

Consistent Monte Carlo methods for non-linear applications in light transport

A Thesis

Submitted to the Faculty

in partial fulfillment of the requirements for the
degree of

Doctor of Philosophy

in

Computer Science

by Zackary Misso

Guarini School of Graduate and Advanced Studies

Dartmouth College

Hanover, New Hampshire

November 2023

Examining Committee:

Wojciech Jarosz, Chair

Peter Winkler

Bo Zhu

Iliyan Georgiev

F. Jon Kull, Ph.D.

Dean of the Guarini School of Graduate and Advanced Studies

Abstract

The study of light transport focuses on describing the propagation of light from emitters to sensors through accurately describing the interactions light can undergo with everything in between. Physically-based rendering is the process of applying the laws of light transport to formulate practical algorithms which simulate the flow of light for the purpose of synthesizing images of virtual environments.

Unfortunately, there are very few interesting scene configurations which can be computed analytically. Instead, modern solutions predominantly rely on Monte Carlo integration to stochastically estimate the transfer of light since the process is both *unbiased* and *consistent*. Meaning, it is expected to give us the correct result, and if we took the average of infinitely many estimates we would obtain the correct answer.

However, the ideal properties of Monte Carlo integration are not guaranteed for all problems. In the presence of non-linear perturbations, Monte Carlo integration will almost always result in *incorrect* solutions. Unfortunately, applications with non-linear perturbations appear throughout light transport from simulating the interactions between light and participating media, to rendering complicated visual phenomena like caustics, to even modern applications in differentiable rendering.

In this dissertation we focus on formulating consistent solutions for general non-linear problems within light transport. We achieve this by devising a general recipe outlining for both scientists and practitioners alike how to address and account for the many difficulties which can arise when estimating these difficult problems. This involves discussing various ways of reconstructing non-linear problems into alternative forms which are amenable to unbiased Monte Carlo integration. We then review ways of verifying that these unbiased estimators are efficient enough to be employed in practice. When this is not the case, we review and introduce methods for constructing biased, but still consistent, solutions. Through the application of our recipe, we introduce the first general unbiased rendering algorithm capable of rendering non-classical participating media, the first truly unbiased photon-mapping algorithm for rendering razor sharp caustics, and various consistent techniques for efficiently rendering participating media comprised of unknown black box densities.

Acknowledgments

Stating that the journey to acquire my PhD has been long and arduous is an understatement to the mental and emotional toils I have undergone during this chapter of my life. However, what has made it bearable, even in the worst of times, has been the people that have consistently been by my side.

The amount of gratitude that I owe certain friends and family can never be fully conveyed in a short acknowledgements section, but I will do my best. First of all, I would like to thank my best friend Garrett Baker. The nights spent discussing guitar, bass, memes, and theorizing about the Xenoblade series were always the "bestest" distractions (inside joke). During Covid the weekend nights spent playing Smash Bros. with Jonathan Verghese and Garrett as well as my time spent online with Ian Neal and Siddarth Kumar (Sidd) were the only things that kept me sane during the months of social isolation. The vacations Ian, Sidd, and I went on to Indianapolis, Niagara Falls, Arcadia, and Montreal alongside Ian's wedding were probably some of the highlights of these past six years.

No one understands and relates to your struggles more than those who sit beside you. I would like to thank my former labmate Benedikt Bitterli for being an awesome person and exceptional role model. During my time at Dartmouth, Dario Seyb, Alex Dejournei, and Kate Salesin were awesome labmates and friends.

I cannot in good conscience write an acknowledgements section without mentioning my advisor Wojciech Jarosz. During my time at Dartmouth, Wojciech has been an outstanding source of knowledge whose door has always been open whenever I needed assistance. The rigor at which Wojciech produces quality research has always been awe-inspiring to me. The amount of effort and level of presentation quality that he instills into his students is something that I will continue to benefit from even if my future leads me outside of Academia.

During my PhD I have had the privilege of collaborating with many amazing researchers and engineers throughout my various internships and research projects. In no particular order, I would like to thank Iliyan Georgiev, Karl Li, Brent Burley, Daniel Teece, Jaroslav Krivanek, Toshiya Hachisuka, and Derek Nowrouzezahrai.

I am immensely grateful to my dissertation committee consisting of Iliyan Georgiev, Bo Zhu, Peter Winkler, and Wojciech Jarosz who took the time to read and evaluate my dissertation.

I would like to additionally thank my undergraduate advisors, Cem Tutum and Etienne Vouga, as well as my first computer science teachers from Cedar Park High School, Galen Goodreau and Cheri Whalen.

I would also like to offer my sincerest gratitude to the Facebook (now Meta) PhD Fellowship program for awarding me with a PhD fellowship that aided in funding my work.

Lastly, I would like to thank both my grandparents Gay'Neil Misso and Bob Misso for supporting me through the final leg of my PhD and always being there for moral support, as well as my dad, Robb Misso.

Publications

Portions of this dissertation are based off of published work I have completed previously in collaboration with other authors. I reproduce portions of these publications within this dissertation with explicit permission from my co-authors. Which chapters are based on which published work as well as my contribution to each paper are listed below:

- ▶ Chapter 6 is mostly based on my previous publication:

Iliyan Georgiev, Zackary Misso, Toshiya Hachisuka, Derek Nowrouzezahrai, Jaroslav Křivánek, Wojciech Jarosz. Integral formulations of volumetric transmittance. ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia), 38(6), November 2019.

I was co-primary investigator for this publication. This chapter will also include discussions regarding a later discovery which was included in my later publication:

Zackary Misso, Yining Karl Li, Brent Burley, Daniel Teece, Wojciech Jarosz. Progressive null-tracking for volumetric rendering. In: ACM SIGGRAPH Conference Papers. July 2023.

in which I was the primary investigator and primary author of. Some of the notation, and the relation to debiasing, the initial value problem derivation, and estimation of limits are based off of later discoveries from my previous publication:

Zackary Misso, Benedikt Bitterli, Iliyan Georgiev, Wojciech Jarosz. Unbiased and consistent rendering using biased estimators. ACM Transactions on Graphics (Proceedings of SIGGRAPH), 41(4), July 2022.

- ▶ Chapter 7 is based on my previous publication:

Zackary Misso, Yining Karl Li, Brent Burley, Daniel Teece, Wojciech Jarosz. Progressive null-tracking for volumetric rendering. In: ACM SIGGRAPH Conference Papers. July 2023.

I was the primary investigator and primary author for this publication.

- ▶ Chapter 8 is based on my previous publication:

Zackary Misso, Benedikt Bitterli, Iliyan Georgiev, Wojciech Jarosz. Unbiased and consistent rendering using biased estimators. ACM Transactions on Graphics (Proceedings of SIGGRAPH), 41(4), July 2022.

I was the primary investigator and primary author for this publication.

List of Algorithms

1	Null-scattering track length transmittance estimator	80
2	Weighted track length transmittance estimator	81
3	Ratio tracking transmittance estimator	81
4	Next-flight ratio tracking transmittance estimator	83
5	Adaptive ratio transmittance estimation	95
6	Adaptive track length transmittance estimator	97
7	Adaptive next-flight transmittance estimator	98
8	Pseries cumulative transmittance estimator	109
9	Pseries CMF transmittance estimator	110
10	Unidirectional-MIS transmittance estimator	116
11	Progressive medium extinction evaluation	140
12	Accumulating $p(\vec{\omega}_i)$ from all points along a line	145
13	Progressive extinction evaluations with emissive indicator update	146

List of Figures

1.1 A brief recipe for creating consistent estimators for non-linear problems.	3
2.1 Visualization of a half-line	6
2.2 Illustration of radiant flux	7
2.3 Illustration of irradiance	7
2.4 Illustration of radiant intensity	7
2.5 Illustration of radiance	8
2.6 Exitant/incident radiance in a vaccum.	9
2.7 Visualization of shape vs. appearance	10
2.8 Visualization of explicit and implicit surfaces	10
2.9 Geometric illustration of the BSDF	11
2.10 Visualization of different BSDFs	11
2.11 Illustration of importance functions	12
2.12 Visualization of the rendering equation integrand	13
2.13 Geometric illustration of the surface area form	15
2.14 Difference in measures between surface area and directional form of RE	15
2.15 Geometrical illustration of a full path	16
3.1 Visualization of continuous and discrete probability distributions	21
3.2 Visualization of different stratified sampling patterns	31
4.1 Visual impact of different albedos	36
4.2 Voxel density grids versus procedural volumetric representations	36
4.3 Visualization of different Henyey-Greenstein phase functions	38
4.4 Illustration of light-medium interactions	38
4.5 Visualization of free-flights and transmittances for simple functions	41
4.6 Geometrical illustration of the path contribution	45
4.7 Decomposition of the combined extinction	47
4.8 Geometrical illustration of the null-scattering path contribution	50
4.9 Visualization of differently correlated scatterers	52
4.10 Visualization of the Davis-Weinstein transmittance model	55
5.1 Illustration of naive unidirectional path tracing	59
5.2 Render time comparison with and without NEE	60
5.3 Illustration of unidirectional path tracing with NEE	60
5.4 Illustration of unidirectional light tracing with NEE	61
5.5 Illustration of bidirectional path tracing	63
5.6 Illustration of photon mapping	64
5.7 Visualization of PM's ability to render reflections of caustics vs. path tracing	64
5.8 Visualization of the bias of PM in relation to radius	66
5.9 Visualization of equi-angular sampling vs. free-flight sampling	72
6.1 Graphical example of bias when naively estimating transmittance	77
6.2 The behaviors of ratio tracking and track length estimation	81
6.3 Visualization of residual decomposition of a medium	82
6.4 Analytical variance of ratio tracking for bounding and non-bounding μ_t	89
6.5 Adaptive vs. traditional free-flight importance sampling	94

6.6	Visualization of the behavior of adaptive ratio tracking estimates	95
6.7	Visualization of the behavior of adaptive track length estimates	96
6.8	Adaptive vs. traditional trackers in a stained glass medium	99
6.9	Adaptive vs. traditional trackers in canonical configurations	100
6.10	Visualization of how μ_t dictates the shape of the iterative power series	104
6.11	Visualization of how non-bounding μ_t 's change the shape of the power series	106
6.12	Visualization of the inefficiencies of PCume	109
6.13	Full canonical comparison of many transmittance estimators	118
6.14	Comparison of transmittance estimators in a homogeneous medium	120
6.15	Comparison of transmittance estimators in a heterogeneous medium	120
6.16	Comparison of the benefits of stratification with Volterra vs. power series estimators	121
6.17	Convergence plots of stratified Volterra and power series estimators	122
6.18	Comparison between transmittance estimators with non-bounding μ_t	122
7.1	Worst-case example of non-bounding μ_t in a procedural medium	125
7.2	Example of a procedurally constructed medium	127
7.3	The behavior of ratio tracking with bounding and non-bounding μ_t	129
7.4	The distinct behaviors of ratio tracking with non-bounding μ_t	130
7.5	How various transmittance estimators degenerate in the presence of non-bounding μ_t	131
7.6	The degradation of weighted delta tracking with non-bounding μ_t	132
7.7	Visualizing a clamped medium	134
7.8	A visualization of naive progressive extinctions	135
7.9	Empirical example of bias converging when initial samples are biased	136
7.10	Empirical convergence of $MSE[\langle I \rangle_n]$, $B^2[\langle I \rangle_n]$, and $V[\langle I \rangle_n]$ for an initially biased secondary estimator	138
7.11	Empirical convergence of $MSE[\langle I \rangle_n]$ with initially biased and unbiased secondary estimators	138
7.12	Example of how long it takes ratio and PCume to discover a bounding μ_t as a function of ϵ	139
7.13	Unbiased progressive NEE versus naive unidirectional volumetric path tracing	147
7.14	Progressive extinctions vs. prior work in non-bounding procedural media	148
7.15	Progressive extinctions vs. prior work in highly scattering media	149
7.16	Convergence of progressive extinctions in practice	150
8.1	Restating Fig. 1.1	153
8.2	Visualization of how the terms in the Taylor series correct for the initial bias	155
8.3	Comparison between our reciprocal estimators and Bernoulli trials	158
8.4	Path tracing reinterpreted as a telescoping series estimator	160
8.5	Error convergence of finite vs. infinite variance estimators	168
8.6	Example of unbiased finite differencing differentiating with respect to roughness	173
8.7	Convergence of progressive finite differences	174
8.8	Progressive finite differences in the presence of discontinuities	175
8.9	Convergence rate improvements due to correlations in functions of expectations	177
8.10	Cloud render using our unbiased path tracer for general participating media	179
8.11	Transmittance estimator comparison across pink-noise and exponential transmittances	180
8.12	Decomposition of unbiased photon mapping as a form of debiasing	182
8.13	Equal cost comparison between progressive photon mapping and our unbiased photon mapping	183
8.14	Error convergence for consistent and unbiased photon mapping	184

Contents

Abstract	i
Acknowledgments	ii
Publications	iii
List of algorithms	iv
List of figures	v
Contents	vii
1 Introduction	1
1.1 Organization of the Dissertation	3
2 Fundamentals of light transport	5
2.1 Notation	6
2.2 Radiometry	7
2.2.1 Radiometric quantities	7
2.2.2 Radiometric relationships	8
2.2.3 Incident and exitant radiance	8
2.3 Surface scattering	9
2.3.1 Surface representation	10
2.3.2 The BSDF	10
2.4 Surface-based light transport	12
2.5 The Cartesian formulation	14
2.6 The path integral formulation	15
3 Fundamentals of Monte Carlo Integration	18
3.1 Fundamentals of probability	18
3.1.1 Random variables	18
3.1.2 Distribution functions	19
3.1.3 Expected value and variance	21
3.2 Basics of Monte Carlo Integration	22
3.2.1 Primary vs. secondary estimation	23
3.2.2 Unbiasedness	23
3.2.3 Consistency	24
3.2.4 Convergence	25
3.2.5 Discrete form	27
3.3 Estimator optimizations	28
3.3.1 Importance Sampling	28
3.3.2 Multiple Importance Sampling	29
3.3.3 The Inversion Method	30
3.3.4 Correlations	30
3.3.5 Control Variates	32
3.3.6 Multilevel Monte Carlo	32

4 Light transport in participating media	34
4.1 Medium properties and light interactions	35
4.2 Classical volumetric transport	39
4.2.1 Transmittance	39
4.2.2 Free-flight distribution	41
4.2.3 Volumetric rendering equation	42
4.2.4 Volumetric path integral	43
4.3 Null-scattering reformulation	46
4.3.1 Null-scattering VRE	48
4.3.2 Null-scattering path integral	49
4.4 Non-classical volumetric transport	51
4.4.1 Stochastic media and ensemble averaging	52
4.4.2 Path throughput	53
4.4.3 Transport Kernel	53
4.4.4 Davis-Weinstein transmittance model	55
5 Monte Carlo based rendering algorithms	57
5.1 Unidirectional surface-only methods	58
5.1.1 Path tracing	58
5.1.2 Next-event estimation	59
5.1.3 Russian Roulette termination	60
5.1.4 Unidirectional light tracing	61
5.2 Bidirectional path tracing	61
5.3 Photon mapping	64
5.3.1 Progressive photon mapping	65
5.4 Volumetric path tracing	67
5.4.1 VRE	67
5.4.2 Free-flight distance sampling	68
5.4.3 Next Event Estimation	71
5.4.4 Aside: Emission	72
5.4.5 Null VRE	73
5.4.6 Non-Classical Transport	75
6 Integral formulations for exponential transmittance	76
6.1 Background	78
6.1.1 Prior unbiased solutions	78
6.2 Volterra integral formulation	83
6.2.1 RTE based derivation	83
6.2.2 IVP derivation	84
6.2.3 Null-collision formulation	85
6.2.4 Re-deriving prior estimators	86
6.2.5 Variance analysis	88
6.2.6 Adaptive importance sampled estimators	93
6.2.7 Relating Null-VRE to VRE using the Volterra	101
6.3 Power-series integral formulation	103
6.3.1 Base formulations	104
6.3.2 Null-collision formulations	105
6.3.3 Re-deriving ratio tracking	106
6.3.4 Novel transmittance estimators	108
6.4 Hypercube formulation	111
6.4.1 Analogies to rendering formulations	111

6.4.2	Base formulation	112
6.4.3	Null formulation	113
6.4.4	Hypercube estimation	114
6.5	Discussion, results, and implementation	116
6.5.1	Summary of contributions	116
6.5.2	Numerical experiments	117
6.5.3	Discussion	123
7	Progressive techniques for volumetric rendering	124
7.1	Case for progressive volumetric rendering	126
7.2	Root cause of degenerate behavior	128
7.2.1	Transmittance estimation	128
7.2.2	Free-flight sampling	132
7.3	Progressive extinctions	133
7.3.1	Progressive extinctions through biasing	134
7.3.2	Asymptotic analysis	135
7.3.3	Adaptive Approach	138
7.4	Progressive NEE in black box media	140
7.4.1	NEE through line integration	141
7.4.2	Progressive volumetric NEE	145
7.4.3	Unbiased progressive volumetric NEE	146
7.5	Results	147
7.6	Conclusion	151
7.6.1	Future work	151
8	Unbiased and consistent rendering using biased estimates	152
8.1	Generalized consistent formulations	153
8.1.1	Taylor series formulation	155
8.1.2	IVP formulation	158
8.1.3	Telescoping series formulation	159
8.1.4	Generalized consistent formulations	161
8.1.5	Progressive formulation	161
8.2	Verifying finite variance	162
8.2.1	General consistent estimation	162
8.2.2	Optimal probability distributions	163
8.2.3	Variance-optimal probability distributions	164
8.2.4	Verifying finite work-normalized variance	166
8.3	Handling finite work-normalized variance	166
8.4	Handling infinite work-normalized variance	168
8.4.1	Unbiased estimators under infinite variance	169
8.4.2	Aside: Alleviating infinite variance through MIS	169
8.4.3	Progressive estimators under infinite variance	170
8.5	Applications	171
8.5.1	Unbiased and progressive finite differencing	171
8.5.2	Unbiased transmittance estimation	175
8.5.3	Unbiased photon mapping	181
8.6	Conclusion	184
8.6.1	Discussion	184
9	Summary and future work	186
9.1	Summary of contributions	186

9.2 Discussion and limitations of our recipe	188
9.3 Future work	189
APPENDIX	191
A Generalized Taylor series based raymarching	192
A.1 Pre-processed Russian Roulette	193
A.2 Unbiased estimates as expansion points	194
A.3 Raymarched integral estimates	195
A.4 Symmetric Means	196
B Non-exponential Volterra and VRE	197
B.1 Generalized null-Volterra	197
B.2 Non-exponential VRE	200
B.3 Null non-exponential VRE discussion	202
Bibliography	203

1

Introduction

The field of *computer graphics* focuses on the generation, manipulation, and simulation of virtual environments as well as the synthesis of these environments into images. As such, computer graphics encompasses a melting pot of topics ranging from real time applications to realistic, physically based simulations, and more recently has even become the playground for machine learning applications. The ideas developed within computer graphics tend to have wide reaching applications from artistic endeavors throughout the entertainment industry to scientific applications where virtual environments are designed to mimic real life physical processes.

Within computer graphics is the sub-field of rendering which focuses on simulating the interactions between light and different solid or volumetric materials. *Physically based rendering* is a specialization of rendering where the flow of light is represented by a mathematical model which roughly matches the behavior of light in the real world. The main goal of physically based rendering is to synthesize, or render, images that appear physically plausible. Due to its emphasis on realism, developments within physically based rendering have broad reaching applications specifically relating to the field of scientific visualization, has ties to the atmospheric sciences and the field of neutron transport, and are instrumental to the production of visual effects for movies as well as video games.

The field of physically based rendering is inspired and driven by both artistic and scientific ambitions. As a consequence, it is an applied science which has had a long history of being influenced and driven by developments from other fields. In fact, when the field's history is observed in its entirety, a common pattern arises: Another field develops the theory which explains the behavior of some visual phenomena that practitioners of physically based rendering want to replicate. Then, given this theory, scientists of physically based rendering derive and develop the formulations and techniques necessary to replicate the desired phenomena in a manner which is efficient enough to render.

Some common examples of this pattern in action consist of

- ▶ The theory of radiative transfer [1] which nearly all physically based rendering algorithms are derived from.
- ▶ The realization of non-exponential attenuation within clouds from the atmospheric sciences [2, 3] which eventually inspired the development of the theory necessary to render them [4].
- ▶ The knowledge that black holes can bend light, and the desire to realistically render them for a production film leading to the development of rendering techniques supporting the bending of light due to gravitational forces [5].

Physically based rendering is based on our current understanding of the real physical processes behind how we perceive the world. It simulates how

1.1 Organization of the Dissertation	3
--	---

[1]: Chandrasekhar (1960), *Radiative Transfer*

[2]: Davis et al. (1999), 'Horizontal Structure of Marine Boundary Layer Clouds from Centimeter to Kilometer Scales'

[3]: Kostinski et al. (2001), 'Scale-Dependent Droplet Clustering in Turbulent Clouds'

[4]: Bitterli et al. (2018), 'A Radiative Transfer Framework for Non-Exponential Media'

[5]: James et al. (2015), 'Gravitational Lensing by Spinning Black Holes in Astrophysics, and in the Movie Interstellar'

light travels from artificially represented emitters, interacts with the objects in the world, and then finally finds its way to artificial sensors which can be thought of as virtual cameras in space. In real life, we perceive the world through the accumulation of an unimaginable number of photons hitting our retinas every millisecond which our brains process into a continuous visual stream of the world around us. Physically based techniques operate similarly, however, they result in rendered images and the time it takes to compute those images is constrained by the computational capacity of a computer.

The accumulation of photons by our retinas mathematically translates into integrating over all paths which light can traverse through space. This is why physically based rendering algorithms are occasionally referred to as *integrators* since they effectively evaluate this high dimensional integration problem.

Analytically solving this high dimensional integration problem would require analytically evaluating *every* possible path light can traverse in a given scene. This is a computationally intractable problem for any but the most simplistic of scene configurations. Instead, modern algorithms within the field rely on *stochastic* methods to *estimate* the high dimensional integrals in a reasonable amount of time. The most popular rendering algorithms used today are based on *Monte Carlo integration*, which stochastically estimates integrands through randomized sampling. Monte Carlo integration is a provably good technique because the integral estimates can typically be guaranteed to be both *consistent* and *unbiased*. Meaning the integral estimates are expected to give us the correct solution on average, and if we were to run our stochastic estimators for an infinite amount of time, the error would disappear giving us the true result.

While Monte Carlo integration is an immensely useful tool, it has one glaring flaw. The ideal properties of Monte Carlo integration are only guaranteed when it is applied to integrals which are unmodified or linearly modified. In situations where the integrals we want to estimate undergo *non-linear transformations*, naive Monte Carlo integration can no longer be applied without resulting in *incorrect* solutions. Unfortunately, these non-linearly transformed integrals occur throughout light transport, from rendering volumetric participating media, to estimating the normalization constants in the form of reciprocal estimation, to even evaluating some objective functions used in machine learning applications on the results of differentiable renderers.

The field of light transport is developing at a rate where we have seen an explosion of theoretical developments, from non-exponential volumetric rendering to differentiable rendering. However, our ability to derive useful stochastic solutions for the increasingly complex problems we now face has significantly lagged behind these developments. In this dissertation we propose a step-by-step recipe (see Fig. 1.1) dedicated to addressing these difficult non-linear problems through designing *consistent* solutions for both scientists and practitioners alike. While there is no one-size-fits-all solution to these difficult non-linear quandaries, this recipe is designed

- ① **Construct consistent formulation.** Reformulate the non-linear problem as a consistent formulation which is amenable to Monte Carlo integration.
- ② **Prove finite work-normalized variance.** Derive as close to a *theoretically* optimal primary estimator as possible for ①, and then determine if it has finite-work-normalized variance.

IF FINITE:

- ③ **Build unbiased finite-variance estimator.** Construct an unbiased primary estimator and a naive secondary estimator then test. If efficiency is unsatisfactory, continue to ④.
- ④ **Build consistent progressive estimator.** Construct a progressive secondary estimator from the consistent formulation which trades bias in return for lower cost, and/or lower variance. If possible, go to ⑤.
- ⑤ **Optional: Build unbiased consistent progressive estimator.** Derive a way to account for the bias of the progressive estimator in an unbiased manner.

IF INFINITE:

- ⑥ **Build unbiased infinite-variance estimator.** Construct a primary estimator by balancing the cost versus variance trade-off such that cost remains finite and variance approaches infinity as slowly as possible. Construct a naive secondary estimator. If performance is acceptable, done, otherwise continue to ⑦.
- ⑦ **Build consistent progressive estimator.** Convert the secondary estimator into a consistent progressive estimator, and reparameterize the consistent formulation such that the variance grows sublinearly. The resulting estimator, will maintain good efficiency in return for bias.

Figure 1.1: A brief recipe for creating consistent estimators for non-linear problems.

to address and account for the challenges which will arise in deriving practical solutions.

Our recipe is constructed from an accumulation of experience addressing and analyzing instances of non-linear estimation problems in the context of physically based rendering. While estimating non-linearly transformed integrals is a widely applicable concept, this dissertation chooses to focus on this classification of problems only in the context of rendering related quandaries¹. However, the recipe, analysis, and techniques we review and introduce are not limited solely to the field. We hope that the work proposed in this dissertation will aid future practitioners and researchers alike in deriving practical estimators for the quandaries we seek to answer and the problems we have yet to discover.

1: Fig. 1.1 contains terms and ideas which we assume the reader is not familiar with. We will define all relevant terminology in Chapter 3 and introduce a finalized recipe in Chapter 8 once all the necessary ideas have been introduced

1.1 Organization of the Dissertation

This dissertation is organized into nine chapters. Chapter 6, Chapter 7, and Chapter 8 contain our novel contributions while the remaining chapters consist of relevant background information or future work. In Chapter 2 we review the fundamentals of light transport followed by a review of Monte

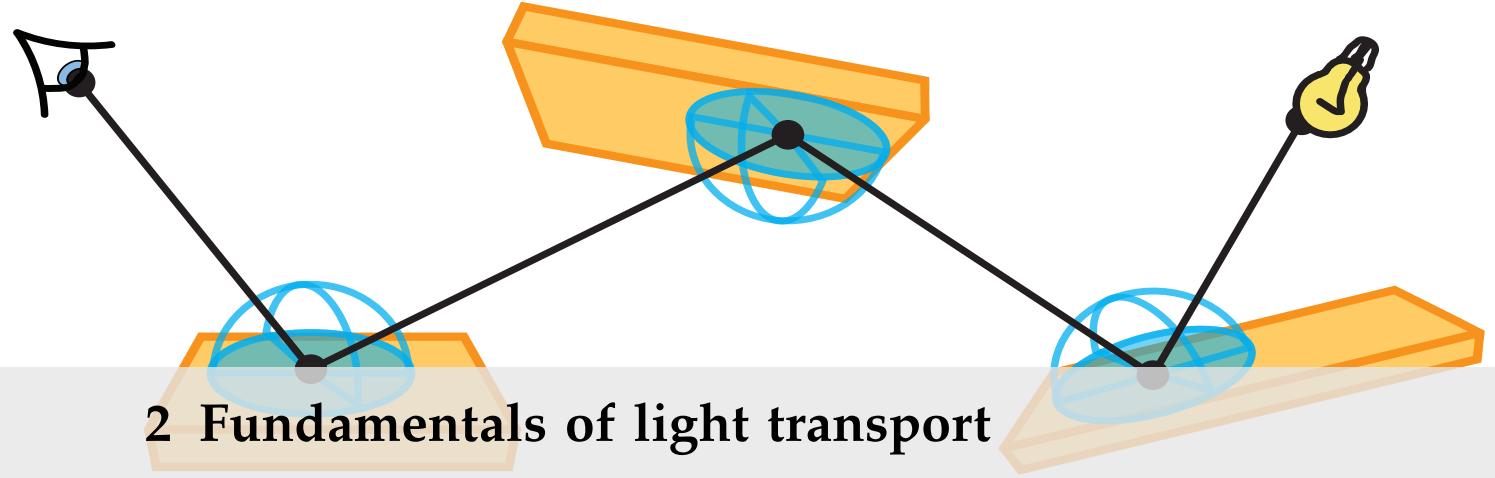
Carlo integration in Chapter 3. We then expand upon the fundamentals introduced in Chapter 2 to support rendering volumetric media such as clouds, smoke, stained glass, etc. In Chapter 5 we review how Monte Carlo integration is typically applied to estimate the high dimensional integrals which arise in light transport. In the process, we review some widely known rendering algorithms as well as how they are derived through the direct application of Monte Carlo integration.

Chapter 6 is the first chapter containing our novel contributions where we focus on the specific non-linear problem of classical transmittance estimation for rendering heterogeneous participating media. We derive two consistent formulations which only involve computing linear terms that can be constructed from the classical exponential transmittance function. We also review ways of estimating our formulations, show how all prior estimators can be rederived from our work, propose novel improvements to prior work, and introduce novel estimators. In terms of our recipe, this chapter reviews points (1,2,3).

In Chapter 7 we review the same problem of classical transmittance estimation, except this time in situations where we have purely unknown, black box extinction functions. Nearly all unbiased estimation strategies rely on having a priori knowledge of the maximum extinction for efficient applications of Monte Carlo. When the extinction is a black box function, practitioners must "guess" a bounding extinction which can result in uncontrollable, but still finite, variance. For this case, we propose a progressive solution which avoids excessive variance in return for small amounts of initial bias that vanishes quickly. We also extend the idea of progressive volumetric rendering to support volumetric emission sampling in black box media. This chapter reviews points (3,4,5) in our recipe.

In Chapter 8 we introduce another consistent formulation which is amenable to unbiased estimation focusing on the idea of *debiasing* biased formulations. We additionally generalize the definition of a "consistent formulation", review how to determine finite versus infinite variance estimators in practice, and analyze the problem of infinite variance estimation. In doing so, we will introduce the first unbiased photon mapping algorithm, as well as the first formulation for transmittance estimation which generalizes to *most* non-classical transmittance function. While this chapter will mostly focus on points (1,2,3,6,7), it will also review the recipe in its entirety.

We summarize this dissertation, indicate limitations, and discuss avenues for future work in Chapter 9.



2 Fundamentals of light transport

The fundamental goal of physically-based rendering is to render images of virtual environments in a manner which is based on how light behaves in reality. This is accomplished by simulating the physical processes light undergoes in real life through defining how light interacts with all materials which encompass a virtual environment. Physically-based rendering then becomes the problem of simulating the flow of light from emitters to sensors, which is equivalent to simulating the direct radiative transfer of energy throughout a virtual scene, aka, computing the transport of light.

In the real world, the interactions between light and the environment are complicated by the fact that light exhibits properties of both particles and waves. The wave properties of light can lead to dispersion, polarization, and interference effects, however, the majority of visual phenomena we perceive on a daily basis can be described more simply without considering these complicated properties of light. This is why we employ a simplified model known as *geometric optics* which makes the assumptions that light is unpolarized, moves at infinite speed, and always travels in a straight line between scattering events. While other models supporting these more complicated properties of light exist [5, 6] or support simulating transient properties of light [7, 8], we choose to employ this simpler model since it drastically improves the computational efficiency of our derived solutions. In practice, if other realistic effects are desired, it is usually more computationally tractable to implement them as specialized material models with customized interactions between incoming and outgoing light (see Sec. 2.3.2). Additionally, while *photons* in the real world exhibit the properties of both waves and particles, whenever we refer to photons in this dissertation we are directly referring to discrete particles.

In the remainder of this chapter, we will review the mathematical model for surface-only light transport governing all of the algorithms that will be discussed in this dissertation. We import this model from prior work [9–11] and include only a brief description of the model itself. For a more thorough discussion, we refer you to the original authors' work.

Chapter organization. We will first provide an overview on specific notation which we will use throughout this dissertation in Sec. 2.1 followed by a brief description of the radiometric quantities we are interested in computing in Sec. 2.2. In Sec. 2.3 we describe the underlying function

2.1 Notation	6
2.2 Radiometry	7
2.2.1 Radiometric quantities . .	7
2.2.2 Radiometric relationships	8
2.2.3 Incident and exitant radiance	8
2.3 Surface scattering	9
2.3.1 Surface representation . .	10
2.3.2 The BSDF	10
2.4 Surface-based light transport	12
2.5 The Cartesian formulation	14
2.6 The path integral formulation	15

[5]: James et al. (2015), ‘Gravitational Lensing by Spinning Black Holes in Astrophysics, and in the Movie Interstellar’

[6]: Mojžík et al. (2016), ‘Bi-Directional Polarised Light Transport’

[7]: Marco (2013), ‘Transient Light Transport in Participating Media’

[8]: Yi et al. (2021), ‘Differentiable Transient Rendering’

[9]: Immel et al. (1986), ‘A Radiosity Method for Non-Diffuse Environments’

[10]: Kajiya et al. (1989), ‘Rendering Fur with Three Dimensional Textures’

[11]: Veach (1997), ‘Robust Monte Carlo Methods for Light Transport Simulation’

governing the scattering of light when interacting with a surface. Using the definition of the radiometric quantities and the surface scattering function, we then derive the rendering equation in Sec. 2.4 which formulates light transport as a recursive integral formulation using directional measures. In Sec. 2.5 we reformulate the rendering equation in Cartesian space and then use that reformulation to derive the path integral formulation for light transport in Sec. 2.6.

2.1 Notation

We refer to scalars using italic script, and vectors using bold letters. Positions in space will use the Latin alphabet (x, y, z), while we will typically reserve the Greek letters $\vec{\omega}$ and $\vec{\psi}$ for representing directions on the unit sphere. Vectors representing directions will be distinguished from vectors describing positions using arrows above the letters. Other than the two previously mentioned Greek letters, we will also commonly use \vec{n} to refer to surface normals, specifically. Additionally, we will also refer to the unit direction starting from one point x pointing towards another point y as $\vec{\omega} = \vec{xy}$.

We define a *ray* as the half-line described by the position and direction vector pair $(x, \vec{\omega})$ where x is the starting position and $\vec{\omega}$ is the unit direction the half-line extends along (Fig. 2.1). We then refer to any point along the half-line as $x_t = x + t\vec{\omega}$ where t is the distance along the half-line from its origin.

We will use blackboard bold fonts to denote spaces and sets such as the Euclidean space, \mathbb{R}^3 . We will introduce sets and spaces as needed, however, we will define common ones which are used throughout the dissertation here. The unit sphere containing the space of all unit directions is defined as,

$$\mathbb{S}^2 := \{\vec{\omega} \mid \vec{\omega} \in \mathbb{R}^3 \text{ and } \|\vec{\omega}\| = 1\}, \quad (2.1)$$

and the hemisphere, or half-sphere, containing the space of all unit directions about some normal \vec{n} is defined as,

$$\mathbb{H}_{\vec{n}} := \{\vec{\omega} \mid \vec{\omega} \in \mathbb{R}^3 \text{ and } \|\vec{\omega}\| = 1 \text{ and } \vec{\omega} \cdot \vec{n} \geq 0\}. \quad (2.2)$$

In nearly every context within this dissertation we assume that the existence of the normal \vec{n} is implied. Thus, from now on we will write the hemisphere of all directions about some normal as the shorthand $\mathbb{H}_{\vec{n}} = \mathbb{H}$ where we drop the dependence on the normal vector.

Measures. Many problems discussed in this dissertation involve integrating over spaces differing in dimensionality and structure. While we use measures to specifically denote this, we avoid an in depth discussion on measure theory itself. We instead refer the interested reader to Wenzel Jakob's PhD thesis [12] for a more detailed discussion on measures in graphics. Otherwise, we use dx to represent the Lebesgue measure on

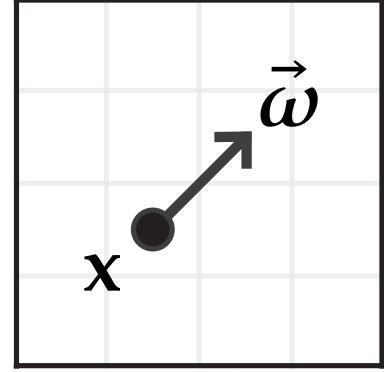


Figure 2.1: A visualization of a half-line characterized by its position-direction pair, $(x, \vec{\omega})$.

[12]: Jakob (2013), 'Light Transport on Path-Space Manifolds.'

Euclidean space, $d\Omega(\vec{\omega})$ to represent the solid angle measure on the unit sphere, $dA(x)$ to represent the area measure on the space of all surface points \mathbb{A} , and $dV(x)$ to represent the volume measure on the space of all volume points \mathbb{V} . When the context is self-explanatory, we may drop the positional and directional arguments and refer to the higher dimensional measures as $d\Omega$, dA , and dV , respectively.

2.2 Radiometry

2.2.1 Radiometric quantities

Here we define the radiometric quantities that algorithms in light transport seek to measure [13]. We additionally provide illustrations depicting the various quantities in Figs. 2.2 to 2.5.

[13]: Nicodemus (1963), 'Radiance'

Radiant Flux. The *radiant flux* represents the amount of energy Q measured in Joules [J] flowing through some finite surface area A per unit time t . Flux is denoted as,

$$\Phi(A) := \frac{dQ(A)}{dt}, \quad (2.3)$$

and its units are Joules per second which are equivalent to Watts [$W = \frac{J}{s}$].

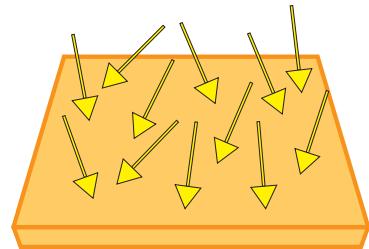


Figure 2.2: An illustration of radiant flux.

Irradiance. The *irradiance* represents the amount of incident power hitting a surface per unit surface area and has units [$W = \frac{J}{m^2}$]. Irradiance can be expressed in terms of flux as,

$$E(x) := \frac{d\Phi(A(x))}{dA(x)}, \quad (2.4)$$

and is always measured at some surface location, x , which we assume has a surface normal, \vec{n} . While irradiance measures incident power, *radiant exitance* or *radiosity* is used to represent the power leaving a surface.

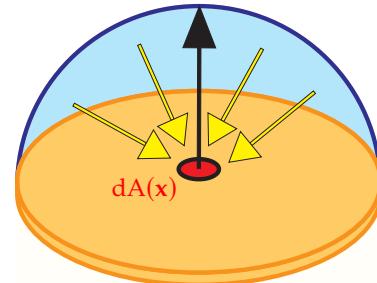


Figure 2.3: An illustration of irradiance.

Radiant intensity. In addition to measuring flux along surface area, we can also measure the directional density of flux per solid angle about some direction $\vec{\omega}$ as,

$$I(\vec{\omega}) := \frac{d\Phi(A)}{d\Omega(\vec{\omega})}, \quad (2.5)$$

which is known as the *radiant intensity*. This quantity has units [$\frac{W}{sr}$] where "sr" denotes steradians.

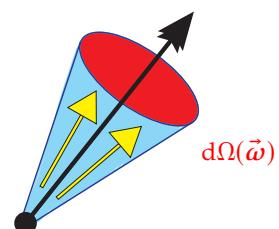


Figure 2.4: An illustration of radiant intensity.

Radiance. Radiance is the quantity we are most interested in computing within the study of light transport since it is the same quantity which we perceive with our eyes or gets captured through the lens of a camera. By considering both direction and area, radiance represents the flux per unit solid angle and perpendicular area,

$$L(\mathbf{x}, \vec{\omega}) := \frac{d^2\Phi(A(\mathbf{x}))}{d\Omega(\vec{\omega})dA^\perp(\mathbf{x})}. \quad (2.6)$$

We can interpret perpendicular area as a projected measure of area, i.e. $dA^\perp(\mathbf{x}) = dA(\mathbf{x})(\vec{n} \cdot \vec{\omega})$ where $\vec{n} \cdot \vec{\omega} = |\cos \theta|$ since both \vec{n} and $\vec{\omega}$ are assumed to be unit vectors pointing away from the surface. This cosine term is referred to as the *forshortening term* and is related to the *Lambert cosine law* [14] explaining how radiant intensity disperses based on the angle θ between the surface normal and viewing direction for an ideal diffuse surface.

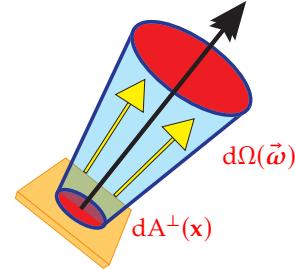


Figure 2.5: An illustration of radiance.

[14]: Lambert (1760), *Photometria: Sive de Mensura et Gradibus Luminis, Colorum et Umbrae*

2.2.2 Radiometric relationships

Since the various radiometric quantities are heavily interrelated with one another and the fact that radiance is instrumental to the field of light transport, it becomes useful to redefine the various radiometric quantities with respect to radiance itself.

We can define flux in terms of radiance by inverting Eq. (2.6) and integrating over both surface area and the hemisphere of directions,

$$\Phi := \int_{\mathbb{A}} \int_{\mathbb{H}} L(\mathbf{x}, \vec{\omega}) |\cos \theta| d\Omega(\vec{\omega}) dA(\mathbf{x}). \quad (2.7)$$

Irradiance can be defined in terms of radiance similarly resulting in,

$$E(\mathbf{x}) := \int_{\mathbb{H}} L(\mathbf{x}, \vec{\omega}) |\cos \theta| d\Omega(\vec{\omega}), \quad (2.8)$$

and we can also express radiant intensity in terms of radiance to arrive at,

$$I(\vec{\omega}) := \int_{\mathbb{A}} L(\mathbf{x}, \vec{\omega}) |\cos \theta| dA(\mathbf{x}). \quad (2.9)$$

2.2.3 Incident and exitant radiance

Similar to how irradiance represents power hitting a surface while radiosity represents power leaving a surface, radiance can also be represented as both hitting and leaving a surface. We refer to these different quantities as *incident radiance* and *exitant radiance*, respectively. In this dissertation we will employ the convention that $L_i(\mathbf{x}, \vec{\omega}_i) = L_i(\mathbf{x}, \vec{y}\mathbf{x})$ denotes incident radiance at \mathbf{x} from direction $\vec{\omega}_i$ while $L_o(\mathbf{x}, \vec{\omega}_o) = L_o(\mathbf{x}, \vec{x}\mathbf{y})$ denotes the exitant radiance from \mathbf{x} in the direction of $\vec{\omega}_o$.

Despite representing different directions of flow, incident and exitant radiance fundamentally represent two different measurements since the set of photon events which happen right before and right after colliding with a surface are distinct sets. Effectively, the light entering and leaving a surface does not have to be equal, thus,

$$L_i(\mathbf{x}, \vec{\omega}) \neq L_o(\mathbf{x}, \vec{\omega}). \quad (2.10)$$

On the other hand, light will travel unobstructed through a vacuum, thus, there will be an equal flow of incident to exitant radiance,

$$L_i(\mathbf{x}, \vec{\omega}) = L_o(\mathbf{x}, -\vec{\omega}). \quad (2.11)$$

from the perspective of any point within the vacuum. In this chapter, we will only consider the case of analyzing the transport of light through a vacuum, but in Chapter 4 we will expand our theory to consider cases where photons interact with participating media where Eq. (2.12) no longer holds.

Assuming that light travels in straight lines we define the *ray tracing* operator, $r(\mathbf{x}, \vec{\omega}) = \mathbf{y}$, which returns the next surface intersection point, \mathbf{y} , given the current surface intersection \mathbf{x} and exitant direction, $\vec{\omega}_o$. Given Eq. (2.11), we can infer that radiance will remain constant along straight lines in a vacuum, thus,

$$L_o(\mathbf{x}, \vec{\omega}_o) = L_i(\mathbf{y}, \vec{\omega}_i), \text{ where } \mathbf{y} = r(\mathbf{x}, \vec{\omega}_o), \quad (2.12)$$

which we illustrate in Fig. 2.6. As a slight simplification of notation, we may generally refer to radiance as $L(\mathbf{x}, \vec{\omega})$ without specifying incidence or exitance in later chapters.

2.3 Surface scattering

The relationship between incident and exitant radiance (2.10) will vary depending on the specific properties of the surface which light interacts with. In this section, we will review a general model representing this relationship, however, to do so we first need to define what "surfaces" are in the context of light transport and how we choose to represent them practically.

In the real world, a surface is the outermost part of any tangible object which we can touch. More scientifically put, it is the interface between two different phases, i.e. between two different solids, a solid and a liquid, or a solid and a gas. On a macroscopic level most surfaces might be perceived as flat by our eyes, however, on a microscopic level solids tend to have very complicated microstructures which give them their unique appearance.

Modeling and designing 3D objects which are equivalent to real world objects is typically impossible due to memory limitations. Explicitly modeling the microstructures at a necessary level of detail would result in 3D models which would likely not fit within the capacity of a typical

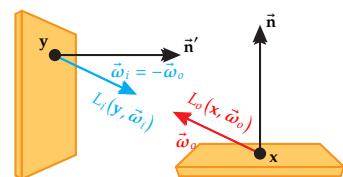


Figure 2.6: Within a vacuum, the exitance radiance leaving point x and arriving as incident radiance at point y will remain constant.

harddrive on a personal computer. This is why we instead decompose the description of surfaces into a representation of their overall *shape* (Sec. 2.3.1) and encode their unique *appearance* (Sec. 2.3.2) separately. We visualize this decomposition in Fig. 2.7.

2.3.1 Surface representation

The overall shape of a surface can be represented either explicitly or implicitly within a virtual environment. Most commonly, surfaces are defined explicitly by interconnected triangles or quadrilaterals. Surfaces can also be defined implicitly through the use of signed distance functions (SDFs). We visualize the difference between these explicit and implicit representations in Fig. 2.8.

All algorithms and techniques discussed in this dissertation should work regardless of how the surfaces are represented as long as we assume to have access to a *ray-tracing operator* $r(x, \vec{\omega})$. Given some current location x , the ray-tracing operator returns the next surface intersection in direction $\vec{\omega}$ and computes the surface normal \vec{n} at the next surface intersection.

Implementations and optimizations for $r(x, \vec{\omega})$ are orthogonal to this dissertation. We refer the interested reader to Embree [15] for supporting modern explicit surface representations and the sphere tracing family of techniques [16] for supporting implicit surface representations using SDFs.

2.3.2 The BSDF

A block of wood and a brick may have similar shapes, however, their appearances are vastly different. What we perceive when we observe an object is directly dependent on how light interacts with that object's surface. When light encounters a surface, it will either be *absorbed* by the surface, *scatter* somewhere else, or *transmit* through the object. The appearances of all real world objects differ by the occurrences of these three interactions.

For example, light interacting with a glass cup mostly results in specular transmission allowing us to see a slightly distorted version of the cup's contents when looking directly through it. A matte wall might scatter light equally in all directions, while a perfect mirror will scatter light only in its reflected direction. Onyx will absorb most light while still allowing specular reflections giving it a beautiful sheen around its mostly black appearance.

The *bidirectional scattering distribution function* (BSDF) mathematically describes the interactions between light and surfaces by quantifying exactly the distribution of light that scatters from some incident direction $\vec{\omega}_i$ into some outgoing direction $\vec{\omega}_o$ given some point along the surface x . The



Figure 2.7: A visualization of the shape (left) of objects comprising a virtual scene, and their actual appearance (right).

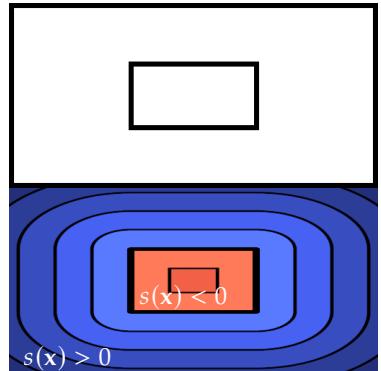


Figure 2.8: A two dimensional visualization of an explicit surface representation (top) in the form of a rectangle comprised of lines, compared to an implicit representation in the form of a signed-distance function $s(x)$ which encodes the same rectangle along the isosurface at $s(x) = 0$ (bottom).

[15]: Wald et al. (2014), ‘Embree: A Kernel Framework for Efficient CPU Ray Tracing’

[16]: Hart (1996), ‘Sphere Tracing’

BSDF is defined as the ratio of the differential outgoing radiance to the differential irradiance at the surface point,

$$\rho_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) := \frac{dL_o(\mathbf{x}, \vec{\omega}_o)}{dE_i(\mathbf{x}, \vec{\omega}_i)} = \frac{dL_o(\mathbf{x}, \vec{\omega}_o)}{L_i(\mathbf{x}, \vec{\omega}_i) |\vec{\omega}_i \cdot \vec{\omega}_o| d\Omega(\vec{\omega}_i)}, \quad (2.13)$$

which we geometrically illustrate in Fig. 2.9.

The specification of a BSDF for a virtual object is essentially equivalent to defining the appearance of that object. In creative pursuits such as video games or film, BSDF's can be artistically authored to achieve certain physically implausible visual styles. However, for a BSDF to be physically-based it has to maintain three properties:

- **Non-negativity.** Scattering cannot lead to the introduction or reflection of "negative light",

$$\rho_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) \geq 0; \forall (\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o). \quad (2.14)$$

- **Energy conservation.** A surface cannot scatter more light than it receives,

$$\int_{\mathbb{S}^2} \rho_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) |\vec{\omega}_i \cdot \vec{\omega}_o| d\Omega(\vec{\omega}_i) \leq 1. \quad (2.15)$$

- **Reciprocity.** The transport of light between the source and the emitter will remain equivalent if the direction of light flow is reversed. The BSDF should remain invariant to flipping the directional arguments,

$$\rho_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) = \rho_s(\mathbf{x}, \vec{\omega}_o, \vec{\omega}_i), \quad (2.16)$$

which is well known as Helmholtz reciprocity [17].

We visualize how different choices for the BSDF can lead to different appearances in Fig. 2.10. The choice of a scattering function is mostly orthogonal to the techniques proposed in this thesis, so we will refrain from enumerating more explicit examples. We instead refer the interested reader to Pharr's book [18] for an introductory overview of existing models.

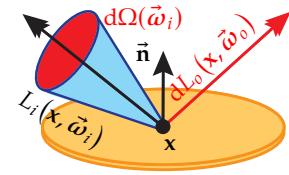
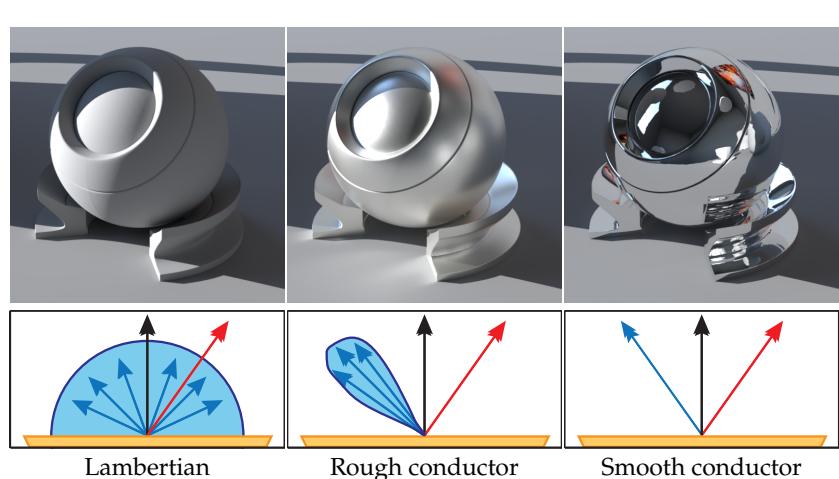


Figure 2.9: An illustration of the geometric configuration of the BSDF. Note that $\vec{\omega}_i$ and $\vec{\omega}_o$ are assumed to point *away* from \mathbf{x} .



[17]: von Helmholtz et al. (1924), *Helmholtz's Treatise on Physiological Optics*

[18]: Pharr et al. (2016), *Physically Based Rendering*

Figure 2.10: An illustration of three different BSDFs (bottom row) and their corresponding rendered appearances (top row).

2.4 Surface-based light transport

Given the description of light (Sec. 2.2) and surfaces (Sec. 2.3) we can now mathematically define the surface-based transport of light which we will need to solve in order to render physically-based images. This can be broken down into three key equations. The *transport* equation, the *measurement* equation, and the *scattering* equation.

Transport. In this chapter we assume that volumetric media is not taken into account and all non-surface regions in a scene exist within a vacuum. Based on these assumptions, we have already shown in Fig. 2.6 that the amount of light scattering from a surface in a direction towards the next scattering event will remain unchanged, meaning the directional derivative of the radiance is always,

$$(\vec{\omega} \cdot \nabla) L(\mathbf{x}, \vec{\omega}) = 0, \quad (2.17)$$

for all points in the vacuum. This means that given the ray-tracing operator, $r(\mathbf{x}, \vec{\omega})$, the transport equation can be fully described by,

$$L_i(\mathbf{x}, \vec{\omega}) = L_o(r(\mathbf{x}, \vec{\omega}), -\vec{\omega}). \quad (2.18)$$

Measurement. The end goal of most rendering algorithms is to compute some final pixel intensity values I_1, \dots, I_n for all n pixels in an image. Historically, we have been mostly focused on computing the radiant flux passing through an arbitrary sensor in space by using Nicodemus' [19] *measurement equation*,

$$I_k = \int_{\mathbb{A}} \int_{\mathbb{H}} W_e^{(k)}(\mathbf{x}, \vec{\omega}) L_i(\mathbf{x}, \vec{\omega}) d\Omega^{\perp}(\vec{\omega}) dA(\mathbf{x}). \quad (2.19)$$

This computes the final pixel value I_k in an image as the product integral of the incident radiance L_i arriving at a position \mathbf{x} on the sensor from a specific direction $\vec{\omega}$ multiplied by the pixel-specific importance function $W_e^{(k)}$. The importance function, $W_e^{(k)}(\mathbf{x}, \vec{\omega})$, represents the sensitivity distribution for the pixels on the final image of the virtual sensor. While we integrate over the entire hemisphere of directions perpendicular to the sensor plane and the entirety of surface space in Eq. (2.19), the importance is typically only non-zero for a small neighboring region around any given pixel. For all images in this dissertation, we define the importance as either a box kernel or a domain-clamped Gaussian filter (visualized in Fig. 2.11).

[19]: Nicodemus (1976), *Self-study manual on optical radiation measurements :Part I-Concepts, Chapters 1 to 3*

Scattering. We can derive the scattering equation by rearranging terms in the definition for the BSDF (2.13),

$$\begin{aligned} \rho_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) &= \frac{dL_o(\mathbf{x}, \vec{\omega}_o)}{L_i(\mathbf{x}, \vec{\omega}_i) |\vec{\omega}_i \cdot \vec{\omega}_o| d\Omega(\vec{\omega}_i)}, \\ dL_o(\mathbf{x}, \vec{\omega}_o) &= \rho_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) L_i(\mathbf{x}, \vec{\omega}_i) |\vec{\omega}_i \cdot \vec{\omega}_o| d\Omega(\vec{\omega}_i), \end{aligned} \quad (2.20)$$

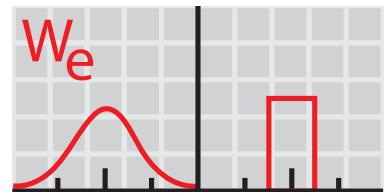


Figure 2.11: An illustration of the two different importance functions used in this dissertation. Gaussian kernel (left) and box kernel (right).

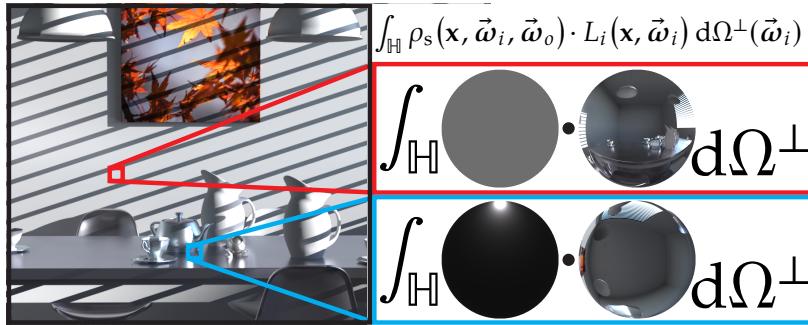


Figure 2.12: A visualization of the product integral within the rendering equation for two different pixels within a scene containing both lambertian diffuse and rough specular materials.

and then integrating both sides over solid angle to arrive at,

$$L_o(x, \vec{w}_o) = \int_H \rho_s(x, \vec{w}_i, \vec{w}_o) L_i(x, \vec{w}_i) |\vec{w}_i \cdot \vec{w}_o| d\Omega(\vec{w}_i). \quad (2.21)$$

Rendering equation. Taking into account the *emitted radiance* $L_e(x, \vec{w}_o)$ on the surface in the outgoing direction \vec{w}_o and replacing the dot product between the directions $\vec{w}_i \cdot \vec{w}_o$ with the cosine of the angle θ between them,

$$L_o(x, \vec{w}_o) = L_e(x, \vec{w}_o) + \int_H \rho_s(x, \vec{w}_i, \vec{w}_o) L_i(x, \vec{w}_i) |\cos \theta| d\Omega(\vec{w}_i). \quad (2.22)$$

we arrive at the well known *rendering equation* [9, 20]. The rendering equation effectively states that the outgoing radiance is equal to the sum of the emitted radiance from the surface and the integral over all incident radiance that gets reflected into the outgoing direction. We visualize the effect of this product integral in Fig. 2.12.

Given our transport equation (2.18) and the ray tracing operator, we can substitute the incident radiance in Eq. (2.22) with the outgoing radiance from some distant surface,

$$L_o(x, \vec{w}_o) = L_e(x, \vec{w}_o) + \int_H \rho_s(x, \vec{w}_i, \vec{w}_o) L_o(r(x, \vec{w}_i), -\vec{w}_i) |\cos \theta| d\Omega(\vec{w}_i), \quad (2.23)$$

to arrive at a recursive form of the rendering equation. After we introduce participating media in Chapter 4, we will refer to Eq. (2.23) as the *surface radiance equation* $L_o = L_s$.

Integral-equation form of the rendering equation. Equation (2.23) takes the form of a Fredholm integral equation of the second kind [21] which is generally represented as,

$$\phi(t) = f(t) + \lambda \int_a^b K(t, s) \phi(s) ds. \quad (2.24)$$

[9]: Immel et al. (1986), 'A Radiosity Method for Non-Diffuse Environments'
[20]: Kajiya (1986), 'The Rendering Equation'

[21]: Grothendieck (1956), 'La théorie de Fredholm'

Some specific properties of this integral formulation are that the integral bounds remains consistant across recursive calls, the kernel function, $K(t, s)$, remains independent of the integral bounds, and the integral formulation is recursive. Since we assume that energy is conserved (2.15), the product of ρ_s and the cosine term become analogous to a kernel function. The emitted radiance takes the form of f , L_o is the recursive call, and $\lambda = 1$.

The integral form of the rendering equation will become useful later in Chapter 6 when we make analogies between the rendering equation Eq. (2.23) and some of our derived formulations.

2.5 The Cartesian formulation

While the rendering equation (2.22) can describe all transport of light, the parameterization over solid angle is not always the most convenient. Consider the problem of visualizing a virtual environment where light originates non-uniformly from an emissive surface. In such a scenario, it might be more efficient to evaluate reflected directions pointing towards the brighter portions of the emissive surface more often than other directions. Doing so while integrating over solid angle is cumbersome due to the difficulties of projecting an arbitrary, non-uniform, emission distribution onto the hemisphere of directions. Instead, integrating emission over *surface area* would be much more convenient for these scenarios. In this section we will re-parameterize the rendering equation (2.22) into a *Cartesian* formulation that integrates over surface area to accomplish this.

Visibility term. A given point in space is not garaunteed to be *visible*, from all other points. Thus, we have to explicitly account for the visibility while integrating over all surface points. We do so by introducing a *visibility term*,

$$V(x, y) := \begin{cases} 1 & \text{if } r(x, \vec{xy}) = y, \\ 0 & \text{otherwise.} \end{cases} \quad (2.25)$$

that takes the form of an indicator function returning 1 if the path between x and y is unobstructed (i.e. a ray-trace from x to y returns y) or 0 otherwise. In the solid-angle formulation (2.23), the visibility term was implicitly encoded within the recursive call.

Change of variables. To arrive at a surface area formulation, we perform a change of variables from solid angle measure to a measure over Cartesian coordinates. To account for the change in coordinate systems, we have to apply the following Jacobian,

$$d\Omega(\vec{\omega}) := V(x, y) \frac{|\vec{n} \cdot \vec{\omega}|}{\|x - y\|^2} dA(y), \quad (2.26)$$

to fully account for the changes in measure. An inverse squared term appears in Eq. (2.26) because points which are uniformly distributed

in direction space have a density fall-off in Cartesian space inversely proportional to the squared distance $\|\mathbf{x} - \mathbf{y}\|^2$ from the ray origin, \mathbf{x} .

Geometry term. To simplify notation, we group all distance, direction, and visibility terms into a single *geometry term* G ,

$$G(\mathbf{x}, \mathbf{y}) := V(\mathbf{x}, \mathbf{y}) \frac{|\vec{n}_x \cdot \vec{xy}| |\vec{n}_y \cdot \vec{yx}|}{\|\mathbf{x} - \mathbf{y}\|^2}. \quad (2.27)$$

Surface area rendering equation. After applying the change of variables, the surface area formulation of the rendering equation becomes,

$$L_o(\mathbf{x}, \vec{xy}) := L_e(\mathbf{x}, \vec{xy}) + \int_{\mathbb{A}} \rho_s(\mathbf{x}, \vec{xz}, \vec{xy}) G(\mathbf{x}, \mathbf{z}) L_i(\mathbf{x}, \vec{xz}) dA(\mathbf{z}). \quad (2.28)$$

Similarly to the notation in Sec. 2.2.3, the direction of the arrows in Eq. (2.28) always point *away* from the surface point, \mathbf{x} . The outgoing radiance is rewritten as $L_o(\mathbf{x}, \vec{\omega}_o) = L_o(\mathbf{x}, \vec{xy})$, and the BSDF is re-parameterized as $\rho_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) = \rho_s(\mathbf{x}, \vec{xz}, \vec{xy})$, where $\vec{\omega}_i = \vec{xz}$ and $\vec{\omega}_o = \vec{xy}$. See Fig. 2.13 for a geometric visualization of the surface area form and Fig. 2.14 for an illustrative comparison of the directional versus surface area measures.

Surface area measurement equation. We can also re-parameterize the measurement equation (2.19) by redefining the importance as $W_e^{(k)}(\mathbf{x}, \vec{\omega}) = W_e^{(k)}(\mathbf{x}, \vec{xz})$ since $\vec{\omega} = \vec{xz}$ to arrive at,

$$I_k := \int_{\mathbb{A}} \int_{\mathbb{A}} W_e^{(k)}(\mathbf{x}, \vec{xz}) G(\mathbf{x}, \mathbf{z}) L_i(\mathbf{x}, \vec{xz}) dA(\mathbf{x}) dA(\mathbf{z}), \quad (2.29)$$

a reformulation of the measurement equation in Cartesian space.

2.6 The path integral formulation

Both the solid angle form (2.22) and the surface area form (2.28) of the rendering equation are circumstantially useful for a variety of different situations. However, since they are recursive formulations, the algorithms which are naturally derived from them focus on making *locally* good decisions regarding the next direction for light to scatter or when and where to evaluate emission.

While making locally good choices works well for many scenes, there exists more advanced rendering techniques which take into account entire paths, or construct full paths from multiple sub-paths. While we will provide a brief review of these more advanced techniques in Chapter 5, most of them are derived from the non-recursive *path integral* formulation introduced by Veach [11].

As its name implies, the path integral formulation is focused on integrating over full light paths where we define a path of length n as the sequence of

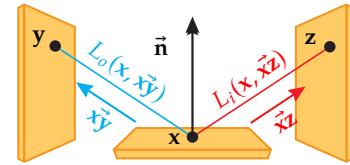
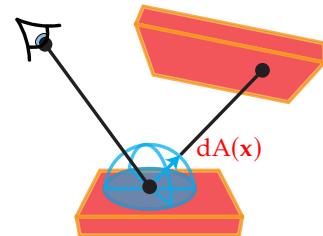


Figure 2.13: An illustration of the geometric configuration in the surface area form of the rendering equation.

Surface area form (2.28)



Directional form (2.22)

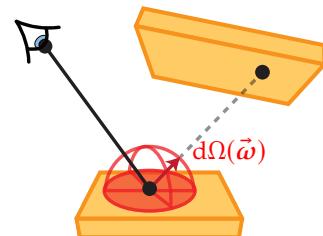


Figure 2.14: We geometrically visualize the difference between the surface area form and the directional form of the rendering equation. Both evaluate the same quantity, however, the surface area form integrates over all surface points, while the directional form integrates over all directions.

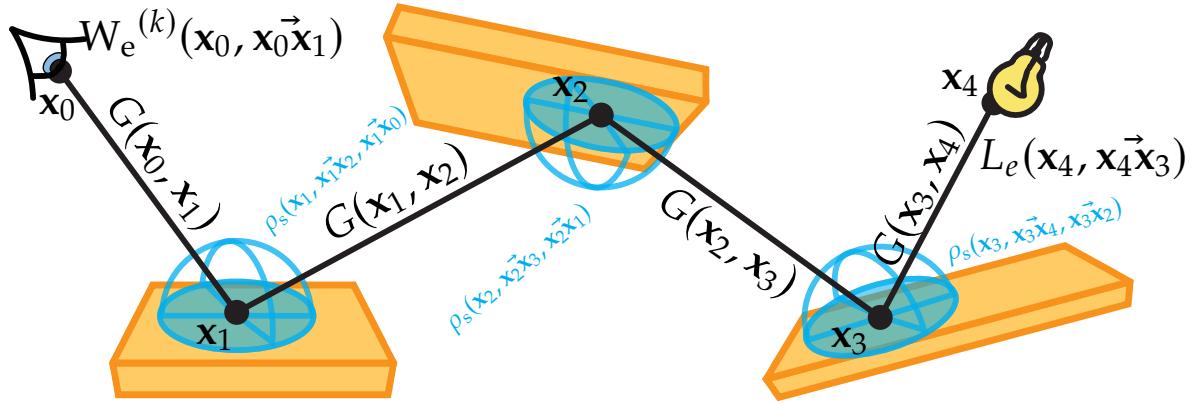


Figure 2.15: A geometrical illustration of the various terms which the path contribution $f^{(k)}(\bar{x})$ for a path with 5 vertices ($\bar{x} = x_0x_1x_2x_3x_4$) is comprised of.

$n + 1$ vertices $x_0 \dots x_n$ along the path. We define the set of all n length light paths as,

$$\mathbb{P}_n := \{x_0 \dots x_n \mid x_0 \dots x_n \in \mathbb{A}\}, \quad (2.30)$$

where x_0 is always on the sensor and x_n is always on a light source (see Fig. 2.15).

Given Eq. (2.30), we then define the entire *path space* as,

$$\mathbb{P} := \bigcup_{j=1}^{\infty} \mathbb{P}_j, \quad (2.31)$$

the union of all sets of n length paths, for all path lengths. We then derive an integral formulation over all of path space by first starting from Eq. (2.29) and plugging in Eq. (2.28),

$$\begin{aligned} I_k &= \int_{\mathbb{A}} \int_{\mathbb{A}} W_e^{(k)}(x_0, x_0 \vec{x}_1) G(x_0, x_1) L_e(x_1, x_1 \vec{x}_0) dA(x_0) dA(x_1) \\ &\quad + \int_{\mathbb{A}} \int_{\mathbb{A}} \int_{\mathbb{A}} W_e^{(k)}(x_0, x_0 \vec{x}_1) G(x_0, x_1) \rho_s(x_1, x_1 \vec{x}_0, x_1 \vec{x}_2) \cdot \\ &\quad \quad G(x_1, x_2) L_e(x_2, x_2 \vec{x}_1) dA(x_2) dA(x_1) dA(x_0) \\ &\quad + \int_{\mathbb{A}} \int_{\mathbb{A}} \int_{\mathbb{A}} \int_{\mathbb{A}} \dots \end{aligned} \quad (2.32)$$

By expanding out the recursion, Eq. (2.32) naturally results in a sum over different length paths,

$$\begin{aligned} I_k &= \sum_{n=1}^{\infty} \int_{\mathbb{A}^{n+1}} W_e^{(k)}(x_0, x_0 \vec{x}_1) G(x_{n-1}, x_n) L_e(x_n, x_n \vec{x}_{n-1}) \cdot \\ &\quad \left[\prod_{j=1}^{n-1} \rho_s(x_j, x_j \vec{x}_{j-1}, x_j \vec{x}_{j+1}) G(x_{j-1}, x_j) \right] dA(x_0) \dots dA(x_n). \end{aligned} \quad (2.33)$$

For each full path $\bar{\mathbf{x}} = \mathbf{x}_0 \dots \mathbf{x}_n$ we specify the *path throughput*,

$$g(\bar{\mathbf{x}}) := G(\mathbf{x}_{n-1}, \mathbf{x}_n) \left[\prod_{j=1}^{n-1} \rho_s(\mathbf{x}_j, \mathbf{x}_j \vec{\mathbf{x}}_{j-1}, \mathbf{x}_j \vec{\mathbf{x}}_{j+1}) G(\mathbf{x}_{j-1}, \mathbf{x}_j) \right], \quad (2.34)$$

as the product of all terms evaluated between the sensor and the light source. The contribution from an entire path on pixel k then becomes,

$$f^{(k)}(\bar{\mathbf{x}}) := W_e^{(k)}(\mathbf{x}_0, \mathbf{x}_0 \vec{\mathbf{x}}_1) g(\bar{\mathbf{x}}) L_e(\mathbf{x}_n, \mathbf{x}_n \vec{\mathbf{x}}_{n-1}), \quad (2.35)$$

the product of the path throughput with the importance and the emission. We visually illustrate the various terms comprising Eq. (2.35) in Fig. 2.15. By replacing the sum over different length paths in Eq. (2.32) with an integral over all of path space,

$$I_k := \int_{\mathbb{P}} f^{(k)}(\bar{\mathbf{x}}) d\mu(\bar{\mathbf{x}}), \quad (2.36)$$

we arrive at the concise *path integral formulation* for light transport where,

$$d\mu(\bar{\mathbf{x}}) := \prod_{j=0}^n dA(\mathbf{x}_j), \quad (2.37)$$

is the product of all area measures.

Fundamentals of Monte Carlo Integration

3

In the previous chapter we derived three different integral formulations for computing the transport of light throughout virtual scenes. Analytically solving these complicated integrals is generally impossible for any but the most simplistic of scenes due to their infinite dimensionality and the computational complexity of the integrand [22]. To handle simulating light transport in interesting virtual environments, we instead have to rely on the use of numerical techniques to *estimate* all of the necessary quantities involved.

This chapter is dedicated to reviewing one such technique known as Monte Carlo integration, which is a stochastic technique that serves as the fundamental backbone of most modern rendering algorithms. Later in Chapter 5, we will review all relevant rendering algorithms discussed within this dissertation and show how they can be derived by applying the Monte Carlo techniques we review in this chapter to the various forms of the rendering equation. All techniques proposed in this dissertation will ultimately rely on or improve upon known Monte Carlo methods reviewed in this section so understanding the fundamentals is imperative before further reading.

The remainder of this chapter is structured as follows: we first review the fundamentals of probability in Sec. 3.1 which are necessary in understanding Monte Carlo methods. We then introduce the concept of Monte Carlo integration in Sec. 3.2 and finish the chapter with an overview of some optimizations in Sec. 3.3 which can be employed to design efficient Monte Carlo estimators.

3.1 Fundamentals of probability

To properly define and introduce the concepts of Monte Carlo integration we first need a basic understanding of probabilities. We briefly review the necessary basics of probability theory needed to understand all future Monte Carlo methods in this section, but refer the interested reader to Grinstead's book [23] for a more formalized introduction into probability theory.

3.1.1 Random variables

A *random variable* X is a function that maps the outcomes of some random process to numbers. Any random variable can be classified as either discrete or continuous. For example, the result of a coin flip is discrete because its potential outcomes are binary (heads or tails), while the amount of rain in inches measured within some region per year is an example of a continuous random variable. While a random variable may commonly be

3.1 Fundamentals of probability	18
3.1.1 Random variables	18
3.1.2 Distribution functions	19
3.1.3 Expected value and variance	21
3.2 Basics of Monte Carlo Integration	22
3.2.1 Primary vs. secondary estimation	23
3.2.2 Unbiasedness	23
3.2.3 Consistency	24
3.2.4 Convergence	25
3.2.5 Discrete form	27
3.3 Estimator optimizations	28
3.3.1 Importance Sampling	28
3.3.2 Multiple Importance Sampling	29
3.3.3 The Inversion Method	30
3.3.4 Correlations	30
3.3.5 Control Variates	32
3.3.6 Multilevel Monte Carlo	32

[22]: Arvo (1995), 'Analytic Methods for Simulated Light Transport'

[23]: Grinstead et al. (2003), *Introduction to Probability*

thought of as returning or representing a single outcome, it is typically more meaningful to consider random variables as representing the *entire* distribution of all possible outcomes.

Independent vs. dependent. Two random variables, X and Y , can either be considered independent or dependent with respect to one another. If X and Y are independent random variables, then the probability of Y occurring given the fact that X has occurred,

$$P(Y|X) = P(Y), \quad (3.1)$$

remains unchanged. For example, the individual outcomes of flipping two fair coins will not influence one another, so they are independent.

On the other hand, if X and Y are *dependent* then one or both of their outcomes will be influenced by the other. For example, consider the case where X and Y represent the outcomes of the height and weight of an average person once they reach adulthood. You would expect that as someone's height increases, their weight would also increase. This specific relationship is known as X and Y exhibiting *positive correlations* since the outcomes of both variables change in the same direction with each other.

Certain mathematical operations or properties involving multiple random variables will change depending on whether the random variables are independent or dependent. In some circumstances, we will be able to control whether or not two random variables are correlated (dependent) through design decisions in the algorithms which we introduce. Sometimes this is beneficial and can lead to more efficient algorithms, while other times it can result in *incorrect* results. We will discuss this more in Sec. 3.3.4.

3.1.2 Distribution functions

In this subsection we review all of the relevant distribution functions which will appear throughout this dissertation and how they interrelate.

Probability functions. The *probability density function* (PDF) of a continuous random variable X , denoted $p(x)$, describes the probability density, or likelihood, that a specific outcome x occurs. A valid PDF is one which,

$$\int_{\Theta} p(x)dx = 1, \quad (3.2)$$

integrates to unity over the entire domain, Θ , spanning all the possible outcomes for the variable X . A valid PDF has to also be non-negative $p(x) \geq 0$, since probabilities cannot be negative.

The *probability mass function* (PMF) is the discrete analog to the PDF for a discrete random variable X . Since it is typically contextually obvious when we are dealing with continuous random variables or discrete random variables, we choose to denote the PMF of a discrete outcome j occurring

in a notationally similar way ($p(j)$) to how we denoted the PDF ($p(x)$). A valid PMF also follows the same properties as the PDF,

$$\sum_{j \in \Theta} p(j) = 1, \quad (3.3)$$

where it sums to unity over the entire domain Θ spanning all outcomes and it has to be non-negative.

Cumulative functions. The *cumulative distribution function* (CDF) of a continuous random variable X ,

$$\text{cdf}(X) := P\{X \leq x\} = \int_{\Theta^-}^x p(y)dy, \quad (3.4)$$

describes the probability that some random value chosen from the probability distribution is less than or equal to x . We use Θ^- here to denote the start of some general domain Θ . Since the PMF is equivalent to the integral of the PDF,

$$p(j) = \int_j^{j+1} p(x)dx, \quad (3.5)$$

between its discrete bounds, the PMF can also be described as $p(j) = \text{cdf}(j + 1) - \text{cdf}(j)$. The PDF is then the infinitesimal limit of the area covered by the PMF,

$$p(x) = \lim_{\Delta \rightarrow 0} \frac{\text{cdf}(x) - \text{cdf}(x + \Delta)}{\Delta}, \quad (3.6)$$

which coincides with the limit definition of the derivative. So, the PDF can thus be derived from the CDF by,

$$p(x) := \frac{d}{dx} \text{cdf}(x). \quad (3.7)$$

Since the PDF is a non-negative function, the CDF has to be a monotonically increasing function.

The discrete analog to the CDF is the *cumulative mass function* (CMF),

$$\text{cmf}(x) := P\{X \leq x\} = \sum_{y=\Theta^-}^x p(y), \quad (3.8)$$

which measures the same quantity but for a discrete random variable.

The *complementary cumulative distribution function* (CCDF) is the complement of the CDF,

$$\begin{aligned} \text{ccdf}(X) &:= 1 - P\{X \leq x\} = 1 - \int_{\Theta^-}^x p(y)dy, \\ &= P\{X \geq x\} = \int_x^{\Theta^+} p(y)dy. \end{aligned} \quad (3.9)$$

A very common example of a CCDF which appears throughout volumetric light transport is the transmittance function (see: Sec. 4.2.1). The CCDF is also necessary to compute when attempting to sample points distributed proportionally to a PDF which we will review in Sec. 3.3.3. We visualize what continuous distribution functions and discrete mass functions might look like for an exponential distribution in Fig. 3.1.

3.1.3 Expected value and variance

The *expected value* of a continuous random variable, $X = f(x)$, over the domain Θ is defined as,

$$\mathbb{E}[X] := \int_{\Theta} f(x)p(x)dx, \quad (3.10)$$

which effectively integrates the product of the random variable's outcomes with the likelihood of those outcomes (the PDF).

Properties of the expected value. For any constants c and b , and random variables X and Y , the expected value exhibits the following mathematical properties,

- **Linearity.**

$$\begin{aligned} \mathbb{E}[cX + b] &= c\mathbb{E}[X] + b, \\ \mathbb{E}[X + Y] &= \mathbb{E}[X] + \mathbb{E}[Y]. \end{aligned} \quad (3.11)$$

- **Product of random variables.** If X and Y are *independent* random variables then the following holds,

$$\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y], \quad (3.12)$$

however, if X and Y are *dependent* random variables this is no longer guaranteed to be the case.

- **Non-linear transformations.** Let g be some non-linear function. It is generally the case that,

$$\mathbb{E}[g(X)] \neq g(\mathbb{E}[X]), \quad (3.13)$$

and has been shown through Jensen's inequality [24] that if g is a convex function then,

$$\mathbb{E}[g(X)] \geq g(\mathbb{E}[X]), \quad (3.14)$$

and vice versa if g is a concave function. As pointed out in Chapter 1, this property of the expected value is very troublesome for deriving stochastic solutions for problems in light transport. The majority of this dissertation will focus on situations where we want to compute the quantity $g(\mathbb{E}[X])$ and we introduce both general and problem-specific techniques for doing so. The problem of generally computing $g(\mathbb{E}[X])$ is known as computing *functions of expectations*.

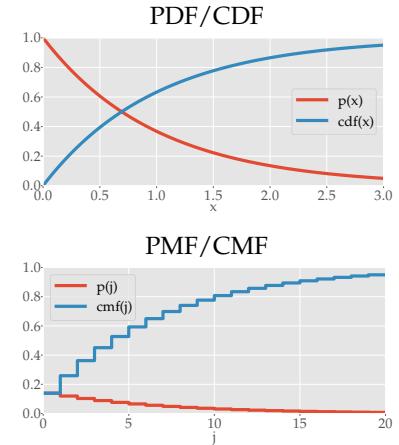


Figure 3.1: A visualization of a PDF and CDF (top) and PMF and CMF (bottom) for the same exponential distribution $p(x) = 1 * e^{-1}(x)$ visualized from $x = [0, 3]$. Note: one j covers a step size of roughly 0.15.

[24]: Jensen (1906), 'Sur les fonctions convexes et les inégalités entre les valeurs moyennes'

Variance and standard deviation. The *variance* of a random variable X ,

$$\text{V}[X] := \text{E}[(X - \text{E}[X])^2] = \text{E}[X^2] - \text{E}^2[X], \quad (3.15)$$

measures the squared deviation between all outcomes for X with the expected outcome, $\text{E}[X]$. Some mathematical properties for the variance are,

► **Variance of summations.**

$$\begin{aligned} \text{V}[X + c] &= \text{V}[X], \\ \text{V}[X + Y] &= \text{V}[X] + \text{V}[Y], \text{ if uncorrelated.} \end{aligned} \quad (3.16)$$

► **Variance of products.**

$$\begin{aligned} \text{V}[cX] &= c^2\text{V}[X], \\ \text{V}[XY] &= \text{V}[X]\text{V}[Y], \text{ if uncorrelated.} \end{aligned} \quad (3.17)$$

The *standard deviation* of a random variable is defined as $\sigma[X] := \sqrt{\text{V}[X]}$.

3.2 Basics of Monte Carlo Integration

Monte Carlo integration is the fundamental building block for most notable rendering algorithms used today. It is employed to derive *estimators* for quantities which are the direct result of integrals or summations. As an example, let us assume that we want to compute some desired quantity,

$$I := \int_{\Theta} f(x)dx, \quad (3.18)$$

which is the result of the integral of some function $f(x)$ over some arbitrary domain Θ . Let us assume that the result of the integral is finite and we cannot compute it analytically. For example, either the domain Θ is too difficult to integrate over as is the case when dealing with integrals over the entirety of path space for scenes of non-trivial complexity, or $f(x)$ is a function that is akin to a *black box* where its general shape is unknown but it can be evaluated at any specific x .

When no analytical solution is available we can instead rely on Monte Carlo integration to *estimate* the quantity I by first defining a valid PDF $p(x)$ (Sec. 3.1.2) over the entirety of the domain Θ that we can sample from. By sampling a point x' from the PDF, we can estimate I as,

$$\langle I \rangle := \frac{f(x')}{p(x')}. \quad (3.19)$$

We employ the notation $\langle \rangle$ to refer to an *estimator* for some quantity. In terms of the probability topics which we have reviewed in the prior section, a Monte Carlo estimator can itself be thought of as a random variable ($\langle I \rangle = X$) with its own expected value and variance.

3.2.1 Primary vs. secondary estimation

Additionally, we refer to Eq. (3.19) as a *primary estimator* since it uses only *one* Monte Carlo sample (x') to estimate the integral. We could alternatively define a *secondary estimator*,

$$\langle I \rangle_n := \frac{1}{n} \sum_{k=1}^n \frac{f(x_k)}{p(x_k)} = \frac{1}{n} \sum_{k=1}^n \langle I \rangle_1 \quad (3.20)$$

which takes the average of n primary estimators. While it is typically assumed that all n estimators are independently evaluated and are identical, they can also be correlated or different primary estimators. We briefly discuss correlations in more detail in Sec. 3.3.4 when we introduce techniques for improving the performance of estimators.

Properties of Monte Carlo estimators. Monte Carlo is an incredibly powerful estimation tool for three key reasons:

- ▶ **Unbiasedness.** Monte Carlo estimators can be constructed to always provide *unbiased* results.
- ▶ **Consistency.** Secondary Monte Carlo estimators are always *consistent* if the primary estimators are unbiased.
- ▶ **Convergence.** The *convergence* behavior of Monte Carlo is typically well understood, adequately performant for practical applications, and is *independent* of the dimensionality of the problem.

In the remainder of this section, we will elaborate on and prove these three properties.

3.2.2 Unbiasedness

An estimator for some quantity, $\langle I \rangle$, is *unbiased* if its expected value equals the true solution, $E[\langle I \rangle] = I$. As previously defined in Eq. (3.10), the expected value of a random variable is the integral of the product between all possible outcomes and the probability density, or likelihood, of those outcomes occurring. Since a Monte Carlo estimator can itself be thought of as a random variable, its expected value is defined as,

$$E[\langle I \rangle] := \int_{\Theta} \langle I \rangle p(x) dx. \quad (3.21)$$

Assuming that the probability density function is a valid PDF, meaning the conditions in Eq. (3.2) hold, and,

$$p(x) \geq 0 \quad \forall x \text{ where } f(x) \geq 0, \quad (3.22)$$

then,

$$\begin{aligned}
 E[\langle I \rangle] &= \int_{\Theta} \langle I \rangle p(x) dx \\
 &= \int_{\Theta} \frac{f(x)}{p(x)} p(x) dx \\
 &= \int_{\Theta} f(x) dx \\
 &= I,
 \end{aligned} \tag{3.23}$$

the primary Monte Carlo estimator is an *unbiased* estimator for the result of an integral.

Discrete case. The same properties apply to the discrete case where the expected value becomes,

$$\begin{aligned}
 E[\langle I \rangle] &= \sum_{x \in \Theta} \langle I \rangle p(x) \\
 &= \sum_{x \in \Theta} \frac{f(x)}{p(x)} p(x) \\
 &= \sum_{x \in \Theta} f(x) \\
 &= I.
 \end{aligned} \tag{3.24}$$

3.2.3 Consistency

Unbiasedness states that the expected value of an estimator is correctly equal to the true solution, however, this does not mean that an estimator has *zero error*. For example, consider the act of rolling a fair six-sided die whose sides are numbered $\{1, 2, 3, 4, 5, 6\}$. In expectation, we would expect to roll exactly 3.5, however, if the die is only rolled once, i.e. a primary Monte Carlo estimator, we will never arrive at the true expected value so there will always be *error*.

Error vs. bias. The existence of error does not imply that an estimator is ultimately incorrect. If an estimator is *expected to be incorrect*, we call it *biased*. The difference between error and bias is that we define the *error* for some secondary estimator $\langle I \rangle_n$ as $E[\langle I \rangle_n] = \langle I \rangle_n - I$, whereas the *bias* for that same estimator is $B[\langle I \rangle_n] = E[\langle I \rangle_n - I]$ the expected error.

A *consistent* estimator is one where the error will converge in probability in the limit of Monte Carlo samples,

$$\lim_{n \rightarrow \infty} P(|\langle I \rangle_n - I| > \epsilon) = 0; \quad \forall \epsilon > 0. \tag{3.25}$$

Formally speaking, Eq. (3.25) states that if a secondary estimator is consistent, then the probability there will exist any error after taking an infinite number of samples is zero. In statistics, $\langle I \rangle_n$ can take any functional form which is dependent on n , so Eq. (3.25) is a strict condition which has

to be proven to show consistency. However, for our purposes, proving consistency is much simpler than showing that Eq. (3.25) holds since we can rely on the law of large numbers.

Law of large numbers. The *law of large numbers* has two different forms which both prove consistency. The *strong law of large numbers* states that if we have an infinite sequence of i.i.d. random variables $X_1, X_2 \dots$ with finite mean μ , then their running average will converge to the true mean,

$$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{k=1}^n X_k = \mu, \quad (3.26)$$

in the limit of the number of variables which are averaged together. The *weak law of large numbers* upholds Eq. (3.25) by proving that the sample mean of a sequence of i.i.d. random variables will also converge in probability.

The law of large numbers effectively states that the sample mean is a consistent estimate for the true mean. If we consider the result returned by a Monte Carlo primary estimator $\langle I \rangle_1$ to be a random variable, a secondary estimator, $\langle I \rangle_n$, then becomes a sample mean (3.20). Thus, by the law of large numbers, any unbiased secondary Monte Carlo estimator is consistent so the error will disappear in the limit.

For a more theoretical discussion on the law of large numbers and its applications, we refer the interested reader to Ibe's book on applied probability [25].

[25]: Ibe (2014), *Fundamentals of Applied Probability and Random Processes*

Unbiasedness does not imply consistency. From the aforementioned argument by the strong law of large numbers, any unbiased secondary Monte Carlo estimator with finite expected value will be consistent. However, an estimator being unbiased does not imply consistency. Any primary Monte Carlo estimator by itself will never be consistent unless it always returns the correct result. Additionally, it is possible for consistent estimators to be formulated out of sequences of biased, unbiased, and combinations of both biased and unbiased estimators. In later chapters we will show how this is possible, as well as discuss situations where using biased, but consistent, estimators can be more beneficial than using unbiased ones.

3.2.4 Convergence

The unbiasedness property states that the expectation of Monte Carlo estimators are correct; consistency states that unbiased secondary Monte Carlo integration will converge to zero error, however, as practitioners, we are also interested in the rate at which our secondary estimators $\langle I \rangle_n$ will converge as we increase n , the number of samples. To define these rates, we need to first quantify how "converged" our estimates are.

In this dissertation, there are three different quantities we are interested in computing or measuring to quantify how converged an estimator is.

Namely, the *absolute error*, the *variance*, and the *mean squared error* of our estimators.

Variance. We previously introduced the variance of a random variable in Eq. (3.15). Since Monte Carlo estimators can be considered random variables, the variance of a Monte Carlo estimator can be described as,

$$\text{V}[\langle I \rangle_n] := \text{E}[(\langle I \rangle_n - \text{E}[\langle I \rangle_n])^2]. \quad (3.27)$$

The variance is effectively a measure of the squared variation of any individual invocation of an estimator from its expected value.

Mean-squared error. Notice that the variance is a measure of the squared error from the expected value of the estimator instead of the true solution I . The variance is not a quantity that will reflect how *biased* a solution is rather just a measure of how the estimator deviates from the expected sample mean. We instead use the mean-squared error (MSE),

$$\text{MSE}[\langle I \rangle_n] := \text{V}[\langle I \rangle_n] + \text{B}[\langle I \rangle_n]^2 = \text{E}[(\langle I \rangle_n - I)^2], \quad (3.28)$$

to provide a measure of the squared deviation while additionally accounting for any potential bias through reporting the sum of the variance and the squared bias. For any unbiased estimator where $\text{B}[\langle I \rangle_n] = 0$, the variance and MSE will be equivalent. Since this dissertation will deal with both biased and unbiased estimators, we will frequently use MSE instead of variance for all quantitative comparisons.

Absolute error. The absolute error,

$$\text{AbsErr}[\langle I \rangle_n] := |\text{e}[\langle I \rangle_n]| = |\langle I \rangle_n - I|, \quad (3.29)$$

conveys the absolute deviation from the true result. Unlike the variance and MSE, this quantity does not measure the squared deviation, but rather the *true deviation*. There arises situations covered by this dissertation where the variance and MSE will become *infinite* while the underlying estimator will still converge to the true solution. In such situations, the absolute error is a more useful tool for measuring convergence.

Monte Carlo convergence rates. In practice, we are interested in employing consistent algorithms where the error disappears in the limit of samples. This is why the majority of modern rendering algorithms take the form of secondary Monte Carlo estimators. To understand the performance of these techniques we are typically interested in the *convergence rate* of these algorithms as we increase n .

Assuming we have an unbiased secondary Monte Carlo estimator exhibiting finite variance, the asymptotic convergence rates for both its absolute error and variance are well known. As we increase n , the error will converge at a rate of $\mathcal{O}(n^{-1/2})$ while the variance (and equivalently the MSE since

the estimator is unbiased) will converge at a rate of $\mathcal{O}(n^{-1})$, assuming independent invocations of the primary estimator [26]. This means that to reduce the current error by a factor of two we would have to quadruple the number of samples, while to reduce the variance by a factor of two, we would have to double n .

The majority of existing rendering algorithms fall under the classification of secondary Monte Carlo estimators whose primary estimates have finite variance, so they will exhibit the same asymptotic convergence rates while differing by only a constant factor. However, this "constant factor" can be significant, and we will see in Chapter 5 that some algorithms are more preferable to others for simulating certain visual effects. However, there exists techniques introduced in this dissertation that exhibit slower than normal Monte Carlo convergence rates due to the primary estimator having *infinite variance*.

Infinite variance. Historically, at least within the rendering literature, variance and MSE have been the primary means of representing how converged an estimator is. However, if a primary estimator has infinite variance both of these quantities will no longer be practically useful since they will be infinite. Before the contributions in this dissertation were introduced, infinite variance estimators within the rendering literature have been sparsely explored. One example being instant radiosity [27–29].

Secondary estimators which have finite expectation but whose primary estimators have infinite variance will still remain consistent. Meaning, in the infinite limit the error of infinite variance estimators will converge to zero. However, due to the heavy tails within the distribution of the outcomes, the variance and MSE might never converge. This is proven by the strong law of large numbers (3.26) as it proves convergence of the absolute error $\text{AbsErr}[\langle I \rangle_n]$ *independent* of the estimator's variance.

[26]: Caflisch (1998), 'Monte Carlo and quasi-Monte Carlo methods'

[27]: Keller (1997), 'Instant Radiosity'

[28]: Laine et al. (2007), 'Incremental Instant Radiosity for Real-Time Indirect Illumination'

[29]: Hašan et al. (2009), 'Virtual Spherical Lights for Many-Light Rendering of Glossy Scenes'

3.2.5 Discrete form

Monte Carlo integration can also be applied to discrete summations as opposed to integrals,

$$I := \sum_{j=1}^N f(j); \quad \langle I \rangle := \frac{f(j)}{p(j)}. \quad (3.30)$$

The only difference between the estimator in Eq. (3.30) and its integral variation (3.19) is that instead of using a PDF, the discrete form employs a PMF, defined for all $j \in [1, N]$ across the entire discrete domain. The expected value is then redefined as a summation instead of an integral of the form,

$$\mathbb{E}[\langle I \rangle] := \sum_{j=1}^N \langle I \rangle p(j), \quad (3.31)$$

and all other properties and quantities follow suit.

3.3 Estimator optimizations

We have introduced an overview of Monte Carlo integration which can be used to formulate stochastic estimators for integral and summation problems in light transport. These estimators naively result in convergence rates which are efficient enough for use in practice when the variance is finite. However, there are a variety of design decisions that arise when developing Monte Carlo estimators allowing us to further optimize their performance. In this section, we will review some general techniques for devising efficient Monte Carlo estimators. Later on in this dissertation we will more thoroughly discuss specific estimators which utilize these concepts and expand upon these optimizations for specific problems.

3.3.1 Importance Sampling

Monte Carlo integration involves sampling integrals or summations proportionally to some user-specified PDFs or PMFs, however, the choice of a PDF or PMF can heavily impact the performance of primary Monte Carlo estimators. Choosing a good PDF or PMF is one of the most critical design decisions in devising efficient Monte Carlo estimators, which is why it is step ③ in our recipe for formulating consistent estimators. However, to define a "good" PDF or PMF we should first understand what the *best* probability distribution to use is.

The best choice for a probability distribution coincides with a Monte Carlo estimator which *always* returns the correct answer. In essence, an estimator which has *zero-variance* and zero-error. This occurs when we sample from a distribution that is *exactly proportional* to our function of interest, $p(x) \propto f(x)$.

Proof. Assume that we define a PDF to be $p(x) = f(x)$. This is not guaranteed to be a valid PDF unless the function integrates to unity, however, assume there exists a normalization constant, $H := \int_{\Theta} f(x)dx$, such that $p(x) = f(x)/H$ is a valid PDF. This normalization constant is exactly equal to the quantity which we want to estimate. Thus, given $p(x) \propto f(x)$, a primary Monte Carlo estimator,

$$\begin{aligned} I &:= \int_{\Theta} f(x)dx \\ \langle I \rangle &:= \frac{f(x')}{p(x')} = \frac{Hf(x')}{f(x')} = H = \int_{\Theta} f(x)dx, \end{aligned} \tag{3.32}$$

will always result in the correct answer while maintaining zero-variance no matter what x' we sample.

Practical choice. While sampling proportionally to our integral is the ideal choice, in practice we typically deal with the problem of estimating high dimensional integrals containing the product of *many* low dimensional quantities such as individual BSDFs. While we can sample proportionally to many of the individual low-dimensional quantities easily, sampling proportionally to the entire high-dimensional product is a non-trivial problem¹ [30–32]. In practice, we typically choose to importance sample the individual terms of the integrand that we can importance sample exactly.

3.3.2 Multiple Importance Sampling

Within a high-dimensional integrand there may exist many different terms which we can choose to importance sample independently. Depending on the context, some terms may be more "important" to sample than others, while some sampling schemes may even perform exceptionally poorly in certain situations. Knowing which term to importance sample in the presence of many for all contexts is a non-trivial problem. A more robust technique was proposed by Veach which *combines* multiple different importance sampling strategies together to alleviate this, and it is known as *multiple importance sampling* (MIS) [33].

MIS involves generating samples from all importance sampling techniques while using a weighted combination of the techniques to compute the final estimate. Given n different importance sampling schemes that produce samples distributed proportionally to n different probability distributions, $p_1(x), \dots, p_n(x)$, an MIS estimator takes the form of,

$$\langle I \rangle_{\text{MIS}} := \sum_{j=1}^n w_j(x) \frac{f(x)}{p_j(x)} \quad (3.33)$$

where $w_j(x)$ is the MIS weight of the j -th sampling scheme. Any choice of MIS weights such that $\sum_{j=1}^n w_j(x) = 1$, will result in a valid MIS heuristic leading to an unbiased estimator if all n importance sampling schemes are independently unbiased estimators [33].

However, we are mostly interested in choosing weights that provably reduce variance over using either a single estimator or their unweighted average. Veach additionally introduced several weighting strategies with provably good performance, meaning any alternative strategies will not significantly outperform them. One of the most commonly used strategies is known as the balance heuristic,

$$w_j(x) := \frac{p_j(x)}{\sum_{k=1}^n p_k(x)}, \quad (3.34)$$

which weights techniques by the relative magnitudes of their probability distributions.

While no other weighting schemes or more advanced importance sampling techniques will be employed by the work covered within this dissertation, there exists a plethora of other techniques such an extension to support a

1: While sampling proportionally to high dimensional integrands is the topic of active research within the field of path-guiding [30–32], most of those concepts are orthogonal to the contributions of this dissertation.

[30]: Müller et al. (2017), 'Practical Path Guiding for Efficient Light-Transport Simulation'

[31]: Zhu et al. (2021), 'Photon-Driven Neural Reconstruction for Path Guiding'

[32]: Rath et al. (2020), 'Variance-Aware Path Guiding'

[33]: Veach et al. (1995), 'Optimally Combining Sampling Techniques for Monte Carlo Rendering'

continuum of importance sampling strategies [34], resampled importance sampling [35], and reservoir resampling [36, 37].

3.3.3 The Inversion Method

Up until this point, we have discussed the idea of sampling points proportionally to a probability distribution, however, we have not yet described how that is accomplished. One such technique is the *inversion method* which is also known as *inverse transform sampling*. This technique is preferred when it is applicable because it results in importance sampling schemes which *perfectly* sample from a desired distribution.

The inversion method requires that we can compute the CDF of a probability distribution (3.4) and its inverse, $\text{cdf}^{-1}(\xi)$, in closed form for any uniform random value $\xi \in [0, 1]$. If this is possible, then uniformly sampling ξ and passing it through the inverse CDF will result in perfectly importance sampling a point x from the probability distribution $p(x)$. For more information regarding this technique, we refer the interested reader to Matt Pharr's book [18], and for an example of its application see Eq. (5.24).

[34]: West et al. (2020), 'Continuous Multiple Importance Sampling'

[35]: Talbot et al. (2005), 'Importance Resampling for Global Illumination'

[36]: Chao (1982), 'A General Purpose Unequal Probability Sampling Plan'

[37]: Bitterli et al. (2020), 'Spatiotemporal Reservoir Resampling for Real-Time Ray Tracing with Dynamic Direct Lighting'

[18]: Pharr et al. (2016), *Physically Based Rendering*

3.3.4 Correlations

We saw in Sec. 3.3.1 that we can improve the performance of our estimators by importance sampling known terms, and we can use techniques like the inversion method (Sec. 3.3.3) to perfectly sample points proportionally to those terms. When using the inversion method we typically choose to distribute samples randomly using uniform random variables $\xi \in [0, 1]$, however, we can further improve the performance of our estimators by carefully choosing to *correlate* those samples.

Stratified sampling. Generally speaking, it is typically preferred that we sample points such that we have a good understanding of the underlying solution space. As a real life example, it might be of little interest for a political poll to only sample opinions from one demographic or "clump" of people. Randomly distributing samples in the space of all possible samples can result in undesirable "clumping" (see: Fig. 3.2).

Instead we can choose to reinterpret the problem of estimating an integral using n samples as estimating n different partial integrals,

$$\begin{aligned} I &:= \int_a^b f(x)dx = \sum_{j=0}^{n-1} \int_{(j)\Delta}^{(j+1)\Delta} f(x)dx, \\ \langle I \rangle_{\text{Strat}} &:= \sum_{j=0}^{n-1} \frac{f(x_j)}{p(x_j)}, \end{aligned} \tag{3.35}$$

using a single sample each where $\Delta := \frac{b-a}{n}$ is the integration range of each integral. Each one of the partial integrals are referred to as *strata* and this form of correlated sampling is known as stratified sampling.

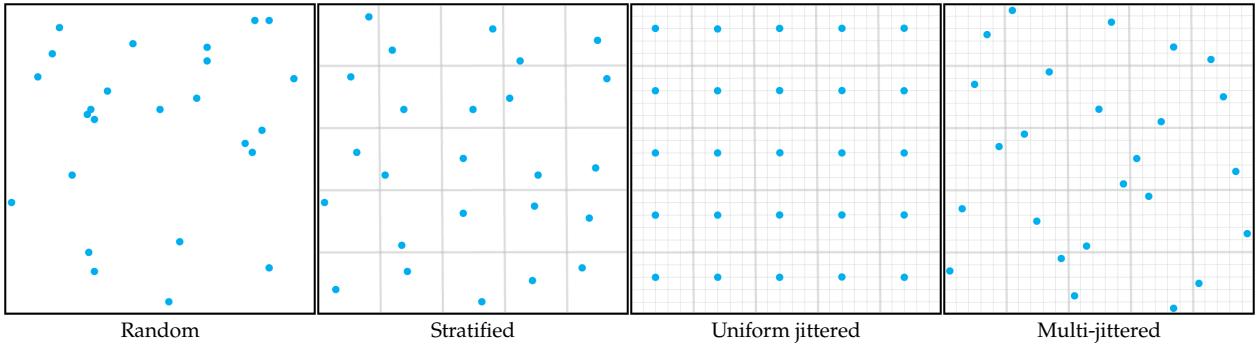


Figure 3.2: A visualization of four different sampling patterns. From left to right: purely randomized sampling (which suffers from unwanted clumping), stratified random sampling, uniformly-jittered random sampling, and multi-jittered random sampling.

Uniform jittered sampling. *Uniform jittered-sampling* is a slight modification to stratified sampling where instead of sampling a different random point per partial integral, we can instead sample one displacement $\xi \in [0, \Delta)$ uniformly,

$$\langle I \rangle_{\text{Jitt}} := \sum_{j=0}^{n-1} \frac{f(j\Delta + \xi)}{p(\xi)}, \quad (3.36)$$

which is used to evaluate sample points $x_j := j\Delta + \xi$ for all partial integrals [38]. This results in a regular shaped sampling pattern which gets randomly "jittered" at every invocation [39]. We illustrate these different sampling techniques for a two dimensional sample space as well as multi-jittered sampling in Fig. 3.2.

[38]: Ramamoorthi et al. (2012), 'A Theory of Monte Carlo Visibility Sampling'

[39]: Pauly et al. (2000), 'Metropolis Light Transport for Participating Media'

Performance improvements. Stratified sampling can never result in higher variance than naive random sampling, and can even result in *asymptotically faster* convergence rates [26]. Thus, we will attempt to correlate samples whenever possible.

[26]: Caflisch (1998), 'Monte Carlo and quasi-Monte Carlo methods'

Bias. Taking advantage of the performance improvements from correlated samples is not always beneficial. Many times the integrand we are interested in evaluating is recursive in nature and thus involves taking the running product of many different terms once expanded. Since the expected value of the product of two correlated random variables is *not* equal to the product of their expectations (3.12), correlating samples across a product can result in *biased* estimators. Practical implementations have to be constructed such that correlations only exist across summations to maintain the unbiased property of Monte Carlo integration.

[40]: McKay et al. (1979), 'A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code'

[41]: Shirley (1990), 'Physically Based Lighting Calculations for Computer Graphics'

[42]: Wang et al. (1999), 'Multi-Stage N-Rooks Sampling Method'

[43]: Chiu et al. (1994), 'Multi-Jittered Sampling'

[44]: Owen (1992), 'Orthogonal Arrays for Computer Experiments, Integration and Visualization'

[45]: Jarosz et al. (2019), 'Orthogonal Array Sampling for Monte Carlo Rendering'

Other stratification techniques. Other techniques which introduce different stratification strategies include Latin hypercube sampling [40], N-rooks sampling [41], multi-stage N-rooks sampling [42], multi-jittered sampling [43], and orthogonal array-sampling [44]. For an in depth discussion of these more advanced strategies, we refer the interested reader to Jarosz's work on adapting orthogonal array-sampling for rendering [45].

3.3.5 Control Variates

Control variates are another variance reduction technique which relies on having some prior knowledge regarding the approximate shape of the function of interest, $f(x)$. The main idea behind control variates is that if we have some function $h(x)$ that is analytically integrable, $\int_{\Theta} h(x)dx = H$, and roughly matches $f(x)$ in shape, then by formulating their difference,

$$I = \int_{\Theta} f(x)dx = \int_{\Theta} f(x) - h(x)dx + H, \quad (3.37)$$

we can derive an estimator,

$$\langle I \rangle = \frac{f(x) - h(x)}{p(x)} + H, \quad (3.38)$$

that will be more efficient if $f(x)$ and $h(x)$ are correlated. In practice, when dealing with the problem of estimating an integral, $h(x)$ should be chosen such that $f(x) - g(x)$ is nearly constant.

Control variates are used frequently within the financial literature [46] and will appear frequently within the work proposed by this dissertation. Recently, a recursive form of control variates was employed to aid in the field of inverse rendering [47].

[46]: Glasserman (2003), 'Monte Carlo Methods in Financial Engineering'

[47]: Nicolet et al. (2023), 'Recursive Control Variates for Inverse Rendering'

3.3.6 Multilevel Monte Carlo

Multilevel Monte Carlo (MLMC) is an advanced Monte Carlo technique which is commonly used in situations where the underlying problem being estimated is too costly to be estimated directly but can instead be *approximated* cheaply. This technique was originally developed for estimating stochastic differential equations using path simulation [48]. MLMC methods have frequently appeared within the financial literature [49] and while slightly outdated, we refer the interested reader to Giles' review of modern MLMC [50] for a more detailed overview of its other applications.

[48]: Giles (2008), 'Multilevel Monte Carlo Path Simulation'

[49]: Sinha et al. (2022), *Multilevel Monte Carlo and its Applications in Financial Engineering*

[50]: Giles (2015), 'Multilevel Monte Carlo methods'

MLMC expands upon the idea of control variates (3.38) where we replace $h(x)$ with a *biased* estimator that is assumed to be less costly than directly estimating Eq. (3.37). Assuming that we can formulate Monte Carlo estimators for different approximation "levels" $\langle I \rangle_n^l$ where n refers to the number of secondary samples, and l is used to refer to the approximation level. The underlying problem which we want to evaluate is assumed to only result in the true expected value $E[\langle I \rangle_n^L] = I$ when evaluated up to some late level L . Evaluating level L is assumed to be computationally expensive, so we can formulate a telescoping estimator,

$$\langle I \rangle_{MLMC} := \langle I \rangle_n^0 + \sum_{l=0}^L \langle I \rangle_n^{l+1} - \langle I \rangle_n^l. \quad (3.39)$$

which combines different estimates within an unbiased "multi-level" formulation.

Proof. Here we prove that Eq. (3.39) is unbiased,

$$\begin{aligned}
 E[\langle I \rangle_{MLMC}] &= E\left[\langle I \rangle_n^0 + \sum_{l=0}^L \langle I \rangle_n^{l+1} - \langle I \rangle_n^l\right] \\
 &= E[\langle I \rangle_n^0] + \sum_{l=0}^L E[\langle I \rangle_n^{l+1}] - E[\langle I \rangle_n^l] \\
 &= E[\langle I \rangle_n^0] + (E[\langle I \rangle_n^1] - E[\langle I \rangle_n^0]) + \dots + (E[\langle I \rangle_n^L] - E[\langle I \rangle_n^{L-1}]) \\
 &= E[\langle I \rangle_n^0] - E[\langle I \rangle_n^0] + E[\langle I \rangle_n^1] - \dots - E[\langle I \rangle_n^{L-1}] + E[\langle I \rangle_n^L] \\
 &= E[\langle I \rangle_n^L] \\
 &= I.
 \end{aligned} \tag{3.40}$$

By formulating an estimator as a telescoping series of approximate estimates, MLMC allows for focusing the majority of the samples on the earlier levels which are computationally less costly to estimate than the later levels. An additional underlying assumption of MLMC is that the variance *decreases* as the level l increases, implying that less samples are needed for later levels. If each level is correlated with each other at the sample level, MLMC can be thought of as a recursive application of control variates.

In Chapter 8 we will show how we can derive unbiased estimators for any consistent problem by expanding upon the base theory of MLMC.



4 Light transport in participating media

In Chapter 2 we made the assumption that the radiance leaving a surface would remain constant until the next surface interaction (2.11). This assumption remains true as long as the two surfaces, and the space directly between the two surface interactions, exist within a vacuum. However, in any scene that we would perceive in real life this is never truly the case. The "empty" space around us is filled with particles of air which can *participate* in the interactions between light and the environment. This allows light to be absorbed, scattered, and emitted within volumes as well as surfaces.

The concept of volumetric interactions are not limited to atmospheric interactions, but also encompass the interactions light can have with *any* volumetric medium such as skin, clouds, stained glass, etc. We generally consider all of these cases to fall within the theory of *volumetric light transport*, which is fundamentally derived from the theory of *radiative transfer* [1]. The theory of radiative transfer governs the energy transfer of electromagnetic radiation through general participating media, encompassing the propagation of light through volumes.

In the remainder of this chapter, we extend the fundamentals of light transport that we introduced in Chapter 2 to directly account for the existence of participating media. We first introduce the fundamental properties of a participating medium and review the various ways light can interact with it in Sec. 4.1. We then derive the analogous rendering equation for scenes containing both surfaces and volumes, and then we extend the path integral formulation (2.36) to account for volumetric media under *classical* assumptions in Sec. 4.2. There are additionally two different extensions to the classical formulation for volumetric rendering which we will review. The first is the null-scattering reformulation based on the work by Miller et al. [51] which we introduce in Sec. 4.3. Lastly, we review an extension to support non-classical participating media in Sec. 4.4, which was originally introduced by Bitterli et al. [4] and Jarabo et al. [52].

4.1	Medium properties and light interactions	35
4.2	Classical volumetric transport	39
4.2.1	Transmittance	39
4.2.2	Free-flight distribution . .	41
4.2.3	Volumetric rendering equation	42
4.2.4	Volumetric path integral .	43
4.3	Null-scattering reformulation	46
4.3.1	Null-scattering VRE	48
4.3.2	Null-scattering path integral	49
4.4	Non-classical volumetric transport	51
4.4.1	Stochastic media and ensemble averaging	52
4.4.2	Path throughput	53
4.4.3	Transport Kernel	53
4.4.4	Davis-Weinstein transmission model	55

[1]: Chandrasekhar (1960), *Radiative Transfer*

[51]: Miller et al. (2019), 'A Null-Scattering Path Integral Formulation of Light Transport'

[4]: Bitterli et al. (2018), 'A Radiative Transfer Framework for Non-Exponential Media'

[52]: Jarabo et al. (2018), 'A Radiative Transfer Framework for Spatially-Correlated Materials'

4.1 Medium properties and light interactions

Similarly to how we decomposed the idea of a surface into its geometric shape and BSDF dictated appearance in Sec. 2.3.1, we apply an analogous decomposition for participating media. A participating medium is defined by its *extinction*, *albedo*, and its *phase function*, where the extinction dictates the medium's overall shape while the albedo and phase function dictate its appearance.

Extinction. We assume that all volumes are comprised of an enormous amount of microscopic particles, and similar to surfaces, it would be infeasible to model every particle comprising a volume in an explicit manner. Instead, we choose to represent the shape of a volume *probabilistically* by using its *extinction coefficient* $\mu_r(\mathbf{x})$ to dictate the *likelihood* that light will interact with the medium at some spatial location $\mathbf{x} \in \mathbb{R}^3$. Physically, the extinction coefficient is defined as the product, $\mu_r(\mathbf{x}) := n(\mathbf{x}) \cdot \phi$, of the cross-sectional area ϕ of the medium's particles and the particle density $n(\mathbf{x})$, or average number of particles per unit volume. Under the classical assumptions of radiative transfer [1], the probability that an interaction will occur between a beam of light and a volumetric particle along an infinitesimal distance Δt is exactly equal to $\Delta t \cdot \mu_r(\mathbf{x})$.

For a participating medium to be physically based, the extinction coefficient has to always be non-negative. All participating media employed within this dissertation are physically-based, thus, $\mu_r(\mathbf{x}) \geq 0$ for all positions $\mathbf{x} \in \mathbb{R}^3$. We use the subscript "*r*" to indicate that we are referring to the *real* extinction which will become more intuitive in Sec. 4.3 when we introduce the concept of the *null* extinction $\mu_n(\mathbf{x})$.

[1]: Chandrasekhar (1960), *Radiative Transfer*

Optical Depth. Since we assume that light always travels in straight lines, we often have to take into account the accumulation of the extinction over some line,

$$\tau_r(\mathbf{x}, d) := \int_0^d \mu_r(\mathbf{x}_t) dt, \quad (4.1)$$

along some distance d through the medium. We refer to τ_r as the *real optical depth* since μ_r is the *real* extinction.

Medium albedo. The visual appearance of a volumetric medium is determined by how much light it scatters, where light scatters, and how much light gets emitted if the volume is emissive. We assume every individual particle within a medium is its own *scatterer* which can interact with incoming light. The scatterer can either absorb or reflect light in some other direction. The relative likelihoods of light being absorbed or scattered are dictated by the *absorption coefficient* $\mu_a(\mathbf{x})$ and the *scattering coefficient* $\mu_s(\mathbf{x})$, respectively. The combined sum of these coefficients $\mu_r(\mathbf{x}) := \mu_a(\mathbf{x}) + \mu_s(\mathbf{x})$ is equal to the real extinction coefficient, and their ratios $\alpha_a(\mathbf{x}) := \mu_a(\mathbf{x})/\mu_r(\mathbf{x})$ and $\alpha_s(\mathbf{x}) := \mu_s(\mathbf{x})/\mu_r(\mathbf{x})$ are referred to as the *absorption albedo*

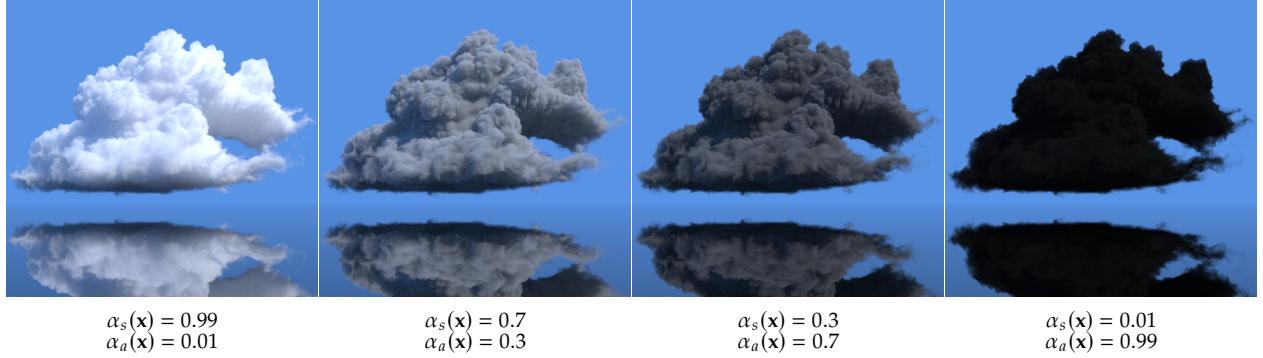


Figure 4.1: Visualizations of the same volumetric medium rendered using a variety of different albedos.

$\alpha_a(\mathbf{x})$ and *scattering albedo* $\alpha_s(\mathbf{x})$, respectively. While the real extinction dictates the likelihood of light interacting with the medium, the albedos dictate which type of interaction occurs. We visualize the effect different configurations of albedos have on the appearance of a medium in Fig. 4.1.

In practice, when we generally refer to the albedo of a medium it is implied that we are referring to the scattering albedo $\alpha_s(\mathbf{x})$. Additionally, the albedos of a medium are always specified with respect to the *total* extinction of the medium so that they will sum to unity. For now, the total extinction is analogous to the real extinction, but when we introduce the null-scattering formulation in Sec. 4.3 this will no longer be the case. The albedos will then be redefined, however, since it will become contextually obvious when we are referring to the null-scattering formulation we will continue to use the same notation for both quantities.

Medium representations. There are a variety of options for specifying $\mu_r(\mathbf{x})$ for all points in space. For brevity, we will only enumerate two of them which occur frequently within production and scientific contexts. One of the most common ways of specifying $\mu_r(\mathbf{x})$ is to explicitly store $\mu_r(\mathbf{x})$ within a *voxel density grid* (typically implemented using OpenVDB [53] or Field3D [54]) where each voxel stores a density value. Voxel density grids allow for fairly fast extinction queries, simple spatial interpolation, and make it convenient for constructing acceleration structures before rendering since all density values are iterable within a grid. However, the visual fidelity of a voxel density grid is limited by its spatial resolution where adding more detail translates into significant increases in storage.

Alternatively, the extinction coefficient $\mu_r(\mathbf{x})$ can be specified *procedurally* or as some combination of procedural functions applied to voxel density grids. While this can result in a near-infinite amount of visual fidelity and requires less storage space than only using a dense voxel grid, it turns the density into a black box function, making it more difficult to prepare acceleration structures before rendering. This causes many problems with state-of-the-art methods, but as a major contribution of this dissertation, we will alleviate this drawback in Chapter 7. We provide an exaggerated visualization of the difference between a voxel density grid and a procedural representation of the same medium in Fig. 4.2.

[53]: Museth (2013), ‘VDB: High-Resolution Sparse Volumes with Dynamic Topology’

[54]: Wrenninge et al. (2010), ‘Volumetric Methods in Visual Effects’

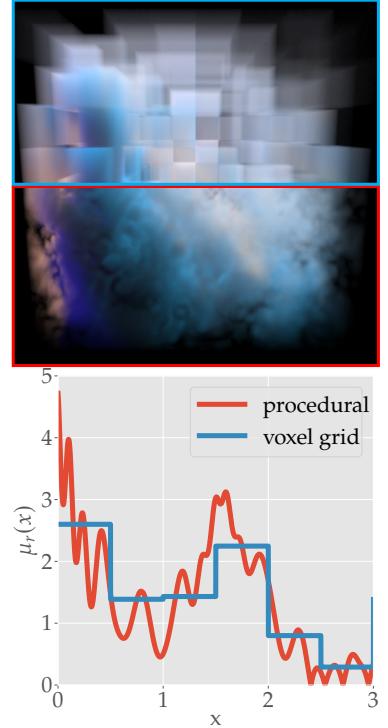


Figure 4.2: We visualize an exaggeration of two different volume representations for the same medium: purely procedural functions (red) and baked voxel density grids (blue). In the bottom graph we conceptualize what the real extinction might look like along a line through the medium for both representations.

Phase function. When light interacts with a surface, the BSDF of the surface dictates the angular distribution of directions that light will scatter after the interaction. The volumetric analog to the BSDF is the *phase function* $\rho_p(x, \vec{\omega}_i, \vec{\omega}_o)$, which represents the angular distribution that light will scatter after a medium interaction at location x . For notational convenience we refer to the phase function using $\rho_p(x, \vec{\omega}_i, \vec{\omega}_o)$, however, all phase functions employed within this dissertation rely only on the angle, $\theta := \cos^{-1}(|\vec{\omega}_i \cdot \vec{\omega}_o|)$, between the incoming and outgoing directions.

Similarly to the BSDF, for a phase function to be physically-based it has to maintain the following properties:

- **Non-negativity.** Scattering within a medium cannot lead to the introduction or reflection of "negative light",

$$\rho_p(x, \vec{\omega}_i, \vec{\omega}_o) \geq 0; \quad \forall (x, \vec{\omega}_i, \vec{\omega}_o). \quad (4.2)$$

- **Normalization.** We assume that all phase functions are normalized distributions over the sphere of directions,

$$\int_{S^2} \rho_p(x, \vec{\omega}_i, \vec{\omega}_o) d\Omega(\vec{\omega}_i) = 1. \quad (4.3)$$

Unlike the BSDF, which is cosine-weighted and not guaranteed to be normalized (2.15), the phase function is defined to be a valid probability density function by itself.

- **Reciprocity.** Just like the BSDF, Helmholtz's law of reciprocity [17] holds for the phase function,

$$\rho_p(x, \vec{\omega}_i, \vec{\omega}_o) = \rho_p(x, \vec{\omega}_o, \vec{\omega}_i). \quad (4.4)$$

Some common phase functions are the Henyey-Greenstein phase function [55], the Schlick approximation to the Henyey-Greenstein phase function [56], Rayleigh scattering [57], or the family of phase functions which can be derived from Lorenz-Mie theory [58, 59]. All volumes in this dissertation are rendered using the Henyey-Greenstein phase function, so we will only review that one in depth. We refer the interested reader to Eugene d'Eon's fantastic book [60] which discusses many other phase functions in depth and how to use them in practice.

Henyey-Greenstein phase function. The Henyey-Greenstein phase function was originally derived to model the scattering of light due to intergalactic dust [55], but it has since been employed within computer graphics to model the scattering of light in clouds, skin, and through the ocean. This phase function is non-isotropic and is defined as,

$$\rho_p(x, \vec{\omega}_i, \vec{\omega}_o) := \rho_p(x, |\vec{\omega}_i \cdot \vec{\omega}_o|) = \rho_p(x, \theta) = \frac{1 - g^2}{4\pi (1 + g^2 - 2g \cos \theta)^{3/2}}, \quad (4.5)$$

where $\theta := \cos^{-1}(|\vec{\omega}_i \cdot \vec{\omega}_o|)$ is the angle between the incident and outgoing directions and $g \in [-1, 1]$ is a user-specified parameter that measures the

[17]: von Helmholtz et al. (1924), *Helmholtz's Treatise on Physiological Optics*

[55]: Henyey et al. (1941), 'Diffuse Radiation in the Galaxy'

[56]: Blasi et al. (1993), 'A Rendering Algorithm for Discrete Volume Density Objects'

[57]: Rayleigh (1871), 'On the Scattering of Light by Small Particles'

[58]: Lorenz (1890), 'Lysbevaegelsen i og uden for en af plane Lysbo lger belyst Kugle'

[59]: Mie (1908), 'Beiträge zur Optik trüber Medien, speziell kolloidaler Metallösungen'

[60]: d'Eon (2016), *A Hitchhiker's Guide to Multiple Scattering*

[55]: Henyey et al. (1941), 'Diffuse Radiation in the Galaxy'

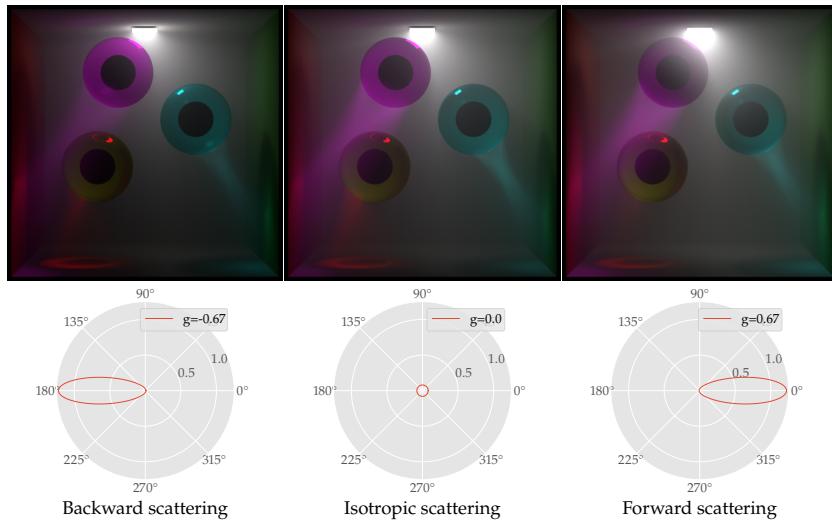


Figure 4.3: A comparison of the difference between using a Henyey-Greenstein phase function with backward ($g = -0.67$), isotropic ($g = 0.0$), and forward scattering ($g = 0.67$), within a heterogeneous medium containing volumetric caustics.

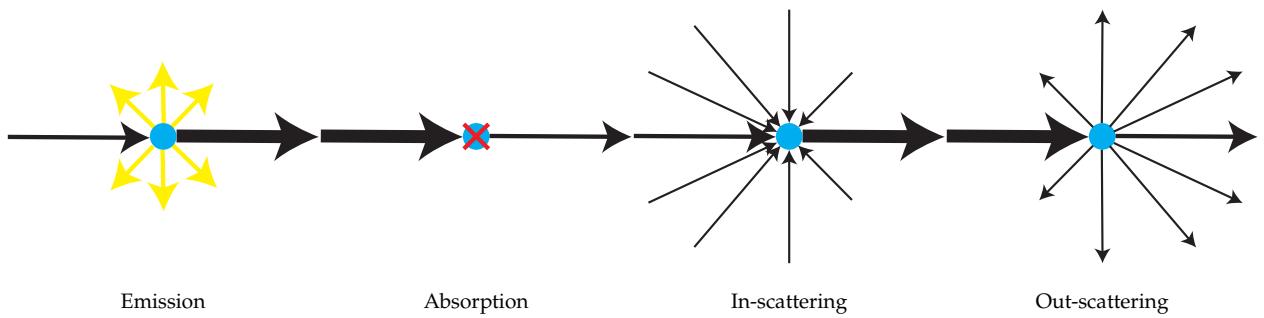


Figure 4.4: We illustrate the four different interactions which can occur between light and a participating medium. From left to right, the amount of light can increase due to volumetric emission, decrease due to volumetric absorption, increase due to volumetric in-scattering, and decrease due to volumetric out-scattering.

relative strength of forward and backward scattering. The user-specified parameter g determines the average cosine of all scattered directions with $g = 0$ being purely isotropic scattering, and $g = 1$ or $g = -1$ resulting in fully forward or backward scattering, respectively. We visualize different settings for g within a homogeneous medium in Fig. 4.3.

Light interaction events. Consider an infinitesimal beam of light traveling straight through some participating medium. At any point in space there might be a non-zero chance of four different interactions occurring (Fig. 4.4), resulting in a modulation of the amount of radiance traveling through the beam. Those four interactions are defined as follows:

- ▶ **Emission.** Light can be *emitted* into the infinitesimal beam by the medium itself.
- ▶ **Absorption.** Light can be *absorbed* by directly interacting with the particles in the medium.
- ▶ **Out-scattering.** Instead of being absorbed when interacting with the particles in the medium, light can instead *scatter off* of those particles *out* of the beam.
- ▶ **In-scattering.** Light can also scatter *into* the beam from other directions.

Simulating the transport of light through a participating medium simplifies to computing the change in radiance, $(\vec{\omega} \cdot \nabla)L(\mathbf{x}, \vec{\omega})$, due to the four aforementioned interaction events. To do so, we first need to decide on a model which describes how light *attenuates* through the medium.

In the next section, we will derive a volumetric analog for the directional form of the surface rendering equation (2.23) and path integral formulations (2.36) under the classical assumption that light attenuates *exponentially* through a volume. Later in Sec. 4.4, we will expand on this theory by reviewing a generalized formulation for supporting non-classical attenuation.

4.2 Classical volumetric transport

Classical volumetric light transport is derived directly from the radiative transfer equation (RTE) [1] which makes the assumption that all particles, or scatterers, within a medium are *statistically independent* of one another. Under this assumption, quantifying the change in radiance becomes,

$$\begin{aligned} (\vec{\omega} \cdot \nabla)L(\mathbf{x}, \vec{\omega}) = & -\mu_a(\mathbf{x})L(\mathbf{x}, \vec{\omega}) \quad (\text{absorption}) \\ & + \mu_a(\mathbf{x})L_e(\mathbf{x}, \vec{\omega}) \quad (\text{emission}) \\ & - \mu_s(\mathbf{x})L(\mathbf{x}, \vec{\omega}) \quad (\text{out-scattering}) \\ & + \mu_s(\mathbf{x})L_m(\mathbf{x}, \vec{\omega}) \quad (\text{in-scattering}), \end{aligned} \quad (4.6)$$

[1]: Chandrasekhar (1960), *Radiative Transfer*

which is known as the integro-differential form of the RTE. We use the subscript *m* in L_m to denote the *in-scattered medium radiance* which we will define later (see: Eq. (4.20)). Since both out-scattering and absorption share a common radiance term, $L(\mathbf{x}, \vec{\omega})$, we can simplify Eq. (4.6) by combining them into a single real extinction term $\mu_r(\mathbf{x}) = \mu_a(\mathbf{x}) + \mu_s(\mathbf{x})$, resulting in,

$$\begin{aligned} (\vec{\omega} \cdot \nabla)L(\mathbf{x}, \vec{\omega}) = & +\mu_a(\mathbf{x})L_e(\mathbf{x}, \vec{\omega}) \quad (\text{emission}) \\ & + \mu_s(\mathbf{x})L_m(\mathbf{x}, \vec{\omega}) \quad (\text{in-scattering}) \\ & - \mu_r(\mathbf{x})L(\mathbf{x}, \vec{\omega}) \quad (\text{extinction}). \end{aligned} \quad (4.7)$$

4.2.1 Transmittance

The differential equation for the extinction,

$$(\vec{\omega} \cdot \nabla_r)L(\mathbf{x}, \vec{\omega}) = -\mu_r(\mathbf{x})L(\mathbf{x}, \vec{\omega}), \quad (4.8)$$

can be solved in isolation to arrive at the equation for *classical transmittance*,

$$T_r(\mathbf{x}, \vec{\omega}, d) := e^{-\int_0^d \mu_r(\mathbf{x}_t) dt}, \quad (4.9)$$

which computes the proportion of light passing between \mathbf{x} and $\mathbf{y} := \mathbf{x} + d\vec{\omega}$ unobstructed where d is the distance from \mathbf{x} to \mathbf{y} . The subscript "r" in T_r indicates that the transmittance is computed using the *real* extinction,

$\mu_r(\mathbf{x})$. Other forms of Eq. (4.9) employing different extinction functions will appear later when we introduce the null-scattering formulation in Sec. 4.3. We will differentiate them by employing different subscripts.

While we specifically focus on classical transport within this section, there are a variety of properties and notations which are shared between both non-classical and classical transmittance functions. For this reason, we will review the general properties of transmittance in the remainder of this subsection.

Notation. Transmittance appears frequently within volumetric rendering, so it becomes convenient to simplify its parameterization or adapt it depending on the context. For brevity, we will drop the distance dependencies from all transmittance evaluations $T_r(\mathbf{x}, \vec{\omega}, d) = T_r(\mathbf{x}, \vec{\omega})$ and assume that they are implied to be returned by the ray tracing operator. Transmittance between two known points (\mathbf{x}, \mathbf{y}) can be denoted $T_r(\mathbf{x}, \mathbf{y}) = T_r(\mathbf{x}, \vec{\mathbf{y}})$, and the transmittance between two points $(\mathbf{x}_a, \mathbf{x}_b)$ a distance "a" and "b" along an implied ray will be denoted $T_r(\mathbf{x}_a, \mathbf{x}_b)$ for convenience. If $a = 0$, we will occasionally choose to drop the subscript, thus $T_r(\mathbf{x}_0, \mathbf{x}_b) = T_r(\mathbf{x}, \mathbf{x}_b)$.

Transmittance is a CCDF. Regardless of whether transmittance is represented as Eq. (4.9) or as one of the non-classical models (see: Sec. 4.4), transmittance is always defined to be,

$$T_r(\mathbf{x}, \vec{\omega}) := 1 - \int_0^d \rho_f^{(r)}(\mathbf{x}_t) dt = \int_d^\infty \rho_f^{(r)}(\mathbf{x}_t) dt, \quad (4.10)$$

the *complementary cumulative distribution function* (CCDF)¹ dictating the distance light can travel unobstructed. The corresponding PDF, $\rho_f(\mathbf{x}_t)$, is known as the *free-flight* distribution which we will review in Sec. 4.2.2.

1: We provide an overview of the various distribution functions and how they interrelate in Sec. 3.1.2

Physically-based transmittance. For a transmittance function to be physically-based, it has to maintain three properties.

- **Reciprocity.** A physically-based transmittance function should always be reciprocal when considered in isolation, thus,

$$T_r(\mathbf{x}, \mathbf{y}) = T_r(\mathbf{y}, \mathbf{x}). \quad (4.11)$$

While non-reciprocal transmittance models have previously been employed with success in artistic applications and volumetric scene reconstruction applications [61], we will only consider reciprocal models within this dissertation.

- **Non-increasing.** A physically-based transmittance function should never increase,

$$T_r(\mathbf{x}_a, \mathbf{x}_t) \geq T_r(\mathbf{x}_a, \mathbf{x}_d) \text{ if } t \leq d, \quad (4.12)$$

[61]: Vicini et al. (2021), 'A Non-Exponential Transmittance Model for Volumetric Scene Representations'

as the distance d increases. This property is also covered by the requirement that transmittance is a valid CCDF (4.10), since a CCDF is a non-increasing function.

- **Non-negative.** A physically-based transmittance function should never be negative, thus,

$$T_r(x_a, x_b) \geq 0. \quad (4.13)$$

This is also another property that is covered by the requirement that transmittance is a valid CCDF (4.10), since the respective PDF has to be non-negative to be a valid PDF (see: Sec. 3.1.2). While transmittance should never be negative, estimates of transmittance *can* be negative. Later in Chapter 7 we will analyze situations where this can become problematic to the performance of volumetric rendering algorithms.

4.2.2 Free-flight distribution

The probability distribution function (PDF) coinciding with the transmittance can be generally derived from Eq. (4.10) as,

$$\rho_f^{(r)}(x_t) := -\frac{d}{dt} T(x, x_t), \quad (4.14)$$

where it represents the probability density that light travels a distance t before the next interaction event. For the specific case of classical exponential transmittance (4.9), the PDF is defined as,

$$\rho_f^{(r)}(x_t) := \mu_r(x_t) T_r(x, x_t), \quad (4.15)$$

where $\rho_f(x_t)$ is referred to as the *free-flight* PDF since $\rho_f(x_t)$ represents the probability density that a ray travels to a point x_t "free" of interacting with the medium. We visualize the transmittance and free-flight distributions for some simple extinction functions in Fig. 4.5.

Notation. Although $\rho_f^{(r)}(x_t)$ is a PDF, we choose to refer to it using " ρ " instead of " p " because the free-flight PDF of a medium, similar to a BSDF, is a physical characteristic of the medium itself and we want to distinguish it from any future application of Monte Carlo importance sampling. We additionally employ the superscript " r " to denote $\rho_f^{(r)}(x_t)$ as the *real* free-flight distribution which will become more intuitive in Sec. 4.3.

Mean free-path. The mean of the free-flight distribution is referred to as the *mean-free path*, and it represents the average distance a ray is expected to travel through a medium unobstructed. For the classical exponential model (4.15), the mean-free path in a homogeneous medium ($\mu_r(x) = \mu_r$) is equal to $1/\mu_r$.

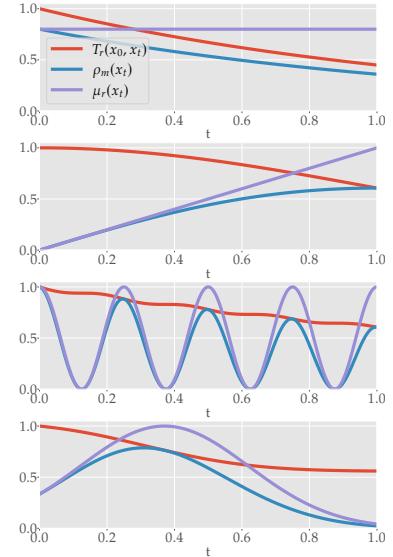


Figure 4.5: Visualization of the shape of classical transmittance and free-flight distributions for a variety of simple extinction functions. From top to bottom, we plot a constant, linearly increasing, cosine, and Gaussian extinction functions with their respective distribution functions.

4.2.3 Volumetric rendering equation

By integrating both sides of Eq. (4.7) while using the surface only rendering equation (2.22) as a boundary condition, we can derive an integral formulation,

$$\begin{aligned} L_o(\mathbf{x}, \vec{\omega}_o) &:= \int_0^d \rho_f^{(r)}(\mathbf{x}_t) \alpha_a(\mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) dt \quad (\text{accumulated emission}) \\ &\quad + \int_0^d \rho_f^{(r)}(\mathbf{x}_t) \alpha_s(\mathbf{x}_t) L_m(\mathbf{x}_t, \vec{\omega}_o) dt \quad (\text{in-scattered radiance}) \\ &\quad + T_r(\mathbf{x}, \mathbf{x}_d) L_s(\mathbf{x}_d, \vec{\omega}_o) \quad (\text{next surface radiance}), \end{aligned} \quad (4.16)$$

which takes the form of the sum of three different quantities. Assuming we have obtained the distance d to the next surface using the ray-tracing operator, the first quantity in the sum is the accumulated emission through the medium to the next surface interaction. The second integral accumulates the in-scattered *medium* radiance along the ray to the next surface interaction, while the third term accumulates the transmittance-modulated radiance from the next surface-scattering event. Surface scattering interactions are still computed using the surface only rendering equation (2.22), except the recursive call in that formulation,

$$L_s(\mathbf{x}, \vec{\omega}_o) := L_e(\mathbf{x}, \vec{\omega}_o) + \int_{\mathbb{H}} \rho_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) L_i(\mathbf{x}, \vec{\omega}_i) |\cos \theta| d\Omega(\vec{\omega}_i), \quad (4.17)$$

now becomes Eq. (4.16).

We integrate the accumulated emission and in-scattered radiance multiplied by the free-flight distribution of the medium because that makes the most intuitive sense regarding the underlying physics. Integrating over a free-flight distribution, which is effectively a PDF multiplied by either emissive or in-scattered radiance, is equivalent to mathematically computing the expected value of the radiance². However, this is not the only way of writing the VRE. Later it will become more convenient to write everything in terms of transmittance,

$$\begin{aligned} L_o(\mathbf{x}, \vec{\omega}_o) &:= \int_0^d T_r(\mathbf{x}, \mathbf{x}_t) \mu_a(\mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) dt \\ &\quad + \int_0^d T_r(\mathbf{x}, \mathbf{x}_t) \mu_s(\mathbf{x}_t) L_m(\mathbf{x}_t, \vec{\omega}_o) dt \\ &\quad + T_r(\mathbf{x}, \mathbf{x}_d) L_s(\mathbf{x}_d, \vec{\omega}_o), \end{aligned} \quad (4.18)$$

instead of free-flight distributions. Note, in this form the albedos are now replaced by their respective extinctions.

2: See Sec. 3.1.3 for more of a discussion on expected value.

Transport. As briefly mentioned in Sec. 4.2.1, light passing through a participating medium gets modulated by the transmittance, so the transport equation we introduced in Eq. (2.18) is no longer applicable. The transport

equation through a classical medium becomes,

$$L_i(\mathbf{x}, \vec{\omega}) := T(\mathbf{x}, r(\mathbf{x}, \vec{\omega})) L_o(r(\mathbf{x}, \vec{\omega}), -\vec{\omega}), \quad (4.19)$$

which is also equivalent to using Eq. (4.16) to compute radiance passing through a fully absorptive ($\alpha_s = 0$), non-emissive ($L_e = 0$) medium.

Measurement. The measurement equation (2.19) remains unchanged since the introduction of participating media only changes the definition of $L_i(\mathbf{x}, \vec{\omega})$ and does not affect how we integrate the product of importance multiplied by radiance.

Scattering. The *in-scattered medium radiance* is computed by evaluating the integral,

$$L_m(\mathbf{x}, \vec{\omega}_o) := \int_{\mathbb{S}^2} \rho_p(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) L_i(\mathbf{x}, \vec{\omega}_i) d\Omega(\vec{\omega}_i), \quad (4.20)$$

over the product of the phase function, $\rho_p(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o)$, and the incident radiance, $L_i(\mathbf{x}, \vec{\omega}_i)$, through recursively evaluating Eq. (4.16).

Volumetric rendering equation. Inserting both Eq. (4.20) and Eq. (4.17) into Eq. (4.16) gives us the full *volumetric rendering equation* (VRE) [62] which fully describes how to compute radiance within a scene containing both surfaces and participating media.

[62]: Arvo (1993), 'Transfer Functions in Global Illumination'

4.2.4 Volumetric path integral

Akin to the surface-only case, the recursive nature of the VRE naturally influences any derived solutions to make only locally good decisions. This makes applying more advanced techniques and optimizations difficult. We can alleviate this by transforming the VRE into an integral formulation which integrates over all of path space. To do so, we adapt the same process we used before in Sec. 2.4, except now we account for the possibility of vertices being located within participating media. For brevity, we will skip the conversion into Cartesian space and the expansion of the recursive call to skip directly to defining the combined volumetric and surface path integral.

Extended path space. We use the same notation as Sec. 2.4 where an n -length path \bar{x} is a path containing $n+1$ vertices $\mathbf{x}_0 \dots \mathbf{x}_n$ where \mathbf{x}_0 originates on a sensor and \mathbf{x}_n lies on an emissive source. However, each vertex can now either be a point on a surface or a scatter location within a participating medium, so the set of all n -length paths (2.30) has to be extended,

$$\mathbb{P}_n := \{\mathbf{x}_0 \dots \mathbf{x}_n \mid \mathbf{x}_0 \dots \mathbf{x}_n \in \mathbb{A} \cup \mathbb{V}\}, \quad (4.21)$$

to account for this where \mathbb{V} refers to the space of all points within participating media.

The entire path space spanning the combination of surface points and volume points is then defined as the union of all n -length extended path sets (4.21),

$$\mathbb{P} := \bigcup_{n=1}^{\infty} \mathbb{P}_n. \quad (4.22)$$

General geometric term. Since the VRE (4.16) exists in a direction-distance space as opposed to just a direction space, we need to introduce an additional Jacobian,

$$dt d\Omega(\vec{\omega}) = V(x, x_t) \frac{1}{\|x - x_t\|^2} dV(x_t), \quad (4.23)$$

for converting between distance-direction space to the space of all volumetric points. Given the difference between the surface only Jacobian (2.26) and Eq. (4.23), we introduce the quantity,

$$D(x, \vec{\omega}) := \begin{cases} 1, & x \in \mathbb{V}, \\ |\vec{n} \cdot \vec{\omega}|, & x \in \mathbb{A}, \end{cases} \quad (4.24)$$

to represent the foreshortening term. This allows us to rewrite the geometric term from the surface-only formulation (2.27) more generally as,

$$G(x, y) := V(x, y) \frac{D(x, \vec{xy}) D(y, \vec{yx})}{\|x - y\|^2}, \quad (4.25)$$

to represent both surface and medium vertices.

Path throughput. An entire path \bar{x} will consist of many vertices where only the sensor vertex will be guaranteed to exist on a surface, since we assume sensors do not exist as volumes. Performing a full change of variables into Cartesian space and then expanding out the recursions (similar to what we did in Sec. 2.4 for the surface only case) results in the general path throughput,

$$g(\bar{x}) := \left[\prod_{j=1}^{n-1} \rho_{\star}(x_j, x_j \vec{x}_{j+1}, x_j \vec{x}_{j-1}) \right] \left[\prod_{j=1}^n G(x_{j-1}, x_j) T_r(x_{j-1}, x_{j-1} \vec{x}_j) \right], \quad (4.26)$$

containing a *generalized scattering function* ρ_{\star} ,

$$\rho_{\star}(x_j, x_j \vec{x}_{j+1}, x_j \vec{x}_{j-1}) := \begin{cases} \rho_s(x_j, x_j \vec{x}_{j+1}, x_j \vec{x}_{j-1}), & x_j \in \mathbb{A} \\ \mu_s(x_j) \rho_p(x_j, x_j \vec{x}_{j+1}, x_j \vec{x}_{j-1}), & x_j \in \mathbb{V} \end{cases} \quad (4.27)$$

that is either the BSDF, ρ_s , or the phase function, ρ_p , depending on whether the vertex x_j lies on a surface or within a medium, respectively.

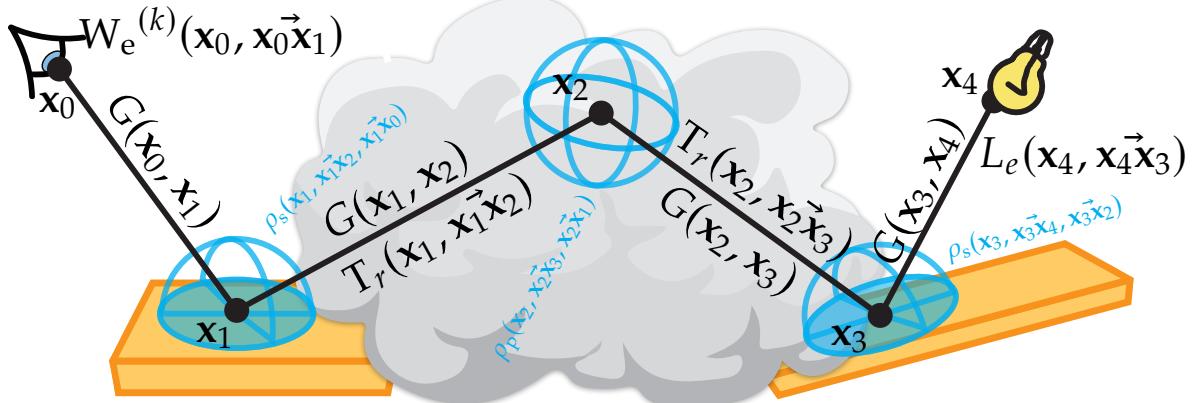


Figure 4.6: A geometrical illustration of the various terms within the path contribution $f^{(k)}(\bar{x})$ Eq. (4.28) for a path with 5 vertices ($\bar{x} = x_0 x_1 x_2 x_3 x_4$) where the path segments (x_1, x_2) and (x_2, x_3) lie within a participating medium. We assume the path segments between (x_0, x_1) and (x_3, x_4) lie completely outside the medium.

Unlike the surface-only case, the path throughput in the presence of participating media has to additionally account for the transmittance, $T_r(x_{j-1}, x_{j-1} \vec{x}_j)$, between every consecutive vertex along a path. We illustrate a full path containing both volume and surface vertices in Fig. 4.6.

Measurement contribution. The entire path contribution is computed equivalently to the surface-only case (2.35) as,

$$f^{(k)}(\bar{x}) := W_e^{(k)}(x_0, x_0 \vec{x}_1) g(\bar{x}) L_e(x_n, x_n \vec{x}_{n-1}), \quad (4.28)$$

however, both the path throughput, $g(\bar{x})$ (4.26), and the emissive term,

$$L_e(x_n, x_n \vec{x}_{n-1}) := \begin{cases} L_e(x_n, x_n \vec{x}_{n-1}), & \text{if } x_n \in \mathbb{A}, \\ \mu_a(x_n) L_e(x_n, x_n \vec{x}_{n-1}), & \text{if } x_n \in \mathbb{V}, \end{cases} \quad (4.29)$$

are modified to account for the possibility of both surface and volume vertices.

Free-flight formulation. Similarly to the different representations of the VRE (4.16 and 4.18), we can alternatively reformulate Eq. (4.26), Eq. (4.27), and Eq. (4.29) in terms of the free-flight distributions as,

$$g(\bar{x}) := \left[\prod_{j=1}^{n-1} \rho_\star(x_j, x_j \vec{x}_{j+1}, x_j \vec{x}_{j-1}) \right] \left[\prod_{j=1}^n G(x_{j-1}, x_j) \rho_f^{(r)}(x_j) \right], \quad (4.30)$$

$$\rho_\star(x_j, x_j \vec{x}_{j-1}, x_j \vec{x}_{j+1}) := \begin{cases} \frac{1}{\mu_r(x_j)} \rho_s(x_j, x_j \vec{x}_{j+1}, x_j \vec{x}_{j-1}), & x_j \in \mathbb{A} \\ \alpha_s(x_j) \rho_p(x_j, x_j \vec{x}_{j+1}, x_j \vec{x}_{j-1}), & x_j \in \mathbb{V}, \end{cases} \quad (4.31)$$

$$L_e(x_n, x_n \vec{x}_{n-1}) := \begin{cases} \frac{1}{\mu_r(x_n)} L_e(x_n, x_n \vec{x}_{n-1}), & \text{if } x_n \in \mathbb{A}, \\ \alpha_a(x_n) L_e(x_n, x_n \vec{x}_{n-1}), & \text{if } x_n \in \mathbb{V}, \end{cases} \quad (4.32)$$

respectively. Note, for the case of surface interactions the inverse extinction terms, $\frac{1}{\mu_r(\mathbf{x}_j)}$, will cancel with the extinction term within $\rho_f^{(r)}(\mathbf{x}_j)$ leaving only the real transmittance.

Path integral formulation. By integrating over all path space after expanding to include both surface and volume points,

$$I_k := \int_{\mathbb{P}} f^{(k)}(\bar{\mathbf{x}}) d\mu(\bar{\mathbf{x}}) \quad (4.33)$$

we arrive at the concise *path integral formulation* for volumetric light transport. Unlike the surface-only case, the corresponding path measure is redefined to be,

$$\begin{aligned} d\mu(\bar{\mathbf{x}}) &:= \prod_{j=0}^n d\mu(\mathbf{x}_j), \\ d\mu(\mathbf{x}_j) &:= \begin{cases} dA(\mathbf{x}_j), \mathbf{x}_j \in \mathbb{A}, \\ dV(\mathbf{x}_j), \mathbf{x}_j \in \mathbb{V}, \end{cases} \end{aligned} \quad (4.34)$$

the product of all area measures or volumetric measures, depending on the location of vertex \mathbf{x}_j .

4.3 Null-scattering reformulation

The VRE and the path integral formulations for volumetric transport provide a solid framework for evaluating the radiance through scenes containing both surfaces and participating media. However, their direct physical interpretations are slightly inconvenient for the direct application of Monte Carlo integration³.

Evaluating the in-scattered medium radiance (4.20) typically requires sampling directions in which light scatters from any intermediate point along a light path. Fortunately enough, these directions follow an intrinsic angular distribution in the form of the medium's phase function ρ_p , which is assumed to be easily importance sampled. Unfortunately, this is not the case for the medium's intrinsic free-flight distribution $\rho_f^{(r)}$ dictating the distances light can travel unobstructed.

3: We previously reviewed the basics of Monte Carlo integration in Sec. 3.2.

The free-flight distribution,

$$\rho_f^{(r)}(\mathbf{x}_t) \propto \mu_r(\mathbf{x}_t) T_r(\mathbf{x}, \mathbf{x}_t), \quad (4.35)$$

is directly dependent on both the real transmittance between two given points and the medium's real extinction. In a purely homogeneous medium where $\mu_r(\mathbf{x}_t) = \mu_r$ is a constant, the transmittance becomes analytic, and the free-flight distribution can be importance sampled analytically (see Eq. (5.24)). However, within a heterogeneous medium where $\mu_r(\mathbf{x}_t)$ remains spatially varying, it becomes impossible to directly sample from

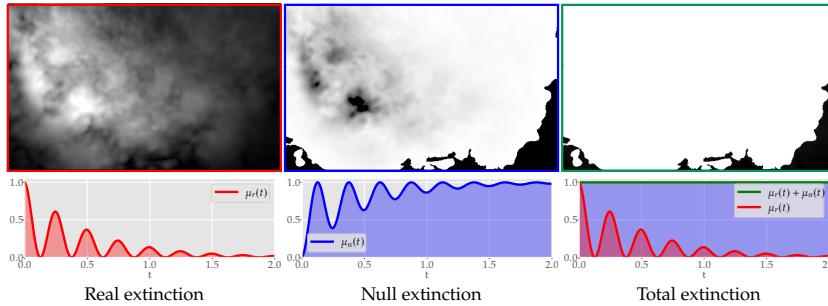


Figure 4.7: A visualization of the decomposition of the combined extinction μ_t (right) into its real $\mu_r(x)$ (left) and null $\mu_n(x)$ (middle) components.

$\rho_f^{(r)}$ directly since Eq. (4.35) is no longer analytic. This previously made deriving efficient stochastic solutions for the VRE difficult.

To get around this, Woodcock et al. [63] introduced the idea of filling all heterogeneous participating media with fictitious particles to *homogenize* the density. Fictitious particles effectively fill participating media with a corresponding null extinction, $\mu_n(x)$, that gets initialized implicitly such that the formerly heterogeneous medium's *total extinction* $\mu_t = \mu_r(x) + \mu_n(x)$ becomes constant. This makes all heterogeneous density effectively homogenized to allow for analytic free-flight sampling. We visualize the resulting combined medium in Fig. 4.7.

The null albedo, $\alpha_n(x) := \frac{\mu_n(x)}{\mu_t}$, of the resulting homogeneous medium can then be used to determine whether a potential scatter location would be a *real* collision where light would undergo one of the aforementioned medium interactions (see: Sec. 4.1), or whether a fictional *null* collision occurs, and is then ignored.

Originally, this technique was developed to be used as a random walk process to stochastically sample free-flights proportional to the medium's *real* free-flight distribution $\rho_f^{(r)}$ [63]. Very recently, Miller et al. [51] reformulated both the VRE and the volumetric path integral formulation with this *null-scattering* paradigm in mind. While the remainder of this section focuses on summarizing Miller et al.'s formulation, we will provide a more thorough review on the history of the developments between Woodcock's original conception of null-scattering and the current null-scattering reformulation of classical light transport in Chapter 5 for the context of free-flight distance sampling techniques and in Chapter 6 for the complementary problem of transmittance estimation.

Notation. From now on, we will explicitly denote whether a quantity refers to the real, total, or null portions of a medium using "r", "t", or "n", respectively, as either subscripts or superscripts. For example, the *total* free-flight distribution is,

$$\rho_f^{(t)}(\mathbf{x}_t) \propto \mu_t(\mathbf{x}_t) T_t(\mathbf{x}, \mathbf{x}_t). \quad (4.36)$$

Additionally, from now on when we refer to albedos we do so with respect to the combined extinction $\mu_t := \mu_r(x) + \mu_n(x)$. For example, the scattering albedo is now $\alpha_s(x) := \frac{\mu_s(x)}{\mu_t}$, where previously it was $\alpha_s(x) = \frac{\mu_s(x)}{\mu_r(x)}$.

[63]: Woodcock et al. (1965), 'Techniques Used in the GEM Code for Monte Carlo Neutronics Calculations in Reactors and Other Systems of Complex Geometry'

[63]: Woodcock et al. (1965), 'Techniques Used in the GEM Code for Monte Carlo Neutronics Calculations in Reactors and Other Systems of Complex Geometry'

[51]: Miller et al. (2019), 'A Null-Scattering Path Integral Formulation of Light Transport'

4.3.1 Null-scattering VRE

Given that $\mu_t = \mu_r(\mathbf{x}) + \mu_n(\mathbf{x})$, we define a null-scattering extension to the classical RTE (4.7) as,

$$\begin{aligned} (\vec{\omega} \cdot \nabla)L(\mathbf{x}, \vec{\omega}) &= \mu_a(\mathbf{x})L_e(\mathbf{x}, \vec{\omega}) && \text{(emission)} \\ &+ \mu_s(\mathbf{x})L_m(\mathbf{x}, \vec{\omega}) && \text{(in-scattering)} \\ &+ \mu_n(\mathbf{x})L(\mathbf{x}, \vec{\omega}) && \text{(null-scattering)} \\ &- \mu_t(\mathbf{x})L(\mathbf{x}, \vec{\omega}) && \text{(extinction).} \end{aligned} \quad (4.37)$$

By integrating both sides of Eq. (4.37),

$$\begin{aligned} L_o(\mathbf{x}, \vec{\omega}_o) &:= \int_0^d \rho_f^{(t)}(\mathbf{x}_t) \alpha_a(\mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) dt && \text{(accumulated emission)} \\ &+ \int_0^d \rho_f^{(t)}(\mathbf{x}_t) \alpha_s(\mathbf{x}_t) L_m(\mathbf{x}_t, \vec{\omega}_o) dt && \text{(in-scattered radiance)} \\ &+ \int_0^d \rho_f^{(t)}(\mathbf{x}_t) \alpha_n(\mathbf{x}_t) L_o(\mathbf{x}_t, \vec{\omega}_o) dt && \text{(null-scattered radiance)} \\ &+ T_t(\mathbf{x}, \mathbf{x}_d) L_s(\mathbf{x}_d, \vec{\omega}_o) && \text{(next surface radiance),} \end{aligned} \quad (4.38)$$

we arrive at the *null-scattering volumetric rendering equation* (null-VRE)⁴. The main difference between the null-VRE and the VRE (4.16) is that every transmittance evaluation,

$$T_t(\mathbf{x}, \mathbf{x}_t) := e^{-\mu_t t}, \quad (4.39)$$

and free-flight distribution (see Eq. (4.36)), in the null-VRE are *analytic* in return for incorporating an additional recursive call to handle null-scattering. The process of null-scattering can be thought of as always scattering in the *forward direction*.

As we briefly mentioned for the case of the VRE, Eq. (4.38) also takes the form of computing the expected radiance⁵. The PDF takes the form of $\rho_f^{(t)}$ while the albedos dictate the relative contributions that emission, scattering, and null-scattering have on the radiance for any point within the medium.

4: As a reminder, the albedos for scattering and absorption have been redefined to $\alpha_s(\mathbf{x}) = \mu_s(\mathbf{x})/\mu_t$ and $\alpha_a(\mathbf{x}) = \mu_a(\mathbf{x})/\mu_t$, such that $\alpha_a(\mathbf{x}_t) + \alpha_s(\mathbf{x}_t) + \alpha_n(\mathbf{x}_t) = 1$.

5: see Sec. 3.1.3 for the definition of the expectation.

Transmittance form. We can rewrite Eq. (4.38) in terms of transmittance evaluations instead of free-flight distributions as,

$$\begin{aligned}
 L_o(\mathbf{x}, \vec{\omega}_o) &:= \int_0^d \mu_a(\mathbf{x}_t) T_t(\mathbf{x}, \mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) dt \quad (\text{accumulated emission}) \\
 &\quad + \int_0^d \mu_s(\mathbf{x}_t) T_t(\mathbf{x}, \mathbf{x}_t) L_m(\mathbf{x}_t, \vec{\omega}_o) dt \quad (\text{in-scattered radiance}) \\
 &\quad + \int_0^d \mu_n(\mathbf{x}_t) T_t(\mathbf{x}, \mathbf{x}_t) L_o(\mathbf{x}_t, \vec{\omega}_o) dt \quad (\text{null-scattered radiance}) \\
 &\quad + T_t(\mathbf{x}, \mathbf{x}_d) L_s(\mathbf{x}_d, \vec{\omega}_o) \quad (\text{next surface radiance}).
 \end{aligned} \tag{4.40}$$

4.3.2 Null-scattering path integral

Similar to prior formulations, the null-VRE is restricted to purely unidirectional sampling, so we can reformulate it as a path integral to support more globally based algorithms. We briefly summarize the work by Miller et al. [51] and refer the interested reader to their work for the full derivation. For brevity, and to not rederive a similar formulation three separate times, we will only summarize the final formulation here.

[51]: Miller et al. (2019), 'A Null-Scattering Path Integral Formulation of Light Transport'

Path integral. We define the null-scattering path integral as,

$$I_k := \int_{\mathbb{P}} f^{(k)}(\bar{\mathbf{x}}) d\mu(\bar{\mathbf{x}}), \tag{4.41}$$

where \mathbb{P} is now redefined to be the *combined* path space,

$$\mathbb{P} := \bigcup_{n=1}^{\infty} (A \cup V \cup V_{\delta})^{n+1}, \tag{4.42}$$

including V_{δ} , the space of all null volume points, where the path measure, $d\mu(\bar{\mathbf{x}})$, becomes the product,

$$d\mu(\bar{\mathbf{x}}) := \prod_{i=0}^{\infty} d\mathbf{x}_i, \quad d\mathbf{x}_i := \begin{cases} dA, & \text{if } \mathbf{x}_i \in A, \\ dV, & \text{if } \mathbf{x}_i \in V, \\ dV_{\delta}, & \text{if } \mathbf{x}_i \in V_{\delta}, \end{cases} \tag{4.43}$$

of the individual measures for all surface vertices, volume vertices, and all null-scattering vertices. All null-scattering vertices are measured along the line between two real scattering vertices,

$$dV_{\delta}(\mathbf{x}_i) := d\delta_{\mathbf{x}_i^{r-} \leftrightarrow \mathbf{x}_i^{r+}}(\mathbf{x}_i), \tag{4.44}$$

where $\delta_{\mathbf{x}_i^{r-} \leftrightarrow \mathbf{x}_i^{r+}}(\mathbf{x}_i)$ is a Dirac measure guaranteeing that integration gets restricted along the lines between real scattering events. The superscripts $r-$ and $r+$ represent the indices of the previous and next real scattering events, respectively. The path length n then becomes the *total* number of scattering interactions including all null interactions.

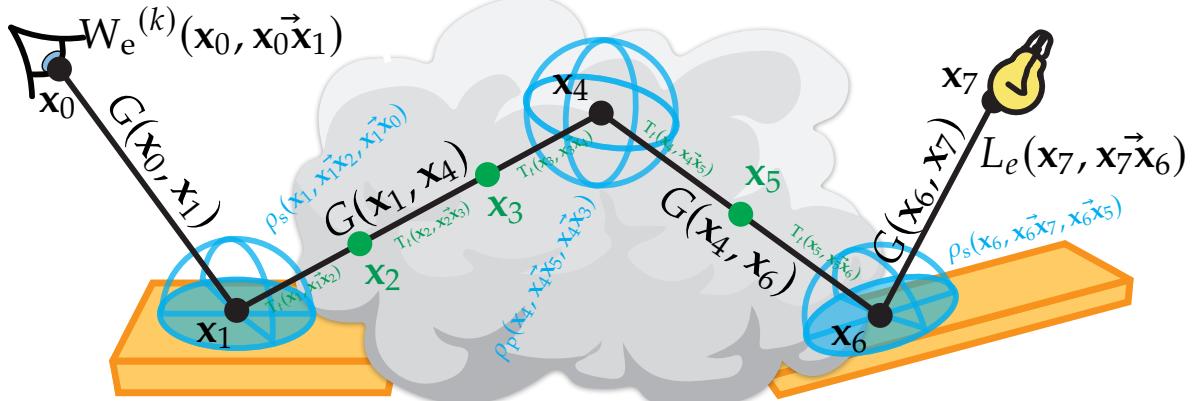


Figure 4.8: A geometrical illustration of the various terms within the combined null-scattering path contribution $f^{(k)}(\bar{x})$ (Eq. (4.41)) for a path with 5 real vertices and 3 null vertices ($\bar{x} = x_0 x_1 x_2 x_3 x_4 x_5 x_6 x_7$) where the path segments between (x_1, x_4) and (x_4, x_6) lie within a participating medium. We assume the path segments between (x_0, x_1) and (x_6, x_7) lie completely outside the medium.

Path throughput. For a given path, \bar{x} , the path throughput is computed as,

$$g(\bar{x}) := \left(\prod_{j=0}^{R-1} G(x_{r_j}, x_{r_{j+1}}) \right) \cdot \left(\prod_{j=0}^{n-1} T_t(x_j, x_j \vec{x}_{j+1}) \right) \cdot \left(\prod_{j=1}^{n-1} \rho_\star(x_j, x_j \vec{x}_{j+1}, x_j \vec{x}_{j-1}) \right), \quad (4.45)$$

where the sequence of vertices $x_{r_0} x_{r_1} \dots x_{r_R}$ refers to the sequence of all *real* scattering vertices ordered from the sensor to the emitter. Note, since the points on the sensor and emitter are always going to be considered real vertices, $R \geq 2$.

The geometric term G is defined between every pair of *real* scattering events and is equivalent to how it was defined in the previous section (4.25). The generalized scattering distribution, ρ_\star , now becomes,

$$\rho_\star(x_j, x_j \vec{x}_{j+1}, x_j \vec{x}_{j-1}) := \begin{cases} \rho_s(x_j, x_j \vec{x}_{j+1}, x_j \vec{x}_{j-1}), & \text{if } x_j \in A \\ \mu_s(x_j) \rho_p(x_j, x_j \vec{x}_{j+1}, x_j \vec{x}_{j-1}), & \text{if } x_j \in V \\ \mu_n(x_j) H(x_j \vec{x}_{j+1} \cdot x_j \vec{x}_{j-1}), & \text{if } x \in V_\delta, \end{cases} \quad (4.46)$$

where $H(x_j \vec{x}_{j+1} \cdot x_j \vec{x}_{j-1})$ is the heaviside function enforcing the order of the null vertices.

Measurement contribution. The measurement contribution is then computed as,

$$f(\bar{x}) := W_e(x_0, x_0 \vec{x}_1) g(\bar{x}) L_e(x_n, x_n \vec{x}_{n-1}), \quad (4.47)$$

where the emission is defined as,

$$L_e(x_n, x_n \vec{x}_{n-1}) := \begin{cases} L_e(x_n, x_n \vec{x}_{n-1}), & \text{if } x_n \in A, \\ \mu_a(x_n) L_e(x_n, x_n \vec{x}_{n-1}), & \text{if } x_n \in V, \end{cases} \quad (4.48)$$

which is the same as the prior path integral formulation. We illustrate an entire path containing surface, volume, and null-scattering vertices in Fig. 4.8.

Free-flight formulation. We can alternatively reformulate Eq. (4.45), in terms of the free-flight distributions instead of transmittances as,

$$g(\vec{x}) := \left(\prod_{j=0}^{R-1} G(\mathbf{x}_{r_j}, \mathbf{x}_{r_{j+1}}) \right) \cdot \left(\prod_{j=0}^{n-1} \frac{1}{\mu_t} \rho_f^{(t)}(\mathbf{x}_j) \right) \cdot \left(\prod_{j=1}^{n-1} \rho_\star(\mathbf{x}_j, \mathbf{x}_j \vec{x}_{j+1}, \mathbf{x}_j \vec{x}_{j-1}) \right), \quad (4.49)$$

respectively. This completes the null path-integral formulation for classical media.

4.4 Non-classical volumetric transport

Up until now, we have only discussed the case of rendering volumetric media under the classical assumption that all scattering particles within a volume are uncorrelated. This implies that the particles are always distributed independently from one another. When light interacts with independently distributed particles, it causes light to undergo a *memoryless* effect where every time light scatters, it does so independently of how it has scattered during previous medium interactions.

In the real world, there exists many substances which exhibit inter-particle correlations where the classical assumptions no longer apply. This occurs in substances such as milk, gels, paint, and clouds. For example, correlations have been shown to exist on the scale of kilometers within clouds [2, 64] where extensive particle clumping has been observed. These correlations have also been widely reported across the literature [65–71] in many different scientific fields.

When significant correlations arise on the scale of the mean free-path of the medium, non-exponential free-flight distributions arise [72]. Unfortunately, supporting non-exponential transport is not as trivial as simply replacing the definition of T_r . By naively replacing T_r with a non-exponential model, the classical formulations for volumetric transport are no longer energy conserving and are thus no longer physically-based [4].

Recently, Bitterli et al. [4] introduced a radiative transfer framework for computing light transport through non-exponential media which we will briefly summarize in the remainder of this section. This radiative transfer framework theoretically allows for supporting participating media with either longer-tail or shorter-tail attenuation compared to classical exponential transmittance which coincides with positively correlated and negatively correlated particles, respectively (see: Fig. 4.9).

Unfortunately, Bitterli et al. did not provide a general technique for deriving unbiased estimators for non-exponential media so most current solutions

[2]: Davis et al. (1999), ‘Horizontal Structure of Marine Boundary Layer Clouds from Centimeter to Kilometer Scales’

[64]: Kostinski et al. (2000), ‘On the Spatial Distribution of Cloud Particles’

[65]: Beresford-Smith et al. (1985), ‘The Electrostatic Interaction in Colloidal Systems with Low Added Electrolyte’

[66]: Gama Goicochea (2013), ‘A Model for the Stability of a TiO₂ Dispersion’

[67]: Grier et al. (2001), ‘Interactions in Colloidal Suspensions’

[68]: Jamali (2015), ‘Rheology of Colloidal Suspensions: A Computational Study’

[69]: Rahmani (2013), ‘Micromechanics and Rheology of Hard and Soft-Sphere Colloidal Glasses’

[70]: Smith et al. (2004), ‘Dusty Plasma Correlation Function Experiment’

[71]: Xu et al. (1998), ‘Fat Particle Structure and Stability of Food Emulsions’

[72]: Davis et al. (2004), ‘Photon Propagation in Heterogeneous Optical Media with Spatial Correlations: Enhanced Mean-Free-Paths and Wider-than-Exponential Free-Path Distributions’

[4]: Bitterli et al. (2018), ‘A Radiative Transfer Framework for Non-Exponential Media’

remain biased [4, 61]. We will later introduce techniques in Chapter 6 and Chapter 8 that will alleviate this problem.

For a more thorough discussion on this theory, we refer the interested reader to Bitterli et al.'s original work [4] and the work by Larsen and Vasques [73] from the neutron transport literature which Bitterli et al. originally sought to improve upon for the needs of computer graphics.

[73]: Larsen et al. (2011), 'A Generalized Linear Boltzmann Equation for Non-Classical Particle Transport'

4.4.1 Stochastic media and ensemble averaging

At the beginning of this chapter we discussed various representations for participating media and explained that we chose to represent all media probabilistically since independently simulating all particles as discrete scatterers would be prohibitively expensive. By representing participating media probabilistically, we implicitly want to compute the expected value over all possible probabilistic realizations of the discrete particles in the medium.

If all particles were distributed independently of one another, i.e. within a *classical* medium, we would expect no probabilistic realization to be significantly different from another. On the other hand, any probabilistic realization of the particles comprising a *non-classical* participating medium are going to exhibit different "clumping" behaviors which are going to impact the transport computed anywhere within the realized medium. For an example of different clumping behaviors see Fig. 4.9.

As stated before, we are interested in computing the *average* behavior within all possible probabilistic realizations of a medium since we do not explicitly simulate the particles comprising the medium. Thus, we want to compute the *ensemble average*,

$$\langle f_\mu(\mathbf{x}) \rangle := \int_{\mathbb{R}} f_\mu(\mathbf{x}) dP_\mu, \quad (4.50)$$

where \mathbb{R} is the space of all possible medium realizations, and dP_μ measures the likelihood of a specific realization μ occurring.

Non-exponential path integral. We introduce the non-exponential path integral by enumerating how it differs from the classical path integral formulation we introduced in Sec. 4.2.4. Assuming that the source and sensor exist outside of a participating medium, the non-exponential path integral is defined as,

$$\begin{aligned} I_k &:= \int_{\mathbb{P}} \langle f(\bar{\mathbf{x}}) \rangle d\mu(\bar{\mathbf{x}}), \\ &= \int_{\mathbb{P}} W_e^{(k)}(\mathbf{x}_0, \mathbf{x}_1) \langle g(\bar{\mathbf{x}}) \rangle L_e(\mathbf{x}_n, \mathbf{x}_{n-1}) d\mu(\bar{\mathbf{x}}), \end{aligned} \quad (4.51)$$

where we want to compute the ensemble average over the path throughput $g(\bar{\mathbf{x}})$.

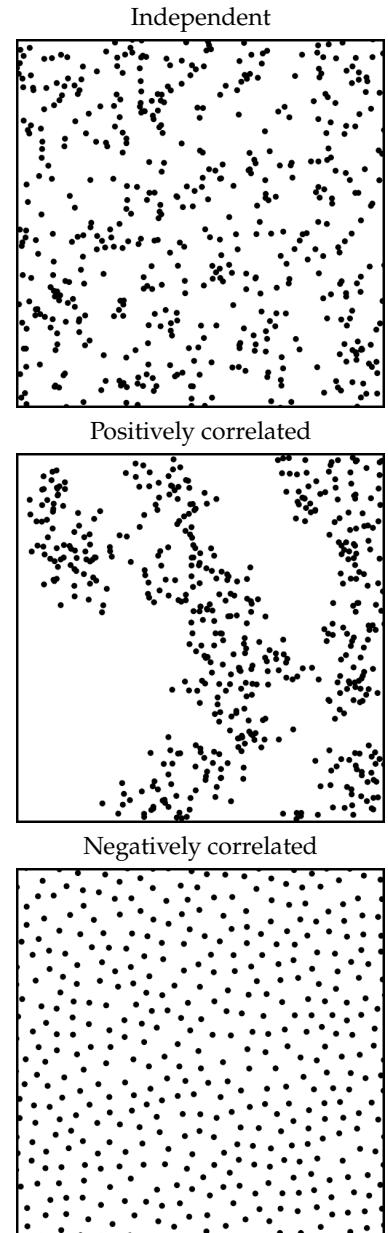


Figure 4.9: A visualization of discrete scatterers which are independently distributed (top), positively correlated (middle), and negatively correlated (bottom). These relationships typically result in exponential attenuation, longer-tail attenuation, and shorter-tail attenuation, respectively.

4.4.2 Path throughput

We use similar definitions for all terms previously defined for the classical path integral in Sec. 4.2.4 with specific alterations. First off, Bitterli et al. makes the assumption that the phase function and the medium albedo are *uncorrelated* for any specific realization μ . For this reason we choose to replace $\mu_s(\mathbf{x})$ with the albedo, $\alpha_s(\mathbf{x})$, within the definition of μ_* in Eq. (4.27) then redefine it as,

$$\rho_\star(\mathbf{x}_j, \mathbf{x}_j \vec{\mathbf{x}}_{j+1}, \mathbf{x}_j \vec{\mathbf{x}}_{j-1}) := \begin{cases} \rho_s(\mathbf{x}_j, \mathbf{x}_j \vec{\mathbf{x}}_{j+1}, \mathbf{x}_j \vec{\mathbf{x}}_{j-1}), & \mathbf{x}_j \in A \\ \alpha_s(\mathbf{x}_j) \rho_p(\mathbf{x}_j, \mathbf{x}_j \vec{\mathbf{x}}_{j+1}, \mathbf{x}_j \vec{\mathbf{x}}_{j-1}), & \mathbf{x}_j \in V. \end{cases} \quad (4.52)$$

We then introduce the quantity $\Sigma_\mu(\mathbf{x})$,

$$\Sigma_\mu(\mathbf{x}) := \begin{cases} \mu_r(\mathbf{x}), & \text{if } \mathbf{x} \in V, \\ 1, & \text{if } \mathbf{x} \in A, \end{cases} \quad (4.53)$$

such that we can formulate the path throughput as,

$$\langle g(\bar{\mathbf{x}}) \rangle := \left(\prod_{j=1}^{n-1} \rho_\star(\mathbf{x}_j, \mathbf{x}_j \vec{\mathbf{x}}_{j+1}, \mathbf{x}_j \vec{\mathbf{x}}_{j-1}) \right) \left(\prod_{j=0}^{n-1} G(\mathbf{x}_{j+1}, \mathbf{x}_j) \right) \quad (4.54)$$

$$\cdot \left(\prod_{j=0}^{n-1} T_\mu(\mathbf{x}_j, \mathbf{x}_{j+1}) \Sigma_\mu(\mathbf{x}_{j+1}) \right). \quad (4.55)$$

Bitterli et al. makes the assumption that the "memory" of the current light ray only keeps track of the last bounce. Meaning, that all free-flight distributions only depend on the previous free-flight sample when accounting for correlations. Intuitively, this means that light will transition into a different medium realization every time there is a medium interaction allowing,

$$\left(\prod_{j=0}^{n-1} T_\mu(\mathbf{x}_j, \mathbf{x}_{j+1}) \Sigma_\mu(\mathbf{x}_{j+1}) \right) = \prod_{j=0}^{n-1} \langle \bar{T}_\mu(\mathbf{x}_j, \mathbf{x}_{j+1}) \rangle, \quad (4.56)$$

the ensemble average to be computed independently for each path segment rather than for an entire path. We have also substituted the segment contribution T_μ with a transport kernel \bar{T}_μ .

4.4.3 Transport Kernel

Based on the assumption that the "memory" of light only depends on the previous and next bounce, The transport kernel becomes comprised of four different transport functions,

$$\bar{T}(\mathbf{x}, \mathbf{y}) = \langle \bar{T}_\mu(\mathbf{x}, \mathbf{y}) \rangle := \begin{cases} ff(\mathbf{x}, \mathbf{y}), & \text{if } \mathbf{x} \in \mathbb{A} \& \mathbf{y} \in \mathbb{A}, \\ fp(\mathbf{x}, \mathbf{y}), & \text{if } \mathbf{x} \in \mathbb{A} \& \mathbf{y} \in \mathbb{V}, \\ pf(\mathbf{x}, \mathbf{y}), & \text{if } \mathbf{x} \in \mathbb{V} \& \mathbf{y} \in \mathbb{A}, \\ pp(\mathbf{x}, \mathbf{y}), & \text{if } \mathbf{x} \in \mathbb{V} \& \mathbf{y} \in \mathbb{V}, \end{cases} \quad (4.57)$$

depending on whether the two vertices \mathbf{x} and \mathbf{y} lie along a surface or exist within a medium.

Notation. The character "f" is used within the function naming scheme to indicate light either starts or ends on a "free-space" while "p" indicates light either starts or ends on a real particle within the medium. The first letter of the function name coincides with the start vertex \mathbf{x} , while the second letter coincides with the second vertex \mathbf{y} . For example, $fp(\mathbf{x}, \mathbf{y})$ means this function is used when the path segment starts on a surface and ends within a medium with respect to the flow of light.

Transport functions. The probability density of a particle existing at \mathbf{x} is directly proportional to $\mu_\mu(\mathbf{x})$. We can use this to express the expectation of some function $f(\mathbf{x})$ conditioned on a scattering particle as a weighted ensemble average: $\langle f(\mathbf{x})\lambda_\mu(\mathbf{x}) \rangle$ with $\lambda_\mu(\mathbf{x}) := \frac{\mu_\mu(\mathbf{x})}{\mu(\mathbf{x})}$. With this weighted ensemble average, we can define the four different transport functions as,

$$\begin{aligned} ff(\mathbf{x}, \mathbf{y}) &:= \langle \bar{T}_\mu(\mathbf{x}, \mathbf{y}) \rangle, \\ fp(\mathbf{x}, \mathbf{y}) &:= \langle \mu_\mu(\mathbf{y}) \bar{T}_\mu(\mathbf{x}, \mathbf{y}) \rangle \\ pf(\mathbf{x}, \mathbf{y}) &:= \langle \lambda_\mu(\mathbf{x}) \bar{T}_\mu(\mathbf{x}, \mathbf{y}) \rangle \\ pp(\mathbf{x}, \mathbf{y}) &:= \langle \mu_\mu(\mathbf{y}) \lambda_\mu(\mathbf{x}) \bar{T}_\mu(\mathbf{x}, \mathbf{y}) \rangle. \end{aligned} \quad (4.58)$$

Relationship between transport functions. The functions ff and pf are the ensemble average of transmittances so they are also transmittances, while fp and pp are the ensemble average of free-flight distributions so they are also themselves free-flight distributions. Since transmittance is a CCDF with the free-flight distribution being its PDF, if we possess a formulation for one transport function,

$$\begin{aligned} ff(\mathbf{x}, \mathbf{x}_t) &:= \int_t^\infty fp(\mathbf{x}, \mathbf{x}_s) ds = 1 - \int_0^s fp(\mathbf{x}, \mathbf{x}_s) ds, \\ pf(\mathbf{x}, \mathbf{x}_t) &:= \int_t^\infty pp(\mathbf{x}, \mathbf{x}_s) ds = 1 - \int_0^s pp(\mathbf{x}, \mathbf{x}_s) ds, \\ fp(\mathbf{x}, \mathbf{x}_t) &:= \mu(\mathbf{x}_t) pf(\mathbf{x}, \mathbf{x}_t), \end{aligned} \quad (4.59)$$

we can naturally derive all of the others.

Reciprocity. The transport kernel in Eq. (4.57) is not reciprocal, i.e. $\bar{T}(\mathbf{x}, \mathbf{y}) \neq \bar{T}(\mathbf{y}, \mathbf{x})$, so energy is still not yet conserved. However, we can easily fix this by applying a simple mathematical manipulation. If we define a modified transport kernel,

$$\tilde{T}(\mathbf{x}, \mathbf{y}) := \frac{\bar{T}(\mathbf{x}, \mathbf{y})}{\Sigma(\mathbf{y})} = \begin{cases} ff(\mathbf{x}, \mathbf{y}), & \text{if } \mathbf{x} \in \mathbb{A} \text{ \& } \mathbf{y} \in \mathbb{A}, \\ \frac{fp(\mathbf{x}, \mathbf{y})}{\mu(\mathbf{y})} = pf(\mathbf{x}, \mathbf{y}), & \text{if } \mathbf{x} \in \mathbb{A} \text{ \& } \mathbf{y} \in \mathbb{V}, \\ pf(\mathbf{x}, \mathbf{y}), & \text{if } \mathbf{x} \in \mathbb{V} \text{ \& } \mathbf{y} \in \mathbb{A}, \\ \frac{pp(\mathbf{x}, \mathbf{y})}{\mu(\mathbf{y})}, & \text{if } \mathbf{x} \in \mathbb{V} \text{ \& } \mathbf{y} \in \mathbb{V}, \end{cases} \quad (4.60)$$

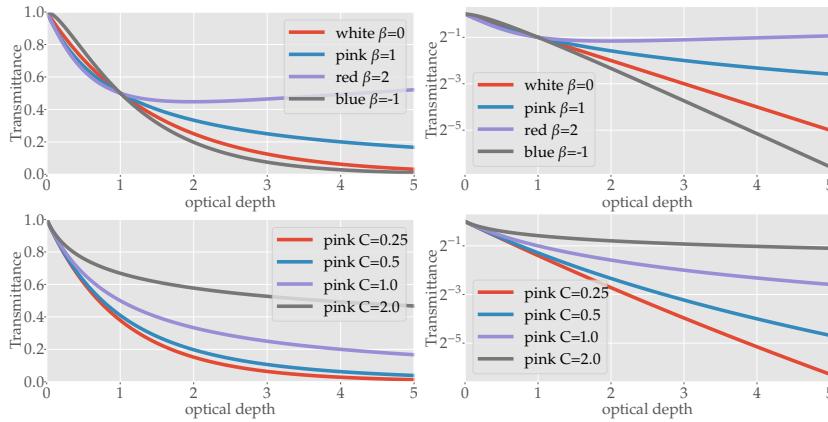


Figure 4.10: We visualize different surface-to-surface transmittance functions supported by the Davis-Weinstein transmittance model (Eq. (4.63)). On the top row, we visualize four different choices for β while keeping $C = 0.8$ constant (left) and graph the same functions in the log domain on the right. In the bottom row, we plot four different choices for C while keeping $\beta = 1$ configured to pink noise transmittance (Eq. (4.64)) (left) and also plot the results in the log domain (right).

such that all transport functions become transmittances, then the path integral becomes naively reciprocal since any transmittance function by definition is reciprocal. Since $\tilde{T}(\mathbf{x}, \mathbf{y})\Sigma(\mathbf{y}) = \bar{T}(\mathbf{x}, \mathbf{y})$ we move $\Sigma(\mathbf{y})$ back into ρ_{\star} , so the path throughput can be written as,

$$\langle g(\bar{\mathbf{x}}) \rangle := \left(\prod_{j=1}^{n-1} \rho_{\star}(\mathbf{x}_j, \mathbf{x}_j \vec{\mathbf{x}}_{j+1}, \mathbf{x}_j \vec{\mathbf{x}}_{j-1}) \right) \left(\prod_{j=0}^{n-1} G(\mathbf{x}_j, \mathbf{x}_{j+1}) \bar{T}(\mathbf{x}_j, \mathbf{x}_{j+1}) \right), \quad (4.61)$$

where the scattering function is defined as,

$$\rho_{\star}(\mathbf{x}_j, \mathbf{x}_j \vec{\mathbf{x}}_{j+1}, \mathbf{x}_j \vec{\mathbf{x}}_{j-1}) := \begin{cases} \rho_s(\mathbf{x}_j, \mathbf{x}_j \vec{\mathbf{x}}_{j+1}, \mathbf{x}_j \vec{\mathbf{x}}_{j-1}), & \mathbf{x}_j \in \mathbb{A} \\ \mu_s(\mathbf{x}_j) \rho_p(\mathbf{x}_j, \mathbf{x}_j \vec{\mathbf{x}}_{j+1}, \mathbf{x}_j \vec{\mathbf{x}}_{j-1}), & \mathbf{x}_j \in \mathbb{V}. \end{cases} \quad (4.62)$$

completing the path integral for non-exponential transmittance.

4.4.4 Davis-Weinstein transmittance model

The Davis-Weinstein transmittance model [74],

$$ff(\mathbf{x}_a, \mathbf{x}_b) := \left(1 + \left(\int_a^b \mu_r(t) dt \right)^{\beta} C^{1+\beta} \right)^{-\frac{\left(\int_a^b \mu_r(t) dt \right)^{1-\beta}}{C^{\beta+1}}}, \quad (4.63)$$

was introduced to graphics by Bitterli et al. alongside the non-exponential radiative transfer framework [4]. This formulation for transmittance was constructed to roughly match fractal noise where β controls how rough the noise appears while C controls its overall amplitude. In Fig. 4.10 we show how this transmittance function varies for a variety of different "colors" and amplitudes of noise. This model is not always guaranteed to result in physically-based transmittance models for all parameter configurations (see red noise plot in Fig. 4.10). For physically-based models, we recommend keeping $C \in [0, 1]$ and $\beta \in [-1, 1]$.

Before the introduction of the contributions within this dissertation, the only class of Davis-Weinstein transmittances which could be rendered in an unbiased manner coincided with $\beta = 0$ (white noise, exponential media)

[74]: Davis et al. (2011), 'Radiation Propagation in Random Media: From Positive to Negative Correlations in High-Frequency Fluctuations'

[4]: Bitterli et al. (2018), 'A Radiative Transfer Framework for Non-Exponential Media'

or $\beta = 1$ (pink noise),

$$\text{ff}(\mathbf{x}_a, \mathbf{x}_b)_{\text{pink}} := \left(1 + C^2 \int_a^b \mu_r(\mathbf{x}_t) dt\right)^{-C^{-2}}, \quad (4.64)$$

out of the entire continuum of possible transmittance functions. We provide a more thorough discussion on how we alleviate this to support general unbiased non-exponential rendering in Chapter 8.



5 Monte Carlo based rendering algorithms

Simulating the transport of light along surfaces (Chapter 2) or through participating media (Chapter 4) involves computing the results of high-dimensional integrals in either a recursive manner, or over all of path space. These integration problems are non-trivial and cannot be evaluated analytically for the majority of scenes we are interested in rendering. This is why most modern solutions to these integration problems rely on Monte Carlo integration (Chapter 3) to instead *estimate* these integrals.

In this chapter, we will combine the light transport theory we introduced in Chapter 2 and Chapter 4 with the concept of Monte Carlo integration (Chapter 3) to review how some modern rendering algorithms are practically derived. All of the algorithms which we introduce in this chapter are either directly impacted, improved by, or involved in conveying the contributions of this dissertation which will be covered in subsequent chapters.

We start by introducing naive unidirectional path tracing in Sec. 5.1 followed by unidirectional path tracing with next event estimation in Sec. 5.1.2. We then introduce the idea of light tracing in Sec. 5.1.4 which is operationally similar to path tracing except it simulates the flow of light in the opposite direction. Next, we introduce bidirectional path tracing which interprets path tracing and light tracing as different importance sampling strategies, thus enabling them to be combined using MIS (Sec. 3.3.2). Afterwards, we introduce the photon-mapping family of algorithms in Sec. 5.3 which, while biased, tends to be a good choice for simulating difficult visual phenomena such as caustics. In Sec. 5.3.1 we introduce a progressive form of photon mapping which is consistent, allowing for the error to disappear in the limit. Finally, we conclude this section by discussing volumetric path tracing within the classical formulation (Sec. 5.4.1), the null-scattering formulation (Sec. 5.4.5) and the non-exponential formulation (Sec. 5.4.6).

5.1	Unidirectional surface-only methods	58
5.1.1	Path tracing	58
5.1.2	Next-event estimation	59
5.1.3	Russian Roulette termination	60
5.1.4	Unidirectional light tracing	61
5.2	Bidirectional path tracing	61
5.3	Photon mapping	64
5.3.1	Progressive photon mapping	65
5.4	Volumetric path tracing	67
5.4.1	VRE	67
5.4.2	Free-flight distance sampling	68
5.4.3	Next Event Estimation	71
5.4.4	Aside: Emission	72
5.4.5	Null VRE	73
5.4.6	Non-Classical Transport	75

5.1 Unidirectional surface-only methods

Rendering an image containing only surfaces using Monte Carlo integration involves first deriving a primary estimator to estimate either the recursive measurement equation (2.23) or the path integral formulation (2.36). After a primary estimator is obtained, we then employ a secondary estimator to drive the error down towards zero. Due to the nature of rendering algorithms, we typically refer to invocations of a primary estimator that estimates the radiant intensity of a pixel as a *pixel sample* where secondary estimators combine multiple pixel samples together. With the exception of the photon mapping family of methods (see Sec. 5.3) all rendering algorithms discussed within this chapter employ simple averaging for secondary estimation.

In this section, we review two unidirectional rendering algorithms known as path tracing and light tracing that construct light paths by *iteratively* importance sampling scattering events. We refer to these techniques as *unidirectional* because they initially start at either the sensor or a source, respectively, and then trace a path that ends at the opposite endpoint. Put more simply, they stochastically simulate the flow of light in a *single* direction. In the process of introducing these algorithms, we additionally review a variance reduction technique known as *next event estimation* which can improve unidirectional rendering algorithms.

We initially choose to consider surface-only transport to more easily convey the process of applying Monte Carlo integration in deriving rendering algorithms. Both of the unidirectional algorithms we introduce in this section can be modified to render participating media, however, due to the additional work required, we postpone the discussion of volumetric rendering until Sec. 5.4.

5.1.1 Path tracing

To derive the path tracing algorithm, we start from the measurement equation,

$$I_k := \int_{\mathbb{A}} \int_{\mathbb{S}^2} W_e^{(k)}(\mathbf{x}, \vec{\omega}) L_i(\mathbf{x}, \vec{\omega}) d\Omega^\perp(\vec{\omega}) dA(\mathbf{x}), \quad (5.1)$$

where we assume that we have an importance function $W_e^{(k)}(\mathbf{x}, \vec{\omega})$ for pixel k that can be evaluated for any valid \mathbf{x} or $\vec{\omega}$. We start by importance sampling the integral over area by uniformly sampling an \mathbf{x} on the image plane spanned by pixel k and then we sample a corresponding direction $\vec{\omega}$ resulting in the estimator,

$$\langle I_k \rangle := \frac{W_e^{(k)}(\mathbf{x}, \vec{\omega}) \langle L_i(\mathbf{x}, \vec{\omega}) \rangle}{p(\mathbf{x}) p(\vec{\omega})}, \quad (5.2)$$

where the PDF for the direction will depend on the properties of the sensor, or virtual camera, being used. While orthogonal to this dissertation, we refer the interested reader to Pharr's book [18] for an introduction to

[18]: Pharr et al. (2016), *Physically Based Rendering*

virtual camera systems and Kolb's work [75] for an introduction to realistic camera models. After sampling an initial position and direction, we need to estimate the radiance.

Estimating radiance. To estimate the radiance in Eq. (5.2), we need to apply Monte Carlo integration to estimate the surface-only rendering equation (2.22),

$$L_o(\mathbf{x}, \vec{\omega}_o) := L_e(\mathbf{x}, \vec{\omega}_o) + \int_{\mathbb{S}^2} \rho_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) L_i(\mathbf{x}, \vec{\omega}_i) |\cos \theta| d\Omega(\vec{\omega}_i). \quad (5.3)$$

Assuming we can directly evaluate $L_e(\mathbf{x}, \vec{\omega}_o)$ at the current scattering location, we can estimate the in-scattered radiance by importance sampling the sphere of directions proportionally to the BSDF and recursing,

$$\langle L_o(\mathbf{x}, \vec{\omega}_o) \rangle := L_e(\mathbf{x}, \vec{\omega}_o) + \frac{\rho_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) |\cos \theta| \langle L_i(\mathbf{x}, \vec{\omega}_i) \rangle}{p(\vec{\omega}_i)}. \quad (5.4)$$

The resulting Monte Carlo estimator is known as unidirectional path tracing. *Naive unidirectional path tracing* is a variant of Eq. (5.4) which imposes a termination criterion where the estimator stops recursing once an emissive surface is encountered, thus, the estimator effectively traces an entire path from the camera to the light source. Unidirectional path tracing was introduced by Kajiya [20], and we illustrate its naive variant in Fig. 5.1.

5.1.2 Next-event estimation

Unidirectional path tracing can be inefficient because it *only* accumulates radiance when a path collides with an emissive source. If the emitters are sufficiently small, a path may scatter hundreds or even thousands of times before the algorithm would return a non-zero pixel intensity.

We can alleviate these inefficiencies by choosing to importance sample the direct illumination at every scattering event instead of accumulating the emission at every bounce. Mechanically, we accomplish this by tracing *shadow connections* directly to emissive sources which effectively simulates the result of terminating on the "next-event" in the recursion. Thus, this technique is referred to as *next-event estimation* (NEE).

These shadow connections compute the *direct illumination* ($L_d(\mathbf{x}, \vec{\omega}_o)$), or direct transport, from the perspective of the current scattering vertex,

$$L_d(\mathbf{x}, \vec{\omega}_o) := \int_{\mathbb{S}^2} \rho_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) L_e(\mathbf{z}, -\vec{\omega}_i) |\cos \theta| d\Omega(\vec{\omega}_i), \quad (5.5)$$

where $\mathbf{z} = \mathbf{r}(\mathbf{x}, \vec{\omega})$ is returned by the ray-tracing operator. Since many emissive sources in virtual environments are defined as surfaces, it becomes

[75]: Kolb et al. (1995), 'A Realistic Camera Model for Computer Graphics'

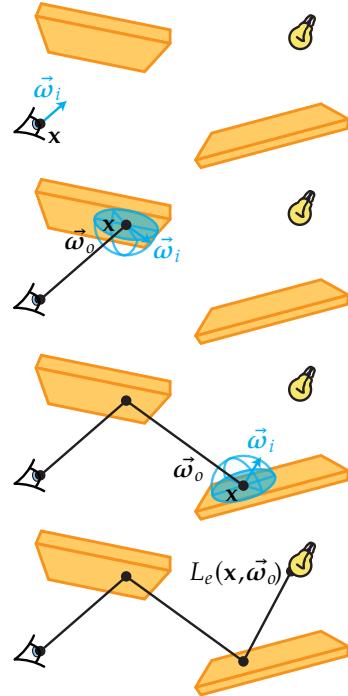


Figure 5.1: An illustration of naive unidirectional path tracing (5.4) which recursively estimates the rendering equation (5.3) by importance sampling directions proportionally to the BSDF.

[20]: Kajiya (1986), 'The Rendering Equation'

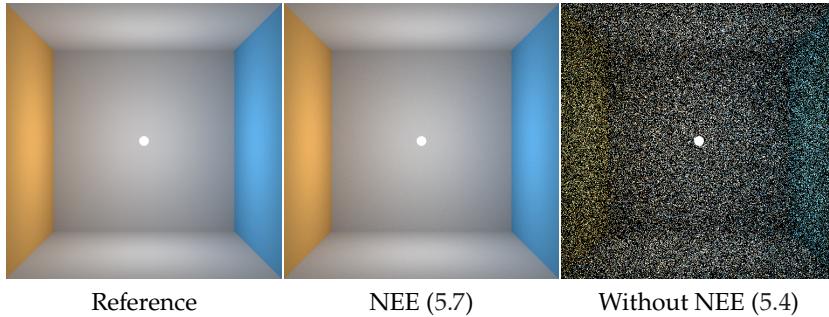


Figure 5.2: An equal render time (roughly 75 seconds) comparison between naive unidirectional path tracing (right) and path tracing with NEE (middle) in a scene containing a single small spherical light.

beneficial to perform a change of variables into Cartesian coordinates,

$$L_d(\mathbf{x}, \vec{\mathbf{xy}}) := \int_{\mathbb{A}} \rho_s(\mathbf{x}, \vec{\mathbf{xz}}, \vec{\mathbf{xy}}) G(\mathbf{x}, \mathbf{z}) L_e(\mathbf{z}, \vec{\mathbf{zx}}) dA(\mathbf{z}), \quad (5.6)$$

so we can formulate a Monte Carlo estimator for the direct illumination,

$$\langle L_d(\mathbf{x}, \vec{\mathbf{xy}}) \rangle := \frac{\rho_s(\mathbf{x}, \vec{\mathbf{xz}}, \vec{\mathbf{xy}}) G(\mathbf{x}, \mathbf{z}) L_e(\mathbf{z}, \vec{\mathbf{zx}})}{p(\mathbf{z})}, \quad (5.7)$$

as a form of emitter sampling¹.

Special care has to be taken to guarantee no double counting of emission occurs when NEE is employed to estimate direct illumination. This typically involves only accumulating emission in Eq. (5.4) after specular scattering events and otherwise relying on Eq. (5.7) to accumulate emission. However, we leave this as an implementation detail. We conceptually illustrate unidirectional path tracing with NEE in Fig. 5.3 and we convey the practical benefits of employing NEE using an equal time comparison to naive unidirectional path tracing in Fig. 5.2.

¹: See Pharr's book [18] for a description of different emitters and how to importance sample them.

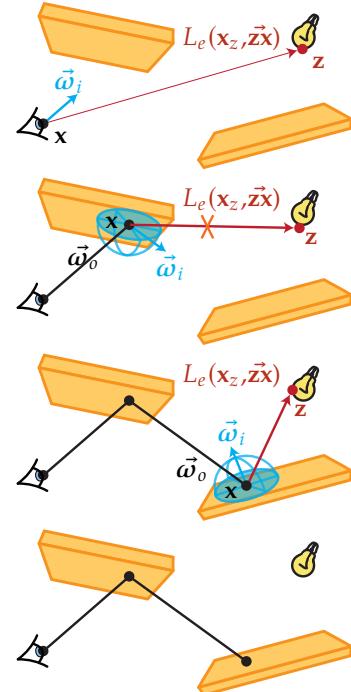


Figure 5.3: An illustration of unidirectional path tracing with NEE (5.7) which recursively estimates the direct (red) and indirect (blue) illumination separately from the perspective of the vertex \mathbf{x} . When there is no visibility between the sampled point on the light and current scatter location (second from top), the direct illumination estimate returns zero.

[76]: Forsythe et al. (1950), 'Matrix inversion by a Monte Carlo method'

[77]: Arvo et al. (1990), 'Particle Transport and Image Synthesis'

Unbiasedness. If $\langle X \rangle$ is unbiased then,

$$E[\langle X \rangle_{RR}] (1 - P) \cdot E\left[\frac{0}{1 - P}\right] + (P) \cdot E\left[\frac{\langle X \rangle}{P}\right] = E[\langle X \rangle], \quad (5.9)$$

Eq. (5.8) is also unbiased.

Performance. There is always a tradeoff between cost and variance when applying Russian Roulette as it tends to *increase* variance in return for *decreasing* cost through early termination. We refer the interested reader to more recent work in efficiency aware Russian Roulette [78] for a more in depth discussion on the optimal application of Russian Roulette to problems in light transport.

[78]: Rath et al. (2022), 'EARS: Efficiency-Aware Russian Roulette and Splitting'

5.1.4 Unidirectional light tracing

Since we assume the transport of light is reciprocal (2.16), we can choose to trace paths originating from either the sensor or an emissive source. Unidirectional path tracing samples paths originating from the sensor while *unidirectional light tracing* is an alternative family of algorithms which sample paths originating from emissive sources [79].

Light tracing is similar enough to path tracing that we choose not to mathematically describe the algorithm. A purely unidirectional light tracing estimator and a light tracing estimator with NEE would be reminiscent to Eq. (5.4) and Eq. (5.7), respectively, except with the directions reversed and the importance flipped with the emissive radiance. We instead illustrate light tracing with NEE in Fig. 5.4.

Light tracing algorithms typically perform better in scenes where emissive sources are visually obstructed from the perspective of the sensor and the paths which light must travel to reach emissive sources are difficult to sample. This specific case can arise frequently when employing physically-based lighting. In the real world, many of our light sources consist of super heated filament that are placed within glass that reflects and refracts the emitted light into the environment. Sampling the exact directions necessary to reflect and refract paths to collide with the emissive filament is probabilistically challenging, thus light tracing can be more efficient than path tracing.

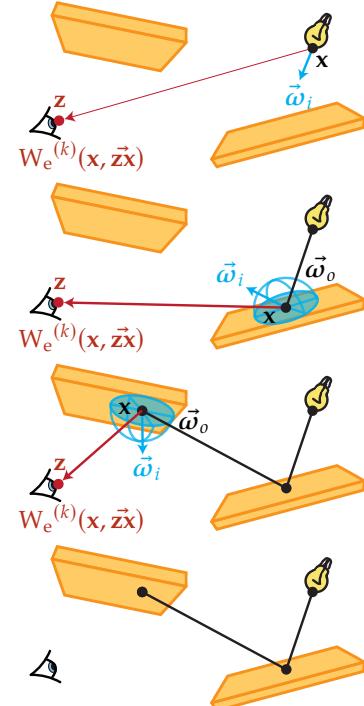


Figure 5.4: An illustration of unidirectional light tracing with NEE.

5.2 Bidirectional path tracing

[79]: Arvo (1986), 'Backward Ray Tracing'

Unidirectional path tracing and light tracing are essentially two Monte Carlo estimators which differ by how they importance sample paths. In Sec. 3.3.2, we reviewed a technique known as multiple importance sampling (MIS) for combining multiple different importance sampling strategies. Applying MIS to combine unidirectional light tracing and path tracing, amongst all other possible path sampling strategies, results in a rendering algorithm known as bidirectional path tracing (BDPT) [33, 80, 81].

[33]: Veach et al. (1995), 'Optimally Combining Sampling Techniques for Monte Carlo Rendering'

[80]: Lafortune et al. (1993), 'Bi-Directional Path Tracing'

[81]: Lafortune et al. (1996), 'Rendering Participating Media with Bidirectional Path Tracing'

Initially, bidirectional path tracing traces two different sub-paths. The first sub-path ($\bar{\mathbf{x}} = \mathbf{x}_0, \dots, \mathbf{x}_{q-1}$) originates from the sensor using unidirectional path tracing, while the other sub-path ($\bar{\mathbf{y}} = \mathbf{y}_{l-1}, \dots, \mathbf{y}_0$) originates from the emissive source using unidirectional light tracing, where q is the length of the sensor sub-path and l is the length of the source sub-path. We denote the indices for both sub-paths to increase in opposing directions such that the conjoined sub-paths $\bar{\mathbf{z}} = \bar{\mathbf{x}}\bar{\mathbf{y}} = \mathbf{x}_0, \dots, \mathbf{x}_{q-1}\mathbf{y}_{l-1}, \dots, \mathbf{y}_0$ denote an entire path from sensor $q = 0$ to source $l = 0$, where we use the notation, $\bar{\mathbf{x}}\bar{\mathbf{y}}$, to denote that the path $\bar{\mathbf{y}}$ is appended to the end of path $\bar{\mathbf{x}}$.

Bidirectional path tracing is a Monte Carlo estimator for the path-integral formulation (2.36),

$$\begin{aligned} I_k &:= \int_{\mathbb{P}} f^{(k)}(\bar{\mathbf{z}}) d\mu(\bar{\mathbf{z}}) \\ \langle I_k \rangle &:= \frac{f^{(k)}(\bar{\mathbf{z}})}{p(\bar{\mathbf{z}})}, \end{aligned} \quad (5.10)$$

which importance samples *entire paths* ($\bar{\mathbf{z}}$). Given $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$, bidirectional path tracing constructs an MIS estimator for any n -length path by considering all possible paths ($\bar{\mathbf{y}}_n, \bar{\mathbf{x}}_0\bar{\mathbf{y}}_{n-1}, \bar{\mathbf{x}}_1\bar{\mathbf{y}}_{n-2}, \dots, \bar{\mathbf{x}}_{n-1}\bar{\mathbf{y}}_0, \bar{\mathbf{x}}_n$) as different sampling strategies. We use the notation $\bar{\mathbf{y}}_n$ to refer to a sub-path containing the first n vertices of the path $\bar{\mathbf{y}}$. For example, $\bar{\mathbf{z}}_5 = \bar{\mathbf{x}}_2\bar{\mathbf{y}}_3 = \mathbf{x}_0\mathbf{x}_1\mathbf{y}_2\mathbf{y}_1\mathbf{y}_0$. We visualize the various path sampling strategies used by bidirectional path tracing in Fig. 5.5 for a path length of $n = 5$.

Bidirectional path tracing is one of the most robust rendering algorithms and typically maintains similar variance to the best available path-sampling strategies. However, combining many different path sampling strategies is computationally expensive and often times bidirectional path tracing performs worse than simpler techniques within equal-time comparisons[82]. In high performance production contexts, unidirectional methods are typically employed instead [83].

In Chapter 6, we will derive an estimator for the problem of classical transmittance estimation which is analogous to bidirectional path tracing, however, it will also exhibit similar computational drawbacks.

[82]: Bitterli et al. (2021), ‘Correlations and Reuse for Fast and Accurate Physically Based Light Transport’

[83]: Burley et al. (2018), ‘The Design and Evolution of Disney’s Hyperion Renderer’

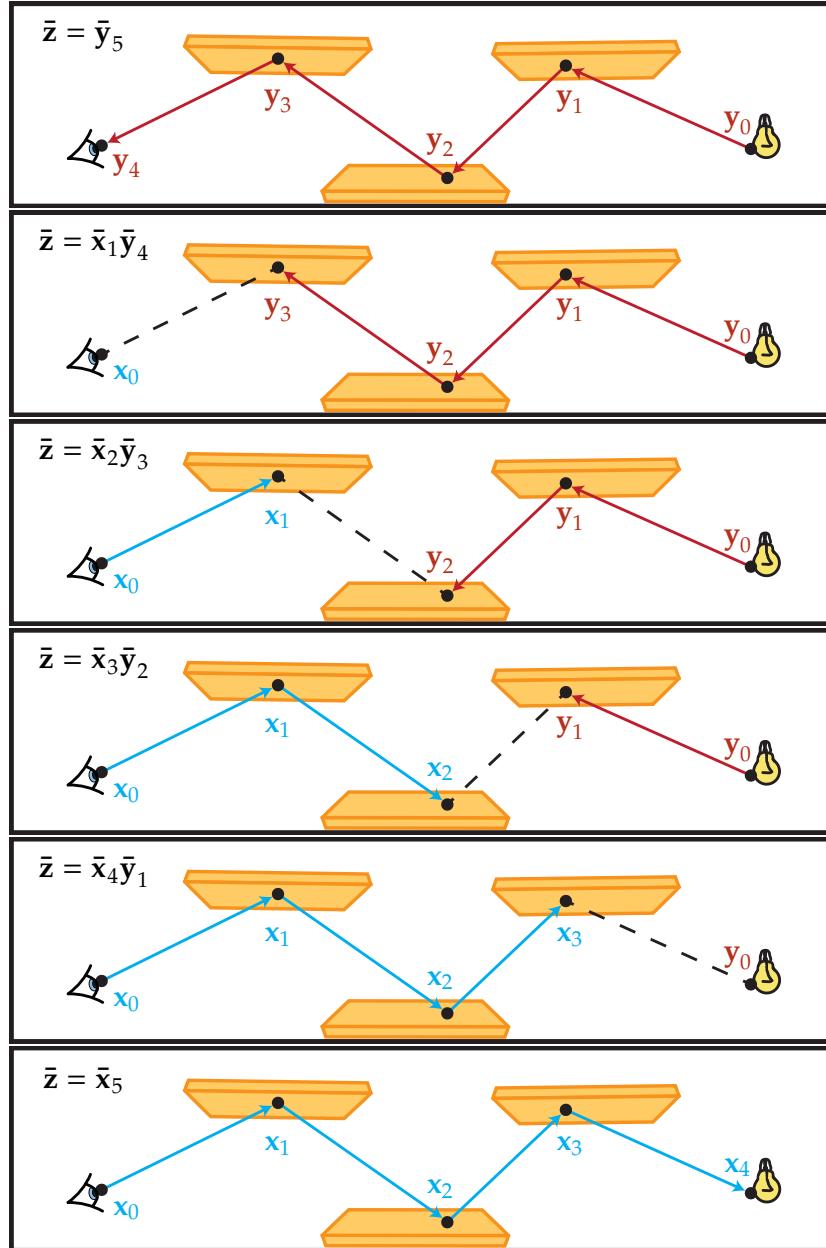


Figure 5.5: An illustration of the various path sampling strategies employed by bidirectional path tracing for a path length of $n = 5$. Blue lines indicate the sensor subpath while red lines indicate the source subpath. The dashed line represents a NEE evaluation between the endpoints of the two subpaths.

5.3 Photon mapping

The photon mapping family of rendering algorithms excel at rendering scenes containing a visual phenomenon known as *caustics*. Caustics are excessively bright regions which arise due to the hyper focusing of light on diffuse surfaces from specular reflections and refractions. One of the most obvious examples of this visual phenomena is the classic movie trope of a child focusing the sun's light through the lens of a magnifying glass at a mound of unsuspecting insects (or a toy in Toy Story). The lens focuses the sun's rays at a specific point in space resulting in a bright (and hot) point on the ground. Other common real life examples of caustics include the focusing of light at the bottom of wine glasses, the bright patterns at the bottom of swimming pools from reflections along the surface of water, and many chandelier light fixtures which illuminate rooms by focusing light through glass.

Since caustics arise due to specular reflections and refractions, which are effectively delta functions, path tracing tends to struggle at capturing them. While bidirectional path tracing and light tracing can more easily render direct caustics, they still struggle to capture reflections of caustics arising from *specular-diffuse-specular* (SDS) sequences of scattering events (see Fig. 5.7). Photon mapping is typically the preferred rendering algorithm for scenes whose pixel intensities are heavily dominated by SDS sub-paths.

Two-pass algorithm. Photon mapping is a two-pass algorithm which, similarly to bidirectional path tracing, traces two sub-paths. The first originating from the emissive source and the other from the sensor [84–88]. However, unlike bidirectional path tracing, the sub-path from the emitter is traced first as a pre-process to *approximate* the indirect illumination by storing *photons* within a global data structure known as a *photon map* [85, 86]. Once the photon map is populated, sensor sub-paths are then traced to measure the direct illumination and *approximate* the indirect illumination from the photon map. We illustrate this process in Fig. 5.6.

Photons within the photon map store estimates for flux which is related to incident radiance by (2.6),

$$L_i(\mathbf{x}, \vec{\omega}_i) = \frac{d^2\Phi(\mathbf{x}, \vec{\omega}_i)}{|\cos \theta| dA(\mathbf{x}) d\Omega(\vec{\omega}_i)}. \quad (5.11)$$

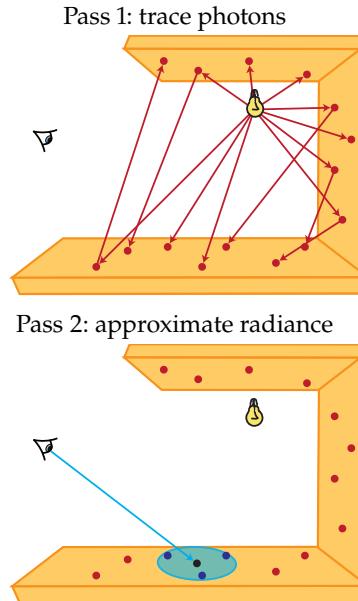
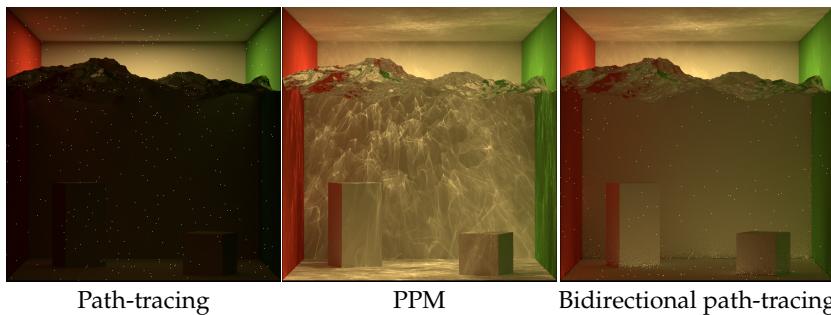


Figure 5.6: An illustration of the photon mapping algorithm where we trace photons originating from an emitter as a first pass (left). In the second pass (right), we trace a path originating from the camera where we accumulate the direct illumination using path tracing and indirect illumination through accumulating the contributions from all (blue) photons within the kernel function (blue disc).

[84]: Jensen (1996), ‘The Photon Map in Global Illumination’

[85]: Jensen et al. (1995), ‘Photon Maps in Bidirectional Monte Carlo Ray Tracing of Complex Objects’

[86]: Jensen (1996), ‘Global Illumination Using Photon Maps’

[87]: Jensen (2001), *Realistic Image Synthesis Using Photon Mapping*

[88]: Hachisuka et al. (2012), ‘A Path Space Extension for Robust Light Transport Simulation’

Figure 5.7: An equal pixel sample (512 ps) comparison between path tracing with NEE (left) progressive photon mapping (middle) and bidirectional path tracing (right) in a scene containing mostly SDS paths. Photon mapping is the only algorithm which is able to capture all reflections of caustics while bidirectional photon mapping only captures the direct caustics on the ceiling.

We then replace the incident radiance within the rendering equation (5.3) with Eq. (5.11),

$$\begin{aligned} L_o(\mathbf{x}, \vec{\omega}_o) &:= L_e(\mathbf{x}, \vec{\omega}_o) + \int_{\mathbb{H}} \rho_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) \frac{d^2\Phi(\mathbf{x}, \vec{\omega}_i)}{|\cos \theta| dA(\mathbf{x}) d\Omega(\vec{\omega}_i)} |\cos \theta| d\Omega(\vec{\omega}_i) \\ &= L_e(\mathbf{x}, \vec{\omega}_o) + \int_{\mathbb{H}} \rho_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) \frac{d^2\Phi(\mathbf{x}, \vec{\omega}_i)}{dA(\mathbf{x}) d\Omega(\vec{\omega}_i)} d\Omega(\vec{\omega}_i), \end{aligned} \quad (5.12)$$

resulting in a formulation which can then be approximated by accumulating the flux ($\Phi_j(\mathbf{x}, \vec{\omega}_i)$),

$$L_o(\mathbf{x}, \vec{\omega}_o) \approx L_e(\mathbf{x}, \vec{\omega}_o) + \sum_{j=1}^{n_p} \rho_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) \frac{\Phi_j(\mathbf{x}, \vec{\omega}_i)}{A(\mathbf{x})}, \quad (5.13)$$

from all n_p photons around the neighborhood ($A(\mathbf{x})$) of the scatter location. Photon mapping then becomes a form of *kernel density estimation*,

$$\langle L_o(\mathbf{x}, \vec{\omega}_o) \rangle := L_e(\mathbf{x}, \vec{\omega}_o) + \sum_{p=1}^{n_p(r)} \rho_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) K(\mathbf{x}, \mathbf{x}_p, r) \Phi_p(\mathbf{x}_p, \vec{\omega}_i), \quad (5.14)$$

over all $n_p(r)$ photons within a radius (r) of the current path vertex to form an approximation of the outgoing radiance.

Kernel functions. Some common choices for kernel functions which we consider within this dissertation are *disk kernels* and *cone kernels*. A disk kernel is defined as,

$$K(\mathbf{x}, \mathbf{x}_p, r) := \frac{1}{\pi r^2}, \quad (5.15)$$

while a cone kernel is defined as

$$K(\mathbf{x}, \mathbf{x}_p, r) := \frac{3}{\pi r^2} \left(1 - \frac{\|\mathbf{x} - \mathbf{x}_p\|}{r}\right). \quad (5.16)$$

Any kernel function can be used as long as it integrates to 1 over the area spanned by the kernel. While the kernel density estimation portion of photon mapping is not a direct application of Monte Carlo, the photon and sensor sub-paths generated by the algorithm are assumed to be sampled in an unbiased manner using Monte Carlo techniques.

5.3.1 Progressive photon mapping

Unfortunately, photon mapping is a *biased* algorithm whose bias is dependent upon the radius of the kernel. The larger the radius the more blurred the results appear (see Fig. 5.8). However, photon mapping is *consistent* with respect to the kernel radius so we would expect the bias to disappear in the limit.

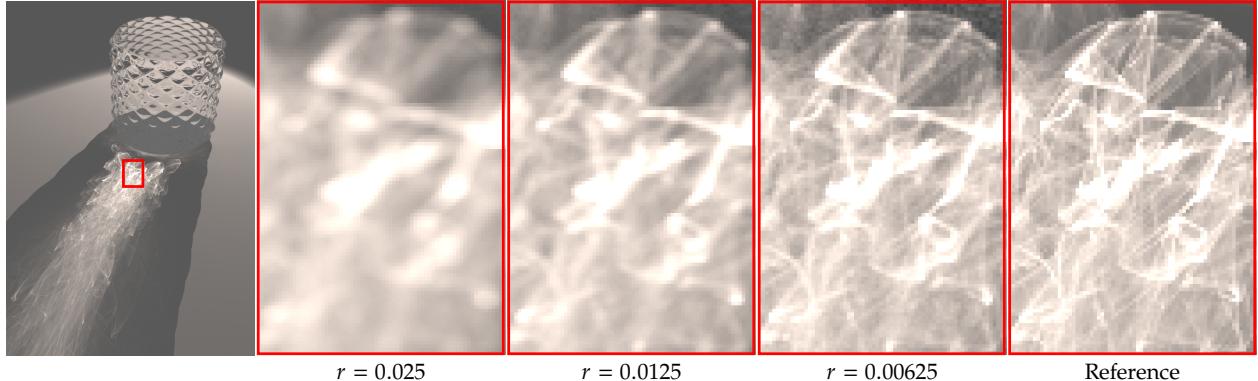


Figure 5.8: Photon mapping is a biased algorithm whose bias depends on the choice of a kernel radius. The larger the radius, the more blurred the image will appear. The amount of bias progressively shrinks as the radius decreases, which is the insight that led to the development of progressive photon mapping.

Progressive photon mapping (PPM) [89, 90] is a technique which takes advantage of this by *progressively* reducing the radius every time photon mapping is ran. By considering the result of invoking photon mapping using a given radius r as a primary Monte Carlo estimator $\langle I(r) \rangle_{\text{PM}}$, progressive photon mapping essentially becomes a secondary Monte Carlo estimator,

$$\langle I(r, n) \rangle_{\text{PPM}} := \frac{1}{n} \sum_{j=1}^n \langle I(r(j)) \rangle_{\text{PM}}, \quad (5.17)$$

where the radius used for each primary sample $r(j)$ decreases as a function of the sample index j . As long as $r(j) \rightarrow 0$ as $j \rightarrow \infty$, progressive photon mapping will converge to the true solution in the limit,

$$\lim_{n \rightarrow \infty} \langle I(r, n) \rangle_{\text{PPM}} = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{j=1}^n \langle I(r(j)) \rangle_{\text{PM}} = I. \quad (5.18)$$

However, as the radius decreases it becomes less likely that photons will land within the kernel. Thus, for an equal number of photons n_p , reducing the radius also increases the variance of evaluating photon mapping as a primary estimator. Many prior works [90–92] have previously shown that valid choices for $r(j)$ correspond to,

$$r(j) = \sqrt{j^{\alpha-1}}, \quad (5.19)$$

where $\alpha \in (0, 1)$ is a user defined parameter meant to control the trade off between the convergence rate of the error and the convergence rate of the variance. It has been shown previously that the best choice for α such that the optimal MSE convergence rate is achieved corresponds to setting $\alpha = 2/3$ [91, 92].

Other developments. There have been a plethora of developments within the literature regarding photon mapping and its progressive alternative. Photon mapping has been extended to support participating media which allowed for the first practical rendering of volumetric caustics [93]. Hachisuka

[89]: Hachisuka et al. (2008), ‘Progressive Photon Mapping’

[90]: Knaus et al. (2011), ‘Progressive Photon Mapping’

[90]: Knaus et al. (2011), ‘Progressive Photon Mapping’

[91]: Georgiev et al. (2012), ‘Light Transport Simulation with Vertex Connection and Merging’

[92]: Kaplanyan et al. (2013), ‘Adaptive Progressive Photon Mapping’

[91]: Georgiev et al. (2012), ‘Light Transport Simulation with Vertex Connection and Merging’

[92]: Kaplanyan et al. (2013), ‘Adaptive Progressive Photon Mapping’

[93]: Jensen et al. (1998), ‘Efficient Simulation of Light Transport in Scenes with Participating Media Using Photon Maps’

et al. [94] later extended progressive photon mapping by incorporating local photon statistics to improve the algorithm when rendering other optical effects such as depth of field. Qin et al. [95] would introduce the first unbiased photon mapping algorithm through unbiased photon gathering, which also introduced the first unbiased reciprocal estimator to graphics, however, their technique does not support SDS paths.

Photon mapping was originally introduced as a technique that traced infinitesimal photons through a scene, however, treating photons as higher dimensional primitives has been shown to result in better performing techniques. Jarosz et al. [96] introduced the idea of tracing photon beams instead of points and later introduced a progressive version of this technique [97]. Bitterli et al. [98] extended this technique to planes and volumes, Deng et al. [99] later generalized the technique to high dimensional photon surfaces, and Liu et al. [100] extended photon primitives to time-of-flight rendering.

5.4 Volumetric path tracing

Since there are various ways of representing the underlying theory of volumetric transport (see Chapter 4), there are subsequently a variety of different ways of formulating Monte Carlo estimators. In this section we will derive unidirectional volumetric path tracing for the VRE in Sec. 5.4.1 and the null-VRE in Sec. 5.4.5, then we finally discuss the non-exponential path integral in Sec. 5.4.6.

While the VRE and null-VRE estimate the same quantity, we will still derive Monte Carlo estimators for both formulations to highlight the fact that estimating the *real* transmittance is a common problem within both formulations. Computing the real transmittance is an example of evaluating a function of expectations, and it will serve as a starting point for our analysis of handling non-linearities within light transport in Chapter 6.

5.4.1 VRE

Deriving a unidirectional estimator for the VRE (4.16),

$$\begin{aligned} L_o(\mathbf{x}, \vec{\omega}_o) &:= \int_0^d \rho_f^{(r)}(\mathbf{x}_t) \alpha_a(\mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) dt \\ &\quad + \int_0^d \rho_f^{(r)}(\mathbf{x}_t) \alpha_s(\mathbf{x}_t) L_m(\mathbf{x}_t, \vec{\omega}_o) dt \\ &\quad + T_r(\mathbf{x}, \mathbf{x}_d) L_s(\mathbf{x}_d, \vec{\omega}_o), \end{aligned} \quad (5.20)$$

requires that we estimate the integral over emission ($L_e(\mathbf{x}_t, \vec{\omega}_o)$), the integral over in-scattered medium radiance ($L_m(\mathbf{x}_t, \vec{\omega}_o)$), and the transmittance attenuated radiance from the next surface intersection ($L_s(\mathbf{x}_d, \vec{\omega}_o)$). Estimators for Eq. (5.20) follow a sequence of three operations which we will address in reverse order.

[94]: Hachisuka et al. (2009), ‘Stochastic Progressive Photon Mapping’

[95]: Qin et al. (2015), ‘Unbiased Photon Gathering for Light Transport Simulation’

[96]: Jarosz et al. (2008), ‘The Beam Radiance Estimate for Volumetric Photon Mapping’

[97]: Jarosz et al. (2011), ‘Progressive Photon Beams’

[98]: Bitterli et al. (2017), ‘Beyond Points and Beams’

[99]: Deng et al. (2019), ‘Photon Surfaces for Robust, Unbiased Volumetric Density Estimation’

[100]: Liu et al. (2022), ‘Temporally sliced photon primitives for time-of-flight rendering’

Scattering. Equation (5.20) contains *two* different recursive evaluations occurring at the next surface intersection, $L_s(\mathbf{x}_d, \vec{\omega}_o)$, and at the medium scattering event, $L_m(\mathbf{x}_t, \vec{\omega}_o)$, respectively. Since we have already reviewed surface-only path tracing we assume that we possess an unbiased estimator for the surface rendering equation (5.4). To formulate an unbiased estimator for the in-scattered medium radiance,

$$L_m(\mathbf{x}_t, \vec{\omega}_o) := \int_{\mathbb{S}^2} \rho_p(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) L_i(\mathbf{x}_t, \vec{\omega}_i) d\vec{\omega}_i \quad (5.21)$$

we simply importance sample the integral over the sphere of directions,

$$\langle L_m(\mathbf{x}_t, \vec{\omega}_o) \rangle := \frac{\rho_p(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) \langle L_i(\mathbf{x}_t, \vec{\omega}_i) \rangle}{p(\vec{\omega}_i)}, \quad (5.22)$$

proportionally to the phase function ($p(\vec{\omega}_i) \propto \rho_p(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o)$).

Choose a term. Light will scatter dozens to hundreds of times within high-albedo media, so deciding to branch the path and evaluate *both* recursions tends to be prohibitively expensive. We instead choose to evaluate a medium interaction with probability P_m and the next surface interaction with probability $1 - P_m$. Additionally, given that a medium interaction occurs, we can choose to evaluate emission with probability $P_a = \alpha_a(\mathbf{x}_t)$ and in-scattered medium radiance with probability $P_s = \alpha_s(\mathbf{x}_t)$ since $\alpha_a(\mathbf{x}_t) + \alpha_s(\mathbf{x}_t) = 1$.

Importance sample \mathbf{x}_t . The obvious choice for importance sampling \mathbf{x}_t in Eq. (5.20) is to sample proportionally to the real free-flight distribution, $\rho_f^{(r)}(\mathbf{x}_t)$. While we will discuss specific techniques for doing so in Sec. 5.4.2, let us assume for the sake of deriving a complete estimator that we already possess such a technique. Given Eq. (5.20), a free-flight sampling technique, and our choice of probabilities for evaluating each term, we can formulate *unidirectional volumetric path-tracing* as a Monte Carlo estimator for Eq. (5.20) as,

$$\langle L_o(\mathbf{x}, \vec{\omega}_o) \rangle := \begin{cases} \frac{1}{P_m P_a} \frac{\rho_f^{(r)}(\mathbf{x}_t) \alpha_a(\mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o)}{p(\mathbf{x}_t)} & \text{with prob. } P_m P_a \\ \frac{1}{P_m P_s} \frac{\rho_f^{(r)}(\mathbf{x}_t) \alpha_s(\mathbf{x}_t) \rho_p(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) \langle L_i(\mathbf{x}_t, \vec{\omega}_i) \rangle}{p(\mathbf{x}_t) p(\vec{\omega}_i)} & \text{with prob. } P_m P_s \\ \frac{1}{1-P_m} \langle T_r(\mathbf{x}, \mathbf{x}_d) \rangle \langle L_s(\mathbf{x}_d, \vec{\omega}_o) \rangle & \text{with prob. } 1 - P_m. \end{cases} \quad (5.23)$$

5.4.2 Free-flight distance sampling

The real free-flight distribution is a normalized probability distribution ($\rho_f^{(r)}(\mathbf{x}_t) = p(\mathbf{x}_t)$) but only when considering the domain $(0, \infty)$. However, its normalization constant for any finite distance d will be exactly equal

to its CDF evaluated at \mathbf{x}_d which is $\text{cdf}(\mathbf{x}_d) = 1 - \text{Tr}(\mathbf{x}, \mathbf{x}_d)$ by Eq. (4.10). Thus, an importance sampling strategy that samples $p(\mathbf{x}_t) \propto \rho_f^{(r)}(\mathbf{x}_t)$ will sample distances $t \geq d$ with a probability $P = \text{Tr}(\mathbf{x}, \mathbf{x}_d)$ and distances $t < d$ with probability $P = 1 - \text{Tr}(\mathbf{x}, \mathbf{x}_d)$ which perfectly coincides with the ideal choices for evaluating either the next surface event or a medium event ($P_m = 1 - \text{Tr}(\mathbf{x}, \mathbf{x}_d)$). Importance sampling proportionally to the real free-flight distribution of the medium conveniently simplifies the math involved in computing the final estimate since it will perfectly cancel out with the real transmittance. Meaning, we would not have to explicitly compute $\langle \text{Tr}(\mathbf{x}, \mathbf{x}_d) \rangle$. Unfortunately, importance sampling proportionally to the real free-flight distribution is not a trivial process in the general case.

Homogeneous media. If a medium is homogeneous ($\mu_r(x) = \mu_r$), we can importance sample proportionally to the real free-flight distribution by using the inversion method (Sec. 3.3.3) to compute the inverse transform, $\text{cdf}^-(\xi)$, as,

$$\begin{aligned} \text{cdf}(t) &= 1 - \text{Tr}(\mathbf{x}, \mathbf{x}_t) \\ \xi &= 1 - e^{-\mu_r t} \\ 1 - \xi &= e^{-\mu_r t} \\ \ln(1 - \xi) &= -\mu_r t \\ -\frac{\ln(1 - \xi)}{\mu_r} &= t = \text{cdf}^-(\xi). \end{aligned} \tag{5.24}$$

Heterogeneous media. If $\mu_r(x)$ is spatially varying and *not* analytic, then the inversion method cannot be simplified enough to derive the inverse transform $\text{cdf}^-(\xi)$ since,

$$\ln(1 - \xi) = - \int_0^t \mu_r(s) ds, \tag{5.25}$$

cannot be solved analytically. Importance sampling proportionally to $\rho_f^{(r)}(\mathbf{x}_t)$ becomes an example of a *root-finding problem* in the general case.

Raymarching. Raymarching is a specific root-finding technique which constructs an *approximate*, but analytic, distribution which can be importance sampled [101]. As its name implies, raymarching generally involves marching along a ray at regular intervals to approximate it as a step function,

$$\int_{\Theta} f(x) dx \approx \sum_{j=1}^n f(x_j) \Delta x_j, \tag{5.26}$$

where $f(x_j)$ is the measured height for marching step j and Δx_j is the step width used. While different or adaptive step sizes could be employed, raymarching is typically implemented using regular steps, $\Delta x_j = \Delta$. All

[101]: Perlin et al. (1989), 'Hypertexture'

x_j are sampled randomly using either stratified sampling (3.35) or jittered sampling (3.36) making raymarching a stochastic form of Riemann summations.

Raymarching constructs an approximate free-flight distribution,

$$p(\mathbf{x}_t)_{\text{Ray}} \propto \mu_r(\mathbf{x}_{j(t)}) e^{-\mu_r(\mathbf{x}_{j(t)}) \Delta_{j(t)} - \sum_{k=1}^{j(t)-1} \mu_r(\mathbf{x}_k) \Delta}, \quad (5.27)$$

where $j(t)$ is the marching step that contains the distance t and $\Delta_{j(t)}$ is the fraction of step $j(t)$ covered by distance t , which has a corresponding normalization constant,

$$G = \sum_{j=1}^n \mu_r(\mathbf{x}_j) \Delta e^{-\sum_{k=1}^j \mu_r(\mathbf{x}_k) \Delta}. \quad (5.28)$$

However, Eq. (5.27) is not guaranteed to cancel out exactly with the medium's real free-flight distribution. So if we employ Eq. (5.27) in Eq. (5.23) we will still need to estimate the real transmittance in an unbiased manner to have an unbiased algorithm.

Unfortunately, due to the non-linear property of the expected value (3.13), using raymarching to naively estimate transmittance,

$$E[T(\mathbf{x}, \mathbf{x}_t)] \neq E\left[e^{-\sum_{j=1}^n \frac{\mu_r(\mathbf{x}_j)}{p(\mathbf{x}_j)}}\right], \quad (5.29)$$

results in a biased estimator in the general case. In fact, any naive application of Monte Carlo estimation to evaluate exponential transmittance will be biased. In Chapter 6 we will thoroughly analyze the problem of estimating classical transmittance and show how, by looking at the problem through the lens of deriving consistent formulations, we can derive many different solutions. However, for now let us assume that we need the means of importance sampling proportionally to the real free-flight distribution *without* evaluating transmittance.

Delta tracking. Within the neutron transport literature, Woodcock [63] derived a technique based on random walks to perfectly importance sample proportionally to a medium's real free-flight distribution. This technique involves homogenizing all heterogeneous media to have some constant *total extinction*, $\mu_t = \mu_n(\mathbf{x}) + \mu_r(\mathbf{x})$, by filling the media with fictitious, or "null", density ($\mu_n(\mathbf{x})$). Once the medium is effectively homogeneous we can importance sample the analytic *total free-flight distribution* ($\rho_f^{(t)}(\mathbf{x}_t) = \mu_t e^{-\mu_t t}$) to sample *tentative* free-flights.

[63]: Woodcock et al. (1965), 'Techniques Used in the GEM Code for Monte Carlo Neutronics Calculations in Reactors and Other Systems of Complex Geometry'

After every tentative free-flight a ray will either collide with a null particle with probability $P_n = \frac{\mu_n(\mathbf{x})}{\mu_t}$, or a real particle with probability $P_r = \frac{\mu_r(\mathbf{x})}{\mu_t} = 1 - P_n$. When a null-collision occurs, the ray continues taking tentative free-flights until it collides with a real particle. Once a real collision occurs, light will either be absorbed or scattered. This technique is known as *delta tracking* within the rendering literature.

Woodcock empirically conveyed that delta tracking importance samples from the real free-flight distribution in his original work while Coleman [102] would later mathematically prove this fact. Delta tracking was the first instance of utilizing null extinctions to importance sample scattering locations within participating media. Kutz et al. [103] would later improve upon this process by decomposing a medium into an analytic control medium and heterogeneous residual medium that has to be estimated, which was based on a similar technique proposed by Novak et al. [104] for the problem of transmittance estimation.

Delta tracking was originally formulated as an empirical random-walk process. We will later show in Sec. 5.4.5 that it can be derived through the direct application of Monte Carlo integration to the null-VRE. Delta tracking can also be converted into an unbiased estimator for transmittance, however, we will wait until Chapter 6 to review transmittance estimation.

5.4.3 Next Event Estimation

We can apply NEE to unidirectional volumetric path tracing by tracing shadow connections to compute the direct illumination similarly to the surface-only case (Sec. 5.1.2). We denote the direct illumination through a volume to the next surface as,

$$L_{\text{NEE}}(\mathbf{x}, \vec{\omega}_o) := T_r(\mathbf{x}, \mathbf{x}_d)L_d(\mathbf{x}_d, \vec{\omega}_o) + \int_0^d \rho_f^{(r)}(\mathbf{x}_t)\alpha_a(\mathbf{x}_t)L_e(\mathbf{x}_t, \vec{\omega}_o)dt, \quad (5.30)$$

where $L_d(\mathbf{x}, \vec{\omega}_o)$ is computed by Eq. (5.5). Similarly to the surface-only case, special care has to be taken to avoid double counting of emission. In practice, the simplest implementation samples a random point either along an emissive surface or an emissive volume proportional to the emission. This results in the estimator,

$$\langle L_d(\mathbf{x}, \vec{\omega}_o) \rangle := \begin{cases} \langle T_r(\mathbf{x}, \mathbf{x}_d) \rangle \langle L_e(\mathbf{x}_d, \vec{\omega}_o) \rangle & \mathbf{x}_d \in \mathbb{A} \\ \mu_a(\mathbf{x}_t) \langle T_r(\mathbf{x}, \mathbf{x}_t) \rangle \frac{L_e(\mathbf{x}_t, \vec{\omega}_o)}{p(\mathbf{x}_t)} & \mathbf{x}_t \in \mathbb{V}, \end{cases} \quad (5.31)$$

where we make note that $\rho_f^{(r)}(\mathbf{x}_t)\alpha_a(\mathbf{x}_t) = \mu_a(\mathbf{x}_t)T_r(\mathbf{x}, \mathbf{x}_t)$ and $\langle L_d(\mathbf{x}_d, \vec{\omega}_o) \rangle$ is equivalent to Eq. (5.7) which already contains $p(\mathbf{x}_d)$. We will discuss more advanced line integration techniques for volumetric NEE in Chapter 7.

Equi-angular sampling. While we refer to Pharr's book [18] for explicit examples of emitter sampling, many methods contain singularities due to the geometric term being proportional to an inverse squared distance, i.e. $G \propto \frac{1}{\|\mathbf{x}-\mathbf{x}_s\|}$. Due to these singularities, naive point sampling for NEE (5.31) can result in explosive variance when the last scattering event is in close proximity to sampled emissive sources. This becomes especially problematic when emissive sources are located inside highly scattering volumes (see Fig. 5.9) and can even result in lower convergence rates.

[102]: Coleman (1968), 'Mathematical Verification of a Certain Monte Carlo Sampling Technique and Applications of the Technique to Radiation Transport Problems'

[103]: Kutz et al. (2017), 'Spectral and Decomposition Tracking for Rendering Heterogeneous Volumes'

[104]: Novák et al. (2014), 'Residual Ratio Tracking for Estimating Attenuation in Participating Media'

[18]: Pharr et al. (2016), *Physically Based Rendering*

[105]: Kulla et al. (2012), 'Importance Sampling Techniques for Path Tracing in Participating Media'

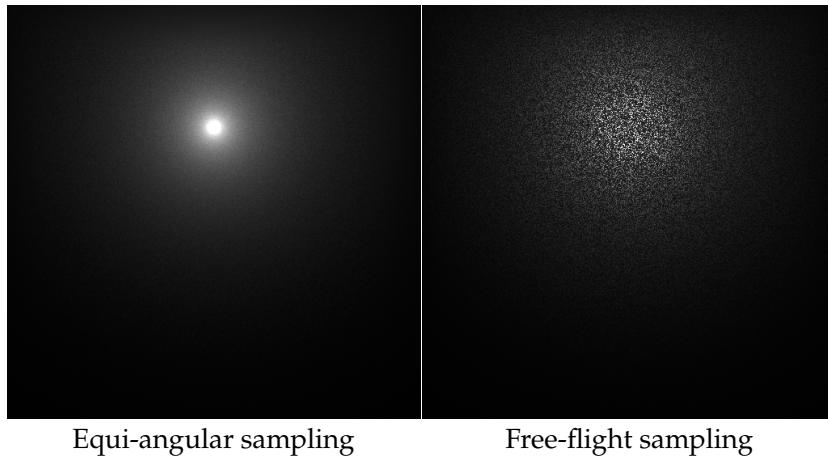


Figure 5.9: A visualization of equi-angular sampling (left) versus free-flight distance sampling (right) in a homogeneous medium containing a point light source. Images were generated using 16 pixel samples.

Equi-angular sampling [105] was introduced to alleviate these drawbacks as an alternative to free-flight distance sampling routines like delta tracing. Instead of importance sampling scatter locations proportionally to the real free-flight distribution, equi-angular sampling first importance samples a point on an emissive source and then importance samples a scattering location proportional to the inverse squared distance,

$$p(\mathbf{x}_t) = \frac{D}{(\theta_b - \theta_a)(D^2 + t^2)}, \quad (5.32)$$

where D is the shortest distance from the emitter to the path segment and θ_a, θ_b are the angles to the start (\mathbf{x}) and end (\mathbf{x}_d) points of the integration bounds, from the perspective of the point on the emissive source, \mathbf{z} .

This technique is known as equi-angular sampling because after applying the inversion method, it involves sampling an angle by linearly interpolating between the angles $[\theta_a, \theta_b]$. By sampling proportionally to the singularity, it is typically more useful for importance sampling direct illumination in the presence of emissive sources within participating media as opposed to free-flight distance sampling, and it does not suffer from the convergence rate loss (see Fig. 5.9). While we will discuss an application which employs this technique in Chapter 7, for nearly all volumetric rendering examples in this dissertation we choose to importance sample distances proportionally to the free-flight distribution using delta tracking.

5.4.4 Aside: Emission

In practice, the extinction within the emissive portion of a medium, such as fire or an explosion, tends to be significantly higher than the rest of the medium. Additionally, the accumulation of the emission tends to have a significant impact on the performance of a render. This is why production renderers, such as Walt Disney's Hyperion renderer [83, 106], tend to treat the accumulation of emission within volumes as a special case by employing much more conservative free-flight distance sampling. They do so through specifying a significantly large μ_t (small tentative flights) to maintain low variance renders through local regions of the medium.

[83]: Burley et al. (2018), 'The Design and Evolution of Disney's Hyperion Renderer'
[106]: Huang et al. (2021), 'Unbiased Emission and Scattering Importance Sampling For Heterogeneous Volumes'

5.4.5 Null VRE

Delta tracking was the first technique to introduce the concept of null-scattering which led to an explosion of developments in both importance sampling free-flight distributions and the estimation of transmittance. Miller et al. [51] later introduced a null-scattering reformulation of the volumetric rendering equation (see Sec. 4.3.1) allowing for free-flight importance sampling techniques, which were initially introduced as random walks, to be directly formulated as Monte Carlo estimators.

[51]: Miller et al. (2019), 'A Null-Scattering Path Integral Formulation of Light Transport'

Free-flight distance sampling. Starting from the null-VRE (4.38),

$$\begin{aligned} L_o(\mathbf{x}, \vec{\omega}_o) := & \int_0^d \rho_f^{(t)}(\mathbf{x}_t) \alpha_a(\mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) dt \\ & + \int_0^d \rho_f^{(t)}(\mathbf{x}_t) \alpha_s(\mathbf{x}_t) L_m(\mathbf{x}_t, \vec{\omega}_o) dt \\ & + \int_0^d \rho_f^{(t)}(\mathbf{x}_t) \alpha_n(\mathbf{x}_t) L_o(\mathbf{x}_t, \vec{\omega}_o) dt \\ & + T_t(\mathbf{x}, \mathbf{x}_d) L_s(\mathbf{x}_d, \vec{\omega}_o), \end{aligned} \quad (5.33)$$

we can apply the same steps that we applied to the VRE in Sec. 5.4.1 without much modification to arrive at an estimator for unidirectional volumetric path tracing,

$$\langle L_o(\mathbf{x}, \vec{\omega}_o) \rangle = \begin{cases} \frac{1}{P_m P_a} \frac{\rho_f^{(t)}(\mathbf{x}_t) \alpha_a(\mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o)}{p(\mathbf{x}_t)} & \text{with prob. } P_m P_a \\ \frac{1}{P_m P_s} \frac{\rho_f^{(t)}(\mathbf{x}_t) \alpha_s(\mathbf{x}_t) \rho_p(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) \langle L_i(\mathbf{x}_t, \vec{\omega}_i) \rangle}{p(\mathbf{x}_t) p(\vec{\omega}_i)} & \text{with prob. } P_m P_s \\ \frac{1}{P_m P_n} \frac{\rho_f^{(t)}(\mathbf{x}_t) \alpha_n(\mathbf{x}_t) \langle L_o(\mathbf{x}_t, \vec{\omega}_o) \rangle}{p(\mathbf{x}_t)} & \text{with prob. } P_m P_n \\ \frac{1}{1-P_m} T_t(\mathbf{x}, \mathbf{x}_d) \langle L_s(\mathbf{x}_d, \vec{\omega}_o) \rangle & \text{with prob. } 1 - P_m. \end{cases} \quad (5.34)$$

Since the total free-flight distribution ($\rho_f^{(t)}$) is analytic, we can easily importance sample from it by using the inversion method (5.24). Similarly to the VRE, the resulting normalization constant perfectly cancels with $\frac{1}{P_m}$ and $1 - P_m = T_t(\mathbf{x}, \mathbf{x}_d)$. Thus, most free-flight distance sampling routines differ by their relative choices for P_a , P_s , and P_n .

Delta tracking. As a reminder, the albedos of a participating medium ($\alpha_*(\mathbf{x})$) are always specified with respect to the combined extinction of the medium. When considering the traditional VRE, the combined extinction is equal to the real extinction ($\alpha_*(\mathbf{x}) = \frac{\mu_r(\mathbf{x})}{\mu_r(\mathbf{x})}$) while for the null-VRE, the albedos are defined with respect to the total extinction ($\alpha_*(\mathbf{x}) = \frac{\mu_t(\mathbf{x})}{\mu_t}$), where $\mu_t = \mu_r(\mathbf{x}) + \mu_n(\mathbf{x})$.

Delta tracking [63] was originally introduced as a technique for importance sampling the real free-flight distribution through evaluating an entire

[63]: Woodcock et al. (1965), 'Techniques Used in the GEM Code for Monte Carlo Neutronics Calculations in Reactors and Other Systems of Complex Geometry'

random walk process. Under the null-scattering VRE, delta tracking can be implemented directly as a Monte Carlo estimator for Eq. (5.34) by setting the probabilities of evaluating emission, scattering, or null-scattering, proportionally to their relative albedos in the combined medium,

$$P_* = \frac{\mu_*(\mathbf{x})}{\mu_t} = \alpha_*(\mathbf{x}). \quad (5.35)$$

Weighted delta tracking. Delta tracking is unbiased as long as $P_a + P_s + P_n = 1$. However, when the total extinction is non-bounding ($\mu_t < \mu_r(\mathbf{x})$), naively setting the probabilities equal to the absolute value of the albedos does not guarantee unbiasedness, i.e. $|\alpha_a(\mathbf{x})| + |\alpha_s(\mathbf{x})| + |\alpha_n(\mathbf{x})| \neq 1$.

Weighted delta tracking [107, 108] alleviates this by instead setting the relative probabilities to,

$$P_* = \frac{\mu_*(\mathbf{x})}{\mu_r(\mathbf{x}) + |\mu_n(\mathbf{x})|}. \quad (5.36)$$

When $\mu_t \geq \mu_r(\mathbf{x})$, the denominator simplifies to the total extinction and weighted delta tracking becomes equivalent to delta tracking (5.35).

[107]: Cramer (1978), ‘Application of the Fictitious Scattering Radiation Transport Model for Deep-Penetration Monte Carlo Calculations’

[108]: Galtier et al. (2013), ‘Integral Formulation of Null-Collision Monte Carlo Algorithms’

Decomposition tracking. Decomposition tracking [103] uses an optimization technique introduced by Novak et al. [104] which involves *decomposing* a medium into an analytic control medium and a residual medium using control variates. The control medium is assumed to be homogeneous, *real*, and can be sampled from analytically, while the residual medium has to employ delta tracking or weighted delta tracking with a modulated extinction. We will review this application of control variates more thoroughly when we review the ratio tracking transmittance estimator in Chapter 6.

Since decomposition tracking involves tracking through the control and the residual medium separately, it relies on the mathematical property that the free-flight distribution of the product of two exponential transmittances with different extinction functions is equal to the CDF of their minimums [103]. Meaning, every time decomposition tracking samples a tentative free-flight distance, it samples one from the control medium and one from the residual medium then takes the minimum of the two.

[103]: Kutz et al. (2017), ‘Spectral and Decomposition Tracking for Rendering Heterogeneous Volumes’

[104]: Novák et al. (2014), ‘Residual Ratio Tracking for Estimating Attenuation in Participating Media’

[103]: Kutz et al. (2017), ‘Spectral and Decomposition Tracking for Rendering Heterogeneous Volumes’

Spectral tracking. Spectral tracking is an alternative form of decomposition tracking for rendering participating media whose real and total extinction is spectrally varying. To simplify the implementations and algorithms for the work in this dissertation, we assume that all scattering media is monochromatic. We refer the interested reader to [103] for an overview of how to handle spectral media.

[103]: Kutz et al. (2017), ‘Spectral and Decomposition Tracking for Rendering Heterogeneous Volumes’

MIS. There are two major practical benefits of the null-scattering formulation. First, by allowing all free-flight distance sampling techniques to be derived as Monte Carlo estimators instead of empirical random walk processes, we have a mathematical guarantee for the convergence of these

algorithms instead of relying on assumptions based on empirical results. Additionally, adapting the null-VRE into the null-scattering path integral allows for null-scattering based free-flight importance sampling routines to be placed in the same measure space with emitter sampling routines such as equi-angular sampling. This enables them to be combined through the use of MIS to improve volumetric rendering algorithms [51].

[51]: Miller et al. (2019), 'A Null-Scattering Path Integral Formulation of Light Transport'

NEE. In scenes containing only emissive surfaces and no emissive media, the direct illumination to the next surface in the VRE case (5.30) would just be the transmittance modulated surface emission. However, for the case of the null-VRE, we have to account for null-scattering. Thus, the direct illumination to the next emissive surface point would be computed as,

$$L_{\text{NEE}}(\mathbf{x}, \vec{\omega}_o) := T_t(\mathbf{x}, \mathbf{x}_d)L_e(\mathbf{x}_d, \vec{\omega}_o) + \int_0^d \rho_f^{(t)}(\mathbf{x}_t)\alpha_n(\mathbf{x}_t)L_{\text{NEE}}(\mathbf{x}_t, \vec{\omega}_o) dt. \quad (5.37)$$

While all $T_t(\mathbf{x}, \mathbf{x}_t)$ evaluations are analytic, we will later show in ?? that Eq. (5.37) is exactly equal to the *real* transmittance multiplied by the surface emission.

The null-scattering formulation implicitly forces the use of forward null-scattering techniques, however, in Chapter 6 and Chapter 8 we will introduce a variety of alternative formulations for transmittance which can result in more efficient transmittance estimators. Thus, when employing NEE in the null-VRE, it is occasionally more beneficial to replace Eq. (5.37) with the real transmittance.

5.4.6 Non-Classical Transport

The non-classical model we rely on for non-exponential volumetric rendering (see Sec. 4.4) differs from the classical formulation in that instead of having a transmittance term, we have to evaluate a transport function. The transport function contains three different transmittances that are invoked depending on the current ray's traversal history. These transmittances can easily be converted into free-flight sampling routines which can be approximately importance sampled using raymarching. However, before the contributions of this dissertation, the only way of estimating non-exponential transmittance in the general case was through biased raymarching.

In Chapter 8 we will introduce an unbiased version of raymarching which can estimate many *general* transmittance functions in an unbiased manner. We will show later that this will allow for the unbiased rendering of general non-classical media.



6 Integral formulations for exponential transmittance

Modern Monte Carlo based rendering algorithms render participating media by stochastically tracing light paths throughout a virtual environment. In Chapter 4, we reviewed three different theoretical models for simulating the interactions light can have with participating media through absorption, scattering, and emission. Two of those models were based on simulating the transport of light within classical media (Secs. 4.2.3 and 4.3) where light attenuates exponentially, and the other model was based on computing non-classical light transport (Sec. 4.4). In Sec. 5.4 we reviewed how Monte Carlo is used to formulate rendering algorithms for each of these models, and we observed how *estimating* the *attenuation* of light along a path segment is a common problem which arises in all three of these formulations. We refer to this attenuation as the *real* transmittance, $T_r(x, x_d)$. Unfortunately, estimating this quantity in a consistent manner turns out to be a *non-trivial* problem.

Evaluating transmittance is an example of a situation where we want to evaluate an integral which is modified by some *non-linear* function,

$$I(a, b) := g \left(\int_a^b f(t) dt \right). \quad (6.1)$$

For classical media, the non-linear function takes the form of a reciprocal exponential, $g(X) = e^{-X}$, while for non-classical media $g(X)$ can take the form of any function which meets the constraints outlined by Bitterli et al. [4].

Within the probabilistic literature [109], the general problem of estimating Eq. (6.1) is known as estimating *non-linear functions of expectations*. This is because the process of estimating the integral can be more generally thought of as replacing the integral with a random variable X whose mean, variance, and distribution are equivalent to the mean, variance, and distribution of an estimator for the integral. Unfortunately, replacing the

6.1	Background	78
6.1.1	Prior unbiased solutions	78
6.2	Volterra integral formulation	83
6.2.1	RTE based derivation	83
6.2.2	IVP derivation	84
6.2.3	Null-collision formulation	85
6.2.4	Re-deriving prior estimators	86
6.2.5	Variance analysis	88
6.2.6	Adaptive importance sampled estimators	93
6.2.7	Relating Null-VRE to VRE using the Volterra .	101
6.3	Power-series integral formulation	103
6.3.1	Base formulations	104
6.3.2	Null-collision formulations	105
6.3.3	Re-deriving ratio tracking	106
6.3.4	Novel transmittance estimators	108
6.4	Hypocube formulation	111
6.4.1	Analogy to rendering formulations	111
6.4.2	Base formulation	112
6.4.3	Null formulation	113
6.4.4	Hypocube estimation	114
6.5	Discussion, results, and implementation	116
6.5.1	Summary of contributions	116
6.5.2	Numerical experiments	117
6.5.3	Discussion	123

integral in Eq. (6.1) with a primary Monte Carlo estimator,

$$\langle I(a, b) \rangle := g \left(\frac{f(t)}{p(t)} \right), \quad (6.2)$$

will generally result in *incorrect* solutions for *most* non-linear functions g ,

$$E \left[g \left(\frac{f(t)}{p(t)} \right) \right] \neq g \left(E \left[\frac{f(t)}{p(t)} \right] \right), \quad (6.3)$$

as long as the primary estimate for the integral has non-zero variance¹. While Jensen's inequality (3.14) fundamentally conveys Eq. (6.3), we additionally provide empirical evidence in Fig. 6.1 where we exhibit the bias of naively estimating classical transmittance for a simple linearly increasing extinction function ($\mu_r(x_t) = 2t$).

Non-linear perturbations induced by non-linear functions will generally distort the results of Monte Carlo estimators, which becomes problematic when we want *consistent* solutions. This dissertation aims to develop a general roadmap (Fig. 1.1) for tackling these difficult non-linear estimation problems head on. Within this chapter, we focus on introducing solutions which become amenable to unbiased estimation by *transforming* the underlying problem into a *consistent formulation* (①) that iteratively computes *linear* terms. This effectively removes the need of estimating any non-linear functions while formulating unbiased transmittance estimators.

We define a *consistent formulation* as one which will result in the true solution in some general limit. Given a reformulation of $I(a, b)$ which depends on some parameter n as $I_n(a, b)$, we would expect that in the limit,

$$I(a, b) = \lim_{n \rightarrow \infty} I_n(a, b), \quad (6.4)$$

we would arrive at the true solution. Once we have a consistent formulation represented as a limit, we can apply Monte Carlo integration to directly estimate Eq. (6.4). However, the exact means by which we accomplish this will depend on the specific construction of the consistent formulation (①).

For the problem of classical transmittance estimation, we propose two different consistent formulations which are amenable to unbiased estimation. The first one is based on converting the definition of transmittance into an *initial value problem* (IVP) which integrates over how g changes over distance. The second formulation is based on the Taylor series expansion of g . For the majority of this chapter, we will mostly focus on analyzing these formulations for the specific problem of estimating classical exponential transmittance. Later in Chapter 8 we will review how our formulations can be applied to estimate general non-linear functions and non-exponential media. In doing so, we will also propose a theoretical null-scattering form of Bitterli et al.'s [4] non-exponential rendering equation.

In the remainder of this chapter we will first review all necessary background information in Sec. 6.1. We will then introduce our Volterra integral formulation for exponential transmittance in Sec. 6.2 which is derived by

1: If we possess a PDF which perfectly importance samples $p(t) \propto f(t)$, then $\langle I(a, b) \rangle$ will be unbiased (and have zero-variance). However, these ideal cases rarely occur in practice.

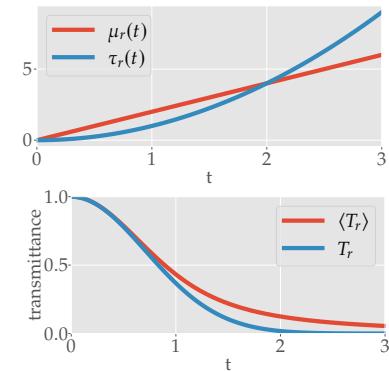


Figure 6.1: A graphical example of the bias which arises when naively estimating classical transmittance with Monte Carlo integration. Given a linearly increasing real extinction (top, red) with corresponding optical depth (top, blue), we plot the ground truth transmittance (bottom, blue) compared to the expected value of naively estimating transmittance using Eq. (6.2) (bottom, red). The expected value is computed as the sample mean of 2^{20} invocations of Eq. (6.2).

[4]: Bitterli et al. (2018), 'A Radiative Transfer Framework for Non-Exponential Media'

converting classical transmittance into an IVP (1). We then derive all prior transmittance estimators as direct Monte Carlo estimators of the Volterra formulation in Sec. 6.2.4, and we devise principled improvements to these estimators in Sec. 6.2.6 based on zero-variance analysis (2). In Sec. 6.3 we introduce the Taylor series integral formulation for classical transmittance as well as introduce a variety of novel transmittance estimators (3). In Sec. 6.5, we review both practical and canonical results of our proposed transmittance estimators. We conclude this chapter in ?? where we discuss a relevant extension to this work completed by other researchers for exponential transmittance, which we adapt for general non-linear functions. Later in Chapter 8 we will discuss generalized forms for both the Taylor series and IVP formulations for transmittance.

6.1 Background

The majority of this chapter deals with the problem of formulating unbiased estimators for transmittance within a classical participating medium. Under classical assumptions, light is expected to attenuate according to,

$$T_r(x_a, x_b) := e^{-\int_a^b \mu_r(x_t) dt}, \quad (6.5)$$

where the rate of attenuation is dictated by the real extinction, $\mu_r(x)$, between any two endpoints (x_a, x_b) , a distance $d = b - a$ apart. For the remainder of this chapter, when we generally refer to transmittance we are specifically referring to exponential transmittance (6.5). We will discuss the more general non-exponential case in Chapter 8.

6.1.1 Prior unbiased solutions

Prior to the introduction of this work [110], all unbiased methods for estimating transmittance within the rendering literature were based on random-walk theory. These random walks relied on the introduction of null, or fictitious particles to augment the medium in such a way that random-walks could be sampled. In the remainder of this section, we will briefly summarize all of the estimators which have appeared within the graphics literature prior to the publication of this work [110]. We additionally provide pseudocode outlining how these estimators can be implemented in practice.

[110]: Georgiev et al. (2019), ‘Integral Formulations of Volumetric Transmittance’

Binary track length estimator. By definition, the real transmittance is a CCDF (4.10) and its corresponding PDF is the real free-flight distribution (4.15). Given an importance sampling routine which perfectly importance samples proportionally to the real free-flight distribution, the probability of importance sample a distance past the endpoint (x_b) ,

$$P(\text{cdf}^-(\xi) > b) = T_r(x, x_b), \quad (6.6)$$

is, by definition, *exactly* equal to transmittance.

If a participating medium is homogeneous, then we can formulate an unbiased estimate for transmittance by employing the inversion method to importance sample free-flight distances (5.24) and then test to see if the sampled distance surpasses the endpoint. This results in the *binary track length estimator*,

$$\langle T_r(x_a, x_b) \rangle_{\text{BinTL}} := \begin{cases} 1, & t > b \\ 0, & t \leq b, \end{cases} \quad (6.7)$$

which effectively performs *rejection sampling* based on whether a *real* interaction occurs before the end of the path segment. We refer to this estimator as a binary track length estimator since it returns only a binary estimate for transmittance while testing sampled distances against the length of a tracking.

Transmittance is analytic within homogeneous media so employing Eq. (6.7) in practice is unnecessary, however, we can derive useful unbiased techniques for the non-trivial heterogeneous case (5.25) by combining Eq. (6.7) with the idea of null-scattering.

Null-scattering track length estimator. Delta tracking (Sec. 5.4.2) was introduced within the neutron transport literature [63] as a technique for perfectly importance sampling proportionally to the real free-flight distribution within heterogeneous media. By effectively homogenizing the real extinction with a fictitious extinction ($\mu_t = \mu_r(x) + \mu_n(x)$), delta tracking importance samples *tentative* free-flight distances proportional to the *total* free-flight distribution,

$$\rho_f^{(t)}(x_t) := \mu_t e^{-\mu_t t}. \quad (6.8)$$

After sampling a tentative free-flight, delta tracking checks to see if a *real* collision occurs using Russian Roulette termination with probability $P_r = \frac{\mu_r(x)}{\mu_t}$ to determine if a medium interaction should be evaluated or if another tentative free-flight should be sampled.

By design, delta tracking samples medium interactions such that the probability of one occurring past the end of the current path segment is exactly equal to the real transmittance (6.6). Thus, we can convert delta tracking into a *null-scattering track length estimator*,

$$\langle T_r(x_a, x_b) \rangle_{\text{TL}} := \begin{cases} 1, & t > b \\ 0, & \text{with prob. } P_r \\ \langle T_r(x_t, x_b) \rangle_{\text{TL}}, & \text{otherwise,} \end{cases} \quad (6.9)$$

which continuously samples free-flight distances until either a real collision occurs or the sampled free-flight surpasses the endpoint. Colemen et al. [102] proved that Eq. (6.9) results in the correct expected transmittance, and we provide pseudocode for this technique in Algorithm 1. From now on, when we generally refer to a track length estimator, we are specifically referring to Eq. (6.9).

[63]: Woodcock et al. (1965), 'Techniques Used in the GEM Code for Monte Carlo Neutronics Calculations in Reactors and Other Systems of Complex Geometry'

[102]: Coleman (1968), 'Mathematical Verification of a Certain Monte Carlo Sampling Technique and Applications of the Technique to Radiation Transport Problems'

Algorithm 1: Null-scattering track length transmittance estimator

```

1  $t = a$ 
2 while  $t < b$  do
3    $t = t - \frac{\log(1-\xi)}{\mu_t}$ 
4   if  $t \geq b$  then
5     break
6   if  $\frac{\mu_r(x_t)}{\mu_t} > \xi$  then
7     return 0
8 return 1

```

Weighted track length estimator. The track length estimator (6.9) will be unbiased as long as the probability of a real collision occurring remains a valid probability, i.e. $P_r \in [0, 1]$. If the user-specified total extinction is an *upperbound* on $\mu_r(x)$ for all points within the medium then $P_r = \frac{\mu_r(x)}{\mu_t}$ will always be a valid probability, thus, Eq. (6.9) will always be unbiased.

Unfortunately, acquiring a bounding total extinction can become a non-trivial problem¹ which can lead to bias if Eq. (6.9) is employed to estimate transmittance. To maintain an unbiased solution, we can formulate an analogous transmittance estimator to weighted delta tracking (5.36) known as *weighted track length estimation* [107, 108]. This estimator can be recursively described as,

$$\langle T_r(x_a, x_b) \rangle_{WTL} := \begin{cases} 1, & t > b \\ 0, & \text{with prob. } P_r^* \\ W(x_t) \langle T_r(x_t, x_b) \rangle_{WTL}, & \text{otherwise,} \end{cases} \quad (6.10)$$

where $W(x) = \frac{1-P_r}{1-P_r^*}$ is the weight which accounts for the modified real probability $P_r^* = \frac{\mu_r(x)}{\mu_r(x) + |\mu_n(x)|}$. The weight simplifies to

$$W(x) := S(x) \frac{\mu_r(x) + |\mu_n(x)|}{\mu_t}, \quad (6.11)$$

where $S(x) := \frac{|\mu_n(x)|}{\mu_n(x)}$ keeps track of the sign. If μ_t is always bounding, Eq. (6.10) will simplify to Eq. (6.9) and return a binary score, otherwise, it will return a running product of weights. We provide pseudocode for this estimator in Algorithm 2.

Ratio tracking. While the track length estimators are unbiased, their variance tends to be high since they often return binary estimates for transmittance which is a fractional scalar value (see Fig. 6.2). To improve the variance of these random walk processes, Novak et al. [104] introduced a technique known as *ratio tracking*.

1: Chapter 7 is dedicated to analyzing and handling this exact problem.

[107]: Cramer (1978), ‘Application of the Fictitious Scattering Radiation Transport Model for Deep-Penetration Monte Carlo Calculations’

[108]: Galtier et al. (2013), ‘Integral Formulation of Null-Collision Monte Carlo Algorithms’

[104]: Novák et al. (2014), ‘Residual Ratio Tracking for Estimating Attenuation in Participating Media’

Algorithm 2: Weighted track length transmittance estimator

```

1  $t = a$ 
2  $T = 1$ 
3 while  $t < b$  do
4    $t = t - \frac{\log(1-\xi)}{\mu_t}$ 
5   if  $t \geq b$  then
6     break
7    $P = \frac{\mu_n(x_t)}{|\mu_n(x_t)| + \mu_r(x_t)}$ 
8   if  $P > \xi$  then
9     return 0
10   $T = T * \frac{\mu_n(x_t)}{\mu_t * (1-P)}$ 
11 return  $T$ 

```

Track length estimators perform Russian Roulette after every tentative free-flight to terminate when a real collision occurs, while ratio tracking instead evaluates the *ratio* of the null extinction to total extinction after every tentative free-flight then continues tracking. This results in an estimator,

$$\langle T_r(x_a, x_b) \rangle_{\text{Rat}} := \frac{\mu_n(x_t)}{\mu_t} \langle T_r(x_t, x_b) \rangle_{\text{Rat}} = \prod_{j=1}^{n-1} \alpha_n(x_j), \quad (6.12)$$

which can be written succinctly as the running product of the null albedo where we use x_j to refer to the location of the j th tentative free-flight assuming there will be a total of n free-flights (the n th free-flight will land past b). We provide pseudocode for this estimator in Algorithm 3 as well as a visualization of the behaviors of ratio and track length estimation in Fig. 6.2.

Algorithm 3: Ratio tracking transmittance estimator

```

1  $t = a$ 
2  $T = 1$ 
3 while  $t < b$  do
4    $t = t - \frac{\log(1-\xi)}{\mu_t}$ 
5   if  $t \geq b$  then
6     return  $T$ 
7    $T = T * \frac{\mu_n(x_t)}{\mu_t}$ 
8 return  $T$ 

```

Residual ratio tracking. Residual ratio tracking was additionally introduced alongside ratio tracking [104] as a variance reduction technique based on the multiplicative property of the exponential function,

$$e^{a+b} = e^a e^b. \quad (6.13)$$

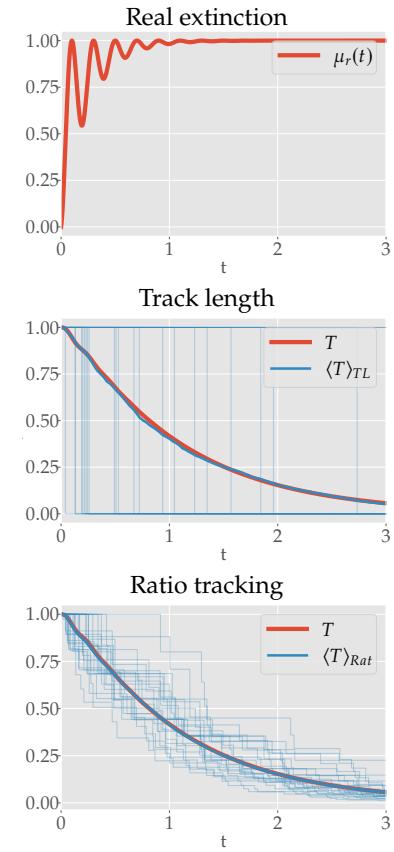


Figure 6.2: We visualize the behaviors of track length estimation (6.9, middle) and ratio tracking (6.12, bottom) for a pre-defined extinction function (top). We plot the average of 1000 trackings for each estimator while also including the first 25 trackings to visualize the intermediate behaviour of each estimator.

[104]: Novák et al. (2014), ‘Residual Ratio Tracking for Estimating Attenuation in Participating Media’

In practice, some volumetric extinctions show small amounts of variation compared to the actual magnitude of the extinction function. For these cases, the real extinction can be decomposed into the sum of a constant density homogeneous "control" extinction, μ_r^c , and a heterogeneous "residual" extinction, $\mu_r^r(x_t) := \mu_r(x_t) - \mu_r^c$ which we visualize in Fig. 6.3. Due to Eq. (6.13), we can decompose exponential transmittance (6.5) into the product of two separate transmittances,

$$\begin{aligned} T_r(x_a, x_b) &= T_r^c(x_a, x_b) T_r^r(x_a, x_b) = e^{-(b-a)\mu_r^c} e^{-\int_a^b \mu_r^r(x_t) dt} \\ \langle T_r(x_a, x_b) \rangle_{\text{ResRat}} &:= T_r^c(x_a, x_b) \langle T_r^r(x_a, x_b) \rangle_{\text{Rat}} = e^{-(b-a)\mu_r^c} \left(\prod_{j=1}^{n-1} \frac{\mu_r^r(x_j)}{\mu_r^r(t)} \right), \end{aligned} \quad (6.14)$$

where the *control* transmittance ($T_r^c(x_a, x_b)$) is evaluated analytically, while the *residual* transmittance ($T_r^r(x_a, x_b)$) is estimated using ratio tracking with a smaller total extinction ($\mu_r^r := \mu_t - \mu_r^c$) compared to typical ratio tracking Eq. (6.12).

Since the magnitude of μ_t directly controls the number of steps used in ratio-tracking, the main benefit of Eq. (6.14) arises from its potential reduction in cost given that $\mu_r^r \leq \mu_t$. Employing residual ratio tracking in practice requires precomputing μ_r^c , and its improvement over ratio tracking will depend on how well μ_r^c matches μ_r . While we only consider a constant μ_r^c in Eq. (6.14), the control extinction can be spatially varying as long as it is analytic for any (x_a, x_b) .

Next-flight ratio tracking. Ratio tracking effectively estimates transmittance by attenuating the transport evaluated at x_b through randomly sampled intermediate points within the medium. We can alternatively choose to accumulate the uncollided transport $e^{-(b-t)\mu_t}$ as a form of NEE to estimate transmittance rather than the attenuated final transport. We do so by recursively computing these *next-flight* estimates,

$$\begin{aligned} \langle T_r(x_a, x_b) \rangle_{\text{NF}} &:= T_t(x_a, x_b) + \alpha_n(x_t) \langle T_r(x_t, x_b) \rangle_{\text{NF}} \\ &= T_t(x_a, x_b) + \sum_{j=1}^{n-1} T_t(x_j, x_b) \prod_{k=1}^j \alpha_n(x_k), \end{aligned} \quad (6.15)$$

to arrive at the *next-flight ratio tracking estimator*. This technique was first introduced by Cramer [107] within the nuclear engineering literature then was later incorporated into graphics [103, 111]. We provide pseudocode for this algorithm in Algorithm 4.

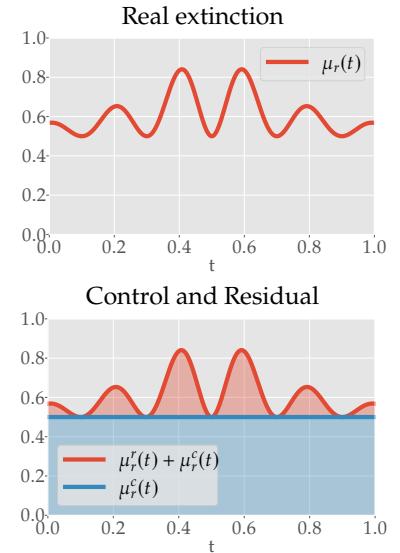


Figure 6.3: We visualize the decomposition of a real extinction function (top) into an analytic control extinction (bottom, blue) and residual extinction (bottom, red), which has to be estimated.

[107]: Cramer (1978), ‘Application of the Fictitious Scattering Radiation Transport Model for Deep-Penetration Monte Carlo Calculations’

[103]: Kutz et al. (2017), ‘Spectral and Decomposition Tracking for Rendering Heterogeneous Volumes’

[111]: Novák et al. (2018), ‘Monte Carlo Methods for Physically Based Volume Rendering’

Algorithm 4: Next-flight ratio tracking transmittance estimator

```

1  $t = a$ 
2  $T = 0$ 
3  $W = 1$ 
4 while  $t < b$  do
5    $T = T + W * e^{-\mu_t(b-t)}$ 
6    $t = t - \frac{\log(1-\xi)}{\mu_t}$ 
7    $W = W * \frac{\mu_n(x_t)}{\mu_t}$ 
8 return  $T$ 

```

6.2 Volterra integral formulation

We refer to our first integral formulation for exponential transmittance as the *Volterra integral formulation* based on the resulting recursive form of its integral. This formulation was originally derived from a reduced form of the RTE [110] which we will replicate in Sec. 6.2.1. However, we later discovered [112] that there is a much more general way of deriving the Volterra formulation through reformulating exponential transmittance as an Initial Value Problem (IVP).

The IVP is more useful since it can be applied to many other non-exponential formulations for transmittance without the need of specifying a more generalized RTE. Re-deriving these formulations from first principles becomes especially cumbersome for non-exponential transmittances due to the fact that deriving useful generalized versions of the RTE is an active area of research. We will review the IVP derivation in Sec. 6.2.2 and later discuss its application to non-exponential transmittance formulations in Chapter 8.

We extend this formulation using the null-collision paradigm in Sec. 6.2.3, re-derive all prior transmittance estimators in Sec. 6.2.4, perform a zero-variance analysis of former transmittance estimators in Sec. 6.2.5, and then use that analysis to derive improved versions of prior work in Sec. 6.2.6.

[110]: Georgiev et al. (2019), ‘Integral Formulations of Volumetric Transmittance’
[112]: Misso et al. (2022), ‘Unbiased and consistent rendering using biased estimators’

6.2.1 RTE based derivation

As light travels along a ray through a participating medium, it can collide with particles that either absorb or out-scatter a portion of its energy. If the particles are distributed statistically independently from one another, the differential change in radiance along a differential distance can be computed by a reduced form of the differential radiative transfer equation (RTE) [1] as,

$$\frac{dL(\mathbf{x}_s)}{ds} = -\mu_r(\mathbf{x}_s)L(\mathbf{x}_s), \quad (6.16)$$

[1]: Chandrasekhar (1960), *Radiative Transfer*

which ignores radiance gains from emission or in-scattering. The spatially varying real extinction coefficient, $\mu_r(\mathbf{x}_s)$, determines the probability density for such collisions to occur.

In rendering, we typically simulate light transport in the opposite direction to the natural flow of light (Sec. 5.1), so we impose a change of variable, $t = -s$, and base all further derivations on the following variant of Eq. (6.16),

$$-\frac{dL(\mathbf{x}_t)}{dt} = -\mu_r(\mathbf{x}_t) L(\mathbf{x}_t). \quad (6.17)$$

The Volterra integral formulation arises trivially from Eq. (6.17) by first integrating both sides over the length of the path segment resulting in,

$$\begin{aligned} \int_a^b -\frac{dL(\mathbf{x}_t)}{dt} dt &= \int_a^b -\mu_r(\mathbf{x}_t) L(\mathbf{x}_t) dt \\ L(\mathbf{x}_a) - L(\mathbf{x}_b) &= - \int_a^b \mu_r(\mathbf{x}_t) L(\mathbf{x}_t) dt \\ L(\mathbf{x}_a) &= L(\mathbf{x}_b) - \int_a^b \mu_r(\mathbf{x}_t) L(\mathbf{x}_t) dt. \end{aligned} \quad (6.18)$$

Since the transmittance between \mathbf{x}_a and \mathbf{x}_b is equal to the proportion of radiance which survives, i.e. $T_r(\mathbf{x}_a, \mathbf{x}_b) = \frac{L(\mathbf{x}_a)}{L(\mathbf{x}_b)}$, we divide both sides by $L(\mathbf{x}_b)$,

$$T_r(\mathbf{x}_a, \mathbf{x}_b) := 1 - \int_a^b \mu_r(\mathbf{x}_t) T_r(\mathbf{x}_t, \mathbf{x}_b) dt, \quad (6.19)$$

to arrive at our first formulation for transmittance. Since $T_r(\mathbf{x}_t, \mathbf{x}_b)$ exists within the integrand, Eq. (6.19) is a recursive integral formulation where its specific form is known as a *Volterra* integral.

6.2.2 IVP derivation

The RTE derivation is rooted in first principles given the transport equation for radiance (6.17). However, deriving the necessary radiative transfer theory for any other specific attenuation model is a cumbersome process, especially for stylistic or non-physical attenuation. Since Bitterli et al.'s non-classical framework [4] allows for arbitrary transmittance functions, we need a more straightforward way to derive a similar integral formulations to Eq. (6.19) for *any* transmittance model. For example, Vicini et al.'s non-exponential model for volumetric scene representations [61] or Wrenninge et al.'s non-exponential subsurface scattering [113].

As an alternative to the RTE derivation, the Volterra formulation can be derived in a more general manner by considering the initial value problem,

$$I(a, b) := I(a, a) + \int_a^b \frac{d}{dt} [I(a, t)] dt, \quad (6.20)$$

which evaluates a general quantity, I , at some *initial* distance and then integrates over how that quantity *changes* up until some desired distance is

[4]: Bitterli et al. (2018), 'A Radiative Transfer Framework for Non-Exponential Media'

[61]: Vicini et al. (2021), 'A Non-Exponential Transmittance Model for Volumetric Scene Representations'

[113]: Wrenninge et al. (2017), *Path Traced Subsurface Scattering Using Anisotropic Phase Functions and Non-Exponential Free Flights*

reached. Since we assume that transmittance is reciprocal ($I(a, b) = I(b, a)$), we can reformulate Eq. (6.20) as,

$$\begin{aligned} I(a, b) &:= I(b, b) + \int_b^a \frac{d}{dt} [I(t, b)] dt \\ &= I(b, b) - \int_a^b \frac{d}{dt} [I(t, b)] dt \end{aligned} \quad (6.21)$$

By plugging in the definition for exponential transmittance (6.5) in for $I(a, b)$,

$$\begin{aligned} T_r(x_a, x_b) &:= T_r(x_b, x_b) - \int_a^b \frac{d}{dt} [T_r(x_t, x_b)] dt, \\ &= 1 - \int_a^b \mu_r(x_t) T_r(x_t, x_b) dt, \end{aligned} \quad (6.22)$$

we arrive at the same Volterra integral formulation as Eq. (6.19)²

IVP is consistent. By defining $I_n(a, b) = I(a, n)$, the generalized IVP (6.21) satisfies Eq. (6.4) since,

$$I(a, b) = \lim_{n \rightarrow b} \left[I(n, n) - \int_a^n \frac{d}{dt} [I(t, n)] dt \right], \quad (6.23)$$

by the definition of an integral. Thus, an IVP is a valid construction for a consistent formulation completing step ① in Fig. 1.1.

Volterra has no non-linear terms. The Volterra formulation for exponential transmittance (6.19) contains the product of a scalar and a recursive call, thus it successfully transforms Eq. (6.5) into a formulation containing no non-linear terms. Meaning, Eq. (6.19) can be estimated through the direct application of Monte Carlo integration.

6.2.3 Null-collision formulation

While Eq. (6.21) fulfills the requirements of being a consistent formulation which is amenable to unbiased estimation, it is not directly apparent how we should importance sample the integrand. For heterogeneous media, both $\mu_r(x_t)$ and $T_r(x_t, x_b)$ are not known ahead of time.

We can alleviate this by reintroducing the null-collision paradigm through the application of *control variates*³,

$$\begin{aligned} T_r(x_a, x_b) &= e^{-\int_0^d \mu_r(x_t) dt} = e^{-\int_0^d \mu_r(x_t) + \mu_t - \mu_t dt} \\ &= e^{-\int_0^d \mu_t dt} e^{\int_0^d \mu_t - \mu_r(x_t) dt} = T_t(x_a, x_b) T_n^{-1}(x_a, x_b), \end{aligned} \quad (6.24)$$

where $\mu_t := \mu_r(x_t) + \mu_n(x_t)$ is the constant total extinction, $T_t(x_a, x_b) := e^{-\int_a^b \mu_t dt}$ is the *combined* transmittance, $T_n(x_a, x_b) := e^{-\int_a^b \mu_t - \mu_r(x_t) dt}$ is the

2: Note: transmittance evaluated over a distance of 0 will always be 1 by the definition of transmittance so $T_r(x_b, x_b) = T_r(x_a, x_a)$. We will use $T_r(x_a, x_a)$ and $T_r(x_b, x_b)$ interchangeably for readability when discussing transmittance. Additionally, transmittance is always a function of the optical depth $\left(T_r(x_t, x_b) = g\left(\int_t^b \mu_r(x_s) ds\right) \right)$ whose derivative with respect to t will always include an additional negative sign since t is in the lower bounds. We will not explicitly mention this in most of our future derivations.

3: While this application of control variates is similar to the control-residual decomposition introduced previously regarding residual ratio tracking (6.14), it differs by the fact that we introduce the fictitious total extinction μ_t instead of breaking apart the real extinction

null transmittance, and we will refer to $T_n^{-1}(\mathbf{x}_a, \mathbf{x}_b) := e^{\int_a^b \mu_t - \mu_r(\mathbf{x}_t) dt}$ as the *null amplification*. By replacing the null amplification with its IVP formulation (6.21) and simplifying,

$$\begin{aligned}
T_r(\mathbf{x}_a, \mathbf{x}_b) &= T_t(\mathbf{x}_a, \mathbf{x}_b) \left(1 - \int_a^b \frac{d}{dt} [T_n^{-1}(\mathbf{x}_t, \mathbf{x}_b)] dt \right) \\
&= T_t(\mathbf{x}_a, \mathbf{x}_b) \left(1 + \int_a^b \mu_n(\mathbf{x}_t) T_n^{-1}(\mathbf{x}_t, \mathbf{x}_b) dt \right) \\
&= T_t(\mathbf{x}_a, \mathbf{x}_b) + \int_a^b \mu_n(\mathbf{x}_t) T_t(\mathbf{x}_a, \mathbf{x}_b) T_n^{-1}(\mathbf{x}_t, \mathbf{x}_b) dt \\
&= T_t(\mathbf{x}_a, \mathbf{x}_b) + \int_0^d \mu_n(\mathbf{x}) T_t(\mathbf{x}_a, \mathbf{x}_t) T_t(\mathbf{x}_t, \mathbf{x}_b) T_n^{-1}(\mathbf{x}_t, \mathbf{x}_b) dt \\
&= T_t(\mathbf{x}_a, \mathbf{x}_b) + \int_0^d \mu_n(\mathbf{x}) T_t(\mathbf{x}_a, \mathbf{x}_t) T_r(\mathbf{x}_t, \mathbf{x}_b) dt \\
&= T_t(\mathbf{x}_a, \mathbf{x}_b) + \int_0^d \rho_f^{(t)}(\mathbf{x}_t) \alpha_n(\mathbf{x}_t) T_r(\mathbf{x}_t, \mathbf{x}_b) dt,
\end{aligned} \tag{6.25}$$

we arrive at an integral formulation containing the combined medium's analytic free-flight distribution, $\rho_f^{(t)}(\mathbf{x}_t)$, which we can perfectly importance sample. Thus, we have derived the *null-collision Volterra formulation* for exponential transmittance.

6.2.4 Re-deriving prior estimators

All of the transmittance estimators we reviewed in Sec. 6.1.1 can be derived by directly applying Monte Carlo integration to our null-collision Volterra formulation (6.25). To prove this, we will first introduce a generalized formulation for a Volterra transmittance estimator, and then we will discuss how each estimator from Sec. 6.1.1 can be derived from our generalized formulation through choosing specific design decisions.

Generalized estimation. We introduce our generalized estimator for Eq. (6.25) as,

$$\langle T_r(\mathbf{x}_a, \mathbf{x}_b) \rangle_{\text{General}} := \frac{T_t(\mathbf{x}_a, \mathbf{x}_b)}{P_{\text{nf}}} + \frac{\rho_f^{(t)}(\mathbf{x}_t) \alpha_n(\mathbf{x}_t) \langle T_r(\mathbf{x}_t, \mathbf{x}_b) \rangle_{\text{General}}}{p(\mathbf{x}_t) P_{\text{rec}}}, \tag{6.26}$$

where P_{nf} is the probability of evaluating the $T_t(\mathbf{x}_a, \mathbf{x}_b)$ term⁴, P_{rec} is the probability of evaluating the recursive call, and $p(\mathbf{x}_t)$ is the PDF for importance sampling \mathbf{x}_t . All prior tracking-based transmittance estimators differ by how they choose to set P_{nf} , P_{rec} , and $p(\mathbf{x}_t)$.

Ratio tracking. Ratio tracking arises by first importance sampling the integral in Eq. (6.25) proportionally to the total free-flight distribution, i.e. $p(\mathbf{x}_t) = \rho_f^{(t)}(\mathbf{x}_t)$. The probability of sampling a distance $t > b$ is exactly equal to $T_t(\mathbf{x}_a, \mathbf{x}_b)$, so ratio tracking then chooses to evaluate the first term

4: This can be considered either the combined transmittance term or the NEE term. We give it the subscript "nf" since it has historically either represented the probability of reaching \mathbf{x}_b during the "next-flight" for both ratio tracking and track length estimation or is always estimated by the "next-flight" estimator.

only when a distance $t \geq b$ is sampled, i.e. $P_{\text{nf}} = T_t(\mathbf{x}_a, \mathbf{x}_b)$. Otherwise, the recursive call is evaluated with probability $P_{\text{rec}} = 1 - T_t(\mathbf{x}_a, \mathbf{x}_b)$. However, since the integral is only evaluated when we importance sample distances between (a, b) , $p(\mathbf{x}_t)$ effectively becomes an *unnormalized* distribution with a normalization constant exactly equal to $\frac{1}{P_{\text{rec}}}$, which cancels with P_{rec} in Eq. (6.26). Thus, plugging,

$$p(\mathbf{x}_t) = \rho_f^{(t)}(\mathbf{x}_t); \quad P_{\text{nf}} = T_t(\mathbf{x}_a, \mathbf{x}_b); \quad P_{\text{rec}} = 1 - T_t(\mathbf{x}_a, \mathbf{x}_b), \quad (6.27)$$

into Eq. (6.26) will simplify to Eq. (6.12).

Track length estimation. Null-scattering track length estimation (6.9) differs from ratio tracking by applying an additional Russian Roulette termination check before every recursion to decide whether the estimator collides with a real particle with probability $P_r = \alpha_r(\mathbf{x}_t)$ and terminates early. Thus, plugging,

$$p(\mathbf{x}_t) = \rho_f^{(t)}(\mathbf{x}_t); \quad P_{\text{nf}} = T_t(\mathbf{x}_a, \mathbf{x}_b); \quad P_{\text{rec}} = \alpha_n(\mathbf{x}_t) \cdot (1 - T_t(\mathbf{x}_a, \mathbf{x}_b)), \quad (6.28)$$

into Eq. (6.26) will simplify to Eq. (6.9)⁵.

5: Note: P_{rec} accounts for the Russian Roulette *continuation* probability which is $1 - P_r = 1 - \alpha_r(\mathbf{x}_t) = \alpha_n(\mathbf{x}_t)$ for this case.

Weighted track length estimation. Weighted track length estimation (6.10) differs from the prior track length estimator by choosing to terminate with probability $P_r = \frac{\mu_r(\mathbf{x}_t)}{\mu_r(\mathbf{x}_t) + |\mu_n(\mathbf{x}_t)|}$ instead of $P_r = \alpha_r(\mathbf{x}_t) = \frac{\mu_r(\mathbf{x}_t)}{\mu_t}$. Thus, plugging,

$$p(\mathbf{x}_t) = \rho_f^{(t)}(\mathbf{x}_t); \quad P_{\text{nf}} = T_t(\mathbf{x}_a, \mathbf{x}_b); \quad P_{\text{rec}} = \frac{\mu_n(\mathbf{x}_t)}{\mu_r(\mathbf{x}_t) + |\mu_n(\mathbf{x}_t)|} \cdot (1 - T_t(\mathbf{x}_a, \mathbf{x}_b)), \quad (6.29)$$

into Eq. (6.26) will simplify to Eq. (6.10).

Next-flight estimation. Next-flight ratio tracking (6.15) differs from traditional ratio tracking (6.12) by *always* choosing to evaluate the combined transmittance term in the Volterra formulation. Thus, plugging,

$$p(\mathbf{x}_t) = \rho_f^{(t)}(\mathbf{x}_t); \quad P_{\text{nf}} = 1; \quad P_{\text{rec}} = 1 - T_t(\mathbf{x}_a, \mathbf{x}_b), \quad (6.30)$$

into Eq. (6.26) will simplify to Eq. (6.15).

General residual estimation. The idea of decomposing a medium into control and residual components is generally applicable to *every* transmittance estimator for classical transmittance. We can formulate a generalized estimator for a decomposed medium as,

$$\begin{aligned} T_r(\mathbf{x}_a, \mathbf{x}_b) &= T_r^c(\mathbf{x}_a, \mathbf{x}_b) T_r^r(\mathbf{x}_a, \mathbf{x}_b) = e^{-(b-a)\mu_r^c} e^{-\int_a^b \mu_r^r(\mathbf{x}_t) dt} \\ \langle T_r(\mathbf{x}_a, \mathbf{x}_b) \rangle_{\text{GeneralRes}} &:= T_r^c(\mathbf{x}_a, \mathbf{x}_b) \langle T_r^r(\mathbf{x}_a, \mathbf{x}_b) \rangle_{\text{General}} \end{aligned} \quad (6.31)$$

where we estimate the residual transmittance using our generalized transmittance estimator (6.26).

In practical implementations, we recommend *always* decomposing a medium to reduce render costs since it will more than likely benefit *all* transmittance estimators. However, due to the fact that the performance benefit of Eq. (6.31) is tied to how μ_r^c fits μ_r instead of any actual design decision involved in formulating a unique estimator, we consider any direct application of residuals orthogonal to this dissertation. Thus, we will not provide any alternative versions of our novel transmittance estimators for decomposed media and instead make note that incorporating residual decomposition is trivial.

6.2.5 Variance analysis

We have derived a consistent formulation which is amenable to unbiased estimation in the form of the null-collision Volterra integral formulation (6.25) fulfilling ① from our recipe (Fig. 1.1) for consistent estimation. We have also enumerated how to derive every transmittance estimator which has appeared previously within the rendering literature. While this may indirectly prove that the Volterra for classical transmittance is a formulation which is amenable to unbiased estimation with finite work-normalized variance, allowing us to skip to ③, it is to our benefit to still attempt to formulate a *theoretically optimal* estimator (fulfilling ②) using our generalized estimator (6.26).

A *theoretically optimal* estimator is loosely defined as: given the most ideal circumstances⁶, can we derive a finite ized variance estimator for a given consistent formulation. This estimator does not have to be practical in all cases⁷, we just need to prove that it exists to decide whether to follow step ③ in Fig. 1.1 to devise estimators with finite variance or step ⑥ to devise infinite variance estimators. For the case of transmittance estimation, this translates to considering our null-collision Volterra formulation (6.26) in homogenous media.

We begin by analyzing the variance of ratio tracking in a homogeneous medium. Using this analysis we will then derive a zero-variance importance sampling scheme which we will employ to additionally derive zero-variance estimators for both the weighted track-length and next-flight estimators. Based on these idealized estimators for homogeneous media, we later propose an adaptive importance sampling scheme for all current tracking based techniques in Sec. 6.2.6 (③) for heterogeneous media.

Variance of ratio tracking in a homogeneous medium. Given a constant density homogeneous medium ($\mu_r(\mathbf{x}) = \mu_r$), ratio tracking (6.12) will return transmittance estimates,

$$T_{Rat}(k) = \left(\frac{\mu_t - \mu_r}{\mu_t} \right)^k = (\alpha_n)^k = \left(\frac{\tau_n}{\tau_t} \right)^k, \quad (6.32)$$

6: For the case of exponential transmittance estimation, "ideal circumstances" means the medium has a constant real extinction.

7: i.e. heterogeneous media.

which can be written as a function of, k , the number of free-flights sampled. Since the actual free-flight distances are sampled according to an exponential distribution dictated by μ_t , this is analogous to directly sampling the number of free-flights from a Poisson distribution defined by the probability mass function,

$$p(k) := e^{-\tau_t} \frac{\tau_t^k}{k!}, \quad (6.33)$$

where $\tau_t = \mu_t \cdot (b - a)$ is the optical depth of the combined medium. Additionally, for the homogeneous case, the null optical depth is defined as $\tau_n = \mu_n \cdot (b - a)$ while real optical depth is $\tau_r = \mu_r \cdot (b - a)$.

Given Eq. (6.32) and Eq. (6.33), we can compute the variance of ratio tracking in a homogeneous medium in terms of the choice of τ_t as,

$$\begin{aligned} V[T_{Rat}] &= \sum_{k=0}^{\infty} (T_{Rat}(k))^2 p(k) - E[T_{Rat}]^2 = \sum_{k=0}^{\infty} \left(\left(\frac{\tau_n}{\tau_t} \right)^k \right)^2 p(k) - e^{-2\tau_r} \\ &= \sum_{k=0}^{\infty} \frac{\tau_n^{2k}}{\tau_t^{2k}} \left(e^{-\tau_t} \frac{\tau_t^k}{k!} \right) - e^{-2\tau_r} = \sum_{k=0}^{\infty} \frac{\tau_n^{2k}}{\tau_t^k} \frac{1}{k!} e^{-\tau_t} - e^{-2\tau_r} \\ &= e^{-\tau_t} \sum_{k=0}^{\infty} \left(\frac{\tau_n^2}{\tau_t} \right)^k \frac{1}{k!} - e^{-2\tau_r} = e^{-\tau_t} e^{\frac{\tau_n^2}{\tau_t}} - e^{-2\tau_r} \\ &= e^{\frac{\tau_n^2 - \tau_t^2}{\tau_t}} - e^{-2\tau_r} = e^{\frac{(\tau_t - \tau_r)^2 - \tau_t^2}{\tau_t}} - e^{-2\tau_r} \\ &= e^{\frac{\tau_r^2 - 2\tau_r \tau_t}{\tau_t}} - e^{-2\tau_r} = e^{-2\tau_r + \frac{\tau_r^2}{\tau_t}} - e^{-2\tau_r} \\ &= e^{-2\tau_r} \left(e^{\frac{\tau_r^2}{\tau_t}} - 1 \right), \end{aligned} \quad (6.34)$$

Equation (6.34) effectively implies that ratio tracking will only have zero variance if we take $\mu_t \rightarrow \infty$. However, this will also result in sampling infinitesimally small free-flight distances resulting in an infinite amount of work to reach zero-variance estimates. This is obviously inefficient considering the fact that the real transmittance is already analytic within a homogeneous medium.

Non-bounding extinctions. Equation (6.34) also proves that ratio-tracking will *always* have finite variance as long as $\mu_t > 0$. Since ratio tracking has finite variance and finite cost, its work-normalized variance is always going to be finite. Unfortunately, in Fig. 6.4 we plot the analytical variance for ratio-tracking in a homogeneous medium with bounding and non-bounding total extinctions, and the results imply that guaranteeing finite variance is *not* enough to have efficient estimators for non-linear functions of expectations. When μ_t is non-bounding, variance exponentially increases to the point where ratio tracking would be unusable in practice.

Unless specified otherwise, we will always assume that we have access to a bounding μ_t within this chapter. Later in Chapter 7 we will directly

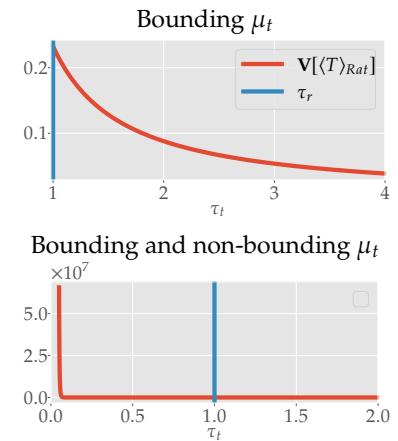


Figure 6.4: We plot the analytical variance of ratio tracking in a homogeneous medium with a bounding total extinction (top), and non-bounding total extinction (bottom). When the total extinction is bounding the variance is always manageable, however, in non-bounding situations the variance becomes exponentially unbounded.

tackle the problem of non-bounding total extinctions and analyze how it is occasionally more efficient to employ biased but consistent alternatives to avoid excessive variance.

Zero-variance ratio tracking. Since there is no variance in the extinction evaluations of a homogeneous medium, the existence of variance in the ratio tracking estimator (6.34) implies that we are sub-optimally importance sampling the integrand when sampling free-flights.

To derive a zero-variance estimator for ratio tracking, we start by rearranging terms in our generalized estimator (6.26),

$$\begin{aligned}\langle T_r(\mathbf{x}_a, \mathbf{x}_b) \rangle &= \frac{T_t(\mathbf{x}_a, \mathbf{x}_b)}{P_{\text{nf}}} + \frac{\rho_f^{(t)}(\mathbf{x}_t) \alpha_n(\mathbf{x}_t) \langle T_r(\mathbf{x}_t, \mathbf{x}_b) \rangle}{p(\mathbf{x}_t) P_{\text{rec}}}, \\ &= \frac{T_t(\mathbf{x}_a, \mathbf{x}_b)}{P_{\text{nf}}} + \frac{\mu_t T_t(\mathbf{x}_a, \mathbf{x}_t) \mu_n(\mathbf{x}_t) \langle T_r(\mathbf{x}_t, \mathbf{x}_b) \rangle}{\mu_t p(\mathbf{x}_t) P_{\text{rec}}}, \\ &= \frac{T_t(\mathbf{x}_a, \mathbf{x}_b)}{P_{\text{nf}}} + \frac{T_t(\mathbf{x}_a, \mathbf{x}_t) \mu_n(\mathbf{x}_t) \langle T_r(\mathbf{x}_t, \mathbf{x}_b) \rangle}{p(\mathbf{x}_t) P_{\text{rec}}},\end{aligned}\quad (6.35)$$

to be in terms of the product of extinctions and transmittances instead of free-flight distributions and albedos. Since $T_t(\mathbf{x}_a, \mathbf{x}_b) = T_n(\mathbf{x}_a, \mathbf{x}_b) T_r(\mathbf{x}_a, \mathbf{x}_b)$, let us replace all instances of T_t with this product,

$$\begin{aligned}\langle T_r(\mathbf{x}_a, \mathbf{x}_b) \rangle &= \frac{[T_n(\mathbf{x}_a, \mathbf{x}_b) T_r(\mathbf{x}_a, \mathbf{x}_b)]}{P_{\text{nf}}} + \\ &\quad \frac{[T_n(\mathbf{x}_a, \mathbf{x}_t) T_r(\mathbf{x}_a, \mathbf{x}_t)] \mu_n(\mathbf{x}_t) \langle T_r(\mathbf{x}_t, \mathbf{x}_b) \rangle}{p(\mathbf{x}_t) P_{\text{rec}}} \\ &= \frac{[T_n(\mathbf{x}_a, \mathbf{x}_b) T_r(\mathbf{x}_a, \mathbf{x}_b)]}{P_{\text{nf}}} + \frac{\left[\rho_f^{(n)}(\mathbf{x}_t) T_r(\mathbf{x}_a, \mathbf{x}_t) \right] \langle T_r(\mathbf{x}_t, \mathbf{x}_b) \rangle}{p(\mathbf{x}_t) P_{\text{rec}}},\end{aligned}\quad (6.36)$$

where we use the notation [...] to denote that we cannot evaluate those quantities directly. However, if we can choose to importance sample proportionally to everything else, we will implicitly arrive at the real transmittance without having to directly evaluate it. We can achieve this by sampling proportionally to the null free-flight distribution,

$$\rho_f^{(n)}(\mathbf{x}_t) := \mu_n(\mathbf{x}_t) T_n(\mathbf{x}_a, \mathbf{x}_b), \quad (6.37)$$

where we then choose to use a similar distance based termination strategy as ratio tracking where $P_{\text{nf}} = T_n(\mathbf{x}_a, \mathbf{x}_b)$ and $P_{\text{rec}} = 1 - T_n(\mathbf{x}_a, \mathbf{x}_b)$ which normalizes $\rho_f^{(n)}(\mathbf{x}_t)$. Plugging,

$$p(\mathbf{x}_t) = \rho_f^{(n)}(\mathbf{x}_t); \quad P_{\text{nf}} = T_n(\mathbf{x}_a, \mathbf{x}_b); \quad P_{\text{rec}} = 1 - T_n(\mathbf{x}_a, \mathbf{x}_b), \quad (6.38)$$

into Eq. (6.36) then simplifying results in,

$$\langle T_r(x_a, x_b) \rangle_{\text{ZeroRat}} := \begin{cases} T_r(x_a, x_b) & t \geq b \\ T_r(x_a, x_t) \langle T_r(x_t, x_b) \rangle & \text{otherwise,} \end{cases} \quad (6.39)$$

which is a zero-variance ratio tracking estimator that only returns the real transmittance in a homogeneous medium no matter the choice of μ_t . This completes ② in Fig. 1.1.

Null versus total free-flights. Traditional ratio tracking samples tentative free-flights by importance sampling the combined free-flight distribution, $\rho_f^{(t)}$, which is dictated by μ_t . We assume that $\mu_t \geq \mu_r$, otherwise the variance of traditional ratio tracking (6.34) will no longer be bounded.

When μ_t is bounding, the null extinction will *never* be greater than the total extinction (i.e. $\mu_n \leq \mu_t$). Since the length of the mean-free-path is exactly equal to the inverse of the extinction ($1/\mu_*$) dictating the free-flight distribution used for sampling, importance sampling the integrand proportionally to $\rho_f^{(n)}$ will result in sampling longer distances and less tentative free-flights on average than sampling from $\rho_f^{(t)}$. Thus, the ideal importance sampling strategy is *less costly* than traditional ratio-tracking, whereas importance sampling proportionally to the combined free-flight distribution turns out to be *too conservative*.

In practice. Unfortunately, our zero-variance importance sampling scheme is not practical for estimating transmittance within heterogeneous media. Employing it would require us to be able to analytically evaluate the null transmittance everywhere which would imply that we could also analytically evaluate the real transmittance. Unfortunately, this is not possible in practice since $T_r(x_a, x_b)$ has to be estimated within general heterogeneous media.

The realization that the null free-flight distribution is the ideal importance sampling scheme for ratio tracking was independently realized by d'Eon et al. [114], while we introduced the same idea shortly after [115]. However, d'Eon et al. did not propose a practical way of applying this theory, while in Sec. 6.2.6 we will employ our zero-variance analysis to propose adaptive importance sampling schemes which can improve the performance of existing tracking based transmittance estimators. Additionally, we will now derive zero-variance estimators for both track length estimation and next-flight estimation.

[114]: d'Eon et al. (2021), 'Zero-variance Transmittance Estimation'

[115]: Misso et al. (2023), 'Progressive null-tracking for volumetric rendering'

Zero-variance track length estimation. As shown in Eq. (6.28), a traditional track length estimator is derived by applying Russian Roulette termination to ratio tracking based on α_n . Mechanically, this results in an estimator which returns a binary estimate for transmittance of either 0 or 1 depending on whether a real collision is encountered or not. For a track length estimator of this nature to have zero variance, the binary estimate

has to always return the real transmittance, meaning the Russian Roulette termination probability has to always be $P_r = 0$ such that a score of 0 is never returned.

We derive a zero-variance track length estimator by adapting the Russian Roulette probabilities used in traditional track length estimation and apply them to our zero-variance ratio tracking estimator (6.39). Instead of basing the continuation probabilities on α_n , we instead base them on what we will refer to as the *sampling albedo*, $\alpha_{MC} := \frac{\mu_n}{\mu_{MC}}$, where μ_{MC} is the extinction dictating the importance sampled free-flight distribution⁸. By importance sampling free-flights proportionally to the null extinction (6.37) ($\mu_{MC} = \mu_n$), $\alpha_{MC} = 1$ in the homogeneous case which causes Russian Roulette termination to never occur. This results in a zero variance version of track length estimation which is algorithmically identical to Eq. (6.39).

We will formulate an adaptive importance sampling strategy for track length estimation based on this analysis in Sec. 6.2.6. One thing to make note of is that in the heterogeneous case where we cannot evaluate the real optical depth analytically, there may arise situations where the sampling albedo $\alpha_{MC} > 1$, or $\alpha_{MC} < 0$. For this reason, we will base the practical application of our zero-variance importance sampling for track length estimation on the weighted track length estimator (6.29).

Zero-variance next-flight. Formulating a zero-variance version of the next-flight transmittance estimator (6.15) is more involved since next-flight always chooses to accumulate the "NEE" term during every recursion by setting $P_{nf} = 1$. Since $T_t(\mathbf{x}_a, \mathbf{x}_b)$ is always evaluated, we have to devise a strategy for evaluating the recursive term such that it corrects for the possibility that $T_t(\mathbf{x}_a, \mathbf{x}_b) \neq T_r(\mathbf{x}_a, \mathbf{x}_b)$.

To accomplish this, we begin by first breaking down the combined transmittance into the product of null and real transmittances,

$$\langle T_r(\mathbf{x}_a, \mathbf{x}_b) \rangle = [T_n(\mathbf{x}_a, \mathbf{x}_b) T_r(\mathbf{x}_a, \mathbf{x}_b)] + \frac{\left[\rho_f^{(n)}(\mathbf{x}_t) T_r(\mathbf{x}_a, \mathbf{x}_t) \right] \langle T_r(\mathbf{x}_t, \mathbf{x}_b) \rangle}{p(\mathbf{x}_t) P_{rec}}. \quad (6.40)$$

By then combining the expected value of the two real transmittances,

$$T_r(\mathbf{x}_a, \mathbf{x}_b) = [T_n(\mathbf{x}_a, \mathbf{x}_b) T_r(\mathbf{x}_a, \mathbf{x}_b)] + \frac{\rho_f^{(n)}(\mathbf{x}_t)}{p(\mathbf{x}_t) P_{rec}} T_r(\mathbf{x}_a, \mathbf{x}_b), \quad (6.41)$$

it becomes obvious that we can formulate a zero-variance next-flight estimator through choosing various design decisions such that,

$$\frac{\left[\rho_f^{(n)}(\mathbf{x}_t) \right]}{p(\mathbf{x}_t) P_{rec}} = (1 - T_n(\mathbf{x}_a, \mathbf{x}_b)). \quad (6.42)$$

All prior estimators employ a form of rejection sampling to choose whether or not to evaluate the recursive term by only recursing when a distance

8: In the case of traditional ratio tracking, $\mu_{MC} = \mu_t$, while $\mu_{MC} = \mu_n$ for the case of our zero-variance formulation.

$t < b$ is sampled. However, by only allowing distances to be sampled between (a, b) , the free-flight distribution which we sample from ($p(\mathbf{x}_t)$) becomes effectively *unnormalized*. In all prior estimators, P_{rec} naturally cancels out the normalization term, but by setting,

$$p(\mathbf{x}_t) = \rho_f^{(n)}(\mathbf{x}_t); \quad P_{\text{nf}} = 1; \quad P_{\text{rec}} = 1, \quad (6.43)$$

we will arrive at a zero-variance next-flight estimator for homogeneous media since the normalization constant for $\rho_f^{(n)}(\mathbf{x}_t)$ will result in Eq. (6.42).

Unfortunately, plugging Eq. (6.43) into Eq. (6.26) will result in an estimator which never terminates, and imposing any termination criterion through stochastic means will introduce variance unless we also apply the complementary termination criterion to the first term. Meaning, we cannot have a zero-variance next-flight estimator that *always* chooses to accumulate a next-event sample at every recursion without also evaluating an infinite number of recursions. Such a purely zero-variance next-flight estimator is mostly of theoretical interest.

By relaxing our definition of a "next-flight" estimator, we propose a zero-variance alternative,

$$\langle T_r(\mathbf{x}_a, \mathbf{x}_b) \rangle_{\text{ZeroNF}} := \frac{T_t(\mathbf{x}_a, \mathbf{x}_b)}{P_{\text{nf}}} + \frac{\rho_f^{(t)}(\mathbf{x}_t) \alpha_n(\mathbf{x}_t) \langle T_r(\mathbf{x}_t, \mathbf{x}_b) \rangle_{\text{Dist}}}{p(\mathbf{x}_t) P_{\text{rec}}}, \quad (6.44)$$

$$\langle T_r(\mathbf{x}_t, \mathbf{x}_b) \rangle_{\text{Dist}} := \begin{cases} \langle T_r(\mathbf{x}_t, \mathbf{x}_b) \rangle_{\text{ZeroNF}}; & b - t < s \\ \langle T_r(\mathbf{x}_t, \mathbf{x}_b) \rangle_{\text{ZeroRat}}; & b - t \geq s, \end{cases}$$

which decides to recurse by evaluating either zero-variance next-flight using the parameters in Eq. (6.43), or zero-variance ratio tracking (??) with the parameters in Eq. (6.38), depending on whether the sampled distance lies within some specified distance, s , from the endpoint⁹. This branching choice induces no variance since both techniques are individually zero-variance estimators, but it does provide a necessary termination condition in return for no longer being a purely "next-flight" technique. We will use this loose interpretation to derive an adaptive estimator for practical heterogeneous applications in Sec. 6.2.6.

9: In practice, we typically set s equal to the mean-free-path of the combined medium ($1/\mu_t$)

6.2.6 Adaptive importance sampled estimators

Importance sampling proportionally to $p(\mathbf{x}_t) = \rho_f^{(n)}(\mathbf{x}_t)$ leads to hypothetical zero-variance estimators. Unfortunately, zero-variance importance sampling cannot be employed in general heterogeneous media since we do not possess an analytic $\rho_f^{(n)}(\mathbf{x}_t)$. Instead, we propose importance sampling proportionally to an *approximate* null free-flight distribution that is *adaptively* constructed based on the *history* of a tracking.

All prior tracking-based transmittance estimators (6.1.1) rely on free-flight distance sampling where the *same* free-flight distribution is employed

for *every* recursive call ($p(\mathbf{x}_t) = \rho_f^{(t)}(\mathbf{x}_t)$). The main insight behind our novel importance sampling technique is that we can use *different* free-flight sampling distributions at every recursion allowing us to *adaptively* approximate $\rho_f^{(n)}(\mathbf{x}_t)$ on the fly.

Since $\mu_n(\mathbf{x}_t)$ is always evaluated within the recursive call as a byproduct of our generalized transmittance estimator (6.26), we propose to utilize this evaluation to dictate an approximate null free-flight distribution,

$$p(\mathbf{x}_t) = \rho_f^{(\text{MC})}(\mathbf{x}_t) = \mu_{\text{MC}} e^{-\mu_{\text{MC}} \mathbf{x}_t(t-a)} = \mu_{\text{MC}} T_{\text{MC}}(\mathbf{x}_a, \mathbf{x}_t), \quad (6.45)$$

to sample from for the next recursion where $\mu_{\text{MC}} = \mu_n(\mathbf{x}_s)$ and \mathbf{x}_s is the location of the last extinction evaluation. In the homogeneous case, the null free-flight distribution (6.37) can be analytically evaluated as long as we know $\mu_n(\mathbf{x}_t) = \mu_n$ ahead of time. Meaning, by using the current null extinction as our sampling extinction, we would be guaranteed to have a perfect zero-variance estimator in the homogeneous case for every recursive evaluation after the first.

This adaptive importance sampling scheme effectively assumes that the medium is always homogeneous and then chooses to employ optimal zero variance importance sampling for that homogeneous medium. In the context of heterogeneous media, our adaptive importance sampling scheme always chooses the *locally* optimal choice for sampling free-flights. We visualize the difference between our adaptive free-flight distribution and the total free-flight distribution in Fig. 6.5 for a specific location within a medium.

Adaptive ratio tracking. Adaptive ratio tracking is formulated by setting,

$$p(\mathbf{x}_t) = \rho_f^{(\text{MC})}(\mathbf{x}_t); \quad P_{\text{nf}} = T_{\text{MC}}(\mathbf{x}_a, \mathbf{x}_t); \quad P_{\text{rec}} = 1 - T_{\text{MC}}(\mathbf{x}_a, \mathbf{x}_t), \quad (6.46)$$

in Eq. (6.26) where μ_{MC} is set to be the previous evaluation of $\mu_n(\mathbf{x})$. This results in an estimator which simplifies to,

$$\begin{aligned} \langle T_r(\mathbf{x}_a, \mathbf{x}_b) \rangle_{\text{AdapRat}} &= e^{(\mu_{\text{MC}}(\mathbf{x}_{n-1}) - \mu_t)\Delta_n} \prod_{j=1}^{n-1} \frac{\mu_n(\mathbf{x}_j)}{\mu_{\text{MC}}(\mathbf{x}_{j-1})} e^{(\mu_{\text{MC}}(\mathbf{x}_{j-1}) - \mu_t)\Delta_j} \\ &= e^{(\mu_n(\mathbf{x}_{n-1}) - \mu_t)\Delta_n} \prod_{j=1}^{n-1} \frac{\mu_n(\mathbf{x}_j)}{\mu_n(\mathbf{x}_{j-1})} e^{(\mu_n(\mathbf{x}_{j-1}) - \mu_t)\Delta_j} \quad (6.47) \\ &= e^{-\mu_r(\mathbf{x}_{n-1})\Delta_n} \prod_{j=1}^{n-1} \frac{\mu_n(\mathbf{x}_j)}{\mu_n(\mathbf{x}_{j-1})} e^{-\mu_r(\mathbf{x}_{j-1})\Delta_j}, \end{aligned}$$

where $\Delta_j = \|\mathbf{x}_j - \mathbf{x}_{j-1}\|$ is the distance between each subsequent extinction evaluation with $\mathbf{x}_0 = \mathbf{x}_a$, $\mu_n(\mathbf{x}_0) = \mu_t$, and n is the total number of tentative free-flights sampled. Note, in traditional ratio tracking n is distributed according to a known poisson distribution, however, this will not be the case for our adaptive estimators since the sampling distribution can differ every recursion. We visualize the difference between our adaptive ratio

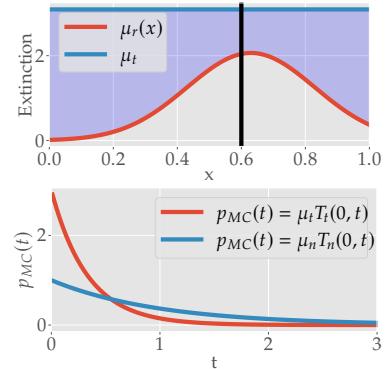


Figure 6.5: An illustration of the difference between our adaptive importance sampling PDF and ratio-tracking's traditional importance sampling PDF. The black vertical line represents the location of the current tracking in the medium (top). The bottom graph shows that our technique will sample the next exponential distance based on $\mu_n(\mathbf{x}_x)$ (blue) instead of $\mu_t(\mathbf{x}_x)$ (red).

tracking estimator (6.47) and traditional ratio tracking (6.12) in Fig. 6.6, and we additionally provide pseudocode for our implementation in Algorithm 5¹⁰.

Algorithm 5: Adaptive ratio transmittance estimator

```

1  $t = s = a$ 
2  $T = 1$ 
3  $\mu_{MC} = \mu_t$ 
4  $\mu_{last} = 0$ 
5 while  $t < b$  do
6    $t = t - \frac{\log(1-\xi)}{\mu_{MC}}$ 
7   if  $t \geq b$  then
8     break
9    $T = T * \frac{\mu_n(\mathbf{x}_t)}{\mu_{MC}} * e^{-\mu_{last}(t-s)}$ 
10   $\mu_{last} = \mu_r(\mathbf{x}_t)$ 
11   $s = t$ 
12   $\mu_{MC} = \text{Min}(\mu_n(\mathbf{x}_t), 0.1 * \mu_t)$ 
13 return  $T * e^{-\mu_{last}(b-s)}$ 
```

10: We clamp μ_{MC} to avoid division by zero errors.

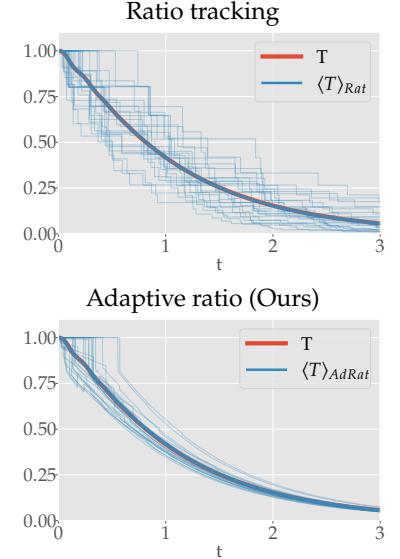


Figure 6.6: We visualize the behaviors of traditional ratio tracking versus our adaptive ratio tracking for the same extinction configuration from Fig. 6.2. Our adaptive version takes longer free-flights on average and exhibits less overall variance in near-constant regions.

[114]: d'Eon et al. (2021), 'Zero-variance Transmittance Estimation'

Residual ratio tracking Comparison. In d'Eon et al.'s analysis of zero-variance ratio tracking [114], they point out that residual ratio tracking is an approximate zero-variance estimator whose variance is dependent on the accuracy of the data structure which is used to store the control extinction (μ_t^c) and μ_t . They also showed that residual ratio tracking can be derived as a direct importance sampling scheme where we always importance sample the integrand using a free-flight distribution whose sampling extinction is equal to the combined extinction of the residual ($\mu_{MC} = \mu_t - \mu_t^c = \mu_t^r$).

By plugging $\mu_{MC} = \mu_t - \mu_t^c$ into our adaptive ratio tracking estimator (6.47),

$$\begin{aligned}
\langle T_r(\mathbf{x}_a, \mathbf{x}_b) \rangle &= e^{(\mu_{MC}(\mathbf{x}_{n-1}) - \mu_t)\Delta_n} \prod_{j=1}^{n-1} \frac{\mu_n(\mathbf{x}_j)}{\mu_{MC}(\mathbf{x}_{j-1})} e^{(\mu_{MC}(\mathbf{x}_{j-1}) - \mu_t)\Delta_j} \\
&= e^{(\mu_t - \mu_t^c - \mu_t)\Delta_n} \prod_{j=1}^{n-1} \frac{\mu_n(\mathbf{x}_j)}{\mu_t - \mu_t^c} e^{(\mu_t - \mu_t^c - \mu_t)\Delta_j} \\
&= e^{-\mu_t^c(b-a)} \prod_{j=1}^{n-1} \frac{\mu_n(\mathbf{x}_j)}{\mu_t^r} \\
&= e^{-\mu_t^c(b-a)} \prod_{j=1}^{n-1} \frac{\mu_t - \mu_r(\mathbf{x}_j)}{\mu_t^r} \\
&= e^{-\mu_t^c(b-a)} \prod_{j=1}^{n-1} \frac{(\mu_t - \mu_t^c) - (\mu_r(\mathbf{x}_j) - \mu_t^c)}{\mu_t^r} \\
&= e^{-\mu_t^c(b-a)} \prod_{j=1}^{n-1} \frac{\mu_t^r - \mu_r(\mathbf{x}_j)}{\mu_t^r}
\end{aligned} \tag{6.48}$$

we see that our technique simplifies to residual ratio tracking (6.14)¹¹. While

11: Note that $\mu_t^c = \mu_r^c$ within the control medium.

our technique differs significantly from residual ratio tracking in how we choose μ_{MC} , this connection implies a fascinating insight.

Our adaptive version of ratio tracking is essentially a residual ratio tracking estimator which uses the real extinction evaluated at the last tentative free-flight as the control extinction ($\mu_t^c = \mu_r(\mathbf{x}_{j-1})$) for the next free-flight sample. Traditional residual ratio tracking constructs its control medium by querying a data structure containing local minimum and maximum extinctions spanning the entire medium extents. On the other hand, our technique constructs the control medium on the fly as a piecewise constant function using the evaluations of the real extinction during its tracking. This insight will become necessary when formulating our adaptive weighted track length transmittance estimator and its complementary adaptive delta tracking free-flight sampling routines.

Adaptive weighted track-length estimation. The defining characteristic of prior track length estimators (Sec. 6.1.1) is that they prematurely return transmittance estimates of 0 whenever they fail to survive the entire length of a tracking. This premature termination makes them useful as both transmittance estimators as well as free-flight distance sampling routines for importance sampling medium scatter events (Sec. 5.4.2).

Any reformulation of prior track length estimators should ideally maintain the property that they can be applied to either transmittance estimation or free-flight distance sampling. Unfortunately, this imposes the restriction that Russian Roulette termination has to happen only when a real collision occurs, so we cannot choose to apply Russian Roulette termination based on the running weight or the ratio between the sampling and combined transmittances. We have to only base it on the ratio between the real and total extinctions at the current location in the tracking.

To do so, we rely on the insight that our adaptive importance sampling scheme can be considered an on-the-fly construction of a control medium. For transmittance estimation, this means we can achieve an adaptive technique analogous to weighted track length estimation (6.29) by setting the Russian Roulette continuation probabilities based on the current residual null extinction,

$$P_{rec} \propto \frac{|\mu_n(\mathbf{x}_j)|}{|\mu_r(\mathbf{x}_j) - \mu_{r,mc}(\mathbf{x}_{j-1})| + |\mu_n(\mathbf{x}_j)|}. \quad (6.49)$$

Thus, by plugging,

$$p(\mathbf{x}_t) = \rho_f^{(MC)}(\mathbf{x}_t); \quad P_{nf} = T_{MC}(\mathbf{x}_a, \mathbf{x}_t); \quad (6.50)$$

$$P_{rec} = \frac{|\mu_n(\mathbf{x}_t)|}{|\mu_r(\mathbf{x}_t) - \mu_{MC}| + |\mu_n(\mathbf{x}_t)|} \cdot (1 - T_{MC}(\mathbf{x}_a, \mathbf{x}_t)), \quad (6.51)$$

into Eq. (6.26) we achieve our adaptive weighted track length estimator. We provide pseudocode for our adaptive track length transmittance estimator in Algorithm 6 and visualize its behavior in Fig. 6.7.

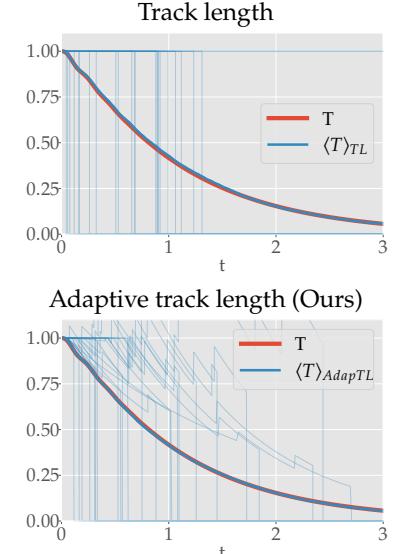


Figure 6.7: We visualize the behaviors of traditional track length estimation versus our adaptive track length estimator for the same extinction configuration from Fig. 6.2 except we plot the average of 10000 trackings to show our technique is correct. Our adaptive version takes longer free-flights on average and applies weighted Russian Roulette termination on the residual medium which increases the transmittance estimates whenever the estimator decides to continue recursing.

Algorithm 6: Adaptive track length transmittance estimator

```

1  $t = s = a$ 
2  $T = 1$ 
3  $\mu_{MC} = \mu_t$ 
4  $\mu_{last} = 0$ 
5 while  $t < b$  do
6    $s = t$ 
7    $t = t - \frac{\log(1-\xi)}{\mu_{MC}}$ 
8   if  $t \geq b$  then
9     break
10   $P = \frac{|\mu_n(x_t)|}{|\mu_n(x_t)| + |\mu_r(x_t) - \mu_{last}|}$ 
11  if  $P < \xi$  then
12    return 0
13   $T = T * \frac{\mu_n(x_t)}{\mu_{MC} * P} * e^{-\mu_{last}(t-s)}$ 
14   $\mu_{last} = \mu_r(x_t)$ 
15   $\mu_{MC} = \text{Min}(\mu_n(x_t), 0.1 * \mu_t)$ 
16 return  $T * e^{-\mu_{last}(b-s)}$ 

```

Adaptive next-flight. Our adaptive next-flight estimator performs the same novel importance sampling as the prior two transmittance estimators, however, it always chooses to evaluate both the NEE term and the recursive term by sampling from the *normalized* free-flight distribution,

$$\rho_f^{(MC)}(x_t) := \frac{(\mu_{MC})e^{-\mu_{MC}x_t(t-a)}}{1 - e^{-\mu_{MC}(b-a)}}, \quad (6.52)$$

rather than the unnormalized one. Since applying the inversion method (Eq. (5.24)) to Eq. (6.52) results in a sampling routine which only samples $t \in (a, b)$, we cannot rely on the same termination condition as prior transmittance estimators which terminate when $t \geq b$. To impose a termination condition, we make the insight that the recursive call expects an unbiased estimate for $\langle T_r(x_t, x_b) \rangle$, however, we do not have to employ the same unbiased estimator every recursion. Instead, we choose to replace the recursive call with our adaptive ratio tracking estimator when we reach a distance within a single mean-free-path of (x_b) similar to our alternative zero variance next-flight estimator (6.44). By utilizing this termination strategy and plugging,

$$p(x_t) = \frac{(\mu_{MC})e^{-\mu_{MC}x_t(t-a)}}{1 - e^{-\mu_{MC}(b-a)}}; \quad P_{nf} = 1; \quad P_{rec} = 1, \quad (6.53)$$

into Eq. (6.26), we arrive at our adaptive next-flight estimator which we provide pseudocode for in Algorithm 7.

Algorithm 7: Adaptive next-flight transmittance estimator

```

1  $t = s = a$ 
2  $T = \mu_{\text{last}} = 0$ 
3  $W = 1$ 
4  $\mu_{\text{MC}} = \mu_t$ 
5 while  $t < b$  do
6    $s = t$ 
7    $T = T + W * e^{-\mu_t(b-t)}$ 
8    $T_n = 1 - e^{-\mu_{\text{MC}}(b-t)}$ 
9    $t = t - \frac{\log(1-(\xi*T_n))}{\mu_{\text{MC}}}$ 
10   $W = W * T_n * \frac{\mu_r(x_t)}{\mu_{\text{MC}}} * e^{-\mu_{\text{last}}(t-s)}$ 
11  if  $t \geq b - \frac{1}{\mu_t}$  then
12    break
13   $\mu_{\text{last}} = \mu_r(x_t)$ 
14   $\mu_{\text{MC}} = \text{Min}(\mu_n(x_t), 0.1 * \mu_t)$ 
15 return  $T + W * \text{AdaptiveRatio}(t, b)$ 

```

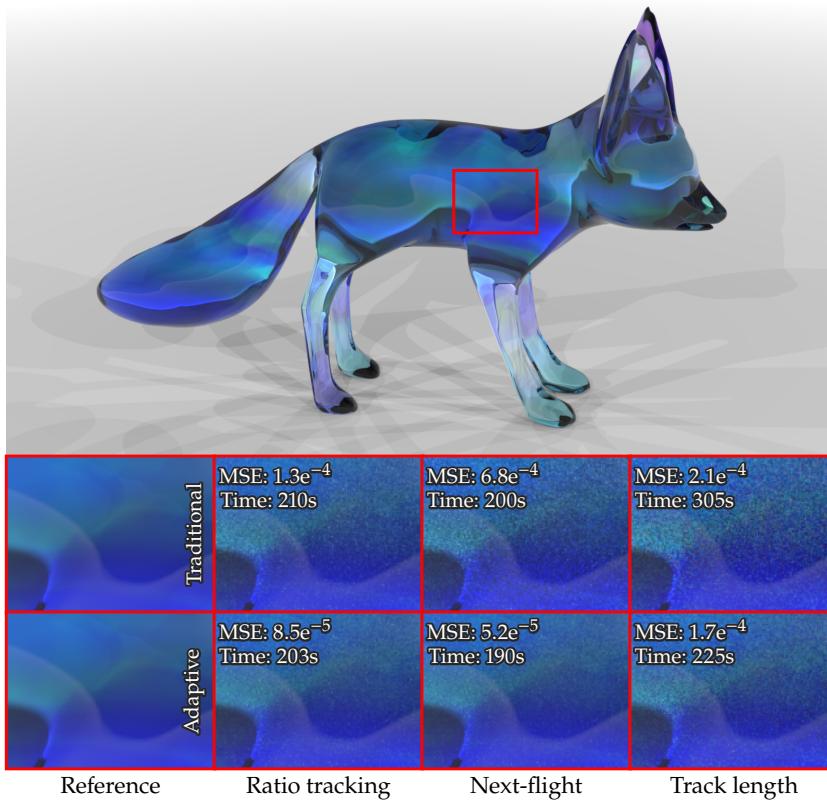


Figure 6.8: We compare traditional tracking-based transmittance estimators against our adaptive alternatives in a scene containing a fully absorptive medium with spectrally varying $\mu_r(x)$. Each technique was rendered until 100 million extinction evaluations had occurred across the entire image where $\mu_t = 2 \cdot \max(\mu_r(x))$ for all renders. Error metrics are computed across the entire image.

Comparison to traditional tracking-based estimators. In Fig. 6.8 we compare our adaptive techniques to their traditional counterparts within a spectrally varying non-scattering medium where our techniques show moderate improvements over prior work. Since the medium is spectrally varying, we choose to set μ_{MC} to the maximum of $\mu_t - \mu_r$ across the spectral channels. Admittedly, naively setting μ_{MC} equal to the average of $\mu_t - \mu_r$ across all spectral channels will result in our adaptive estimators performing *worse* than their traditional counterparts in regions which only contain non-zero density in one channel (dark blue regions). Devising more intelligent settings for μ_{MC} within spectral media, or incorporating the ideas from spectral tracking [103] is left for future work.

We illustrate the performance improvements of our adaptive tracking-based techniques for a variety of canonical extinction functions in Fig. 6.9 for two choices of μ_t . Our adaptive alternatives provide more of a benefit when used alongside a relatively loose μ_t . Since our adaptive techniques sample from a distribution which *underestimates* the free-flight distribution of the combined medium, our estimators can exhibit higher variance (see constant extinction with adaptive next-flight in Fig. 6.9). However, since they result in *cheaper* estimators they tend to be more efficient than their traditional counterparts.

[103]: Kutz et al. (2017), ‘Spectral and Decomposition Tracking for Rendering Heterogeneous Volumes’

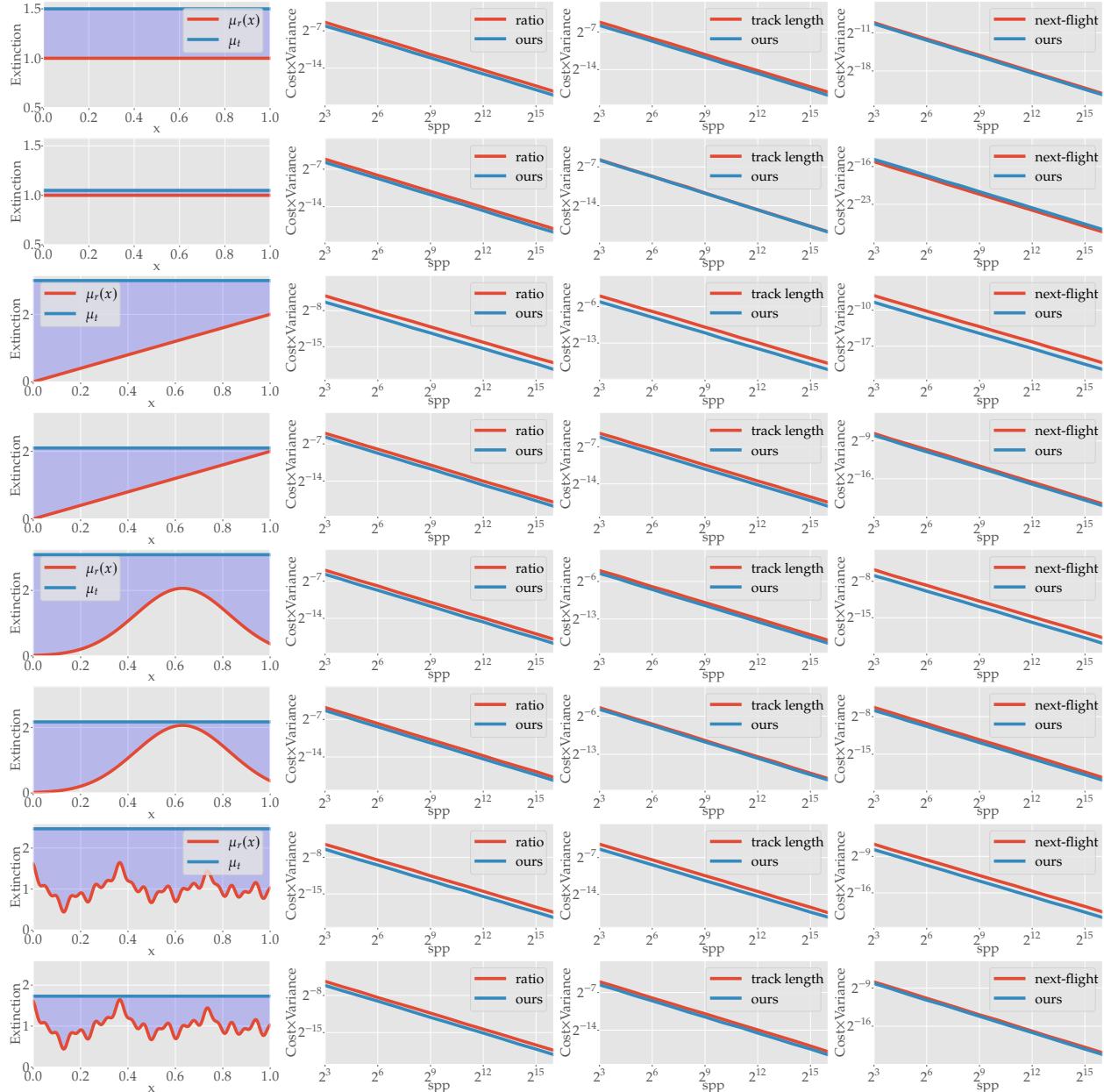


Figure 6.9: We compare the work-normalized variance (cost \times variance) of traditional tracking-based techniques against our adaptive alternatives across a variety of canonical extinction functions (left) with both relatively loose and tight μ_t . Our adaptive alternatives exhibit better performance the "looser" μ_t is for ratio tracking, next-flight, and the track length estimators.

6.2.7 Relating Null-VRE to VRE using the Volterra

At the beginning of this chapter we claimed that both the null-VRE (4.38) and the VRE (4.16) required unbiased estimators for estimating the real transmittance. However, the null-VRE contains only instances of the combined transmittance ($T_t(x_a, x_b)$) in its formulation (4.40). Since the combined transmittance is analytic, this would initially imply that having unbiased estimators for the real transmittance is unnecessary under the null-scattering formulation for light transport. We will now show using our novel Volterra formulation that the real transmittance will still arise within the null-scattering formulation of volumetric light transport when employing NEE.

When applying NEE to the null-VRE, we accumulate estimates for the direct illumination at every scattering vertex through tracing shadow connections to an emissive source. In the original VRE (4.16) this is equivalent to ignoring both in-scattered radiance and emission while only evaluating the *real* transmittance attenuated emissive radiance (5.31). For the null-VRE case we showed in Eq. (5.37) that computing the *direct illumination* to the next *point-sampled* emissive source involves computing the quantity,

$$L_{\text{NEE}}(\mathbf{x}, \vec{\omega}_o) = T_t(\mathbf{x}, \mathbf{x}_d)L_e(\mathbf{x}_d, \vec{\omega}_o) + \int_0^d \rho_f^{(t)}(\mathbf{x}_t)\alpha_n(\mathbf{x}_t)L_{\text{NEE}}(\mathbf{x}_t, \vec{\omega}_o) dt. \quad (6.54)$$

Expanding out the recursion,

$$\begin{aligned} L_{\text{NEE}}(\mathbf{x}, \vec{\omega}_o) &= L_e(\mathbf{x}_d, \vec{\omega}_o)T_t(\mathbf{x}, \mathbf{x}_d) + \\ &\quad L_e(\mathbf{x}_d, \vec{\omega}_o) \int_0^d \rho_f^{(t)}(\mathbf{x}_t)\alpha_n(\mathbf{x}_t)T_t(\mathbf{x}_t, \mathbf{x}_d)dt + \\ &\quad L_e(\mathbf{x}_d, \vec{\omega}_o) \int_0^d \int_t^d \rho_f^{(t)}(\mathbf{x}_t)\rho_f^{(s)}(\mathbf{x}_s)\alpha_n(\mathbf{x}_t)\alpha_n(\mathbf{x}_s)\dots \end{aligned} \quad (6.55)$$

makes it apparent that $L_e(\mathbf{x}_d, \vec{\omega}_o)$ is shared by every term. Through factoring out the emissive radiance in Eq. (6.54),

$$\begin{aligned} L_{\text{NEE}}(\mathbf{x}, \vec{\omega}_o) &= L_e(\mathbf{x}_d, \vec{\omega}_o) \left(\frac{T_t(\mathbf{x}, \mathbf{x}_d)L_e(\mathbf{x}_d, \vec{\omega}_o)}{L_e(\mathbf{x}_d, \vec{\omega}_o)} + \int_0^d \rho_f^{(t)}(\mathbf{x}_t)\alpha_n(\mathbf{x}_t)\frac{L_{\text{NEE}}(\mathbf{x}_t, \vec{\omega}_o)}{L_e(\mathbf{x}_d, \vec{\omega}_o)}dt \right) \\ &= L_e(\mathbf{x}_d, \vec{\omega}_o) \left(T_t(\mathbf{x}, \mathbf{x}_d) + \int_0^d \rho_f^{(t)}(\mathbf{x}_t)\alpha_n(\mathbf{x}_t)T_r(\mathbf{x}_t, \mathbf{x}_d)dt \right) \end{aligned} \quad (6.56)$$

the null-collision Volterra integral formulation for the real transmittance (6.25) appears which simplifies the direct illumination within the null-scattering formulation,

$$L_{\text{NEE}}(\mathbf{x}, \vec{\omega}_o) := T_r(\mathbf{x}, \mathbf{x}_d)L_e(\mathbf{x}_d, \vec{\omega}_o) \quad (6.57)$$

into the same product which appears in the traditional VRE (5.31). Thus, any practical rendering algorithm derived from the null-scattering formulation for volumetric transport still requires estimating the *real* transmittance.

Deriving null-VRE from the VRE. Miller et al. [51] originally derived the null-VRE (4.40) from first principles by introducing a null formulation of the RTE. However, for generalized non-exponential attenuation, formulating generalized versions of the RTE for every possible attenuation function is a cumbersome problem. By mathematically showing that we can derive the null-VRE from the VRE through using the null-Volterra form of our attenuation function, this will allow us to later discuss why a null-formulation for Bitterli et al.'s general non-exponential VRE [4] will not simplify nicely (see Appendix B).

To derive the null-VRE (4.38) from the classical VRE (4.16) we first make the assumption that there is no volumetric emission¹². We additionally choose to simplify all functional arguments by dropping all directional parameters $L(\mathbf{x}, \vec{\omega}) \rightarrow L(\mathbf{x}_t)$ and then further simplify the notation by replacing all positional arguments with the distances along the ray $L(\mathbf{x}_t) \rightarrow L(t)$.

We start with the transmittance form of the classical VRE (4.18),

$$L_o(0) := T_r(0, d)L_s(d) + \int_0^d \mu_s(t) T_r(0, t)L_m(t)dt \quad (6.58)$$

and replace all instances of the real transmittance with our null formulation for the Volterra (6.25),

$$T_r(a, b) := T_t(a, b) + \int_a^b \mu_n(t) T_t(a, t) T_r(t, b)dt, \quad (6.59)$$

resulting in,

$$\begin{aligned} L_o(0) = & L_s(d) \left(T_t(a, b) + \int_0^d \mu_n(t) T_t(0, t) T_r(t, d)dt \right) \\ & + \int_0^d \mu_s(t) \left(T_t(0, t) + \int_0^t \mu_n(s) T_t(0, s) T_r(s, t)ds \right) L_m(t) dt, \end{aligned} \quad (6.60)$$

which can be rewritten as the sum of four different terms that we define to be,

$$\begin{aligned} L_o(0) &:= L_1 + L_2 + L_3 + L_4 \\ L_1 &:= T_t(0, d)L_s(d) \\ L_2 &:= \int_0^d \mu_s(t) T_t(0, t)L_m(t) dt \\ L_3 &:= \int_0^d \mu_n(t) T_t(0, t)L_s(d) T_r(t, d)dt \\ L_4 &:= \int_0^d \int_0^t \mu_n(s) T_t(0, s)\mu_s(t) T_r(s, t)L_m(t) ds dt. \end{aligned} \quad (6.61)$$

Both L_1 and L_2 appear directly within the null-VRE (4.40) as the combined transmittance attenuated surface radiance and in-scattered radiance. To prove that we can derive the null-VRE from the VRE through our null-Volterra, we will now show that $L_3 + L_4$ is equivalent to the null-scattering

[51]: Miller et al. (2019), 'A Null-Scattering Path Integral Formulation of Light Transport'

[4]: Bitterli et al. (2018), 'A Radiative Transfer Framework for Non-Exponential Media'

12: This assumption is made to simplify the derivation. Including emission is trivial so it is left for the interested reader to do as an exercise.

term within Eq. (4.40).

By flipping the order of integration in L_4 ,

$$L_4 = \int_0^d \int_t^d \mu_n(s) T_t(0, s) \mu_s(t) T_r(s, t) L_m(t) dt ds. \quad (6.62)$$

and then renaming the variables $t \leftrightarrow s$,

$$L_4 = \int_0^d \int_s^d \mu_n(t) T_t(0, t) \mu_s(s) T_r(t, s) L_m(s) ds dt. \quad (6.63)$$

we then factor out the common terms within the sum of L_3 and L_4 ,

$$\begin{aligned} L_3 + L_4 &= \int_0^d \int_s^d \mu_n(t) T_t(0, t) \mu_s(s) T_r(t, s) L_m(s) ds dt + \\ &\quad \int_0^d \mu_n(t) T_t(0, t) L_s(d) T_r(t, d) dt \\ &= \int_0^d \mu_n(t) T_t(0, t) \left(T_r(t, d) L_s(d) + \int_s^d \mu_s(s) T_r(t, s) L_m(s) ds \right) dt, \end{aligned} \quad (6.64)$$

to arrive at a formulation where all of the terms within the parenthesis,

$$L_3 + L_4 = \int_0^d \mu_n(t) T_t(0, t) L_o(t) dt \quad (6.65)$$

are exactly equal to the out-going radiance evaluated at t (6.58). Thus, by replacing T_r with its null-collision Volterra formulation (6.59),

$$\begin{aligned} L_o(0) &:= L_1 + L_2 + L_3 + L_4 \\ &= T_t(0, d) L_s(d) \\ &\quad + \int_0^d \mu_s(t) T_t(0, t) L_m(t) dt \\ &\quad + \int_0^d \mu_n(t) T_t(0, t) L_o(t) dt \end{aligned} \quad (6.66)$$

we have rederived the null-scattering formulation for volumetric transmittance without beginning our derivations from the RTE.

6.3 Power-series integral formulation

In this section we present two different integral formulations based on the Maclaurin series expansion of the classical exponential transmittance function (6.5). We first present our base formulations free of null-collisions in Sec. 6.3.1, and then later derive the analogous null-collision formulations in Sec. 6.3.2 through the application of control variates. Later in Chapter 8 we will generalize this process for Taylor series expansions of general functions and further relate control variates to the expansion points used

in Taylor series expansions. After presenting our formulations, we rederive ratio tracking as a power series estimator in Sec. 6.3.3 and then introduce our novel transmittance estimators in Sec. 6.3.4.

6.3.1 Base formulations

The well known Maclaurin series expansion of the exponential function,

$$T_r(x_a, x_b) := e^{-\int_a^b \mu_r(x_t) dt} = e^X = \sum_{j=0}^{\infty} \frac{(X)^j}{j!} = \sum_{j=0}^{\infty} \frac{\left(-\int_a^b \mu_r(x_t) dt\right)^j}{j!}, \quad (6.67)$$

converts the problem of computing transmittance into computing the result of an infinite power series. Each summand in this infinite series becomes proportional to the real optical depth ($\tau_r := \int_a^b \mu_r(x_t) dt$) raised to incrementally increasing powers.

Iterative formulation. In its base form, Eq. (6.67) is not naively amenable to unbiased estimation since raising the optical depth to a power is a non-linear function, i.e. $E[X^j] \neq E[X]^j$. However, this is easily alleviated by the multiplicative property of independent random variables (3.12). Thus, we choose to evaluate the j -th power by taking the product of j independent evaluations of the optical depth,

$$T_r(x_a, x_b) := \sum_{j=0}^{\infty} T_r(x_a, x_b, j) = \sum_{j=0}^{\infty} \left(\frac{1}{j!} \prod_{k=1}^j \left(-\int_a^b \mu_r(x_t) dt \right) \right). \quad (6.68)$$

This trick has been used previously within the neutron transport literature for reciprocal estimation [116] which was later incorporated into graphics for photon mapping [95]. We will discuss the related problem of reciprocal estimation later in Chapter 8.

The individual summands in the infinite series ($T_r(x_a, x_b, j)$) resemble an unnormalized Poisson pmf, $p(j) = e^{-\lambda} \lambda^j / j!$ with mean $\lambda = \tau_r$ except the sign of each term alternates. We visualize this for two different choices of τ_r in Fig. 6.10.

Recursive formulation. By replacing the integral over the extinction with the real optical depth (τ_r) in Eq. (6.68) then expanding the sum and regrouping terms,

$$T_r(x_a, x_b) = \sum_{j=0}^{\infty} \prod_{k=0}^j \left(\frac{-\tau_r}{k} \right) = 1 - \frac{\tau_r}{1} \left[1 - \frac{\tau_r}{2} \left[\left[1 - \frac{\tau_r}{3} [\dots] \right] \right] \right], \quad (6.69)$$

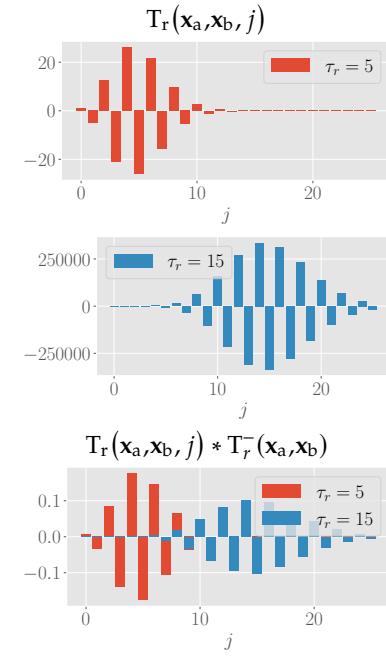


Figure 6.10: A visualization of the first few terms from the iterative power series (6.68) for two different choices of τ_r (top). On the bottom we plot both series' terms multiplied by the inverse of their respective real transmittances to directly compare their overall shape.

[116]: Booth (2007), ‘Unbiased Monte Carlo Estimation of the Reciprocal of an Integral’ [95]: Qin et al. (2015), ‘Unbiased Photon Gathering for Light Transport Simulation’

we can extract the recurrence relation,

$$\begin{aligned} T_r(x_a, x_b, j) &:= 1 - \frac{\tau_r}{j} T_r(x_a, x_b, j+1) \\ &= 1 - \frac{1}{j} T_r(x_a, x_b, j+1) \int_a^b \mu_r(x_t) dt, \end{aligned} \quad (6.70)$$

to arrive at an alternative recursive formulation for Eq. (6.68).

Unlike the prior Volterra integral formulation (6.19), the integration bounds for the optical depth in Eq. (6.70) do not change but instead the integrand gets modulated by the scaling factor $1/j$. We thus redefine transmittance as $T_r(x_a, x_b) = T_r(x_a, x_b, 1)$ to take into account that j gets incremented after every recursion. In addition, the evaluations for the optical depth are now *independent* of the recursive call which makes Eq. (6.70) *not* a "Volterra" integral formulation.

The process we used to derive Eq. (6.70) through analyzing the recurrence relation between subsequent terms in the power series is very similar to how general differential equations are solved using power series. However, since exponential transmittance is only a first-order differential equation, its solution (6.70) is mathematically trivial to obtain.

Taylor series is consistent. The Maclaurin series expansion, and by extension the Taylor series expansion, directly satisfies the condition of being a consistent formulation,

$$T_r(x_a, x_b) = \lim_{n \rightarrow \infty} \sum_{j=0}^n T_r(x_a, x_b, j), \quad (6.71)$$

since it is the result of an infinite series¹³. Thus, a Taylor series expansion is a valid construction for a consistent formulation completing ① in Fig. 1.1.

13: Assuming that the series expansion is convergent, of course.

6.3.2 Null-collision formulations

Similarly to the Volterra integral formulation (6.25), we can derive a null-collision formulation for the iterative and recursive forms of the exponential power series by applying control variates before taking the series expansion.

Iterative formulation. The null-collision form of the iterative formulation is derived as,

$$\begin{aligned} T_r(x_a, x_b) &= e^{-\int_a^b \mu_r(x_t) dt} = e^{-\int_a^b \mu_r(x_t) + \mu_t - \mu_t dt} \\ &= e^{-\int_a^b \mu_t dt} e^{\int_a^b \mu_t - \mu_r(x_t) dt} = T_t(x_a, x_b) e^X \\ &= T_t(x_a, x_b) \sum_{j=0}^{\infty} \left(\frac{1}{j!} \prod_{k=1}^j \left(\int_a^b \mu_n(x_t) dt \right) \right). \end{aligned} \quad (6.72)$$



Figure 6.11: A visualization of the first few terms from the null iterative power series (6.72) for two different choices of τ_t where one is non-bounding ($\tau_t = 3$) and the other is bounding ($\tau_t = 5$). The real optical depth ($\tau_r = 4$) is chosen such that it is equidistant to both choices of τ_t . The bounding τ_t naturally exhibits less variation than the non-bounding one.

Recursive formulation. By applying the same expansion to Eq. (6.72) as we did for the prior formulation in Eq. (6.70),

$$\begin{aligned}
 T_r(\mathbf{x}_a, \mathbf{x}_b) &= T_t(\mathbf{x}_a, \mathbf{x}_b) (T_n^-(\mathbf{x}_a, \mathbf{x}_b)) \\
 &= T_t(\mathbf{x}_a, \mathbf{x}_b) \sum_{j=0}^{\infty} \prod_{k=0}^j \left(\frac{\tau_n}{j} \right) \\
 &= T_t(\mathbf{x}_a, \mathbf{x}_b) \left(1 + \frac{\tau_n}{1} \left[1 + \frac{\tau_n}{2} \left[1 + \frac{\tau_n}{3} [\dots] \right] \right] \right) \\
 &= T_t(\mathbf{x}_a, \mathbf{x}_b) \left(1 + \frac{\tau_n}{j} (T_n(\mathbf{x}_a, \mathbf{x}_b, j+1))^{-1} \right) \\
 &= T_t(\mathbf{x}_a, \mathbf{x}_b) + \frac{\tau_n}{j} T_t(\mathbf{x}_a, \mathbf{x}_b) (T_n^-(\mathbf{x}_a, \mathbf{x}_b, j+1)) \\
 &= T_t(\mathbf{x}_a, \mathbf{x}_b) + \frac{\tau_n}{j} T_r(\mathbf{x}_a, \mathbf{x}_b, j+1),
 \end{aligned} \tag{6.73}$$

we arrive at the null-collision formulation of the recursive form of the exponential power series.

Alternating sign. Unlike the base formulations (Eqs. (6.68) and (6.70)), there are no negative signs within the null-collision formulations (Eqs. (6.72) and (6.73)). Since we evaluate $\mu_n(\mathbf{x}_t)$ instead of the negative real extinction the individual summands will always be positive as long as $\mu_t > \mu_r(\mathbf{x}_t)$.

In Fig. 6.11 we visualize the difference between the terms of Eq. (6.72) for the same constant real density ($\tau_r = 4$) while varying the choice of the combined extinction μ_t between bounding and non-bounding. In the presence of negative null extinctions, the terms in the series expansion drastically flip between positive and negative values. This implies that the choice of introducing fictitious density to the power series formulations is, in essence, a variance reduction technique that avoids excess variation due to alternating signs. Similarly, it implies that having a tightly bounding combined extinction is preferred in reducing the amount of work needed to evaluate these estimators.

6.3.3 Re-deriving ratio tracking

In Sec. 6.2.4 we reviewed how to derive all prior tracking based transmittance estimators using the null-collision Volterra integral formulation. Estimators for the Volterra formulation are naturally random walks which start from \mathbf{x}_a and sample towards \mathbf{x}_b . By its definition, the integral bounds

for the Volterra formulation change at every recursion whereas the integral bounds in the recursive power series formulation (6.70) remain the same. Meaning we most likely will not be able to derive all prior tracking-based transmittance estimators exactly within the power series paradigm, however, we will be able to re-derive one of them.

Pseries ratio. Traditional ratio tracking (6.12) evaluates transmittance after sampling an entire path by taking j free-flight samples. Thus, we can consider ratio tracking as an estimator which evaluates a single-term within the power series by indirectly choosing to sample a path of length j through Russian Roulette decisions. These decisions take the form of accumulating exponential distance samples until \mathbf{x}_b has been reached. Mathematically, ratio tracking takes the form of a running product of the null albedo.

It is well known that the distribution of the number of exponential distances taken to reach a certain distance follows a Poisson distribution. We even used this fact previously to derive an analytical equation for the variance of ratio tracking within a homogenous media (see Sec. 6.2.5). Additionally, the final mathematical form of ratio tracking has no direct dependence on the sampled distances themselves, only the ratios between null and total extinctions. This implies that by carefully importance sampling the iterative null power series (6.72), we can rederive ratio tracking as a power series estimator.

We accomplish this by first importance sampling a path length j from the Poisson PMF $p(j) \propto e^{-\tau_t} \frac{\tau_t^j}{j!}$, resulting in,

$$\begin{aligned} T_r(\mathbf{x}_a, \mathbf{x}_b)_{\text{PRat}} &= T_t(\mathbf{x}_a, \mathbf{x}_b) \sum_{j=0}^{\infty} \left(\frac{1}{j!} \prod_{k=1}^j \left(\int_a^b \mu_n(\mathbf{x}_t) dt \right) \right) \\ &= \frac{T_t(\mathbf{x}_a, \mathbf{x}_b)}{p(j)} \left(\frac{1}{j!} \prod_{k=1}^j \left(\int_a^b \mu_n(\mathbf{x}_t) dt \right) \right) \\ &= \prod_{k=1}^j \frac{\int_a^b \mu_n(\mathbf{x}_t) dt}{\tau_t}. \end{aligned} \quad (6.74)$$

We then importance sample the integrand uniformly with respect to the integration bounds,

$$T_r(\mathbf{x}_a, \mathbf{x}_b)_{\text{PRat}} := \prod_{k=1}^j \frac{\int_a^b \mu_n(\mathbf{x}_t) dt}{\tau_t p(\mathbf{x}_t)} = \prod_{k=1}^j \alpha_n(\mathbf{x}_t). \quad (6.75)$$

which mathematically simplifies to traditional ratio tracking (6.12).

Unlike traditional Volterra based ratio-tracking, the integral samples are not ordered within the power series formulation and are distributed randomly. While these estimators are mathematically equivalent, due to the nature of how they sample the integrand they will behave differently when entire light paths are correlated. We discuss this more in detail in Sec. 6.5.

Converting between Volterra and power series formulations. In our original work [110] we provided a theoretical argument for how to convert between the power series formulation and the Volterra formulation. However, the process for doing so is only applicable to the exponential function due to its convenient properties. The same process does not generalize cleanly to other nonlinear functions, so we choose to point the interested reader to the original work for that discussion. It has no theoretical significance to the rest of this dissertation so we will not replicate it here.

[110]: Georgiev et al. (2019), ‘Integral Formulations of Volumetric Transmittance’

6.3.4 Novel transmittance estimators

We will now introduce some novel transmittance estimators which are derived from the recursive null formulation of the exponential power series (6.73). These estimators fulfill step ③ from Fig. 1.1.

Pseries cumulative (PCume). Efficient implementations of path tracing employ Russian Roulette termination based on the current path throughput to determine when to stop tracing a path. Inspired by this, our first estimator employs a similar strategy to determine when to stop evaluating the infinite power series by keeping track of the *cumulative* weight that gets applied to the recursive call,

$$W(j) := W(j-1) * \frac{\mu_n(x_j)}{j P_{\text{acc}}(j-1)}, \quad (6.76)$$

where $W(j-1)$ is the weight from the previous recursive call, $\mu_n(x_j)$ is the j th random evaluation of the null extinction, and $P_{\text{acc}}(j)$ is the Russian Roulette continuation probability,

$$P_{\text{acc}}(j) := \text{Min}(|W(j)|, 1). \quad (6.77)$$

This results in a prefix-sum estimator described using the recursive null formulation as,

$$\langle T_r(x_a, x_b, j) \rangle_{\text{PCume}} := \begin{cases} W(j), & \text{otherwise} \\ W(j) + \langle T_r(x_a, x_b, j+1) \rangle_{\text{PCume}}, & \text{with prob. } P_{\text{acc}}(j), \end{cases} \quad (6.78)$$

where $T_r(x_a, x_b) = T_t(x_a, x_b) T_r(x_a, x_b, 0)$ and $W(0) = 1^{14}$. We provide pseudocode for our PCume estimator in Algorithm 8.

14: Note: the combined transmittance is multiplied in at the end.

Algorithm 8: Pseries cumulative transmittance estimator

```

1  $t = a$ 
2  $T = 0$ 
3  $W = j = 1$ 
4 while True do
5    $T = T + W$ 
6    $t = \xi * (b - a)$ 
7    $W = W * \frac{\mu_n(x_t)}{j}$ 
8    $P_{\text{acc}}(j) = \text{Min}(1, W)$ 
9   if  $\xi > P_{\text{acc}}(j)$  then
10    | break
11    $W = W/P_{\text{acc}}(j)$ 
12    $j = j + 1$ 
13 return  $T * e^{-\mu_t(b - a)}$ 

```

Pseries CMF (PCMF). Pseries cumulative is generally a decent performing estimator, however, it has two key weaknesses stemming from how it chooses to terminate. Depending on the specific extinction function and choice of μ_t , pseries cumulative tends to either be too costly or terminates too early. Our more pragmatic estimator, pseries CMF (PCMF), is designed to directly alleviate these issues.

If we plot the individual summands of the null power series (see Fig. 6.12), the resulting distribution takes the shape of an unnormalized Poisson distribution whose mean is $\tau_n = \int_a^b \mu_n(x_t) dt$. A prefix-sum style estimator, like pseries cumulative, will ideally accumulate the majority of this unnormalized Poisson distribution and then employ Russian roulette termination to stop evaluating the power series when it reaches the rightmost "tail" of the distribution. This can be considered analogous to path tracing with NEE terminating a path when the throughput of evaluating the next path vertex is expected to be insignificant to the overall path contribution.

Terminating based on the cumulative weight is a strategy that is designed to make the estimator terminate while evaluating a tail of the distribution, but it does not guarantee that the estimator will terminate while evaluating the *rightmost* tail. If $\mu_n(x_t)$ varies such that it can become near zero, then there is a significant chance that pseries cumulative will terminate *too* early resulting in high variance.

Additionally, if τ_n is large, it may take pseries cumulative an excessive amount of time for the $\frac{1}{j!}$ term to overpower the running product of the optical depth estimates (6.12). Meaning, when the combined extinction is loose, pseries cumulative tends to be significantly more costly than it needs to be.

Our pseries CMF estimator attempts to alleviate these two issues by pragmatically choosing to only apply Russian Roulette termination after we have already accumulated a significant portion of the expected "throughput" of the null Poisson distribution. We do so by realizing that the Poisson distribution dictated by μ_t will always be an *overestimate* of

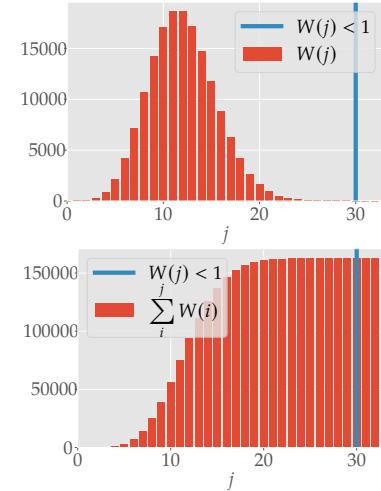


Figure 6.12: We visualize an example of a situation where the PCume estimator (6.78) is too costly by plotting the contributions of each term in the power series (top, red) for a constant density medium ($\tau_r = 3$, $\tau_t = 15$) alongside where its Russian Roulette termination (6.77) starts having a non-zero probability (blue vertical line.) We additionally visualize the running sum on the bottom indicating that PCume should employ more aggressive Russian Roulette termination.

the unknown null Poisson distribution when the combined extinction is bounding ($\mu_t > \mu_r(\mathbf{x})$). By choosing to apply Russian Roulette termination only after we have accumulated a significant portion of the combined Poisson distribution we will have also indirectly accumulated a significant portion of the null distribution.

This translates into modifying the Russian Roulette continuation probabilities from pseries cumulative to become,

$$P_{\text{acc}}(j) := \begin{cases} 1, & e^{-\tau_t} \sum_{k=0}^j \frac{\tau_t^k}{k!} < \epsilon \\ \frac{\tau_t}{j}, & \text{otherwise,} \end{cases} \quad (6.79)$$

where ϵ is a user-defined parameter determining the aggressiveness of the Russian Roulette termination. We typically use $\epsilon = 0.99$ unless specified otherwise. While Eq. (6.79) involves computing an additional power series for every j , the series can be evaluated incrementally so it should not impact performance.

The weights and overall estimator for PCMF remain unchanged from PCume, thus, PCMF can also be mathematically described as,

$$\langle T_r(\mathbf{x}_a, \mathbf{x}_b, j) \rangle_{\text{PCMF}} := \begin{cases} W(j), & \text{otherwise} \\ W(j) + \langle T_r(\mathbf{x}_a, \mathbf{x}_b, j+1) \rangle_{\text{PCMF}}, & \text{with prob. } P_{\text{acc}}(j). \end{cases} \quad (6.80)$$

We provide pseudocode for our PCMF estimator in Algorithm 9.

Algorithm 9: Pseries CMF transmittance estimator

```

1  $t = a$ 
2  $T = \text{CMF} = 0$ 
3  $W = W_t = j = 1$ 
4  $\epsilon = 0.99$ 
5 while True do
6    $T = T + W$ 
7    $\text{CMF} = \text{CMF} + W_t * e^{-\mu_t(b-a)}$ 
8    $t = \xi * (b - a)$ 
9    $W = W * \frac{\mu_t(\mathbf{x}_t)}{j}$ 
10   $P_{\text{acc}}(j) = \text{Min}(1, \frac{\mu_t}{j})$ 
11  if  $\text{CMF} < \epsilon$  then
12     $P_{\text{acc}}(j) = 1$ 
13  if  $\xi > P_{\text{acc}}(j)$  then
14    break
15   $W = W / P_{\text{acc}}(j)$ 
16   $j = j + 1$ 
17   $W_t = W_t * \frac{\mu_t}{j}$ 
18 return  $T * e^{-\mu_t(b-a)}$ 

```

6.4 Hypercube formulation

In this section we introduce another integral formulation for exponential transmittance which draws inspiration from analogies between our previous two integral formulations and the various forms of the traditional rendering equation (see Chapter 2). Based on these analogies (6.4.1), we then introduce a hypercube formulation for the exponential power series in Sec. 6.4.2 as well as its null-collision reformulation in Sec. 6.4.3. Using the null form, we then derive novel transmittance estimators which rely on MIS in Sec. 6.4.4.

6.4.1 Analogies to rendering formulations

The traditional rendering equation (2.22) takes the form of a Fredholm integral equation of the second kind (2.24), which is a recursive formulation where the quantity we are evaluating, i.e. the radiance, appears within the integrand. A Fredholm integral equation differs from a Volterra integral equation only by the fact that the integration bounds used in the recursive evaluation remain constant within a Fredholm integral, while they shrink within a Volterra formulation¹⁵.

However, due to the similarity in their mathematical structures, we can derive estimators for the Volterra formulation of transmittance which mimic the behaviors of prior solutions to the rendering equation. In fact, we have already observed this in our overview of prior work and the introduction of our pseries cumulative transmittance estimator (6.78).

Prior techniques. Ratio tracking is a single-term estimator which traverses a path segment by sampling exponential distances until it passes the segment’s endpoint. This is analogous to naive unidirectional path tracing (5.4) which continues to sample bounces for a path until an emitter is encountered.

Next-flight estimation samples the same path as ratio tracking, except instead of computing the contribution of the entire path, it performs NEE to accumulate the contributions of all intermediate paths up until the endpoint. This is analogous to unidirectional path tracing with NEE (Eq. (5.7)).

Our estimators. Pseries cumulative (6.78) is designed around utilizing similar throughput-based Russian Roulette termination that more intelligent implementations of unidirectional path-tracing with NEE employ. Pseries CMF (6.80) is then designed around the idea of delaying the application of Russian Roulette to avoid excessive variance from premature termination. Deciding not to use Russian Roulette early on is another technique that is occasionally employed in modern implementations of unidirectional path-tracing with NEE [18].

¹⁵: While we choose to write the traditional rendering equation (2.22) as an integral over a hemisphere dictated by a surface normal (\mathbb{H}_n) which *can* change, the rendering equation can be more generally written as an integral over the entire sphere of directions (which remains constant). Since we do not perform a deep dive into specific BSDF’s within this dissertation, we assume for the purpose of introducing surface-only theory that there are no refractions, thus the hemisphere beneath the surface has zero contributions and can be ignored.

[18]: Pharr et al. (2016), *Physically Based Rendering*

Our adaptive importance sampling technique (6.2.6) could be considered analogous to either an on-the-fly style of path guiding or a form of Dwivedi sampling [117, 118] which intelligently "biases" path sampling strategies towards higher contribution regions in order to terminate the paths sooner¹⁶.

There is a significant overlap between solutions for the rendering equation and solutions for the integral formulations of transmittance. This implies that the more advanced techniques, such as the bidirectional family of solutions to the rendering equation (see Sec. 5.2), should also have analogous applications to the problem of transmittance estimation. Thus, we additionally propose an integral formulation for exponential transmittance derived from the exponential power series that is analogous to the path integral formulation of the rendering equation (2.36).

6.4.2 Base formulation

We derive an analogous path integral formulation for exponential transmittance by starting from the power series formulation (6.68),

$$T_r(x_a, x_b) := \sum_{j=0}^{\infty} \left(\frac{1}{j!} \prod_{k=1}^j \left(- \int_a^b \mu_r(x_t) dt \right) \right) = \sum_{j=0}^{\infty} \left(\frac{1}{j!} T_j \right) \quad (6.81)$$

and redefine the contribution due to the optical depth for term j as T_j . Since T_j is the product of j integrals over the optical depth, we can rewrite it as the iterated integral of the product (see proof in Eq. (6.89)),

$$T_j = \prod_{k=1}^j \left(- \int_a^b \mu_r(x_t) dt \right) = \int_a^b \dots \int_a^b \left[\prod_{k=1}^j -\mu_r(x_k) \right] dx_j \dots dx_1. \quad (6.82)$$

Thus, we can rewrite the power series formulation as,

$$T_r(a, b) = \sum_{j=0}^{\infty} \frac{1}{j!} \int_0^t \dots \int_0^t \left[\prod_{k=1}^j -\mu_r(x_k) \right] dx_1 \dots dx_k, \quad (6.83)$$

where each term within the power series spans over a j -dimensional hypercube, $\mathbb{H}_j = [a, b]^j$. This allows us to replace the iterated integral with an integral over the j -dimensional hypercube,

$$T_r(a, b) = \sum_{j=0}^{\infty} \frac{1}{j!} \int_{\mathbb{H}_j} \left[\prod_{k=1}^j -\mu_r(x_k) \right] dx_1 \dots dx_k, \quad (6.84)$$

where the power series of the exponential then becomes a summation over all hypercubes. We then replace the summation over every \mathbb{H}_j with an integral over the space of all hypercubes,

$$T_r(a, b) := \int_{\mathbb{H}} f(\bar{x}) d\bar{x}; \quad \mathbb{H} := \bigcup_{j=0}^{\infty} \mathbb{H}_j, \quad (6.85)$$

[117]: Dwivedi (1982), 'A New Importance Biasing Scheme for Deep-Penetration Monte Carlo'

[118]: Meng et al. (2016), 'Improving the Dwivedi Sampling Scheme'

16: The term "biases" here does not mean that the algorithm itself is biased, rather that we employ an importance sampling scheme which purposefully does not match the medium's physical distributions.

where $\bar{\mathbf{x}} = (\mathbf{x}_1, \dots, \mathbf{x}_j)$ is a sequence of locations where the extinction gets evaluated. This is analogous to a light path within the path integral formulation for light transport. The integral contribution $f(\bar{\mathbf{x}})$ for a path of length j in the context of evaluating transmittance is then defined to be,

$$f(\bar{\mathbf{x}}) := \frac{1}{j!} \prod_{k=1}^j -\mu_r(\mathbf{x}_k), \quad (6.86)$$

where the measure becomes,

$$d\bar{\mathbf{x}} := \prod_{k=1}^j d\mathbf{x}_k. \quad (6.87)$$

similarly to the path integral formulation for light transport 2.6¹⁷.

Similarly to the traditional path integral formulations used in rendering (2.36), our hypercube formulation (6.85) expresses transmittance as a pure integration problem where the integral runs over all hypercubes $\mathbb{H}_j[a, b]^j$ and we consider entire sequences of extinction evaluations $\bar{\mathbf{x}}$ as analogous paths.

17: The zeroth hypercube (\mathbb{H}_0) for $j = 0$ simplifies to a single point, thus $\int_{\mathbb{H}_0} f(\bar{\mathbf{x}}) d\bar{\mathbf{x}} = 1$, which is equal to the zeroth summand in the power series formulation (6.68).

Proof of Eq. (6.82). Given the product of two independent integrals which are both guaranteed to integrate to a constant, we can replace one with its expected scalar value $\int_a^b \mu_r(\mathbf{x}_t) dt = \tau_r$,

$$\left(\int_a^b \mu_r(\mathbf{x}_t) dt \right) \left(\int_a^b \mu_r(\mathbf{x}_t) dt \right) = \tau_r \left(\int_a^b \mu_r(\mathbf{x}_t) dt \right). \quad (6.88)$$

Then, by using the linear property of integration,

$$\begin{aligned} \tau_r \left(\int_a^b \mu_r(\mathbf{x}_t) dt \right) &= \int_a^b \tau_r \mu_r(\mathbf{x}_t) dt \\ &= \int_a^b \left(\int_a^b \mu_r(\mathbf{x}_s) ds \right) \mu_r(\mathbf{x}_t) dt \\ &= \int_a^b \int_a^b \mu_r(\mathbf{x}_s) \mu_r(\mathbf{x}_t) ds dt, \end{aligned} \quad (6.89)$$

we see that the product of two finite independent integrals is equal to the iterated integral of the product of their integrands allowing us to derive Eq. (6.82).

6.4.3 Null formulation

The hypercube formulation can be rederived while employing the null-collision paradigm by starting from the null formulation of the power series (6.72) and following the same sequence of operations we performed

in Sec. 6.4.2. Doing so results in the null hypercube formulation,

$$T_r(a, b) := \int_{\mathbb{H}} f_n(\bar{x}) d\bar{x}; \quad (6.90)$$

where the measure remains the same as Eq. (6.87), and the null contribution function is defined as,

$$f_n(\bar{x}) := \frac{T_t(x_a, x_b)}{j!} \prod_{k=1}^j \mu_n(x_k), \quad (6.91)$$

for a sequence of extinction evaluations of length j .

6.4.4 Hypercube estimation

Representing the power series formulation of transmittance as an integral over the space of all hypercubes enables us to consider prior transmittance estimators as analogies to path sampling strategies for light transport. Instead of formulating transmittance estimators by incrementally sampling additional extinction evaluations, we can now design estimators by combining different sampling strategies together for the same sequence of extinction query locations (\bar{x}).

A single-term estimator for Eq. (6.90) has the well-known general form

$$\langle T_r(x_a, x_b) \rangle := \frac{f_n(\bar{x})}{p(\bar{x})}, \quad (6.92)$$

where \bar{x} represents a point sampled within the k -dimensional hypercube, \mathbb{H}_k , with corresponding probability density $p(\bar{x})$. Within the hypercube formulation the only difference between transmittance estimators is due to the choice of PDFs for their corresponding sampling techniques.

Similarly to bidirectional path tracing (Sec. 5.2), we explore a family of techniques that construct sequences of points, $\bar{x} = x_1 \dots x_r x_{r+1} \dots x_j$, where the first r points are sampled via a random walk starting from x_a while the remaining $j - r$ points are sampled in the opposite direction starting from x_b . We denote these sampling strategies by the pair (r, j) where $x_0 = x_a$ and $x_{j+1} = x_b$. The PDF for technique (r, j) becomes,

$$\begin{aligned} p_{r,j}(\bar{x}) &:= \frac{1}{j!} \left(\prod_{i=1}^r \mu_t T_t(x_{i-1}, x_i) \right) \left(\prod_{i=r+1}^j \mu_t T_t(x_i, x_{i+1}) \right) \\ &= \frac{1}{j!} T_t(x_a, x_r) T_t(x_{r+1}, x_b) \left(\prod_{i=1}^j \mu_t \right), \end{aligned} \quad (6.93)$$

where the coordinates of \bar{x} are sampled in sorted order, i.e. in a j -simplex of \mathbb{H}_j which has a volume of $\frac{|\mathbb{H}_j|}{j!}$. The Jacobian, $\frac{1}{j!}$, converts the PDF measure to the measure of the hypercube.

Hypercube ratio and next-flight. Plugging $(j+1, j)$ into Eq. (6.93) and then simplifying will result in the same probability distribution that pseries-ratio uses to importance sample the iterative power series. Once $p_{j+1,j}(\bar{\mathbf{x}})$ and Eq. (6.91) are plugged into Eq. (6.92), the resulting estimator simplifies to traditional ratio tracking (6.12). Next-flight uses techniques (i, i) for all $i \in [0, j]$ to construct prefix sets $\bar{\mathbf{x}}_i = \mathbf{x}_0 \dots \mathbf{x}_i$ with PDFs $p_{i,i}(\bar{\mathbf{x}}_i)$ which share common prefixes. Accumulating the resulting estimates results in the traditional next-flight estimator (6.15).

Naive bidirectional next-flight. Next-flight is a unidirectional estimator which incrementally evaluates prefix sets $\bar{\mathbf{x}}_i$ from $\bar{\mathbf{x}}_0$ to $\bar{\mathbf{x}}_j$. Since transmittance is reciprocal ($T_r(\mathbf{x}_a, \mathbf{x}_b) = T_r(\mathbf{x}_b, \mathbf{x}_a)$), we can choose to evaluate next-flight starting from \mathbf{x}_b instead, which is similar to the difference between path tracing and light tracing (Sec. 5.1). Choosing to estimate $T_r(\mathbf{x}_b, \mathbf{x}_a)$ in the reverse direction results in a variant of the next-flight estimator which evaluates postfix sets instead of prefix sets, i.e. the reverse direction estimator considers techniques $(0, i)$ for all $i \in [0, j]$. Since the distribution of extinction evaluations arising from taking a sequence of exponential distances resembles that of a uniform distribution, this allows us to sample one set of points to evaluate both the set of prefixes and the set of postfixes.

We can naively combine these two estimators by applying simple averaging,

$$\begin{aligned} \langle T_r(\mathbf{x}_a, \mathbf{x}_b) \rangle_{\text{BIDir-NF}} &:= \frac{1}{2} (\langle T_r(\mathbf{x}_a, \mathbf{x}_b) \rangle_{\text{NF}} + \langle T_r(\mathbf{x}_a, \mathbf{x}_b) \rangle_{\text{Rev-NF}}) \\ &= \frac{1}{2} \left(T_t(\mathbf{x}_a, \mathbf{x}_b) + \sum_{j=1}^{n-1} T_t(\mathbf{x}_j, \mathbf{x}_b) \prod_{k=1}^j \alpha_n(\mathbf{x}_k) \right) + \\ &\quad \frac{1}{2} \left(T_t(\mathbf{x}_a, \mathbf{x}_b) + \sum_{j=1}^{n-1} T_t(\mathbf{x}_a, \mathbf{x}_{n-j}) \prod_{k=1}^j \alpha_n(\mathbf{x}_{n-k}) \right), \end{aligned} \quad (6.94)$$

to arrive at a naively bidirectional form of the next-flight estimator (6.15). ?? effectively allows for evaluating two different invocations of the next-flight estimator for the cost of one tracking, resulting in less overall variance and better efficiency¹⁸. However, we can easily do better than simple averaging by employing MIS to combine these and other strategies.

Unidirectional-MIS. The hypercube formulation allows for combining traditional ratio tracking and next-flight via MIS in a similar way to how path tracing allows for combining unidirectional sampling and next-event estimation. Our unidirectional-MIS estimator performs a random walk starting from \mathbf{x}_a then constructs a next-flight estimate with PDF $p_{i,i}(\bar{\mathbf{x}}_i)$ at every step which gets weighted by the balance-heuristic [119] and also accounts for the corresponding ratio-tracking technique with PDF $p_{i+1,i}(\bar{\mathbf{x}}_i)$. Upon reaching \mathbf{x}_b , a ratio tracking estimate with PDF $p_{j+1,j}(\bar{\mathbf{x}}_j)$ is accumulated, where the MIS weight also accounts for the next-flight technique with PDF $p_{j,j}(\bar{\mathbf{x}}_j)$. We provide pseudocode for our unidirectional-MIS estimator in Algorithm 10.

18: Ratio tracking and track length estimators are perfectly symmetric when evaluated in either the forward or backward directions using the same $\bar{\mathbf{x}}$. Thus, formulating bidirectional estimators which only combine instances of those techniques would be redundant.

[119]: Veach et al. (1995), ‘Bidirectional Estimators for Light Transport’

Algorithm 10: Unidirectional-MIS transmittance estimator

```

1  $t = a$ 
2  $T = 0$ 
3  $W = 1$ 
4 while  $t < b$  do
5    $C = W * e^{-\mu_t(b-t)}$ 
6    $P_{NF} = 1$ 
7    $P_{Rat} = e^{-\mu_t(b-t)}$ 
8    $t = t - \frac{\log(1-\xi)}{\mu_t}$ 
9   if  $t \geq b$  then
10    |  $T = T + C * (\frac{P_{NF}}{P_{NF}+P_{Rat}} + \frac{P_{Rat}}{P_{NF}+P_{Rat}})$ 
11    | break
12    $T = T + C * (\frac{P_{NF}}{P_{NF}+P_{Rat}})$ 
13    $W = W * \frac{\mu_n(x_t)}{\mu_t}$ 
14 return  $T$ 

```

Full bidirectional-MIS. We can also combine all techniques (r, j) for every point \bar{x}_j via MIS, similarly to how bidirectional path tracing (BPT) combines all possible vertex connection techniques for a given light transport path [119]. Our implementation samples coordinates by sampling two different trackings starting from x_a and x_b , respectively, and then constructs points \bar{x}_j by concatenating coordinate prefixes and postfixes. This is analogous to BPT which connects pairs of vertices on sensor (prefix) and emitter (postfix) paths.

We refer the interested reader to our original work [110] for our implementation of the bidirectional-MIS transmittance estimator. For brevity, we omit its implementation from this dissertation since its computational cost makes it impractical in practice. Deriving an efficient bidirectional transmittance estimator is an open area of future work.

[119]: Veach et al. (1995), ‘Bidirectional Estimators for Light Transport’

[110]: Georgiev et al. (2019), ‘Integral Formulations of Volumetric Transmittance’

6.5 Discussion, results, and implementation

In this section, we summarize our integral formulations for transmittance and the various estimators we have derived from them throughout this chapter. We then benchmark these estimators on a range of scenes, summarize the results, and then conclude this chapter with a short discussion.

6.5.1 Summary of contributions

We have introduced three different integral formulations for exponential transmittance which are amenable to unbiased estimation through the direct application of Monte Carlo methods. We have done so by introducing two different consistent formulations (①): the IVP representation and Maclaurin series expansion of the non-linear exponential function. Converting the

exponential function into an IVP results in the Volterra formulation (6.25), while taking the Maclaurin series expansion results in the power series (6.72) and hypercube formulations (6.90). All of which convert the problem of estimating the non-linear exponential function into formulations where only the product of linear terms have to be estimated¹⁹.

We have analyzed the most optimal importance sampling strategy (2) for the Volterra formulation and successfully devised zero-variance variants of all prior tracking-based transmittance estimators (ratio, track length and next-flight) in Sec. 6.2.5. However, we made the realization that this zero-variance importance sampling technique cannot be employed in practice because the real extinction function is not analytic. Thus, we still need to explore techniques for devising efficient Monte Carlo solutions (3).

In Sec. 6.2.4 we rederived all transmittance estimators which appeared within the graphics literature prior to this work [110] directly from our Volterra formulation. Inspired by our zero-variance analysis we proposed adaptive variants for each in Sec. 6.2.6 which we have empirically demonstrated in Fig. 6.5 and Fig. 6.8 that they can be more efficient than their traditional counterparts.

Our power series formulations (Sec. 6.3) express transmittance as an infinite sum of the product of linear terms. From this formulation we have rederived ratio tracking which takes the form of a single-term estimator. We additionally devised two novel recursive prefix-sum estimators that only differ in their termination probabilities: PCume (6.78) and PCMF (6.80).

Existing tracking-based estimators can be expressed as sampling techniques in our hypercube integral formulation for transmittance. We proposed two novel estimators that combine such techniques using MIS. One which combines ratio tracking and next-flight in a unidirectional manner, while the other combines all instances of next-flight and ratio in a bidirectional manner (Sec. 6.4.4). Additionally, we demonstrated that a bidirectional form of next-flight can be formulated which combines two transmittance estimates for the price of one (6.94).

6.5.2 Numerical experiments

We compare the efficiency of the various transmittance estimators on both canonical configurations and real scenes. To isolate variance due to only transmittance estimation, we employ a specialized path tracer that follows all surface reflection and refraction branches while avoiding all forms of Russian Roulette external to evaluating transmittance which was implemented in PBRT [18]. Unless specified otherwise, the number of medium extinction lookups (evaluations of $\mu_r(\mathbf{x})$) are kept approximately constant in all our rendering comparisons while we report the root mean squared error (RMSE) and render time (in seconds) of the images. We utilize equal-lookup comparisons because they are renderer-agnostic and arguably more representative for large, complex scenes where medium lookups dominate the render time [104, 111]. All comparisons of including the tracking-based techniques employ their traditional variant rather than our

19: All of the non-linear terms in the Volterra, power series, and hypercube formulations are analytic and thus do not have to be estimated.

[110]: Georgiev et al. (2019), ‘Integral Formulations of Volumetric Transmittance’

[18]: Pharr et al. (2016), *Physically Based Rendering*

[104]: Novák et al. (2014), ‘Residual Ratio Tracking for Estimating Attenuation in Participating Media’

[111]: Novák et al. (2018), ‘Monte Carlo Methods for Physically Based Volume Rendering’

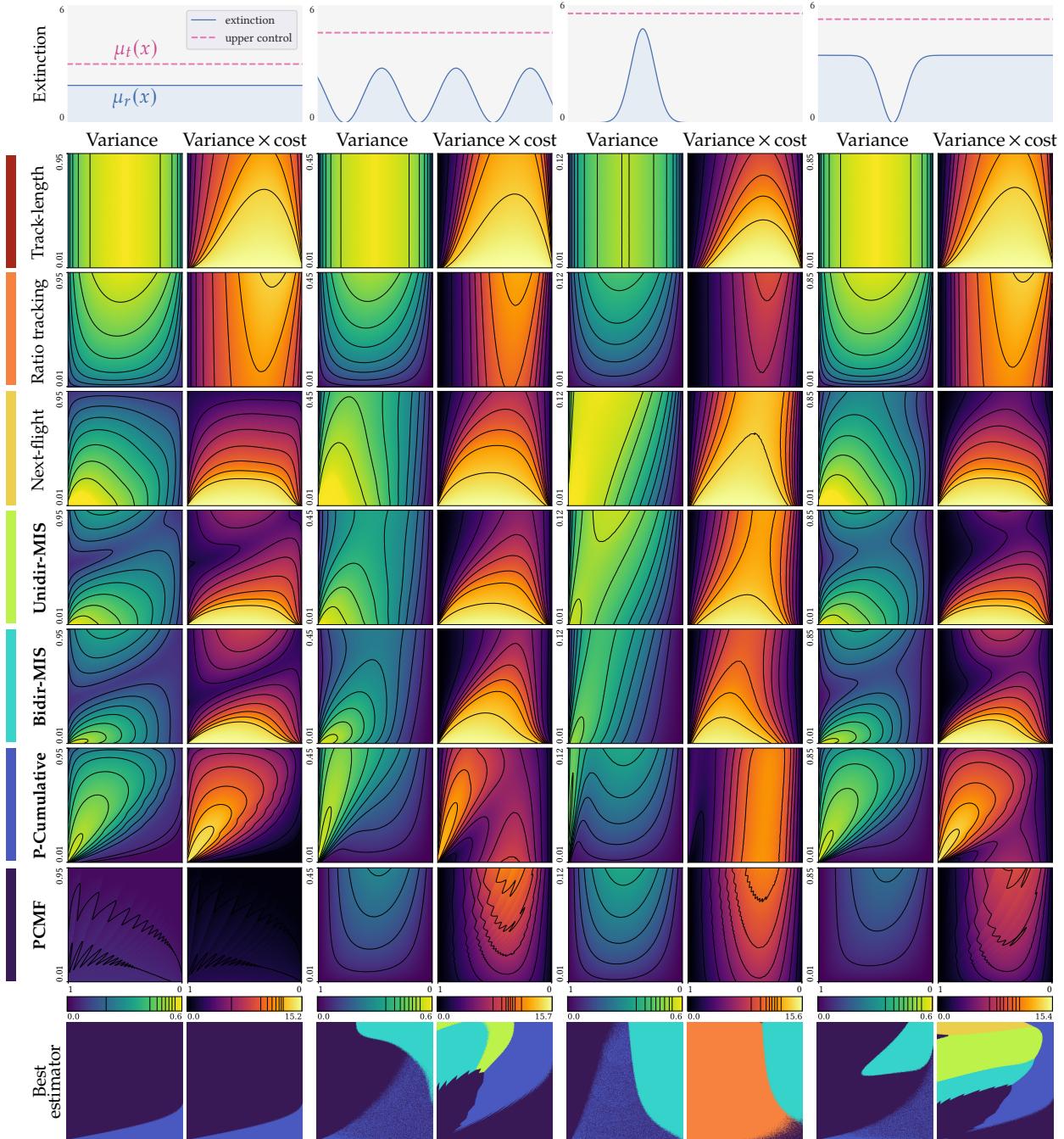


Figure 6.13: Canonical comparison of transmittance estimators (ours are in bold) for several simple extinction functions (top row). Each pixel in every 2D plot represents a separate experiment with a unique combination of a real extinction function $\mu_r(x)$ and total extinction μ_t . In each plot, we scale $\mu_r(x)$ exponentially along the x-axis so that the transmittance drops linearly from 1 to 0. Along the y-axis we scale μ_t so that the optical depth efficiency $\frac{\tau_r}{\tau_t}$ increases linearly from 0.01 to an extinction-dependent upper bound. For each pixel, we run the estimator several times to compute its variance and cost (i.e. number of μ_r lookups). We then plot the variance and cost \times variance (i.e. inverse efficiency, work-normalized variance), where lower is better. In the bottom-row plots we show the best estimator for every pixel where each estimator is color-coded. Our PCMF estimator performs best overall thanks to trading slightly higher cost for lower variance. A notable exception is the Gaussian extinction function, where the total extinction is naturally always loose and the less costly ratio tracking estimator performs marginally better.

adaptive techniques. For comparisons regarding our adaptive techniques see Fig. 6.8 and Fig. 6.9.

Canonical configurations. In Fig. 6.13 we study the performance of seven estimators across four extinction functions within a canonical configuration previously employed in prior work [104, 111]. We plot the estimators' variance and variance \times cost (i.e. work-normalized variance), where lower values are better. In each 2D plot, we scale the real extinction function uniformly such that the transmittance varies linearly from 1 to 0, left to right. Vertically we scale μ_t such that the total optical depth efficiency $\frac{\tau_r}{\tau_t}$ [120] increases linearly from 0.01 to an extinction-dependent upper bound, bottom-to-top. The plots convey the relative differences in performance between estimators on the full range of possible transmittance values and *bounding* total extinctions: from tight (top) to almost infinitely loose (bottom).

The performance of the existing estimators (top three rows in Fig. 6.13) has been studied in prior work²⁰ [104, 111]. As μ_t increases, ratio tracking's Poisson PMF becomes better at importance sampling the transmittance sum, making up for the increased cost to improve the estimator's overall efficiency. The track length estimator's variance is never smaller than that of ratio tracking, but its cost can be lower due to its early termination, resulting in occasionally being more efficient when τ_r is large and μ_t is tight (top right regions of the plots). These observations match our analytical variance analysis (6.34). The next-flight estimator has been known to perform better when μ_t is tight [111]. In such cases, the mean of its termination Poisson PMF is larger than that of the (unnormalized) transmittance PMF, so next-flight terminates after accumulating the majority of the transmittance contribution. For loose μ_t , termination happens too early for the estimator to be efficient, and when μ_t is infinitely loose, the two PMF means coincide where only half the contribution is accumulated on average. The estimator is thus bound to produce high variance when μ_t is naturally loose, e.g. with Gaussian-shaped extinctions.

Our PCume prefix-sum estimator is designed to delay termination until its cumulative weight (i.e. the analog of path throughput in surface rendering) drops below one (6.77). This typically occurs well past the transmittance PMF mean, resulting in better performance than the next-flight estimator for the majority of cases. The estimator is still susceptible to some higher variance when the total extinction is loose and when $\tau_n = \tau_t - \tau_r$ is small.

Our PCMF estimator mitigates the high variance and increased cost of PCume by making sure to suppress Russian Roulette termination until almost all of the contribution has been accumulated, and then apply more aggressive termination probabilities. This allows it to have the lowest variance amongst all estimators and the lowest work-normalized variance, except for the Gaussian extinction case where ratio tracking performs slightly better due to PCMF being more conservative, i.e. trading higher cost for lower variance.

Our Unidir-MIS estimator is the combination of the ratio tracking and next-flight estimators using MIS, while Bidir-MIS additionally includes other sampling strategies. MIS improves robustness by eliminating the worst-case behaviors of any individual strategy, but it also does not attain the best-case behavior of any strategy in isolation.

[104]: Novák et al. (2014), 'Residual Ratio Tracking for Estimating Attenuation in Participating Media'

[111]: Novák et al. (2018), 'Monte Carlo Methods for Physically Based Volume Rendering'

[120]: Leppänen (2010), 'Performance of Woodcock Delta-Tracking in Lattice Physics Applications Using the Serpent Monte Carlo Reactor Physics Burnup Calculation Code'

20: Note that the 'variance' and 'variance \times cost' plots of [104, Fig. 9] and [111, Fig. 5] incorrectly use the estimators' standard deviation instead of their variance, hence the discrepancies between their plots and ours.



Figure 6.14: Equal-cost (92.5M extinction lookups) comparison of several estimators (ours are in bold) on a homogeneous medium with the total extinction set to $\mu_t = 1.5 \times \mu_r$. The relatively tight μ_t favors next-flight and MIS estimators over ratio tracking. Our PCMF estimator produces a virtually noise-free result, and its render time is also low since lookups are cheap in this scene while this estimator makes more lookups per ray than the others.

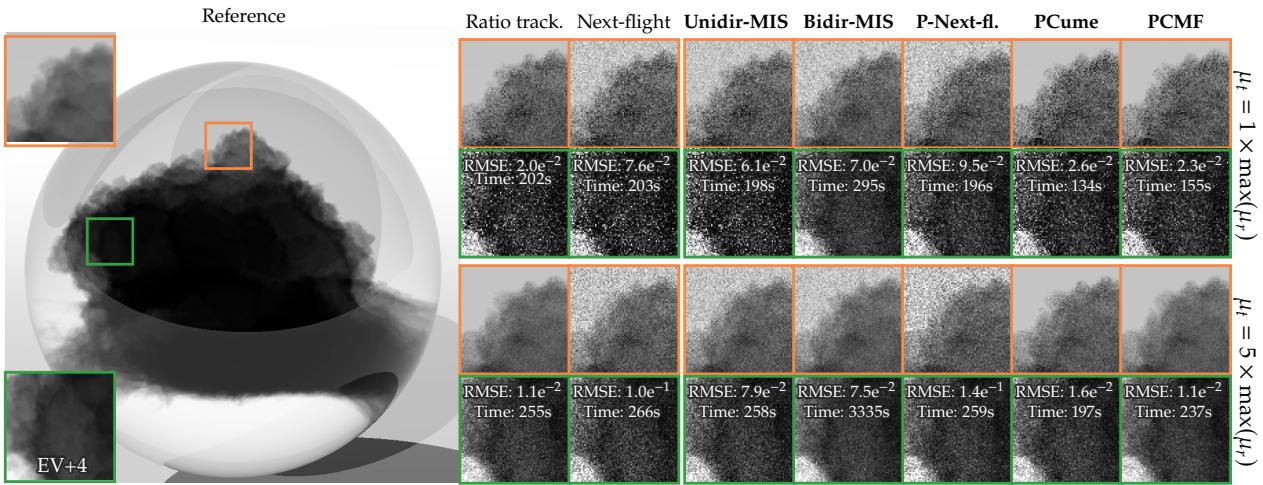


Figure 6.15: Equal-cost (0.5B and 1.3B lookups, top and bottom respectively) comparisons of estimators (ours are in bold) on a heterogeneous medium for two different total extinctions. The tighter control (top rows) are still relatively loose in this scene, favoring ratio tracking. Our Pcume and PCMF estimators do almost as well.

Homogeneous medium. In participating media with constant real density, such as Fig. 6.14, the only source of variance in a transmittance estimator is the discrete random choice of terms/recursion termination in their corresponding formulations. μ_t is relatively tight in this scene so traditional ratio tracking performs poorly as its Poisson PMF does not importance sample the null-extinction Poisson well. Next-flight and PCume perform better than ratio tracking however, some noise remains while PCMF is more conservative in this case and provides near-perfect estimation. PCMF also accomplishes this in a fraction of the time as other techniques since lookups are cheap within this medium, and it makes more lookups per ray due to its higher cost. The benefits of combining techniques via MIS can also be observed here and is consistent with the plots in Fig. 6.13.

Heterogeneous medium. We consider a more complex heterogeneous medium in Fig. 6.15 where we compare the estimators' performance for two different choices of μ_t : one tight (top rows), equal to the medium's majorant extinction $\max(\mu_r)$, and one loose (bottom rows) equal to $5 \times \max(\mu_r)$. This scene contains a small absorptive cloud contained within a refractive surface such that the globe focuses light through the cloud. The resulting

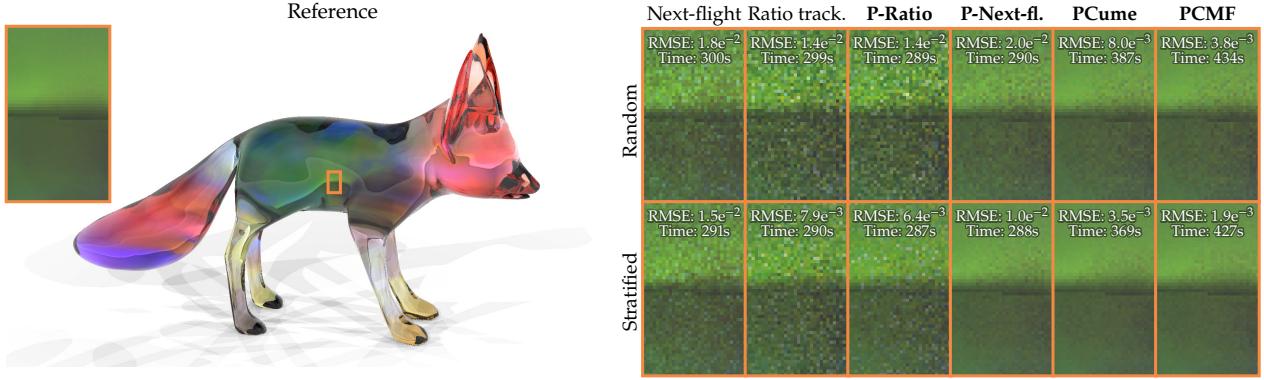


Figure 6.16: Comparing the benefits from Latin-hypercube sample stratification that various estimators (ours are in bold) get, using 8 samples/pixel. The power series ratio tracking and next-flight variants benefit more from stratification than their traditional (Volterra) counterparts.

extinction function along the rays through the globe resembles that of the Gaussian function in Fig. 6.13. The tighter μ_t still produces a relatively loose τ_t , which is the best case for traditional ratio tracking. PCMF has a slightly worse RMSE but is faster in terms of render speed as it evaluates fewer samples per pixel for an equal number of extinction lookups. As μ_t loosens, the performance of PCMF approaches ratio tracking. PCume performs third best. The next-flight and MIS estimators perform relatively poorly due to the next-flight techniques terminating significantly too early for this scene. Estimating a transmittance value of 1 with a loose μ_t is particularly difficult for such estimators, as can be seen in the orange zoom-ins and also in the bottom left corners of the plots in Fig. 6.13.

Despite its high overall error, Bidir-MIS performs well in the dense, low-transmittance region of the medium (top green zoom-in in Fig. 6.15) corresponding to the right part of the estimator’s plots in Fig. 6.13. However, it does not scale well to loose total extinctions as the number of techniques it needs to evaluate and compute weights for is proportional to the total optical depth which is reflected in its drastically increased render time. Nevertheless, it may provide motivation for future work in deriving more efficient bidirectional estimators.

Stratification. In Fig. 6.16 we show the benefit of sample stratification by comparing traditional Volterra formulations of the ratio tracking and next-flight estimators to our power series variants. While the two variants of ratio tracking perform identically with independent sampling, our power series formulations benefit more from stratification. In our variant, the discrete choice of the term to evaluate is explicit and easily stratified which significantly reduces variance. The plots in Fig. 6.17 also exhibit an improvement in the convergence rate of the variance. The amount of improvement depends on the configuration, and the Hammersley point set consistently outperforms Latin-hypercube sampling. Our other novel power series estimators in Fig. 6.16 benefit from stratification as well, roughly halving the error on average.

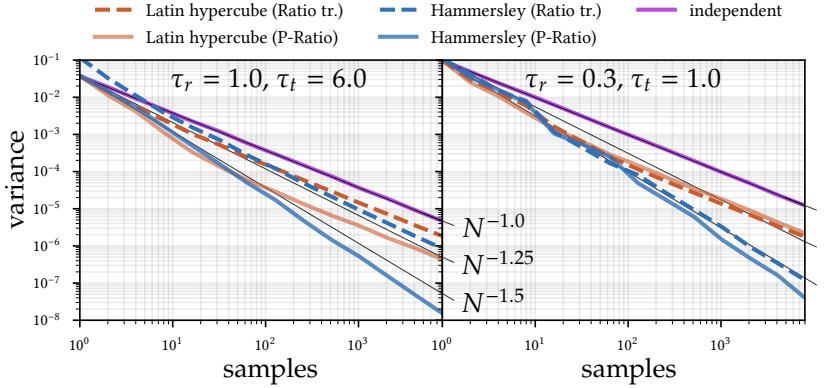


Figure 6.17: Convergence plots of ratio tracking vs. our power series ratio tracking using various sampling methods on a cosine extinction function for two choices of τ_t and τ_r . Stratification benefits are dependent on both parameters.

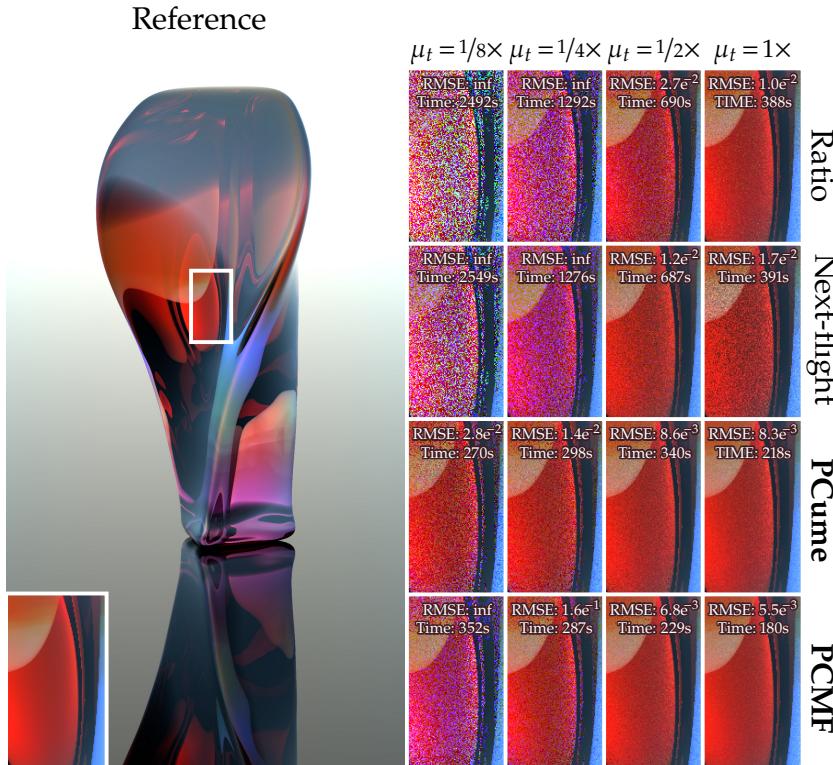


Figure 6.18: Equal-cost (160M lookups) render comparison, where in each column we set the extinction upper control μ_t to a different fraction of the medium's majorant extinction, $\max(\mu_t)$. Our power series estimators (in bold) are more resilient to non-bounding extinctions than prior work.

Non-bounding extinctions. In practice, it can be difficult or costly to compute a bounding μ_t , e.g. if the medium density is modified procedurally. In addition, the cost of all unbiased transmittance estimators using any known technique depends proportionally on the total extinction. So for sparse media where only a few small regions reach the maximum density, it would be desirable for an estimator to support non-bounding extinctions to reduce render time.

In Fig. 6.18 we compare our power series estimators against the current state of the art when utilizing varying degrees of non-bounding extinctions. The render times and variance of the existing estimators explode, even reaching numerical infinity for a handful of pixels. For ratio tracking, this aligns with the result in Eq. (6.34) which suggests the variance grows exponentially with decreasing total extinction.

μ_t	Track-length	Ratio track.	Next-flight	P-Ratio	P-Next-fl.	Unidir-MIS	Bidir-MIS	PCume	PCMF
Fig. 6.14	0.1			2.1×10^0	2.8×10^{-1}	8.8×10^{-1}	1.8×10^{-1}	7.3×10^{-1}	9.6×10^{-2}
	0.3			3.4×10^{-2}	9.3×10^{-3}	3.4×10^{-2}	8.1×10^{-3}	1.6×10^{-2}	1.1×10^{-2}
	0.6							8.0×10^{-3}	2.7×10^{-5}
	1.0	8.6×10^{-3}	8.5×10^{-3}	0.0×10^0	8.5×10^{-3}	0.0×10^0	4.6×10^{-3}	0.0×10^0	0.0×10^0
	1.5	8.2×10^{-3}	1.3×10^{-2}	3.6×10^{-3}	1.3×10^{-2}	2.9×10^{-3}	2.2×10^{-3}	3.8×10^{-3}	6.0×10^{-3}
	2.5	7.8×10^{-3}	8.8×10^{-3}	6.1×10^{-3}	8.8×10^{-3}	4.9×10^{-3}	4.5×10^{-3}	6.1×10^{-3}	4.5×10^{-3}
	5.0	7.6×10^{-3}	5.6×10^{-3}	7.9×10^{-3}	5.6×10^{-3}	6.7×10^{-3}	6.2×10^{-3}	7.1×10^{-3}	1.7×10^{-3}
Fig. 6.15	0.1								4.9×10^{-2}
	0.3			3.1×10^1	2.9×10^0	4.1×10^0	2.7×10^0	8.6×10^0	1.7×10^{-1}
	0.6			2.2×10^{-2}	1.8×10^{-2}	2.2×10^{-2}	1.7×10^{-2}	1.3×10^{-2}	2.0×10^{-2}
	1.0	1.2×10^{-2}	1.0×10^{-2}	1.7×10^{-2}	1.0×10^{-2}	1.9×10^{-2}	1.2×10^{-2}	2.1×10^{-2}	8.9×10^{-3}
	1.5	1.2×10^{-2}	7.4×10^{-3}	1.8×10^{-2}	7.4×10^{-3}	2.0×10^{-2}	1.3×10^{-2}	1.9×10^{-2}	6.0×10^{-3}
	2.5	1.1×10^{-2}	5.1×10^{-3}	1.8×10^{-2}	5.1×10^{-3}	2.1×10^{-2}	1.3×10^{-2}	1.8×10^{-2}	4.0×10^{-3}
	5.0	1.1×10^{-2}	3.4×10^{-3}	1.8×10^{-2}	3.4×10^{-3}	2.4×10^{-2}	1.3×10^{-2}	1.6×10^{-2}	2.8×10^{-3}
Fig. 6.18	0.1							3.0×10^0	4.5×10^{-1}
	0.3			8.9×10^{-2}	3.5×10^{-2}	1.1×10^{-1}	3.6×10^{-2}	4.9×10^{-2}	2.7×10^{-2}
	0.6			2.4×10^{-2}	1.9×10^{-2}	2.4×10^{-2}	2.0×10^{-2}	1.6×10^{-2}	2.0×10^{-2}
	1.0	1.4×10^{-2}	1.5×10^{-2}	2.0×10^{-2}	1.5×10^{-2}	2.1×10^{-2}	1.4×10^{-2}	2.3×10^{-2}	1.1×10^{-2}
	1.5	1.7×10^{-2}	1.4×10^{-2}	2.5×10^{-2}	1.4×10^{-2}	2.8×10^{-2}	1.8×10^{-2}	2.6×10^{-2}	6.9×10^{-3}
	2.5	1.6×10^{-2}	9.6×10^{-3}	2.5×10^{-2}	9.6×10^{-3}	3.0×10^{-2}	1.9×10^{-2}	2.6×10^{-2}	6.7×10^{-3}
	5.0	1.6×10^{-2}	6.5×10^{-3}	2.6×10^{-2}	6.6×10^{-3}	3.2×10^{-2}	2.0×10^{-2}	2.5×10^{-2}	3.8×10^{-3}
Fig. 6.16	0.1							6.9×10^1	
	0.3			3.7×10^{-1}	3.7×10^{-1}	3.7×10^{-1}	2.3×10^{-1}	3.5×10^{-1}	2.0×10^{-1}
	0.6			3.3×10^{-2}	8.2×10^{-2}	3.3×10^{-2}	9.7×10^{-2}	6.7×10^{-2}	8.4×10^{-2}
	1.0	1.9×10^{-2}	2.0×10^{-2}	7.6×10^{-2}	2.0×10^{-2}	9.5×10^{-2}	6.1×10^{-2}	7.0×10^{-2}	2.6×10^{-2}
	1.5	1.9×10^{-2}	1.5×10^{-2}	7.4×10^{-2}	1.5×10^{-2}	9.5×10^{-2}	5.9×10^{-2}	6.3×10^{-2}	2.2×10^{-2}
	2.5	1.9×10^{-2}	1.1×10^{-2}	7.3×10^{-2}	1.1×10^{-2}	9.6×10^{-2}	5.7×10^{-2}	5.7×10^{-2}	1.7×10^{-2}
	5.0	2.7×10^{-2}	1.1×10^{-2}	1.0×10^{-1}	1.1×10^{-2}	1.4×10^{-1}	7.9×10^{-2}	7.5×10^{-2}	1.6×10^{-2}

Table 6.1: RMSE values for the rendered scenes in this chapter, where in each row the total extinction μ_t is set to a different scale of the medium's majorant extinction, $\max(\mu_r)$. Cell colors are linearly interpolated per row between the best (green) and worst (red) estimator. Black means numerical infinity or not applicable (e.g. track length does not work with non-bounding controls).

Referring back to Fig. 6.11, non-bounding extinctions produce terms with alternating signs and excessively large magnitudes. The prefix sum of these terms also has significant variation initially, but after accumulating enough terms the oscillation stops and the sum starts approaching the transmittance. The next-flight estimator performs poorly as it always terminates too early. Our power series prefix-sum estimators degrade much more gracefully due to their higher cost. PCume effectively suppresses Russian roulette termination until the absolute value of its cumulative weight drops below 1. PCMF does slightly worse as its termination is directly driven by μ_t . In Chapter 7 we will propose a method which will alleviate this for most transmittance estimators.

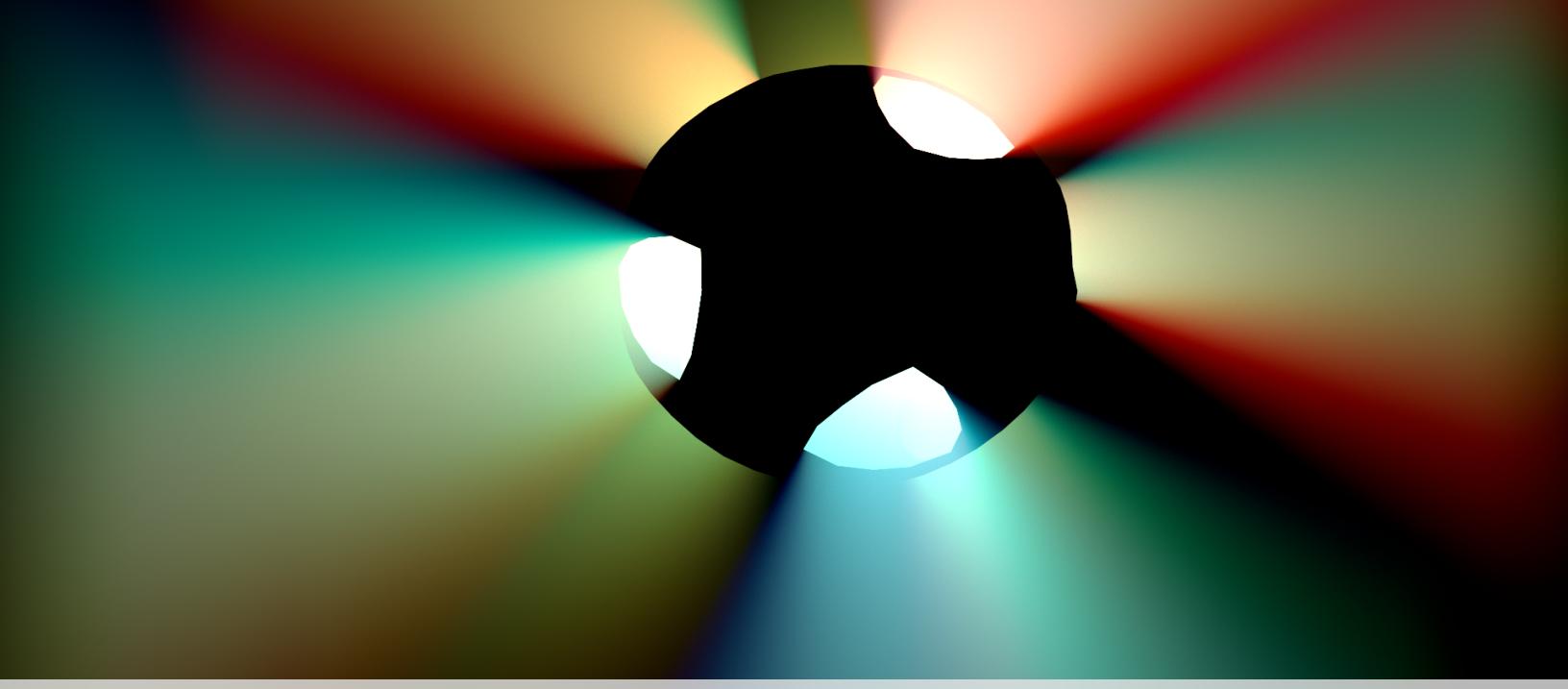
6.5.3 Discussion

In summary, for scenes containing sparse media with loose total extinctions, ratio tracking is the preferred estimator when the extinction lookups dominate the render time (Fig. 6.15). When stratification is available, our power series variants should be employed instead (Figs. 6.16 and 6.17). Our PCume estimator is the most robust when dealing with non-bounding extinctions out of all techniques reviewed in this chapter (Fig. 6.18). Our PCMF estimator is the preferred estimator in all other cases, as suggested by the best-estimator plots at the bottom of Fig. 6.13. Figure 6.15 also shows that PCMF can be faster than ratio tracking in terms of render time, and Table 6.1 indicates its performance will approach that of ratio tracking as μ_t becomes more loose.

In Chapter 8 we will discuss two estimators for exponential transmittance which are more performant than all techniques proposed within this chapter for fully absorptive media. One was introduced by Kettunen et al. [121] while the other is based on the idea of debiasing biased estimators which we explored in a future work [122].

[121]: Kettunen et al. (2021), 'An Unbiased Ray-Marching Transmittance Estimator'

[122]: Misso et al. (2022), 'Unbiased and consistent rendering using biased estimators'



7 Progressive techniques for volumetric rendering

Given a general quantity which is the result of some non-linear function of expectations,

$$I := g \left(\int f(t) dt \right), \quad (7.1)$$

this chapter focuses on analyzing steps ③, ④, and ⑤ from our recipe for consistent estimation (1.1). We assume that we have already completed step ① and have ascertained some consistent reformulation of Eq. (7.1),

$$\begin{aligned} I &= \lim_{n \rightarrow \infty} I_n \\ \langle I \rangle &= \left\langle \lim_{n \rightarrow \infty} I_n \right\rangle, \end{aligned} \quad (7.2)$$

which is amenable to unbiased estimation through the estimation of the formulation-dependent limit of I_n . In this chapter, we also assume that we have already derived an idealized Monte Carlo estimator for Eq. (7.2), and have proven that the estimator possesses finite work-normalized variance fulfilling ②. However, possessing an idealized estimator with finite work-normalized variance does not directly imply that a practical implementation of that estimator (i.e. step ③) will be efficient enough to be employed in practice for *all* situations. In fact, an unbiased estimator with finite variance can still exhibit variance which becomes so large that it can no longer fit within floating point precision. We refer to this as an estimator exhibiting *numerically infinite* variance.

As a practical example of this problem, consider the case where the consistent limit in Eq. (7.2) can take the form of either the base Volterra (6.23) or base power series (6.71) formulations for classical transmittance. While

7.1	Case for progressive volumetric rendering	126
7.2	Root cause of degenerate behavior	128
7.2.1	Transmittance estimation	128
7.2.2	Free-flight sampling	132
7.3	Progressive extinctions	133
7.3.1	Progressive extinctions through biasing	134
7.3.2	Asymptotic analysis	135
7.3.3	Adaptive Approach	138
7.4	Progressive NEE in black box media	140
7.4.1	NEE through line integration	141
7.4.2	Progressive volumetric NEE	145
7.4.3	Unbiased progressive volumetric NEE	146
7.5	Results	147
7.6	Conclusion	151
7.6.1	Future work	151

these formulations are technically amenable to unbiased estimation, choosing a PDF or PMF to sample from is not a straightforward decision. Thus, we reformulated the Volterra (6.25) and power series (6.72) formulations using the null-collision paradigm through introducing μ_t , and indirectly $\mu_n(\mathbf{x}) := \mu_t - \mu_r(\mathbf{x})$, for the purpose of incorporating an analytic term ($\rho_f^{(t)}$) which we could then perfectly importance sample.

The crux of this process is the arbitrary choice for establishing the value of the total extinction (μ_t) which dictates the overall shape of the Monte Carlo sampling distributions employed by most volumetric rendering algorithms. When μ_t is bounding, the corresponding PDFs and PMFs are good fits to the contribution of the Volterra or power series, thus, all classical volumetric rendering algorithms will be at least usable in practice. However, when μ_t is non-bounding, the corresponding PDFs and PMFs are no longer good fits, resulting in significant increases in variance (see Fig. 6.4) or even practical renders exhibiting numerically infinite variance (see Figs. 6.18 and 7.1).

In such cases, it can be more convenient to construct biased but consistent estimators where the bias is designed to *disappear* in the infinite limit of work. Such solutions are constructed *progressively*,

$$\langle I \rangle_n := \frac{1}{n} \sum_{j=1}^n \langle I(j) \rangle_1, \quad (7.3)$$

in the form of secondary estimators where the primary estimator $\langle I(j) \rangle_1$ *progressively* changes as more samples are taken. For our interests, we consider j to be the pixel sample index and the $\langle I(j) \rangle_1$ to be the result of running a *potentially biased* rendering algorithm where the bias is designed to decrease as a function of j . A common example of this progressive estimation methodology is the progressive photon mapping family of techniques, which we previously reviewed in Sec. 5.3.

Within this chapter, we show that progressive estimators can be utilized instead of unbiased ones to alleviate the potential drawbacks of the stochastic nature of unbiased estimation through the lens of classical volumetric rendering. For now, we only consider the case where the unbiased estimators we are replacing exhibit only finite work-normalized variance. In the next chapter, we will relax these assumptions and consider the infinite variance case (step ⑦). In certain ideal scenarios, we will prove that the squared bias of our formulations can disappear at a *faster* rate than the variance, meaning that the bias of these progressive techniques should be practically inconsequential to the asymptotic performance of our estimators.

We ultimately propose two practical solutions for two different problems within the context of rendering classical volumetric media whose extinction is a black box function. Meaning, we can *never* guarantee that a given μ_t is bounding ($\mu_t > \mu_r(\mathbf{x})$), for all points within a participating medium. The first solution (Sec. 7.3) proposes a practical way of modifying current volumetric rendering algorithms in a progressive but consistent manner such that participating media can be rendered using non-bounding extinctions without resulting in the excessive variance we witnessed previously in Fig. 6.4 (fulfilling ④). This first solution was originally completed as

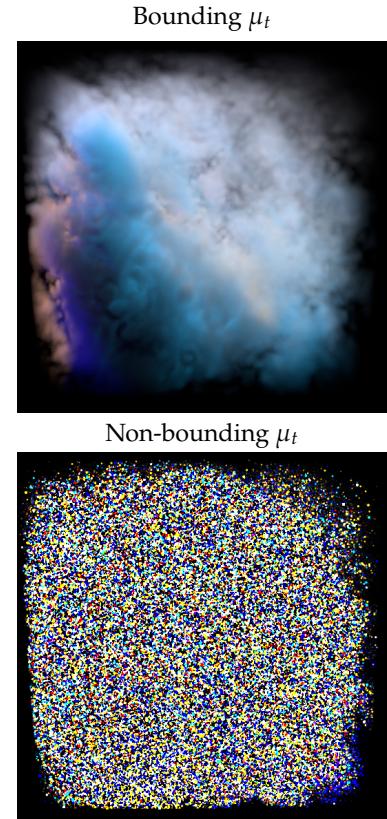


Figure 7.1: A visual comparison of 8192 pixel sample renders using two different choices for the total extinction. On the top we use the upper bound of the real extinction as the total extinction ($\mu_t = 3.0$), and on the bottom we employ a near-zero worst-case total extinction ($\mu_t = 0.01$). Both renders employ traditional ratio tracking + weighted delta tracking as the transmittance estimator and free-flight distance sampling routine within a traditional implementation of volumetric path tracing with NEE.

an industrial collaboration with Yining Karl Li, Brent Burley, and Daniel Teece from Walt Disney Animation Studios [115] and is planned to be incorporated into the production renderer Hyperion [83].

The second solution (Sec. 7.4) provides an extension where we show how to support emitter sampling within emissive participating media with black box extinctions when applying NEE. While this can be done in a biased, but consistent, manner (step ④), we ultimately propose an unbiased version¹ of this technique which requires slightly modifying traditional volumetric path-tracing with NEE (⑤).

Before we propose our novel contributions we make the case for why biased, but consistent, techniques are even necessary in Sec. 7.1. We then provide an analysis of the root cause behind the degenerate behavior that unbiased estimators exhibit due to the presence of non-bounding extinctions in Sec. 7.2. We provide practical results of our novel techniques in Sec. 7.5, and we conclude the chapter in Sec. 7.6.

7.1 Case for progressive volumetric rendering

In the context of rendering classical heterogeneous participating media, practitioners have a wide variety of unbiased solutions at their disposal for generating correct images. We previously reviewed unbiased free-flight distance sampling techniques in Sec. 5.4.5 and introduced formulations for transmittance which are amenable to unbiased estimation as well as derived many unbiased estimators in Chapter 6. Unfortunately, all of these techniques suffer from the same fatal flaw in practical applications which can lead to disastrous results (see Fig. 7.1).

With the exception of one technique which we will introduce in Chapter 8, all unbiased volumetric rendering algorithms for classical media are fully reliant on the paradigm of null-scattering (Sec. 4.3). Due to a variety of factors, these null-scattering based methods have become commonplace within the current generation of production renderers [83, 122, 123] used in both the animation and visual effects industries. Null-scattering based methods reap impressive performance improvements from a variety of acceleration techniques [103, 104, 124, 125] which allow for efficiently traversing or analytically accounting for portions of a medium. Additionally, the null-scattering path integral [51] has enabled combining null-scattering via multiple importance sampling (MIS) [33] with other techniques, such as equi-angular sampling [105], for improved robustness. This combination of factors has made null-scattering techniques preferred over biased ray marching, which used to be the most common alternative despite being biased and inconsistent.

However, the need to directly specify μ_t becomes problematic in practice since, when μ_t is *non-bounding*, most current methods [104, 107, 108, 110, 126, 127] perform exceptionally poorly (see Fig. 7.1, Fig. 7.5, or Fig. 6.18). This can become prohibitive in the context of production rendering since volumetric assets are often the result of a series of physical simulations, procedural manipulations, or other artistic workflows where the final density becomes

[115]: Misso et al. (2023), ‘Progressive null-tracking for volumetric rendering’

[83]: Burley et al. (2018), ‘The Design and Evolution of Disney’s Hyperion Renderer’

¹: The unbiased version assumes we have an initially bounding μ_t . Otherwise, it can also be formulated as a consistent technique.

[83]: Burley et al. (2018), ‘The Design and Evolution of Disney’s Hyperion Renderer’

[122]: Fascione et al. (2018), ‘Path Tracing in Production’

[123]: Novák et al. (2018), ‘Monte Carlo Methods for Volumetric Light Transport Simulation’

[51]: Miller et al. (2019), ‘A Null-Scattering Path Integral Formulation of Light Transport’

[33]: Veach et al. (1995), ‘Optimally Combining Sampling Techniques for Monte Carlo Rendering’

[105]: Kulla et al. (2012), ‘Importance Sampling Techniques for Path Tracing in Participating Media’

[104]: Novák et al. (2014), ‘Residual Ratio Tracking for Estimating Attenuation in Participating Media’

[107]: Cramer (1978), ‘Application of the Fictitious Scattering Radiation Transport Model for Deep-Penetration Monte Carlo Calculations’

[108]: Galtier et al. (2013), ‘Integral Formulation of Null-Collision Monte Carlo Algorithms’

[110]: Georgiev et al. (2019), ‘Integral Formulations of Volumetric Transmittance’

[126]: Carter et al. (1972), ‘Monte Carlo Sampling with Continuously Varying Cross Sections along Flight Paths’

[127]: Jonsson et al. (2020), ‘Direct Transmittance Estimation in Heterogeneous Participating Media Using Approximated Taylor Expansions’

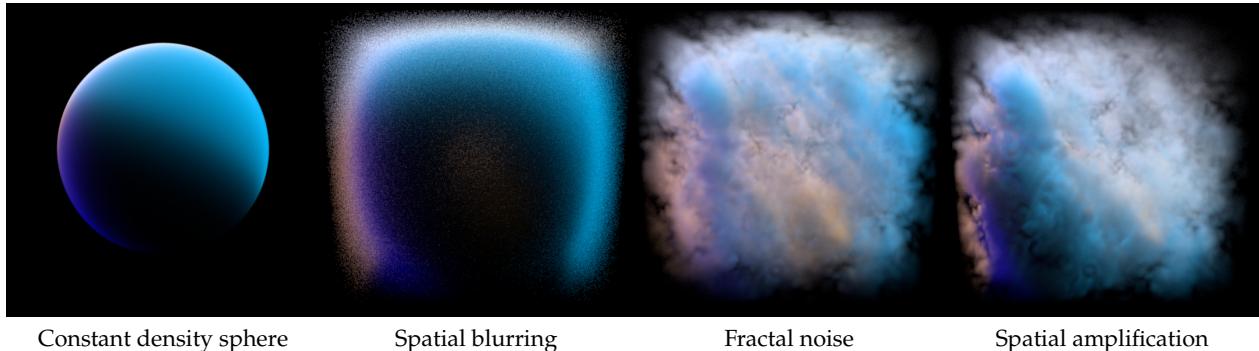


Figure 7.2: A visualization of the construction of a procedural medium through applying mathematical operations to a constant density sphere. The medium is modified in order from left to right with a brief description of the operations involved beneath each image.

a black box that can only be point-evaluated (for example, see Fig. 7.2). Any choice of μ_t based on point evaluations becomes only a *guess* for a truly bounding combined extinction.

In the context of production rendering, practitioners have effectively two options to remain robust and consistent when employing null-scattering techniques: they can specify an often overly conservative value for μ_t , however this will most likely result in prohibitively expensive renders if μ_t is excessively loose for some or all of the volume. Alternatively, all volumes can be baked into voxel density grids as a pre-process (see Fig. 4.2), but this incurs a large startup cost, is memory intensive, and reduces the visual fidelity of all participating media in a scene.

Outside of production rendering, a recent and impressive technique in the field of novel-view synthesis, known as *Neural radiance fields* (NeRFs) [128], has become a prevalent topic of study within the computer graphics and machine learning literature [129, 130]. NeRFs are effectively learned volumetric representations for entire scenes which are encoded by a neural network. Rendering a scene represented by a NeRF requires computing an effective *transmittance* through the NeRF as it effectively encodes both the radiance and extinction throughout space.

However, a neural network is a textbook example of a black box function. Once a neural network learns how to represent a participating medium through learning a modified extinction function and radiance, any renderer which attempts to synthesize the neural volume will have no *apriori* knowledge of the medium's shape. Thus, null-scattering based techniques could not be employed previously to render NeRFs without excessive variance due to non-bounding extinctions. In absence of consistent solutions, biased (inconsistent) ray marching (5.29) is employed instead [128, 130], which is not ideal.

The methods we will propose in Sec. 7.3 and Sec. 7.4 will alleviate these constraints allowing for null-scattering techniques to be employed in both production rendering and in the visualization of black box volumetric scene representations even when in the presence of non-bounding extinctions. While some specialized techniques do exist for side-stepping negative null coefficients [107, 126, 131, 132], many of them often require splitting light

[128]: Mildenhall et al. (2020), ‘NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis’

[129]: Lombardi et al. (2021), *Mixture of Volumetric Primitives for Efficient Neural Rendering*

[130]: Shafiei et al. (2021), *Learning Neural Transmittance for Efficient Rendering of Reflectance Fields*

[128]: Mildenhall et al. (2020), ‘NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis’

[130]: Shafiei et al. (2021), *Learning Neural Transmittance for Efficient Rendering of Reflectance Fields*

[107]: Cramer (1978), ‘Application of the Fictitious Scattering Radiation Transport Model for Deep-Penetration Monte Carlo Calculations’

[126]: Carter et al. (1972), ‘Monte Carlo Sampling with Continuously Varying Cross Sections along Flight Paths’

[131]: Galtier et al. (2016), ‘Radiative Transfer and Spectroscopic Databases: A Line-Sampling Monte Carlo Approach’

[132]: Szirmay-Kalos et al. (2017), ‘Unbiased Estimators to Render Procedurally Generated Inhomogeneous Participating Media’

paths to reduce variance or handle negative extinctions in an unbiased manner. We instead propose generalized techniques for making most null-scattering methods resilient to non-bounding extinctions without drastically changing the underlying rendering algorithm or tracing extra paths.

7.2 Root cause of degenerate behavior

To improve the robustness of null-scattering techniques, we must first analyze how existing methods degrade when non-bounding estimates of μ_t are employed. Since there are two different processes involved in stochastically computing volumetric light transport, i.e. sampling scatter locations and estimating transmittance, we will analyze each case separately starting with transmittance estimation.

7.2.1 Transmittance estimation

Previously in Sec. 6.2.4, we reviewed how all tracking-based transmittance estimators can be derived by directly applying Monte Carlo integration to estimate the null-collision form of the Volterra representation for classical transmittance (6.25). We additionally introduced a generalized formulation for a Volterra estimator (6.26) and showed how all prior tracking-based estimators could be derived from it, which we will restate as,

$$\langle T_r(x_a, x_b) \rangle_{\text{Gen}} := \frac{T_t(x_a, x_b)}{P_{\text{nf}}} + \frac{\rho_f^{(t)}(x_t) \alpha_n(x_t) \langle T_r(x_t, x_b) \rangle_{\text{Gen}}}{p(x_t) P_{\text{rec}}}. \quad (7.4)$$

for convenience. As a reminder, P_{nf} is the probability of evaluating the combined transmittance term, P_{rec} is the probability of evaluating the recursive term, $p(x_t)$ is the probability distribution used for sampling position x_t , and $\rho_f^{(t)}(x_t)$ is the free-flight distribution of the combined medium,

$$\rho_f^{(t)}(x_t) := \mu_t e^{-\mu_t \cdot (t-a)}, \quad (7.5)$$

where $t - a$ is the distance to position x_t from the start of the segment (x_a).

Ratio tracking. Ratio tracking (6.12) is one of the most common transmittance estimators due to its simplicity and the fact that it is employed in production rendering [83]. For these reasons, we will analyze the impact of non-bounding extinctions on transmittance estimators through the lens of the ratio tracking family of techniques [104].

Ratio tracking can be derived from Eq. (7.4) by setting,

$$p(x_t) = \rho_f^{(t)}(x_t); \quad P_{\text{nf}} = T_t(x_a, x_b); \quad P_{\text{rec}} = 1 - T_t(x_a, x_b), \quad (7.6)$$

[83]: Burley et al. (2018), ‘The Design and Evolution of Disney’s Hyperion Renderer’

[104]: Novák et al. (2014), ‘Residual Ratio Tracking for Estimating Attenuation in Participating Media’

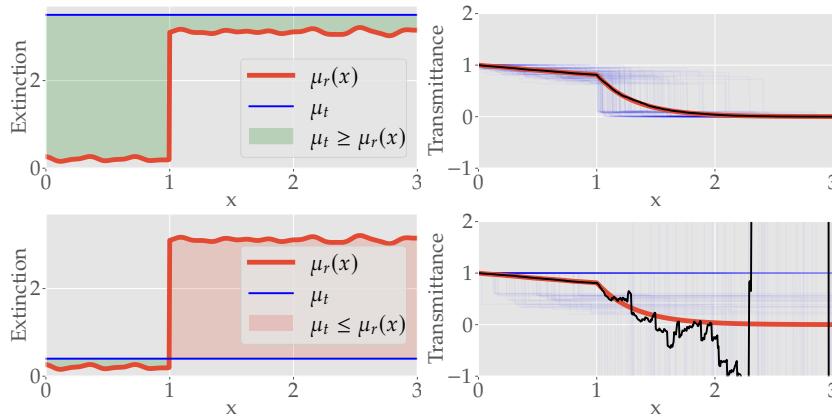


Figure 7.3: A comparison of the behavior of ratio tracking with bounding (top) and non-bounding (bottom) total extinctions. On the right we plot the ground truth transmittance (red), the average of 100 individual invocations of ratio tracking (black), and the individual trackings (blue).

where we mechanically decide to evaluate either the combined extinction term or the recursive call based on whether the sampled position has reached the end of the path segment ($t \geq b$). As we have seen previously (6.12), ratio tracking can be mathematically described as,

$$\langle T_r(\mathbf{x}_a, \mathbf{x}_b) \rangle_{\text{Rat}} := \frac{\mu_n(\mathbf{x}_t)}{\mu_t} \langle T_r(\mathbf{x}_t, \mathbf{x}_b) \rangle_{\text{Rat}} = \prod_{j=1}^{n-1} \alpha_n(\mathbf{x}_j) = \prod_{j=1}^{n-1} \frac{\mu_t - \mu_r(\mathbf{x}_j)}{\mu_t}, \quad (7.7)$$

where n is the total number of recursions in Eq. (7.4) or steps taken, and x_j is the j th ordered location along the ray. The resulting running product of the null extinction is a nice monotonically decreasing, positive function when μ_t is bounding,, however when μ_t becomes non-bounding the behavior of the product rapidly deteriorates (see Fig. 7.3).

We visualize the degradation in the quality of the transmittance estimates returned by ratio tracking due to the presence of varying degrees of non-bounding extinctions in Fig. 7.4. This empirical analysis reveals that there are two distinct cases which can arise leading to potentially controllable and completely uncontrollable variance.

When $\mu_t < \mu_r(x) < 2\mu_t$, the sign of the running product (6.12) will begin to flip, but its absolute value will remain a monotonically decreasing function bounded within $[0, 1]$. Since each term in the product remains between $[-1, 1]$, the magnitude of the returned transmittances will always be well behaved so the increased variance may still be manageable. We refer to this case as having *slightly non-bounding* extinctions.

Once $\mu_r(x) > 2\mu_t$ the null-albedo ($\alpha_n(\mathbf{x})$) will become *unbounded*. This results in both the running product and the variance to also become unbounded as the number of steps n increases (i.e. as a function of distance), which can lead to numerically infinite variance. We refer to this case as having *significantly non-bounding* extinctions.

Variance analysis. The two distinct behaviors which arose during our empirical analysis of ratio tracking (see Fig. 7.4) can also be explained using the analytical equation for ratio tracking's variance within a homogeneous medium (6.34). While we previously derived the variance to be in terms of

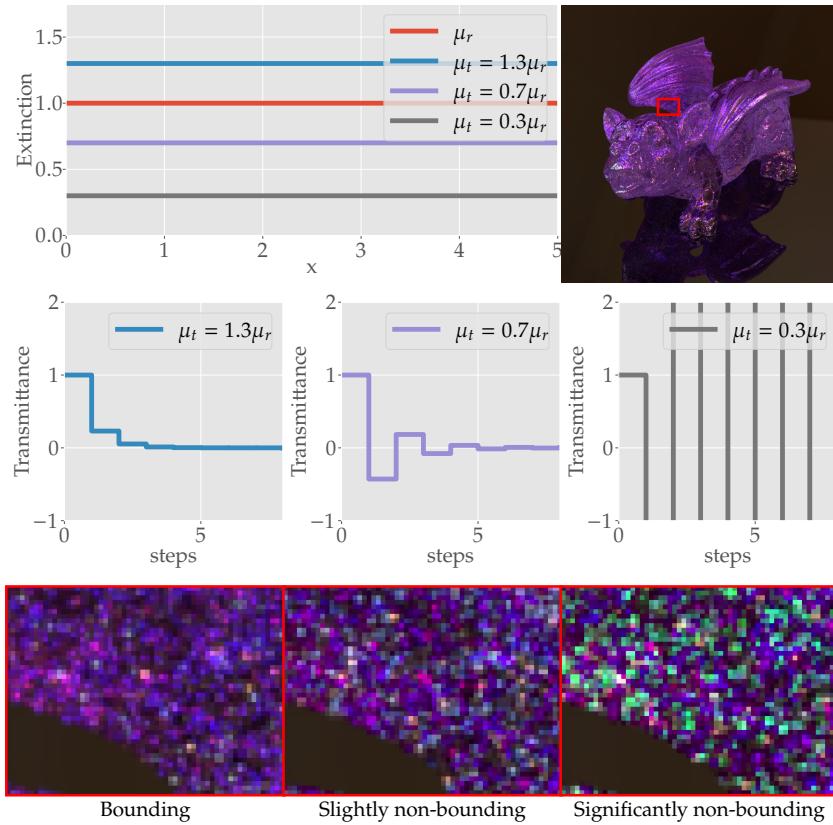


Figure 7.4: A comparison of the different behaviors exhibited by ratio tracking when μ_t is bounding, slightly non-bounding, and significantly non-bounding for a constant density absorbing medium (top row). We visualize the returned transmittance as a function of the number of exponential distances sampled (middle row) and visualize the difference in render quality for approximately equal spp renders (bottom row).

optical thicknesses (τ_r, τ_t), it is more convenient for the purposes of our analysis to rewrite Eq. (6.34),

$$V[T_{Rat}] = e^{-2\tau_r} \left(e^{\frac{\tau_r^2}{\tau_t}} - 1 \right) = e^{-2\mu_r d} \left(e^{\frac{(\mu_r d)^2}{\mu_t d}} - 1 \right), \quad (7.8)$$

so that the variance becomes a function of the extinctions and distance ($d = b - a$) along the path segment. In Chapter 6, we used Eq. (7.8) to reflect on the performance of ratio tracking when employing tightly and loosely bounding extinctions while we mostly avoided the non-bounding case.

To analyze the variance of ratio tracking in the presence of non-bounding homogeneous extinctions, we first rewrite Eq. (7.8),

$$V[T_{Rat}] = e^{\left(\frac{\mu_r - 2\mu_t}{\mu_t} \right) \mu_r d} - e^{-2\mu_r d}, \quad (7.9)$$

by distributing the real exponential and simplifying terms. This results in a formulation that analytically explains the two distinct behaviors which we empirically observed in Fig. 7.4. When the extinction is at most slightly non-bounding ($\mu_r < 2\mu_t$), the exponent within the first exponential will remain negative ($\frac{\mu_r - 2\mu_t}{\mu_t} < 0$). Thus, the variance will always be bounded from above by $e^0 = 1$ and will *decrease* as a function of distance (d). Once μ_t becomes significantly non-bounding ($\mu_r > 2\mu_t$), the exponent will turn positive allowing the variance to become unbounded as the variance will now *increase exponentially* as a function of d .

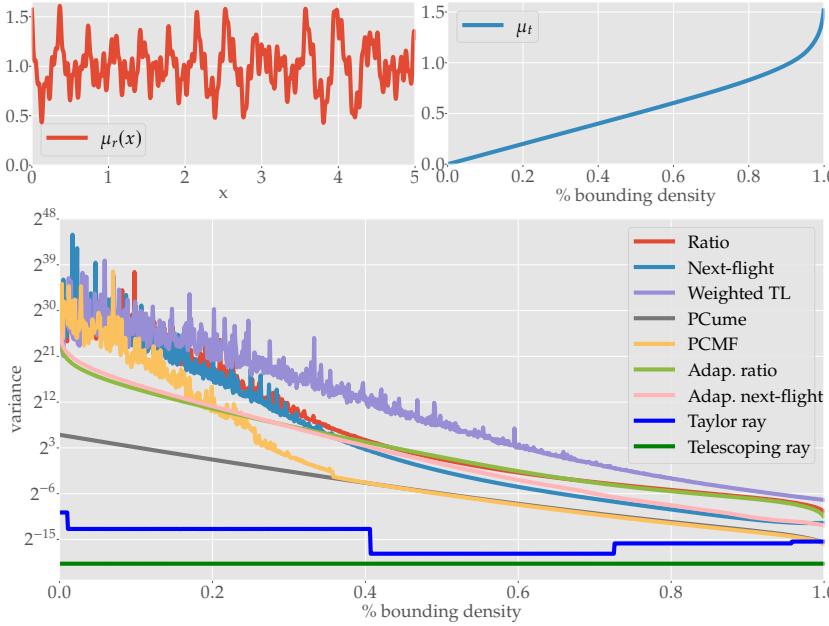


Figure 7.5: We plot how the variance of many transmittance estimators behave (bottom) when used to estimate the transmittance through a medium given by the top-left graph as a function of the percent of μ_r which μ_t bounds. We additionally plot how μ_t behaves in the top right graph. All estimators from Chapter 6 exhibit increased variance as μ_t becomes more non-bounding. Variance values were computed from 2^{22} independent primary estimates from each estimator.

As long as d is *finite* and $\mu_t > 0$, the variance and work-normalized variance will remain *finite* but have the potential to become numerically infinite (see Fig. 6.16). However, if infinite media ($b = \infty$) is supported by the renderer, then ratio tracking will exhibit provably *infinite variance* in the case of significantly non-bounding μ_t . For all applications in this dissertation, we assume that all participating media is bounded by the virtual scene extents which we also assume to be finite. Thus, we will continue to refer to ratio tracking as a finite variance estimator which can exhibit numerically infinite variance in the presence of poor importance sampling decisions.

Discussion. All transmittance estimators reviewed up until now rely on μ_t to either directly or indirectly influence their importance sampling decisions. In Fig. 7.5, we show how all previously introduced transmittance estimators will exhibit increased variance as larger degrees of non-bounding extinctions are employed. However, they all exhibit different degrees of resiliency as we observed in Fig. 6.18. In that practical experiment, all other transmittance estimators exhibited numerically infinite variance when $\mu_r = 8\mu_t$, while PCume only suffered an approximate $3.4\times$ increase in variance for an equal number of extinction evaluations. Even though this can be considered manageable, a $3.4\times$ *decrease* in performance due to the existence of non-bounding extinctions is still egregious, especially in the context of production rendering.

However, there are two unbiased transmittance estimators which do not fully rely on μ_t , making them completely resilient to non-bounding extinctions. Those estimators are the Taylor series based raymarching [121] and our debiased telescoping series based raymarching [112] techniques. While we will introduce these techniques in Chapter 8, they generally trade excessively high cost for lower variance. Meaning, that while they are the undisputed best techniques for rendering absorption-only media, they are

[121]: Kettunen et al. (2021), ‘An Unbiased Ray-Marching Transmittance Estimator’

[112]: Misso et al. (2022), ‘Unbiased and consistent rendering using biased estimators’

usually outperformed in highly scattering media by cheaper techniques (see Fig. 7.15), such as ratio tracking. In such cases, the majority of the variance naturally arises due to the process of importance sampling medium scatter events or other factors. So, for the practical needs of production rendering where highly scattering media is common, it is more pertinent to derive ways of making cheaper transmittance estimators resilient to non-bounding extinctions rather than deriving estimators that evaluate the real extinction excessively enough to return manageable variance.

7.2.2 Free-flight sampling

Similarly to the problem of estimating transmittance, there is currently no *consistent* and *cheap* solution for the problem of importance sampling medium interactions in the presence of non-bounding μ_t . Consider a generalized Monte Carlo estimator for the null-VRE (5.34),

$$\langle L_o(\mathbf{x}, \vec{\omega}_o) \rangle := \begin{cases} \frac{1}{P_m P_a} \frac{\rho_f^{(t)}(\mathbf{x}_t) \alpha_a(\mathbf{x}_t) L_c(\mathbf{x}_t, \vec{\omega}_o)}{p(\mathbf{x}_t)} & \text{with prob. } P_m P_a \\ \frac{1}{P_m P_s} \frac{\rho_f^{(t)}(\mathbf{x}_t) \alpha_s(\mathbf{x}_t) p_p(\mathbf{x}, \vec{\omega}_t, \vec{\omega}_o) \langle L_i(\mathbf{x}_t, \vec{\omega}_i) \rangle}{p(\mathbf{x}_t) p(\vec{\omega}_i)} & \text{with prob. } P_m P_s \\ \frac{1}{P_m P_n} \frac{\rho_f^{(t)}(\mathbf{x}_t) \alpha_n(\mathbf{x}_t) \langle L_o(\mathbf{x}_t, \vec{\omega}_o) \rangle}{p(\mathbf{x}_t)} & \text{with prob. } P_m P_n \\ \frac{1}{1-P_m} T_t(\mathbf{x}, \mathbf{x}_d) \langle L_s(\mathbf{x}_d, \vec{\omega}_o) \rangle & \text{with prob. } 1 - P_m. \end{cases} \quad (7.10)$$

which is employed to estimate the transport of light through a scene containing participating media within the null-scattering paradigm.

All existing unbiased techniques for estimating Eq. (7.10) differ in how they assign the probabilities P_a, P_n, P_s, P_m to evaluate emission, null-scattering, in-scattered radiance, decide between evaluating medium and surface scattering events, and how they choose to importance sample the interaction location \mathbf{x}_t within each integrand. With the exception of equiangular sampling (5.32), most techniques rely on importance sampling $p(\mathbf{x}_t) \propto \rho_f^{(t)}(\mathbf{x}_t)$ given by Eq. (7.5) and then set P_m accordingly (see Sec. 5.4.5).

Most existing techniques (see Sec. 5.4.5) choose P_a, P_n, P_s to be defined with respect to μ_t . However, to remain unbiased in the presence of non-bounding extinctions, these techniques have to be employed as weighted variations where the probabilities are relatively redefined as,

$$P_*(\mathbf{x}) := \frac{\mu_*(\mathbf{x})}{\mu_r(\mathbf{x}) + |\mu_n(\mathbf{x})|}. \quad (7.11)$$

Inserting these probabilities into Eq. (7.10) leads to a shared factor of $\frac{\mu_r(\mathbf{x}) + |\mu_n(\mathbf{x})|}{\mu_t}$ which is multiplied into each of the three medium event branches. When μ_t is significantly non-bounding with respect to $\mu_a(\mathbf{x})$, $\mu_s(\mathbf{x})$, or $\mu_n(\mathbf{x})$, the absolute value of this factor becomes greater than 1, thus the weight and variance of the estimator will grow exponentially as a function of distance and number of path vertices between sensor and source (see Fig. 7.6) in a similar manner to ratio tracking.

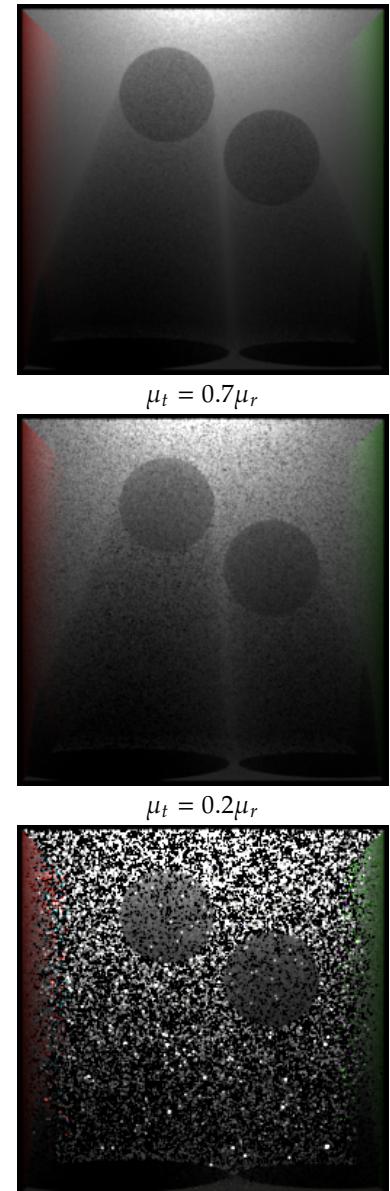


Figure 7.6: We visualize the degradation of weighted delta tracking (7.11) for free-flight distance sampling in the presence of non-bounding extinctions within a scene containing a homogeneous medium while only rendering direct illumination. All renders are roughly equal extinction calls, and transmittance is evaluated analytically.

Discussion. When no limit is placed on the number of real bounces a light path can undergo, employing Eq. (7.11) will lead to *infinite variance* in the presence of significantly non-bounding extinctions. Even worse, since the path contribution will tend to remain larger than 1 as the absolute value of the running weight will continue to grow exponentially, contribution based Russian Roulette termination may never occur. Thus, algorithms such as volumetric path-tracing with NEE may *never* terminate.

However, in practice, a maximum path depth is always specified before rendering due to the potential of total internal reflection where light paths can naturally enter seemingly infinite loops of reflections within refractive objects [18]². Thus, we always assume that a maximum path depth is specified for all of our results (Usually 20. For highly scattering media: 80). While these algorithms can theoretically exhibit infinite variance similar to their transmittance estimator counterparts, this never happens in practice and they will at worst exhibit numerically infinite variance (see Fig. 7.1).

Always relying on equiangular sampling may initially seem like a decent alternative in cases where non-bounding extinctions cannot be avoided, however, equiangular sampling relies on having *a priori* information on all emitters within a scene to importance sample medium events. In the case of a virtual environment being lit by an emissive procedural volumetric source, equiangular sampling cannot be used since we would not know how to importance sample the black box volumetric emission. We propose a solution for this specific problem in Sec. 7.4. Additionally, equiangular sampling does not importance sample proportionally to the real transmittance, thus, we would still need a transmittance estimator which was resilient to non-bounding extinctions while being efficient enough to employ in production.

7.3 Progressive extinctions

Non-bounding extinctions are an obvious problem for any volumetric rendering algorithm which is employed to simulate the transport of light through highly-scattering participating media. Prior work has attempted to address the problem of handling negative null coefficients [107, 108, 126, 131, 132], however, their solutions either do not generalize for the case where the extinction is an unknown black box function, result in reduced overall convergence rates, or only incrementally improve the resiliency of estimators without solving the underlying problem.

We instead propose a progressive two-pass approach which trades initial bias for avoiding the problem of excessive variance in its entirety. Our method is consistent, requires no major direct modifications to any existing volumetric rendering algorithm, and we will also prove that its expected impact on the performance of any implementation should not impede the asymptotic efficiency of any practical implementation.

We start by introducing the general idea behind our approach in Sec. 7.3.1, followed by an idealized analysis of how our approach’s initial bias impacts the efficiency of a generalized rendering algorithm in Sec. 7.3.2. We conclude

[18]: Pharr et al. (2016), *Physically Based Rendering*

2: As an example of this, see the black regions of the glass object in Fig. 6.18

[107]: Cramer (1978), ‘Application of the Fictitious Scattering Radiation Transport Model for Deep-Penetration Monte Carlo Calculations’

[108]: Galtier et al. (2013), ‘Integral Formulation of Null-Collision Monte Carlo Algorithms’

[126]: Carter et al. (1972), ‘Monte Carlo Sampling with Continuously Varying Cross Sections along Flight Paths’

[131]: Galtier et al. (2016), ‘Radiative Transfer and Spectroscopic Databases: A Line-Sampling Monte Carlo Approach’

[132]: Szirmay-Kalos et al. (2017), ‘Unbiased Estimators to Render Procedurally Generated Inhomogeneous Participating Media’

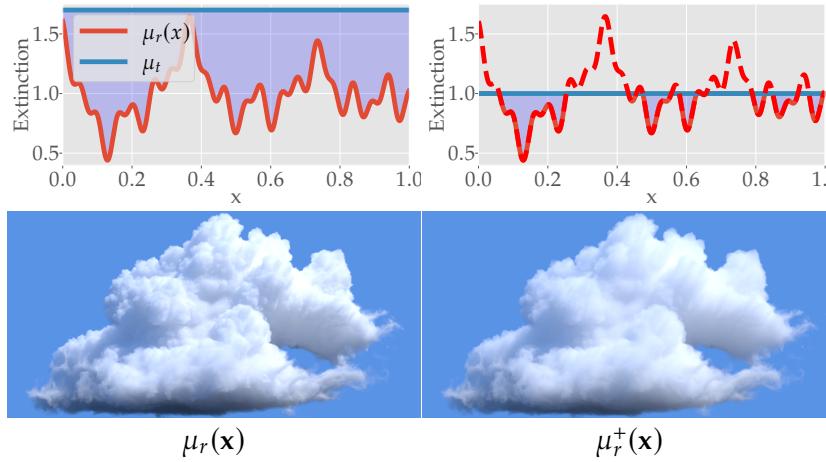


Figure 7.7: We visualize the idea of biasing a medium through clamping its real density to avoid $\mu_n(x) < 0$. Note that clamping $\mu_r(x)$ to $\mu_n(x)$ allows for the use of null-tracking techniques but results in lost detail.

this section in Sec. 7.3.3 with an introduction of a practical adaptive formulation of our approach which can be employed agnostically in *any* volumetric renderer.

7.3.1 Progressive extinctions through biasing

Variance is expected to increase whenever the user-specified total extinction is *less* than the real extinction ($\mu_t < \mu_r(x)$) anywhere within a participating medium. To avoid this increase in variance, we simply choose to never allow non-bounding extinctions by directly *biasing* the medium through clamping $\mu_r(x)$ to μ_t ,

$$\begin{aligned}\mu_r^+(x) &:= \text{Min}(\mu_t, \mu_r(x)) \\ \mu_n^+(x) &:= \mu_t - \mu_r^+(x).\end{aligned}\quad (7.12)$$

This will obviously result in the loss of any high-frequency details within the medium (see Fig. 7.7) in exchange for forcibly guaranteeing that the variance stays bounded and the corresponding clamped null extinction ($\mu_n^+(x)$) remains non-negative. Note that Eq. (7.12) is estimator agnostic and only involves directly modulating the medium extinction. No significant changes have to be applied to *any* aforementioned transmittance estimator or free-flight sampling routine except for replacing all extinction calls with their clamped alternatives to make them resilient to excessive variance.

Progressive extinctions. Employing Eq. (7.12) effectively trades excessive variance in return for rendering biased media. Next, we choose to formulate a progressive and consistent estimator that will ensure the bias we introduce through clamping will converge to zero in the limit.

Let us denote the expected value of rendering a scene containing participating media whose real extinction has been clamped to μ_t as $I(\mu_t)$. We then formulate a progressive secondary estimator by averaging n images,

$$\langle I \rangle_n := \frac{1}{n} \sum_{k=1}^n \left\langle I \left(\mu_t^{(k)} \right) \right\rangle, \quad (7.13)$$

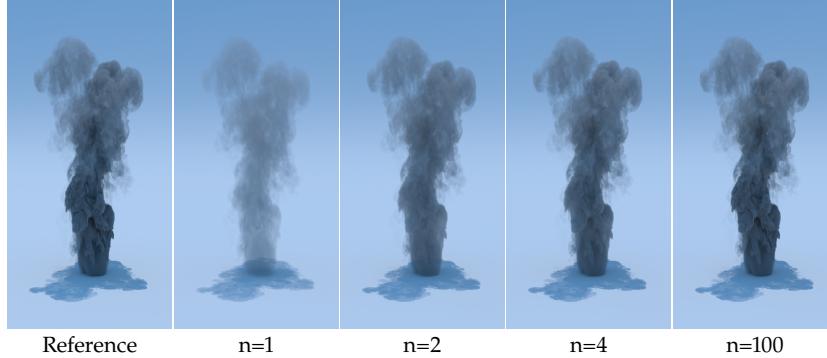


Figure 7.8: We visually illustrate an example of how taking the average of n renders while linearly increasing the total extinction can result in converging images. For visualization purposes, each iteration is rendered to near convergence instead of just one pixel sample.

where the images are rendered using a sequence of non-decreasing total extinctions, $\{\mu_t^{(k)}\} := \{\mu_t^{(1)}, \mu_t^{(2)}, \dots, \mu_t^{(n)}\}$, where we denote $\mu_t^{(k)}$ as the total extinction used to render image k . As long as $\mu_t^{(k)}$ bounds $\mu_r(x)$ after some finite iteration j , the bias will disappear in the limit of n ,

$$I = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{k=1}^n \left\langle I \left(\mu_t^{(k)} \right) \right\rangle, \quad (7.14)$$

since an infinite number of unbiased iterations will overpower any bias accumulated from the first j iterations. Thus, this progressive formulation will always be consistent (see formal consistency proof in Sec. 7.3.2).

A naive choice for $\mu_t^{(k)}$ which will satisfy our necessary conditions for consistency is to set $\mu_t^{(k)}$,

$$\mu_t^{(k+1)} = c \mu_t^{(k)}, \quad (7.15)$$

such that it grows linearly with the pixel sample index for any constant $c > 1$. We visualize how this hypothetical progressive estimator behaves in Fig. 7.8. Unfortunately, Eq. (7.15) is impractical since the cost of rendering the image increases as $\mu_t^{(k)}$ increases. In Sec. 7.3.3, we will introduce a practical adaptive updating scheme for the total extinction which is cost effective and useful in practice.

7.3.2 Asymptotic analysis

Proof of consistency. To prove Eq. (7.14), we make the assumption that there exists some finite iteration $k = j$ where all subsequent iterations,

$$I \left(\mu_t^{(k)} \mid k > j \right) = I \quad (7.16)$$

become unbiased. Thus, only the first j iterations will introduce any bias to a secondary estimator. Based on this assumption, let us consider the expected value of our progressive secondary estimator (7.13) as,

$$I_n := \frac{1}{n} \sum_{k=1}^n I \left(\mu_t^{(k)} \right). \quad (7.17)$$

Since only the first j iterations are biased, we can split the average into the contributions from the biased and unbiased terms separately,

$$I_n = \left(\frac{1}{n} \sum_{k=1}^j I(\mu_t^{(k)}) \right) + \left(\frac{1}{n} \sum_{k=j+1}^n I(\mu_t^{(k)}) \right), \quad (7.18)$$

where we can replace the unbiased terms with the true expected value (I) then decompose the biased terms,

$$I_n = \left(\frac{1}{n} \sum_{k=1}^j I(\mu_t^{(k)}) + (I(\mu_t^{(k)}) - I) \right) + \left(\frac{1}{n} \sum_{k=j+1}^n I \right), \quad (7.19)$$

into the sum of the true expected value and an iteration dependent bias $(I(\mu_t^{(k)}) - I)$. By grouping all I terms together,

$$\begin{aligned} I_n &= \left(\frac{1}{n} \sum_{k=1}^j I(\mu_t^{(k)}) - I \right) + \left(\frac{1}{n} \sum_{k=1}^n I \right) \\ I_n &= \left(\frac{1}{n} \sum_{k=1}^j I(\mu_t^{(k)}) - I \right) + I \end{aligned} \quad (7.20)$$

we derive an expression for the bias of a general progressive formulation after n iterations as,

$$B[I_n] := I - I_n = \left(\frac{1}{n} \sum_{k=1}^j I(\mu_t^{(k)}) - I \right). \quad (7.21)$$

Since the first ($k = 1$) iteration will have the most bias³, we can derive an upper bound on the bias of our progressive formulations by replacing the bias at every iteration with the bias of the first iteration,

$$B^+[I_n] := (I - I_n)^+ = \frac{j}{n} (I(\mu_t^{(1)}) - I). \quad (7.22)$$

If both the bias of the first iteration $(I(\mu_t^{(1)}) - I)$ and the iteration $k = j$ where all subsequent iterations become unbiased is finite, then it becomes trivially true that the bias of a progressive formulation,

$$\lim_{n \rightarrow \infty} B^+[I_n] = \lim_{n \rightarrow \infty} \frac{j \cdot (I(\mu_t^{(1)}) - I)}{n} = 0, \quad (7.23)$$

will disappear in the limit, since the numerator will be a finite value while the denominator will approach infinity. Thus, *any* progressive secondary estimator whose returned iterations become unbiased estimators after any finite point in time will be consistent. We provide an empirical example of Eq. (7.23) in Fig. 7.9 where we purposely bias the first 16 invocations of a transmittance estimator.

While progressive estimators have appeared in computer graphics previously, to the knowledge of the author of this dissertation, the original

3: We assume the bias of each subsequent iteration is non-increasing by construction since $\mu_t^{(k)}$ is non-decreasing, thus it is correct to assume that iteration $k = 1$ will have the most bias.

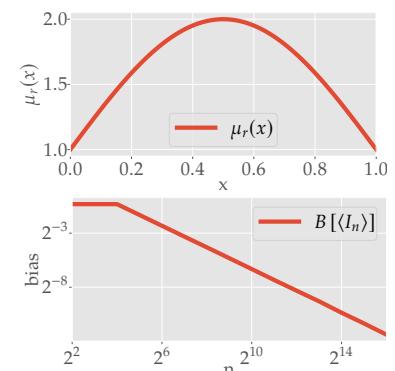


Figure 7.9: An empirical example of the bias converging (bottom) for a secondary estimator of transmittance ($I = \text{Tr}(x_a, x_b)$) where the initial 16 primary estimates are hard coded to incorrectly return $\text{Tr}(x_a, x_b) = 1$, while the remaining primary estimators employ ratio tracking (7.7) through a medium where $\mu_r(x)$ is given by the top graph.

[115]: Misso et al. (2023), ‘Progressive null-tracking for volumetric rendering’

work this contribution was from [115] was the first instance of progressive formulations which were designed around their iterations independently becoming unbiased after a finite amount of time. All other prior instances of progressive estimation were applied to problems whose secondary iterations were *always* biased. Such situations require more rigorous proofs of consistency, and we will analyze such problems later in Chapter 8 when dealing with infinite work-normalized variance.

Asymptotic analysis of MSE convergence. While Eq. (7.23) proves the consistency of our progressive formulations, we need to consider how the introduction of bias will impact the performance of our estimators in practice. We choose to do so by analyzing the asymptotic convergence rates of the mean-squared-error (3.28) as a function of n ,

$$\text{MSE}[\langle I \rangle_n] := V[\langle I \rangle_n] + B^2[\langle I \rangle_n], \quad (7.24)$$

since the MSE quantifies the impact of both the variance and bias. We additionally choose to only consider the asymptotic behavior of our progressive formulation (i.e. $n > j$) since it is easier to analyze the expected performance after the demarcation of biased and unbiased iterations has been reached.

In Eq. (7.22), we derived that the bias of our progressive formulations,

$$B[\langle I \rangle_n] := (I - I_n) = \mathcal{O}(n^{-1}), \quad (7.25)$$

will behave in a predictable manner. This also implies that the squared bias,

$$B^2[\langle I \rangle_n] := (I - I_n)^2 = \mathcal{O}(n^{-2}) \quad (7.26)$$

will disappear at a rate of $\mathcal{O}(n^{-2})$ as more progressive iterations are evaluated.

The variance of our secondary progressive estimator (7.13) will simplify,

$$\begin{aligned} V[\langle I \rangle_n] &:= V\left[\frac{1}{n} \sum_{k=1}^n \left\langle I\left(\mu_t^{(k)}\right)\right\rangle\right] \\ &= \frac{1}{n^2} \sum_{k=1}^n V\left[\left\langle I\left(\mu_t^{(k)}\right)\right\rangle\right], \end{aligned} \quad (7.27)$$

to the n^{-2} attenuated sum over the variance of each iteration, assuming that each iteration is evaluated independently. However, the variances of the iterations will not be the same since the first j iterations will effectively be rendering different scenes, while the variances of the last $n - j$ iterations will be identical, thus,

$$V[\langle I \rangle_n] = \frac{1}{n^2} \left(\left(\sum_{k=1}^j V\left[\left\langle I\left(\mu_t^{(k)}\right)\right\rangle\right] \right) + (n - j)V\left[\left\langle I\left(\mu_t^{(n)}\right)\right\rangle\right] \right). \quad (7.28)$$

Let us define V^+ such that,

$$V^+ := \max \left\{ V \left[\left\langle I \left(\mu_t^{(k)} \right) \right\rangle \right] : k = 1 \dots n \right\}, \quad (7.29)$$

it is the maximum of all iteration variances. Assuming that the variance of every iteration is independently finite,

$$\begin{aligned} V[\langle I \rangle_n] &< \frac{1}{n^2} (n V^+) \\ V[\langle I \rangle_n] &< \frac{V^+}{n} \\ V[\langle I \rangle_n] &\approx \mathcal{O}(n^{-1}), \end{aligned} \quad (7.30)$$

which is equivalent to the typical Monte Carlo convergence rate for the variance of secondary estimators with only unbiased primary estimators.

Given both the asymptotic convergence rates for the squared bias (7.26) and the variance (7.30), we can compute the asymptotic convergence rates for the MSE of our progressive estimator (7.13) as,

$$\begin{aligned} \text{MSE}[\langle I \rangle_n] &= V[\langle I \rangle_n] + B[\langle I \rangle_n]^2 \\ &\approx \mathcal{O}(n^{-1}) + \mathcal{O}(n^{-2}) \\ &\approx \mathcal{O}(n^{-1}). \end{aligned} \quad (7.31)$$

Thus, even though we introduce bias through the use of our progressive technique, the mean squared error will eventually be dominated by the *variance* instead of the bias. Once our combined extinction μ_t has converged to become bounding, the squared bias will converge at a rate which is an order of magnitude faster than the variance. This implies that in practice, it would be ideal to initialize μ_t and employ updating strategies which discover bounding extinctions quickly.

Discussion. The asymptotic convergence rates for both the bias (7.26) and MSE (7.31) imply two fascinating insights. Firstly, the initial bias of Eq. (7.13) will be *insignificant* to the estimator's overall performance in the limit of pixel samples (see 7.10) causing the MSE to become dominated by the *average variance of the unbiased iterations*. Secondly, this means that in the limit of pixel samples, the performance of Eq. (7.13) will converge to the performance of an originally unbiased estimator, which we empirically show in Fig. 7.11 with ratio tracking.

7.3.3 Adaptive Approach

From our analysis in Sec. 7.3.2 and the insights from our naive updating strategy for μ_t in Eq. (7.15), it is apparent that we would like a progressive sequence $\{\mu_t^{(k)}\}$ which reaches a bounding μ_t quickly and then stops increasing μ_t to maintain low cost. Fortunately, the individual images we compute as iterations in our progressive formulation (7.13) become

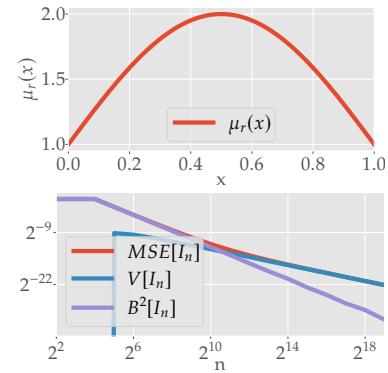


Figure 7.10: An empirical example of the convergence of squared bias, variance, and MSE for an initially biased secondary transmittance estimator. We employ the same experiment configuration as Fig. 7.9.

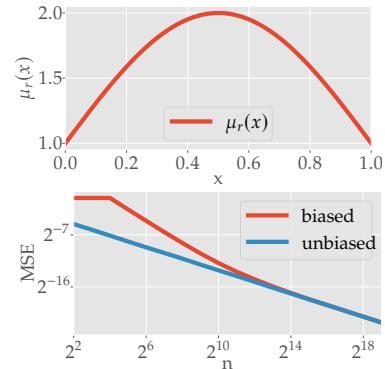


Figure 7.11: An empirical example of the MSE convergence for an unbiased secondary transmittance estimator and another secondary estimator whose first 16 primary estimates are hard coded to be biased. We use the same experiment configuration as Fig. 7.9.

unbiased once we reach $\mu_t^{(k)} \geq \mu_r$, allowing us considerable flexibility in adaptively updating $\mu_t^{(k)}$ from one iteration to the next.

To accomplish these goals, we choose to update μ_t only after encountering non-bounding real extinctions in the previous rendering iteration. This translates into an adaptive updating scheme,

$$\mu_t^{(k+1)} = \max_{\forall x^{(k)}}(\mu_t^{(k)}, \mu_r(x^{(k)}) + \epsilon), \quad (7.32)$$

where $x^{(k)}$ is any medium lookup performed in iteration k , and $\epsilon > 0$ is a small user-defined constant.

Unfortunately, while rendering black box extinction functions there is no way of knowing when $\mu_t^{(k)}$ has become a truly bounding extinction where all subsequent renders are unbiased. We instead choose to include the user-defined constant (ϵ) to *probabilistically* guarantee that we will *eventually* discover a bounding extinction.

Assuming that there is a non-zero measure of points within an ϵ of the real extinction function's maximum, Eq. (7.32) guarantees that a bounding extinction is expected to be found in finite time by,

$$\mathbb{P} [\text{Max}(\mu_r(x)) - \mu_r(x_\epsilon) < \epsilon] > 0. \quad (7.33)$$

In other words, all extinction functions where we have a non-zero chance of randomly sampling a position x whose extinction is close to (but not necessarily equal to) the peak extinction are supported by our adaptive strategy. If $\epsilon = 0$, then the probability of sampling exactly the maximum real extinction could become effectively zero for some extinction functions, meaning we would not expect to find a bounding extinction in finite time. We provide an example of this in Fig. 7.12 where we run both ratio tracking and the PCume transmittance estimators for 2^{20} iterations and plot at which iteration they discover bounding μ_t while employing our adaptive updating scheme (7.32).

While our formulation may still remain consistent in the case where $\epsilon = 0$, proving so would require a significantly more thorough proof than what we provided in Sec. 7.3.2. We leave this as a potential challenge to the interested reader.

Our adaptive updating strategy accounts for all extinction functions typically employed in production, like tri-linearly or tri-cubically interpolated, layered grids, and most procedural manipulations. The user-defined constant also provides practitioners complete control over the time it takes to discover bounding extinctions versus the additional render time incurred by having more loose μ_t .

In summary, our entire progressive null-tracking method can be practically described as replacing all extinction evaluations in a renderer with Algorithm 11 which returns a potentially clamped density, and an updated total extinction for use in the next pixel sample. Thus, our progressive null-tracking approach is an example of ④ in Fig. 1.1.

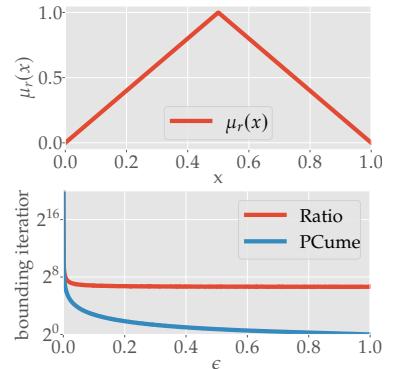


Figure 7.12: We plot the iterations where both ratio tracking and PCume discover bounding μ_t (bottom) for the given real extinction function (top) while starting from $\mu_t^{(1)} = 0.01$. When $\epsilon = 0$, neither estimator discovers a bounding extinction and the maximum iterations (2^{20}) is recorded. PCume is a power series estimator, and as such it always chooses to evaluate μ_r at least once every iteration while ratio tracking only evaluates μ_r when it samples $x_t < x_b$. Since the initial mean-free-path of ratio tracking is 100, this explains why ratio tracking takes longer to discover bounding μ_t .

Algorithm 11: Progressive medium extinction evaluation

```

1  $\mu_r^+ \leftarrow \text{Min}(\mu_r(\mathbf{x}_t), \mu_t^{(k)})$ 
2  $\mu_t^{(k+1)} \leftarrow \text{Max}(\mu_t^{(k)}, \mu_r(\mathbf{x}_t) + \epsilon)$ 
3 return  $\mu_r^+, \mu_t^{(k+1)}$ 

```

$\mu_t(\mathbf{x})$ in practice. While we have been referring to μ_t as a global constant throughout this dissertation, in practical implementations it tends to be specified as a spatially varying piecewise constant function ($\mu_t(\mathbf{x})$). It is common for $\mu_t(\mathbf{x})$ to be stored within a coarse grid-like data structure where each voxel (v) within the grid contains a local measure for $\mu_t(v)$ to more tightly fit the medium [104, 125]. In the context of our progressive techniques, each voxel contains its own independent progressive sequence $\{\mu_t^{(k)}\}$ of total extinctions which are independently updated using Eq. (7.32).

For the remainder of this chapter, we will assume that $\mu_t(\mathbf{x})$ takes the form of a piecewise constant function defined on a low-resolution grid that gets scaled to fit the spatial extents of the underlying medium.

[104]: Novák et al. (2014), ‘Residual Ratio Tracking for Estimating Attenuation in Participating Media’

[125]: Szirmay-Kalos et al. (2011), ‘Free Path Sampling in High Resolution Inhomogeneous Participating Media’

7.4 Progressive NEE in black box media

In Sec. 5.1.2, we introduced the concept of NEE and the performance benefits it provides to unidirectional rendering algorithms (see Fig. 5.2). NEE works by connecting intermediate vertices along a path to sampled points on emissive sources through tracing shadow connections. This allows rendering algorithms to iteratively accumulate radiance instead of waiting until an emissive source is interacted with to compute the final estimate (see Fig. 5.3 for a surface-only example).

Unfortunately, a significant problem arises when dealing with black box emissive media which ultimately prohibits NEE from being robustly employed in practice. If an emissive medium has a truly black box real extinction, then there is no way of knowing where the emissive portions of the volume are located to importance sample them. Without a reliable way of performing emitter sampling, NEE cannot be employed.

To alleviate this problem, we propose a progressive technique for building an on-the-fly registry of emissive regions within black box media to allow for efficient applications of NEE. Unlike progressive extinctions (Sec. 7.3), progressive NEE can be made unbiased (assuming we possess an initially bounding μ_t) without any significant increases in cost or variance through directly accounting for the bias during rendering.

While NEE can be employed in many different ways within volumetric rendering, we choose to compute NEE through line integration by employing a technique proposed by prior work [133] as their construction is easily adaptable into our existing progressive framework from Sec. 7.3.

The remainder of this section is organized as follows, we provide a brief overview of volumetric NEE using both point-sampling and line integration

[133]: Simon et al. (2017), ‘Line Integration for Rendering Heterogeneous Emissive Volumes’

in Sec. 7.4.1. Afterwards, we propose a biased progressive variation of NEE in Sec. 7.4.2 allowing for NEE to be applied to emissive procedural media. We conclude this section in Sec. 7.4.3 by proposing an unbiased alternative where we directly account for the bias on-the-fly.

7.4.1 NEE through line integration

Point-sampled NEE. Let us initially assume that we are rendering a scene containing emissive media and emissive surfaces where we have apriori knowledge that allows us to directly importance sample all emissive points. We can apply point-based NEE from some intermediate vertex, \mathbf{x} , by formulating the direct contribution from all shadow connections to emissive points (\mathbf{z}) as,

$$L_{\text{NEE}}(\mathbf{x}, \vec{\omega}_o) := \int_{\mathbb{V} \cup \mathbb{A}} \rho_{\star}(\mathbf{x}, \vec{\omega}_o, \vec{\mathbf{x}}\mathbf{z}) T_r(\mathbf{x}, \mathbf{z}) G(\mathbf{x}, \mathbf{z}) L_e(\mathbf{z}, \vec{\mathbf{x}}\mathbf{z}) d\mu(\mathbf{z}), \quad (7.34)$$

where the generalized scattering function (ρ_{\star}) is defined as,

$$\rho_{\star}(\mathbf{x}, \vec{\omega}_o, \vec{\mathbf{x}}\mathbf{z}) := \begin{cases} \rho_s(\mathbf{x}, \vec{\omega}_o, \vec{\mathbf{x}}\mathbf{z}), & \mathbf{x} \in \mathbb{A} \\ \mu_s(x_j) \rho_p(\mathbf{x}, \vec{\omega}_o, \vec{\mathbf{x}}\mathbf{z}), & \mathbf{x} \in \mathbb{V}, \end{cases} \quad (7.35)$$

the geometry term ($G(\mathbf{x}, \mathbf{z})$) is given by Eq. (4.25), the emission is defined as,

$$L_e(\mathbf{z}, \vec{\mathbf{x}}\mathbf{z}) := \begin{cases} L_e(\mathbf{z}, \vec{\mathbf{x}}\mathbf{z}), & \text{if } \mathbf{z} \in \mathbb{A}, \\ \mu_a(\mathbf{z}) L_e(\mathbf{z}, \vec{\mathbf{x}}\mathbf{z}), & \text{if } \mathbf{z} \in \mathbb{V}, \end{cases} \quad (7.36)$$

and the measure is defined to be,

$$d\mu(\mathbf{z}) := \begin{cases} dA(\mathbf{z}), & \mathbf{z} \in \mathbb{A}, \\ dV(\mathbf{z}), & \mathbf{z} \in \mathbb{V}. \end{cases} \quad (7.37)$$

We can now formulate an estimator for point-sampled NEE as,

$$\langle L_{\text{NEE}}(\mathbf{x}, \vec{\omega}_o) \rangle_{\text{point}} := \frac{\rho_{\star}(\mathbf{x}, \vec{\omega}_o, \vec{\mathbf{x}}\mathbf{z}) \langle T_r(\mathbf{x}, \mathbf{z}) \rangle G(\mathbf{x}, \mathbf{z}) L_e(\mathbf{z}, \vec{\mathbf{x}}\mathbf{z})}{p(\mathbf{z})}, \quad (7.38)$$

by point-sampling using some PDF $p(\mathbf{z})$, and then estimating the real transmittance to the emissive point⁴. We assume that importance sampling $p(\mathbf{z})$ is trivial and naively supported by the renderer.

NEE through line integration. Due to its dimensionality, sampling emissive points within a volume tends to be inefficient compared to point-sampling on surfaces. Within a volume, emission can occur anywhere along a path segment while in the case of surface-only transport, emission will only occur at the endpoint. Thus, it is usually more efficient to importance sample a direction, or a line, instead of singular points.

4: Note, it is implied through our derivations in Sec. 6.2.7 that the null-scattering form of Eq. (7.34) can easily be derived by replacing $T_r(\mathbf{x}, \mathbf{z})$ with the null-scattering Volterra (6.25) then simplifying. We choose to employ the non-null formulation since it is notationally simpler.

Unfortunately, importance sampling emission along a line is not as simple as importance sampling a single point. To do so, we need to first convert the NEE integral over points from Eq. (7.34) into one over directions. However, we *only* want to integrate over the set of directions pointing towards emissive parts of the scene similarly to how we only importance sample emissive points in Eq. (7.38). We denote this reduced set of directions along the sphere towards emissive vertices as \mathbb{S}_e^2 . Applying the necessary change of variables to Eq. (7.34) results in,

$$L_{\text{NEE}}(\mathbf{x}, \vec{\omega}_o) := \int_{\mathbb{S}^2} \rho_\star(\mathbf{x}, \vec{\omega}_o, \vec{\omega}_i) \int_0^d T_r(\mathbf{x}, \mathbf{x}_t) L_e(\mathbf{x}_t, -\vec{\omega}_i) dt d\Omega(\vec{\omega}_i), \quad (7.39)$$

which integrates over all directions and valid distances along those directions. The endpoint (\mathbf{x}_d) and distance (d) are given by the raytracing operator and can either be a point on a (potentially emissive) surface, or the end of the participating medium if no surface interaction occurs, thus, the emission is still defined by Eq. (7.36).

We can formulate another point-sampled alternative to Eq. (7.38) by importance sampling a distance and direction resulting in,

$$\langle L_{\text{NEE}}(\mathbf{x}, \vec{\omega}_o) \rangle_{\text{point}} := \frac{\rho_\star(\mathbf{x}, \vec{\omega}_o, \vec{\omega}_i) \langle T_r(\mathbf{x}, \mathbf{x}_t) \rangle L_e(\mathbf{x}_t, -\vec{\omega}_i)}{p(t)p(\vec{\omega}_i)}, \quad (7.40)$$

which has similar drawbacks to Eq. (7.38). We can alternatively choose to only importance sample a direction,

$$\begin{aligned} \langle L_{\text{NEE}}(\mathbf{x}, \vec{\omega}_o) \rangle_{\text{line}} &:= \frac{\rho_\star(\mathbf{x}, \vec{\omega}_o, \vec{\omega}_i) \left\langle \int_0^d T_r(\mathbf{x}, \mathbf{x}_t) L_e(\mathbf{x}_t, -\vec{\omega}_i) dt \right\rangle}{p(\vec{\omega}_i)}, \\ &= \frac{\rho_\star(\mathbf{x}, \vec{\omega}_o, \vec{\omega}_i) \langle L_{\text{line}}(\mathbf{x}, -\vec{\omega}_i) \rangle}{p(\vec{\omega}_i)}, \end{aligned} \quad (7.41)$$

and estimate the line integral over emission directly. However, formulating an estimator for the line integral in Eq. (7.41) involves solving two problems. First we need the means of unbiasedly estimating $T_r(\mathbf{x}, \mathbf{x}_t)$ on-the-fly for the line integral to be efficient. Next, we need to devise a way of computing and importance sampling $p(\vec{\omega}_i)$. We will address these two points in order.

Line integral over transmittance attenuated emission. Prior work [133] has derived a weighted technique for computing the line integral over emission which we denote as,

$$L_{\text{line}}(\mathbf{x}, \vec{\omega}_o) := \int_0^d T_r(\mathbf{x}, \mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) dt, \quad (7.42)$$

however, it requires explicitly evaluating a voxel density grid and is not applicable to black box media. Other works [106] have proposed a modified emission tracker from a simplified version of the null-VRE (7.10). While we employ that same emission tracker in our practical implementation, we

[133]: Simon et al. (2017), ‘Line Integration for Rendering Heterogeneous Emissive Volumes’

[106]: Huang et al. (2021), ‘Unbiased Emission and Scattering Importance Sampling For Heterogeneous Volumes’

will now derive a theoretical Volterra formulation for Eq. (7.42) which can be used to derive general line integrators for emission.

We start by replacing the real transmittance in Eq. (7.42) with the null-Volterra formulation for the exponential (6.25) to arrive at,

$$L_{\text{line}}(\mathbf{x}, \vec{\omega}_o) := \int_0^d \left(T_t(\mathbf{x}, \mathbf{x}_t) + \int_0^t \mu_n(\mathbf{x}_s) T_t(\mathbf{x}, \mathbf{x}_s) T_r(\mathbf{x}_s, \mathbf{x}_t) ds \right) L_e(\mathbf{x}_t, \vec{\omega}_o) dt, \quad (7.43)$$

where we then split the result into the sum of two terms,

$$\begin{aligned} L_{\text{line}}(\mathbf{x}, \vec{\omega}_o) &:= L_1 + L_2 \\ L_1 &:= \int_0^d T_t(\mathbf{x}, \mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) dt + \\ L_2 &:= \int_0^d \int_0^t \mu_n(\mathbf{x}_s) T_t(\mathbf{x}, \mathbf{x}_s) T_r(\mathbf{x}_s, \mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) ds dt. \end{aligned} \quad (7.44)$$

Let us now consider L_2 in isolation,

$$L_2 = \int_0^d \int_0^t \mu_n(\mathbf{x}_s) T_t(\mathbf{x}, \mathbf{x}_s) T_r(\mathbf{x}_s, \mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) ds dt. \quad (7.45)$$

By reversing the order of integration to get,

$$L_2 = \int_0^d \int_s^d \mu_n(\mathbf{x}_s) T_t(\mathbf{x}, \mathbf{x}_s) T_r(\mathbf{x}_s, \mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) dt ds, \quad (7.46)$$

we can then move terms outside of the inner integral giving us,

$$L_2 = \int_0^d \mu_n(\mathbf{x}_s) T_t(\mathbf{x}, \mathbf{x}_s) \int_s^d T_r(\mathbf{x}_s, \mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) dt ds, \quad (7.47)$$

so that we can then replace the inner integral with Eq. (7.42),

$$L_2 = \int_0^d \mu_n(\mathbf{x}_s) T_t(\mathbf{x}, \mathbf{x}_s) L_{\text{line}}(\mathbf{x}_s, \vec{\omega}_o) ds. \quad (7.48)$$

Finally, we rename $s \rightarrow t$ to arrive at,

$$L_2 = \int_0^d \mu_n(\mathbf{x}_t) T_t(\mathbf{x}, \mathbf{x}_t) L_{\text{line}}(\mathbf{x}_t, \vec{\omega}_o) dt. \quad (7.49)$$

We now reconsider the sum of $L_1 + L_2$,

$$\begin{aligned} L_{\text{line}}(\mathbf{x}, \vec{\omega}_o) &:= L_1 + L_2 \\ &= \int_0^d T_t(\mathbf{x}, \mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) + \mu_n(\mathbf{x}_t) T_t(\mathbf{x}, \mathbf{x}_t) L_{\text{line}}(\mathbf{x}_t, \vec{\omega}_o) dt \end{aligned} \quad (7.50)$$

to arrive at a Volterra formulation for the line integral of emission which

can be generally estimated using the following generalized estimator,

$$\langle L_{\text{line}}(\mathbf{x}, \vec{\omega}_o) \rangle_{\text{General}} := \frac{1}{p(\mathbf{x}_t)} \left(T_t(\mathbf{x}, \mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) + \frac{\mu_n(\mathbf{x}_t) T_t(\mathbf{x}, \mathbf{x}_t) \langle L_{\text{line}}(\mathbf{x}_t, \vec{\omega}_o) \rangle}{P_{\text{rec}}} \right), \quad (7.51)$$

where P_{rec} represents the Russian Roulette continuation probability for recursing. The main three differences between Eq. (7.51) and our generalized Volterra formulation for estimating exponential transmittance (6.26) is that the combined transmittance term in Eq. (7.51) is *always* evaluated, the combined transmittance term does not span the entire path segment, and it is also multiplied by the emission.

We will now quickly rederive some transmittance estimators from Chapter 6 as line integrators for emission. Note that all instances of $p(t)$ are effectively unnormalized and a portion of P_{rec} renormalizes them (see Chapter 6 for a thorough explanation).

Ratio tracking and next-flight. Since the combined transmittance term in Eq. (7.51) is always evaluated, ratio tracking and next-flight ratio tracking are equivalently described as,

$$p(\mathbf{x}_t) = \rho_f^{(t)}(\mathbf{x}_t); \quad P_{\text{rec}} = 1 - T_t(\mathbf{x}_a, \mathbf{x}_t), \quad (7.52)$$

Null-scattering track length. A null-scattering track length estimator for integrating the emission is described as,

$$p(\mathbf{x}_t) = \rho_f^{(t)}(\mathbf{x}_t); \quad P_{\text{rec}} = \frac{\mu_n(t)}{\mu_t} (1 - T_t(\mathbf{x}_a, \mathbf{x}_t)), \quad (7.53)$$

and weighted track length integrator can be derived similarly using,

$$p(\mathbf{x}_t) = \rho_f^{(t)}(\mathbf{x}_t); \quad P_{\text{rec}} = \frac{\mu_n(\mathbf{x}_t)}{\mu_r(\mathbf{x}_t) + |\mu_n(\mathbf{x}_t)|} \cdot (1 - T_t(\mathbf{x}_a, \mathbf{x}_t)). \quad (7.54)$$

Adaptive ratio tracking. An adaptive ratio tracking line integrator can be described using,

$$p(\mathbf{x}_t) = \rho_f^{(\text{MC})}(\mathbf{x}_t); \quad P_{\text{rec}} = 1 - T_{\text{MC}}(\mathbf{x}_a, \mathbf{x}_t), \quad (7.55)$$

where μ_{MC} is set using the null-extinction from the last free-flight sample.

While surface emission is technically encoded in Eq. (7.51) at $L_e(\mathbf{x}_d, \vec{\omega}_o)$, we recommend accounting for it separately since the probability of sampling d is effectively 0. In practice, we employ ratio tracking (7.52) as our line integrator.

Importance sampling and computing $p(\vec{\omega}_i)$. To importance sample the integral over all emissive directions in Eq. (7.41), prior work [133] proposed a two step process. First we importance sample an emissive *point* within the medium as if we were estimating Eq. (7.34). Then we discard that point sample while retaining the direction towards that point as our sampled direction, $\vec{\omega}_i$. To then compute the PDF of choosing that direction, we have to directly account for the fact that an entire continuum of points along $\vec{\omega}_i$ could have been sampled by computing,

$$p(\vec{\omega}_i) := \int_0^s p(\mathbf{x}_t) t^2 dt, \quad (7.56)$$

where $p(\mathbf{x}_t)$ is the probability of sampling \mathbf{x}_t using point sampling and t^2 is the Jacobian for transforming from volume to solid angle measure [133].

Unfortunately, it is computationally intractable to compute Eq. (7.56) for general heterogeneous media. Instead, prior work [106, 133] has proposed to compute the directional PDF over a piecewise approximation of the underlying medium (see Algorithm 12). As long as $\vec{\omega}_i$ is a valid probability distribution defined over the entirety of \mathbb{S}_e^2 , any resulting estimator will still be unbiased.

Algorithm 12: Accumulating $p(\vec{\omega}_i)$ from all points along a line

```

1  $p(\vec{\omega}_i) = 0$ 
2 for all voxels  $v$  along  $\mathbf{x} + t\vec{\omega}_i$  do
3   | compute  $p_v(t)$  probability of point sampling voxel  $v$ 
4   | compute  $(t_a, t_b)$  voxel entry/exit points
5   |  $p(\vec{\omega}_i) = p(\vec{\omega}_i) + \frac{t_b^3 - t_a^3}{3} p_v(t)$ 
6 return  $p(\vec{\omega}_i)$ 
```

[133]: Simon et al. (2017), 'Line Integration for Rendering Heterogeneous Emissive Volumes'

[133]: Simon et al. (2017), 'Line Integration for Rendering Heterogeneous Emissive Volumes'

[106]: Huang et al. (2021), 'Unbiased Emission and Scattering Importance Sampling For Heterogeneous Volumes'

[133]: Simon et al. (2017), 'Line Integration for Rendering Heterogeneous Emissive Volumes'

7.4.2 Progressive volumetric NEE

Applying line integration based NEE requires us to compute directional sampling distributions across a piecewise constant approximation of the underlying medium. Conveniently, the data structure used to store our locally varying values for μ_t satisfies this requirement and can be adapted to allow us to progressively support NEE in black box emissive media.

Biased approach. Let us denote $1_{MC}(v)$ to be an indicator function defined across all voxels which,

$$1_{MC}(v) := \begin{cases} 1, & L_e > 0 \text{ for some } \mathbf{x} \in v \\ 0, & \text{otherwise} \end{cases} \quad (7.57)$$

indicates whether a specific voxel (v) contains emissive media or not. We employ the subscript "MC" to convey that this indicator function will be used to dictate a Monte Carlo sampling process and that the actual value it stores *can* be incorrect.

We employ Eq. (7.57) to choose which voxels to consider when computing the directional PDF using Algorithm 12 and while also sampling the initial point \mathbf{z} when importance sampling line integrals. When $1_{\text{MC}}(v) = 0$, the probability of point sampling a voxel will be zero ($p_v(t) = 0$), causing us to ignore any contribution that voxel would have to the NEE-contributed emission. Thus, any volumetric rendering algorithm which only accumulates emission through NEE will be biased unless $1_{\text{MC}}(v) = 1$ for all voxels containing emissive sources.

Progressive approach. Similarly to our approach for progressive extinctions (see Sec. 7.3), let us define,

$$\left\{1_{\text{MC}}^{(k)}(v)\right\} := \left\{1_{\text{MC}}^{(1)}(v), 1_{\text{MC}}^{(2)}(v), \dots, 1_{\text{MC}}^{(n)}(v)\right\}, \quad (7.58)$$

to be the sequence of indicator functions which are employed for each rendering pass of a progressive volumetric rendering algorithm where k is the index of the current pass. Let us assume that $1_{\text{MC}}^{(1)}(v)$ may be initialized incorrectly, resulting in bias. We propose to adaptively update the indicator function used in the next pass $\left(1_{\text{MC}}^{(k+1)}(v)\right)$ by *turning on* any disabled voxels when emissive material is encountered during extinction evaluations on the current pass. This process simplifies to updating our density evaluations from the previous section (see Algorithm 11) to account for this additional check which we algorithmically describe in Algorithm 13.

Algorithm 13: Progressive extinction evaluations with emissive indicator update

```

1  $\mu_r^+ \leftarrow \text{Min}(\mu_r(\mathbf{x}_t), \mu_t^{(k)})$ 
2  $\mu_t^{(k+1)} \leftarrow \text{Max}(\mu_t^{(k)}, \mu_r(\mathbf{x}_t) + \epsilon)$ 
3 if  $1_{\text{MC}}^{(k)}(v) = 0$  and  $\alpha_a(\mathbf{x}_t)L_e(\mathbf{x}_t, -\omega) > 0$  then
4    $1_{\text{MC}}^{(k+1)}(v) = 1$ 
5 return  $\mu_r^+, \mu_t^{(k+1)}, 1_{\text{MC}}^{(k+1)}(v)$ 
```

Choosing to update $1_{\text{MC}}^{(k)}(v)$ over the course of a render as we discover emissive voxels results in a biased but progressive rendering algorithm since our sampling PDF will only become a valid PDF once $1_{\text{MC}}^{(k)}(v)$ has converged. While we can naively make this process unbiased by initially setting $p_v(t)$ to some small, but finite, constant, this would ultimately be inefficient since it would result in wasting samples on non-emissive portions of the medium. We instead compensate for the bias in a different way.

7.4.3 Unbiased progressive volumetric NEE

We previously introduced an indicator function (7.57) dictating where we should perform emitter sampling when accumulating the radiance contribution from NEE. However, this indicator function is assumed to

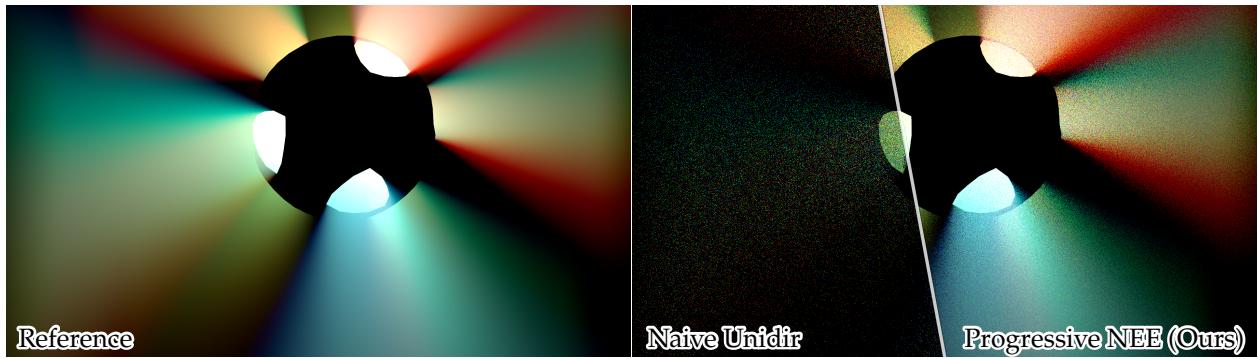


Figure 7.13: A comparison of our progressive NEE versus naively unidirectional volumetric path tracing in a scene that is lit by highly concentrated emissive regions of a medium which are mostly obscured from view. Our unbiased progressive NEE significantly improves upon what was previously possible for this type of scene configuration. The comparison renders are roughly equal time (5 minute) renders which is approximately 128 pixel samples each.

be noisy, meaning, that our importance sampling routine for sampling directions towards emissive portions of the medium might not cover the entire space of valid directions for any specific pass k .

However, one crucial insight is that the indicator function can be interpreted to dictate where we know with complete certainty there exists emissive material and what portions of the medium where we are unsure. We can effectively decompose the emissive contributions from a virtual scene,

$$L_e(\mathbf{x}, \vec{\omega}_o) = L_e^{\text{NEE}}(\mathbf{x}, \vec{\omega}_o) + L_e^{\text{ACC}}(\mathbf{x}, \vec{\omega}_o) \quad (7.59)$$

into the sum of a component computed through NEE and a component which accumulates the emission during tracking. However, only one of these components will be non-zero for any \mathbf{x} depending on the value of the indicator function,

$$L_e(\mathbf{x}, \vec{\omega}_o) := \begin{cases} L_e^{\text{NEE}}(\mathbf{x}, \vec{\omega}_o), & 1_{\text{MC}}(v) = 1 \\ L_e^{\text{ACC}}(\mathbf{x}, \vec{\omega}_o), & 1_{\text{MC}}(v) = 0, \end{cases} \quad (7.60)$$

to avoid double counting emission. Incorporating this decomposition within a volumetric path tracer with NEE involves setting $P_a = 1$ within the null-VRE (7.10), replacing L_e with L_e^{ACC} in the null-VRE, then additionally tracing shadow connections to compute L_e^{NEE} . Our unbiased progressive NEE is an instance of ⑤ from our recipe for consistent estimation (Fig. 1.1), and we show that re-enabling NEE through our progressive method can lead to huge performance improvements in Fig. 7.13.

7.5 Results

All results included in this chapter have been implemented within the research oriented renderer, PBRT [18]. We store the spatially varying μ_t by employing an 80^3 grid fit to the medium extents where each grid cell, or voxel, stores its own estimate of μ_t for the portion of the medium it spans.

[18]: Pharr et al. (2016), *Physically Based Rendering*

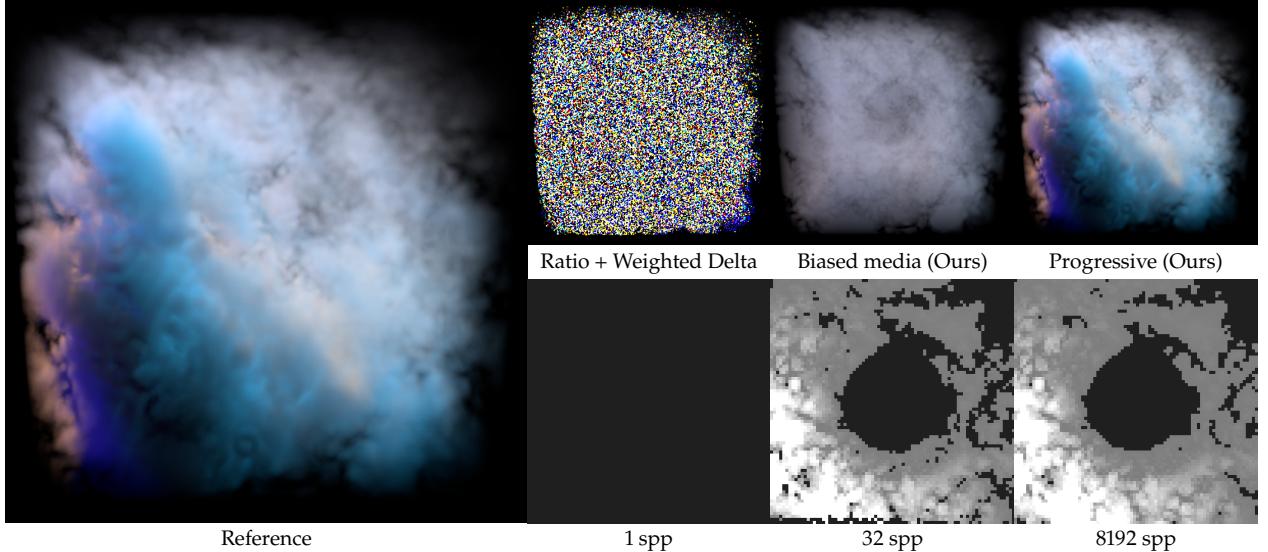


Figure 7.14: Existing methods like ratio tracking and weighted delta tracking (top, left) suffer from extreme variance if the total extinction ($\mu_t = 0.01$) significantly underestimates the maximum density of the medium ($\mu_t \approx 3.0$). Starting with the same poor estimate for a bounding extinction ($\mu_t = 0.01$), we propose to instead clamp the medium density to the chosen μ_t . This allows fast, low-variance rendering, but of a modified (biased) medium (top, center). We then progressively update the estimates for μ_t (bottom row) to rapidly reduce this bias and ensure that the running average (top right) across multiple pixel samples converges to the correct result in the limit.

When rendering pass k we update the total extinction $\mu_t^{(k+1)}$ for the next pass independently for each voxel.

For all of our comparisons, we choose to initialize the extinctions for all voxels to be near zero ($\mu_t^{(1)} = .01$) when employing our progressive extinctions (7.32). This corresponds to the near-worst case scenario for prior methods, and we choose to do so for the purpose of conveying the robustness of our technique. In practical implementations, we recommend initializing μ_t to roughly match the medium for best initial performance, however, as we saw in Fig. 7.11, we would expect the performance of our method to approach that of an unbiased technique in the limit.

The only exception to this is Fig. 7.13 where we observe the benefits of our unbiased progressive NEE against a naively unidirectional rendering algorithm where we assume prior knowledge of a tightly bounding μ_t . This assumption is not required to employ our progressive NEE since it would be trivial to combine it with our progressive extinctions from Sec. 7.3, however, it is required for our progressive NEE to be fully unbiased. Similar to the naively unidirectional versus NEE surface-only case we saw previously in Fig. 5.2, our unbiased progressive technique significantly improves upon the naive alternative allowing for efficient emitter sampling in emissive black box media.

In Fig. 7.14, we visualize just how poorly prior work performs for the same worst-case scenario our progressive null-tracking is initialized to by comparing our progressive technique against a combination of ratio tracking and weighted delta tracking in a scene containing a fully procedural medium. While prior methods exhibit numerically infinite variance, which can never converge in any practical amount of time, our technique effectively removes

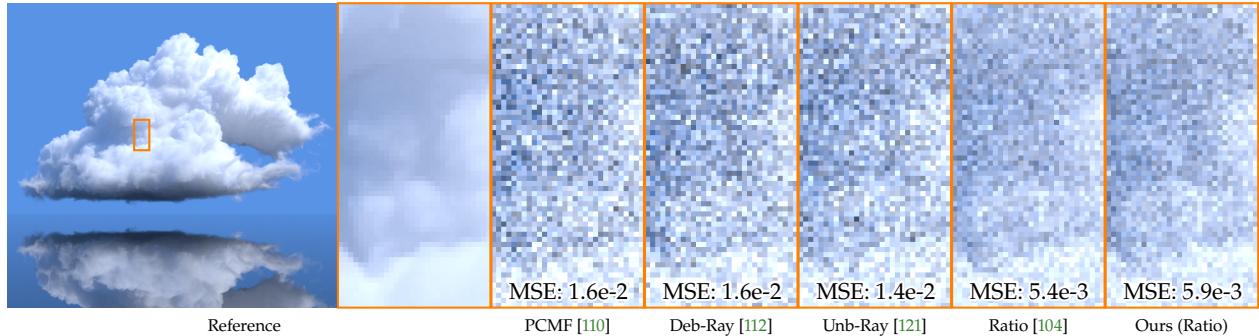


Figure 7.15: An equal extinction lookups comparison of our progressive method initialized with near-zero total extinction versus many of the state-of-the-art transmittance estimators given bounding extinctions for a highly scattering medium. All methods employ weighted-delta tracking for free-flight sampling. MSE values are computed as the average MSE across the entire image.

this variance through the two-step process of biasing the medium and then progressively discovering a bounding μ_t at render-time to eventually remove the bias.

In Fig. 7.15, we compare our progressive null-tracking technique to the state of the art in transmittance estimation in a scene containing highly scattering media. Most of the more recent transmittance estimators such as unbiased raymarching [121], debiased raymarching [112] (which we introduce later in Chapter 8), and PCMF [110] trade lower variance transmittance estimates for higher costs. However, in highly scattering scenes such as this one which are very common in production rendering, the majority of the noise in the renders are attributed to other sources outside of transmittance estimation so employing a cheaper technique like ratio tracking is currently more viable in these scenes. Our progressive approach with near-worst case initializations for μ_t still maintains similar performance to ratio tracking with perfect knowledge of bounding μ_t while beating the high-cost low-variance alternatives.

In Fig. 7.16, we instead give ratio tracking the benefit of knowing the true bounds apriori, and despite using a near-worst case initialization, our technique very quickly discovers bounding values for μ_t (top graph), and reports a nearly equivalent MSE to ratio tracking (bottom graph). The convergence of our technique greatly benefits from the fact that every path generated during a render can contribute to discovering bounding μ_t for all voxels. For a standard 1920x1080 resolution image, a single pixel sample spawns over 2 million paths which cause enough bounding extinctions to have populated the combined extinction grid that any bias is both visually and numerically negligible after just a few pixel samples. Our progressive approach maintains the same good performance of prior techniques while successfully making them *resilient* to non-bounding extinctions.

[121]: Kettunen et al. (2021), ‘An Unbiased Ray-Marching Transmittance Estimator’

[112]: Misso et al. (2022), ‘Unbiased and consistent rendering using biased estimators’

[110]: Georgiev et al. (2019), ‘Integral Formulations of Volumetric Transmittance’

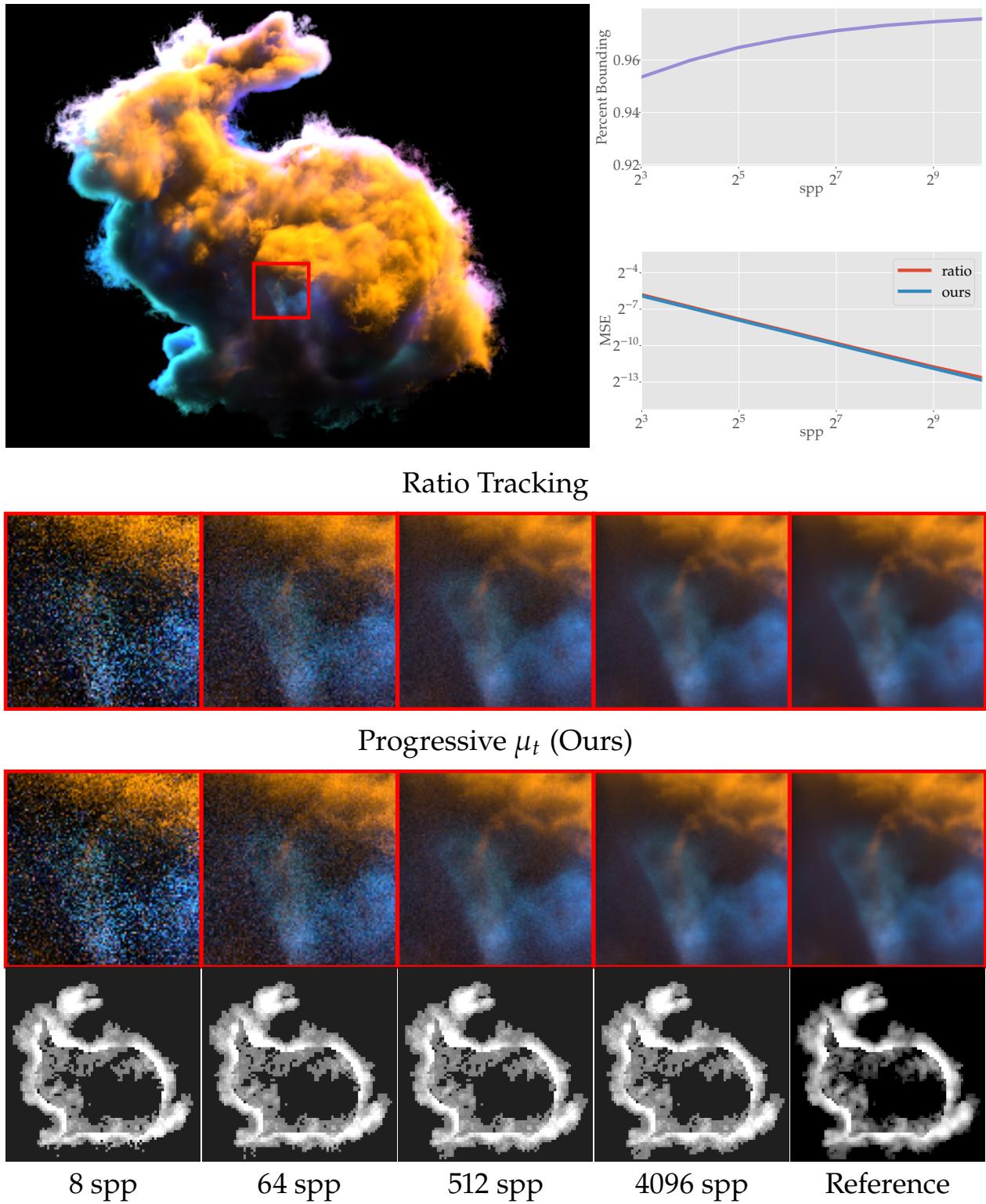


Figure 7.16: We compare the performance of our progressive estimator to ratio-tracking while also showing the convergence of our super-voxel grid. We visualize a slice of the super-voxel grid (bottom row) throughout a render. We also compare the convergence of the MSE of our method to ratio tracking (bottom graph) and convey the convergence of our super-voxel grid by plotting the percentage of voxels with bounding μ_t 's (top graph).

7.6 Conclusion

The original goal of this work was to alleviate the drawbacks of employing null-scattering based techniques in scenes where we have to render black box procedural media. While we previously enabled unbiased algorithms for rendering classical participating media (see Chapter 6), the Monte Carlo sampling process involved in those algorithms was previously not robust enough to handle black box extinctions for most transmittance estimators (see Fig. 7.5) and all current free-flight sampling routines. By exploring the space of progressive, but consistent solutions, we have proposed techniques which make most current unbiased volumetric rendering algorithms completely resilient to non-bounding extinctions in return for a small amount of initial bias which disappears rapidly. We additionally provided significant efficiency improvements through re-enabling variance reduction techniques like NEE in cases where we do not know where the emissive sources are initially.

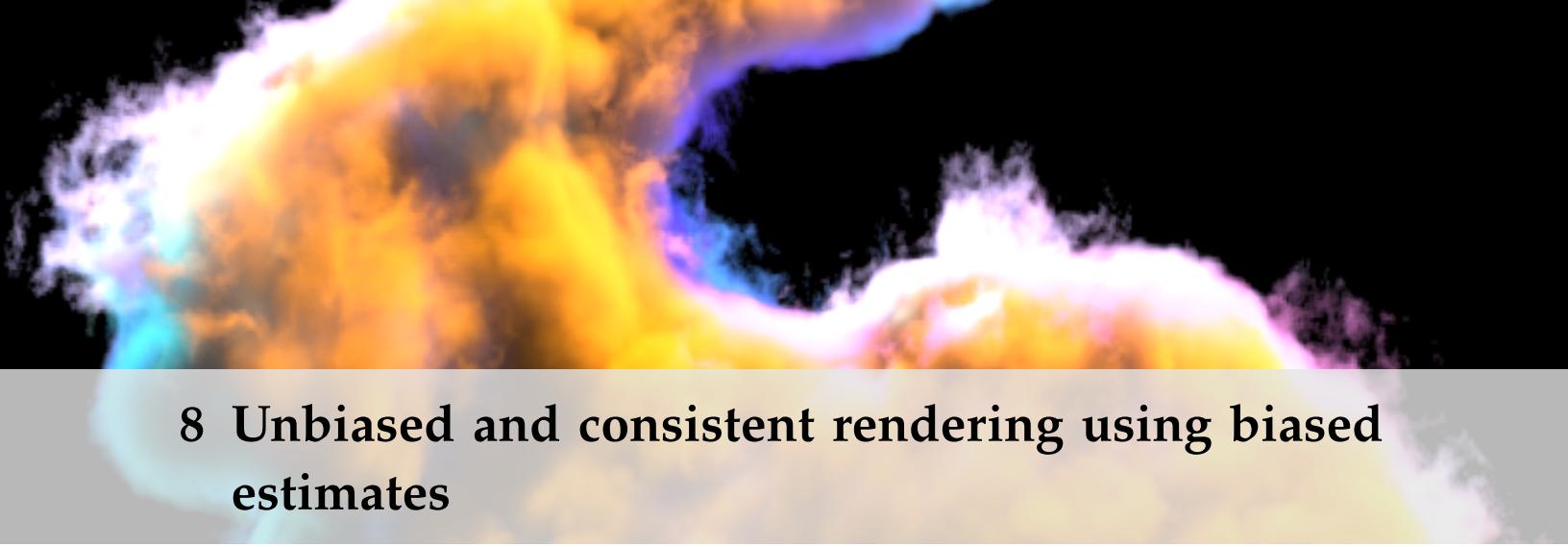
We accomplished this while exploring two different variations of consistent solutions for non-linear problems of expectations: biased progressive estimators ④, and progressive estimators which can be made unbiased by directly accounting for their bias ⑤.

7.6.1 Future work

For future work, we plan on exploring different strategies for initializing, clamping, and updating bounding μ_t 's. Currently, we conservatively clamp densities to ensure the total extinction is bounding, but our analysis in Sec. 7.2 implies that allowing some negative null-density (as long as the medium remains only slightly non-bounding) may further reduce bias while still ensuring bounded variance. Additionally, we plan on expanding our null-tracking method to support residual trackers, which would simply involve storing additional local information other than μ_t .

While line integration is a good sampling technique, it is not always the best emitter-sampling strategy in practice. Within extremely sparse, high-albedo media, a combination of point sampling with equiangular distance sampling tends to be preferred [133]. So, in terms of progressive NEE, we plan on looking into incorporating MIS to further improve our method.

[133]: Simon et al. (2017), ‘Line Integration for Rendering Heterogeneous Emissive Volumes’



8 Unbiased and consistent rendering using biased estimates

In the prior chapters (Chapter 6 and Chapter 7), we analyzed a specific instance of a prevalent non-linear application within light transport in the form of rendering classical participating media. Chapter 6 was mostly focused on deriving and improving upon unbiased solutions for efficient transmittance estimation through our consistent reformulations. Unfortunately, we observed through our analysis of non-bounding extinctions in Chapter 7 that there is no free lunch. While unbiased solutions with good performance could exist, the stochastic process of Monte Carlo integration did not guarantee that these solutions would always be robust. In such cases, we showed how loosening our constraints to allow biased but progressively consistent solutions could still lead to efficient algorithms.

One of the main goals of this dissertation is to provide a *general* recipe conveying how to approach formulating estimators for non-linear functions of expectations,

$$I := g \left(\int_{\Theta} f(x) dx \right) = g(F), \quad (8.1)$$

which arise in practice. In this chapter we aim to accomplish this by mostly focusing on generalizing the process we applied previously to rendering classical participating media. Afterwards, we will directly apply this generalized process to derive solutions for some novel applications relevant to the field of light transport. We will fulfill both of these tasks by enumerating and addressing each point comprising our recipe in order. For convenience, we restate our recipe for consistent estimation in Fig. 8.1.

The remainder of this chapter is organized in order of the points within our general recipe. In Sec. 8.1, we generalize the consistent formulations we have previously introduced in prior chapters while discussing their limitations and applicability. We additionally introduce another consistent formulation based on a telescoping series expansion which is more broadly applicable and can relate all aforementioned formulations to the idea of *debiasing* biased solutions. This covers step ① of our recipe. Afterwards in Sec. 8.2 we discuss a general technique for verifying that a given consistent solution can be estimated with finite variance. We do so by deriving the *asymptotically* optimal sampling distributions in terms of variance for all the

8.1 Generalized consistent formulations	153
8.1.1 Taylor series formulation	155
8.1.2 IVP formulation	158
8.1.3 Telescoping series formulation	159
8.1.4 Generalized consistent formulations	161
8.1.5 Progressive formulation	161
8.2 Verifying finite variance	162
8.2.1 General consistent estimation	162
8.2.2 Optimal probability distributions	163
8.2.3 Variance-optimal probability distributions . . .	164
8.2.4 Verifying finite work-normalized variance . . .	166
8.3 Handling finite work-normalized variance	166
8.4 Handling infinite work-normalized variance	168
8.4.1 Unbiased estimators under infinite variance .	169
8.4.2 Aside: Alleviating infinite variance through MIS . .	169
8.4.3 Progressive estimators under infinite variance .	170
8.5 Applications	171
8.5.1 Unbiased and progressive finite differencing	171
8.5.2 Unbiased transmittance estimation	175
8.5.3 Unbiased photon mapping	181
8.6 Conclusion	184
8.6.1 Discussion	184

- ① **Construct consistent formulation** Reformulate the non-linear problem as a consistent formulation which is amenable to Monte Carlo integration.
- ② **Prove finite work-normalized variance** Derive as close to a *theoretically* optimal primary estimator as possible for ①, and then determine if it has finite-work-normalized variance.

IF FINITE:

- ③ **Build unbiased finite-variance estimator:** Construct an unbiased primary estimator and a naive secondary estimator then test. If efficiency is unsatisfactory, continue to ④.
- ④ **Build consistent progressive estimator:** Construct a progressive secondary estimator from the consistent formulation which trades bias in return for lower cost, and/or lower variance. If possible, go to ⑤.
- ⑤ **Optional: Build unbiased consistent progressive estimator:** Derive a way to account for the bias of the progressive estimator in an unbiased manner.

IF INFINITE:

- ⑥ **Build unbiased infinite-variance estimator:** Construct a primary estimator by balancing the cost versus variance trade-off such that cost remains finite and variance approaches infinity as slow as possible. Construct a naive secondary estimator. If performance is acceptable, done, otherwise continue to ⑦.
- ⑦ **Build consistent progressive estimator:** Convert the secondary estimator into a consistent progressive estimator, and reparameterize the consistent formulation such that the variance grows sublinearly. The resulting estimator will maintain good efficiency in return for bias.

Figure 8.1: A restatement of our recipe for formulating consistent estimators for non-linear functions of expectations. We originally defined this recipe in Fig. 1.1.

consistent formulations from Sec. 8.1. Given these sampling distributions, we can generally determine whether or not a given problem will possess an unbiased primary estimator with finite work-normalized variance. This fulfills step ②. If a finite work-normalized variance estimator exists, we discuss the cases of handling its estimation in Sec. 8.3 covering points ③, ④, and ⑤. Since we have already addressed these points throughout the prior chapters, our discussion in Sec. 8.3 will mostly be a brief overview of what we have already discussed. The infinite work-normalized case is discussed in Sec. 8.4 covering steps ⑥ and ⑦ in Fig. 8.1. We apply our recipe to deriving novel solutions to non-linear problems of expectations in Sec. 8.5, and conclude the chapter in Sec. 8.6.

8.1 Generalized consistent formulations

Given a non-linear function of expectations (8.1), the first step (①) in our recipe for formulating consistent estimators involves reformulating the non-linear problem,

$$I := g(F) = \lim_{n \rightarrow \infty} I(n), \quad (8.2)$$

into a form which is expected to return the true solution *at* some limit $I = I(\infty)$. Thus, we refer to Eq. (8.2) as a *consistent formulation* where we assume by its construction that n is a parameter dictating the bias of $I(n)$. This implies that from any intermediate point j , we would expect to have a biased solution ($I(j) \neq I$)¹. In practice, if the limit cannot be evaluated in closed form, i.e. if $n \rightarrow \infty$ implies infinite work, then we have to derive ways of *estimating* the limit.

Since our consistent formulations are defined either directly or indirectly as limits (8.2), we can choose any intermediate starting point (j),

$$I := I(j) + \underbrace{\lim_{n \rightarrow \infty} I(n) - I(j)}_{B(j)}. \quad (8.3)$$

By denoting consistent formulations using Eq. (8.3) as a basis, they can be perceived as a means for *debiasing* biased solutions. This is because the initial evaluation ($I(j)$) is a biased solution since we assume $I(j) \neq I$, while the limit is exactly equal to the bias ($B(j)$). By combining unbiased estimates of the two quantities,

$$\langle I \rangle := \langle I(j) \rangle + \langle B(j) \rangle, \quad (8.4)$$

we will obtain an unbiased estimate for I . All of the various consistent formulations which we consider in this dissertation fit within this paradigm of debiasing (8.3). However, they all differ by how they define $I(j)$ and $B(j)$.

Throughout this dissertation, we have observed three different instances of consistent formulations. The first two were formulations for primary estimation consisting of the Volterra formulation and the Maclaurin series expansion for classical exponential transmittance (see Chapter 6). Later in this section, we will walk through how both of these formulations can be reinterpreted as instances of *debiasing* (8.3). Additionally, a general secondary Monte Carlo estimator is a consistent formulation that in certain circumstances (i.e. progressive estimation from Chapter 7) can also be interpreted as a form of debiasing.

In the remainder of this section, we will generalize these aforementioned consistent formulations, discuss their applicability, review instances of them which have appeared previously in the literature, and introduce a fourth consistent formulation which is more broadly applicable to a wider range of non-linear applications. In doing so, we can relate all consistent techniques for primary estimation using two generalized primary estimators depending on if n is discrete or continuous. Later in Sec. 8.2, we will discuss how to choose sampling distributions for primary estimation to verify finite or infinite work-normalized variance.

In Sec. 8.1.1, we take the Maclaurin series used in Chapter 6 and generalize it into the Taylor series expansion for general non-linear applications. Afterwards in Sec. 8.1.2, we briefly review the IVP used to derive the Volterra form for classical exponential transmittance (6.19) and discuss how it generalizes. In Sec. 8.1.3, we derive another consistent formulation based on

¹: While this is generally true, there can arise situations where the actual limit of Eq. (8.2) is the result of a random process, meaning there may exist a finite j where $I(j) = I$. The problem of non-bounding extinctions with clamped media, which we reviewed in Chapter 7, is an example of this.

a telescoping series derived from a general IVP, and we review its previous use outside of the graphics literature. In Sec. 8.2.1, we introduce two general formulations for primary estimation which cover all consistent formulations introduced within this dissertation. We conclude this section in Sec. 8.1.5 with a review of how the prior interpretation of progressive estimation can be directly derived from our telescoping series formulation.

8.1.1 Taylor series formulation

In situations where the non-linear function g is infinitely differentiable and has a convergent Taylor series around some expansion point α , we can apply a Taylor series expansion to Eq. (8.1),

$$I = g(F) := \underbrace{\sum_{k=0}^{j-1} \Delta_k^{\text{Tay}}}_{I(j)} + \underbrace{\sum_{k=j}^{\infty} \Delta_k^{\text{Tay}}}_{B(j)} \quad (8.5)$$

$$\Delta_k^{\text{Tay}} := g^{(k)}(\alpha) \frac{(F - \alpha)^k}{k!},$$

to arrive at a generalization of the Maclaurin series we previously employed in Chapter 6. We denote the analytic k th derivative of g evaluated at α as $g^{(k)}(\alpha)$.

For any fixed bias parameter $j \geq 0$, Eq. (8.5) decomposes I into a biased formulation ($I(j)$) taking the form of a prefix-sum of the first j terms while the remaining terms constitute the bias ($B(j)$). For example, if $j = 1$, then $I(j) = g(\alpha)$, which will generally be a biased evaluation of I unless we happen to choose an expansion point such that $\alpha = F$. We provide a visualization of how the infinite sum corrects for the bias of the initial prefix-sum in Fig. 8.2.

The Taylor series expansion effectively separates the evaluation of g from the eventual estimation of F , thus removing the problematic non-linearities. While $(F - \alpha)^j$ is still a non-linear operation, we can estimate it in an

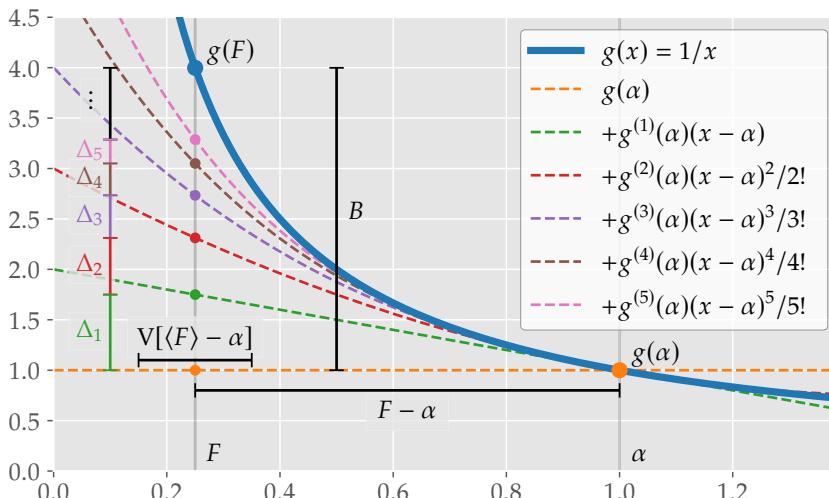


Figure 8.2: Equation (8.5) enables the unbiased estimation of $g(F)$ using only unbiased estimates of F and the analytic derivatives of g . Starting with $I(1) = g(\alpha)$, Eq. (8.5) progressively adds terms (vertical colored bars labeled Δ_j on the left) from the Taylor series expansion of $g(x)$ (dashed curves), where each is evaluated at a stochastic distance $\langle F \rangle - \alpha$.

unbiased manner by treating it as the product of j independent evaluations of $(F - \alpha)$ as reviewed previously in Sec. 6.3.

Applicability. The Taylor series as a consistent formulation is applicable to any non-linear function of expectations having a convergent Taylor expansion. However, there are many functions with convergent Taylor expansions that are *only* convergent for a specific set of expansion points (α). For one-dimensional problems, this set of expansion points is typically referred to as the *interval of convergence*.

Special care has to be taken when applying variance reduction techniques (see Appendix A.2) or blindly choosing α such that the Taylor series remains convergent. This can be problematic in some instances (see the problem of reciprocal estimation below) since the interval of convergence can be a function of F , which is often not known and has to be estimated using Monte Carlo.

Prior work. We were the first to introduce the Taylor series formulation as a general debiasing technique to the graphics literature [112]. However, formulations like Eq. (8.5) have appeared previously for problems like exponential transmittance and reciprocal estimation via other mathematical manipulations (i.e. the Maclaurin series expansion and control variates). While we have previously reviewed the case of exponential transmittance in Chapter 6, we will now review the problem of reciprocal estimation.

[112]: Misso et al. (2022), ‘Unbiased and consistent rendering using biased estimators’

Reciprocal estimation. Within various applications in light transport, we occasionally need to estimate the reciprocal of an integral defined as,

$$g(F) := \frac{1}{F}. \quad (8.6)$$

For instance, this happens in both photon mapping [95] and differentiable rendering [134] where Eq. (8.6) is the normalization constant of a Monte Carlo sampling PDF that is not available in closed form, so it must be estimated numerically. Applying Eq. (8.5) to Eq. (8.6), choosing $j = 1$, and performing some elementary simplifications results in,

$$g(F) := \frac{1}{\alpha} + \sum_{k=1}^{\infty} \Delta_k^{\text{recip}}, \quad \text{where } \Delta_k^{\text{recip}} := \alpha^{-k-1} (\alpha - F)^k, \quad (8.7)$$

[95]: Qin et al. (2015), ‘Unbiased Photon Gathering for Light Transport Simulation’

[134]: Bangaru et al. (2020), ‘Unbiased Warped-Area Sampling for Differentiable Rendering’

which can be readily estimated in an unbiased manner assuming that it is convergent. The interval of convergence for Eq. (8.7) is $\alpha \in (\frac{F}{2}, \infty)$ assuming that F and α are positive². While this implies that Eq. (8.7) cannot be employed for unbiased reciprocal estimation when no a priori information regarding F is available, progressive clamping (see Sec. 7.3.1) can always be employed as a consistent alternative.

2: More generally Eq. (8.7) is convergent when $|\alpha - F| < |\alpha|$.

Unbiased reciprocal estimation was first introduced to graphics by Qin et al. [95] where they imported the technique directly from the nuclear engineering literature [116]. Booth originally derived his formulation within

[116]: Booth (2007), ‘Unbiased Monte Carlo Estimation of the Reciprocal of an Integral’

the nuclear engineering literature by applying control variates to Eq. (8.6) and then taking a Maclaurin series expansion (Taylor series about $\alpha = 0$) through pattern matching with the well-known analytical solution of the geometric series.

Applying control variates first and then taking the Maclaurin series expansion is mathematically equivalent to just taking the Taylor series expansion where α is set to the control variate. Thus, our formulation (8.7) generalizes the control function Booth introduced in his derivation.

Qin et al. later introduced an estimator for the reciprocal (Eq. (8.7) with $\alpha = 1$) based on randomly evaluating Bernoulli trials which has been employed for specular manifold sampling [135] and path connections in refractive media [136]. Since both of these instances of unbiased reciprocal estimation involved evaluating the normalization constant of a PDF, it was always known that $F < 1$, allowing prior work to circumvent dealing with the reciprocal function's interval of convergence.

By first redefining Eq. (8.7) into a recursive formulation,

$$g(F, j) := \frac{1}{\alpha} + \frac{\alpha - F}{\alpha} g(F, j + 1); \quad g(F) := g(F, 0), \quad (8.8)$$

we can rederive a generalized version of the Bernoulli trials estimator for any given expansion point α as the prefix-sum estimator,

$$\langle g(F, j) \rangle_{\text{Bern}} := \begin{cases} W(j), & \text{otherwise} \\ W(j) + \langle g(F, j + 1) \rangle_{\text{Bern}}, & \text{with prob. } P_{\text{acc}}(j), \end{cases} \quad (8.9)$$

where the cumulative weight is defined as,

$$W(j) := W(j - 1) * \frac{\alpha - \langle F \rangle}{\alpha \cdot P_{\text{acc}}(j)}, \quad (8.10)$$

with Russian Roulette continuation probabilities,

$$P_{\text{acc}}(j) := \text{Min}(|W(j)|, 1). \quad (8.11)$$

The Russian Roulette probabilities are set proportionally to the current contribution which turns out to be identical to the choices we made for the PCume estimator in Eq. (6.77). When $\alpha = 1$, Eq. (8.9) simplifies to an integer counting process (assuming $\langle F \rangle \leq 1$), which is where the Bernoulli trial estimator received its name. In Fig. 8.3, we provide a comparison between our reciprocal PCume estimator (Eq. (8.9) with a location specific α), traditional Bernoulli trials for reciprocal estimation (Eq. (8.9) with $\alpha = 1$), and our telescoping series (see Sec. 8.1.3) where we progressively introduce bias to avoid divisions by zero.

Transmittance estimators as reciprocal estimators. While the similarities between Eq. (8.9) and the PCume estimator (6.78) imply that all exponential transmittance estimators can have analogous forms for reciprocal estimation, we have found in practice that the Bernoulli trials estimator performs the best when it is known that $\langle F \rangle \leq 1$. Taylor series based

[135]: Zeltner et al. (2020), ‘Specular Manifold Sampling for Rendering High-Frequency Caustics and Glints’

[136]: Pedireddla et al. (2020), ‘Path Tracing Estimators for Refractive Radiative Transfer’

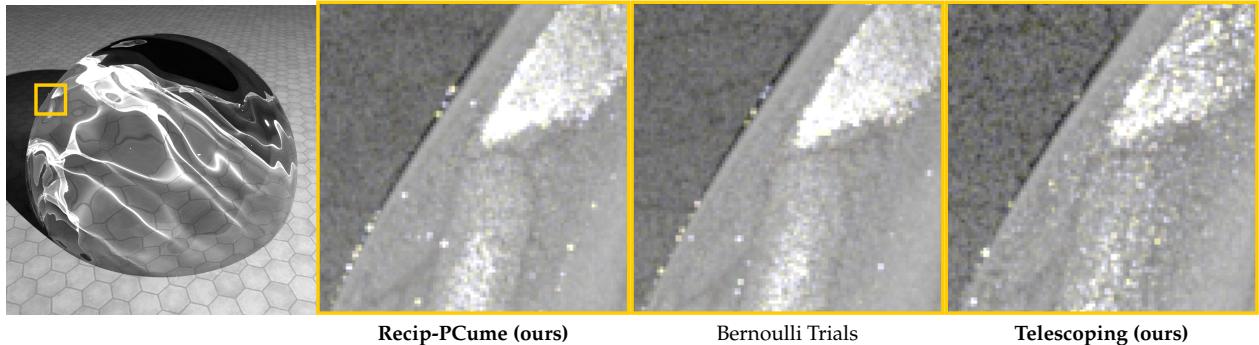


Figure 8.3: Preliminary equal-time (10 minutes) results showing a potential application in formulating unbiased estimation for specular manifold sampling [135] (i.e. reciprocal estimation). Our PCume inspired reciprocal estimator (left) performs similarly to Bernoulli trials. Making our telescoping-series estimator (right) competitive requires further investigation.

raymarching (see Appendix A) and telescoping series based raymarching (see Sec. 8.5.2) can perform significantly better, however, they cannot be employed unbiasedly in situations where it is possible for $\langle F \rangle \leq \frac{F}{2}$ and $\langle F \rangle = 0$. We believe that these techniques can instead be implemented *progressively*, however, we leave that as an avenue for future work.

Outside of graphics. Blanchet et al. [137] introduced the idea of employing the Taylor series as a generalized means of formulating unbiased estimators for functions of expectations outside of computer graphics. Dauchet et al. [138] mentions this idea as well in the context of handling non-linearities with Monte Carlo. Outside of these instances, the Taylor series expansion has been employed throughout statistics to reduce bias of general estimators of differentiable functions [139, 140] and is generally used to form biased approximations of functions where only $I(j)$ is evaluated.

[137]: Blanchet et al. (2015), ‘Unbiased Monte Carlo Computation of Smooth Functions of Expectations via Taylor Expansions’

[138]: Dauchet et al. (2018), ‘Addressing Nonlinearities in Monte Carlo’

[139]: Withers (1987), ‘Bias Reduction by Taylor Series’

[140]: Jiao et al. (2020), ‘Bias Correction with Jackknife, Bootstrap, and Taylor Series’

8.1.2 IVP formulation

When $g(F)$ is differentiable with respect to a continuous bias parameter, j , we can reformulate $g(F)$ as an initial value problem (IVP),

$$I := g(F) = I(j) + \int_j^\infty \frac{d}{dx} I(x) dx, \quad (8.12)$$

to compute the desired quantity by integrating over how it *changes*. To our knowledge, the only direct application of Eq. (8.12) that has been explored within rendering for non-linear functions is classical transmittance estimation, which we previously reviewed in Chapter 6. We extend this application to consider non-classical transmittances in Appendix B. Unfortunately, for the case of the generalized IVP, its derivatives can introduce additional non-linear terms which then have to be estimated in an unbiased manner (see Appendix B).

8.1.3 Telescoping series formulation

The previous two formulations require us to be able to compute derivatives of the non-linear function g with respect to different quantities. Unfortunately, this is not always conveniently computable since non-linear functions of expectations are not always simple "functions." They can instead represent the application of entire non-linear processes. For example, differentiation, stochastic root-finding, or the process of rendering an entire scene.

For formulating consistent algorithms of non-linear processes where derivatives are not readily available, we instead propose another consistent formulation. Starting from Eq. (8.12), we convert the integrated bias parameter, x , into a discrete parameter,

$$I = I(j) + \sum_{k=j}^{\infty} \int_k^{k+1} \frac{d}{dx} I(x) dx, \quad (8.13)$$

by first taking the sum of unit length integrals. By replacing the integral over the derivative with its antiderivative, we arrive at,

$$\begin{aligned} I &:= I(j) + \sum_{k=j}^{\infty} \Delta_k^{\text{tele}} \\ \Delta_k^{\text{tele}} &:= I(k+1) - I(k), \end{aligned} \quad (8.14)$$

which is an infinite telescoping summation that is readily estimatable by Monte Carlo integration³.

Outside of graphics. Within the statistical literature, McLeish [141] and Rhee et al. [142] independently introduced the first estimation schemes for Eq. (8.14). Rhee et al. focused on the specific problem of creating unbiased estimators for stochastic differential equations (SDEs) through adding randomized truncation (akin to Russian Roulette) to multi-level Monte Carlo⁴ [48, 143, 144], while McLeish [141] introduced a more generalized prefix-sum estimator.

Rhee [145] analyzed the optimal design decisions in deriving estimators for Eq. (8.14) and later Blanchet et al. [137] equated this with the optimal design decisions for the Taylor expansion approach in Eq. (8.5). Rhee et al. [146] later used these findings to introduce practical estimators for SDEs while Blanchet et al. [137] introduced a variance reduction technique to improve the convergence rate of a certain class of estimators. Their variance reduction technique is similar to the symmetric means variance reduction technique later introduced by Kettunen et al. [121] for exponential transmittance (see Appendix A.4). More recently, Moka et al. [147] has looked into the special case of formulating non-negative estimators for the reciprocal.

Infinite series rederivation. Any consistent formulation which is the result of an infinite summation can be derived as an instance of Eq. (8.14)

3: Note, that any finite prefix summation up to index k simplifies to $I(k)$ in Eq. (8.14).

[141]: McLeish (2011), 'A General Method for Debiasing a Monte Carlo Estimator'

[142]: Rhee et al. (2012), 'A New Approach to Unbiased Estimation for SDE's'

4: We discussed multi-level Monte Carlo (MLMC) briefly in Sec. 3.3.6.

[48]: Giles (2008), 'Multilevel Monte Carlo Path Simulation'

[143]: Heinrich (1998), 'A Multilevel Version of the Method of Dependent Tests'

[144]: Keller (2001), 'Hierarchical Monte Carlo Image Synthesis'

[145]: Rhee (2013), 'Unbiased Estimation with Biased Samplers'

[137]: Blanchet et al. (2015), 'Unbiased Monte Carlo Computation of Smooth Functions of Expectations via Taylor Expansions'

[146]: Rhee et al. (2015), 'Unbiased Estimation with Square Root Convergence for SDE Models'

[121]: Kettunen et al. (2021), 'An Unbiased Ray-Marching Transmittance Estimator'

[147]: Moka et al. (2019), 'Unbiased Estimation of the Reciprocal Mean for Non-Negative Random Variables'

by setting $I(j)$ to be the prefix-sum of the first j terms. For example, by plugging in the definition of $I(j)$ from the Taylor series (8.5) into Eq. (8.14),

$$\begin{aligned} I &= I(j) + \sum_{k=j}^{\infty} I(k+1) - I(k) \\ &= \left(\sum_{q=0}^j \Delta_q^{\text{Tay}} \right) + \sum_{k=j+1}^{\infty} \left(\left(\sum_{q=0}^{k+1} \Delta_q^{\text{Tay}} \right) - \left(\sum_{q=0}^k \Delta_q^{\text{Tay}} \right) \right) \\ &= \left(\sum_{q=0}^j \Delta_q^{\text{Tay}} \right) + \sum_{k=j+1}^{\infty} \Delta_k^{\text{Tay}}, \end{aligned} \quad (8.15)$$

we arrive at the Taylor series expansion.

Prior work. *Path tracing.* We can extend the idea that any infinite summation can be reformulated as an instance of the telescoping series formulation (8.14) by applying it to path tracing with NEE. Path tracing with NEE computes an unbiased estimate of an image by accumulating the contribution from all length paths. If we define $I(j+1) - I(j)$ to represent the difference between $(j+1)$ -length transport and j -length transport, simplifying to the contribution from the $(j+1)$ bounce (see Fig. 8.4), we can interpret path tracing as an instance of a telescoping sum Eq. (8.14). From this perspective, path tracing then compensates for the bias of direct illumination by evaluating each additional path length up to infinity.

Reciprocal estimation. Bangaru et al. [134] introduced an unbiased differentiable rendering algorithm which employed the divergence theorem to handle discontinuous integrands. Their formulations introduced a discontinuous warp field which they make continuous through convolution with a smoothing kernel whose normalization factor is not analytic and must be estimated. This amounts to formulating unbiased estimates for the reciprocal (8.6) of a random variable for which either Eq. (8.5) or Eq. (8.14) can be used. They employed the estimator proposed by McLeish [141] which is an instance of Eq. (8.14).

[134]: Bangaru et al. (2020), ‘Unbiased Warped-Area Sampling for Differentiable Rendering’

[141]: McLeish (2011), ‘A General Method for Debiasing a Monte Carlo Estimator’

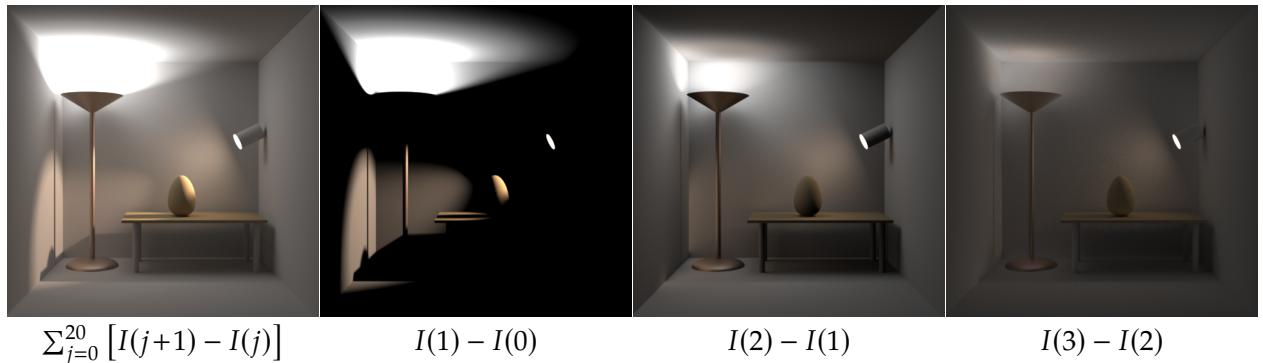


Figure 8.4: Illustration of path tracing as an instance of Eq. (8.14). Each term in the summation is the difference in contribution between $(j+1)$ - and j -length paths, which simplifies to a sum over the contribution of individual bounces.

8.1.4 Generalized consistent formulations

Originally, we defined general consistent formulations to be ones which were guaranteed to return the true expected solution in some limit (8.2). Based on this paradigm, we have introduced three different practical consistent formulations, namely the Taylor series (8.5), telescoping series (8.14), and the IVP (8.12). From our analysis, we observe that consistent formulations in practice take one of two forms depending on whether the variable dictating the bias is a discrete or continuous variable. For the case of a discrete variable, we can generalize all previous consistent formulations as the infinite summation,

$$I := I(j) + \sum_{k=j}^{\infty} \Delta_k, \quad (8.16)$$

where each formulation differs by their specific construction of Δ_k . By construction, we assume the bias $B(j) = \sum_{k=j}^{\infty} \Delta_k$ is always finite and $\Delta_{\infty} = 0$.

In the case of a continuous variable, λ , a general consistent formulation takes the form of an integral,

$$I := I(j) + \int_j^{\infty} \Delta_{\lambda} d\lambda. \quad (8.17)$$

Outside of classical transmittance estimation, we found the application of Eq. (8.17) to generally perform worse than Eq. (8.16) due to the derivatives in the continuous case having the potential for unbounded variance. For this reason, we will only focus on the discrete formulation in the remainder of this dissertation. However, all of the concepts which we cover will still generalize analogously to the continuous case.

8.1.5 Progressive formulation

We can derive progressive formulations from our generalized discrete formulation (8.16) by constructing a secondary summation,

$$\begin{aligned} I &:= \frac{1}{N} \sum_{j=0}^N \left(I(j) + \sum_{k=j}^{\infty} \Delta_k \right) \\ &= \frac{1}{N} \sum_{j=0}^N I(j) + \sum_{j=0}^N \sum_{k=j}^{\infty} \frac{\Delta_k}{N}. \end{aligned} \quad (8.18)$$

where the initial bias parameter (j) is initialized to the secondary index. From here it is easy to observe that since the summation over all Δ_k 's is finite by construction ($\sum_{k=j}^{\infty} \Delta_k = B(j)$), the right (i.e. bias) term in Eq. (8.18),

$$\lim_{N \rightarrow \infty} \sum_{j=0}^N \sum_{k=j}^{\infty} \frac{\Delta_k}{N} = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{j=0}^N \sum_{k=j}^{\infty} \Delta_k = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{j=0}^N a_j = 0, \quad (8.19)$$

converges to zero. Choosing to never evaluate the bias yields a biased but still consistent progressive formulation,

$$I = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{j=0}^N I(j). \quad (8.20)$$

Reformulating a given non-linear function of expectations as either an instance of Eq. (8.16) or Eq. (8.17) completes step ① in our general recipe (see Fig. 8.1) for formulating consistent solutions for non-linear functions of expectations. Additionally we can choose to employ a progressive formulation (8.20) for biased, but still consistent, solutions.

8.2 Verifying finite variance

After constructing a consistent formulation which is amenable to unbiased estimation, we have to decide on how to go about estimating the desired quantity. Unfortunately, as we have seen in Chapter 7, naively formulating an estimator can result in impractical solutions which may never converge in any reasonable amount of time. From our practical investigations we have discovered two classifications of problems which require slightly different processes for formulating useful solutions. These two classifications differ by whether the *primary estimator* has finite or infinite work-normalized variance⁵. The purpose of this section is to discuss a generalized process for verifying whether a given consistent formulation will have finite or infinite work-normalized variance, which is step ② in our recipe.

The remainder of this section is organized as follows: First, we formulate primary and secondary estimators for Eq. (8.16) in Sec. 8.2.1. Based on prior work, we then introduce the optimal probability distributions for minimizing the work-normalized variance of a primary estimator in Sec. 8.2.2. Unfortunately, computing these idealized probability distributions typically requires already knowing the solution, I , ahead of time. Thus, in Sec. 8.2.3 we discuss a pragmatic approach where we instead derive probability distributions which minimize variance rather than work-normalized variance. We then use the variance-optimal probability distributions to determine if finite work-normalized variance is possible in Sec. 8.2.4.

5: Estimators for Eq. (8.16) and Eq. (8.17) will typically always be primary estimators while estimators for Eq. (8.20) will typically always be secondary estimators.

8.2.1 General consistent estimation

Primary estimation. We generally consider two different constructions when devising primary estimators for Eq. (8.16). The first primary estimator is the single-term estimator⁶,

$$\langle I \rangle_1^j := \langle I(j) \rangle + \frac{\langle \Delta_k \rangle}{p(k)}, \quad (8.21)$$

which involves importance sampling a single-term, k , from a given probability distribution $p(k)$.

6: The Taylor series formulation of ratio tracking (6.75) is an example of a single-term estimator.

The second primary estimator that we consider is the prefix-sum estimator⁷,

$$\langle I \rangle_1^j := \langle I(j) \rangle + \sum_{k=j}^i \frac{\langle \Delta_k \rangle}{P(i \geq k)}, \quad (8.22)$$

where $P(i \geq k)$ is the CMF representing the probability of sampling $i \geq k$. When employing prefix-sum estimators in practice, it is typically more convenient to specify $P(i \geq k)$ indirectly through the use of Russian Roulette probabilities.

While we do not discuss the continuous case, analogous estimators could easily be devised for Eq. (8.17) where $p(k)$ becomes a PDF and the prefix-sum in Eq. (8.22) becomes a prefix-integral.

7: Prior examples of prefix-sum estimators include next-flight (6.15), PCume (6.78), and PCMF (6.80).

Secondary estimation. We define a naive secondary estimator as,

$$\langle I \rangle_n^j := \frac{1}{N} \sum_{k=0}^N \langle I \rangle_1^j. \quad (8.23)$$

and a progressive secondary estimator as,

$$\langle I \rangle_n^j := \frac{1}{N} \sum_{k=0}^N \langle I \rangle_1^{j+k}. \quad (8.24)$$

These two secondary estimators take the average of n primary estimators ($\langle I \rangle_1^j$), and they only differ by how they choose to specify the starting parameter (j) for each primary invocation. A naive secondary estimator will always use the same starting parameter while a progressive estimator will progressively modify it. While Eq. (8.24) is written such that the bias parameter is always incremented, it can instead be *adaptively* changed in a similar manner to our progressive extinctions technique (see Sec. 7.3.3). For simplicity of notation, we will assume the bias parameter is always incremented after each primary invocation unless specified otherwise.

8.2.2 Optimal probability distributions

Ideally, we want to formulate estimators which have low work-normalized variance (WNV),

$$\text{WNV}\left[\langle I \rangle_1^j\right] := V\left[\langle I \rangle_1^j\right] C\left[\langle I \rangle_1^j\right], \quad (8.25)$$

where $V\left[\langle I \rangle_1^j\right]$ is the variance of the primary estimator and $C\left[\langle I \rangle_1^j\right]$ is its expected cost, or the expected work required to run the estimator. The ideal estimator constructs $\langle \Delta_j \rangle$ and chooses $p(k)$ to minimize Eq. (8.25),

$$\arg \min_{p(k)} V\left[\langle I \rangle_1^j\right] C\left[\langle I \rangle_1^j\right]. \quad (8.26)$$

Rhee [145] has previously analyzed this minimization problem for both single-term and various prefix-sum formulations⁸. For brevity, we will only review their single-term analysis since single-term estimators are simpler to construct, easier to verify and, in practice, we typically do not directly specify the CMF or CDF ($P(i \geq k)$) and instead employ Russian Roulette. We refer the interested reader to Rhee's dissertation for an extended analysis on the prefix-sum case.

Plugging Eq. (8.21) into Eq. (8.26) results in the minimization problem,

$$\arg \min_{p(k)} \left(\sum_{k=j}^{\infty} \frac{(\mathbb{E}[\langle \Delta_k \rangle])^2}{p(k)} - (I - I(j))^2 \right) \left(\sum_{k=j}^{\infty} C[\langle \Delta_k \rangle] p(k) \right), \quad (8.27)$$

where $C[\langle \Delta_k \rangle]$ is the expected cost of estimating the term $\langle \Delta_k \rangle$, and we assume the cost and variance of evaluating $\langle I(j) \rangle$ in Eq. (8.21) is negligible. Note, we are assuming that Eq. (8.16) is an unbiased formulation so $E[\langle I \rangle_1^j] = I$. Thus, the estimator's variance and mean-squared error are equivalent ($V[\langle I \rangle_1^j] = MSE[\langle I \rangle_1^j]$). Assuming that $C[\langle \Delta_k \rangle]$ is a positive, non-decreasing sequence and $V[\langle \Delta_k \rangle] > 0$ for all finite k , Rhee [145] proved that the PMF,

$$p(k) = \sqrt{\frac{(\mathbb{E}[\langle \Delta_k \rangle])^2}{(I - I(j))^2 + \alpha C[\langle \Delta_k \rangle]}} \quad (8.28)$$

where,

$$\alpha := \frac{V[\langle I \rangle_1^j]}{C[\langle I \rangle_1^j]}, \quad (8.29)$$

successfully minimizes the work-normalized variance (8.25) of the single-term estimator (8.21), thus solving Eq. (8.27). Unfortunately, this analysis is only of theoretical interest, since practically deriving $p(k)$ would require knowing I , $C[\langle I \rangle_1^j]$, and $V[\langle I \rangle_1^j]$ in closed form which we do not have access to in practical applications.

8.2.3 Variance-optimal probability distributions

Instead, we propose a pragmatic two-step approach which first involves deriving a variance-optimal $p(k)$ that just minimizes the variance,

$$\arg \min_{p(k)} V[\langle I \rangle_1^j]. \quad (8.30)$$

Afterwards, we insert our derived $p(k)$ into Eq. (8.25) to verify that $WNV[\langle I \rangle_1^j]$ is finite and $p(k)$ is *normalizable*. We will now derive the variance-optimal PMF for the discrete case, and we include the derivations for the continuous case afterwards.

[145]: Rhee (2013), 'Unbiased Estimation with Biased Samplers'

8: Rhee also discusses multi-term estimation which we tend to not employ in practice.

Discrete variance-optimal single-term PDF. Since we construct a single-term primary estimator (8.21) to be unbiased, minimizing the estimator's variance is equivalent to minimizing its second moment,

$$\mathbb{E} \left[\left(\frac{\langle \Delta_k \rangle}{p(k)} \right)^2 \right] = \sum_{k=j}^{\infty} \frac{\mathbb{E}[\langle \Delta_k \rangle^2]}{p(k)}, \quad (8.31)$$

under the condition that $p(j)$ is normalized. To find a valid PMF which minimizes Eq. (8.31) we employ Lagrange multipliers. We start by defining a new minimization problem,

$$\mathcal{L}(\lambda, p) := \sum_{k=j}^{\infty} \frac{\mathbb{E}[\langle \Delta_k \rangle^2]}{p(k)} - \lambda \left(1 - \sum_{k=j}^{\infty} p(k) \right), \quad (8.32)$$

and solve for the roots of its gradients to find its local extrema where the sum in the parentheses encodes the PMF normalization condition. Solving for $p(j)$ results in,

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial p(k)} = 0 &\Leftrightarrow -\frac{\mathbb{E}[\langle \Delta_k \rangle^2]}{p(k)^2} + \lambda = 0 \\ \Leftrightarrow p(k) = \sqrt{\frac{\mathbb{E}[\langle \Delta_k \rangle^2]}{\lambda}} &\Leftrightarrow p(k) \propto \sqrt{\mathbb{V}[\langle \Delta_k \rangle] + (\Delta_k)^2}, \end{aligned} \quad (8.33)$$

where λ is a normalization constant. Thus, we arrive at an optimal PMF.

Variance-optimal PMF outside of graphics. Rhee [145] derived a variance-optimal PMF for the discrete telescoping series (8.14) which matches Eq. (8.33). Later, Blanchet et al. [137] showed that Eq. (8.33) was also the variance-optimal PMF for the general Taylor series formulation (8.5).

[145]: Rhee (2013), ‘Unbiased Estimation with Biased Samplers’

[137]: Blanchet et al. (2015), ‘Unbiased Monte Carlo Computation of Smooth Functions of Expectations via Taylor Expansions’

Continuous variance-optimal single-term PDF. For the continuous case, the second moment takes the form of,

$$\mathbb{E} \left[\left(\frac{\langle \Delta_k \rangle}{p(k)} \right)^2 \right] = \int_k^{\infty} \frac{\mathbb{E}[\langle \Delta_x \rangle^2]}{p(x)^2} p(x) dx = \int_k^{\infty} \frac{\mathbb{E}[\langle \Delta_x \rangle^2]}{p(x)} dx, \quad (8.34)$$

under the same condition that $p(x)$ is normalized. We compute the variance-optimal PDF by employing the Euler-Lagrange equation from the calculus of variations by first defining,

$$\mathcal{L}(x, \lambda, p) := \frac{\mathbb{E}[\langle \Delta_x \rangle^2]}{p(x)} + \lambda p(x), \quad (8.35)$$

where λ is the unknown Lagrange multiplier. We then write the Euler-Lagrange equation for q and then solve for p ,

$$\frac{d\mathcal{L}}{dp} - \frac{d}{dx} \left(\underbrace{\frac{d\mathcal{L}}{dp'}}_{=0} \right) = 0 \Leftrightarrow -\frac{E[\langle \Delta_x \rangle^2]}{p^2(x)} + \lambda = 0 \quad (8.36)$$

$$\Leftrightarrow p(x) = \sqrt{\frac{E[\langle \Delta_x \rangle^2]}{\lambda}}. \quad (8.37)$$

Since the PDF must be normalized, λ is its normalization constant. Finally, since $V[\langle \Delta_x \rangle] = E[\langle \Delta_x \rangle^2] - (\Delta_x)^2$, the variance of the primary estimator with a continuous bias parameter is minimized when,

$$p(x) \propto \sqrt{V[\langle \Delta_x \rangle] + (\Delta_x)^2}. \quad (8.38)$$

8.2.4 Verifying finite work-normalized variance

As we saw in Eq. (8.33), the variance-optimal PMF for estimating a single-term estimator with a discrete bias parameter (8.21) is,

$$p(k) \propto \sqrt{V[\langle \Delta_k \rangle] + (\Delta_k)^2}. \quad (8.39)$$

While we might not know $V[\langle \Delta_k \rangle]$ or Δ_k in closed form, we can often derive their *asymptotic rates* to instead derive an *asymptotically variance-optimal* $p(k)$. To verify finite work-normalized variance, we first insert the asymptotic rates for $V[\langle \Delta_k \rangle]$ and Δ_k into Eq. (8.39) to compute $p(k)$. We then insert $p(k)$ and the asymptotic rates for $V[\langle \Delta_k \rangle]$ and $C[\langle \Delta_k \rangle]$ into Eq. (8.25) and finally simplify the resulting expression. If the result is finite and Eq. (8.39) is a normalizable distribution, then we have proven finite work-normalized variance so we move onto formulating practical estimators in Sec. 8.3. Otherwise, we move onto formulating useful infinite work-normalized variance estimators in Sec. 8.4. Thus, we have completed step ②.

8.3 Handling finite work-normalized variance

This section reviews steps ③, ④ and ⑤ from Fig. 8.1.

Since our primary estimators (Eqs. (8.21) and (8.22)) with finite work-normalized variance are designed to be unbiased, the central limit theorem tells us that the variance of any naive secondary estimator (8.23) will converge at an asymptotic rate of $V[\langle I \rangle_n^j] = \mathcal{O}(n^{-1})$ while its error will converge at a rate of $e[\langle I \rangle_n^j] = \mathcal{O}(n^{-1/2})$ [26]. Unfortunately, due to the construction of Δ_k , the specific choice of $p(k)$, and the stochastic nature of estimating $\langle \Delta_k \rangle$, the constant factor embedded within the asymptotic convergence rate can vary wildly. As we observed in Chapter 7 when analyzing significantly non-bounding extinctions in classical volumetric

[26]: Caflisch (1998), ‘Monte Carlo and quasi-Monte Carlo methods’

rendering, the constant factor can even cause a technically finite work-normalized variance estimator to exhibit numerically infinite variance. Meaning, just because we have verified that a certain primary estimator will have finite work-normalized variance does not mean that we have generally derived a practically useful estimator. Certain optimizations or design decisions might have to be made, or progressive (8.24) alternatives may need to be considered before we arrive at an efficient solution.

While we have previously analyzed the problem of finite work-normalized variance estimation in Chapter 6 and Chapter 7, we will briefly summarize some general recommendations for formulating efficient finite work-normalized variance estimators in the remainder of this section.

Take advantage of correlations. Correlations should always be taken advantage of whenever they can be applied to improve the performance of an estimator without directly biasing the result. Key examples of correlations being employed in practice are the variance reduction techniques proposed by Kettunen et al. [121] (see Appendix A), reusing prefix-products across summations, and sharing samples across telescoping invocations (see Sec. 8.5.2).

[121]: Kettunen et al. (2021), ‘An Unbiased Ray-Marching Transmittance Estimator’

Single-term versus prefix-sum. Single-term estimators should be considered in situations where the underlying probability distribution ($p(k)$) matches the contribution (Δ_k) well, is known, and can be directly sampled. In such cases, prefix-sum estimators can lead to lower variance estimates (PCume, PCMF), but are typically more costly than single-term estimators since they accumulate over the entire CMF instead of sampling from $p(k)$.

When $p(k)$ is not known beforehand and arises as a result of some Monte Carlo termination process (i.e. naive path tracing hitting an emissive light source), then prefix-sum estimation (i.e. path tracing with NEE) is usually more efficient.

Control variates for importance sampling. In many cases, knowing the asymptotically variance-optimal $p(k)$ may require knowing Δ_k beforehand. Unfortunately, Δ_k is a quantity which generally has to be estimated, so we often have to make a *conservative estimate* of the asymptotic rate of Δ_k to derive an approximately optimal $p(k) \approx p^*(k)$ through the application of *control variates*.

For example, in the case of classical transmittance estimation (6.68), Δ_k directly depends on the real optical depth which is not known at render time. To introduce a $p^*(k)$ which is amenable to importance sampling, we introduced the concept of a μ_t -dictated free-flight distribution through the application of control variates. When efficiently incorporated⁹, the combined free-flight distribution is an asymptotically conservative upper bound on the real $p(k)$.

9: By efficiently incorporated, we mean that the combined extinction is chosen such that it is always bounding.

While control variates can be directly applied to the estimator (8.16), it is typically more convenient in practice to apply control variates to the

non-linear function before converting to the desired consistent formulation for estimation¹⁰.

10: For an example, see Eq. (6.20).

Progressive solutions for stochastic situations. When we are unable to robustly devise a conservative $p^*(k)$ for a given estimator, naively employing a user-defined $p^*(k)$ can result in impractical solutions. In the worst case, an excessively poor $p^*(k)$ can even result in an estimator with numerically infinite variance (see Chapter 7). In such situations, we can instead choose to employ a progressive secondary estimator (8.1.5) to trade initial bias for avoiding excessive variance from poor importance sampling. These progressive secondary estimators are designed to adaptively modify $p^*(k)$ after every primary invocation until an efficient and conservative one is discovered.

We have previously analyzed an instance of this idea in Chapter 7 when we devised progressive solutions for rendering classical participating media in the presence of a black box real extinctions. These techniques adapted the free-flight (and emission) sampling distributions over time through changing μ_t , and clamping the null extinction. By doing so, we effectively guaranteed that $p^*(k)$ never asymptotically approaches 0 faster than Δ_k .

While we only reviewed instances where we were guaranteed to discover a good $p^*(k)$ in finite time, we will consider the infinite time case in Sec. 8.4.

8.4 Handling infinite work-normalized variance

The construction of a consistent formulation as one which returns the true solution in some limit (8.2) guarantees that any naive unbiased secondary estimator (8.23) will converge in probability to the true solution in the limit of secondary samples (in terms of error, not variance). This remains true by the law of large numbers (see Sec. 3.2.4) even if a primary estimator has infinite work-normalized variance.

However, the error convergence rate of a naive secondary estimator (8.23) whose primary estimates have infinite work-normalized variance may be *slower* than $\mathcal{O}(n^{-1/2})$. We demonstrate this in Fig. 8.5 for the case of estimating the integral of a simple function with a controllable amount of variance. Even in cases where the primary estimator has infinite variance, a secondary estimator will converge in expectation, but at a slower rate.

This implies that by constructing primary estimators which have infinite variance, but *finite* cost, we can still devise solutions that can be employed for practical use. In the remainder of this section, we will discuss how we go about formulating unbiased estimators for such scenarios in Sec. 8.4.1 followed by a discussion on progressive estimation as an alternative in Sec. 8.4.3.

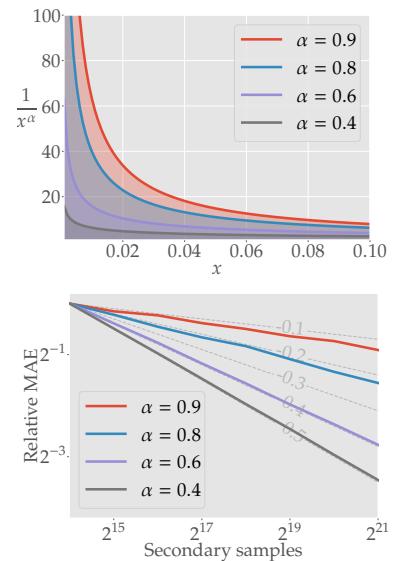


Figure 8.5: The integral $\int_0^1 \frac{1}{x^\alpha} dx$ (top) provides a useful test case for finite vs. infinite variance primary estimators. The integral takes on a closed-form value of $(1 - \alpha)^{-1}$ for any $\alpha \in (0, 1)$, but the variance of a primary estimator using uniform sampling for $p(x)$ is finite only for $\alpha < 0.5$. For $\alpha \geq 0.5$ the variance is infinite, leading to slower error convergence rates for naive secondary estimators (bottom).

8.4.1 Unbiased estimators under infinite variance

Unfortunately, selecting an optimal configuration for Δ_k and $p(k)$ in the case of primary estimation under infinite work-normalized variance is a challenging problem since Eq. (8.25) will *always* be infinite¹¹. Instead, we choose to take a more pragmatic approach of formulating finite cost, infinite variance estimators while tweaking the estimators' design decisions until performance is acceptable.

For many applications, $V[\langle \Delta_k \rangle]$ is inversely related with $C[\langle \Delta_k \rangle]$, so by increasing the rate at which the cost grows, we reduce the rate at which the variance increases¹². Doing so may allow us to achieve practical error convergence rates when employing naive secondary estimators whose primary estimators have infinite variance. Managing the trade-off between variance and cost is problem-dependent and may not always result in practical estimators. For applications where we can trade bias for higher efficiency, we can alternatively consider progressive secondary estimation (see Sec. 8.4.3).

8.4.2 Aside: Alleviating infinite variance through MIS

While we do not explicitly provide a practical example of this within this dissertation, in certain scenarios the problem of infinite work-normalized variance can be alleviated through the application of MIS. Consider two estimators,

$$\langle I \rangle_p := \frac{\Delta_k}{p(k)} \quad \langle I \rangle_q := \frac{\Delta_k}{q(k)}, \quad (8.40)$$

which estimate the same quantity ($\sum_{k=0}^{\infty} \Delta_k$) using two different probability distributions $p(k)$ and $q(k)$. Assume that employing $p(k)$ results in an infinite variance estimator while $q(k)$ results in a finite variance estimator. Since all Δ_k 's are held constant between the two estimators, we can assume that $p(k)$ results in infinite variance because it converges to 0 at a *faster* rate than Δ_k .

If we combine the two estimators using MIS with one sample each using the balance heuristic,

$$\begin{aligned} \langle I \rangle_{\text{MIS}} &:= \frac{\Delta_k}{p(k)} \left(\frac{p(k)}{p(k) + q(k)} \right) + \frac{\Delta_k}{q(k)} \left(\frac{q(k)}{p(k) + q(k)} \right) \\ &= \frac{2\Delta_k}{p(k) + q(k)}, \end{aligned} \quad (8.41)$$

we see that the estimator simplifies to having $p(k) + q(k)$ in the denominator. Since $\mathcal{O}(p(k)) + \mathcal{O}(q(k)) = \mathcal{O}(q(k))$ and $q(k)$ results in finite work-normalized variance, the MIS combination of both the finite and infinite variance estimators will have finite work-normalized variance.

Unfortunately, if we have reached step ⑥ then that means a $q(k)$ should not be readily available. However, this application of MIS may still have some applicability in combining photon mapping (infinite work-normalized

11: We assume that if we have reached this point, then we have already shown that the variance-optimal configuration has infinite work-normalized variance. Thus, we assume the work-normalized variance will *always* be infinite for any practical configuration.

12: As an example, consider photon mapping where tracing more photons will reduce the variance of $\langle \Delta_k \rangle$.

variance) with bidirectional path tracing (finite work-normalized variance) through the vertex connections and merging algorithm [91]. This will unfortunately be left to future work.

[91]: Georgiev et al. (2012), ‘Light Transport Simulation with Vertex Connection and Merging’

8.4.3 Progressive estimators under infinite variance

While we can devise unbiased infinite work-normalized variance estimators, the noise which arises from their slower convergence rates may not be acceptable for all applications. For these cases, we can instead formulate progressive estimators (8.24) which trade noise for bias by choosing to never evaluate Δ_k (see Sec. 8.1.5). Our consistent progressive formulation is a generalization of the one used by progressive photon mapping, so we will employ the analysis of prior work to construct efficient estimators.

Finite variance progressive construction. We can derive a progressive secondary estimator which is both consistent and has *finite* variance, even if the primary estimator has infinite variance, if we can reformulate the secondary estimator such that its MSE converges,

$$\lim_{n \rightarrow \infty} \text{MSE}[\langle I \rangle_n^0] = \lim_{N \rightarrow \infty} \left(V[\langle I \rangle_n^0] + B[\langle I \rangle_n^0]^2 \right) = 0. \quad (8.42)$$

For simplicity of notation, we assume the bias parameter is initialized to $j = 0$. To guarantee that the MSE converges, we need both the bias and the variance,

$$\begin{aligned} \lim_{n \rightarrow \infty} V[\langle I \rangle_n^0] &= \lim_{n \rightarrow \infty} \frac{1}{n^2} \sum_{k=1}^n V[\langle I \rangle_1^k] = \frac{\mathcal{O}(V[\langle I \rangle_1^n])}{n} = 0, \\ \lim_{n \rightarrow \infty} B[\langle I \rangle_n] &= \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{k=1}^n B[\langle I \rangle_1^k] = \mathcal{O}(B[\langle I \rangle_1^n]) = 0. \end{aligned} \quad (8.43)$$

to converge. This is achievable when the primary estimator’s bias vanishes (which is true by construction (8.20)) and the variance¹³ of $\langle I \rangle_1^k$ increases at most *sublinearly* with respect to k . Among all parameterizations for $\langle I \rangle_1^k$, the one that maximizes the MSE convergence rate is the one that achieves $\mathcal{O}(V[\langle I \rangle_1^n]) = \mathcal{O}(B[\langle I \rangle_1^n]^2)$.

13: The variance of $\langle I \rangle_1^k$ grows at the same rate as $V[\langle \Delta_k \rangle]$ for biased but consistent estimation (8.42).

Pragmatic approach. We choose to construct consistent progressive estimators by attempting to reparameterize Δ_k such that the primary estimator’s variance $V[\langle I \rangle_1^k]$ behaves strictly asymptotically sublinearly with respect to the index k . Meaning, $V[\langle I \rangle_1^k]$ is allowed to increase, but not faster than k itself ensuring that the variance of our progressive estimator vanishes.

Once we have constructed a consistent progressive estimator, we have completed the final step 7, and we can now apply our recipe to formulate consistent solutions for novel non-linear problems.

Unbiased progressive estimation. While we have found unbiased progressive estimation to be situationally useful for the case of finite work-normalized variance (see Sec. 7.4), we did not discover the same to be true for the infinite work-normalized variance case. Following the steps outlined previously for progressive estimation results in a finite variance secondary formulation (8.42), however, incorporating unbiased estimates for the bias back into the consistent estimator tends to ruin the beneficial properties of progressive estimation for this case. We will provide an example of this later when we devise unbiased progressive photon mapping (see Sec. 8.5.3).

8.5 Applications

In this section, we explicitly apply the recipe outlined in Fig. 8.1 to formulate consistent solutions to three separate problems in light transport. In Sec. 8.5.1, we introduce unbiased and consistent estimators for forward differentiation through debiasing the result of finite differences. In Sec. 8.5.2, we formulate an unbiased raymarching based transmittance estimator which can be applied to general non-classical transmittance functions. Finally we introduce an unbiased form of photon mapping and progressive photon mapping in Sec. 8.5.3.

8.5.1 Unbiased and progressive finite differencing

As an example of directly applying Fig. 8.1 to formulate a consistent estimator for a novel problem, consider computing the derivative of the integral $F(x) := \int_{\Omega} f(x, y) dy$ with respect to the parameter x :

$$I := \frac{\partial F(x)}{\partial x} = \frac{\partial}{\partial x} \int_{\Omega} f(x, y) dy. \quad (8.44)$$

For example, the function $f(x, y)$ could represent the contribution of a light path y in a scene parameterized by x . Note that the domain of integration (Ω) may depend on x .

Devising ways to estimate Eq. (8.44) in an efficient and accurate manner is a major goal of differentiable rendering [148, 149]. Although finite differencing is not a practical approach for estimating high-dimensional derivatives in a backward differentiation or inverse rendering context, it remains a common baseline for validating new differentiable rendering techniques thanks to their generality and ease of implementation. Unfortunately, since finite differences are biased, such "ground truth" baseline comparisons remain imperfect. By following all steps outlined in Fig. 8.1, we will derive both unbiased and progressively consistent alternatives to biased finite differences, with the hope that they can serve as more accurate ground-truth techniques for validating future differentiable rendering methods.

[148]: Kato et al. (2020), 'Differentiable Rendering: A Survey'

[149]: Nimier-David et al. (2019), 'Mitsuba 2'

We first denote the biased finite difference approximation of Eq. (8.44) as $I(j)$,

$$\frac{\partial F(x)}{\partial x} \approx I(j) := \frac{1}{h_j} \left(\int_{\Omega_1} f(x + h_j, y_1) dy_1 - \int_{\Omega_2} f(x, y_2) dy_2 \right), \quad (8.45)$$

where j maps to a finite difference step-size h_j , and $\Omega_1 \neq \Omega_2$ if the integration domain depends on x . By the definition of a derivative, this formulation satisfies the conditions for being considered a consistent formulation (8.2) as long as $h_j \rightarrow 0$ as $j \rightarrow \infty$. In our formulations, we choose to map j to the finite difference step-size such that $h_j \propto 2^{-j}$.

We can now formulate unbiased finite differencing using our telescoping series based single-term primary estimator (8.21) as,

$$\begin{aligned} I &:= I(j) + \sum_{k=j}^{\infty} \Delta_k^{\text{FD}}, \quad \text{where } \Delta_k^{\text{FD}} := I(k+1) - I(k) \\ \langle I \rangle &:= \langle I(j) \rangle + \frac{\langle \Delta_k^{\text{FD}} \rangle}{p(k)}, \quad \text{where } \langle \Delta_k^{\text{FD}} \rangle := \langle I(k+1) - I(k) \rangle. \end{aligned} \quad (8.46)$$

This completes step ①. Under this formulation, $\langle I(j) \rangle$ takes the form of an unbiased estimate of a single finite difference with step size $h_j \propto 2^{-j}$ while $\langle I(k+1) - I(k) \rangle$ is an unbiased estimate of the *difference* between two finite differences where $h_{k+1} = \frac{1}{2} \cdot h_k$. The efficiency of Eq. (8.46) is directly dependent on how well we can *correlate* the estimation of $\langle I(j) \rangle$, $\langle I(k) \rangle$, and $\langle I(k+1) \rangle$.

Unbiased estimation with finite variance. For the cases where $f(x, y)$ is finite, continuously differentiable with respect to x , and Ω does not depend on x , we can formulate an unbiased estimator with *finite work-normalized variance*.

In this ideal case, we can choose to perfectly correlate evaluations of $f(x, y)$ through the use of the same samples for y . For an example, consider the problem of differentiating with respect to the roughness parameter of a metallic surface (visualized in Fig. 8.6). Changing the roughness parameter will not alter the path space (Ω) of possible paths, but it will modify a given path's (y) contribution ($f(x, y)$). For these situations, we can importance sample a *single* path (y) while correlating all evaluations of $f(x, y)$ using that path, so, for brevity, we will simplify $f(x, y)$ to $f(x)$ in the following derivations.

Plugging Eq. (8.45) into the definition for $\langle \Delta_k^{\text{FD}} \rangle$ in Eq. (8.46) results in,

$$\langle \Delta_k^{\text{FD}} \rangle = \frac{\langle f(x + h_{k+1}) \rangle - \langle f(x) \rangle}{h_{k+1}} - \frac{\langle f(x + h_k) \rangle - \langle f(x) \rangle}{h_k}. \quad (8.47)$$

If $f(x)$ is continuously differentiable with respect to x , then

$$f(x + h) = f(x) + h \frac{\partial}{\partial x} f(x) + h^2 \frac{\partial^2}{\partial x^2} f(x) + \mathcal{O}(h^3). \quad (8.48)$$

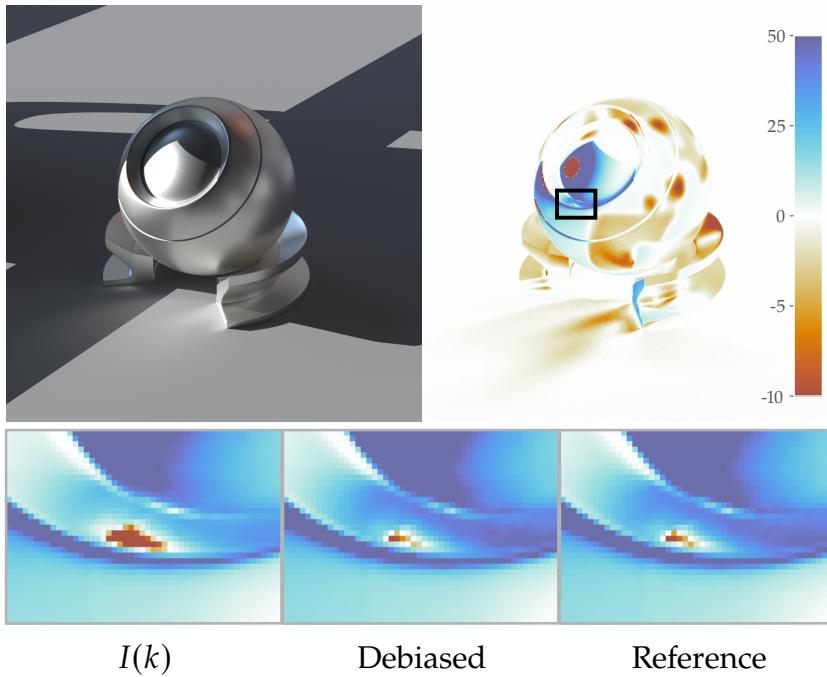


Figure 8.6: Debiased finite differences with respect to the roughness of a metallic object in a scene where the main light source is partially obstructed. Our method debiases the initial $I(k)(h = 0.005)$ to match the reference (right) which was rendered using ($h = 0.0001$).

Inserting Eq. (8.48) into Eq. (8.47) and canceling terms gives

$$\langle \Delta_k^{\text{FD}} \rangle = (h_{k+1} - h_k) \left\langle \frac{\partial^2}{\partial x^2} f(x) \right\rangle + \mathcal{O}(h_{k+1}^2) - \mathcal{O}(h_k^2), \quad (8.49)$$

and since $h_{k+1} < h_k$ by construction, we have

$$\Delta_k^{\text{FD}} \propto (h_{k+1} - h_k), \quad \text{and} \quad V[\langle \Delta_k^{\text{FD}} \rangle] \propto (h_{k+1} - h_k)^2. \quad (8.50)$$

Inserting Eq. (8.50) into Eq. (8.39), we arrive at the variance optimal $p(k) \propto 2^{-k}$, which we then insert into our single-term primary estimator (8.46), completing step ②. Since the cost of evaluating finite differences is constant, we have arrived at an unbiased estimator with finite work-normalized variance completing ③. We show a practical implementation of the surface roughness example from before in Fig. 8.6 where we choose to fully correlate paths by always performing BSDF importance sampling using the same BSDF and evaluate all integral estimates simultaneously.

Consistent estimation. If $f(x, y)$ is not continuous with respect to x (for example, when x impacts visibility), then the variance of the primary estimator can become infinite, so we skip to step ⑥. Through experimentation we found that unbiased finite difference estimators are too noisy in practice for the infinite variance case, so we continue to step ⑦ to formulate a consistent progressive estimator.

For the ideal case where all evaluations of $f(x, y)$ can be evaluated using the same samples for y (8.47), estimating $I(k)$ in Eq. (8.45) is equivalent to

estimating the true derivative convolved with a box kernel,

$$\langle I(k) \rangle = \frac{\langle f(x + h_k) - f(x) \rangle}{h_k} = \frac{1}{h_k} \int_x^{x+h_k} \langle \frac{\partial}{\partial x} f(t) \rangle dt \quad (8.51)$$

$$= \int_{-\infty}^{\infty} K(x-t) \langle \frac{\partial}{\partial x} f(t) \rangle dt, \quad \text{where} \quad (8.52)$$

$$K(\tau) = \begin{cases} \frac{1}{h_k} & 0 < \tau < h_k \\ 0 & \text{otherwise} \end{cases} \quad \text{is a normalized box kernel.} \quad (8.53)$$

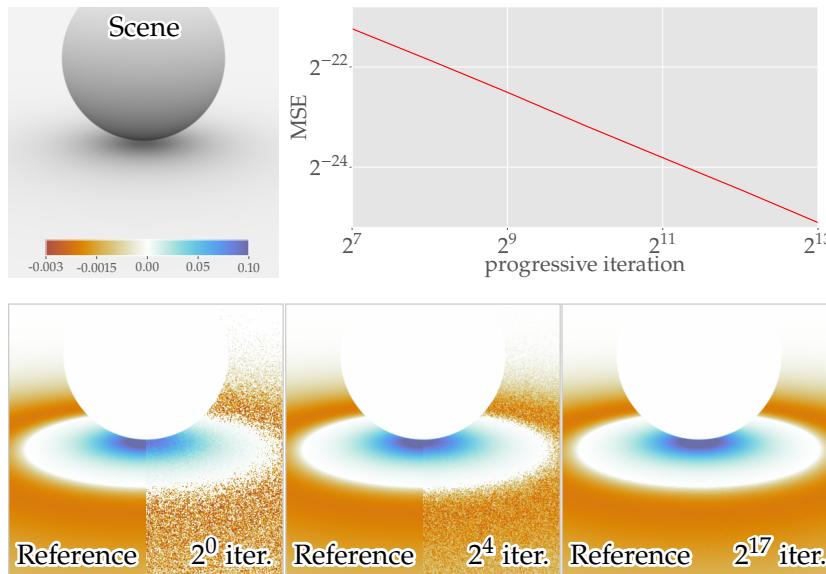
If we replace $\langle \frac{\partial}{\partial x} f(t) \rangle$ with its expectation, this shows that the bias in finite difference estimators results from a one-dimensional blur along the direction of differentiation. This is analogous to how photon beams [150] return a one-dimensional blurred version of volumetric radiance.

Given this equivalence, a Monte Carlo estimator for Eq. (8.45) will have the same bias and variance convergence rates as derived by Jarosz et al. [97] for photon beams when the parameter of differentiation does not modify the path space. In which case, the finite difference step size h_k stands in for the blur kernel width, resulting in,

$$B[\langle I(k) \rangle] = \mathcal{O}(h_k) \quad \text{and} \quad V[\langle I(k) \rangle] = \mathcal{O}(h_k^{-1}). \quad (8.54)$$

This implies that we can formulate a progressive finite difference estimator which maintains finite variance by reducing h_k according to $h_k = h_0 \sqrt{k^{a-1}}$ between secondary samples. This is guaranteed to be a consistent estimator for parameter $a \in (0, 1)$, and provides its optimal MSE convergence rate when $a = 2/3$.

More generally, by the law of large numbers progressive estimators will still converge even if all y samples cannot be correlated, however, the convergence rate will be slower than our derived ideal rates (8.54).



[150]: Jarosz et al. (2011), 'A Comprehensive Theory of Volumetric Radiance Estimation Using Photon Points and Beams'

[97]: Jarosz et al. (2011), 'Progressive Photon Beams'

Figure 8.7: We show the convergence of our progressive finite differences when evaluating the derivative of ambient occlusion on a plane by a sphere as the sphere moves upwards. We employ this scene configuration since its solution is analytic, thus we can measure our techniques convergence accurately.

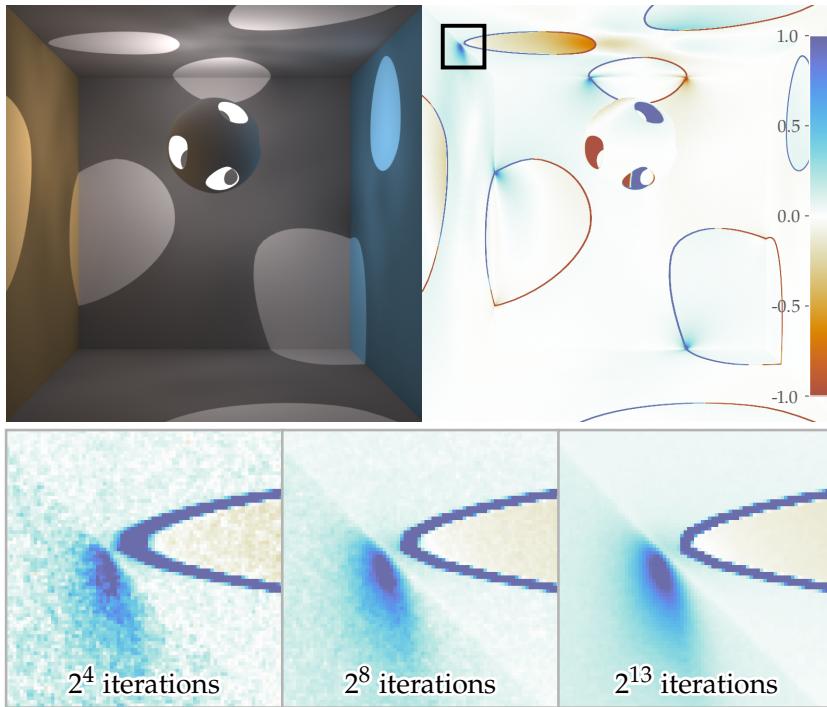


Figure 8.8: Convergence of progressive finite differences w.r.t. moving a point light source to the right in a scene with visibility discontinuities. Each progressive iteration uses 16 samples/pixel.

In Fig. 8.7, we verify that our progressive estimator is consistent in a scene where we render ambient occlusion which has a closed-form solution. In Fig. 8.8, we show a more complex example by employing our progressive finite differences in a scene with obvious discontinuities. This example exhibits how the secondary progressive iterations of our estimator reduce the blurring of finite differences in much the same way as progressive photon beams.

8.5.2 Unbiased transmittance estimation

Estimating the attenuation of light between two points within a participating medium involves estimating a function of expectations ($I = g(F)$) where F is the real optical depth and g is some transmittance function. In classical volumetric transport, the transmittance is simply the exponential function, $g(F) = e^{-F}$, but recent non-exponential formulations [4, 52] allow for employing any monotonically decreasing function starting from $g(0) = 1$ that also vanishes in the limit ($g(\infty) = 0$).

Before the introduction of the work this chapter is based on [112], estimation of general transmittance functions lagged behind the advances in classical transport because most available techniques [103, 104, 110, 121] were explicitly developed for the exponential function (see Chapter 6). This leaves only raymarching (biased) or regular tracking (slow and not applicable to black box extinctions) as possible techniques for general forms of transmittance. Luckily, since transmittance is a function of expectations, it is readily estimatable using both Eqs. (8.5) and (8.14).

[4]: Bitterli et al. (2018), ‘A Radiative Transfer Framework for Non-Exponential Media’

[52]: Jarabo et al. (2018), ‘A Radiative Transfer Framework for Spatially-Correlated Materials’

[112]: Misso et al. (2022), ‘Unbiased and consistent rendering using biased estimators’

[103]: Kutz et al. (2017), ‘Spectral and Decomposition Tracking for Rendering Heterogeneous Volumes’

[104]: Novák et al. (2014), ‘Residual Ratio Tracking for Estimating Attenuation in Participating Media’

[110]: Georgiev et al. (2019), ‘Integral Formulations of Volumetric Transmittance’

[121]: Kettunen et al. (2021), ‘An Unbiased Ray-Marching Transmittance Estimator’

Telescoping series for general transmittances. The telescoping series formation (8.14) allows us—for the first time—to formulate unbiased estimators for general transmittance functions. Let us first make the realization that a function of expectations,

$$I := g(F) = \lim_{n \rightarrow \infty} g\left(\frac{1}{n} \sum_{k=1}^n \langle \tau_r \rangle\right), \quad (8.55)$$

is itself a valid consistent formulation (completing step ①) if we replace F with a secondary Monte Carlo estimator where $\langle \tau_r \rangle$ is an *unbiased* estimate of the real optical depth. Let us define the biased¹⁴ evaluation of a general transmittance function using the average of 2^j optical depth estimates as $I(j)$. We can now formulate an unbiased estimator for a general transmittance function as,

$$\begin{aligned} I &:= I(j) + \sum_{k=j}^{\infty} \Delta_k^{\text{MC}} \quad \text{with } \Delta_k^{\text{MC}} := I(k+1) - I(k), \\ \langle I \rangle &:= \langle I(j) \rangle + \frac{\langle \Delta_k^{\text{MC}} \rangle}{p(k)}, \quad \text{with } \langle \Delta_k^{\text{MC}} \rangle := \langle I(k+1) - I(k) \rangle. \end{aligned} \quad (8.56)$$

where we use MC to mean naive Monte Carlo. To make Eq. (8.56) useful in practice, we have to fully utilize correlations.

Maximally correlated single-term estimation. To explain how we take full advantage of correlating samples between $I(j)$, $I(k)$, and $I(k+1)$, let us consider an explicit construction of Eq. (8.56),

$$\langle I \rangle = g(\langle \tau_r \rangle) + \frac{1}{p(k)} \left(g\left(\frac{1}{2^{k+1}} \sum_{i=1}^{2^{k+1}} \langle \tau_r \rangle_i\right) - g\left(\frac{1}{2^k} \sum_{i=1}^{2^k} \langle \tau_r \rangle_i\right) \right), \quad (8.57)$$

where $j = 0$. Let us denote a sequence of 2^{k+1} real optical depth estimates as $\{\langle \tau_r \rangle_k\} = \{\langle \tau_r \rangle_1, \langle \tau_r \rangle_2, \dots, \langle \tau_r \rangle_{2^{k+1}}\}$. Given these integral estimates, we can naively correlate all evaluations of g by employing optical depth estimates for $I(j)$, $I(k)$, and $I(k+1)$ from the same initial sequence,

$$\langle I \rangle = g(\langle \tau_r \rangle_1) + \frac{1}{p(k)} \left(g\left(\frac{1}{2^{k+1}} \sum_{i=1}^{2^{k+1}} \langle \tau_r \rangle_i\right) - g\left(\frac{1}{2^k} \sum_{i=1}^{2^k} \langle \tau_r \rangle_i\right) \right). \quad (8.58)$$

However, this is not the most optimal use of correlations since $I(0)$ only uses 1 optical depth estimate, $I(k)$ uses half of them, while $I(k+1)$ uses all of them. Blanchet et al. [109] introduced the idea of estimating $I(k)$ twice using two independent halves of samples which modifies Eq. (8.58) into,

$$\begin{aligned} \langle I \rangle &= g(\langle \tau_r \rangle_1) + \frac{1}{p(k)} g\left(\frac{1}{2^{k+1}} \sum_{i=1}^{2^{k+1}} \langle \tau_r \rangle_i\right) \\ &\quad - \frac{1}{2p(k)} \left(g\left(\frac{1}{2^k} \sum_{i=1}^{2^k} \langle \tau_r \rangle_i\right) + g\left(\frac{1}{2^k} \sum_{i=1+2^k}^{2^{k+1}} \langle \tau_r \rangle_i\right) \right). \end{aligned} \quad (8.59)$$

14: We saw how this was biased for the case of exponential transmittance for $j = 0$ in Fig. 6.1.

[109]: Blanchet et al. (2015), ‘Unbiased Monte Carlo for Optimization and Functions of Expectations via Multi-Level Randomization’

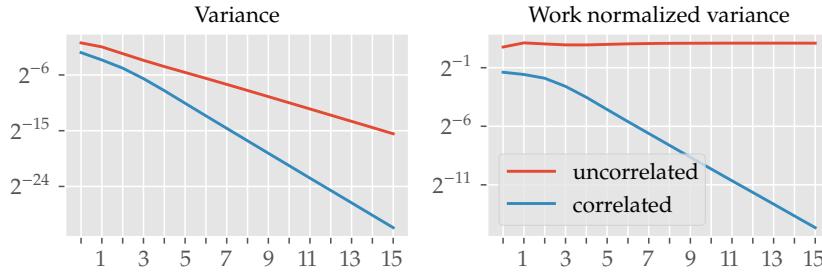


Figure 8.9: Correlating samples can have a dramatic impact on the convergence rates. Here $I := e^{\int -f(x)dx}$ where $f(x)$ is a noise function, and the integral is estimated with a 2^k -sample Monte Carlo estimator (k is plotted along the x-axis). On the left we visualize the variance and on the right we visualize the work-normalized variance when comparing Eq. (8.60) (correlated) to Eq. (8.57) (uncorrelated).

Blanchet et al. originally recommended breaking the samples into even and odd indices, but for simplicity of notation we split by the first 2^k and last 2^k samples. However, Eq. (8.59) is not yet optimally correlated because the evaluation of $I(j) = I(0)$ only employs 1 of the 2^{k+1} integral evaluations. We can further improve Blanchet et al.'s technique for correlating all samples to arrive at,

$$\begin{aligned} \langle I \rangle &= \frac{1}{2^{k+1}} \left(\sum_{i=1}^{2^{k+1}} g(\langle \tau_r \rangle_k) \right) \\ &\quad + \frac{1}{p(k)} g \left(\frac{1}{2^{k+1}} \sum_{i=1}^{2^{k+1}} \langle \tau_r \rangle_i \right) \\ &\quad - \frac{1}{2p(k)} \left(g \left(\frac{1}{2^k} \sum_{i=1}^{2^k} \langle \tau_r \rangle_i \right) + g \left(\frac{1}{2^k} \sum_{i=1+2^k}^{2^{k+1}} \langle \tau_r \rangle_i \right) \right). \end{aligned} \quad (8.60)$$

We compare Eq. (8.60) (correlated) to Eq. (8.57) (uncorrelated) in Fig. 8.9 where we plot both their variance and work-normalized variance. Correlating the Monte Carlo samples leads to a direct improvement in the convergence rates of both quantities. In our implementations we further improve Eq. (8.60) by taking advantage of stratification through replacing all naive Monte Carlo estimators with jittered raymarches,

$$\begin{aligned} \langle I \rangle &= \frac{2^j}{2^{k+1}} \left(\sum_{i=1}^{2^{k+1}/2^j} g(\langle \tau_r \rangle_{RM}^{2^j}) \right) \\ &\quad + \frac{1}{p(k)} g(\langle \tau_r \rangle_{RM}^{2^{k+1}}) \\ &\quad - \frac{1}{2p(k)} \left(g(\langle \tau_r \rangle_{RM}^{2^k}) + g(\langle \tau_r \rangle_{RM}^{2^k}) \right). \end{aligned} \quad (8.61)$$

where $\langle \tau_r \rangle_{RM}^{2^k}$ is a raymarch using a step-size proportional to 2^{-k} . Thus, Eq. (8.61) is an unbiased telescoping series based raymarching estimator. Note that we can construct $2^k/2^j$ raymarches of step-size proportional to 2^{-j} from the steps sampled in an initial jittered raymarch with step-sizes proportional to 2^{-k} . However, in practice, special care has to be taken while implementing Eq. (8.61) to make sure that all raymarches are unbiased estimates of the integrand. For example, if the first $\langle \tau_r \rangle_{RM}^{2^k}$ in Eq. (8.61) always uses the odd steps while the second $\langle \tau_r \rangle_{RM}^{2^k}$ only uses the even steps similar to Blanchet et al.'s [109] recommendation, then both raymarches

[109]: Blanchet et al. (2015), 'Unbiased Monte Carlo for Optimization and Functions of Expectations via Multi-Level Randomization'

will never evaluate the entire optical depth, thus, $\langle I \rangle$ will not be unbiased. In practice, we typically set $j = 0$ for practical renders and initialize the base step-size based on the same μ_t that the null-scattering methods use.

Pragmatic PMF. We choose to employ a geometric PMF ($p(k) = r(1-r)^k$) with $r = 0.65$, as recommended by Blanchet et al. [109] when applying a single-term estimator for Eq. (8.14) to smooth functions of expectations.

Unbiased volumetric path tracing in general media. In Appendix B.2, we introduce a generalized VRE¹⁵,

$$\begin{aligned} L_o(\mathbf{x}, \vec{\omega}_o, \lambda) := & \int_a^b T_r(\mathbf{x}_a, \mathbf{x}_t, \lambda+1) \mu_a(\mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) dt \\ & + \int_a^b T_r(\mathbf{x}_a, \mathbf{x}_t, \lambda+1) \mu_s(\mathbf{x}_t) L_m(\mathbf{x}_t, \vec{\omega}_o, 1) dt \\ & + T_r(\mathbf{x}_a, \mathbf{x}_b, \lambda) L_s(\mathbf{x}_d, \vec{\omega}_o, 0), \end{aligned} \quad (8.62)$$

that supports any transmittance model fulfilling the conditions of a valid transmittance function in Bitterli et al.'s non-exponential framework [4]. By employing our generalized unbiased raymarching estimator (8.61),

$$\langle L_o(\mathbf{x}, \vec{\omega}_o, \lambda) \rangle := \begin{cases} \frac{1}{P_m P_a} \frac{\langle T_r(\mathbf{x}_a, \mathbf{x}_t, \lambda+1) \rangle \mu_a(\mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o)}{p(\mathbf{x}_t)} & \text{with prob. } P_m P_a \\ \frac{1}{P_m P_s} \frac{\langle T_r(\mathbf{x}_a, \mathbf{x}_t, \lambda+1) \rangle \mu_s(\mathbf{x}_t) \rho_p(\mathbf{x}_t, \vec{\omega}_i, \vec{\omega}_o) \langle L_i(\mathbf{x}_t, \vec{\omega}_i, 1) \rangle}{p(\mathbf{x}_t) p(\vec{\omega}_i)} & \text{with prob. } P_m P_s \\ \frac{1}{1-P_m} \langle T_r(\mathbf{x}_a, \mathbf{x}_b, \lambda) \rangle \langle L_s(\mathbf{x}_d, \vec{\omega}_o, 0) \rangle & \text{with prob. } 1 - P_m. \end{cases} \quad (8.63)$$

we can now formulate the first generally unbiased volumetric path tracer which supports both classical and non-classical media¹⁶. We assume that $p(\mathbf{x}_t)$ and P_m are determined using *biased* raymarching. We also assume P_a and P_s are chosen similarly to Eq. (5.23).

Unlike Eq. (8.63), most modern path tracers derived from the classical VRE never have to explicitly evaluate the transmittance since they importance sample $p(\mathbf{x}_t) \propto T_r(\mathbf{x}_a, \mathbf{x}_b)$. While this can also be possible for Eq. (8.63), it requires transmittance-specific analysis. While we do not perform that analysis within this dissertation, we provide a generalization of the classical Volterra integral formulation for general transmittances in Appendix B which can be used to derive specific importance sampling schemes by the interested reader.

In Fig. 8.10, we demonstrate this estimator on the WDAS cloud scene using the two-parameter Davis and Mineev-Weinstein transmittance model which we previously introduced in Chapter 4,

$$g(F) := (1 + F^\beta C^{1+\beta})^{-\frac{F^{1-\beta}}{C^{1+\beta}}}. \quad (8.64)$$

We visualize a variety of settings for β and C which together control the shape of the transmittance function. We compare our results to regular

[109]: Blanchet et al. (2015), 'Unbiased Monte Carlo for Optimization and Functions of Expectations via Multi-Level Randomization'

¹⁵: In this formulation, λ is an integer that represents which transmittance function out of the three within Bitterli et al.'s transport function is being employed. Please see Appendix B for a more thorough definition of λ .

[4]: Bitterli et al. (2018), 'A Radiative Transfer Framework for Non-Exponential Media'

¹⁶: Bitterli et al. only introduced unbiased rendering algorithms for pink noise transmittance

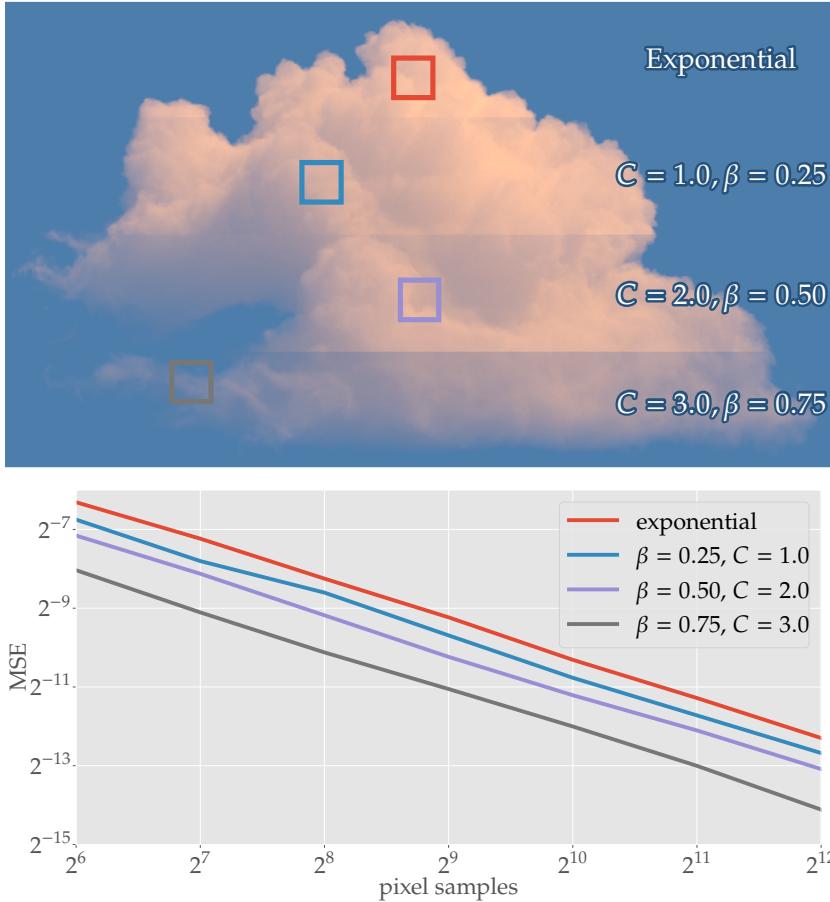


Figure 8.10: We visualize a converged render (top) and the variance plots (bottom) of our unbiased path tracer for rendering general volumetric media (8.63) using telescoping based unbiased ray-marching (8.61). The split image shows classical transmittance as well as non-exponential transmittances with three different choices for C and β . The plot shows the convergence of four select pixels towards a regular-tracked ground truth.

tracking, confirming that Eq. (8.63) is unbiased and that the variance $V[\langle I \rangle_n]$ falls off at the expected $\mathcal{O}(n^{-1})$ convergence rate.

Taylor series for pink-noise transmittance. We can also apply the Taylor series formulation to formulate estimators for Eq. (8.64). For arbitrary values of β , we found that the complexity of the derivative terms in the Taylor expansion grows exponentially for increasing values of j . However, for the special case of pink-noise ($\beta = 1$), Eq. (8.64) simplifies to,

$$g(F) := (1 + FC^2)^{-\frac{1}{C^2}}, \quad (8.65)$$

which has a compact Taylor series formulation (8.5):

$$\begin{aligned} I &:= g(F) = \sum_{j=0}^{j-1} \Delta_k + \sum_{j=j}^{\infty} \Delta_k, \quad \text{where} \\ \Delta_k^{\text{pink}} &:= \frac{(\alpha - F)^k}{k!} (1 + \alpha C^2)^{\frac{-1}{C^2} - k} \prod_{i=1}^{k-1} (1 + i C^2). \end{aligned} \quad (8.66)$$

We create a prefix-sum primary estimator of Eq. (8.66) by defining $p(k)$

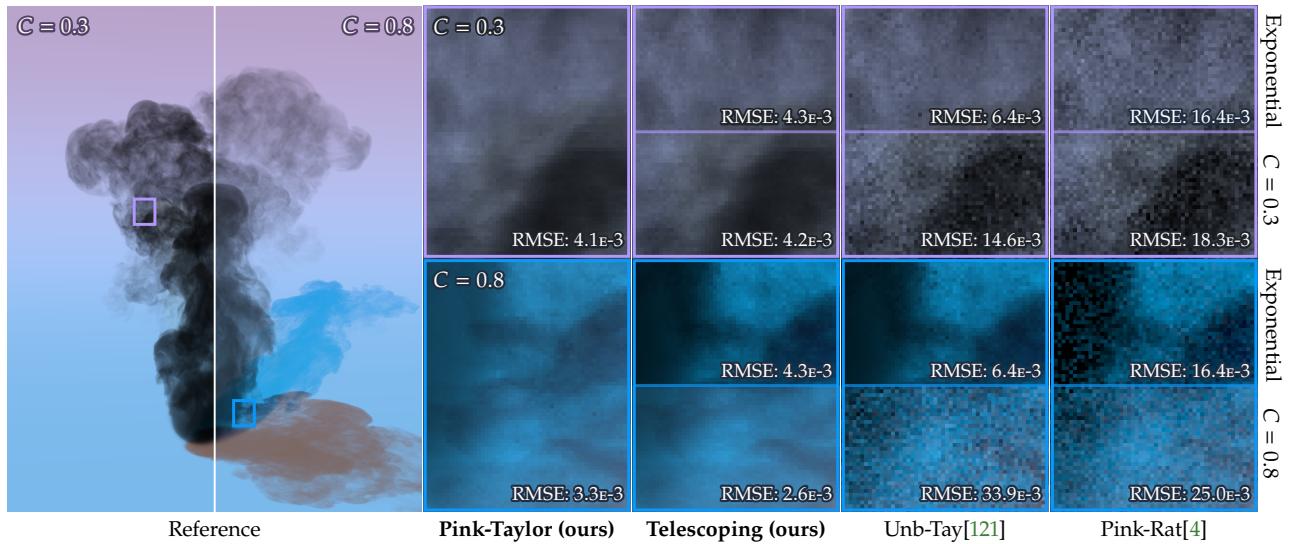


Figure 8.11: Comparing a variety of transmittance estimators for a non-exponential medium with the Davis and Mineev-Weinstein transmittance model (8.64) for two different values of C and an equal number of extinction calls ($1.2\text{e}8$ for $C = 0.3, 0.8$, and $1.0\text{e}8$ for exponential). The three estimators on the right can be used as estimators for exponential transmittance, so we also include inset comparisons for an exponential medium at the top of their respective insets. RMSE values are measured across the entire image.

indirectly via Russian Roulette probabilities:

$$P_{\text{acc}}(k) := \frac{1 + (k - 1)C^2}{k}, \quad p(k) := (1 - P_{\text{acc}}(k)) \prod_{i=0}^{k-1} P_{\text{acc}}(i). \quad (8.67)$$

Since this is independent of both the expansion point, α , and any $\langle F \rangle$, we can take advantage of all the same variance reduction techniques that Kettunen et al. [121] introduced for exponential transmittance (see Appendix A). We use uniformly jittered regular samples with a fixed step-size (0.2) to estimate $\langle F \rangle$. We set $j = 3$, and choose to evaluate the bias only a portion ($P = 0.1$) of the time which reduces the cost while slightly increasing the variance of the estimator.

In Fig. 8.11, we compare both our Taylor series (8.66) and telescoping series (8.61) estimators against prior techniques for unbiased estimation of both pink-noise and exponential transmittance functions. Bitterli et al. [4] showed that ratio tracking (or any existing exponential transmittance estimator) can be reused for pink noise by first sampling a multiplicative factor (γ) from a gamma distribution, and then estimating exponential transmittance in a medium with its density scaled by γ . We also apply this trick to Kettunen et al.’s [121] more recent unbiased Taylor series based ray marching technique to allow it to handle both exponential and pink-noise transmittance¹⁷. As shown in Fig. 8.11, our estimators consistently outperform these prior methods, sometimes reducing RMSE by a factor of 10× at equal cost in the non-exponential case.

[121]: Kettunen et al. (2021), ‘An Unbiased Ray-Marching Transmittance Estimator’

[4]: Bitterli et al. (2018), ‘A Radiative Transfer Framework for Non-Exponential Media’

17: The main difference between our Eq. (8.66) and Kettunen et al.’s estimator is that ours uses a tailored set of Russian Roulette probabilities (8.67) based on Eq. (8.65). Our estimator is derived from our generalization of Kettunen et al.’s technique, which we include in Appendix A.

8.5.3 Unbiased photon mapping

The photon mapping family of algorithms are the most practical choice when rendering complicated caustic effects. This is especially true in scenes containing many reflections of caustics where unbiased alternatives, such as bidirectional path tracing, struggle.

Photon mapping algorithms first emit and trace photons from light sources as a first pass which are then used during a second pass to approximate the radiance at a point in the scene via density estimation¹⁸. However, density estimation yields biased results, where the bias is fully dependent on the radius (r) of the density kernel. While the bias is guaranteed to vanish as the radius approaches zero, the probability of a photon landing within the kernel radius of a path vertex will quickly vanish. Meaning, by taking the radius to its limit, naive photon mapping will be probabilistically *unrenderable*.

Prior "unbiased" photon mapping. While an unbiased formulation of photon mapping exists [95], it cannot handle purely specular caustics as it requires forming direct connections between a photon subpath and camera subpath. If there is a specular surface between the previous vertex on the photon subpath and the current vertex on the camera subpath, then the contribution of the entire path, once the direct connection through the specular surface is made, will probabilistically always be 0.

¹⁸: We previously reviewed the photon mapping family of algorithms in Sec. 5.3.

[95]: Qin et al. (2015), 'Unbiased Photon Gathering for Light Transport Simulation'

Photon mapping as a consistent formulation. To convert photon mapping into a consistent formulation, we first denote the biased expectation of the photon mapping radiance estimate as $I(j)$, where j maps to a monotonically decreasing sequence of radii (r_j) used for density estimation. We can then construct a telescoping series formulation (8.14) and single-term primary estimator (8.21) for this problem as,

$$\begin{aligned} I &:= I(j) + \sum_{k=j}^{\infty} \Delta_k^{\text{PM}}, \quad \text{where } \Delta_k^{\text{PM}} := I(k+1) - I(k) \\ \langle I \rangle &:= \langle I(j) \rangle + \frac{\langle \Delta_k^{\text{PM}} \rangle}{p(k)}, \quad \text{where } \langle \Delta_k^{\text{PM}} \rangle := \langle I(k+1) - I(k) \rangle, \end{aligned} \tag{8.68}$$

which is a fully debiased form of photon mapping, even in the presence of specular reflections of caustics. In essence, Eq. (8.68) represents the bias as consecutive differences of kernel density estimates with monotonically decreasing radii. We provide a visualization of Eq. (8.68) as a form of debiasing in Fig. 8.12 where we visually decompose our consistent formulation. Unfortunately, we will prove that it is *not possible* to formulate an unbiased photon mapping algorithm with finite work-normalized variance.

Unbiased photon mapping has infinite work-normalized variance. We begin with the biased but consistent base estimator, $\langle I(k) \rangle$, which traces M_k photons and performs density estimation with a kernel of radius r_k . For suitably chosen M_k and r_k , this estimator converges to I as $k \rightarrow \infty$ [87].

[87]: Jensen (2001), *Realistic Image Synthesis Using Photon Mapping*

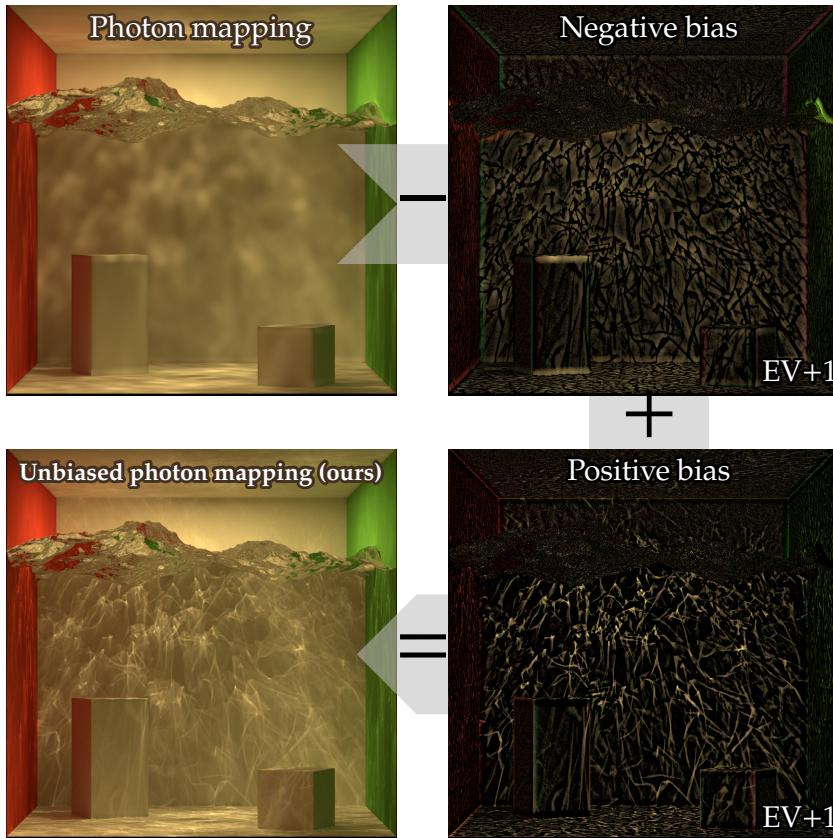


Figure 8.12: Decomposition of our unbiased photon mapping algorithm (8.68) into the sum of a biased solution and the compensation for that bias. We separate the bias into positive and negative parts for the purpose of visualization.

We form a single-term telescoping series estimator (8.68) assuming perfectly correlated evaluations of $\langle \Delta_k \rangle = \langle I(k+1) - I(k) \rangle$. Within the context of photon mapping, two radiance estimates are perfectly correlated when they both share the same set of photons. If both $\langle I(k) \rangle$ and $\langle I(k+1) \rangle$ trace different amounts of photons ($M_{k+1} \neq M_k$), we ignore all unshared photons for the rest of this analysis, thus giving a lower bound on the variance and bias. In the presence of perfectly correlated photons, the evaluation of the two estimators differ only in the employed kernel radius (r_k and r_{k+1}), thus we can express the difference of the two estimators as a single photon mapping estimator that uses the difference of the two kernels for density estimation ($\langle I(k+1) \rangle - \langle I(k) \rangle = \langle I(k+1) - I(k) \rangle$).

This allows us to use Knaus et al.'s [90] analysis as a basis to derive the expectation and variance of $\langle \Delta_k \rangle$ in order to derive the variance-optimal PMF for importance sampling the telescoping series. Repeating the derivations of Knaus and Zwicker using the modified difference kernel yields the following asymptotic relationships:

$$V[\langle \Delta_k \rangle] \propto \frac{1}{M_k} \int (K(r_{k+1}, x) - K(r_k, x))^2 dx, \quad (8.69)$$

$$\Delta_k \propto r_k^2, \quad (8.70)$$

where $K(r, x)$ evaluates the density-estimation kernel of radius r at location x . We previously reviewed different kernel density functions in Sec. 5.3.

The asymptotic rates depend on the relationship between the number of

[90]: Knaus et al. (2011), 'Progressive Photon Mapping'

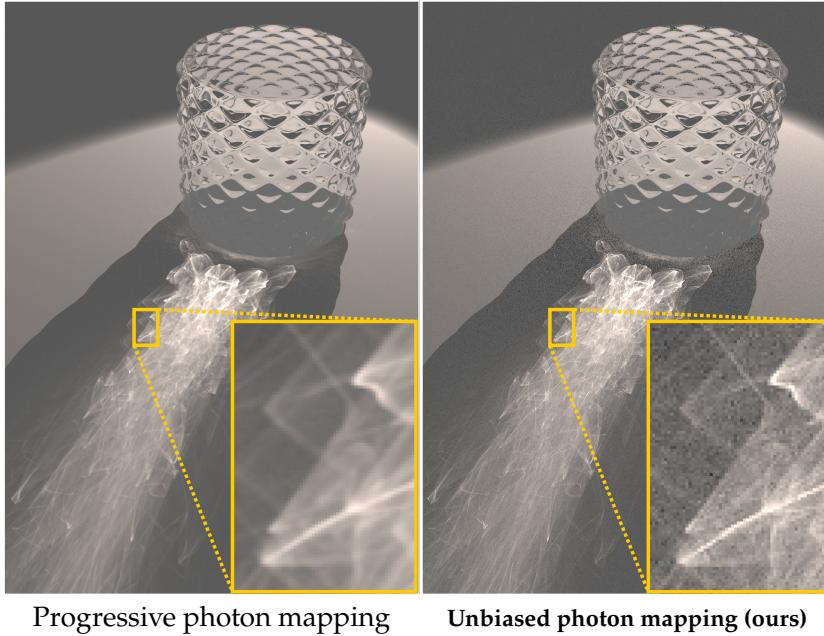


Figure 8.13: Compared to progressive photon mapping (with $\alpha = \frac{2}{3}$) at equal number (3.3×10^6) of emitted photons, our unbiased photon mapping trades blurring (i.e., bias) for noise.

photons (M_k) and the kernel radius r_k with respect to k , as well as the specific choice of kernel. Prior work [90, 91] have shown that $r_k = \sqrt{k^{\alpha-1}}$ with $0 < \alpha < 1$ yields consistent progressive estimation. This leads to the following asymptotic rates for the expectation and the variance:

$$\Delta_k = \mathcal{O}(k^{\alpha-2}), \quad (8.71)$$

$$\text{V}[\langle \Delta_k \rangle] = M_k^{-1} \mathcal{O}(k^{-\alpha}), \quad (\text{constant kernel (5.15)}) \quad (8.72)$$

$$\text{V}[\langle \Delta_k \rangle] = M_k^{-1} \mathcal{O}(k^{-\alpha-1}). \quad (\text{cone kernel (5.16)}) \quad (8.73)$$

[90]: Knaus et al. (2011), ‘Progressive Photon Mapping’

[91]: Georgiev et al. (2012), ‘Light Transport Simulation with Vertex Connection and Merging’

Given these rates, we can now test whether our primary estimator (8.68) can yield finite work-normalized variance. First, the optimal PMF,

$$p(k) \propto \sqrt{\text{V}[\langle \Delta_k \rangle] + (\Delta_k)^2} = \sqrt{M_k^{-1} \mathcal{O}(k^{-\alpha-1}) + \mathcal{O}(k^{2\alpha-4})}, \quad (8.74)$$

must be normalizable where we use the best-case cone-kernel variance (8.73). Second, the expected work

$$\text{C}[\langle I \rangle_1^0] \propto \sum_{k=1}^{\infty} M_k \cdot p(k) \propto \sum_{k=1}^{\infty} \sqrt{M_k \mathcal{O}(k^{-\alpha-1}) + M_k^2 \mathcal{O}(k^{2\alpha-4})}, \quad (8.75)$$

must be finite where we use the number of emitted photons as an estimate for the cost. Unfortunately, there is no choice for M_k and α that can satisfy both of these conditions. The estimator is still debiasable, but with infinite work-normalized variance, i.e., with a reduced error convergence rate for a naive secondary estimator (8.23).

Unbiased photon mapping. In practice, we formulate unbiased photon mapping via a single-term primary estimator (8.21), and a naive secondary estimator (8.23) completing step ⑥. We use a cone kernel (5.16) and

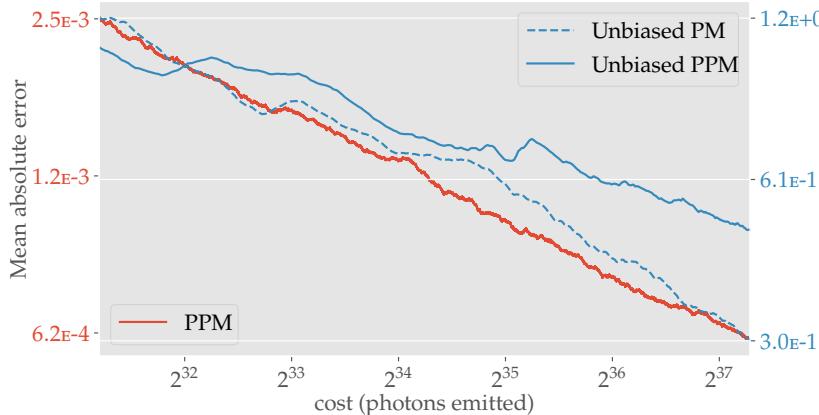


Figure 8.14: We show the convergence of the mean absolute difference of our unbiased photon mapping and unbiased progressive photon mapping against traditional progressive photon mapping for a canonical scene containing only a plane and a point light source while measured from a single location for equal cost. The convergence rates between progressive photon mapping and our unbiased photon mapping are similar by design, but the initial cost of the unbiased methods is not insignificant (see different error bars.).

fully correlate all kernel evaluations by essentially estimating $\langle \Delta_k^{\text{PM}} \rangle := \langle I(k+1) - I(k) \rangle$ instead of $\langle I(k+1) \rangle - \langle I(k) \rangle$. We use $p(k) \propto \mathcal{O}(k^{\alpha-2})$, and set the number of global photons to $M_k \propto \mathcal{O}(k^{1-ca})$ where $c = 1.001$ and $\alpha = 2/3$. We choose this specific parameter configuration because the resulting convergence rate should be similar to progressive photon mapping (see Fig. 8.14) once all defined terms are plugged in. Compared to prior progressive photon mapping methods, our debiased photon mapping trades bias (i.e., blurring) for noise (Fig. 8.13).

Unbiased progressive photon mapping. We can also debias progressive photon mapping by estimating the bias and biased terms simultaneously (8.42). In Fig. 8.14, we compare convergence rates of our debiased methods and prior progressive photon mapping in terms of emitted photons. While our unbiased photon mapping can maintain similar convergence rates to biased progressive photon mapping, our unbiased progressive photon mapping tends to perform significantly worse as after every progressive iteration it is more prone to sampling higher variance terms.

8.6 Conclusion

We have walked through our general recipe (see Fig. 8.1) for formulating consistent estimators and showed how to apply it to derive novel solutions for existing problems in light transport. Using our framework, we have introduced a debiased raymarching approach which is the first unbiased estimator to support rendering general non-exponential media. We have introduced the first unbiased formulation for photon mapping and progressive photon mapping which supports all caustics. Additionally, we have also shown how to make finite differences unbiased for smooth functions, and formulated a consistent estimator in the presence of discontinuities.

8.6.1 Discussion

The primary estimators we have employed rely on either a Taylor series (8.5) or telescoping series (8.14) expansion, so a reasonable question to ask

is which one should be employed when they both apply? In practice, we have found the answer to this question to be extremely problem-dependent. For example, some of the variance reduction techniques introduced by Kettunen et al. [121] for classical transmittance estimation (see Appendix A) cannot be utilized for reciprocal estimation due to its restricted interval of convergence. While an extensive comparative analysis could prove invaluable in guiding such design decisions in future work, there are some clear trade-offs between the two techniques that are already apparent from our investigations.

The telescoping series performs best when implemented as a global algorithm that combines evaluations of entire pixel samples as a post process. This requires re-structuring rendering algorithms to correlate entire paths through integrating multiple values simultaneously for maximum performance. For example, to support our unbiased telescoping based raymarching algorithm, we modified our volumetric path tracer to keep track of $I(j)$, $I(k + 1)$, and the two estimates of $I(k)$ separately while tracing a single path. On the other hand, the Taylor series is applicable to only smooth functions of expectations, making it less general, but more self-contained and easier to maintain. By specifying relatively large initial j 's and each formulations' respective variance reduction techniques, both the Taylor and the telescoping series can reach comparably low work-normalized variances¹⁹.

Many algorithms in graphics use Russian Roulette to probabilistically terminate the estimation of an infinite series in finite time. However, employing Russian Roulette means that we do not directly select a PMF ahead of time, rather an effective PMF arises as a consequence of each Russian Roulette decision. This allows many algorithms to adapt to local information on the fly. So far, our applications of the telescoping series (8.14) have only explored specifying a single global PMF. In the case of path tracing, this would be like specifying the probability of sampling specific length paths without knowing the contents of a scene. Exploring and constructing estimators which exploit local information using Russian Roulette and weight windows [151] could be a promising avenue for future work.

[121]: Kettunen et al. (2021), 'An Unbiased Ray-Marching Transmittance Estimator'

19: As an example, both unbiased ray-marching techniques perform phenomenally in the presence of absorption-only exponential media.

[151]: Vorba et al. (2016), 'Adjoint-Driven Russian Roulette and Splitting in Light Transport Simulation'

Summary and future work

In this dissertation we sought to develop consistent solutions for non-linear functions of expectations within light transport. While no single solution exists for the broad range of all possible non-linear problems, we instead provided a generalized recipe (see Fig. 8.1) for scientists and practitioners alike to employ in deriving practical solutions, which was inspired by our exploration of the problem space.

In performing these theoretical and practical explorations, we derived many novel techniques and developed many insights for non-linear applications within the field of light transport which hopefully have far reaching applications. We will now briefly review the novel contributions we have made throughout this dissertation.

9.1 Summary of contributions

Integral formulations of volumetric transmittance. In Chapter 6, we derived and introduced three integral formulations for exponential transmittance based on two general consistent formulations (IVP and Maclaurin series). These integral formulations effectively removed the need to estimate any nonlinearities allowing for the direct application of Monte Carlo integration. Using the IVP formulation, we were able to perform extensive variance analysis to propose idealized and practical adaptive variants of all prior tracking-based transmittance estimators, which can result in more efficient solutions. We then proposed novel transmittance estimators based off of the Maclaurin series expansion of the exponential function, and rederived ratio tracking as a single-term estimator of the Maclaurin series formulation. Next, we derived a hypercube formulation from the exponential power series and drew analogies to classical forms of the rendering equation, allowing us to formulate transmittance estimators using MIS. All the estimators we introduced in Chapter 6 exhibit finite work-normalized variance under the condition that we possess a bounding μ_t or all volumes have finite spatial extents.

Progressive techniques for volumetric rendering. In Chapter 7, we made the realization that even if we possess an estimator that has finite work-normalized variance, the estimator can still exhibit *numerically infinite* variance. When dealing with practical applications, such as the rendering process for generating animated feature films, practitioners desire both accurate and efficient solutions. Due to the nature of stochastic estimation, we do not always have a priori knowledge of good importance sampling distributions, so we occasionally have to introduce our own through the application of control variates¹. The lack of a priori knowledge

9.1 Summary of contributions	186
9.2 Discussion and limitations of our recipe	188
9.3 Future work	189

1: i.e. the combined free-flight distribution.

in combination with the practical desire to employ a fast importance sampling scheme can be disastrous, and has been one of the major roadblocks inhibiting production renderers from being able to render black box media (where we have no a priori knowledge of its extinction) in a consistent manner.

We exhibited how progressive estimation can be employed instead of unbiased estimation to completely alleviate this problem. In doing so, we devised two progressive solutions, namely progressive extinctions and progressive emitter sampling. Our progressive extinctions technique allows for adaptively developing tightly conservative estimates for free-flight sampling distributions on-the-fly while purposefully biasing all media to avoid excessive variance. Progressive emitter sampling slowly builds a registry of voxels containing emissive media over time, which is queried during emitter sampling. We also discussed how progressive emitter sampling can be made unbiased by directly accounting for the non-active voxels in the emitter registry on-the-fly during tracking. All of these progressive techniques are probabilistically guaranteed to discover "bounding" distributions in finite time for most common medium representations and procedural modifications. Thus, every individual pixel sample will be an unbiased estimate after a "bounding" distribution has been discovered.

We theoretically and empirically showed that the bias of a general progressive estimator whose iterations become independently unbiased after a finite amount of time will converge *faster* than the variance. Meaning, the asymptotic convergence of the MSE of such a progressive estimator will be *equivalent* to the asymptotic convergence of the MSE of an unbiased estimator which employs the same "bounding" distribution the progressive estimator converges to.

Unbiased and consistent rendering using biased estimates. In Chapter 8, we reviewed our general recipe for formulating consistent solutions of non-linear functions of expectations. We reviewed all four of the basic consistent formulations (Taylor, IVP, telescoping, progressive) which we have employed within this dissertation, discussed how to derive variance-optimal sampling distributions, and showed how those distributions could be used to determine whether an estimator will have finite or infinite work-normalized variance. Since the prior chapters have covered the finite work-normalized variance case, we briefly reviewed recommendations for formulating practical estimators. For the case of estimation under infinite work-normalized, we showed that unbiased estimation is still possible, but with reduced error convergence rates. As an alternative, we showed that progressive estimation can actually result in finite variance estimators if they are designed carefully, even if the primary estimators have infinite work-normalized variance. We additionally discovered that MIS theoretically has the potential to alleviate infinite work-normalized variance, however, we do not have any practical examples of this.

We additionally walked through how to apply our recipe to three novel problems in light transport. In summary, we introduced unbiased and progressive finite differencing for generating more accurate validation

baselines for differentiable rendering, we introduced unbiased volumetric rendering for general non-classical participating media through unbiased raymarching, and we introduced unbiased photon mapping, and unbiased progressive photon mapping.

9.2 Discussion and limitations of our recipe

Attempting to propose a list of steps for practitioners to follow to devise efficient estimators for a problem space as broad as the space of *all non-linear functions* is, admittedly, ambitious. Due to the broad range of possibilities, there are obviously topics and weaknesses which are not fully explored by the limited scope of applications we have considered within light transport. In this section, key topics regarding the applicability and limitations of our recipe will be reviewed.

Correlations. Nearly all techniques we have proposed are fully reliant on correlating samples to be useful in practice. Correlations are such a core aspect of the usability of our recipe that one obvious flaw might be the question of what to do when correlating samples is nearly impossible, and are there any problems where this will always be the case? The only application we have considered where this worst-case scenario might happen is during finite differencing when the path space is significantly dependent on what we are differentiating with respect to. Arguably, a technique for forward differentiation should not be employed in that case.

Discontinuities. Discontinuous functions (i.e. a step function) may not have reliable consistent formulations at the discontinuities, resulting in primary estimators which may not have the correct expected value. Fortunately, in light transport, we often only need to evaluate the integral of the functions of interest² which *can* be estimated consistently since the probability of sampling the discontinuity exactly is probabilistically zero. Unfortunately, in the presence of discontinuities, their convergence rates can be slow.

2: For example, we compute radiance by integrating over a pixel, or integrate the transmittance over distance.

Unbiased infinite work-normalized variance estimators. We have introduced a few estimators which possess infinite work-normalized variance, specifically unbiased finite differencing in the presence of discontinuities and unbiased photon mapping. In both of these instances, one question that arises is, why employ unbiased techniques when progressive alternatives can be designed to have finite variance and remain convergent? The unfortunate answer is that more analysis needs to be done before we can give a conclusive answer.

According to the practical results within this dissertation, unbiased estimators always perform worse than their progressive alternatives in the infinite work-normalized variance case. However, our brief theoretical analysis of MIS in Sec. 8.4.2 implies that if we can combine infinite work-normalized

variance estimators with finite work-normalized variance estimators, the resulting MIS formulation can exhibit finite work-normalized variance. Meaning, if an infinite work-normalized variance estimator is more efficient at capturing certain effects (i.e. caustics), then deriving MIS estimators should be *more* efficient than their progressive alternatives.

Unbiased vs. consistent estimation in the finite case. Chapter 7 revealed a fascinating insight. An initially biased progressive estimator whose future progressive iterations become unbiased after a finite point in time³, will *converge asymptotically* to the behavior of a fully unbiased secondary estimator. This is due to the bias disappearing at a *faster* rate than the variance. This implies that progressive estimators under finite work-normalized variance can be extremely useful, but they have been vastly ignored by the graphics literature⁴. A question then arises, given that progressive estimators outperform unbiased estimators in the infinite work-normalized variance case, can we utilize the flexibility of (biased) progressive estimation through clever construction to devise estimators which are generally *more* efficient than their unbiased alternatives?

3: $E[\langle I(j) \rangle] = I$ for all iterations $j > k$.

4: Specifically in the finite variance case. All prior instances of progressive estimation known to the author of this dissertation have been utilized for infinite work-normalized variance problems.

9.3 Future work

In this section we review potential avenues for future work relating to this dissertation as a whole.

MIS to alleviate infinite work-normalized variance. As mentioned in the prior section, employing MIS to alleviate infinite work-normalized variance is an obvious future endeavor which could significantly improve our recipe for formulating consistent estimators. The obvious application for this is in regards to rendering caustics unbiasedly.

The space of stable distributions. Estimators which have finite mean but infinite variance have a connection to the family of *stable distributions* which are also known as the family of Levy alpha-stable distributions [152, 153]. Specifically analyzing the connection between stable distributions and our infinite variance estimators is a potential avenue for future work.

[152]: Lévy (1925), *Calcul des probabilités*

[153]: Mandelbrot (1960), 'The Pareto-Lévy Law and the Distribution of Income'

"Optimal" infinite work-normalized variance PMFs. One unfortunate part of our process for formulating unbiased infinite work-normalized variances estimators is that our choice of a "pragmatic PMF" is slightly arbitrary. Future work should focus on devising what an "optimal" PMF should be where potentially looking into devising and minimizing a "work-normalized-error" is one option.

Clamped normalizable PMFs. A variance-optimal PMF derived for a problem which exhibits infinite work-normalized variance will usually be unnormalizable. However, any unnormalizable PMF is only unnormalizable because of the behavior of its heavy-tail distribution. Any *clamped* unnormalizable PMF *can* be normalized. Meaning, there is an avenue of future work where we analyze progressive estimators which, instead of incrementing the starting index, they increment the maximum index while employing the variance-optimal PMF for the truncated series.

We look forward to see what difficult problems arise in the future of physically based rendering, and we hope our simple exploration contributes to future discoveries.

APPENDIX

Generalized Taylor series based raymarching

A

Kettunen et al. [121] introduced a transmittance estimator based on the iterative power series formulation for the exponential function (6.72) which was derived by applying a series of variance reduction techniques. The resulting prefix-sum transmittance estimator was referred to as a form of unbiased raymarching since the first term in the prefix-sum becomes an instance of biased raymarching. In terms of variance, their estimator is by far the *best* estimator out of all current Taylor series based methods for exponential transmittance.

Due to the phenomenal performance of their variance reduction techniques, any efficient Taylor series based estimator for non-linear functions of expectations should take inspiration from their design. For this reason, in this appendix we will generalize Kettunen et al.'s Taylor series based raymarching estimator [121] to be applicable to general non-linear functions of expectations.

We begin by restating the generalized Taylor series formulation for non-linear functions of expectations (8.5) in the context of transmittance estimation as¹,

$$I(a, b) := g \left(- \int_a^b \mu_r(x_t) dt \right) = \underbrace{\sum_{k=0}^{j-1} \Delta_k^{\text{Tay}}(a, b)}_{I(j)} + \underbrace{\sum_{k=j}^{\infty} \Delta_k^{\text{Tay}}(a, b)}_{B(j)} \quad (\text{A.1})$$

$$\Delta_k^{\text{Tay}}(a, b) := g^{(k)} \left(- \int_a^b \mu_t dt \right) \frac{\left(\int_a^b \mu_n(x_t) dt \right)^k}{k!},$$

where $g^{(k)}$ is the k th derivative of the non-linear function evaluated at the expansion point $\alpha = - \int_a^b \mu_t dt = -\tau_t$. A general prefix-sum estimator for Eq. (A.1) is formulated as,

$$\langle I(a, b) \rangle := \sum_{k=0}^{j-1} \langle \Delta_k^{\text{Tay}}(a, b) \rangle + \sum_{k=j}^N \frac{\langle \Delta_k^{\text{Tay}}(a, b) \rangle}{P_{\text{acc}}(k)} \quad (\text{A.2})$$

$$\langle \Delta_k^{\text{Tay}}(a, b) \rangle := g^{(k)} \left(- \int_a^b \mu_t dt \right) \frac{1}{k!} \prod_{i=1}^k \frac{\mu_n(x_i)}{p(x_i)},$$

where $P_{\text{acc}}(k)$ is the accumulated Russian Roulette weight for iteration k , N is the index where Russian Roulette terminates the prefix-sum estimator, and $p(x_i)$ is the probability density of importance sampling the i th integral sample.

A.1 Pre-processed Russian Roulette	193
A.2 Unbiased estimates as expansion points	194
A.3 Raymarched integral estimates	195
A.4 Symmetric Means	196

[121]: Kettunen et al. (2021), 'An Unbiased Ray-Marching Transmittance Estimator'

¹: Transmittance functions are always parameterized by the integral of $\mu_r(x)$ over some path segment (i.e. the real optical depth). We additionally make the assumption that g is a function of the *negative* optical depth in our derivations causing the individual terms to have alternating signs. For this reason, our expansion point (α) is always set to be negative when it is non-zero to counteract the alternating sign (i.e to get $\mu_n(x_t)$ in the integrand). While this assumption is not always true, it is for the transmittance models we consider in this dissertation. Thus, we keep it for notational consistency and to simplify our derivations. Performing our derivations on non-linear transmittance functions which depend on a positive optical depth should be trivial.

A.1 Pre-processed Russian Roulette

For brevity, let us assume that $j = 1$ and set $P_{\text{acc}}(0) = 1$ so we can rewrite Eq. (A.2) as,

$$\langle I(a, b) \rangle = \sum_{k=0}^N \frac{\left\langle \Delta_k^{\text{Tay}}(a, b) \right\rangle}{P_{\text{acc}}(k)}. \quad (\text{A.3})$$

Let us then define the product of independent *null optical depth estimates* to be,

$$\langle \tau_n^N \rangle := \frac{\mu_n(x_1)}{p(x_1)} \cdot \frac{\mu_n(x_2)}{p(x_2)} \cdot \dots \cdot \frac{\mu_n(x_N)}{p(x_N)}, \quad (\text{A.4})$$

where $\langle \tau_n^0 \rangle = 1$, so that we can reparameterize Eq. (A.3) as,

$$\begin{aligned} \langle I(a, b) \rangle &= \sum_{k=0}^N \frac{\left\langle \Delta_k^{\text{Tay}}(\langle \tau_n^k \rangle) \right\rangle}{P_{\text{acc}}(k)} \\ \left\langle \Delta_k^{\text{Tay}}(\langle \tau_n^k \rangle) \right\rangle &= g^{(k)}(-\tau_t) \frac{\langle \tau_n^k \rangle}{k!}. \end{aligned} \quad (\text{A.5})$$

In the context of classical transmittance estimation, our PCume estimator (6.78) effectively chose $P_{\text{acc}}(k) \propto \langle \tau_n^k \rangle / k!$ while our PCMF (6.80) estimator effectively chose $P_{\text{acc}}(k) \propto \tau_t^k / k!$. Both of these decisions are naturally made in relation to the individual summands which are encountered while estimating the Taylor series *on-the-fly*.

If we define the values of $P_{\text{acc}}(k)$ and sample N as a preprocess *before* estimating the Taylor series, thus making N *independent* of any evaluation of $\langle I(a, b) \rangle$, then we can freely combine N estimators for the price of *one additional* integral estimate. To show how this is possible, let us define the product of $N + 1$ independent null optical depth estimates as,

$$\langle \bar{\tau}_n^{N+1} \rangle := \frac{\mu_n(x_1)}{p(x_1)} \cdot \frac{\mu_n(x_2)}{p(x_2)} \cdot \dots \cdot \frac{\mu_n(x_{N+1})}{p(x_{N+1})}. \quad (\text{A.6})$$

We then define the sub-product of N integral estimates by removing the k th estimate from Eq. (A.6) as,

$$\langle \bar{\tau}_n^{N+1}(k) \rangle := \begin{cases} \langle \bar{\tau}_n^{N+1} \rangle & k > M \\ \frac{\langle \bar{\tau}_n^{N+1} \rangle}{\mu_n(x_k)/p(x_k)} & k \leq N + 1. \end{cases} \quad (\text{A.7})$$

Using Eq. (A.7), we can construct $N + 1$ different evaluations of $\langle I(a, b) \rangle$,

$$\begin{aligned} \langle I(a, b) \rangle &= \left(\sum_{k=0}^N \frac{\langle \Delta_k^{\text{Tay}} (\langle \bar{\tau}_n^k(1) \rangle) \rangle}{P_{\text{acc}}(k)} \right) + \left(\sum_{k=0}^N \frac{\langle \Delta_k^{\text{Tay}} (\langle \bar{\tau}_n^k(2) \rangle) \rangle}{P_{\text{acc}}(k)} \right) + \dots + \left(\sum_{k=0}^N \frac{\langle \Delta_k^{\text{Tay}} (\langle \bar{\tau}_n^k(N+1) \rangle) \rangle}{P_{\text{acc}}(k)} \right) \\ &= \frac{1}{N+1} \sum_{i=1}^{N+1} \left(\sum_{k=0}^N \frac{\langle \Delta_k^{\text{Tay}} (\langle \bar{\tau}_n^k(i) \rangle) \rangle}{P_{\text{acc}}(k)} \right) = \frac{1}{N+1} \sum_{k=1}^{N+1} \langle I(a, b) \rangle_k \end{aligned} \quad (\text{A.8})$$

which can be averaged together to form a lower variance estimator by only requiring one additional estimate for the null optical depth. We refer to this variance reduction technique as pre-processed Russian Roulette. For brevity, from now on we will insert the definition of Δ_k^{Tay} into Eq. (A.8),

$$\langle I(a, b) \rangle := \sum_{i=1}^{N+1} \left(\sum_{k=0}^N \frac{g^{(k)}(\alpha) \cdot \langle \bar{\tau}_n^k(i) \rangle}{P_{\text{acc}}(k) \cdot k!} \right) \quad (\text{A.9})$$

The drawback. To employ pre-processed Russian Roulette in practice, one has to make sure that $P_{\text{acc}}(k)$ does not converge too quickly, otherwise the variance of $\langle I(a, b) \rangle$ can become infinite. While contribution-based on-the-fly Russian Roulette avoids this problem, pre-processed Russian Roulette can easily perform poorly if too aggressive of $P_{\text{acc}}(k)$ is employed. The next variance reduction technique can alleviate this drawback when it is applicable.

A.2 Unbiased estimates as expansion points

Let us restate Eq. (A.6) as,

$$\langle \bar{\tau}_n^{N+1} \rangle := \left(\alpha - \frac{\mu_r(\mathbf{x}_1)}{p(\mathbf{x}_1)} \right) \cdot \left(\alpha - \frac{\mu_r(\mathbf{x}_2)}{p(\mathbf{x}_2)} \right) \cdot \dots \cdot \left(\alpha - \frac{\mu_r(\mathbf{x}_{N+1})}{p(\mathbf{x}_{N+1})} \right), \quad (\text{A.10})$$

by explicitly taking the difference between the expansion point ($\alpha = -\tau_t$) with an estimate for the real optical depth². We then restate the sub-product which removes the k th integral estimate (A.7) accordingly as,

$$\langle \bar{\tau}_n^{N+1}(k, \alpha) \rangle := \begin{cases} \langle \bar{\tau}_n^{N+1} \rangle & k > N+1 \\ \frac{\langle \bar{\tau}_n^{N+1} \rangle}{\alpha - \mu_r(\mathbf{x}_k)/p(\mathbf{x}_k)} & k \leq N+1, \end{cases} \quad (\text{A.11})$$

so that we can reparameterize Eq. (A.8) to be,

$$\langle I(a, b) \rangle := \sum_{i=1}^{N+1} \left(\sum_{k=0}^N \frac{g^{(k)}(\alpha) \cdot \langle \bar{\tau}_n^{N+1}(i, \alpha) \rangle}{P_{\text{acc}}(k) \cdot k!} \right) \quad (\text{A.12})$$

where the expansion point $\alpha = -\tau_t$ is explicitly accounted for³.

2: Each term in Eq. (A.10) should actually be $\left(-\frac{\mu_r(\mathbf{x}_1)}{p(\mathbf{x}_1)} - (-\alpha) \right)$ by the definition of a Taylor expansion. We chose to prematurely simplify each term for brevity.

3: Note that $\langle \bar{\tau}_n^k(i) \rangle = \langle \bar{\tau}_n^k(i, -\tau_t) \rangle$.

Prior work [104, 121] has shown that the ideal expansion point should be $\alpha = -\tau_r$, since plugging that into Eq. (A.1) would simplify to the true transmittance while every $\Delta_k^{\text{Tay}}(a, b) = 0$ for all $k > 0$. By employing an unbiased estimate of the real optical depth as the expansion point ($\alpha = -\langle \tau_r \rangle$), the expected value of every term in the Taylor series expansion except for the 0th term will be near-zero, allowing for the application of very aggressive Russian Roulette termination.

Additionally, by choosing to leave out the i th integral evaluation we only evaluate the product of N integral evaluations in Eq. (A.12). Meaning, we can use the i th integral evaluation as our expansion point for iteration i , resulting in,

$$\langle I(a, b) \rangle := \sum_{i=1}^{N+1} \left(\sum_{k=0}^N \frac{g^{(k)} \left(-\frac{\mu_r(\mathbf{x}_i)}{p(\mathbf{x}_i)} \right) \cdot \langle \bar{\tau}_n^{N+1} \left(i, -\frac{\mu_r(\mathbf{x}_i)}{p(\mathbf{x}_i)} \right) \rangle}{P_{\text{acc}}(k) \cdot k!} \right). \quad (\text{A.13})$$

The drawback. Unfortunately, one of the conditions for the Taylor series to be applicable for a given g , is that the Taylor series has to be *convergent* in expectation. If an $\alpha = -\langle \tau_r \rangle$ is sampled such that it lies *outside* the interval of convergence for a given Taylor series expansion, then employing unbiased estimates as expansion points will result in *biased estimators*⁴. Kettunen et al.'s original estimator for classical exponential transmittance did not discuss this drawback since the exponential function has an infinite interval of convergence.

[104]: Novák et al. (2014), 'Residual Ratio Tracking for Estimating Attenuation in Participating Media'
[121]: Kettunen et al. (2021), 'An Unbiased Ray-Marching Transmittance Estimator'

4: As an example of a function with a narrow interval of convergence consider a reciprocal, $\frac{1}{\tau_r}$ where τ_r is near-zero (see Sec. 8.1.1).

A.3 Raymarched integral estimates

Up until this point, we have only employed one-sample Monte Carlo estimates for evaluating the real optical depth. We can instead replace each optical depth estimate with a uniformly jittered raymarch (3.36),

$$\langle \tau_r \rangle := \sum_{j=0}^{n-1} \mu_r(\mathbf{x}_{j,s+\xi}); \quad s = \frac{b-a}{n}, \quad (\text{A.14})$$

which uses high-density correlated samples while still remaining unbiased. Denoting $\langle \tau_r \rangle_k$ as the k th independent raymarch, we redefine the product of $N + 1$ null optical depth estimates (A.10) as⁵,

$$\langle \bar{\tau}_n^{N+1} \rangle := (\alpha - \langle \tau_r \rangle_1) \cdot (\alpha - \langle \tau_r \rangle_2) \cdot \dots \cdot (\alpha - \langle \tau_r \rangle_{N+1}), \quad (\text{A.15})$$

which we derive the product of N estimates by dividing out the k th raymarch,

$$\langle \bar{\tau}_n^{N+1}(k, \alpha) \rangle := \begin{cases} \langle \bar{\tau}_n^{N+1} \rangle & k > N + 1 \\ \frac{\langle \bar{\tau}_n^{N+1} \rangle}{\alpha - \langle \tau_r \rangle_k} & k \leq N + 1. \end{cases} \quad (\text{A.16})$$

5: Again, each term in Eq. (A.15) should actually be $(-\langle \tau_r \rangle_k - (-\alpha))$ by the definition of a Taylor expansion. We chose to prematurely simplify each term for brevity.

This results in the estimator,

$$\langle I(a, b) \rangle := \frac{1}{N+1} \sum_{i=1}^{N+1} \left(\sum_{k=0}^N \frac{g^{(k)}(-\langle \tau_r \rangle_i) \cdot \langle \bar{\tau}_n^{N+1}(i, -\langle \tau_r \rangle_i) \rangle}{P_{\text{acc}}(k) \cdot k!} \right). \quad (\text{A.17})$$

A.4 Symmetric Means

By generally expanding a power series estimator,

$$\sum_{k=1}^N \frac{X^k}{k!} = \frac{\langle X \rangle_1}{1!} + \frac{\langle X \rangle_1 \langle X \rangle_2}{2!} + \frac{\langle X \rangle_1 \langle X \rangle_2 \langle X \rangle_3}{3!} + \dots \quad (\text{A.18})$$

it becomes apparent that the result of the first integral estimate ($\langle X \rangle_1$) impacts every subsequent term in the series while the last integral estimate ($\langle X \rangle_N$) only impacts the last term. We can use correlations to significantly improve the performance of a power series estimator by making the simple insight that the k th term is estimating the k th power of X . So, we can instead replace the product of the first k estimates with the k th-order *symmetric mean* which is the average of all k th-order products. For example, the $k = 2$ symmetric mean would be,

$$m_2 := \frac{\langle X \rangle_1 \langle X \rangle_2 + \dots + \langle X \rangle_1 \langle X \rangle_N + \langle X \rangle_2 \langle X \rangle_3 + \dots + \langle X \rangle_{N-1} \langle X \rangle_N}{\binom{N}{2}}. \quad (\text{A.19})$$

Let us denote $m_k(-\langle \tau_r \rangle_i)$ to be the symmetric mean, where $\langle X \rangle_k = \langle \tau_r \rangle_i - \langle \tau_r \rangle_k$. Thus, replacing the product of N null optical depth estimates in Eq. (A.17) with their symmetric means (A.19),

$$\langle I(a, b) \rangle := \frac{1}{N+1} \sum_{i=1}^{N+1} \left(\sum_{k=0}^N \frac{g^{(k)}(-\langle \tau_r \rangle_i) \cdot m_k(-\langle \tau_r \rangle_i)}{P_{\text{acc}}(k) \cdot k!} \right). \quad (\text{A.20})$$

results in our final generalized Taylor series based raymarching estimator⁶. We refer the interested reader to Kettunen et al.'s original work for an efficient algorithm to compute the k th ordered symmetric mean [121].

6: We still assume the i th raymarched estimate is excluded from the set used to compute the symmetric mean.

[121]: Kettunen et al. (2021), 'An Unbiased Ray-Marching Transmittance Estimator'

Non-exponential Volterra and VRE

B

B.1 Generalized null-Volterra

One of our consistent formulations for classical transmittance (see Chapter 6) was derived by inserting the definition of exponential transmittance (6.5) into an IVP (6.21), which simplified to the Volterra integral formulation (6.19). For convenience, we restate the two base IVPs for transmittance as follows,

$$\begin{aligned} T_r(x_a, x_b) &:= T_r(x_a, x_a) + \int_a^b \frac{d}{dt} [T_r(x_a, x_t)] dt \\ &= T_r(x_b, x_b) - \int_a^b \frac{d}{dt} [T_r(x_t, x_b)] dt. \end{aligned} \quad (\text{B.1})$$

General base Volterra. Given a general non-linear transmittance function ($g(F) = T_r(x_a, x_b)$), we can easily derive a general base Volterra formulation for any valid transmittance model,

$$\begin{aligned} T_r(x_a, x_b) &:= T_r(x_b, x_b) - \int_a^b \frac{d}{dt} [T_r(x_t, x_b)] \frac{T_r(x_a, x_b)}{T_r(x_t, x_b)} dt \\ &= 1 - \int_a^b \frac{\frac{d}{dt} [T_r(x_t, x_b)]}{T_r(x_t, x_b)} T_r(x_t, x_b) dt, \end{aligned} \quad (\text{B.2})$$

through the clever application of control variates. The quantity,

$$\frac{\frac{d}{dt} [T_r(x_t, x_b)]}{T_r(x_t, x_b)} = \frac{d}{dt} [\ln(T_r(x_t, x_b))], \quad (\text{B.3})$$

is known as the *logarithmic derivative* of the transmittance function, and it represents a proportional rate of change between the transmittance and its derivative¹. By rewriting Eq. (B.2) in terms of the first IVP (B.1),

$$\begin{aligned} T_r(x_a, x_b) &= 1 + \int_a^b \frac{\frac{d}{dt} [T_r(x_a, x_t)]}{T_r(x_a, x_t)} T_r(x_a, x_t) dt, \\ &= 1 + \int_a^b -\frac{\frac{d}{dt} [-T_r(x_a, x_t)]}{T_r(x_a, x_t)} T_r(x_a, x_t) dt, \\ &= 1 - \int_a^b \frac{\rho_f^{(r)}(x_t)}{T_r(x_a, x_t)} T_r(x_a, x_t) dt, \end{aligned} \quad (\text{B.4})$$

we see that the logarithmic derivative simplifies to the negative ratio between the real transmittance and its corresponding free-flight distribution.

1: For the problem of classical exponential transmittance, this proportional rate of change simplifies to $\mu_r(x_t)$.

Negative logarithmic derivatives of transmittance functions. The construction of the transport function in Bitterli et al.'s [4] non-exponential radiative transfer framework, guarantees that we can simplify Eq. (B.3) instead of separately evaluating a free-flight distribution and a transmittance. For the case of the classical exponential transmittance function (6.5),

$$\frac{\rho_f^{(r)}(\mathbf{x}_t)}{T_r(\mathbf{x}_a, \mathbf{x}_t)} = \mu_r(\mathbf{x}_t). \quad (\text{B.5})$$

For the case of pink-noise transmittance (4.64),

$$\frac{\rho_f^{(r)}(\mathbf{x}_t)}{T_r(\mathbf{x}_a, \mathbf{x}_t)} = \frac{\mu_r(\mathbf{x}_t)}{C^2} \left(\frac{1}{1 + C^2 \int_a^t \mu_r(\mathbf{x}_s) ds} \right). \quad (\text{B.6})$$

General multiplicative rule. The null-collision Volterra formulation for classical exponential transmittance (6.25) was derived by initially applying control variates to the definition for exponential transmittance,

$$T_r(\mathbf{x}_a, \mathbf{x}_b) = e^{-\tau_r} = e^{-\tau_r - \tau_t + \tau_t} = e^{-\tau_t} e^{\tau_t - \tau_r}. \quad (\text{B.7})$$

Unfortunately the same multiplicative property for general non-linear functions,

$$g(-\tau_r - \tau_t + \tau_t) \neq g(-\tau_t)g(\tau_t - \tau_r) \quad (\text{B.8})$$

does not hold at first glance. However, by making the realization that Eq. (B.7) can be reformulated as,

$$e^{-\tau_t} e^{\tau_t - \tau_r} = e^{-\tau_r} \frac{e^{-\tau_t}}{e^{-\tau_t}}, \quad (\text{B.9})$$

we can derive an equivalent multiplicative property for any general non-linear function as,

$$g(-\tau_r) = \frac{g(-\tau_t)}{g(-\tau_t)} g(-\tau_r) = g(-\tau_t) \frac{g(-\tau_r)}{g(-\tau_t)}, \quad (\text{B.10})$$

where we define the generalized *combined* transmittance to be,

$$T_t(\mathbf{x}_a, \mathbf{x}_b) := g(-\tau_t). \quad (\text{B.11})$$

We choose to define Eq. (B.11) with respect to $-\tau_t$ since both the Davis model and exponential transmittance employ negative optical depths in their exponents, however, this is for convenience and is not a restriction on general transmittance functions.

[4]: Bitterli et al. (2018), 'A Radiative Transfer Framework for Non-Exponential Media'

Deriving the general null-Volterra. Given the multiplicative property for general non-linear functions of expectations (B.10) and our definition for the generalized combined transmittance (B.11), we can now derive the

null-collision Volterra formulation for general transmittance functions as,

$$\begin{aligned}
T_r(x_a, x_b) &= T_t(x_a, x_b) \frac{T_r(x_a, x_b)}{T_t(x_a, x_b)} \\
&= T_t(x_a, x_b) \left(1 - \int_a^b \frac{d}{dt} \left[\frac{T_r(x_t, x_b)}{T_t(x_t, x_b)} \right] dt \right) \\
&= T_t(x_a, x_b) - T_t(x_a, x_b) \int_a^b \frac{d}{dt} \left[\frac{T_r(x_t, x_b)}{T_t(x_t, x_b)} \right] dt \\
&= T_t(x_a, x_b) - T_t(x_a, x_b) \int_a^b \frac{d}{dt} \left[\frac{T_r(x_t, x_b)}{T_t(x_t, x_b)} \right] dt \\
&= T_t(x_a, x_b) - T_t(x_a, x_b) \int_a^b \frac{T_t(x_t, x_b) \frac{d}{dt} [T_r(x_t, x_b)] - T_r(x_t, x_b) \frac{d}{dt} [T_t(x_t, x_b)]}{(T_t(x_t, x_b))^2} dt \\
&= T_t(x_a, x_b) - \int_a^b \frac{T_t(x_a, x_b)}{(T_t(x_t, x_b))^2} \left(T_t(x_t, x_b) \frac{d}{dt} [T_r(x_t, x_b)] - T_r(x_t, x_b) \frac{d}{dt} [T_t(x_t, x_b)] \right) dt \\
&= T_t(x_a, x_b) - \int_a^b \frac{T_t(x_a, x_b)}{(T_t(x_t, x_b))^2} \left(T_t(x_t, x_b) \frac{d}{dt} [T_r(x_t, x_b)] \frac{T_r(x_t, x_b)}{T_r(x_t, x_b)} - T_r(x_t, x_b) \frac{d}{dt} [T_t(x_t, x_b)] \right) dt \\
&= T_t(x_a, x_b) - \int_a^b \frac{T_t(x_a, x_b)}{(T_t(x_t, x_b))^2} \left(T_t(x_t, x_b) \frac{\frac{d}{dt} [T_r(x_t, x_b)]}{T_r(x_t, x_b)} - \frac{d}{dt} [T_t(x_t, x_b)] \right) T_r(x_t, x_b) dt \\
&= T_t(x_a, x_b) - \int_a^b \frac{T_t(x_a, x_b)}{T_t(x_t, x_b)} \left(\frac{T_t(x_t, x_b) \frac{d}{dt} [T_r(x_t, x_b)]}{T_t(x_t, x_b)} - \frac{\frac{d}{dt} [T_t(x_t, x_b)]}{T_t(x_t, x_b)} \right) T_r(x_t, x_b) dt \\
&= T_t(x_a, x_b) - \int_a^b \frac{T_t(x_a, x_b)}{T_t(x_t, x_b)} \left(\frac{\frac{d}{dt} [T_r(x_t, x_b)]}{T_r(x_t, x_b)} - \frac{\frac{d}{dt} [T_t(x_t, x_b)]}{T_t(x_t, x_b)} \right) T_r(x_t, x_b) dt \\
&= T_t(x_a, x_b) + \int_a^b \frac{T_t(x_a, x_b)}{T_t(x_t, x_b)} \left(\frac{\frac{d}{dt} [T_t(x_t, x_b)]}{T_t(x_t, x_b)} - \frac{\frac{d}{dt} [T_r(x_t, x_b)]}{T_r(x_t, x_b)} \right) T_r(x_t, x_b) dt.
\end{aligned} \tag{B.12}$$

We then introduce the combined free-flight distribution resulting in,

$$\begin{aligned}
T_r(x_a, x_b) &= T_t(x_a, x_b) + \int_a^b \frac{\rho_f^{(t)}(x_t)}{\rho_f^{(t)}(x_t)} \frac{T_t(x_a, x_b)}{T_t(x_t, x_b)} \left(\frac{\frac{d}{dt} [T_t(x_t, x_b)]}{T_t(x_t, x_b)} - \frac{\frac{d}{dt} [T_r(x_t, x_b)]}{T_r(x_t, x_b)} \right) T_r(x_t, x_b) dt \\
&= T_t(x_a, x_b) + \int_a^b \rho_f^{(t)}(x_t) \frac{T_t(x_a, x_b)}{\rho_f^{(t)}(x_t) T_t(x_t, x_b)} \left(\frac{\frac{d}{dt} [T_t(x_t, x_b)]}{T_t(x_t, x_b)} - \frac{\frac{d}{dt} [T_r(x_t, x_b)]}{T_r(x_t, x_b)} \right) T_r(x_t, x_b) dt.
\end{aligned} \tag{B.13}$$

To simplify the notation in Eq. (B.13), we choose to introduce two terms.

We denote the first term the *survival ratio*,

$$S(x_a, x_t, x_b) := \frac{T_t(x_a, x_b)}{\rho_f^{(t)}(x_t) T_t(x_t, x_b)} \tag{B.14}$$

since it represents the ratio between the probability of importance sampling a distance $t > b$ on this free-flight sample, and the probability of importance sampling $t > b$ on the next-event after sampling x_t . We refer to the second

term as the *null term*,

$$N(x_t, x_b) := \frac{\frac{d}{dt} [T_t(x_t, x_b)]}{T_t(x_t, x_b)} - \frac{\frac{d}{dt} [T_r(x_t, x_b)]}{T_r(x_t, x_b)}, \quad (\text{B.15})$$

since it is the difference between the logarithmic derivatives of the combined medium and real medium. Plugging Eq. (B.14) and Eq. (B.15) into Eq. (B.13),

$$T_r(x_a, x_b) := T_t(x_a, x_b) + \int_a^b \rho_f^{(t)}(x_t) S(x_a, x_t, x_b) N(x_t, x_b) T_r(x_t, x_b) dt. \quad (\text{B.16})$$

gives us our *general null-collision Volterra formulation* for both classical and non-classical media.

Exponential media. For the case of classical volumetric transport,

$$S(x_a, x_t, x_b) = \frac{1}{\mu_t}, \quad (\text{B.17})$$

$$N(x_t, x_b) = \mu_n(x_t). \quad (\text{B.18})$$

By multiplying Eq. (B.17) into $\rho_f^{(t)}(x_t)$ in Eq. (B.16) we arrive at the transmittance form of the classical null-Volterra (second to last equation in Eq. (6.25)), and by multiplying Eq. (B.17) into Eq. (B.18) we arrive at the free-flight form (6.25).

Pink-noise fractal media. For the case of pink noise transmittance between two surface scattering events (4.64),

$$S(x_a, x_t, x_b) = \frac{C^2}{\mu_t} \left(1 + C^2 \int_a^t \mu_s ds \right) \left(\frac{1 + C^2 \int_a^b \mu_t dt}{\left(1 + C^2 \int_a^t \mu_s ds \right) \left(1 + C^2 \int_t^b \mu_s ds \right)} \right)^{-\frac{1}{C^2}} \quad (\text{B.19})$$

$$N(x_t, x_b) = \frac{1}{C^2} \left(\frac{\mu_t}{1 + C^2 \int_t^b \mu_s ds} - \frac{\mu_r(x_t)}{1 + C^2 \int_t^b \mu_r(x_s) ds} \right). \quad (\text{B.20})$$

B.2 Non-exponential VRE

Bitterli et al. [4] introduced a path integral formulation for non-exponential volumetric transport, however, they did not provide a non-exponential volumetric rendering equation, so we will formulate one in this section.

[4]: Bitterli et al. (2018), 'A Radiative Transfer Framework for Non-Exponential Media'

The non-exponential path integral (see Sec. 4.4) differs from the classical path integral (see Sec. 4.2.4) in that the non-exponential case incorporates a transport function (4.60),

$$\tilde{T}_r(x, y) := \frac{\bar{T}(x, y)}{\Sigma(y)} = \begin{cases} ff(x, y), & \text{if } x \in \mathbb{A} \text{ & } y \in \mathbb{A}, \\ pf(x, y), & \text{if } x \in \mathbb{A} \text{ & } y \in \mathbb{V}, \\ pf(x, y), & \text{if } x \in \mathbb{V} \text{ & } y \in \mathbb{A}, \\ pp(x, y), & \text{if } x \in \mathbb{V} \text{ & } y \in \mathbb{V}, \end{cases} \quad (\text{B.21})$$

that contains three different transmittance functions instead of just the real transmittance. These three transmittances are employed depending on whether the next and previous (real) scattering events occur within a volume or along a surface. To convert the classical VRE (4.16) into one which supports non-exponential transport, we have to directly account for the classifications of the real vertices.

Let us denote the parameter λ such that it takes the following values,

$$\lambda := \begin{cases} 0, & \text{if } x \in \mathbb{A} \text{ & } y \in \mathbb{A}, \\ 1, & \text{if } x \in \mathbb{A} \text{ & } y \in \mathbb{V}, \\ 1, & \text{if } x \in \mathbb{V} \text{ & } y \in \mathbb{A}, \\ 2, & \text{if } x \in \mathbb{V} \text{ & } y \in \mathbb{V}, \end{cases} \quad (\text{B.22})$$

depending on where the vertices (x, y) are located. Given this variable, we can alter Bitterli et al.'s [4] transport function,

$$T_r(x, y, \lambda) := \begin{cases} ff(x, y), & \text{if } \lambda = 0, \\ pf(x, y), & \text{if } \lambda = 1, \\ pp(x, y), & \text{if } \lambda = 2. \end{cases} \quad (\text{B.23})$$

into one which determines the transmittance function based on an *integer* value (λ). In fact, that integer value exactly represents the number of vertices which exist within participating media. Since the various transmittance functions are inter-related through differentiation and integration (4.60), λ also represents the order of the derivative for each transmittance function.

Given the classical VRE (4.18),

$$\begin{aligned} L_o(x, \vec{\omega}_o) := & \int_a^b T_r(x_a, x_t) \mu_a(x_t) L_e(x_t, \vec{\omega}_o) dt \\ & + \int_a^b T_r(x_a, x_t) \mu_s(x_t) L_m(x_t, \vec{\omega}_o) dt \\ & + T_r(x_a, x_b) L_s(x_d, \vec{\omega}_o), \end{aligned} \quad (\text{B.24})$$

we can reparameterize the out-going, in-scattered, and surface radiances using λ^2 . Thus, we can formulate a general non-exponential VRE as,

[4]: Bitterli et al. (2018), 'A Radiative Transfer Framework for Non-Exponential Media'

2: We assume that emissive radiance does not depend on λ .

$$\begin{aligned}
L_o(\mathbf{x}, \vec{\omega}_o, \lambda) &:= \int_a^b T_r(\mathbf{x}_a, \mathbf{x}_t, \lambda+1) \mu_a(\mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) dt \\
&\quad + \int_a^b T_r(\mathbf{x}_a, \mathbf{x}_t, \lambda+1) \mu_s(\mathbf{x}_t) L_m(\mathbf{x}_t, \vec{\omega}_o, 1) dt \\
&\quad + T_r(\mathbf{x}_a, \mathbf{x}_b, \lambda) L_s(\mathbf{x}_d, \vec{\omega}_o, 0),
\end{aligned} \tag{B.25}$$

where recursing to evaluate surface scattering always sets $\lambda = 0$ since \mathbf{x}_b lies along a surface, while recursing to evaluate in-scattered radiance sets $\lambda = 1$ since \mathbf{x}_t exists within participating media. We can additionally reformulate Eq. (B.25) in terms of free-flight distributions as,

$$\begin{aligned}
L_o(\mathbf{x}, \vec{\omega}_o, \lambda) &:= \int_a^b \rho_f^{(r)}(\mathbf{x}_t, \lambda+1) \alpha_a(\mathbf{x}_t) L_e(\mathbf{x}_t, \vec{\omega}_o) dt \\
&\quad + \int_a^b \rho_f^{(r)}(\mathbf{x}_t, \lambda+1) \alpha_s(\mathbf{x}_t) L_m(\mathbf{x}_t, \vec{\omega}_o, 1) dt \\
&\quad + T_r(\mathbf{x}_a, \mathbf{x}_b, \lambda) L_s(\mathbf{x}_d, \vec{\omega}_o, 0),
\end{aligned} \tag{B.26}$$

where,

$$\rho_f^{(r)}(\mathbf{x}_t, \lambda) := \begin{cases} \mu_r(\mathbf{x}_t) \text{pf}(\mathbf{x}_a, \mathbf{x}_t), & \text{if } \lambda = 1, \\ \mu_r(\mathbf{x}_t) \text{pp}(\mathbf{x}_a, \mathbf{x}_t), & \text{if } \lambda = 2. \end{cases} \tag{B.27}$$

Make note that while for the vast majority of this dissertation we have considered $\alpha_*(\mathbf{x}) = \frac{\mu_*(\mathbf{x})}{\mu_t}$, in Eq. (B.26) we are using $\alpha_*(\mathbf{x}) := \frac{\mu_*(\mathbf{x})}{\mu_r(\mathbf{x})}$.

B.3 Null non-exponential VRE discussion

Given the non-exponential VRE (B.25) and our general null-Volterra formulation (B.16), one might consider deriving a null formulation for the non-exponential VRE using the same process we applied for the classical case (see Sec. 6.2.4). Unfortunately, the resulting formulation does not simplify nicely into a convenient recursive null term. The reason for this is because the dependence that both the null-term (B.15) and survival term (B.14) have on the future and past histories prohibits the mathematical trick of flipping the order of integration (6.54) to move all "null" terms dependent on μ_t out of the internal integrand. This inhibits one from encapsulating everything into a simple null term. However, this does not prohibit employing our general null-collision Volterra formulation (B.16) to formulate unbiased estimators for any $T_r(\mathbf{x}, \mathbf{y}, \lambda)$.

In practice, we have found our telescoping-based raymarching algorithm to be more generally useful than our null-Volterra due to its simplicity. However, unbiased raymarching has to be employed to estimate *every* $T_r(\mathbf{x}, \mathbf{y}, \lambda)$ call while our null-Volterra (B.16) has the potential to be more efficient since it can be employed to devise free-flight distance routines which perfectly importance sample proportionally to $T_r(\mathbf{x}, \mathbf{y}, \lambda)$. We leave the analysis of specific transmittance models to future work.

Bibliography

Here is the complete list of references in the order by which they were cited.

- [1] Subrahmanyan Chandrasekhar. *Radiative Transfer*. Dover Books on Advanced Mathematics. NY: Dover Publications, 1960 (cited on pages 1, 34, 35, 39, 83).
- [2] Anthony B. Davis et al. 'Horizontal Structure of Marine Boundary Layer Clouds from Centimeter to Kilometer Scales'. In: *Journal of Geophysical Research: Atmospheres* 104.D6 (1999), pp. 6123–6144. doi: [10/fd2bcz](https://doi.org/10/fd2bcz) (cited on pages 1, 51).
- [3] Alexander B. Kostinski and Raymond A. Shaw. 'Scale-Dependent Droplet Clustering in Turbulent Clouds'. In: *Journal of Fluid Mechanics* 434 (May 2001), pp. 389–398. doi: [10/dx8c5r](https://doi.org/10/dx8c5r) (cited on page 1).
- [4] Benedikt Bitterli et al. 'A Radiative Transfer Framework for Non-Exponential Media'. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 37.6 (Nov. 2018), 225:1–225:17. doi: [10/gfz2cm](https://doi.org/10/gfz2cm) (cited on pages 1, 34, 51, 52, 55, 76, 77, 84, 102, 175, 178, 180, 198, 200, 201).
- [5] Oliver James et al. 'Gravitational Lensing by Spinning Black Holes in Astrophysics, and in the Movie Interstellar'. In: *Classical and Quantum Gravity* 32.6 (Feb. 2015), p. 065001. doi: [10/gdvj4r](https://doi.org/10/gdvj4r) (cited on pages 1, 5).
- [6] Michal Možík et al. 'Bi-Directional Polarised Light Transport'. In: *Proceedings of EGSR (Experimental Ideas & Implementations)*. Ed. by Elmar Eisemann and Eugene Fiume. The Eurographics Association, 2016. doi: [10/gfzsrx](https://doi.org/10/gfzsrx) (cited on page 5).
- [7] Julio Marco. 'Transient Light Transport in Participating Media'. Ph.D. Thesis. Universidad de Zaragoza, 2013 (cited on page 5).
- [8] Shinyoung Yi et al. 'Differentiable Transient Rendering'. In: *ACM Transactions on Graphics (Proc. SIGGRAPH Asia 2021)* 40.6 (2021) (cited on page 5).
- [9] David S. Immel, Michael F. Cohen, and Donald P. Greenberg. 'A Radiosity Method for Non-Diffuse Environments'. In: *Computer Graphics (Proceedings of SIGGRAPH)* 20.4 (Aug. 1986), pp. 133–142. doi: [10/dmj9t](https://doi.org/10/dmj9t) (cited on pages 5, 13).
- [10] J. T. Kajiya and T. L. Kay. 'Rendering Fur with Three Dimensional Textures'. In: *Computer Graphics (Proceedings of SIGGRAPH)* 23.3 (July 1989), pp. 271–280. doi: [10/c92hfh](https://doi.org/10/c92hfh) (cited on page 5).
- [11] Eric Veach. 'Robust Monte Carlo Methods for Light Transport Simulation'. Ph.D. Thesis. Stanford University, Dec. 1997 (cited on pages 5, 15).

- [12] Wenzel Jakob. ‘Light Transport on Path-Space Manifolds.’ PhD thesis. Cornell University, 2013 (cited on page 6).
- [13] Fred E. Nicodemus. ‘Radiance’. In: *American Journal of Physics* 31 (1963), pp. 368–377 (cited on page 7).
- [14] Johann Heinrich Lambert. *Photometria: Sive de Mensura et Gradibus Luminis, Colorum et Umbrae.* 1760 (cited on page 8).
- [15] Ingo Wald et al. ‘Embree: A Kernel Framework for Efficient CPU Ray Tracing’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 33.4 (2014). doi: [10/gfzwck](https://doi.org/10/gfzwck) (cited on page 10).
- [16] John C. Hart. ‘Sphere Tracing: A Geometric Method for the Anti-aliased Ray Tracing of Implicit Surfaces’. In: *The Visual Computer* 12.10 (Dec. 1996), pp. 527–545. doi: [10/b3q2p6](https://doi.org/10/b3q2p6) (cited on page 10).
- [17] H. von Helmholtz and J.P.C. Southall. *Helmholtz’s Treatise on Physiological Optics*. The Optical Society of America, 1924 (cited on pages 11, 37).
- [18] Matt Pharr, Wenzel Jakob, and Greg Humphreys. *Physically Based Rendering: From Theory to Implementation*. 3rd. Cambridge, MA: Morgan Kaufmann, 2016 (cited on pages 11, 30, 58, 60, 71, 111, 117, 133, 147).
- [19] Fred E Nicodemus. *Self-study manual on optical radiation measurements ::Part I-Concepts, Chapters 1 to 3.* en. Jan. 1976. doi: <https://doi.org/10.6028/NBS.TN.910-1> (cited on page 12).
- [20] James T. Kajiya. ‘The Rendering Equation’. In: *Computer Graphics (Proceedings of SIGGRAPH)* 20.4 (Aug. 1986), pp. 143–150. doi: [10/cvf53j](https://doi.org/10/cvf53j) (cited on pages 13, 59).
- [21] Alexander Grothendieck. ‘La théorie de Fredholm’. fr. In: *Bulletin de la Société Mathématique de France* 84 (1956), pp. 319–384. doi: [10.24033/bsmf.1476](https://doi.org/10.24033/bsmf.1476) (cited on page 13).
- [22] James Richard Arvo. ‘Analytic Methods for Simulated Light Transport’. Ph.D. Thesis. New Haven, Connecticut: Yale University, Dec. 1995 (cited on page 18).
- [23] Charles M. Grinstead and J. Laurie Snell. *Introduction to Probability*. AMS, 2003 (cited on page 18).
- [24] J. L. W. V. Jensen. ‘Sur les fonctions convexes et les inégalités entre les valeurs moyennes’. In: *Acta Mathematica* 30 (1906), pp. 175–193. doi: [10/cm24qd](https://doi.org/10/cm24qd) (cited on page 21).
- [25] O. Ibe. *Fundamentals of Applied Probability and Random Processes*. Elsevier Science, 2014 (cited on page 25).
- [26] Russel E. Caflisch. ‘Monte Carlo and quasi-Monte Carlo methods’. In: *Acta Numerica* 7 (1998), pp. 1–49. doi: [10.1017/S0962492900002804](https://doi.org/10.1017/S0962492900002804) (cited on pages 27, 31, 166).
- [27] Alexander Keller. ‘Instant Radiosity’. In: *Annual Conference Series (Proceedings of SIGGRAPH)*. ACM Press, Aug. 1997, pp. 49–56. doi: [10/fqch2z](https://doi.org/10/fqch2z) (cited on page 27).

- [28] Samuli Laine et al. ‘Incremental Instant Radiosity for Real-Time Indirect Illumination’. In: *Rendering Techniques (Proceedings of the Eurographics Symposium on Rendering)*. Eurographics Association, June 2007, pp. 277–286 (cited on page 27).
- [29] Miloš Hašan et al. ‘Virtual Spherical Lights for Many-Light Rendering of Glossy Scenes’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 28.5* (2009), 143:1–143:6. doi: [10/bb9r45](https://doi.org/10/bb9r45) (cited on page 27).
- [30] Thomas Müller, Markus Gross, and Jan Novák. ‘Practical Path Guiding for Efficient Light-Transport Simulation’. In: *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering) 36.4* (June 2017), pp. 91–100. doi: [10/gbnvrs](https://doi.org/10/gbnvrs) (cited on page 29).
- [31] Shilin Zhu et al. ‘Photon-Driven Neural Reconstruction for Path Guiding’. In: *ACM Trans. Graph.* 41.1 (2021). doi: [10.1145/3476828](https://doi.org/10.1145/3476828) (cited on page 29).
- [32] Alexander Rath et al. ‘Variance-Aware Path Guiding’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 39.4* (July 2020). doi: [10/gg8xdb](https://doi.org/10/gg8xdb) (cited on page 29).
- [33] Eric Veach and Leonidas J. Guibas. ‘Optimally Combining Sampling Techniques for Monte Carlo Rendering’. In: *Annual Conference Series (Proceedings of SIGGRAPH)*. Vol. 29. ACM Press, Aug. 1995, pp. 419–428. doi: [10/d7b6n4](https://doi.org/10/d7b6n4) (cited on pages 29, 61, 126).
- [34] Rex West et al. ‘Continuous Multiple Importance Sampling’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 39.4* (July 2020). doi: [10/gg8xcr](https://doi.org/10/gg8xcr) (cited on page 30).
- [35] Justin F. Talbot, David Cline, and Parris Egbert. ‘Importance Resampling for Global Illumination’. In: *Rendering Techniques (Proceedings of the Eurographics Symposium on Rendering)*. Eurographics Association, June 2005, pp. 139–146. doi: [10/gfzsm2](https://doi.org/10/gfzsm2) (cited on page 30).
- [36] Min-Te Chao. ‘A General Purpose Unequal Probability Sampling Plan’. In: *Biometrika* 69.3 (Dec. 1982), pp. 653–656. doi: [10/fd87zs](https://doi.org/10/fd87zs) (cited on page 30).
- [37] Benedikt Bitterli et al. ‘Spatiotemporal Reservoir Resampling for Real-Time Ray Tracing with Dynamic Direct Lighting’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 39.4* (July 2020). doi: [10/gg8xc7](https://doi.org/10/gg8xc7) (cited on page 30).
- [38] Ravi Ramamoorthi et al. ‘A Theory of Monte Carlo Visibility Sampling’. In: *ACM Transactions on Graphics* 31.5 (Sept. 2012), 121:1–121:16. doi: [10/gbbrnz](https://doi.org/10/gbbrnz) (cited on page 31).
- [39] Mark Pauly, Thomas Kollig, and Alexander Keller. ‘Metropolis Light Transport for Participating Media’. In: *Rendering Techniques (Proceedings of the Eurographics Workshop on Rendering)*. Vienna: Springer-Verlag, 2000, pp. 11–22. doi: [10/gfzm93](https://doi.org/10/gfzm93) (cited on page 31).

- [40] Michael D. McKay, Richard J. Beckman, and William Jay Conover. ‘A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code’. In: *Technometrics* 21.2 (May 1979), pp. 239–245. doi: [10/gfzncd](https://doi.org/10/gfzncd) (cited on page 31).
- [41] Peter Shirley. ‘Physically Based Lighting Calculations for Computer Graphics’. Ph.D. Thesis. University of Illinois, Urbana–Champaign, Nov. 1990 (cited on page 31).
- [42] Changyaw Wang and Kelvin Sung. ‘Multi-Stage N-Rooks Sampling Method’. In: *Journal of Graphics Tools* 4.1 (1999), pp. 39–47. doi: [10/gfzqh7](https://doi.org/10/gfzqh7) (cited on page 31).
- [43] Kenneth Chiu, Peter Shirley, and Changyaw Wang. ‘Multi-Jittered Sampling’. In: *Graphics Gems IV*. Ed. by Paul S. Heckbert. San Diego, CA, USA: Academic Press, 1994, pp. 370–374 (cited on page 31).
- [44] Art B. Owen. ‘Orthogonal Arrays for Computer Experiments, Integration and Visualization’. In: *Statistica Sinica* 2.2 (1992), pp. 439–452 (cited on page 31).
- [45] Wojciech Jarosz et al. ‘Orthogonal Array Sampling for Monte Carlo Rendering’. In: *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering)* 38.4 (July 2019), pp. 135–147. doi: [10/gf6rx5](https://doi.org/10/gf6rx5) (cited on page 31).
- [46] Paul Glasserman. ‘Monte Carlo Methods in Financial Engineering’. In: vol. 53. Stochastic Modelling and Applied Probability. Springer New York, 2003. Chap. 4, pp. 185–204. doi: [bf7w](https://doi.org/bf7w) (cited on page 32).
- [47] Baptiste Nicolet et al. ‘Recursive Control Variates for Inverse Rendering’. In: *Transactions on Graphics (Proceedings of SIGGRAPH)* 42.4 (Aug. 2023). doi: [10.1145/3592139](https://doi.org/10.1145/3592139) (cited on page 32).
- [48] Michael B. Giles. ‘Multilevel Monte Carlo Path Simulation’. In: *Operations Research* 56.3 (June 2008), pp. 607–617. doi: [10/fmrpf](https://doi.org/10/fmrpf) (cited on pages 32, 159).
- [49] Devang Sinha and Siddhartha P. Chakrabarty. *Multilevel Monte Carlo and its Applications in Financial Engineering*. 2022 (cited on page 32).
- [50] Michael B. Giles. ‘Multilevel Monte Carlo methods’. In: *Acta Numerica* 24 (2015), pp. 259–328. doi: [10.1017/S096249291500001X](https://doi.org/10.1017/S096249291500001X) (cited on page 32).
- [51] Bailey Miller, Iliyan Georgiev, and Wojciech Jarosz. ‘A Null-Scattering Path Integral Formulation of Light Transport’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 38.4 (July 2019), 44:1–44:13. doi: [10/gf6rzb](https://doi.org/10/gf6rzb) (cited on pages 34, 47, 49, 73, 75, 102, 126).
- [52] Adrian Jarabo, Carlos Aliaga, and Diego Gutierrez. ‘A Radiative Transfer Framework for Spatially-Correlated Materials’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 37.4 (July 2018), 83:1–83:13. doi: [10/gd52qq](https://doi.org/10/gd52qq) (cited on pages 34, 175).
- [53] Ken Museth. ‘VDB: High-Resolution Sparse Volumes with Dynamic Topology’. In: *ACM Transactions on Graphics* 32.3 (July 2013), 27:1–27:22. doi: [10/gfzq7s](https://doi.org/10/gfzq7s) (cited on page 36).

- [54] Magnus Wrenninge et al. 'Volumetric Methods in Visual Effects'. In: *ACM SIGGRAPH Courses*. July 2010 (cited on page 36).
- [55] Louis George Henyey and Jesse Leonard Greenstein. 'Diffuse Radiation in the Galaxy'. In: *Astrophysical Journal* 93 (1941), pp. 70–83. doi: [10/bb4ffv](https://doi.org/10/bb4ffv) (cited on page 37).
- [56] Philippe Blasi, Bertrand Le Saec, and Christophe Schlick. 'A Rendering Algorithm for Discrete Volume Density Objects'. In: *Computer Graphics Forum (Proceedings of Eurographics)* 12.3 (Sept. 1993), pp. 201–210. doi: [10/c27hjz](https://doi.org/10/c27hjz) (cited on page 37).
- [57] John William Strutt Lord Rayleigh. 'On the Scattering of Light by Small Particles'. In: *Philosophical Magazine* 61 (1871), pp. 447–454 (cited on page 37).
- [58] Ludwig Lorenz. 'Lysbevaegelsen i og uden for en af plane Lysbo lger belyst Kugle'. In: *Det Kongelige Danske Videnskabernes Selskabs Skrifter (trykt utg.)*: *Naturvidenskabelig og Mathematiske Afdeling* 6 (1890), pp. 2–62 (cited on page 37).
- [59] Gustav Mie. 'Beiträge zur Optik trüber Medien, speziell kolloidaler Metallösungen'. In: *Annalen der Physik* 330.3 (Jan. 1908), pp. 377–445. doi: [10/frvnpq](https://doi.org/10/frvnpq) (cited on page 37).
- [60] Eugene d'Eon. *A Hitchhiker's Guide to Multiple Scattering*. Self-published, 2016 (cited on page 37).
- [61] Delio Vicini, Wenzel Jakob, and Anton Kaplanyan. 'A Non-Exponential Transmittance Model for Volumetric Scene Representations'. In: *Transactions on Graphics (Proceedings of SIGGRAPH)* 40.4 (Aug. 2021), 136:1–136:16. doi: [10.1145/3450626.3459815](https://doi.org/10.1145/3450626.3459815) (cited on pages 40, 52, 84).
- [62] James Richard Arvo. 'Transfer Functions in Global Illumination'. In: *Global Illumination*. ACM SIGGRAPH Courses. 1993. Chap. 1, pp. 1–28 (cited on page 43).
- [63] E. R. Woodcock et al. 'Techniques Used in the GEM Code for Monte Carlo Neutronics Calculations in Reactors and Other Systems of Complex Geometry'. In: *Applications of Computing Methods to Reactor Problems*. 1965 (cited on pages 47, 70, 73, 79).
- [64] A. B. Kostinski and A. R. Jameson. 'On the Spatial Distribution of Cloud Particles'. In: *Journal of the Atmospheric Sciences* 57.7 (Apr. 2000), pp. 901–915. doi: [10/fhs4w5](https://doi.org/10/fhs4w5) (cited on page 51).
- [65] Bryan Beresford-Smith, Derek Y. C Chan, and D. John Mitchell. 'The Electrostatic Interaction in Colloidal Systems with Low Added Electrolyte'. In: *Journal of Colloid and Interface Science* 105.1 (May 1985), pp. 216–234. doi: [10/fdzbmd](https://doi.org/10/fdzbmd) (cited on page 51).
- [66] Armando Gama Goicochea. 'A Model for the Stability of a TiO₂ Dispersion'. In: *ISRN Materials Science* 2013 (2013). doi: [10/gb67j6](https://doi.org/10/gb67j6) (cited on page 51).
- [67] David G Grier and Sven H Behrens. 'Interactions in Colloidal Suspensions'. In: *Electrostatic Effects in Soft Matter and Biophysics*. Springer-Verlag, 2001, pp. 87–116. doi: [10/bwvjd4](https://doi.org/10/bwvjd4) (cited on page 51).

- [68] Safa Jamali. 'Rheology of Colloidal Suspensions: A Computational Study'. Ph.D. Thesis. Case Western Reserve University, Aug. 2015 (cited on page 51).
- [69] Y. Rahmani. 'Micromechanics and Rheology of Hard and Soft-Sphere Colloidal Glasses'. Ph.D. Thesis. Universiteit van Amsterdam, June 2013 (cited on page 51).
- [70] B. Smith et al. 'Dusty Plasma Correlation Function Experiment'. In: *Advances in Space Research*. Scientific Exploration, Planetary Protection, Active Experiments and Dusty Plasmas 34.11 (Jan. 2004), pp. 2379–2383. doi: [10/dsg362](https://doi.org/10/dsg362) (cited on page 51).
- [71] Wen Xu et al. 'Fat Particle Structure and Stability of Food Emulsions'. In: *Journal of Food Science* 63.2 (1998), pp. 183–188. doi: [10/c9drqt](https://doi.org/10/c9drqt) (cited on page 51).
- [72] Anthony B. Davis and Alexander Marshak. 'Photon Propagation in Heterogeneous Optical Media with Spatial Correlations: Enhanced Mean-Free-Paths and Wider-than-Exponential Free-Path Distributions'. In: *Journal of Quantitative Spectroscopy and Radiative Transfer* 84.1 (2004), pp. 3–34. doi: [10/c9mc66](https://doi.org/10/c9mc66) (cited on page 51).
- [73] Edward W. Larsen and Richard Vasques. 'A Generalized Linear Boltzmann Equation for Non-Classical Particle Transport'. In: *Journal of Quantitative Spectroscopy and Radiative Transfer* 112.4 (2011), pp. 619–631. doi: [10/dfbvnn](https://doi.org/10/dfbvnn) (cited on page 52).
- [74] Anthony B. Davis and Mark B. Mineev-Weinstein. 'Radiation Propagation in Random Media: From Positive to Negative Correlations in High-Frequency Fluctuations'. In: *Journal of Quantitative Spectroscopy and Radiative Transfer* 112.4 (Mar. 2011), pp. 632–645. doi: [10/fgxnc4](https://doi.org/10/fgxnc4) (cited on pages 55, 178, 180).
- [75] Craig Kolb, Don Mitchell, and Pat Hanrahan. 'A Realistic Camera Model for Computer Graphics'. In: *Annual Conference Series (Proceedings of SIGGRAPH)*. Vol. 29. New York, NY, USA: ACM Press, 1995, pp. 317–324. doi: [10/dkgphpb](https://doi.org/10/dkgphpb) (cited on page 59).
- [76] George E. Forsythe and R. A. Leibler. 'Matrix inversion by a Monte Carlo method'. In: *Mathematics of Computation* 4 (1950), pp. 127–129 (cited on page 60).
- [77] James Richard Arvo and David Kirk. 'Particle Transport and Image Synthesis'. In: *Computer Graphics (Proceedings of SIGGRAPH)* 24.4 (Sept. 1990), pp. 63–66. doi: [10/dtp6gd](https://doi.org/10/dtp6gd) (cited on page 60).
- [78] Alexander Rath et al. 'EARS: Efficiency-Aware Russian Roulette and Splitting'. In: *ACM Trans. Graph.* 41.4 (2022). doi: [10.1145/3528223.3530168](https://doi.org/10.1145/3528223.3530168) (cited on page 61).
- [79] James Arvo. 'Backward Ray Tracing'. In: *Developments in Ray Tracing*. Vol. 12. ACM SIGGRAPH Courses. Aug. 1986, pp. 259–263 (cited on page 61).
- [80] Eric P. Lafontaine and Yves D. Willems. 'Bi-Directional Path Tracing'. In: *Proceedings of the International Conference on Computational Graphics and Visualization Techniques (Compugraphics)*. Vol. 93. Alvor, Portugal, Dec. 1993, pp. 145–153 (cited on page 61).

- [81] Eric P. Lafourture and Yves D. Willems. ‘Rendering Participating Media with Bidirectional Path Tracing’. In: *Rendering Techniques (Proceedings of the Eurographics Workshop on Rendering)*. Vienna: Springer-Verlag, June 1996, pp. 91–100. doi: [10/fzth2c](#) (cited on page 61).
- [82] Benedikt Bitterli et al. ‘Correlations and Reuse for Fast and Accurate Physically Based Light Transport’. AAI28966216. PhD thesis. USA, 2021 (cited on page 62).
- [83] Brent Burley et al. ‘The Design and Evolution of Disney’s Hyperion Renderer’. In: *ACM Transactions on Graphics* 37.3 (July 2018), 33:1–33:22. doi: [10/gfjm8w](#) (cited on pages 62, 72, 126, 128).
- [84] Henrik Wann Jensen. ‘The Photon Map in Global Illumination’. Ph.D. Thesis. Technical University of Denmark, Sept. 1996 (cited on page 64).
- [85] Henrik Wann Jensen and Niels Jorgen Christensen. ‘Photon Maps in Bidirectional Monte Carlo Ray Tracing of Complex Objects’. In: *Computers & Graphics* 19.2 (Mar. 1995), pp. 215–224. doi: [10/d9xr6q](#) (cited on page 64).
- [86] Henrik Wann Jensen. ‘Global Illumination Using Photon Maps’. In: *Rendering Techniques (Proceedings of the Eurographics Workshop on Rendering)*. Vienna: Springer-Verlag, June 1996, pp. 21–30. doi: [10/fzc6t9](#) (cited on page 64).
- [87] Henrik Wann Jensen. *Realistic Image Synthesis Using Photon Mapping*. Natick, MA, USA: AK Peters, Ltd., 2001 (cited on pages 64, 181).
- [88] Toshiya Hachisuka, Jacopo Pantaleoni, and Henrik Wann Jensen. ‘A Path Space Extension for Robust Light Transport Simulation’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 31.6 (Nov. 2012), 191:1–191:10. doi: [10/gbb6n3](#) (cited on page 64).
- [89] Toshiya Hachisuka, Shinji Ogaki, and Henrik Wann Jensen. ‘Progressive Photon Mapping’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 27.5 (Dec. 2008), 130:1–130:8. doi: [10/cn8h39](#) (cited on page 66).
- [90] Claude Knous and Matthias Zwicker. ‘Progressive Photon Mapping: A Probabilistic Approach’. In: *ACM Transactions on Graphics* 30.3 (May 2011), 25:1–25:13. doi: [10/bcw2ph](#) (cited on pages 66, 182, 183).
- [91] Iliyan Georgiev et al. ‘Light Transport Simulation with Vertex Connection and Merging’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 31.6 (Nov. 2012), 192:1–192:10. doi: [10/gbb6q7](#) (cited on pages 66, 170, 183).
- [92] Anton S. Kaplanyan and Carsten Dachsbacher. ‘Adaptive Progressive Photon Mapping’. In: *ACM Transactions on Graphics* 32.2 (Apr. 2013), 16:1–16:13. doi: [10/gbc2fq](#) (cited on page 66).
- [93] Henrik Wann Jensen and Per H. Christensen. ‘Efficient Simulation of Light Transport in Scenes with Participating Media Using Photon Maps’. In: *Annual Conference Series (Proceedings of SIGGRAPH)*. ACM Press, July 1998, pp. 311–320. doi: [10/b64p36](#) (cited on page 66).

- [94] Toshiya Hachisuka and Henrik Wann Jensen. ‘Stochastic Progressive Photon Mapping’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 28.5 (Dec. 2009), 130:1–130:8. doi: [10/d8xxn3](#) (cited on page 67).
- [95] Hao Qin et al. ‘Unbiased Photon Gathering for Light Transport Simulation’. In: *ACM Transactions on Graphics* 34.6 (Oct. 2015). doi: [10/f7wrc6](#) (cited on pages 67, 104, 156, 157, 181).
- [96] Wojciech Jarosz, Matthias Zwicker, and Henrik Wann Jensen. ‘The Beam Radiance Estimate for Volumetric Photon Mapping’. In: *Computer Graphics Forum (Proceedings of Eurographics)* 27.2 (Apr. 2008), pp. 557–566. doi: [10/bjsfsx](#) (cited on page 67).
- [97] Wojciech Jarosz et al. ‘Progressive Photon Beams’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 30.6 (Dec. 2011), 181:1–181:12. doi: [10/fn5xzj](#) (cited on pages 67, 174).
- [98] Benedikt Bitterli and Wojciech Jarosz. ‘Beyond Points and Beams: Higher-Dimensional Photon Samples for Volumetric Light Transport’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 36.4 (July 2017), 112:1–112:12. doi: [10/gfznbr](#) (cited on page 67).
- [99] Xi Deng et al. ‘Photon Surfaces for Robust, Unbiased Volumetric Density Estimation’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 38.4 (July 2019). doi: [10.1145/3306346.3323041](#) (cited on page 67).
- [100] Yang Liu, Shaojie Jiao, and Wojciech Jarosz. ‘Temporally sliced photon primitives for time-of-flight rendering’. In: *Computer Graphics Forum (Proceedings of EGSR)* 41.4 (July 2022). doi: [10/jgzq](#) (cited on page 67).
- [101] Ken H. Perlin and Eric M. Hoffert. ‘Hypertexture’. In: *Computer Graphics (Proceedings of SIGGRAPH)* 23.3 (July 1989), pp. 253–262. doi: [10/fdmsxd](#) (cited on page 69).
- [102] W. A. Coleman. ‘Mathematical Verification of a Certain Monte Carlo Sampling Technique and Applications of the Technique to Radiation Transport Problems’. In: *Nuclear Science and Engineering* 32.1 (Apr. 1968), pp. 76–81. doi: [10/gfzndg](#) (cited on pages 71, 79).
- [103] Peter Kutz et al. ‘Spectral and Decomposition Tracking for Rendering Heterogeneous Volumes’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 36.4 (July 2017), 111:1–111:16. doi: [10/gbxjxg](#) (cited on pages 71, 74, 82, 99, 126, 175).
- [104] Jan Novák, Andrew Selle, and Wojciech Jarosz. ‘Residual Ratio Tracking for Estimating Attenuation in Participating Media’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 33.6 (Nov. 2014), 179:1–179:11. doi: [10/f6r2nq](#) (cited on pages 71, 74, 80, 81, 117, 119, 126, 128, 140, 149, 175, 195).
- [105] Christopher Kulla and Marcos Fajardo. ‘Importance Sampling Techniques for Path Tracing in Participating Media’. In: *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering)* 31.4 (June 2012), pp. 1519–1528. doi: [10/f35f4k](#) (cited on pages 71, 72, 126).

- [106] Wei-Feng Wayne Huang et al. ‘Unbiased Emission and Scattering Importance Sampling For Heterogeneous Volumes’. In: *ACM SIGGRAPH 2021 Talks*. SIGGRAPH ’21. Virtual Event, USA: Association for Computing Machinery, 2021. doi: [10.1145/3450623.3464644](https://doi.org/10.1145/3450623.3464644) (cited on pages 72, 142, 145).
- [107] S. N. Cramer. ‘Application of the Fictitious Scattering Radiation Transport Model for Deep-Penetration Monte Carlo Calculations’. In: *Nuclear Science and Engineering* 65.2 (1978), pp. 237–253. doi: [10/gfqzq74](https://doi.org/10/gfqzq74) (cited on pages 74, 80, 82, 126, 127, 133).
- [108] M. Galtier et al. ‘Integral Formulation of Null-Collision Monte Carlo Algorithms’. In: *Journal of Quantitative Spectroscopy and Radiative Transfer* 125 (Aug. 2013), pp. 57–68. doi: [10/f446pg](https://doi.org/10/f446pg) (cited on pages 74, 80, 126, 133).
- [109] Jose H. Blanchet and Peter W. Glynn. ‘Unbiased Monte Carlo for Optimization and Functions of Expectations via Multi-Level Randomization’. In: *2015 Winter Simulation Conference (WSC)*. Dec. 2015, pp. 3656–3667. doi: [10/ggr94h](https://doi.org/10/ggr94h) (cited on pages 76, 176–178).
- [110] Iliyan Georgiev et al. ‘Integral Formulations of Volumetric Transmittance’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 38.6 (Nov. 2019), 154:1–154:17. doi: [10/dffn](https://doi.org/10/dffn) (cited on pages 78, 83, 108, 116, 117, 126, 149, 175).
- [111] Jan Novák et al. ‘Monte Carlo Methods for Physically Based Volume Rendering’. In: *ACM SIGGRAPH Courses*. Aug. 2018. doi: [10/c5fj](https://doi.org/10/c5fj) (cited on pages 82, 117, 119).
- [112] Zackary Misso et al. ‘Unbiased and consistent rendering using biased estimators’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 41.4 (July 2022). doi: [10/gqjn66](https://doi.org/10/gqjn66) (cited on pages 83, 123, 131, 149, 156, 175).
- [113] Magnus Wrenninge, Ryusuke Villemain, and Christophe Hery. *Path Traced Subsurface Scattering Using Anisotropic Phase Functions and Non-Exponential Free Flights*. Technical Memo 17-07. Pixar, 2017 (cited on page 84).
- [114] Eugene d’Eon and Jan Novák. ‘Zero-variance Transmittance Estimation’. In: *Eurographics Symposium on Rendering*. The Eurographics Association, June 2021 (cited on pages 91, 95).
- [115] Zackary Misso et al. ‘Progressive null-tracking for volumetric rendering’. In: *ACM SIGGRAPH Conference Papers*. July 2023. doi: [10/kmdw](https://doi.org/10/kmdw) (cited on pages 91, 126, 137).
- [116] T. E. Booth. ‘Unbiased Monte Carlo Estimation of the Reciprocal of an Integral’. In: *Nuclear Science and Engineering* 156.3 (July 2007), pp. 403–407. doi: [10/gfqzq76](https://doi.org/10/gfqzq76) (cited on pages 104, 156, 157).
- [117] S. R. Dwivedi. ‘A New Importance Biasing Scheme for Deep-Penetration Monte Carlo’. In: *Annals of Nuclear Energy* 9.7 (Jan. 1982), pp. 359–368. doi: [10/czbz26](https://doi.org/10/czbz26) (cited on page 112).

- [118] Johannes Meng, Johannes Hanika, and Carsten Dachsbacher. ‘Improving the Dwivedi Sampling Scheme’. In: *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering)* 35.4 (July 2016), pp. 37–44. doi: [10/f84272](https://doi.org/10/f84272) (cited on page 112).
- [119] Eric Veach and Leonidas J. Guibas. ‘Bidirectional Estimators for Light Transport’. In: *Photorealistic Rendering Techniques (Proceedings of the Eurographics Workshop on Rendering)*. Springer-Verlag, 1995, pp. 145–167. doi: [10/gfznbh](https://doi.org/10/gfznbh) (cited on pages 115, 116).
- [120] Jaakko Leppänen. ‘Performance of Woodcock Delta-Tracking in Lattice Physics Applications Using the Serpent Monte Carlo Reactor Physics Burnup Calculation Code’. In: *Annals of Nuclear Energy* 37.5 (May 2010), pp. 715–722. doi: [10/d3fg8p](https://doi.org/10/d3fg8p) (cited on page 119).
- [121] Markus Kettunen et al. ‘An Unbiased Ray-Marching Transmittance Estimator’. In: (Feb. 2021) (cited on pages 123, 131, 149, 159, 167, 175, 180, 185, 192, 195, 196).
- [122] Luca Fascione et al. ‘Path Tracing in Production’. In: *ACM SIGGRAPH 2018 Courses*. 2018. doi: [10.1145/3214834.3214864](https://doi.org/10.1145/3214834.3214864) (cited on page 126).
- [123] Jan Novák et al. ‘Monte Carlo Methods for Volumetric Light Transport Simulation’. In: *Computer Graphics Forum (Proceedings of Eurographics State of the Art Reports)* 37.2 (May 2018), pp. 551–576. doi: [10/gd2jqq](https://doi.org/10/gd2jqq) (cited on page 126).
- [124] Yonghao Yue et al. ‘Unbiased, Adaptive Stochastic Sampling for Rendering Inhomogeneous Participating Media’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 29.6 (Dec. 2010), 177:1–177:8. doi: [10/bp73xb](https://doi.org/10/bp73xb) (cited on page 126).
- [125] László Szirmay-Kalos, Balázs Tóth, and Milán Magdics. ‘Free Path Sampling in High Resolution Inhomogeneous Participating Media’. In: *Computer Graphics Forum* 30.1 (Mar. 2011), pp. 85–97. doi: [10/bpkdzm](https://doi.org/10/bpkdzm) (cited on pages 126, 140).
- [126] L. L. Carter, E. D. Cashwell, and W. M. Taylor. ‘Monte Carlo Sampling with Continuously Varying Cross Sections along Flight Paths’. In: *Nuclear Science and Engineering* 48.4 (Aug. 1972), pp. 403–411. doi: [10/gfznng](https://doi.org/10/gfznng) (cited on pages 126, 127, 133).
- [127] Daniel Jonsson et al. ‘Direct Transmittance Estimation in Heterogeneous Participating Media Using Approximated Taylor Expansions’. In: *IEEE Transactions on Visualization and Computer Graphics* (2020), pp. 1–1. doi: [10/gj32rc](https://doi.org/10/gj32rc) (cited on page 126).
- [128] Ben Mildenhall et al. ‘NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis’. In: *ECCV*. 2020 (cited on page 127).
- [129] Stephen Lombardi et al. *Mixture of Volumetric Primitives for Efficient Neural Rendering*. 2021 (cited on page 127).
- [130] Mohammad Shafiei et al. *Learning Neural Transmittance for Efficient Rendering of Reflectance Fields*. 2021 (cited on page 127).

- [131] Mathieu Galtier et al. ‘Radiative Transfer and Spectroscopic Databases: A Line-Sampling Monte Carlo Approach’. In: *Journal of Quantitative Spectroscopy and Radiative Transfer* 172 (Mar. 2016), pp. 83–97. doi: [10/gfzq7j](https://doi.org/10/gfzq7j) (cited on pages 127, 133).
- [132] László Szirmay-Kalos et al. ‘Unbiased Estimators to Render Procedurally Generated Inhomogeneous Participating Media’. In: *Computer Graphics Forum (Proceedings of Eurographics)* 36.2 (2017) (cited on pages 127, 133).
- [133] Florian Simon et al. ‘Line Integration for Rendering Heterogeneous Emissive Volumes’. In: *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering)* 36.4 (June 2017) (cited on pages 140, 142, 145, 151).
- [134] Sai Praveen Bangaru, Tzu-Mao Li, and Frédo Durand. ‘Unbiased Warped-Area Sampling for Differentiable Rendering’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 39.6 (Nov. 2020), 245:1–245:18. doi: [10/gj32sk](https://doi.org/10/gj32sk) (cited on pages 156, 160).
- [135] Tizian Zeltner, Iliyan Georgiev, and Wenzel Jakob. ‘Specular Manifold Sampling for Rendering High-Frequency Caustics and Glints’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 39.4 (July 2020). doi: [10/gg8xc8](https://doi.org/10/gg8xc8) (cited on pages 157, 158).
- [136] Adithya Pediredla et al. ‘Path Tracing Estimators for Refractive Radiative Transfer’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 39.6 (Nov. 2020), 241:1–241:15. doi: [10/gj4msh](https://doi.org/10/gj4msh) (cited on page 157).
- [137] Jose H. Blanchet, Nan Chen, and Peter W. Glynn. ‘Unbiased Monte Carlo Computation of Smooth Functions of Expectations via Taylor Expansions’. In: *2015 Winter Simulation Conference (WSC)*. Dec. 2015, pp. 360–367. doi: [10/ggr94p](https://doi.org/10/ggr94p) (cited on pages 158, 159, 165).
- [138] Jérémie Dauchet et al. ‘Addressing Nonlinearities in Monte Carlo’. In: *Scientific Reports* 8.1 (Dec. 2018), p. 13302. doi: [10/gd76pc](https://doi.org/10/gd76pc) (cited on page 158).
- [139] Christopher S. Withers. ‘Bias Reduction by Taylor Series’. In: *Communications in Statistics - Theory and Methods* 16.8 (Jan. 1987), pp. 2369–2383. doi: [10/bmqbg4](https://doi.org/10/bmqbg4) (cited on page 158).
- [140] Jiantao Jiao and Yanjun Han. ‘Bias Correction with Jackknife, Bootstrap, and Taylor Series’. In: *IEEE Transactions on Information Theory* 66.7 (July 2020), pp. 4392–4418. doi: [10/gj32t9](https://doi.org/10/gj32t9) (cited on page 158).
- [141] Don McLeish. ‘A General Method for Debiasing a Monte Carlo Estimator’. In: *Monte Carlo Methods and Applications* 17.4 (Jan. 2011). doi: [10/c2fpdx](https://doi.org/10/c2fpdx) (cited on pages 159, 160).
- [142] Chang-han Rhee and Peter W. Glynn. ‘A New Approach to Unbiased Estimation for SDE’s’. In: (July 2012) (cited on page 159).
- [143] S. Heinrich. ‘A Multilevel Version of the Method of Dependent Tests’. In: *Proc. of the 3rd St. Petersburg Workshop on Simulation*. St. Petersburg University Press, 1998, pp. 31–35 (cited on page 159).

- [144] A. Keller. ‘Hierarchical Monte Carlo Image Synthesis’. In: *Mathematics and Computers in Simulation* 55.1-3 (2001), pp. 79–92 (cited on page 159).
- [145] Chang-han Rhee. ‘Unbiased Estimation with Biased Samplers’. PhD thesis. Stanford University, 2013 (cited on pages 159, 164, 165).
- [146] Chang-Han Rhee and Peter W. Glynn. ‘Unbiased Estimation with Square Root Convergence for SDE Models’. In: *Operations Research* 63.5 (Oct. 2015), pp. 1026–1043. doi: [10/f7xf6m](https://doi.org/10/f7xf6m) (cited on page 159).
- [147] Sarat Babu Moka, Dirk P. Kroese, and Sandeep Juneja. ‘Unbiased Estimation of the Reciprocal Mean for Non-Negative Random Variables’. In: *2019 Winter Simulation Conference (WSC)*. Dec. 2019, pp. 404–415. doi: [10/ggr944](https://doi.org/10/ggr944) (cited on page 159).
- [148] Hiroharu Kato et al. ‘Differentiable Rendering: A Survey’. In: *CoRR* abs/2006.12057 (2020) (cited on page 171).
- [149] Merlin Nimier-David et al. ‘Mitsuba 2: A Retargetable Forward and Inverse Renderer’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 38.6 (Nov. 2019), 203:1–203:17. doi: [10/ggd8m8](https://doi.org/10/ggd8m8) (cited on page 171).
- [150] Wojciech Jarosz et al. ‘A Comprehensive Theory of Volumetric Radiance Estimation Using Photon Points and Beams’. In: *ACM Transactions on Graphics* 30.1 (Jan. 2011), 5:1–5:19. doi: [10/fcdh2f](https://doi.org/10/fcdh2f) (cited on page 174).
- [151] Jiří Vorba and Jaroslav Křivánek. ‘Adjoint-Driven Russian Roulette and Splitting in Light Transport Simulation’. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 35.4 (July 2016), 42:1–42:11. doi: [10/f89mcz](https://doi.org/10/f89mcz) (cited on page 185).
- [152] Paul Lévy. *Calcul des probabilités*. fre. Section: viii, 350 pages, 1 leaf diagrams 26 cm. Paris: Gauthier-Villars Paris, 1925 (cited on page 189).
- [153] Benoit Mandelbrot. ‘The Pareto-Lévy Law and the Distribution of Income’. In: *International Economic Review* 1.2 (1960), pp. 79–106. (Visited on 10/14/2023) (cited on page 189).