsmodell som kan hantera och simulera spridning av miljarder fotoner genom grumliga prover. Mjukvaran, *Multi-Scattering*, kan skapa avbildningar av simuleringarnas slutresultat samt mäta tiden det tar för ljuset att spridas. *Multi-Scattering* kan också hjälpa till för att storleksbestämma mätfel. Fokuset för dess utveckling har legat på att få programvaran både användarvänlig samt mångsidig. Den är baserad på en så kallat Monte Carlo-metod där vägen för varje foton kontrolleras genom en serie av slumpmässiga händelser. Programvaran har validerats genom jämförelse mot experimentella resultat och är fritt tillgängligt online och redo att användas av vem som helst: <https://multi-scattering.com>

Slutligen så har detta simuleringsverktyg använts för: 1) Simulering av ljusspridning i sprayer i samband med förbränning. 2) Verifiera och förbättra prestanda för optiska instrument såsom Structured Laser Planar Imaging (SLIPI). 3) Upptäcka blodflöden i människohuvud. 4) Kvantifiera energiupptagning för att på så vis kunna upptäcka och behandla tumörer genom att spåra fotoner inuti vävnader.

Multi-Scattering: Computational light transport in turbid media

Chapter 1

Introduction

Examples of scattering media are countless. They include human tissues, blood samples, turbid liquids, smoke, fog, under water environments and clouds to name only a few. Despite being widely different, they all share a common trait: they are challenging to image and analyze due to the occurrence of multiple light scattering when illuminated with visible light. In other words, photons are interacting several times with randomly distributed scattering centers. While being commonly encountered this complex phenomenon is rarely desired. It induces blurring effects in imaging, strongly reducing visibility. Most often it reaches the point where visibility is actually fully lost.

For the case of highly scattering media, the amount of single scattering in the detected signal is so limited that it becomes negligible, and consequently only the diffuse light can be used. In medicine, a variety of approaches belonging to the field of Near Infrared Spectroscopy (NIRS) are based on recording the diffuse light in order to deduce the optical properties of human tissues. Those techniques include: Continuous-Wave (CW-NIRS) [1], Frequency Domain (FD-NIRS) [2], Time Domain (TD-NIRS) [3, 4], Diffuse Correlation Spectroscopy (DSC) [5] and Gas in Scattering Media Absorption Spectroscopy (GASMAS) [6]. The common point and limitation of those techniques is that they rely on the average distance photons are propagating in the probed tissue between the source and the detector. In such case, knowledge of photon transport and absorption is very important.

Scattering media are also present in many industrial processes, especially those related to the production of beverages such as beer, milk and wine. For the production of those turbid liquids, it is important to monitor their production process. Another relevant industrial example are spray systems. For instance, atomizing sprays are widely

employed for the production of powders in the food and pharmaceutical industries (e.g. antibiotics, milk powder). They are also very important for the injection of liquid fuels in combustion engines to generate the required power used for transportation in cars, trucks, ships, planes *etc.* Clean and optimized combustion of liquid fuels (including liquid bio-fuels) requires, thus, to image and observe how atomizing sprays are formed. This requires understanding and suppressing the unwanted effects from multiple light scattering.

The most versatile and flexible way of modeling photon transport through scattering media is by means of the Monte Carlo method. The main concept behind a Monte Carlo algorithm is to break down a complex problem into a series of smaller sets of probability calculations which are repeated a large number of times. The different possibilities/paths are then sampled using random numbers and a succession of probability density functions. This random sampling process must be sufficiently repeated in order to reduce statistical fluctuations and obtain converging results. Due to the large improvements in computer performance and the trend towards the adaptation of general-purpose computing on graphics processing units (GPU), modern Monte Carlo codes can now simulate billions of photon packets within a reasonable time frame. This has led to the possibility of obtaining realistic simulations within complex three-dimensional geometries.

In this thesis a multi-purpose, GPU-accelerated, Monte Carlo software called *Multi-Scattering* is presented. The core of the code originates from the numerical Monte Carlo model written in C programming and developed by Edouard Berrocal in 2006 [7]. In contrast to multi-layer based models, this model is designed around the concept of decomposing the simulated scattering medium into multiple voxels. While originally developed to simulate light propagation through clouds of droplets such as spray systems, the model has been extensively extended and can now consider complex biomedical tissue samples. In addition, several new features have been implemented and numerous tasks have been solved, including:

1. Simulation speed-up (up to 800 times) using parallel computing.
 - CUDA programming and use of Nvidia graphics card.
 - Having multiple GPUs running a single simulation.
 - Managing resources needed for running a simulation on the GPUs efficiently.
 - Storing resulting data in a format usable both by a web-based user interface and by external software (Matlab).
2. Open access software available online at <https://multi-scattering.com>

- Web page to login and for simulation management.
- Web page to handle simulation creation.
- Web page to directly visualize the results.

3. Expansion of the model to new detection features.

- Versatile imaging configurations.
- Recording photon paths to analyze their trajectory.
- Quantifying energy deposition.
- Setting up several detection condition per simulation.

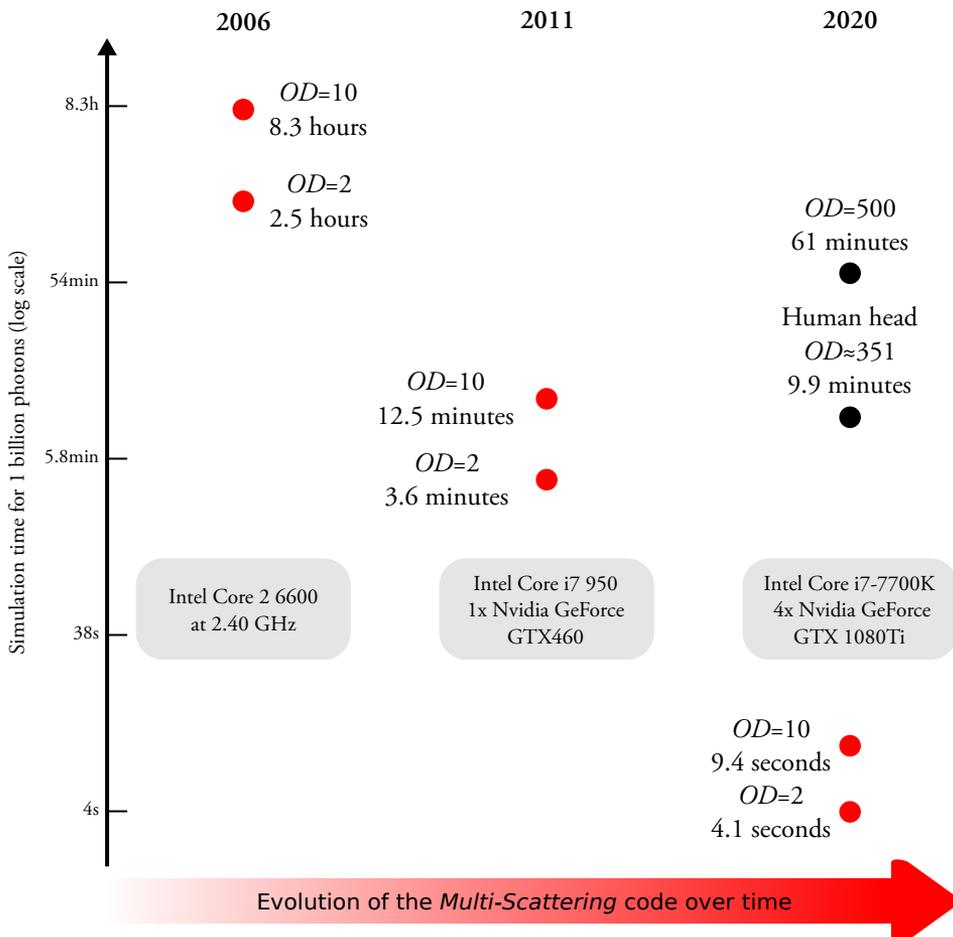


Figure 1.1: Evolution of the Monte Carlo code over the years, since its first development in 2006 by Edouard Berrocal [7], its initial parallel computation in 2011 by Joakim Jönsson [8] and its latest development where 4 GPUs are used per simulation [9]. Here *OD* is the optical depth, which approximates the average number of scattering events in the simulation. The drastic reduction of simulation time from several hours in 2006, to minutes in 2011 and only seconds in 2020 is shown.

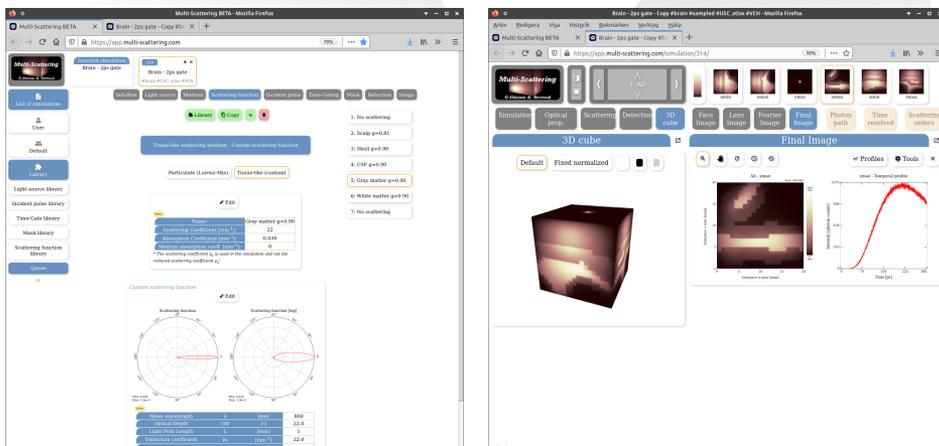
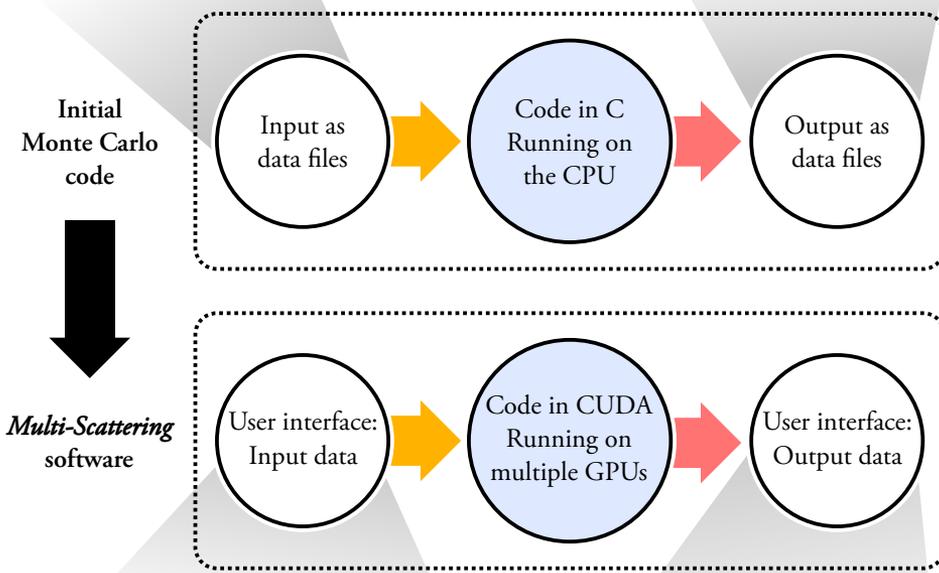
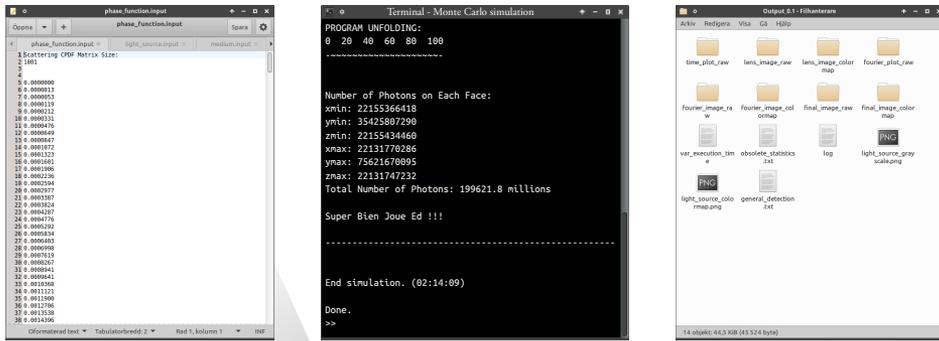


Figure 1.2: The original code was limited to input and output files. With the expansion into parallel computing, the software became more viable as a simulation tool while setting up simulations was still difficult. At the same time the requirements for more advanced computer hardware were unavoidable. An open access web-based system was created to balance those needs and make the software available and accessible.

During its development *Multi-Scattering* has been subjected to several iterations where features have been added and the model expanded to include user interfaces in order to make the software more accessible to non-experts in the field. This development has enabled the model to become versatile and it can now be employed for a large variety of studies involving medical, environmental and industrial applications. A few examples are listed below:

1. Biomedical applications:

- Assisting of blood analysis, finding the optical properties of human tissues (e.g. skin, muscles, *etc*) as well as supporting the detection and treatment of malignant tumors.
- Estimating photon path-length or time-of-flight to obtain quantitative measurements from near-infrared spectroscopy based techniques.
- Calculating the fluence distributions in tissues for photon delivery used in photodynamic therapy.

2. Industrial applications:

- Studying atomizing fuel sprays in combustion, spray drying for powder production and turbid beverages such as milk solutions.
- Developing and optimizing optical instruments for quantitative measurements of scattering particles and droplets. This includes deducing their size distribution and concentration.
- Predicting the contribution of multiple light scattering intensity and correcting for measurement errors.

3. Environmental applications:

- Investigating light scattering through the sky and clouds, predicting visibility through fog and smoke and assessing the turbidity of dirty water.
- Quantifying image transfer when imaging through scattering media.
- Testing and comparing optical filtering strategies to suppress undesired effects from multiple light scattering.

Chapter 2

Monte Carlo simulation of photon transport

2.1 General description

To explain how the Monte Carlo method works an example is shown in Fig. 2.1. Here the goal is to deduce the surface area of the blue region. This can be achieved by generating randomly distributed points within the overall known square area. By now counting the number of points within the blue region in relation to the total number of points, the ratio between the unknown and known areas can be approximated. If a sufficiently large number of sampled points are used, the estimate will then converge towards the exact solution. This process of generating random points is known as random sampling and is the core of any Monte Carlo simulation algorithm.

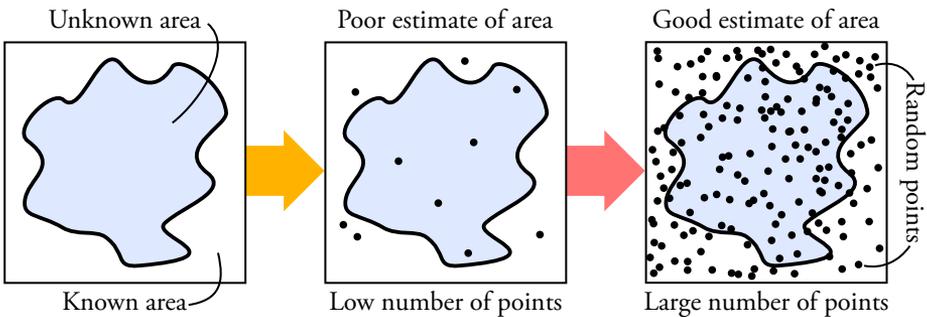


Figure 2.1: A simple exercise in Monte Carlo simulation. The area of the shape can be estimated by adding randomly scattered points throughout the known simulation area. By counting the number of points within the shape an estimate can be calculated. The precision of the result is improved by adding more points.

The Monte Carlo method can be used for more complex problems than the one given in fig. 2.1, such as for deducing photon transport through scattering media. This includes, as illustrated in Fig. 2.2, light propagation through skin tissues, analysis of blood, visibility through smoke and fog, turbidity measurement in beverages and turbid water as well as laser imaging of spray systems, to give a few examples.

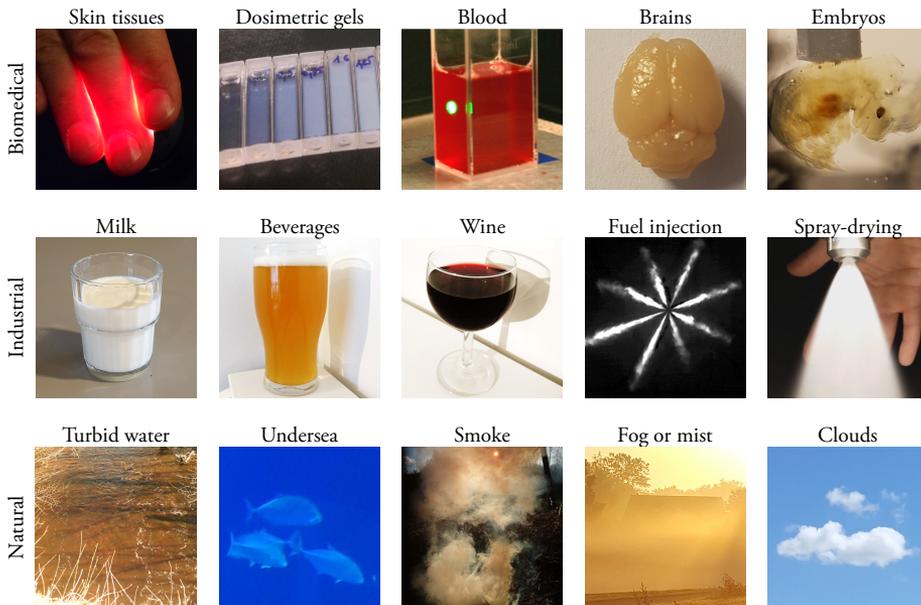


Figure 2.2: Examples of scattering media encounter in medicine (e.g. biomedical tissues), in the industry (e.g. beverages, spray systems) as well as in natural environments (e.g. turbid water, fog). Each medium is characterized by its own optical properties. Light propagation through those media can be simulated using the Monte Carlo method.

Even though the examples in Fig. 2.2 can be challenging to analyze the simulation of light transport through them can be broken down to a series of more simple probability events. A problem well suited for a Monte Carlo algorithm where the successive steps can be explained as follows:

1. The start of the simulation consists in launching photon packets from a light source into the simulated scattering volume. Thus the initial direction of propagation as well as position of each photon packet must be defined.
2. The second step consists in deducing the distance traveled by a launched photon packet prior to interaction. This distance is named the free path length and is calculated using a random number and based on the Beer-Lambert law. A highly scattering medium results in very short free path lengths while a dilute medium will let photons travel much further before interaction.

3. During an interaction event, a portion of the light may be absorbed while the rest of the photon packet will be scattered. The ratio between scattering and absorption depends on the optical properties of the scattering medium at the wavelength of the photon.
4. A new direction of propagation is defined for the scattered light. This is deduced by means of two random numbers and the scattering phase function. Depending on the characteristics of the scattering center and of the incident wavelength, the appropriate scattering phase function must be chosen.
5. While still inside the simulated volume, the photon continuously repeats step 2, 3 and 4. This process is stopped once the photon packet leaves the scattering medium. The photon is considered detected if it matches the pre-defined detection conditions.
6. By repeating these steps for a large amount of photon packets (over several billion) a converging solution is deduced.

To help understand the process of photon transport, some examples of photon paths within a scattering medium are given in Fig. 2.3. When characterizing particulate media, valuable information can be gained from ballistic and single scattered photons. However, photons from higher scattering orders may obscure the results. As those highly scattered photons carries less amount of valuable information they are referred to as diffuse photons. Due to the nature of scattering, more scattering events increases the time for the photons to reach the detector.

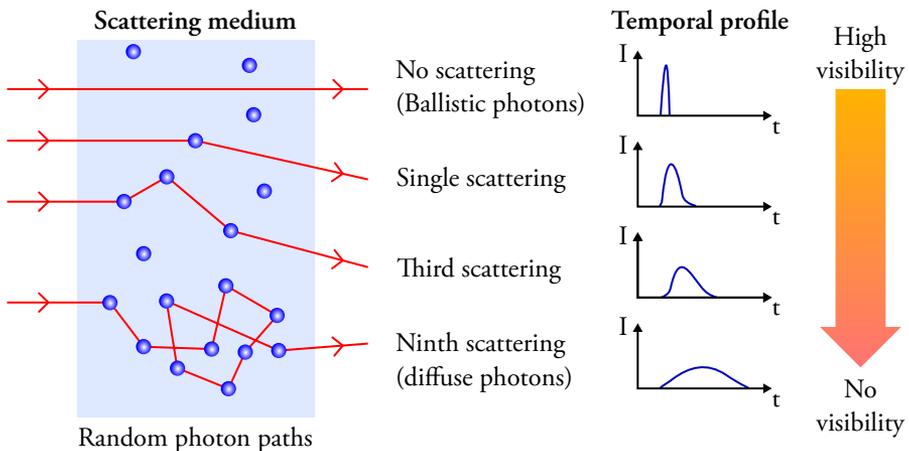


Figure 2.3: Examples of photon paths in a scattering medium showing the transition between non-scattered light and diffuse photons. The corresponding temporal profiles are illustrated on the right.

2.2 Radiative Transfer Equation

The radiative transfer theory is based on the energy conservation of incoming, outgoing, absorbed and scattered photons within an infinitesimal volume element. The optical properties of the volume element are assumed homogeneous and the central equation describing the average transport of photons through such volume is known as the Radiative Transfer Equation (RTE) (or equation of radiative transfer). An illustration describing the RTE is given in Fig.2.4. In the RTE only quantities of power or intensity are considered. Light propagation is envisioned as a photon stream and correlations between the radiation fields such as interference are neglected. Such assumptions can only be assumed if the wavelength of the incident radiation is small when compared to the dimensions of the scattering medium and for well separated scattering centers (independent scattering). For random scattering media, these conditions are respected. The RTE can be described as follows:

- The change of radiance along an incident direction corresponds to the loss of radiance due to the extinction of the incident light plus the amount of radiance that is scattered from all other directions into the incident direction.

As previously defined, light extinction equals the loss of radiance due to scattering of the incident light in all other directions, plus the loss of the radiance due to absorption.

$$\frac{1}{C} \frac{\partial I(\vec{r}, \vec{s}, t)}{\partial t} = -\mu_s I(\vec{r}, \vec{s}, t) - \mu_a I(\vec{r}, \vec{s}, t) + \mu_s \int_{4\pi} f(\vec{s}', \vec{s}) I(\vec{r}, \vec{s}', t) d\Omega' \quad (2.1)$$

The RTE is given above, in Eq.2.1, where t is time, \vec{r} is the vector position, \vec{s} is the incident direction of propagation, $f(\vec{s}', \vec{s})$ is the scattering phase function derived from the appropriate scattering theory (e.g. Lorenz-Mie or Rayleigh-Gans theory), $d\Omega$ is the spanning of the solid angle and C is the speed of the light in the surrounding medium. While the RTE is applicable for a wide range of turbid media, analytical solutions are only available in rather simple circumstances where assumptions and simplifications are introduced. For realistic cases of scattering media numerical techniques have been developed and utilized to solve the RTE. The most versatile and widely used numerical solution is based on the statistical Monte Carlo technique.

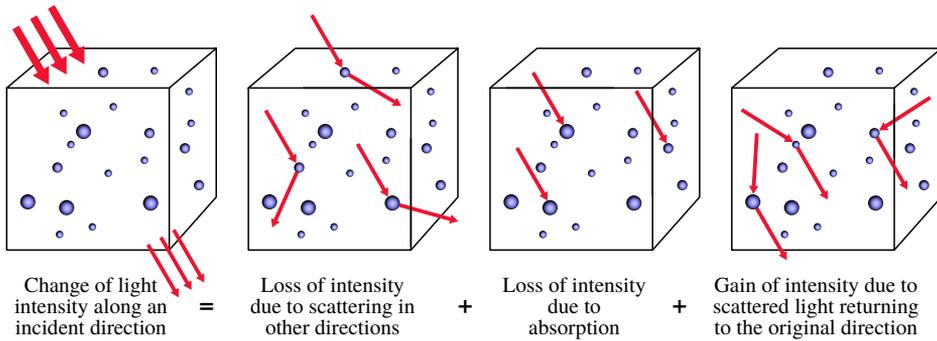


Figure 2.4: Visual description of Radiative Transfer Equation given in Eq. 2.1.

Despite the large variety of natural and industrial examples of scattering environments (see Fig.2.2) biomedical optics and medicine remains the research fields with the strongest interest in the modeling of photon transport using Monte Carlo simulation.

2.3 History of Monte Carlo simulation for light propagation in scattering media

Most scattering media can be divided into two groups: biomedical tissues and particulate media. The scattering in tissues results from inhomogeneities in the structure of the material. Particulate media differentiate by instead having randomly scattered and isolated particles [10].

The early adaptation of Monte Carlo simulation for light propagation through particulate scattering media began in the **1960s** in the field of Atmospheric research.

In **1965** a confidential report [11] was written by Collins and Wells. It presented the applications of two Monte Carlo procedures. Their aim was to study photon transport through the atmosphere under various environmental conditions.

In **1968** the focus shifts to light scattering in clouds [12]. The simulation introduced by Plass and Kattawar uses a scattering phase function generated from the Lorenz-Mie theory. Using the same code, the authors went on to deduce the radiance of multiple scattered light for a typical haze [13], for an atmosphere-ocean system [14] as well as for investigating time-of-flight Lidar measurements in oceans [15].

In **1973** research related to undersea optical imaging and to earth-satellite communication was presented. Funk used Monte Carlo simulations to determine the perfor-

mance of imaging systems undersea [16] while Bucher studied light pulse propagation through clouds [17].

Apart from the research in nuclear medicine, reaching back as early as the 1960s, the introduction of Monte Carlo simulations for light propagation in biomedical tissues was first introduced in 1983. Wilson and Adam introduced an applied Monte Carlo simulation to help predict and optimize light delivery for photodynamic therapy in light dosimetry [18].

It was not until 1983 that the first use of Monte Carlo simulation for the study of light propagation through biomedical tissues was reported by Wilson and Adam [18]. The technique was applied to light dosimetry in order to predict and optimize light delivery for photodynamic therapy.

This work inspired further research as two notable Ph.D. thesis were produced by Prah1 [19] in 1988 followed by Keijzer [20] in 1993. Together they co-author several publications [21], [22].

Using the results from Monte Carlo simulations, Flock et al. presented in 1989 a first comparisons which demonstrates the limits of the diffusion theory to accurately predict the fluence distribution through soft tissues [23].

Based on the previous work of Prah1 and Keijzer, the “Monte Carlo Multi-Layered” (MCML) simulation tool was introduced in 1994 by Wang et al [24]. Due to its source code being freely available it has become a popular and widely used tool for simulating photon transport through skin tissues.

In the beginning of the **first decade of 2000** new code was developed to account for successive wavelengths to deduce the absorption and reflectance spectra from skin layers [25]. This development by Meglinski and Matcher was soon followed by further improvements made by Churmakov for studying the localization of fluorescence within skin tissues [26].

Meanwhile in the fields of particulate scattering media, from 1983 and onward, Bruscaaglioni and Zaccanti co-published numerous articles on Monte Carlo simulations. Some of the applications includes the transmission of a light beam through a turbid atmosphere [27] (1983), the effect of a turbid medium on the Modulation Transfer Function of an optical system [28] (1989), the experimental validation of the Monte Carlo procedure [29] (1991) and the calculations of Lidar returns [30] (1995).

At the same time period, in 1989, Briton developed a Monte Carlo model called MUSCAT during his Ph.D. It is capable of a broad field of applications [31] such as the studies of visibility in foggy weather.

With this code, in 1994, Roze et al. was able to investigate the visibility through fog for realistic driving conditions [32]. The model took into consideration the presence of street lamps and it assumed that the road was absorbing the light.

A study of light propagation through smoke mixtures to better visualize lighted exit signs during fire situations was presented in 1991 by Roysam et al [33].

By introducing a voxel based Monte Carlo code in 2002, Boas et al. was able to improve the simulation of highly heterogeneous scattering media. Instead of infinite tissue layers, the new approach allowed for the use of a three-dimensional matrix consisting of 151x171x232 voxels to simulate light propagation in a human head. In turn the temporal evolution of photon migration could be visualized.

In 2005 Berrocal et al. laid the groundwork for making this thesis possible when he presented a new Monte Carlo model with applications in spray diagnostics. The model was able to simulate light propagation in highly inhomogeneous polydisperse turbid media [34, 35]. The simulation in [35] used 73x73x50 voxels and 25 different scattering phase functions to model an inhomogeneous polydisperse spray system typically seen in combustion applications.

This model was later validated in 2007 with experimental results [36]. Two years later in 2009 the contribution from individual scattering orders was analyzed and presented [37].

The Monte Carlo modeling of photon transport had a breakthrough in 2008 when Alerstam et al. proposed the use of a modern GPU to increasing the simulation speed [38]. With the use of general-purpose computing on graphics processing units (GPGPU) a computational acceleration of up to three orders of magnitude was reached compared to a single standard processor.

Using the same approach Fang and Boas demonstrated in 2009 an acceleration by a factor of 300 times [39]. The following year, in 2010, Alerstam, Lo et. al presented the MCML code running on a GPU acceleration [40]. This was successfully accomplished by analyzing and solving performance bottlenecks in the code in relation to how a GPU operates.

For an even more realistic 3D representation of complex tissue structures, Shen and Wang in 2010 [41], approached the problem with tetrahedrons. This was part of a software project called TIM-OS. During the same year Fang took a similar approach and proposed a mesh-based method referred to as MMCM [42].

A GPU accelerated Monte Carlo simulation together with an online web interface was proposed by Doronin and Meglinski in 2011 [43].

FullMonte, a GPU accelerated Monte Carlo code was introduced by Cassidy et al. in 2013 [44] followed by an updated and optimized version in 2019 [45].

MMC (Mesh-based Monte Carlo) was developed by Fang and Yan and introduced in 2019. By leveraging the OpenCL computing framework the code presented can be used both on CPUs and GPUs [46].

The application presented for this thesis, announced in 2020 [9]. It has been developed to consider both particulate scattering media as well as tissue-like media. To make it truly multi-purpose the model has been accelerated via parallel computing and part of the original code has been converted from C programming language to CUDA.

Chapter 3

Definition and terminology

3.1 Conversion from photon counts to irradiance

For each simulation numerous photon packets are being launched through a simulated scattering volume. Consequently the resulting data consists of photon counts being stored into matrices. In many situations, such as in light dosimetry in medicine, those numbers must be converted into more meaningful physical quantities. By definition, photon flux is defined as the number of photons per second per unit area:

$$\Phi = \frac{\textit{photon\#}}{\textit{seconds} \cdot \textit{meter}^2} \quad (3.1)$$

Thus, the flux can be directly obtained from the Monte Carlo simulation results. Now, the energy of a single photon (in Joules) is given as:

$$E = \frac{hc}{\lambda} \quad (3.2)$$

Here $h = 6.6261 \cdot 10^{-34}$ J·s is the Planck's constant, $c = 2.9979 \cdot 10^8$ m/s is the speed of light in vacuum and λ is the photon wavelength. Note that E can also be expressed in eV through the conversion factor $1 \text{ eV} = 1.602176634 \cdot 10^{19}$ J.

By now multiplying the photon flux Φ by the energy E of a single photon, the irradiance I (W/m^2) at a given wavelength can be deduced. The irradiance is the power of electromagnetic radiation incident per unit area on a surface and is determined as:

$$I = \Phi \cdot E = \Phi \cdot \frac{hc}{\lambda} \quad (3.3)$$

3.2 Extinction, scattering and absorption

When a collimated light beam of given wavelength and initial direction is crossing a homogeneous medium containing scattering and absorbing centers, the transmitted light intensity I_t can be calculated from the Beer-Lambert law such as:

$$I_t = I_i e^{-\mu_e \cdot l} \quad \text{where} \quad \mu_e = \mu_s + \mu_a \quad (3.4)$$

Thus, the incident light intensity I_i is exponentially reduced as a function of the length l traveled by the light through the medium and of the extinction coefficient μ_e , in mm^{-1} , which corresponds the sum of the scattering and absorption coefficients, μ_s and μ_a respectively. Assuming that the medium contains distinct scattering/absorbing centers, such as particles, droplets or molecules, then:

$$\mu_e = N \cdot \sigma_e \quad \text{and} \quad \sigma_e = \sigma_s + \sigma_a \quad (3.5)$$

where σ_e is the extinction cross-section in mm^2 and N is the number density of scattering/absorbing centers in $\#/\text{mm}^3$. At each interaction between light and a center, the fraction of light that is scattered over the total extinction is called the single scattering albedo, ω , and is calculated such as:

$$\omega = \frac{\mu_s}{\mu_s + \mu_a} = \frac{\sigma_s}{\sigma_s + \sigma_a} \quad (3.6)$$

For non-absorbing media $\omega = 1$ and for highly absorbing media ω tends to 0. For example, considering the visible light spectrum soot particles are mostly absorbing, water droplets are mostly scattering and red blood cells are both scattering and absorbing centers. The mean free path length corresponds to the average distance between two scattering/absorbing events. It is calculated from the inverse of the extinction coefficient such as:

$$\bar{l}_{fp} = \frac{1}{\mu_e} \quad (3.7)$$

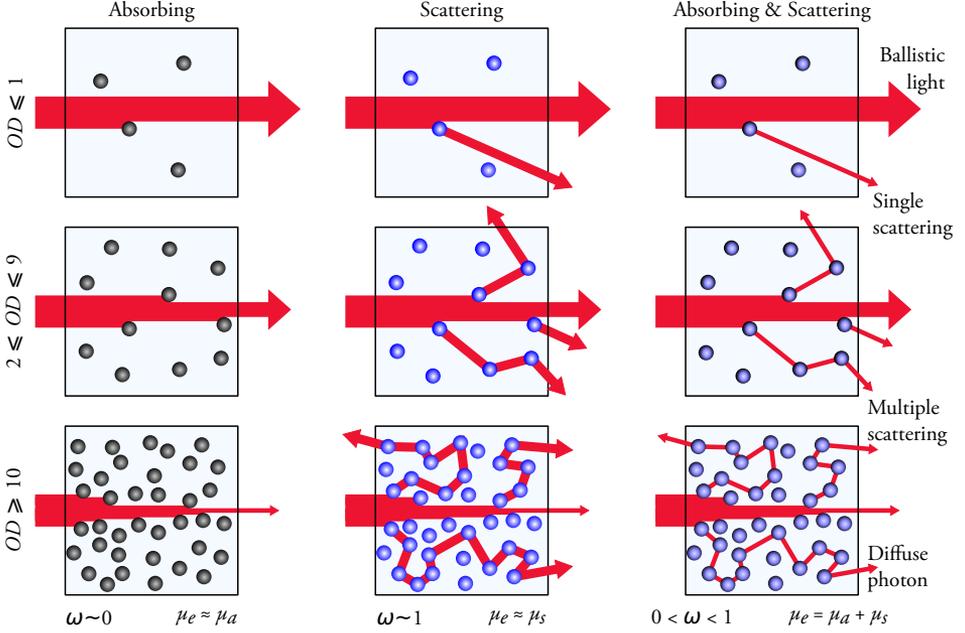


Figure 3.1: Classification of turbid media as a function of the optical depth OD and single scattering albedo ω . When $OD \leq 1$, then the *single scattering regime* is assumed where most photons are unscattered (ballistic light). When the optical depth ranges between $2 \leq OD \leq 9$, the *intermediate scattering regime* applies involving blurring effects. At $OD \geq 10$, the *multiple scattering regime* is assumed and visibility is lost. Scatterers can be mostly absorbing $\omega \sim 0$ (e.g. soot particles), mostly scattering $\omega \sim 1$ (e.g. water droplets), or both absorbing and scattering $0 < \omega < 1$ (e.g. blood).

By dividing the total length l of a turbid medium with its mean free path length \bar{l}_{fp} , the average number of scattering events along l is deduced. This average number of scattering events is known as the optical depth and denoted in this article as OD (also commonly denoted as τ in the literature):

$$OD = \frac{l}{\bar{l}_{fp}} = \mu_e \cdot l \quad (3.8)$$

As illustrated in Fig.3.1 turbid media can be categorized from their optical depth and albedo. Based on the optical depth three scattering regimes can be identified. If the optical depth is smaller than one, $OD \leq 1$, then the *single scattering regime* is assumed where single scattering events are dominant. If the optical depth ranges between $2 \leq OD \leq 9$, the *intermediate scattering regime* applies where visibility is reduced. This corresponds to the transition between seeing an object through a scattering medium, to not seeing it. In this regime the dominant scattering order is usually close to the OD value especially for forward scattering phase functions. Note also that the Modulation Transfer Function of an imaging system through scattering media can be calculated in this regime [28]. If now $OD \geq 10$, then the *multiple scattering regime* is assumed. In this case visibility is lost. Retrieving visibility can be done via the use of some

filtering strategies such as time-gating [47, 48], Fourier spatial filtering and/or using Structured Illumination based techniques [49]. For $OD \geq 15$ retrieving visibility is extremely challenging. In the multiple scattering regime the scattering orders tend to contribute nearly equally, allowing the diffusion approximation to be applicable. This is the case when light is crossing a few millimeters of medical tissues.

3.3 Scattering phase function and anisotropy factor

By definition, the scattering phase function $f(\vec{s}', \vec{s})$ is the angular distribution of light intensity scattered by a scattering center, such as a molecule, particle, cell, or droplet, at a given wavelength. Note that this is a misleading term as $f(\vec{s}', \vec{s})$ has nothing to do with the phase of a light wave. The parameters governing the scattering phase function are the incident light characteristics (wavelength, polarization state, intensity profile), the optical properties of the surrounding medium (external refractive index) and the characteristics of the scattering center (size, shape, refractive index, orientation). The scattering phase function is usually given under its normalized form:

$$\int_{4\pi} f(\vec{s}', \vec{s}) d\Omega' = 1 \quad (3.9)$$

When the scattering phase function is limited to the scattering plane, it is defined as $f(\theta_s)$, where θ_s is the polar scattering angle as described in Fig.3.2. For axial symmetry the scattering probability is homogeneous over the azimuthal angle ϕ_s , defined between 0 and 2π and the scattering phase function depends only on its scattering angle θ_s defined between 0 and π . This is the case for spherical particles/droplets and for scattering phase function derived from the Henyey-Greenstein theory [50].

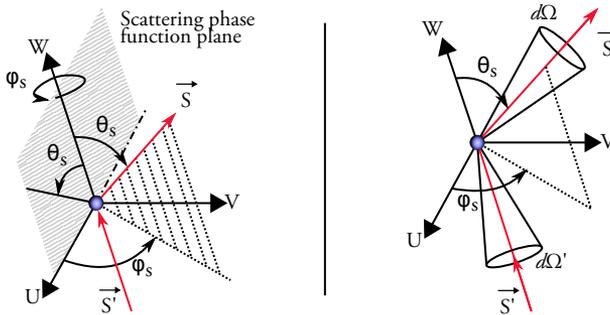
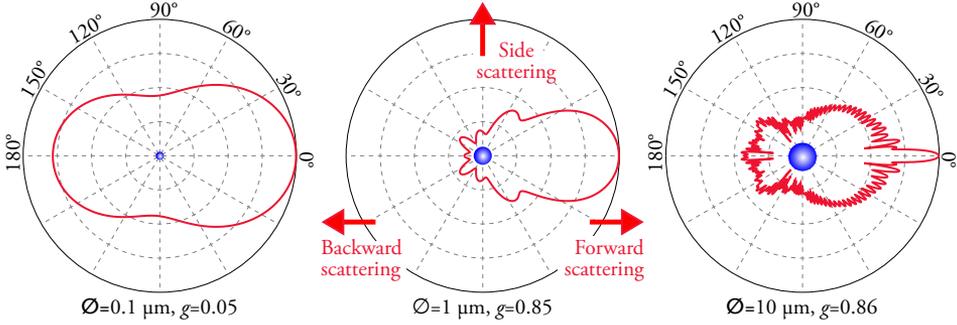


Figure 3.2: Directions and angles of a photon being scattered in the local UVW coordinate system. The polar and azimuthal scattering angles θ_s and ϕ_s are shown on the left side while the solid angles Ω'_s and Ω_s around the respective incident and scattered vectors are shown on the right side.

Logarithmic scale:



Linear scale:

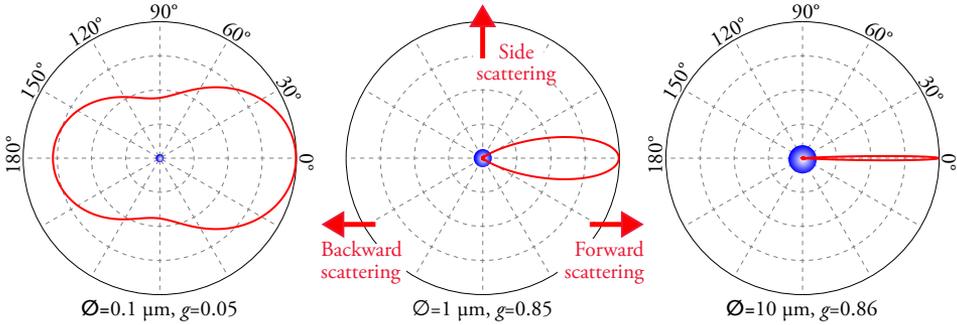


Figure 3.3: Examples of polar scattering phase functions for three size of spherical water droplets of refractive index $n_p = 1.333 - 0.0i$ suspended in air of refractive index $n_m = 1.000293 - 0.0i$. The incident light is assumed to be unpolarized and monochromatic at wavelength $\lambda = 600$ nm. As the size of the droplet increases, light is significantly scattered in the forward direction and scattering ripples are appearing.

Three examples of Lorenz-Mie scattering phase function calculated using the computational script provided by Bohren and Huffman in [51] are given in Fig.3.3. Here the incident light is assumed unpolarized at wavelength $\lambda = 600$ nm. For the $10 \mu\text{m}$ droplet, a highly forward scattering lobe is visible while for the $0.1 \mu\text{m}$ case light scatters more homogeneously corresponding to the Rayleigh scattering regime. Scattering phase functions can also be averaged over several wavelengths, when the incident light consists of a broad light spectrum, and/or when considering a distribution of particles. In the case of a homogeneous distribution of spherical particles of diameter D , the representative averaged scattering phase function $\bar{f}(\theta_s)$ is calculated such as:

$$\bar{f}(\theta_s) = \frac{\int_{D=0}^{\infty} n(D) \cdot \sigma_s(D) f(D, \theta_s) \cdot dD}{\int_{D=0}^{\infty} n(D) \cdot \sigma_s(D) \cdot dD} \quad (3.10)$$

To quantify the amount of light that is scattered in the forward direction from a scattering phase function, the mean cosine of the scattering angle can be calculated,

corresponding to the so-called anisotropy factor g :

$$g = \int_0^\pi f(\theta_s) \cdot \cos(\theta_s) \cdot 2\pi \cdot \sin(\theta_s) d\theta_s \quad (3.11)$$

$$\int_0^\pi f(\theta_s) \cdot 2\pi \cdot \sin(\theta_s) d\theta_s = 1 \quad (3.12)$$

where $-1 \leq g \leq 1$. Isotropic scattering occurs for $g = 0$ while a value near 1 indicates highly forward scattering. The anisotropy factor can also be used to define the scattering phase function via the Henyey-Greenstein equation [50], defined as:

$$f(\theta_s) = \frac{1}{4\pi} \frac{1 - g^2}{[1 + g^2 - 2g \cdot \cos(\theta_s)]^{3/2}} \quad (3.13)$$

In *Multi-Scattering* pre-calculated Henyey-Greenstein scattering phase functions can be chosen as input data. Finally, a quantity known as the reduced scattering coefficient is deduced using g such as:

$$\mu'_s = \mu_s \cdot (1 - g) \quad (3.14)$$

Note that μ'_s aims at reducing the calculation time from Monte Carlo simulations by extending the mean free path length when forward scattering phase functions are considered. Despite its merits this approximation does not allow for the consideration of complex scattering phase functions and can only be applied within the multiple scattering regime. For those reasons, the reduced scattering coefficient is not used in *Multi-Scattering* and μ_s is employed by default.

Chapter 4

Multi-Scattering: Successive steps and algorithm

4.1 General simulation algorithm

The simulation algorithm used in the *Multi-Scattering* software is divided into two sections:

- The *Monte Carlo code* that tracks photon packets through the simulated scattering medium.
- The *Detection code* that selects the desired photon exiting the scattering medium and simulates image formation using a ray-tracing approach.

The general algorithm of the software is shown in Fig.4.1. When a simulation is initiated, the *Monte Carlo code* starts and the initial position of photons is defined on a face of the simulated volume using the desired light source matrix and photon direction. Then, the distance to the next scattering/absorption interaction center is calculated. If the interaction center is within the boundaries of the simulated volume, the *Monte Carlo code* continues tracking photons; otherwise the photon packets leaves the simulated scattering volume and the simulation continues to the *Detection code* as shown on the right hand side of the flow chart in Fig.4.1. Then, depending on their properties photons are sorted by applying several filtering conditions, according to the desired detection case. Finally, various detection cases, set by the user prior to starting the simulation, are processed, resulting in the formation of image data. Results can be opened in the output web page where they are displayed and can directly be analyzed.

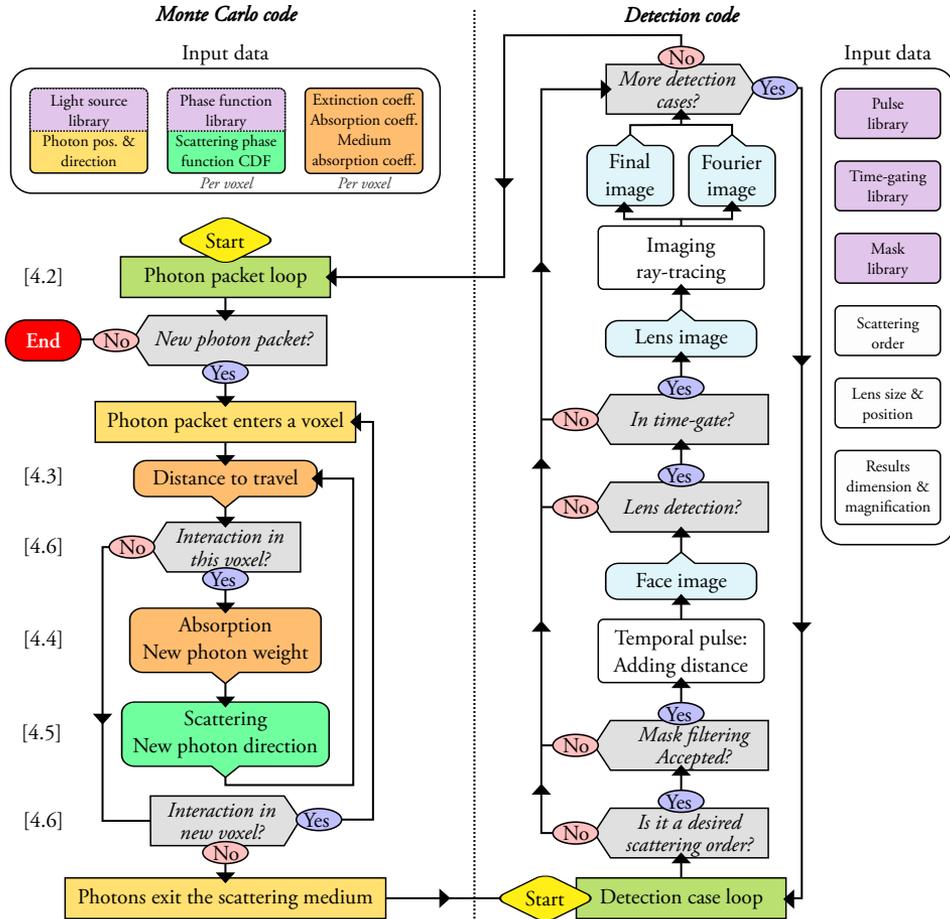


Figure 4.1: Flow chart describing the algorithm of the Multi-Scattering simulation. The algorithm is divided into two successive codes: The *Monte Carlo code* and the *Detection code*.

4.2 Launching photon packets

The simulation starts by launching photon packets from an image, see Fig. 4.2. This two-dimensional light source matrix is both scaled and positioned on the surface of the scattering medium as desired. It can either be uploaded by the user or alternatively, selected from pre-defined images proposed in the “Light Source Library”. Using this approach any experimental light source can be faithfully simulated, allowing for closer comparisons with experimental results as shown in [52].

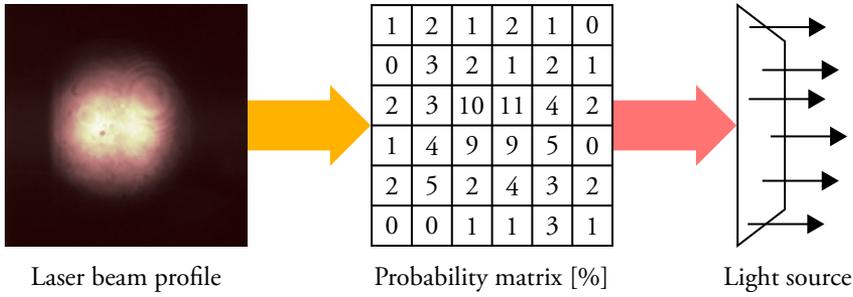


Figure 4.2: The principle of the simulation light source. A profile image is selected. The pixel intensity values are counted and the values are converted to a matrix of probabilities. New photons are launched randomly based on the probabilities.

Prior to launching photons, the source matrix is normalized and the probability weight of photons for each pixel is calculated accordingly. The pixel from which a photon packet is launched is deduced using two random numbers. By generating two more random numbers, a uniformly random point is defined within the boundaries of the chosen pixel. Finally, the absolute starting position on the face of the scattering medium is calculated by taking into consideration the size and position of the source matrix and the dimensions of the scattering medium. In contrast to sequential launching, this random positioning of the photons starting point is beneficial in preserving the intensity profile of the source in situations where a simulation is stopped prior to completion. Using this approach, results from aborted simulations remain meaningful as the entire light source is being statistically reconstructed. Furthermore, thanks to this feature additional photons can be launched to an already completed simulation. Once the starting photon position is chosen, the initial direction of photon propagation must also be defined by setting a value to the polar Θ and azimuthal Φ angles in the main coordinate system (XYZ). Depending on the directional features of the light source, two different approaches are available:

1. By setting up a photon dispersion cone angle: In this case photons are given a random direction vector that is uniformly constrained within the cone angle defined by θ_{start} . If this angle equals zero, a perfectly collimated beam is assumed resulting in the initial direction being a vector along one direction X, Y or Z. If, however, the angle is close to 90 degrees then a highly diffusing source is being simulated. Semi-directional light sources, such as LEDs, can easily be defined by setting up θ_{start} according to the illumination angle specifications.
2. By setting up a focusing distance: In this case, a focal point is defined by the user. This distance can either be positive for a focusing beam or negative for a diverging beam. The initial vector is pointing towards the focus point and photon direction is calculated accordingly. Thus, the incident angle differs

significantly depending on photon position on the source matrix. A useful application of this feature is the focusing of a laser beam into medical tissues.

Based on the chosen option and on the inserted input data, the angles Θ_i and Φ_i defining the incident directions of a photon packet are deduced accordingly. Finally, an initial weight equal to one, $W_0 = 1$, is set to a photon packet prior to launching.

4.3 Determination of the free path length

The free path length l_{fp} is the distance of propagation until an interaction event, scattering and/or absorption, occurs. The cumulative distribution function for the mean free path length is derived from the Beer-Lambert law and equals:

$$C(l_{fp}) = 1 - e^{-\mu_e \cdot l_{fp}} \quad (4.1)$$

This yields the probability density function of a photon packet to be scattered and/or absorbed after traveling the distance l_{fp} :

$$P(l_{fp}) = \mu_e \cdot e^{-\mu_e \cdot l_{fp}} \quad (4.2)$$

Following the process of random sampling and considering a given random number ξ_1 where $\xi_1 \in (0, 1)$ and the corresponding distance l_1 , it is deduced that:

$$\int_0^{l_1} P(l_{fp}) \cdot d(l_{fp}) = \xi_1 \quad (4.3)$$

Which leads to

$$\int_0^{l_1} \mu_e \cdot e^{-\mu_e \cdot l_{fp}} \cdot d(l_{fp}) = \xi_1 \quad (4.4)$$

Finally,

$$1 - e^{-\mu_e \cdot l_1} = \xi_1 \quad \text{and} \quad l_1 = -\frac{\ln(1 - \xi_1)}{\mu_e} \quad (4.5)$$

However, as $(1 - \xi_1)$ equals another random number uniformly distributed between 0 and 1, the free path length l_{fp} between two light-particle interactions can be deduced from any sampled random number ξ as:

$$l_{fp} = -\frac{\ln \xi}{\mu_e} \quad (4.6)$$

By knowing the distance l_{fp} and the initial direction of propagation, the position of the next scattering/absorbing center is calculated within the absolute (XYZ) coordinate system. If this interaction center is located inside the medium, then a new direction is calculated, otherwise the photon's journey in the scattering volume is ending.

4.4 New photon weight

Photon packets start with a photon weight equal to unity, $W_0 = 1$. Once an interaction event occurs, the amount of scattered light must be adjusted accordingly depending on the absorption properties of the medium. Thus, at each successive scattering/absorption center the photon weight is gradually reduced by the fraction of light being absorbed. This loss of energy, named energy deposit, is an important parameter in optical dosimetry. At the interaction n the fraction of the photon packet being absorbed corresponds to:

$$\Delta W_a = W_n \cdot \left[\frac{\sigma_a}{\sigma_s + \sigma_a} \right] = W_n \cdot (1 - \omega) \quad (4.7)$$

Where σ_a and σ_s are the respective absorption and scattering cross-sections of the interaction center, while ω is the albedo. Finally the new photon weight W_{n+1} is given as:

$$W_{n+1} = W_n - \Delta W_a \quad (4.8)$$

If the interaction center is non-absorbing, the absorption cross-section is set to zero and the photon weight remains unchanged.

In the case of a particulate scattering medium (particles in a surrounding medium), the photon weight can be further reduced if the surrounding medium is absorbing. Note that the absorption of the surrounding medium differs from the absorption of the scattering centers. Examples of this would include situations in which individual particles or droplets are suspended in a medium that absorbs the probing light. On the other hand, for tissue media, there is no distinction between scattering centers and surrounding medium. So rather than using absorption coefficients for each, an overall coefficient is applied.

The final reduction in photon weight is calculated as:

$$W_{final} = W_n \cdot e^{-\mu_{am} \cdot l_{total}} \quad (4.9)$$

Where μ_{am} is the absorption coefficient of the surrounding medium and l_{total} is the total distance traveled once photons have exited the medium.

4.5 New photon direction

When scattering occurs, a new propagation direction must be specified. The directions of propagation of a photon packet before and after a scattering event are respectively defined by the vectors \vec{S}' and \vec{S} within the absolute coordinate system (XYZ).

The transformation from \vec{S}' to \vec{S} is performed using the polar and azimuthal scattering angles θ_s and φ_s defined within a local coordinate system (UVW) as illustrated in Fig.4.3. This transformation is mathematically expressed in Eq.4.10 where scattering angles θ_s and φ_s are calculated from the appropriate scattering phase function:

$$\begin{pmatrix} S_x \\ S_y \\ S_z \end{pmatrix} = \begin{bmatrix} \frac{S'_x S'_z}{(1-S_z'^2)^{1/2}} & -\frac{S'_y}{(1-S_z'^2)^{1/2}} & S'_x \\ \frac{S'_y S'_z}{(1-S_z'^2)^{1/2}} & -\frac{S'_x}{(1-S_z'^2)^{1/2}} & S'_y \\ -(1-S_z'^2)^{1/2} & 0 & S'_z \end{bmatrix} \begin{pmatrix} \sin \theta_s \cos \varphi_s \\ \sin \theta_s \sin \varphi_s \\ \cos \varphi_s \end{pmatrix} \quad (4.10)$$

For the simple case of isotropic scattering, the scattering angles are calculated using random numbers ξ such as $\theta_s = \cos^{-1}(2\xi - 1)$ and $\varphi_s = 2\pi\xi$. For spherical particles and droplets, the scattering process is derived from the Lorenz-Mie theory (see Fig.3.3). In this case, as well as for any axisymmetric scattering phase function, the most straightforward approach of generating the polar angle θ_s is to use the inverse Cumulative Density Function (CDF) method, Fig. 4.4.

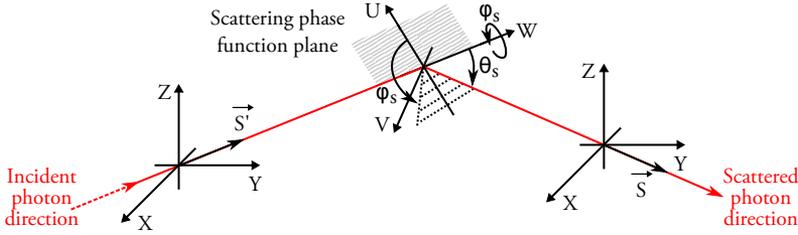


Figure 4.3: To calculate a new photon direction \vec{S} after a scattering event, a change of coordinate system is used such as: $(X, Y, Z) \Rightarrow (U, V, W) \Rightarrow (X, Y, Z)$. The scattering center is located at the origin of (U, V, W) , the W axis is aligned with the incident propagation direction and the scattering phase function is defined in the (U, W) plane. Finally, the vector \vec{S} is deduced using θ_s and φ_s through the matrix transformation given in Eq. 4.10.

The function $CDF(\theta_s)$ is calculated by integrating the scattering phase function $f(\theta_s)$ over the solid angle $d\Omega'$. Then, a random variable θ_s can be generated by taking the inverse transform of a random number uniformly distributed between zero and one, such as: $\theta_1 = CDF^{-1}\xi_1$ where:

$$CDF(\theta_1) = \int_0^{\theta_1} f(\theta_s) \cdot d\Omega(\theta_s) \quad (4.11)$$

When the analytical form of the inverse CDF is not available, the CDF is stored in a lookup table and the inverse transformation is performed at each scattering event using a random number, Fig. 4.4. By considering the scattering phase function to be axisymmetric, the azimuthal angle φ_s is deduced from $\varphi_s = 2\pi\xi$.

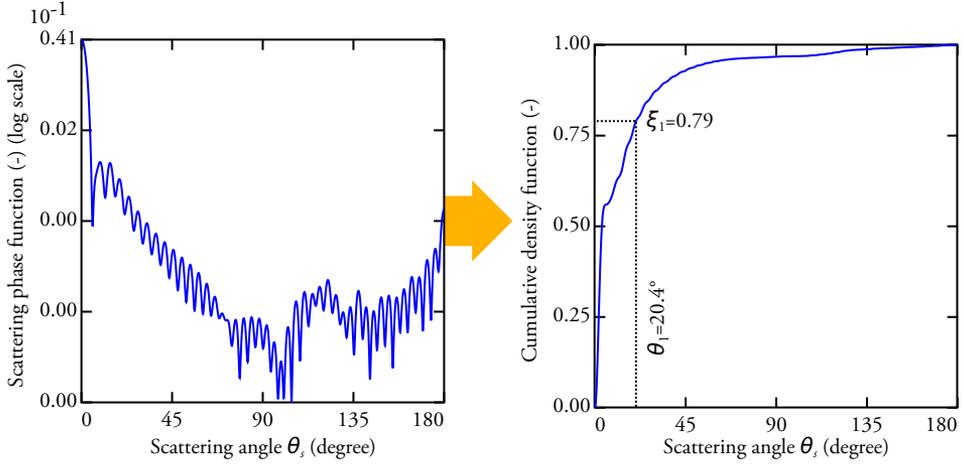


Figure 4.4: An example of the calculated CDF from a scattering phase function. A new scattering angle is found through the CDF by randomly selecting a new scattering probability (between 0 and 1). The plots correspond to particles of 4.9 μm polystyrene microspheres in water and illuminated by $\lambda = 473 \text{ nm}$. A random variable equal to $\xi_1 = 0.79$ would give a scattering angle of $\theta_1 = 20.4^\circ$

4.6 Photon crossing voxels and exiting the scattering volume

Each time photons scatter from an interaction center, a new free path length l_{fp} is calculated defining the distance traveled by photons prior to the next interaction. This distance must then be compared to the the distance corresponding to the voxel boundary, l_{exit} , where photons will exit the voxel if no further interactions occur. Note that the scattering medium can either be defined by a single voxel or by multiple voxels as illustrated in Fig.4.5(a). The challenge in determining l_{exit} , is figuring out which face of the voxel is concerned. This is done by calculating the three distances l_x , l_y , and l_z corresponding to the respective distances traveled by the photon packet until the local closest (xz), (yz) or (xy) planes are encountered. The smallest l is used as the length l_{exit} as illustrated in Fig.4.5(b). By now comparing l_{fp} with l_{exit} , three scenarios can occur:

1. If $l_{fp} < l_{exit}$: Then, the next scattering event will take place inside the voxel and the process will be repeated.
2. If $l_{fp} > l_{exit}$ and photons enter a new voxel: Then, the free path length in the new voxel $l_{fp(new)}$ must be adjusted as a function of the ratio of the extinction coefficients between the “old” and the “new” voxels such that:

$$l_{fp(new)} = (l_{fp} - l_{exit}) \cdot \frac{\mu_e(old)}{\mu_e(new)} \quad (4.12)$$

Note that in the current model refractive-index-mismatch between adjacent voxels is not considered. Further developments will take into account refractive-index-mismatch in the same fashion as described in MCML [24].

3. If $l_{fp} > l_{exit}$ and the photon packet exit the simulated volume: Then, the photon journey inside the medium is terminated.

When photon packets exit the simulated volume, their direction of propagation, total distance traveled, weight and position are known. The *Monte Carlo code* ends and the photons are processed through a second code referred as the *Detection code*.

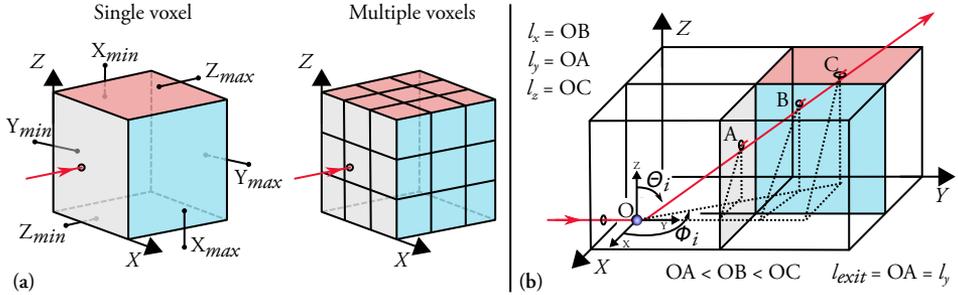


Figure 4.5: (a) Illustration of simulated volumes within the absolute (XYZ) coordinate system; including a single voxel and a 3x3x3 voxel matrix. The exit faces are termed X_{min} , X_{max} , Y_{min} , Y_{max} , Z_{min} and Z_{max} respectively. (b) Illustration of a scattering event (adapted from Fig.5.6 in [7] with permission) with the respective polar and azimuthal angles Θ and Φ in (XYZ).

4.7 Photon detection and image formation

The *Detection code* aims at filtering out photons for various detection cases that are set, prior to starting the simulation, see Fig. 4.6. Having multiple detection settings presents two main advantages: First a given simulation does not need to be run again for each new detection case, saving computational time. Second, it allows for the easy exploration of multiple detection cases where results can be directly compared to one another. This provides further insight on how photons are propagating through the simulated volume and how they contribute to image formation. Each detection case is defined by a number of criteria that must be fulfilled. The successive detection filtering criteria used for sorting out and selecting the desired photons are described below together with the successive steps characterizing the *Detection code*:

- **Scattering order filtering:** This is the first filter applied. If this filtering is activated, then only photon packets having undergone a given number of scattering events are detected. Note that this filtering process does not need to be activated and in this case all photons can be considered.

- **Mask filtering:** A mask, consisting of a binary matrix is applied on the face of the simulated medium. At locations where zero is set, light is blocked, and photons are rejected. On the contrary, at locations where unity is set, light is fully transmitted, and photons are accepted. This filtering process is found to be advantageous for the simulation of optical fibers located on the surface of the simulated medium. An option called “No mask” can be selected by the user where all light exiting the medium is considered. Various masks can be selected from the “Masks Library”.
- **Temporal pulse:** At this stage of the simulation, the temporal profile of the illuminating pulse is considered. By doing this operation at the end of the simulation, instead of the beginning, a variety of pulse durations and shapes can be considered from the “Incident Pulse Library”. This is implemented by simply adding a distance to the total path traveled by the exiting photon packet. This added distance is calculated through random sampling using a temporal profile of a given shape and duration that has been pre-selected for each detection case prior to running the simulation.
- **Lens filtering:** To simulate image formation, a lens is now positioned at a defined distance from the exiting face. The lens is assumed to be a disk of known diameter. Only photon packets reaching the lens are collected and contribute to image formation.
- **Time-gating:** A temporal gate is applied to photons reaching the collecting lens. The temporal gate is pre-selected from the “Time-Gate Library”. The temporal gate is defined by its shape, duration and start. Only photons having a time-of-flight within the temporal gate are selected. Note that the photon weight is modified according to the value of the gate and arrival time. An option called “No Gating” can be selected by the user where all light reaching the lens is considered no matter their time-of-flight.
- **Spatial filtering:** A spatial gate is applied to photons reaching the lens. Photons are filtered and detected when their scattering angle is within an acceptance half-angle defined by the detection setting. This type of filtering corresponds to an aperture experimentally located at the Fourier plane of the collecting lens.
- **Image Formation:** After having undergone the successive filtering processes described above the selected photon packets contribute to the formation of an image using a ray-tracing approach. Two images are created during this process: One corresponding to the Fourier plane and one at the image plane. To form the final image, the number of pixels employed and the desired magnification must be specified beforehand.

At the end of the *Detection code* four types of images, Fig. 4.7, are generated and referred to as:

- The **Face Image** showing the photon distribution at the face of the simulated medium.
- The **Lens Image** showing the light intensity spreading over the collecting lens.
- The **Fourier Image** showing the angular distribution of photon direction.
- The **Final Image** showing the image being formed at the image plane.

	All photons	1 scattering	Mask	Lens far
File name [name]	All photons	1 scattering	Mask	Lens far
Image quality [name]	Low ▾	Low ▾	Low ▾	Low ▾
Incident pulse [name]	5ps ▾	5ps ▾	5ps ▾	5ps ▾
Face select. [name]	all ▾	all ▾	all ▾	all ▾
Lens diam. [mm]	50	50	50	50
Face-Lens [mm]	75	75	75	100
Object-Lens [mm]	100	100	100	125
Fourier angle [degree]	20	20	20	20
Scat. order #	all ▾	1 ▾	all ▾	all ▾
Time-Gating [name]	No gating ▾	No gating ▾	No gating ▾	No gating ▾
Gate start [ps]	0	0	0	0
Polarization [yes/no]	no ▾	no ▾	no ▾	no ▾
Min. pola. [degree]	0	0	0	0
Max. pola. [degree]	360	360	360	360
Window start [ps]	0	0	0	0
Window duration [ps]	10	10	10	10
Mask [ID nb.]	0	0	1	1
Mask size X [mm]	0	0	0.6	5
Mask size Y [mm]	0	0	0.6	5
Mask offset X [mm]	0	0	10	0
Mask offset Y [mm]	0	0	0	0
Photon path channel (0),1-5	0	1	2	3
Copy / Remove	⊕ ⊖	⊕ ⊖	⊕ ⊖	⊕ ⊖
	1	2	3	4

Figure 4.6: A single simulation can have multiple detection cases active. Each detection case has multiple options controlling the *Detection code*.

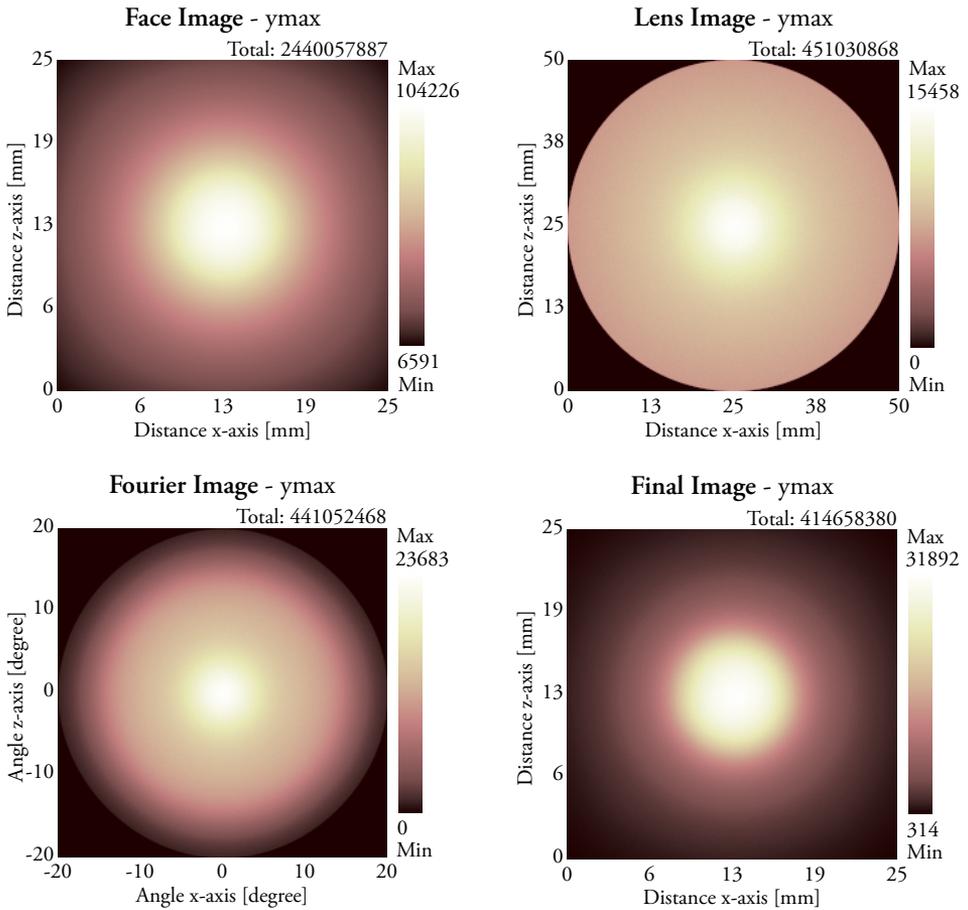
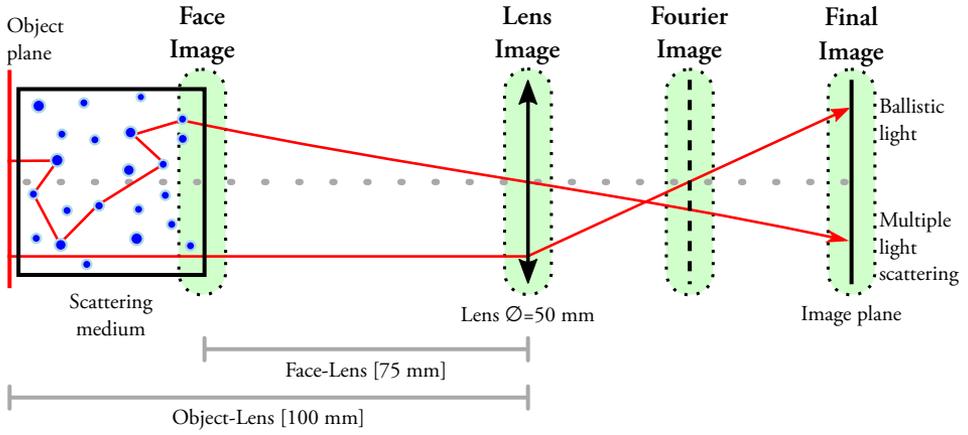


Figure 4.7: Example of the four different imaging results at the different locations highlighted in the drawing. The simulation uses a 10 mm laser beam ($\lambda = 473$ nm) and crosses a 25 mm scattering medium ($OD = 10$) of 5 μ m water droplets in air.

4.8 Libraries used in Multi-Scattering

The purpose of adding libraries to *Multi-Scattering* is for users to be able to quickly and intuitively find suitable simulation parameters, examples in Fig. 4.8. The user can choose to use one of the pre-defined input parameters in the libraries. They can also extend the libraries to include personal alternatives. So far the libraries cover the light source, the time-gate, the incident pulse, the scattering phase function and the mask.

- The **light source** library includes light source matrices. The library includes Gaussian light sources, experimental sources and structured beams to give a few examples.
- The **time-gate** and the **incident pulse** library contains a variety for different waveforms. They cover both experimental time-gates as well as incident pulses in the frequency domain.
- The **scattering function** library allows users to quickly select various types of optical properties for the scattering medium. The user may select a phase function, either as a particulate medium based on the Lorenz-Mie theory or they can choose a Henyey-Greenstein scattering functions.
- The **mask** library allows users to select a mask matrix to be used in the simulation. Examples such as an optical fiber or optical fibers utilizing symmetry are available.

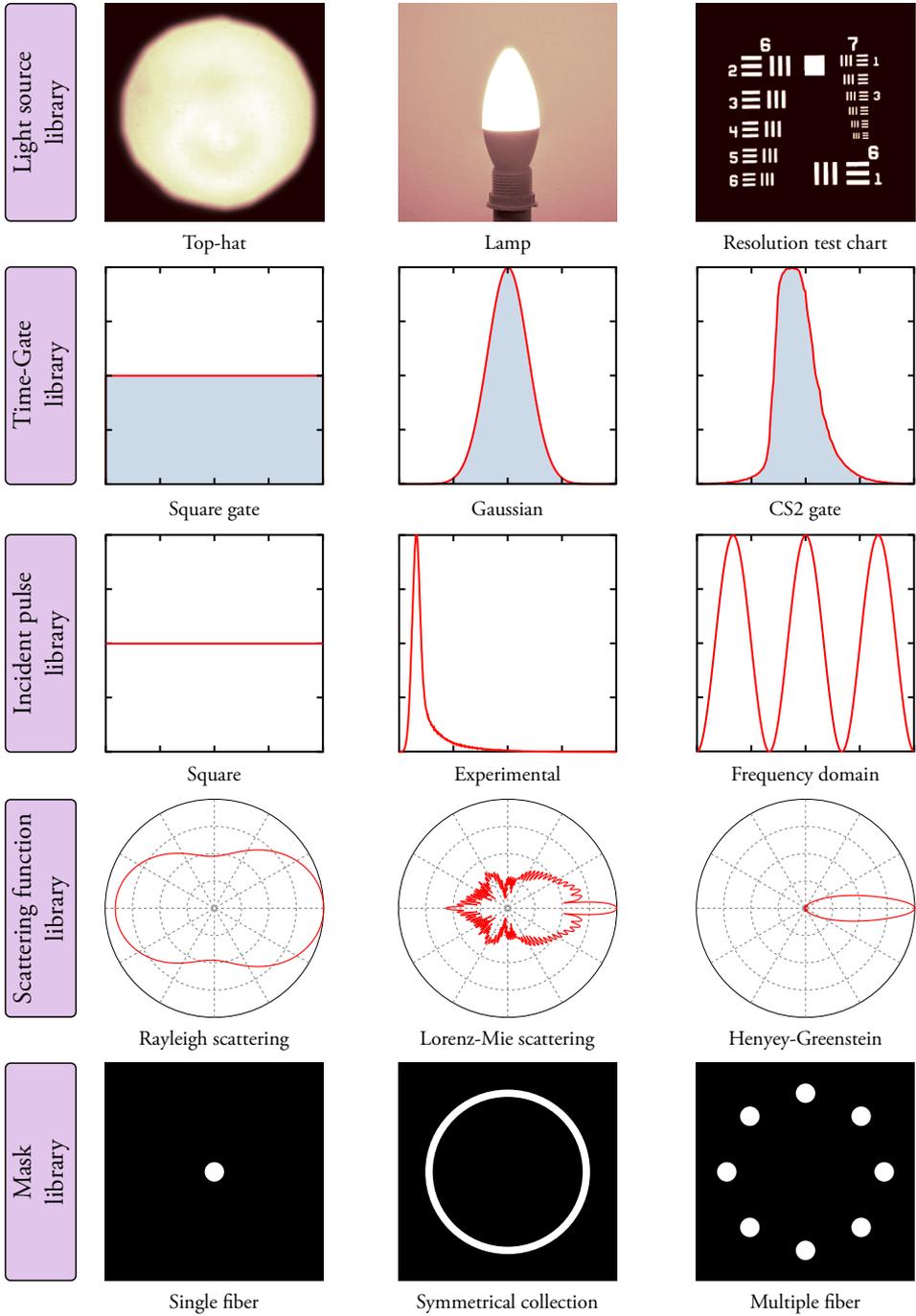


Figure 4.8: By having libraries as a part of *Multi-Scattering*, users can quickly configure their simulations by using pre-defined content or by adding their own.

Chapter 5

Accelerated computing with modern hardware

5.1 CUDA and modern GPU's architecture

Parallel processing can be employed to improve the performance of a heavy computational algorithm. The benefit of accelerating the simulation via parallel processing is not only the significant reduction in turnaround time but also that it allows for a much larger number of photon packets to be tracked and processed. This is an important detail, as it reduces inaccuracy induced by minimal statistics and ensures the convergence of the results toward the exact solution of the RTE. In this work, CUDA is used as the programming interface to parallelize the simulation and to efficiently reduce the execution time. CUDA was created by Nvidia with the intention of using GPUs for general-purpose calculations; calculations that can be performed in parallel. While CPUs usually have threads counted in tens, GPUs threads are counted in thousands. Transitioning an application from a single thread to two threads splits, at best, the simulation time by half. The large number of threads with modern GPUs opens up the possibility of accelerating the simulations by several orders of magnitude.

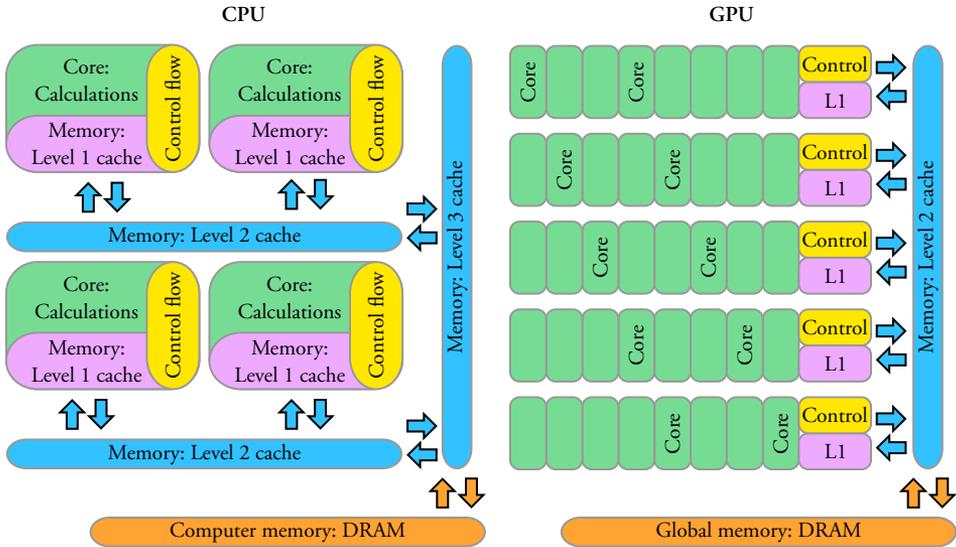


Figure 5.1: Simplified representation of the circuit design showing the major differences between a modern CPU on the left and a modern GPU on the right. The GPU has a much more compact layout with cores tightly packed together. The core block executes instructions according to a flow of instruction (program code). The control block selects which thread should be running or if it should be suspended while waiting for memory.

The reason for the vast difference in the number of threads between CPUs and GPUs stems from the design of the two circuits. As illustrated in Fig. 5.1 the cores on a GPU are much more tightly packed together with a lower number of control blocks. In general, only one thread can be executed on a single core at a given time while multiple threads can be kept ready by the control block. By grouping cores together with one control block more space becomes available for additional cores. The trade-off of this configuration is that the threads on the GPU cannot work independently. Thus, all the cores in one control group must follow the same flow of instructions. If, for example, the function of one thread is to multiply two numbers, then all threads in the group of cores must also multiply two numbers. However, the benefit of this architecture comes from the fact that each core can access different regions of memory which means that the GPU is highly optimized for calculations using multiple input data and producing multiple output data. Consequently, this design approach works particularly well in favor for Monte Carlo simulations. By means of thousands of threads, each one tackling an individual photon packet in parallel, the overall simulation time is greatly reduced.

5.2 Optimizing the code for running on the GPU

To make a code run efficiently on the GPU a few important aspects must be considered. First, all data that is processed on the GPU originate usually from a code running on the CPU. Thus, the flow of operation relies on the preparation of some data by the CPU that will later be processed by the GPU. Once this operation is accomplished, the prepared data is transferred from the CPU's memory to the GPU's global memory. The CPU instructs the GPU to process the data and finally the resulting data can be transferred back to the memory of the CPU as illustrated in Fig. 5.2(a). The time needed for transferring memory back and forth should not be neglected, unless the time required to process the data on the GPU is significantly longer (by several orders of magnitude). The amount of time lost due to memory transfers and readying the GPU could potentially negate all the speed benefits from the GPU. As shown in Fig.5.2(b) some of the slow memory transfers can be avoided by organizing the code smartly. This optimization utilizes a feature of the GPU which allows for memory transfers at the same time as a task is being computed. To reduce the total simulation time further, CPU idle time can be used to process the resulting data or to prepare the next task while the GPU is busy.

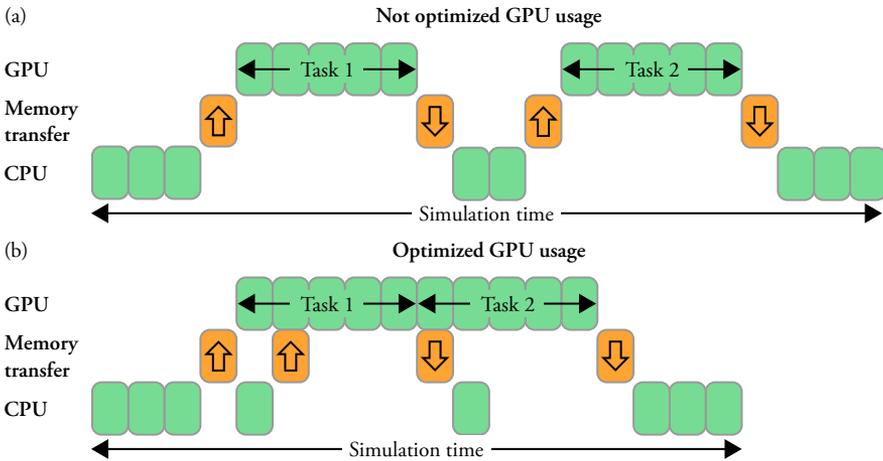


Figure 5.2: The simulation time can be reduced by reorganizing tasks and memory transfers. In (a), even if Task 1 & Task 2 are running on the GPU to its full capacity, efficiency can be lost due to delays (waiting for the next task and for memory to be copied). Those delays can be suppressed by adequately rearranging memory transfer as shown in (b).

Multi-Scattering is optimized by first creating two queues, or streams as they are called in CUDA, for the GPU. It then immediately gives each queue a task to process a number of photon packets. The CPU code will then alternate between the two queues performing a synchronized wait for it to finish. Once finished more photons are sent

to the idle queue so that the GPU always remains busy. Instead of transferring a lot of data back and forth, almost all of the Monte Carlo code is run by the GPU including the generation of the final results. Furthermore, the results from the image formation are matrices of photon packet counts. These matrices are also stored in the GPU's global memory since modern GPUs have almost the same amount of memory available as is typically installed alongside the CPU. Occasionally multiple threads will try to write data to the same memory segment simultaneously. With the help of atomic operations the race conditions are avoided and data loss in those situations are prevented.

5.3 Generation of random numbers

Monte Carlo simulations are based on the use of a very large series of random numbers. To create and generate random numbers on a computer, pseudo-random number generators (PRNG) are most often used. There are many existing varieties of PRNG algorithms to choose from, each defined by their own series length and limitations affecting computational cost. A common issue with PRNGs is that the series of random numbers eventually repeats itself due to memory limitations in storing the state of the generator. A state using more memory would in principle result in a larger period. To initialize the state of a PRNG a seed is used. Initializing the generator with the same seed leads to an identical series of random numbers. Using different seeds is thus important when repeating the same simulation in order to increase the statistics. To boost the performance of an application with multi-threading it is beneficial to assign one PRNG per thread. When selecting a random number generator for Monte Carlo simulations it is important to consider the following aspects: 1) The period of the PRNG must be larger than the number of random numbers needed. 2) When multiple PRNGs are used in parallel the series of numbers generated must be uncorrelated. 3) Random numbers must be generated fast enough to avoid slowing down the Monte Carlo algorithm.

In *Multi-Scattering* simulations, several random numbers are generated for different purposes during the tracking of each photon packet:

- Two random numbers are used to select the pixel position from the light source matrix.
- Two additional random numbers are used to select the starting position within the pixel.
- For non-collimated beams - diverging, converging or diffused light sources -

three additional random numbers are needed to define the incident direction within a cone angle.

- One random number is employed to determine the distance traveled prior to the first scattering/absorption event.
- At each scattering event three random numbers are employed: Two for the determination of the new photon direction of propagation and one for determining the position of the next scattering event.
- Finally, when considering a light pulse of given duration, one random number is used to define at which location within the pulse the tracked photon packet is generated.

Consider a high optical depth such as, $OD = 100$, and launching $nb_{photon} = 10^{10}$ photons, the amount of random numbers needed would reach:

$$nb_{\xi} \approx nb_{photon} \cdot (2 + 2 + 1 + 3 \cdot OD + 1) \approx 3 \cdot 10^{12} \quad (5.1)$$

Multi-Scattering is designed to use the cuRAND library part of the official CUDA toolkit. With the use of XORWOW a period greater than $2^{190} \approx 1.5 \cdot 10^{57}$ is guaranteed. Running a simulation on the GPU with *Multi-Scattering* will result in the creation of multiple PRNGs. First, each GPU will have their own starting generator initialized from a unique seed. Then each thread will be assigned their own sequence number. This will allow all the threads to track multiple photon packets in parallel without any risk of running out of random numbers. By initializing the generator with a unique seed, from the cuRAND library, a unique starting state of the PRNG is obtained resulting in uncorrelated sequences of random numbers between the threads.

5.4 Optimized light source matrix

The light source used in the simulations of *Multi-Scattering* has the ability to stop a running simulation at any point in time. It can also load a previous simulation and start adding more photons to it. The concept of this approach works by loading the light source as a matrix of cumulative probabilities. A random number is then used to find an index of the matrix and a photon is launched from the area covered by the corresponding index (pixel) location of the light source. The difficulty here is to locate the correct probability according to the random number. A simple but inefficient solution to this problem is to iterate over the matrix until the correct probability has been found. However, large part of the light source matrix might have a very low

probability or even zero probability of sending a photon. Thus, computational cycles might be lost while iterating over the full matrix.

Optimizing the algorithm of the light source will help mitigate this effect. However it will mostly benefit simulations with lower OD as higher numbers will reduce the impact of the time spent in the light source code. The simulation time per photon in the medium can be seen as the time to generate a photon plus the time needed to complete the scattering events:

$$t_{\text{photon}} \approx t_{\text{light source}} + t_{\text{scattering}} \cdot OD \tag{5.2}$$

The algorithm used in *Multi-Scattering* is based around a binary tree, see Fig. 5.3. Since the light source matrix is 32 bit integer intensity values, the nodes of the tree stores 64 bit values (cumulative). The index positions do not need to be saved in the memory as they can be calculated while searching the tree. In the worst case the binary tree will be twice the size of the original matrix. However, searching the tree only requires $\log_2(\#nb_pixels)$ iterations. As the algorithm only uses integer values and the input data are intensity integer values, *Multi-Scattering* will always generate photons according to the exact original data.

To find a new starting position, a random value is generated between 0 and the maximum cumulated value. The corresponding index is then located through the binary tree.

Future version of the code might optimize this algorithm further as the nodes of the tree only has to store values as big as the total sum of the light source matrix. If the sum is less than 32 bit it would be possible to cut the memory requirement of the tree in half.

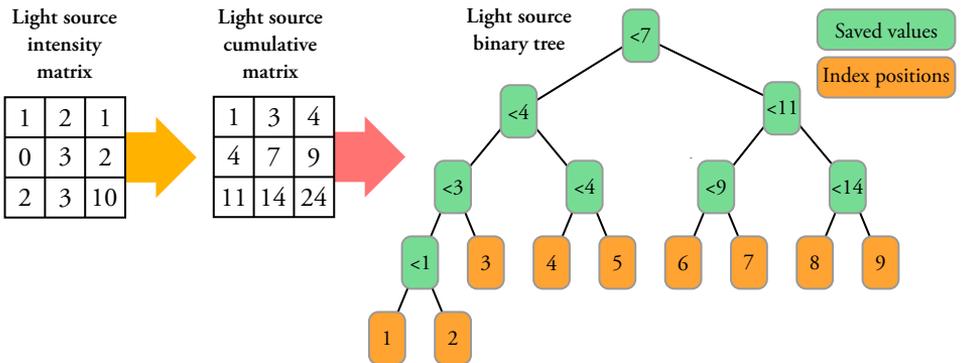


Figure 5.3: The generation of the binary tree used by *Multi-Scattering* to generates photons from a laser beam profile image. The intensity values are cumulated and converted into a binary tree. The tree is used to efficiently find a corresponding index position in the light source matrix.

5.5 Accelerated simulation speed

The scattering medium considered here is a cube of 50 mm side. The incident light is assumed to be unpolarized and monochromatic at a wavelength $\lambda = 600$ nm. The scatterers are non-absorbing spherical water droplets with a refractive index of $n_p = 1.333 - 0.0i$. They are suspended in air that has a refractive index $n_m = 1.000293 - 0.0i$. Two sizes have been considered resulting in two scattering phase functions calculated from the Lorenz-Mie algorithm which is built into the *Multi-Scattering* software. This leads to scattering phase functions with the following properties:

- Fairly isotropic scattering from 0.1 μm diameter scatterers, shown in Fig.3.3(a), resulting to an anisotropy factor $g = 0.05$.
- Highly forward scattering from 10 μm diameter scatterers, shown in Fig.3.3(c), resulting to an anisotropy factor $g = 0.86$.

In addition to the scattering phase function, other parameters have been varied, such as: the optical depth ranging from 2 up to 500 (corresponding to $\mu_e = 0.04 \text{ mm}^{-1}$ and $\mu_e = 10 \text{ mm}^{-1}$ respectively). The main goal of these simulations is to compare the computing time and performance of using a CPU (single thread, multiple threads) versus a GPU (single unit, four units). Three identical computers, see Fig. 1.4, forming a server are used in this study to produce the “GPU results”. The hardware of each computer consists of:

- An Intel CPU (Core i7-7700K).
- Four GPUs from Nvidia (GeForce GTX 1080Ti).

For the case of the “CPU results”, the simulations are run using another computer having a modern CPU processor from AMD (Ryzen Threadripper 2950X 16-Core) where 32 threads can be exploited. For each simulation case, 1 billion photon packets have been launched.

The increase in simulation time as a function of the optical depth is shown in Fig.5.4. As can be seen in Fig.5.4(a), a single CPU thread has very long processing times. This can be problematic when various parameters need to be investigated. Using 32 threads of the AMD processor, the simulation time can be reduced by more than one order of magnitude, as shown from the 20x zoomed results in Fig.5.4(b). In this case, the simulations can be computed in less than 6 minutes at optical depths

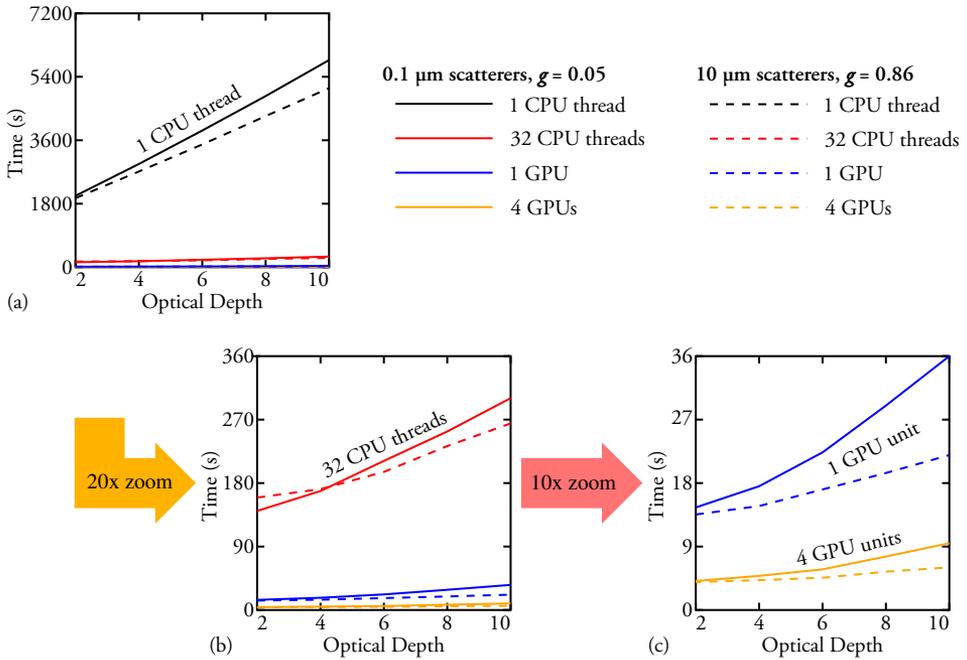


Figure 5.4: Comparison of the simulation execution time as a function of the optical depth for a single CPU thread (in black), 32 CPU thread (in red), a single GPU unit (in blue) and four GPU units (in yellow). Two sizes of scatterer are considered: The continuous lines correspond to 0.1 μm scatterers ($g = 0.05$) and the dashed lines to 10 μm scatterers ($g = 0.86$).

$OD \leq 10$. Despite this significant improvement, the running time of the simulation would become significantly larger for higher OD and would not allow multiple users to share access the software and the servers due to long queuing delays.

It is worth mentioning here that the 32 CPU threads are limited to 16 threads running simultaneously while the other 16 threads are kept available to make a rapid transition when a running thread is stalled due to delays in the data being transferred to the cache. This feature explains why the time improvement with 32 CPU threads over a single thread, is significantly less than 32x. From those results, it can be concluded that the execution of the Monte Carlo simulations using a modern CPU is too time consuming for many simulation cases; especially those related to light propagation through medical tissues where the optical depth can easily reach $OD = 100$.

Figure 5.4(c) shows that a single GPU is a far more suitable computing platform, completing the same tasks in 15 to 30 seconds. Running with four GPUs, available on a single computer, all simulations are completed in less than 10 seconds at $OD \leq 10$. This fast turnaround time enables far more challenging simulations, such as higher optical depths, and allows multiple users to access *Multi-Scattering* with short queuing delays. The results given in Fig.5.5 show the relative gains in speed for the case given

in Fig.5.4.

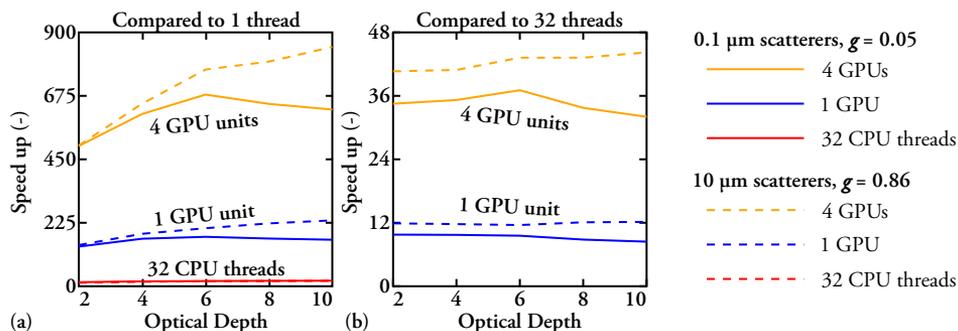


Figure 5.5: Simulation speed as a function of the optical depth comparing a single CPU thread in (a) with 32 CPU threads in (b). Those results are extracted from the data given in Fig.5.4.

Fig.5.6 shows the execution times for higher optical depths, $50 \leq OD \leq 500$. The computational expense of running this case would be prohibitively high for conventional CPUs, so it was only investigated with the four GPU configuration. The results show that simulation times for that arrangement are on the order of minutes. We observe here that the scattering phase function has a noticeable effect on the simulation time. For instance, at $OD = 500$ the execution time increases from 21 to 61 minutes if the scatterers are reduced from 10 to 0.1 μm. These differences increase with the OD as more isotropic scattering photons tend to spread out within the medium, resulting in a larger number of scattering/absorbing interactions.

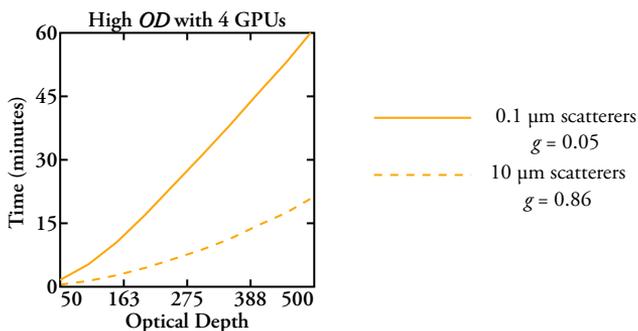


Figure 5.6: The simulation execution time is given for the case of highly scattering media where the OD ranges from 50 to 500. In this case only the GPU results are shown and the time is indicated in minutes.

Chapter 6

Experimental validation

6.1 Comparison for the light intensity distribution

To validate the numerical model used in *Multi-Scattering* comparisons between simulated results and experimental data have been performed¹. The experiment consists in using a cubic glass cell containing a series of aqueous dispersions of either polystyrene spheres or intralipids. The size of the scatters was known and their concentration adjusted in order to accurately define and control the desired optical properties for each scattering phantom. Two validation approaches are presented in this section, one in the space domain and one in the time domain, respectively.

6.1.1 Experimental method

In the first experiment a 30 mm cubic scattering cuvette, containing monodisperse polystyrene spheres suspended in water, was imaged by two cameras. One camera positioned at 90° angle, corresponding to the side scattering detection and the other one positioned at 0°, corresponding to the forward scattering detection. The two camera systems were identical consisting of a 16-bit 5.5 Mpixel sCMOS Andor Zyla cameras mounted Nikon Nikkor objective lens of 50 mm focal distance. The collecting objective lenses were of 37 mm diameter and positioned at 92 mm away from the external faces of the cuvette. The cameras were recording the light scattered by the phantom medium and creating an image the side and forward external faces of the cuvette were respectively with a magnification ratio of 1.17.

¹See paper II and paper VI

The cuvette was illuminated with a CW laser of 473 nm wavelength. The laser is coupled with an optical fiber which vibrates to help mitigating speckle effects. Once light exits the fiber, the beam was expanded and collimated using a spherical lens. Then, only the central part of the beam was selected, resulting to an illumination beam of 3.8 mm diameter with a FWHM of 2.5 mm.

To monitor the incident intensity and verify laser stability, 10% of the light was guided into a power meter using a beam cube splitter. The 90% remaining light was then crossing two neutral density wheels in order to accurately fix the irradiance of the incident beam without modifying its spatial profile. Using this approach, the signal level reaching the cameras was well optimized while avoiding any saturation of the sensor arrays. A picture of this optical arrangement is given in Fig.6.1.

The scattering media investigated here consisted of aqueous dispersions of non-absorbing polystyrene spheres of known size. Two parameters affecting the optical properties of the probed scattering medium have been varied:

- 1) The concentration of the polystyrene microspheres allowing to set up the optical depth to $OD=8$, $OD=10$ and $OD=12$.
- 2) The size of the polystyrene microspheres corresponding to four different diameters: $0.5\ \mu\text{m}$, $2.1\ \mu\text{m}$, $4.9\ \mu\text{m}$ and $21.3\ \mu\text{m}$.

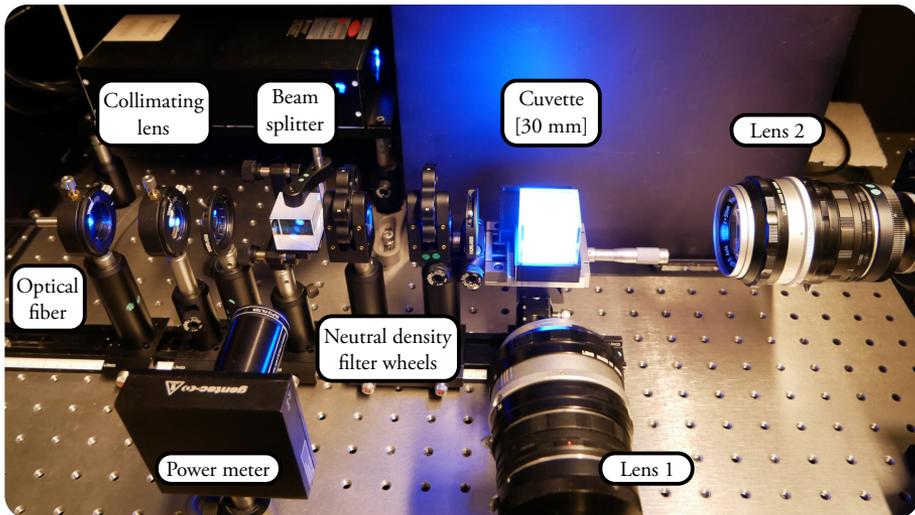


Figure 6.1: Photography of the experimental setup. A 473 nm continuous wave laser beam is guided into an optical fiber. Once exiting the fiber the beam is then collimated and an aperture is selecting the central area to obtain a more homogeneous intensity profile. A beam splitter reflects 10% of the beam into a power meter. Two identical neutral density filter wheels are used to give 36 different levels of light attenuation. After crossing two neutral density filters a 3.8 mm beam is illuminating a scattering dispersion of polystyrene microspheres contained within a 30 mm cubic cuvette. Finally, two identical objectives and sCMOS cameras were used to record the scattered light distribution exiting from the front and side face of the cuvette.

At each a given particle size and concentration the experiment was carried out following the same procedure:

1. A $\lambda = 473$ nm wavelength CW laser was turned on and setup to always run at the same power.
2. The irradiance was attenuated using neutral density filters in order not to saturate the forward scattering detection camera.
3. A reference image of the laser beam was recorded using deionized water only (without any polystyrene spheres) in the cuvette. Note that this image matrix was used as incident light source in the simulation.
4. An aqueous dispersion of monodisperse microspheres of defined size and concentration was created and inserted in the 30 mm cuvette to create the probed scattering medium.
5. The irradiance was fixed again using the neutral density filters to optimize the dynamic range of the forward scattering detection camera.
6. The was adjusted again to improve the dynamic range.
7. Two images were simultaneously recorded, the front view (lens 2) and the side view (lens 1).
8. The intensity was normalized using the data recorded by the power meter and the value of the combined neutral density filters.

6.1.2 Simulation method

Most aspect of the experiment described above could be accurately configured in *Multi-Scattering* allowing the software to closely mimic each experimental case. Each experimental reference image (using distilled water only) was used as an input matrix representing the light source. This allows the simulated medium to be illuminated in the exactly the same fashion as for the experiment. Similarly to the experimental setup virtual lenses collecting photons are positioned 92 mm away from the scattering cube and their diameter was set to 37 mm. To mimic the effective f-number an acceptance angle of 9° was chosen. Additional options for the virtual cameras such as the number of pixels (2160x2160 px) and the magnification was also adjusted for to match with the experiment. The configurations that characterize the scattering medium includes options such as the refractive index of the particles (here: $n_p = 1.5935 - 0.00033i$) and of the surrounding medium (here: $n_p = 1.333 - 0.0i$), the size of the particles

and the wavelength of the light source. All those parameters were needed to deduce the scattering phase function, see Fig. 6.3, and the extinction cross-section for each particle size, using the Lorenz-Mie theory. Finally, *Multi-Scattering* allows to either fix the optical depth and deduce the particle concentration, or the other way around. However, what *Multi-Scattering* cannot mimic is the cuvette itself, the meniscus of the liquid phantom and any phenomena caused by the transitions between liquid and air at the top of the cuvette.

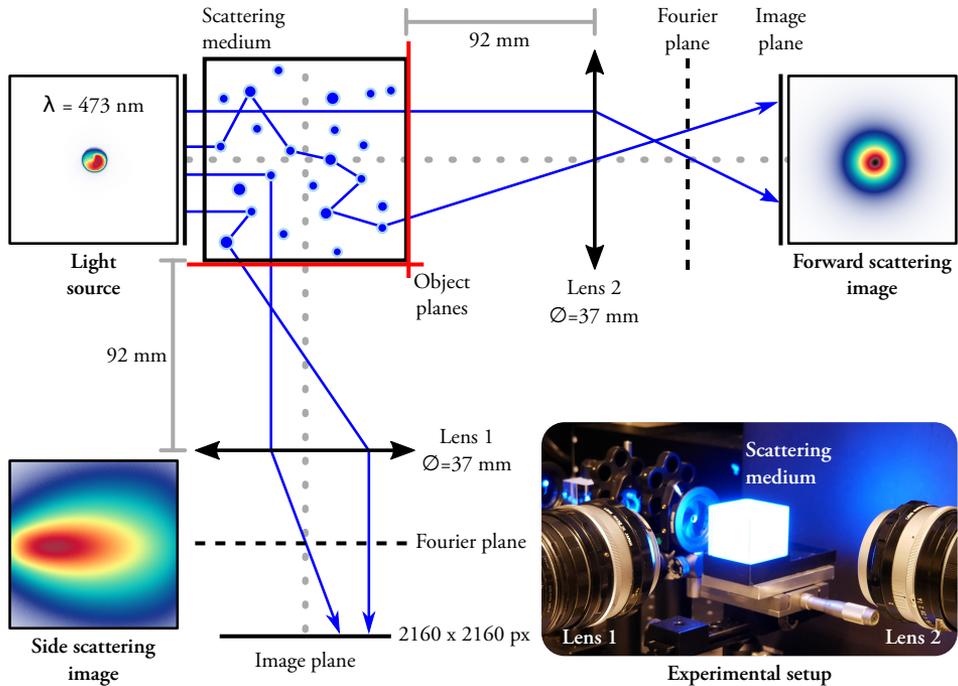
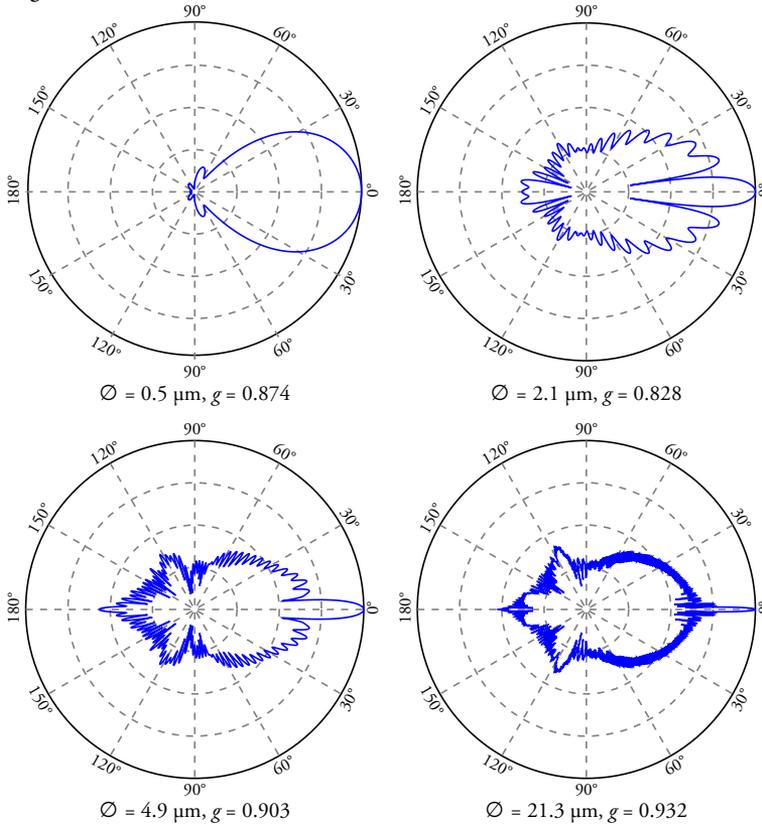


Figure 6.2: Illustration of the simulated setup together with a photo of the corresponding experimental setup. In the simulations a light source matrix is used and the light scattered from a perfect scattering cube is being detected using two discs simulating the collecting lenses 1 and 2. An image of the face of the cube is then constructed on a matrix, having the same pixel resolution and optical magnification than those used in the experiment.

Logarithmic scale:



Linear scale:

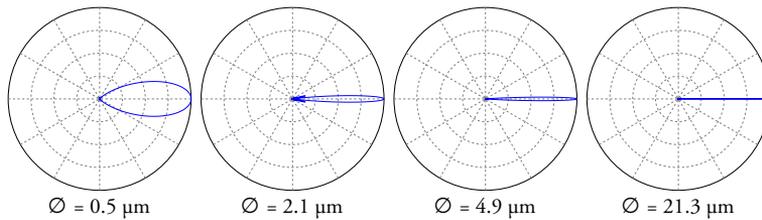


Figure 6.3: The four different scattering phase functions used in the simulations. They were calculated using the Lorenz-Mie theory functionality integrated in *Multi-Scattering*. The larger the particles the more forward scattering are the scattering phase functions.

6.1.3 Results

The experimental and simulated results are compared for both the front and side views in Fig. 6.4, 6.5 and 6.6, corresponding to the three optical depths, $OD = 8$, $OD = 10$ and $OD = 12$. Additionally, each figure includes the four sizes of polystyrene microspheres that have been investigated experimentally: $0.5\ \mu\text{m}$, $2.1\ \mu\text{m}$, $4.9\ \mu\text{m}$ and $21.3\ \mu\text{m}$. To make this comparison quantitative, the detected intensity has been normalized with the photon counts corresponding to the situation without any particles (by experimentally recording the images with water only). This normalization provides a relative intensity (I_f/I_i) indicated on the side of each figure. From this comparison it is seen that the simulation results are most often in a very good agreement with the experimental results both qualitatively and quantitatively.

Larger particles are characterized by being more forward scattering than small particles. This effect is clearly shown as the intensity is larger in the front view for large particles sizes while the opposite is true for the side view. Conversely, smaller particles spread photons more homogeneously. This can be seen on the front face where the smallest particle size spreads photons to a much higher degree than for the three other particle sizes. On the side face this effect is also visible as the shape of the scattered light is more round and spreads more homogeneously in all directions.

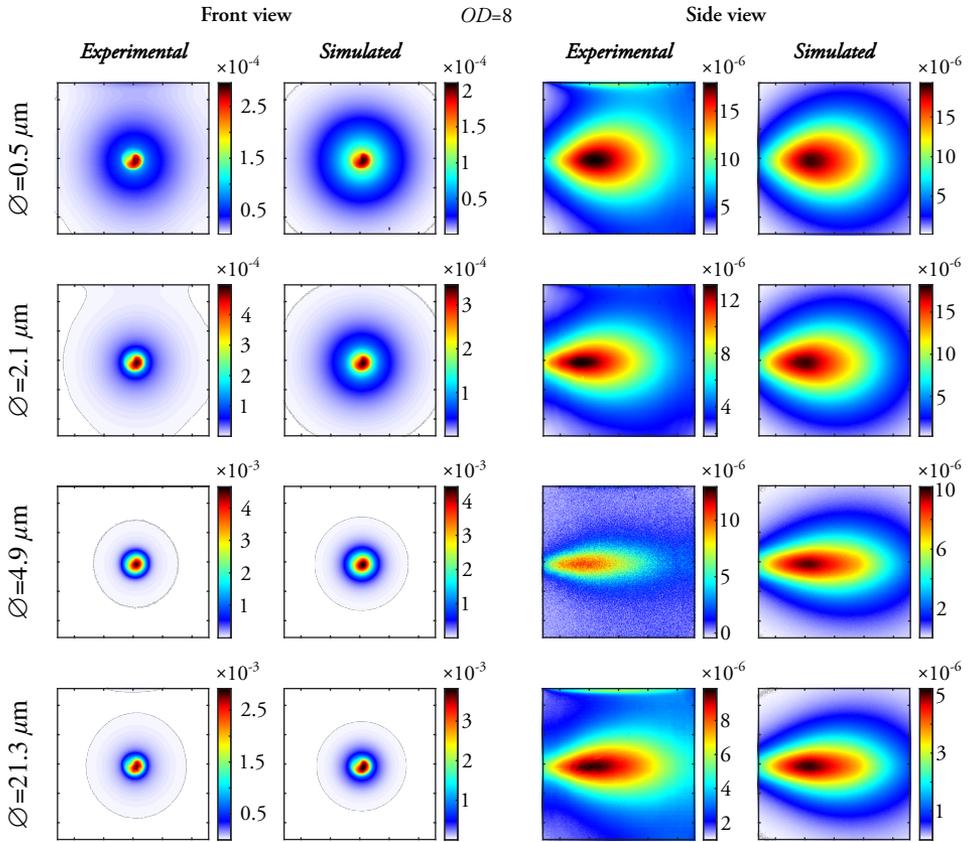


Figure 6.4: Intensity distribution for the front and side face with a comparison between experimental and simulated results. Four different particle sizes are tested with an optical depth of $OD = 8$ and a wavelength of $\lambda = 473 \text{ nm}$. The dimensions of the images are $25 \times 25 \text{ mm}$. In the experimental results (side view and the smallest particle size) the effect caused by the liquid surface is clearly visible at the top.

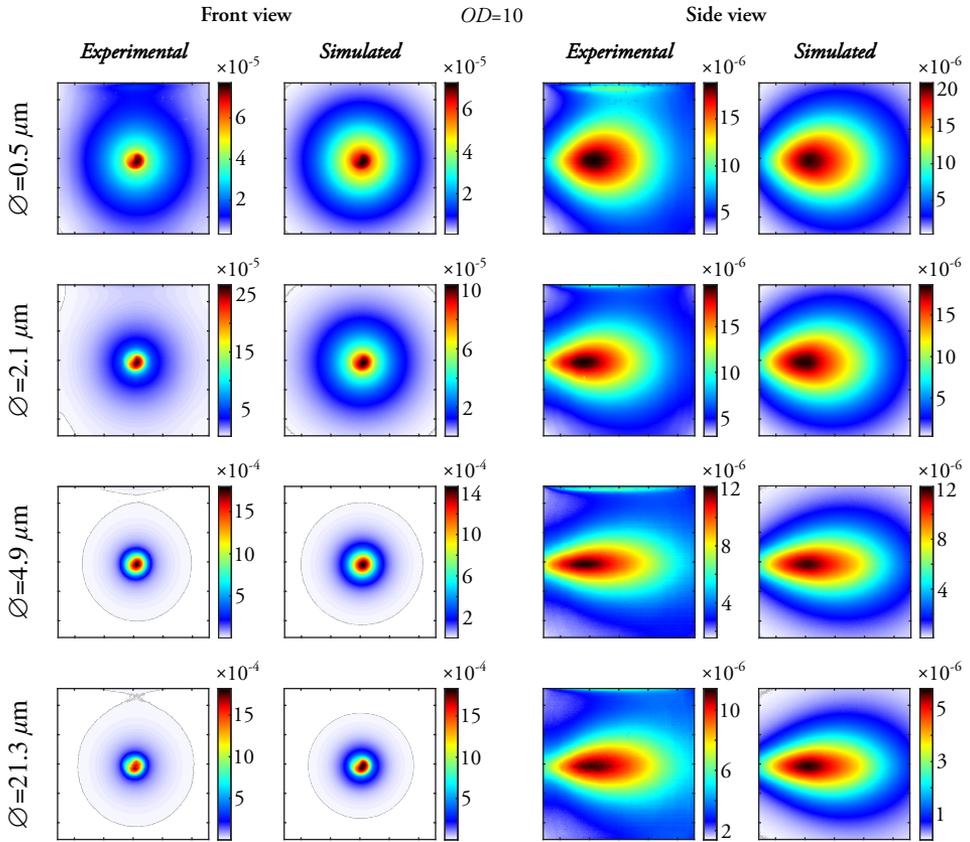


Figure 6.5: Intensity distribution for the front and side face with a comparison between experimental and simulated results. Four different particle sizes are tested with an optical depth of $OD = 10$ and a wavelength of $\lambda = 473 \text{ nm}$. The dimensions of the images are $25 \times 25 \text{ mm}$. In the experimental results (side view and the smallest particle size) the effect caused by the liquid surface is clearly visible at the top.

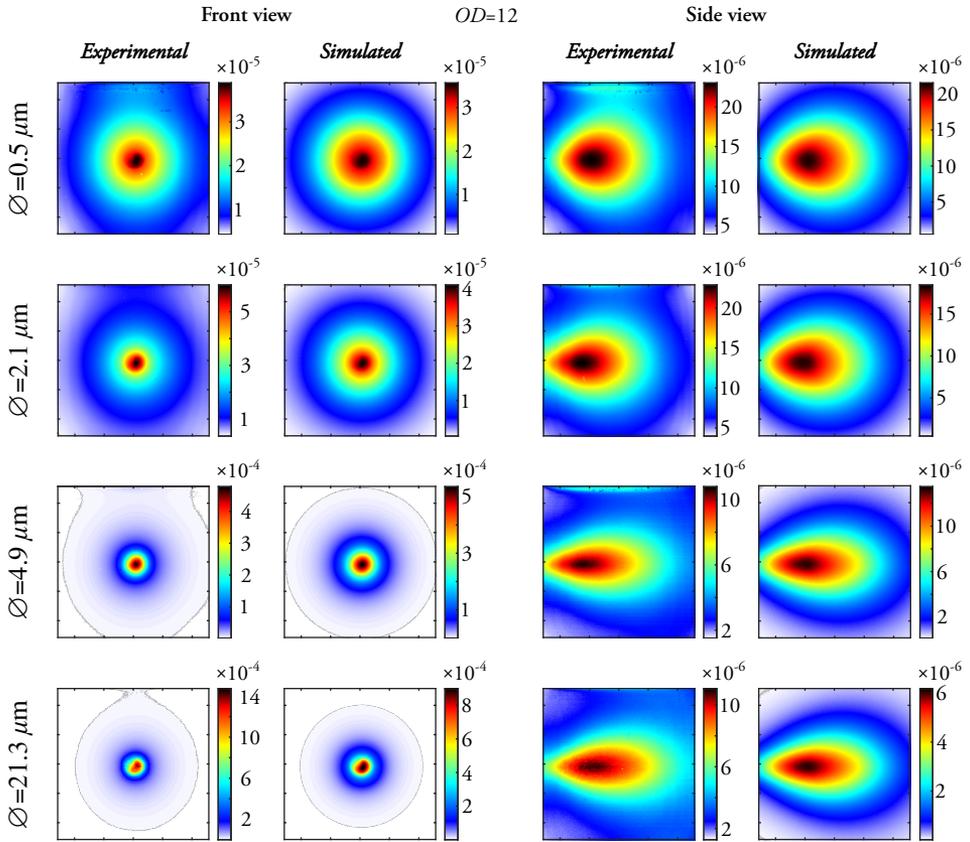


Figure 6.6: Intensity distribution for the front and side face with a comparison between experimental and simulated results. Four different particle sizes are tested with an optical depth of $OD = 12$ and a wavelength of $\lambda = 473 \text{ nm}$. The dimensions of the images are $25 \times 25 \text{ mm}$. In the experimental results (side view and the smaller particle sizes) the effect caused by the liquid surface is clearly visible at the top.

6.2 Comparison for time-of-flight in tissue-like phantoms

The experimental results shown in this sub-section were performed by Brian Angeli [53] and the simulated results were generated using *Multi-Scattering*. The intention was to make a simulation/experiment comparison in the temporal domain, which, is complementary to the spatial domain results presented in the previous sub-section. Here, a short pulse of light is guided by an optical fiber into a phantom medium consisting of an aqueous solution of intralipids. Then, another optical fiber positioned side-way at a fixed distance, is collecting part of the back-scattered photons. Due to the distance traveled by the photons within the phantom medium and the high number of scattering events occurring, the temporal profile of the detected pulse is much larger than for the original pulse. This temporal broadening is relevant as it can be used to deduce the optical properties of the probed tissue-like phantom.

6.2.1 Methodology

The experiments presented here are based on using mixtures of non-absorbing intralipid 20 % (FreseniusKabi, Uppsala, Sweden) and de-ionized water. Various con-

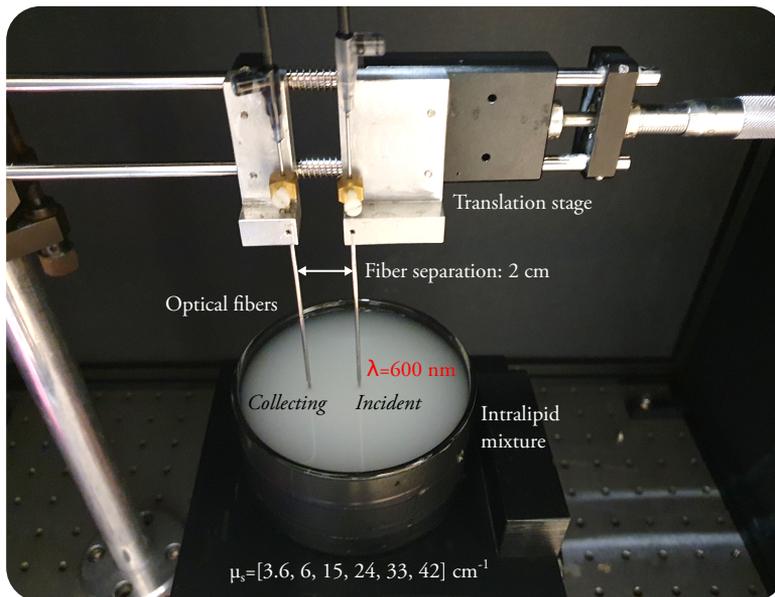


Figure 6.7: Photography of the experimental setup. A 600 nm light pulse of duration 88 ps at FWHM is guided by an optical fiber into a water solution of intralipids. Photons scatter through this phantom medium and those reaching a collecting optical fiber located at 2 cm away from the source are detected. The time-of-flight of the detected pulse provides a direct information on the scattering and absorption coefficients of the probed medium.

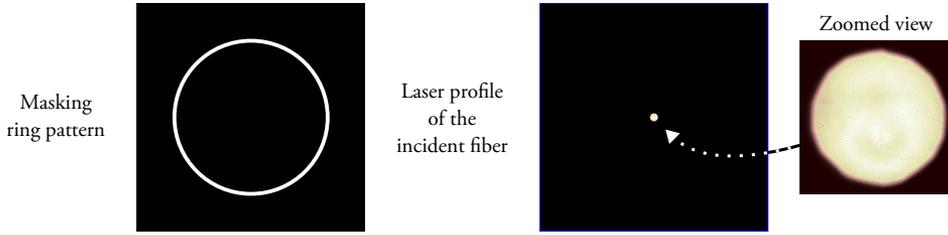


Figure 6.8: Input matrices for the *Multi-Scattering* simulation. An example ring pattern to filter photons at the face of the scattering medium. Black parts blocks light. It was used to simulate a collecting optical fiber. The laser profile was used as the light source for the simulation. It was scaled according to the experimental setup to simulate the incident optical fiber.

concentrations of intralipids were created in order to obtain a scattering coefficient ranging from 3.6 cm^{-1} to 42 cm^{-1} [53]. Each non-absorbing phantom medium was illuminated at $\lambda = 600 \text{ nm}$ using an optical fiber of $600 \mu\text{m}$ diameter. The temporal profile of the incident pulse was recorded and later used in the simulations as an input data characterizing the light source. The collecting fiber was also $600 \mu\text{m}$ in diameter and both fibers were inserted just below the surface level with a separation distance of 2 cm . To simulate light exiting the incident optical fiber and illuminating the phantom medium, a top-hat like laser profile was assumed of dimension corresponding to $600 \mu\text{m}$ diameter. A zoomed view of the intensity profile is given in Fig. 6.8.

The collecting fiber was simulated here using a specific feature provided by *Multi-Scattering* where a mask can be applied on the surface of the simulated volume prior to photon detection. Thus, certain parts of the surface of the scattering cube are blocked (shown in black) while other parts allow light to be detected (shown in white). By positioning the incident light in the center, see the right side of Fig. 6.10, it can be assumed that light will spread evenly in all directions from it, respecting a radial symmetry. Using a ring pattern as a mask (as shown on the left side of Fig. 6.8) instead of a single fiber, better statistics can be obtained in the simulation. Here, a 2 cm radius circle was used to mimic the experimental configuration while increasing the amount of photons being detected.

The only unknown parameter related to the the phantom medium concerns the size distribution of the intralipids. This is impacting the scattering phase function as well as the anisotropy factor that should be used in the simulation. Here, two anisotropy factors were assumed such as $g = 0.765$ and $g = 0.890$ for comparison purpose. The corresponding scattering phase functions are shown in Fig. 6.9. Finally, the simulated volume was of dimension $110 \times 110 \times 110 \text{ mm}$.

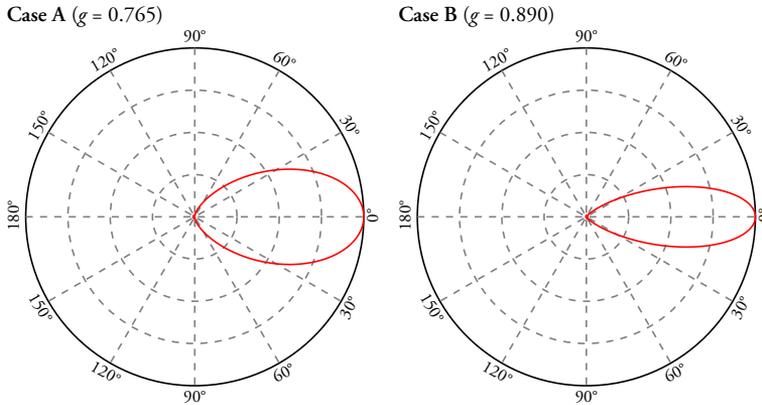


Figure 6.9: The experimental results were compared against the simulation results generated from two different test cases of scattering phase functions with an anisotropy factor $g = 0.765$ for Case A and $g = 0.890$ for Case B.

6.2.2 Results

Multi-Scattering has the ability to track the path of a limited selection of photons. This allows scattering events and the locations to be counted and binned in a two-dimensional representation. Thus, it is possible to analyze the spreading of the scattering events inside the medium. In Fig. 6.10 the results corresponding to *Case B* with $\mu_s = 24 \text{ cm}^{-1}$ are shown. In a) there is no collecting fiber. Shown here are the recorded scattering events of the photons as they travel until they exit the medium. The photons scattering events spread out from the fiber homogeneously with the highest concentration of scattering events close to the area around the incident fiber. In average a photon scatters 191 times and 90% of the photons are back-scattered (exits the surface).

By adding a collecting optical fiber to the simulation the photons tracked can be filtered to only collect and follow the scattering events of the photons reaching it. Those results are shown in b). Here we can see that the scattering events forms a “banana” shaped path to the collecting fiber. Out of all the photons sent, only 0.002% are collected with an average of 507 scattering events per detected photon. Most scattering events tends towards following a path close to the surface of the medium as this is the shortest route. However, scattering events occurs deeper down in the phantom as well. The purpose of a time-of-flight investigation in tissue is to find the optical properties of the tissue some distance underneath the surface. Thus, the the ability of *Multi-Scattering* to track photons can be a valuable asset in quantifying and understanding the reach of the photons within the phantom medium. For example by varying the fiber separation distance different depths can be reached and consequently studied.

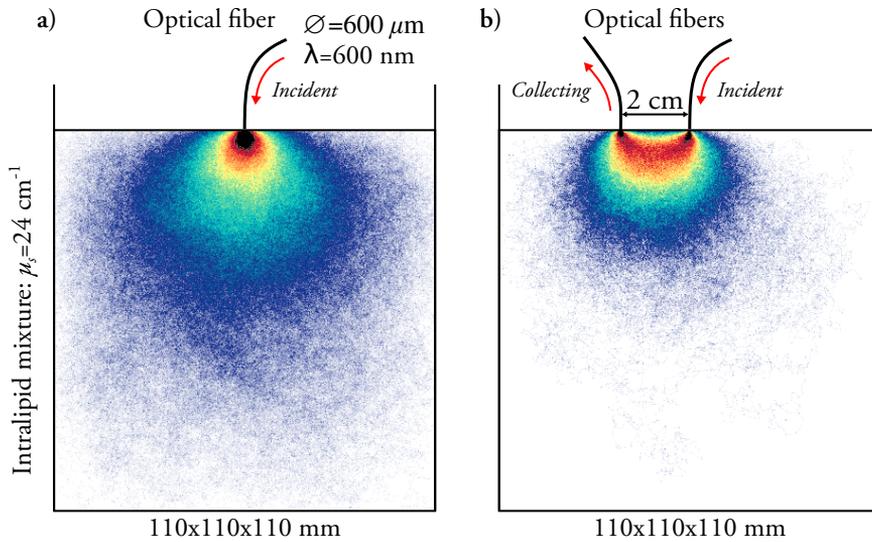


Figure 6.10: A descriptive drawing of the experimental and simulated setup combined with simulated results. Those results represent the scattering events throughout the scattering medium (*Case B*) for a limited amount of photons. In a) all scattering events are shown and in b) only the scattering events of photons reaching the collecting fiber are shown. In order to closely mimic realistic conditions, the simulated collecting fiber is here a single fiber and not the circular mask shown in 6.8.

The temporal domain comparison between experimental and simulated time-of-flight results are presented in Fig. 6.11. In total, the results from six different variations of concentrations are shown. For each experiment the recording of the incident pulse is shown together with the experimental results and the simulated results from *Case A* and *Case B*.

In general, the simulated results follow the general trends of the experimental results. The effect of higher concentrations increases the number of scattering events. With the increase of scattering events, the photons will spread out more. This causes a broadening of the pulse when compared to the initial incident pulse. This effect is visible in both the experimental and in the simulated results with increasing concentration of intralipids.

The comparison of *Case A* with the experimental data reveals that, in the data shown, it is constantly underestimating the results from the six experiments. For the two lower concentrations, *Case B* also underestimates the experimental data. However, for the two concentrations in the middle the results are matching the experimental data. Finally, for *Case B*, with the two higher concentrations, the results are overestimated.

Considering those observations, none of the two cases fully matches the experimental results. Before further analysis can be made, the correct particle size distribution of the intralipids needs to be determined.

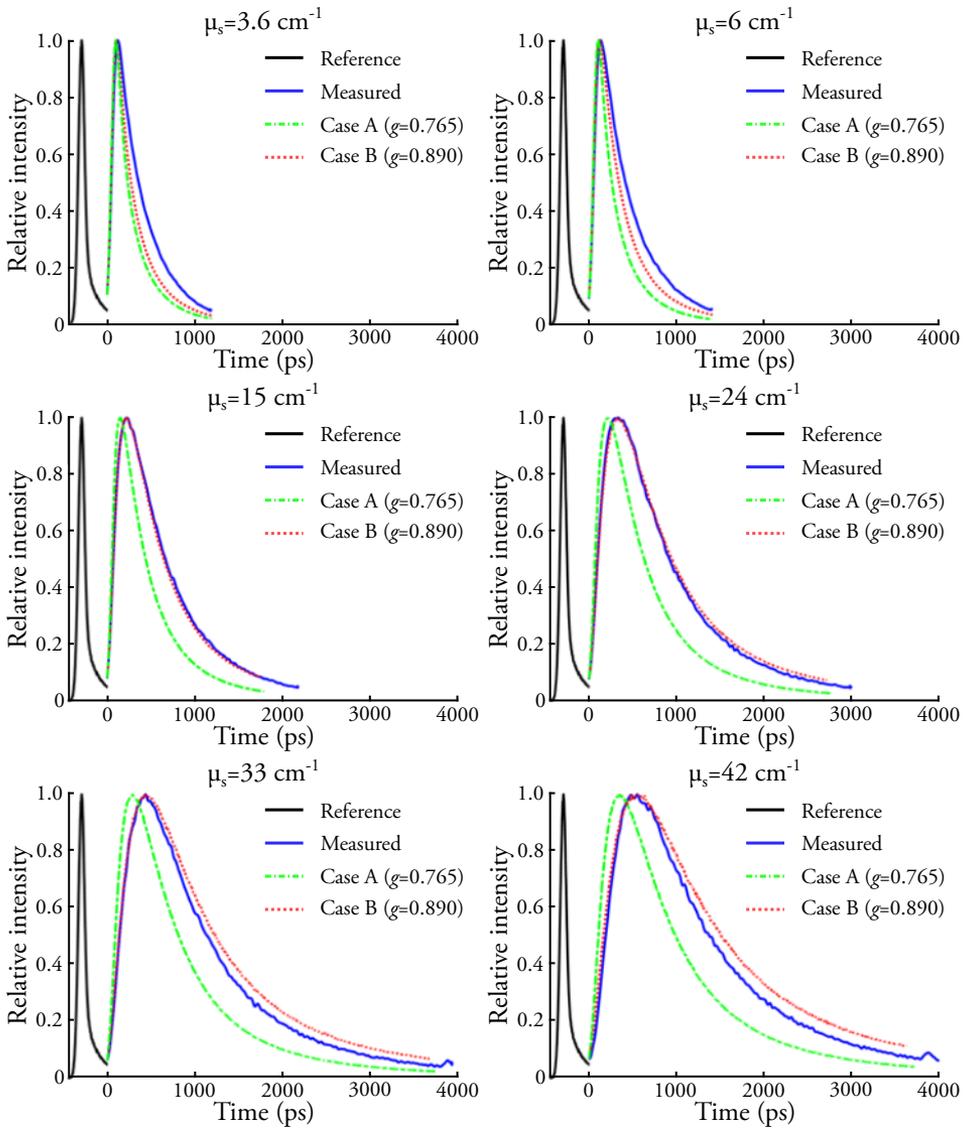


Figure 6.11: Time-of-flight responses to a reference incident pulse with different concentrations of Intralipid. Two cases (A and B) of scattering phase functions were used for the simulated results. These results were published by Brian Angeli [53].

Chapter 7

Conclusions and Outlook

In this thesis a computational tool called *Multi-Scattering* has been designed and created for multi-purpose Monte Carlo simulations of photon transport through scattering and absorbing media. The software is a practical and versatile numerical platform for studying the effects of various light propagation situations and detection parameters. It includes several libraries of input data to simplify configuring and setting up a simulation. Thus, simulation cases can easily be created and quickly modified as desired. To further assist the users in defining the scattering phase functions from spherical scatterers, the Lorenz-Mie theory has been integrated as part of the software. Once run on a local server (currently consisting of 3 computers), located at the Department of Physics at Lund University, the simulation results can easily be viewed and analyzed online. Consequently, *Multi-Scattering* can be used by researchers who are not expert programmers, but wish to numerically model their own scattering systems (e.g. particulate scattering media, medical tissues *etc*) and directly obtain graphical results for various detection cases.

The GPU implementation developed in this thesis for powering and accelerating *Multi-Scattering* allows for significantly faster simulations, which is an important feature when implementing the Monte Carlo method. The use of GPUs allows reducing the computation time by a factor up to 200x in comparison with a single central processing unit thread. By using four graphic cards on a single computer, the simulation speed increases by a factor of 800x. Results have shown that a simulation with an optical depth of $OD = 10$ and for a single voxel is able to be completed within less than 10 seconds. For a challenging simulation involving 1000 voxels at optical depth $OD = 100$, the computational run was completed in under 7 minutes. These short processing times allow the server to be shared among several users with acceptable queuing delays.

Multi-Scattering has been validated by generating and comparing experimental with simulated results. The spatial light intensity distribution was analyzed for various test cases covering different values of particles sizes (from 0.5 μm to 21 μm in diameter) and optical depths (from $OD = 2$ to 17.5), as well as considering the forward and side scattering detection. It was demonstrated that the simulated results were on the overall in very good agreements with the experimental results, both qualitatively and quantitatively. Additional comparisons were involving time-of-flight results within intralipids solutions. It was found from this temporal comparison that an accurate knowledge of the scattering phase function is required in order to reach more reliable computational predictions.

To conclude, *Multi-Scattering* is an extensive software package offering accelerated Monte Carlo simulations using modern computer hardware and easy handling via a powerful online user interface. Therefore it may have a significant impact in the research fields involving light propagation through scattering media. Future improvements include the possibility of running a single simulation on the 3 computers in parallel, allowing the use of 12 GPUs for a given run. Additional computers could also be added to the server, ultimately making *Multi-Scattering* one of the world's fastest online Monte Carlo simulation of photon transport through scattering media.

References

- [1] Felix Scholkmann, Stefan Kleiser, Andreas Jaakko Metz, Raphael Zimmermann, Juan Mata Pavia, Ursula Wolf, and Martin Wolf. A review on continuous wave functional near-infrared spectroscopy and imaging instrumentation and methodology. *NeuroImage*, 85:6 – 27, 2014. ISSN 1053-8119. doi: <https://doi.org/10.1016/j.neuroimage.2013.05.004>. Celebrating 20 Years of Functional Near Infrared Spectroscopy (fNIRS).
- [2] Sergio Fantini and Maria Angela Franceschini. *Chapter 7: Frequency-Domain Techniques for Tissue Spectroscopy and Imaging*. 2001.
- [3] D T Delpy, M Cope, P van der Zee, S Arridge, S Wray, and J Wyatt. Estimation of optical pathlength through tissue from direct time of flight measurement. *Physics in Medicine and Biology*, 33(12):1433–1442, dec 1988. doi: 10.1088/0031-9155/33/12/008.
- [4] Alessandro Torricelli, Davide Contini, Antonio Pifferi, Matteo Caffini, Rebecca Re, Lucia Zucchelli, and Lorenzo Spinelli. Time domain functional NIRS imaging for human brain mapping. *NeuroImage*, 85:28 – 50, 2014. ISSN 1053-8119. doi: <https://doi.org/10.1016/j.neuroimage.2013.05.106>. Celebrating 20 Years of Functional Near Infrared Spectroscopy (fNIRS).
- [5] Erin M. Buckley, Ashwin B. Parthasarathy, P. Ellen Grant, Arjun G. Yodh, and Maria Angela Franceschini. Diffuse correlation spectroscopy for measurement of cerebral blood flow: future prospects. *Neurophotonics*, 1(1):1 – 7, 2014. doi: 10.1117/1.NPh.1.1.011009.
- [6] Sune Svanberg. Gas in scattering media absorption spectroscopy - from basic studies to biomedical applications. *Laser & Photonics Reviews*, 7(5):779–796, 2013. doi: 10.1002/lpor.201200073.
- [7] Edouard Berrocal. *Multiple scattering of light in optical diagnostics of dense sprays and other complex turbid media*. PhD thesis, Cranfield University, 2006.

- [8] Joakim Jönsson. Acceleration and optimisation of a Monte Carlo code for light propagation in sprays and other scattering media, 2011. Student Paper.
- [9] Joakim Jönsson and Edouard Berrocal. Multi-scattering software: part i: online accelerated monte carlo simulation of light transport through scattering media. *Opt. Express*, 28(25):37612–37638, Dec 2020. doi: 10.1364/OE.404005.
- [10] Michael I. Mishchenko, Li Liu, Daniel W. Mackowski, Brian Cairns, and Gordon Videen. Multiple scattering by random particulate media: exact 3D results. *Opt. Express*, 15(6):2822–2836, Mar 2007. doi: 10.1364/OE.15.002822.
- [11] D. G. Collins and M. B. Wells. Monte Carlo codes for the study of light transport in the atmosphere, 1965.
- [12] Gilbert N. Plass and George W. Kattawar. Monte Carlo calculations of light scattering from clouds. *Appl. Opt.*, 7(3):415–419, Mar 1968. doi: 10.1364/AO.7.000415.
- [13] George W. Kattawar and Gilbert N. Plass. Radiance and polarization of multiple scattered light from haze and clouds. *Appl. Opt.*, 7(8):1519–1527, Aug 1968. doi: 10.1364/AO.7.001519.
- [14] Gilbert N. Plass and George W. Kattawar. Radiative transfer in an Atmosphere–Ocean system. *Appl. Opt.*, 8(2):455–466, Feb 1969. doi: 10.1364/AO.8.000455.
- [15] George W. Kattawar and Gilbert N. Plass. Time of flight lidar measurements as an ocean probe. *Appl. Opt.*, 11(3):662–666, Mar 1972. doi: 10.1364/AO.11.000662.
- [16] Clarence J. Funk. Multiple scattering calculations of light propagation in ocean water. *Appl. Opt.*, 12(2):301–313, Feb 1973. doi: 10.1364/AO.12.000301.
- [17] Edward A. Bucher. Computer simulation of light pulse propagation for communication through thick clouds. *Appl. Opt.*, 12(10):2391–2400, Oct 1973. doi: 10.1364/AO.12.002391.
- [18] B. C. Wilson and G. Adam. A Monte Carlo model for the absorption and flux distributions of light in tissue. *Medical Physics*, 10(6):824–830, 1983. doi: 10.1118/1.595361.
- [19] S. A. Prahl. *Light Transport in Tissue*. PhD thesis, University of Texas at Austin, 1988.
- [20] M. Keijzer. *Light transport for medical laser treatments*. PhD thesis, Technische Univ., Delft (Netherlands)., December 1993.

- [21] Marleen Keijzer, Steven L. Jacques, Scott A. Prahl, and Ashley J. Welch. Light distributions in artery tissue: Monte Carlo simulations for finite-diameter laser beams. *Lasers in Surgery and Medicine*, 9(2):148–154, 1989. doi: 10.1002/lsm.1900090210.
- [22] S. A. Prahl. A Monte Carlo model of light propagation in tissue. In Gerhard J. Mueller, David H. Sliney, and Roy F. Potter, editors, *Dosimetry of Laser Radiation in Medicine and Biology*, volume 10305, pages 105 – 114. International Society for Optics and Photonics, SPIE, 1989. doi: 10.1117/12.2283590.
- [23] S. T. Flock, M. S. Patterson, B. C. Wilson, and D. R. Wyman. Monte Carlo modeling of light propagation in highly scattering tissues. I. model predictions and comparison with diffusion theory. *IEEE Transactions on Biomedical Engineering*, 36(12):1162–1168, 1989.
- [24] Lihong Wang, Steven L. Jacques, and Liqiong Zheng. MCML-Monte Carlo modeling of light transport in multi-layered tissues. *Computer Methods and Programs in Biomedicine*, 47(2):131 – 146, 1995. ISSN 0169-2607. doi: [https://doi.org/10.1016/0169-2607\(95\)01640-F](https://doi.org/10.1016/0169-2607(95)01640-F).
- [25] Igor V Meglinski and Stephen J Matcher. Quantitative assessment of skin layers absorption and skin reflectance spectra simulation in the visible and near-infrared spectral regions. *Physiological Measurement*, 23(4):741–753, oct 2002. doi: 10.1088/0967-3334/23/4/312.
- [26] Dmitry Y. Churmakov, Igor Vladislavovich Meglinski, and Douglas A. Greenhalgh. Amending of fluorescence sensor signal localization in human skin by matching of the refractive index. *Journal of Biomedical Optics*, 9(2):339 – 346, 2004. doi: 10.1117/1.1645796.
- [27] P. Bruscaaglioni, A. Ismaelli, G. Zaccanti, and L. Pantani. Modified Monte-Carlo Method To Evaluate Multiple Scattering Effects On Lightbeam Transmission Through A Turbid Atmosphere. In M. John Colles and D. William Swift, editors, *Max Born Centenary Conf*, volume 0369, pages 164 – 173. International Society for Optics and Photonics, SPIE, 1983. doi: 10.1117/12.934363.
- [28] P. Bruscaaglioni, P. Donelli, A. Ismaelli, and G. Zaccanti. A numerical procedure for calculating the effect of a turbid medium on the MTF of an optical system. *Journal of Modern Optics*, 38(1):129–142, 1991. doi: 10.1080/09500349114550161.
- [29] P. Donelli, P. Bruscaaglioni, A. Ismaelli, and G. Zaccanti. Experimental validation of a Monte Carlo procedure for the evaluation of the effect of a turbid medium

- on the point spread function of an optical system. *Journal of Modern Optics*, 38 (11):2189–2201, 1991. doi: 10.1080/09500349114552271.
- [30] P. Brusaglioni, A. Ismaelli, and G. Zaccanti. Monte-Carlo calculations of LIDAR returns: Procedure and results. *Applied Physics B*, 60(4):325–329, 1995. ISSN 1432-0649. doi: 10.1007/BF01082266.
- [31] Jean-Philippe Briton, Bruno Maheu, Gérard Gréhan, and Gérard Gouesbet. Monte Carlo Simulation of Multiple Scattering in arbitrary 3-D geometry. *Particle & Particle Systems Characterization*, 9(104):52–58, 1992. doi: 10.1002/ppsc.19920090109.
- [32] C. Rozé, B. Maheu, G. Gréhan, and J. Ménard. Evaluations of the sighting distance in a foggy atmosphere by Monte Carlo simulation. *Atmospheric Environment*, 28(5):769 – 775, 1994. ISSN 1352-2310. doi: [https://doi.org/10.1016/1352-2310\(94\)90235-6](https://doi.org/10.1016/1352-2310(94)90235-6). Conference on visibility and fine particles.
- [33] B. Roysam, A. Cohen, P. Getto, and P. Boyce. A numerical approach to the computation of light propagation through turbid media: Application to the evaluation of lighted exit signs. In *Conference Record of the 1991 IEEE Industry Applications Society Annual Meeting*, pages 1876–1882 vol.2, Sep. 1991. doi: 10.1109/IAS.1991.178117.
- [34] Edouard Berrocal, Dmitry Y. Churmakov, Vadim P. Romanov, Mark C. Jermy, and Igor V. Meglinski. Crossed source–detector geometry for a novel spray diagnostic: Monte Carlo simulation and analytical results. *Appl. Opt.*, 44(13): 2519–2529, May 2005. doi: 10.1364/AO.44.002519.
- [35] Edouard Berrocal, Igor Meglinski, and Mark Jermy. New model for light propagation in highly inhomogeneous polydisperse turbid media with applications in spray diagnostics. *Opt. Express*, 13(23):9181–9195, Nov 2005. doi: 10.1364/OPEX.13.009181.
- [36] Edouard Berrocal, David L. Sedarsky, Megan E. Paciaroni, Igor V. Meglinski, and Mark A. Linne. Laser light scattering in turbid media Part I: Experimental and simulated results for the spatial intensity distribution. *Opt. Express*, 15(17): 10649–10665, Aug 2007. doi: 10.1364/OE.15.010649.
- [37] Edouard Berrocal, David L. Sedarsky, Megan E. Paciaroni, Igor V. Meglinski, and Mark A. Linne. Laser light scattering in turbid media Part II: Spatial and temporal analysis of individual scattering orders via Monte Carlo simulation. *Opt. Express*, 17(16):13792–13809, Aug 2009. doi: 10.1364/OE.17.013792.

- [38] Erik Alerstam, Tomas Svensson, and Stefan Andersson-Engels. Parallel computing with graphics processing units for high-speed Monte Carlo simulation of photon migration. *Journal of Biomedical Optics*, 13(6):1 – 3, 2008. doi: 10.1117/1.3041496.
- [39] Qianqian Fang and David A. Boas. Monte Carlo simulation of photon migration in 3D turbid media accelerated by Graphics Processing Units. *Opt. Express*, 17(22):20178–20190, Oct 2009. doi: 10.1364/OE.17.020178.
- [40] Erik Alerstam, William Chun Yip Lo, Tianyi David Han, Jonathan Rose, Stefan Andersson-Engels, and Lothar Lilge. Next-generation acceleration and code optimization for light transport in turbid media using GPUs. *Biomed. Opt. Express*, 1(2):658–675, Sep 2010. doi: 10.1364/BOE.1.000658.
- [41] H Shen and G Wang. A tetrahedron-based inhomogeneous monte carlo optical simulator. *Physics in Medicine and Biology*, 55(4):947–962, jan 2010. doi: 10.1088/0031-9155/55/4/003.
- [42] Qianqian Fang. Mesh-based Monte Carlo method using fast ray-tracing in Plücker coordinates. *Biomed. Opt. Express*, 1(1):165–175, Aug 2010. doi: 10.1364/BOE.1.000165.
- [43] Alexander Doronin and Igor Meglinski. Online object oriented Monte Carlo computational tool for the needs of biomedical optics. *Biomed. Opt. Express*, 2(9):2461–2469, Sep 2011. doi: 10.1364/BOE.2.002461.
- [44] Jeffrey Cassidy, Lothar Lilge, and Vaughn Betz. FullMonte: a framework for high-performance Monte Carlo simulation of light through turbid media with complex geometry. In Adam P. Wax and Vadim Backman, editors, *Biomedical Applications of Light Scattering VII*, volume 8592, pages 41 – 54. International Society for Optics and Photonics, SPIE, 2013. doi: 10.1117/12.2002933.
- [45] Tanner Young-Schultz, Stephen Brown, Lothar Lilge, and Vaughn Betz. Full-MonteCUDA: a fast, flexible, and accurate GPU-accelerated Monte Carlo simulator for light propagation in turbid media. *Biomed. Opt. Express*, 10(9):4711–4726, Sep 2019. doi: 10.1364/BOE.10.004711.
- [46] Qianqian Fang and Shijie Yan. Graphics processing unit-accelerated mesh-based Monte Carlo photon transport simulations. *Journal of Biomedical Optics*, 24(11): 1 – 6, 2019. doi: 10.1117/1.JBO.24.11.115002.
- [47] L. Wang, P. P. Ho, C. Liu, G. Zhang, and R. R. Alfano. Ballistic 2-D imaging through scattering walls using an ultrafast optical Kerr gate. *Science*, 253(5021): 769–771, 1991. ISSN 00368075, 10959203.

- [48] David Sedarsky, Edouard Berrocal, and Mark Linne. Quantitative image contrast enhancement in time-gated transillumination of scattering media. *Opt. Express*, 19(3):1866–1883, Jan 2011. doi: 10.1364/OE.19.001866.
- [49] Edouard Berrocal, Sven-Göran Pettersson, and Elias Kristensson. High-contrast imaging through scattering media using structured illumination and Fourier filtering. *Opt. Lett.*, 41(23):5612–5615, Dec 2016. doi: 10.1364/OL.41.005612.
- [50] L. G. Henyey and J. L. Greenstein. Diffuse radiation in the galaxy. *Astrophysical Journal*, 93:70–83, January 1941. doi: 10.1086/144246.
- [51] C.F. Bohren and D.R. Huffman. *Absorption and Scattering of Light by Small Particles*. Wiley, 1983. ISBN 9780471293408.
- [52] E Berrocal, I V Meglinski, D A Greenhalgh, and M A Linne. Image transfer through the complex scattering turbid media. *Laser Physics Letters*, 3(9):464–467, sep 2006. doi: 10.1002/lapl.200610035.
- [53] Brian Angeli. Photon time-of-flight in tissue mimicking phantoms for the validation of a monte carlo software, 2019. Student Paper.

Summary of papers

Author contributions

Paper I: *Multi-Scattering* software: part I: online accelerated Monte Carlo simulation of light transport through scattering media

This article is the first part of an article series introducing the *Multi-Scattering* software that I have created. This part I is describing the algorithm and the features of the software. It starts by introducing the history of Monte Carlo simulations including light propagation through particulate media (atmospheric layers, clouds) and through biomedical tissues (skin layers, blood). After the introduction a general description as well as the major features of *Multi-Scattering* are provided. Then, the terminology and definitions of the physical parameters used in the numerical model are given. The next chapter focuses and describes the successive steps of the Monte Carlo code. Then, an explanation of how GPUs are applied to significantly reduce the computing running time is given and results of GPU accelerated simulations are provided showing improvements reaching up to 800 times speed-up. Additionally the performance for different types of scattering media (including various scattering phase functions and optical depths) are provided. The final section shows a complex 3D simulation example of laser light propagation through a portion of a human head.

I have built and designed, on my own, the three web interfaces that are needed to connect to and use the online Multi-Scattering software. Those interfaces are used to 1) Create and run simulations, 2) Manage the user account and finally 3) Directly visualize and display the results. I have re-written, modified and improved the original Monte Carlo C-code developed by Edouard Berrocal in order to use the CUDA platform. Using this platform the software I was able to parallelized the code so it could run on Nvidia GPU cards. I have gone through several iterations to solve many bottle neck problems which were affecting the performance of the model. To run Multi-Scattering I have created a server at Lund University consisting of 3 computers having 4 Nvidia graphic cards each. I have further

parallelized the code so that each simulation can be run on multiple GPUs. Finally, while the original code was focused on particulate scattering media (e.g. spray systems) I have broaden its application for more challenging scattering media such as biomedical tissues (e.g. human head). I have created all the figures, I have been involved in the analysis of the data and I have participated to a large extent in the writing of the article.

Paper II: *Multi-Scattering* software: part II: experimental validation for the light intensity distribution

To be submitted

This article is the second part of an article series introducing the *Multi-Scattering* software that I have created. This part II focuses on the experimental validation of the Monte Carlo model used in *Multi-Scattering*. A rigorous comparisons between simulated and experimental results is presented by analyzing the spatial intensity distribution. The comparisons covers different optical depths (ranging from $OD = 2$ to $OD = 17.5$) and size of scattering particles (0.5 μm , 2 μm and 5 μm in diameter). The phantom media used here are aqueous dispersions of polystyrene spheres.

The main author, David Frantz has recorded the experimental data. I have been involved in the creation and the generation of the simulated results. All the simulated data provided in this article have been obtained through the Multi-Scattering software which I have developed. For this article I have also developed numerical image magnification and Fourier filtering to obtain better comparison with experimental results.

Paper III: Comparison between two-phase and one-phase SLIPI for instantaneous imaging of transient sprays

This article presents a comparison of two approaches used for imaging spray systems. The goal of the article is to analyze and quantify the benefits of two-phase and one-phase SLIPI and to compare it with conventional planar imaging. This has been done numerically using Monte Carlo simulation were the processed images were compared with single light scattering images. Additionally the techniques were compared experimentally with averaged LIF images of a direct-injection spark-ignition (DISI) ethanol spray system.

I have generated all Monte Carlo simulation results and created the corresponding figures (Fig. 2, Fig. 3 and Fig. 12). For this article the modulated light sheet was incorporated as a light source and imaged on the side. I have implemented those specific features in Multi-Scattering.

Paper iv: Simulation of light scattering and imaging of spray systems using the open-access software “Multi-Scattering”

This paper presents the use of the *Multi-Scattering* software for spray applications where photons are tracked through a cloud of droplets. During the simulation process, each scattering event is described by the Lorenz-Mie theory to accurately simulate light interaction between spherical droplets and plane waves. This paper shows how ligaments are blurred in a spray system and how light intensity is distributed in the Fourier plane of a collecting lens.

I have created all the simulations, I have created all the figures, I have been involved in the analysis of the data and I have participated to a large extent in the writing of the article.

Paper v: Fast and sensitive diffuse correlation spectroscopy with highly parallelized single photon detection

Diffuse correlation spectroscopy (DCS) technique is a well-established method that measures rapid changes in scattered coherent light to identify blood flow and functional dynamics within a tissue. This article presents a novel approach using an integrated SPAD array to enhance sensitivity as compared to a single or a small number of SPADs. Using a tissue-like phantom the article demonstrates the possibility of accurate detection in the mm-scale perturbations under 1 cm of tissue. Further on an *in-vivo* study is also presented in the article.

My contribution for this article is located in the supplementary material (note 1, Fig. 1). However, the simulations I provided using Multi-Scattering was useful to quantify one of the limiting factors of DCS. In order to generate those banana-shape results from inside the scattering medium I had to develop and integrate a new tool for Multi-Scattering to be able to save and visualize the simulated data.

Paper vi: *Multi-Scattering* software: part III: Comparison with experimental results for photon time-of-flight in tissue mimicking phantoms”

Manuscript in preparation

This work focuses on the validation of *Multi-Scattering* in the time domain using Photon Time-of-Flight Spectroscopy (PTOFS) on tissue-like phantoms. In this article a series of phantoms are studied with varying optical properties (absorption and scattering coefficients). Those experimental results are then compared with simulated results produced by the *Multi-Scattering* software.

I have developed new capabilities for Multi-Scattering to record the temporal resolved data needed for the comparison with the experimental. I have also developed a new mask feature to mimic light collection by an optical fiber such as used in the experiment. Finally I have assisted Brian Angeli with the setup of the simulations and correcting several problems related to the simulation work.

<https://multi-scattering.com>



Faculty of Engineering
Department of Physics
Division of Combustion Physics

Lund Reports on Combustion Physics: LRCP-233
ISRN LUTFD2/TFCP 233 SE
ISBN 978-91-7895-897-9
ISSN 1102-8718

