ROBUST MONTE CARLO METHODS FOR LIGHT TRANSPORT SIMULATION

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

by
Eric Veach
December 1997

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Leonidas J. Guibas
(Principal Advisor)

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Pat Hanrahan

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

James Arvo

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Art Owen

Approved for the University Committee on Graduate Studies:

Abstract

Light transport algorithms generate realistic images by simulating the emission and scattering of light in an artificial environment. Applications include lighting design, architecture, and computer animation, while related engineering disciplines include neutron transport and radiative heat transfer. The main challenge with these algorithms is the high complexity of the geometric, scattering, and illumination models that are typically used. In this dissertation, we develop new Monte Carlo techniques that greatly extend the range of input models for which light transport simulations are practical. Our contributions include new theoretical models, statistical methods, and rendering algorithms.

We start by developing a rigorous theoretical basis for *bidirectional light transport algorithms* (those that combine direct and adjoint techniques). First, we propose a linear operator formulation that does not depend on any assumptions about the physical validity of the input scene. We show how to obtain mathematically correct results using a variety of bidirectional techniques. Next we derive a different formulation, such that for any physically valid input scene, the transport operators are symmetric. This symmetry is important for both theory and implementations, and is based on a new reciprocity condition that we derive for transmissive materials. Finally, we show how light transport can be formulated as an integral over a space of paths. This framework allows new sampling and integration techniques to be applied, such as the Metropolis sampling algorithm. We also use this model to investigate the limitations of unbiased Monte Carlo methods, and to show that certain kinds of paths cannot be sampled.

Our statistical contributions include a new technique called *multiple importance sampling*, which can greatly increase the robustness of Monte Carlo integration. It uses more than one sampling technique to evaluate an integral, and then combines these samples in a way that is provably close to optimal. This leads to estimators that have low variance for a broad class of integrands. We also describe a new variance reduction technique called *efficiency-optimized Russian roulette*.

Finally, we link these ideas together to obtain new Monte Carlo light transport algorithms. *Bidirectional path tracing* uses a family of different path sampling techniques that generate some path vertices starting from a light source, and some starting from a sensor. We show that when these techniques are combined using multiple importance sampling, a large range of difficult lighting effects can be handled efficiently. The algorithm is unbiased, handles arbitrary geometry and materials, and is relatively simple to implement.

The second algorithm we describe is *Metropolis light transport*, inspired by the Metropolis sampling method from computational physics. Paths are generated by following a random walk through path space, such that the probability density of visiting each path is proportional to the contribution it makes to the ideal image. The resulting algorithm is unbiased, uses little storage, handles arbitrary geometry and materials, and can be orders of magnitude more efficient than previous unbiased approaches. It performs especially well on problems that are usually considered difficult, e.g. those involving bright indirect light, small geometric holes, or glossy surfaces. To our knowledge, this is the first application of the Metropolis method to transport problems of any kind.

Acknowledgements

I would like to thank all the people who have helped me in one way or another during my time at Stanford. First I would like to express my sincere gratitude to my advisor, Leo Guibas, who sparked my interest in computer graphics and provided support and encouragement at key times. He was always there with stimulating discussions and new suggestions, yet also willing to let things develop in unexpected ways. His ongoing confidence in me was essential to the completion of this work.

Next I would like to thank the other members of my reading committee, Pat Hanrahan, Jim Arvo, and Art Owen. Pat's insights and differing points of view were a great help in developing new ideas, while his simultaneous enthusiasm and support were also much appreciated. Discussions with Jim Arvo were always a great pleasure, and led to significant improvements in the content and exposition of this work. Jim's own Ph.D. thesis provided a superlative body of research upon which to build. Finally I would like to express my sincere thanks to Art Owen for his interest, encouragement, and perceptive comments. Art also provided important references in the Monte Carlo and survey sampling literature, and suggested the term *multiple importance sampling* for the technique described in Chapter 9.

Next I would like to thank my colleagues in the Stanford graphics lab. Marc Levoy gave freely of his encouragement and enthusiasm, invited me to lab events in the early days, and provided unerring advice on how to improve my presentations. I owe a great debt to Matt Pharr, who should really be an honorary member of my reading committee: he read the whole thing twice, giving detailed comments at every stage. I can also recommend his services as a last-minute international courier. John Owens read several chapters with great enthusiasm; his feedback was very useful and much appreciated. Phil Lacroute helped with production by providing important scripts and templates. Craig Kolb answered tricky LaTeX

questions and was a great officemate.

Peter Shirley has given me encouragement for the past few years; I suspect that he has also provided a great deal of useful feedback in the form of anonymous reviews. George Papanicolau served as the able chair of my orals committee. Jorge Stolfi and Stephen Harrison helped to get me started in graphics at the Digital Systems Research Center. Bill Kalsow answered many questions about Modula-3 (the language used for my rendering system), and in particular the SRC compiler (as I ported it to IRIX). Special thanks go to Jutta McCormick for helping me through the maze of university bureaucracy.

This research has been supported by generous grants from the National Science Foundation (CCR-9215219 and CCR-9623851), MURI contract DAAH04-96-1-0007, the Digital Systems Research Center, and the Digital External Research Program. Parts of this work have appeared in previous publications (see [Veach & Guibas 1994, Veach & Guibas 1995, Veach 1996, Veach & Guibas 1997]).

I would like to thank my parents, Hugh and Doreen Veach, for their love, support, and guidance over the years. I would also like to thank my parents-in-law, Peter and Rose Lemmer, for their love and encouragement. Finally, my deepest gratitude goes to my wife, Luanne. She supported me in times of doubt, and has enriched my life beyond measure. I dedicate this thesis to her.

To Luanne

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The goal of this dissertation is to develop robust, general-purpose algorithms for solving light transport problems. To meet our goal of generality, we concentrate on Monte Carlo methods. Currently, only Monte Carlo approaches can handle the wide range of surface geometries, reflection models, and lighting effects that occur in real environments. By a *robust* algorithm, we mean one that produces an acceptably accurate output for as wide a range of inputs as possible. In this dissertation, we make substantial progress toward these goals, by developing new theoretical models, statistical methods, and rendering algorithms. We also investigate what cannot be achieved — the inherent limitations of certain approaches to light transport.

Despite a great deal of research, current light transport methods are fairly limited in their capabilities. They are optimized for a very restricted class of input models, and typically require a huge increase in resources to handle other types of inputs. For example, they often have problems on scenes with strong indirect lighting, or scenes where most surfaces are non-diffuse. These are not pathological examples by any means, and in fact there is considerable interest in solving these cases well (e.g. in architectural applications).

For light transport algorithms to be widely used, it is important to find techniques that are less fragile. Rendering algorithms must run within acceptable time bounds on real models, yielding images that are physically plausible and visually pleasing. They must support complex geometry, materials, and illumination, since these are all important components of real environments.

In our research, we seek to develop algorithms with reasonable, predictable performance over the widest possible range of real models. Because we have chosen to focus on Monte Carlo approaches, which support complex geometry and materials with relative ease, our main interest is to develop algorithms that can handle complex *illumination* efficiently. This includes features such as glossy surfaces, concentrated indirect lighting, small geometric objects, and caustics, all of which cause problems for a wide variety of current rendering algorithms. Our goal is to find general-purpose algorithms that handle these difficult cases well, without special treatment; in other words, light transport algorithms that are robust.

In the following sections, we start with an overview of the light transport problem and why it is important. We also discuss our assumptions about the transport model (discussed in more detail in Section 1.5). After this brief introduction, we summarize the original contributions of this dissertation, and outline its organization.

In the rest of the chapter, we step back to see how these results fit into a larger context. In Section 1.4 we give a high-level view of the various types of light transport algorithms used in graphics, and explain the advantages of unbiased Monte Carlo algorithms. In Section 1.5, we consider the various phenomena that occur with real light (such as diffraction), and the reasons why these phenomena are easy or difficult to simulate. Finally, in Section 1.6 we look at problems from physics and engineering that are closely related to light transport. The viewpoints in these other fields are often quite different from one another, which has led to a variety of different solution techniques for problems that are actually quite similar.

1.1 The light transport problem

In computer graphics, the simulation of light transport is a tool that helps us to create convincing images of an artificial world. We are given a description of the environment, including the geometry and scattering properties of the surfaces. We are also given a description of the light sources, and the viewpoints from which images should be generated. Light transport algorithms then simulate the physics of this world, in order to generate realistic, accurate images.

1.1.1 Why light transport is important

One of the main goals of light transport algorithms is to increase human efficiency in the modeling of realistic virtual environments. In computer animations, for example, a great deal of effort is currently spent on designing realistic lighting. The main problem is that the algorithms used for production work (such as scan-line rendering and ray tracing) do not have the ability to simulate indirect lighting. Thus indirect illumination does not happen automatically when lights are placed: instead, it must be imitated by carefully placing additional lights. If we could find robust light transport algorithms, then the indirect lighting could be computed automatically, which would make the lighting task far easier.

Another important application of light transport is *predictive modeling*, where we wish to predict the appearance of objects before they are built. This idea has obvious uses in architecture and product design. For these applications, it is important that the results be objectively accurate, as well as visually pleasing.

Finally, better techniques for light transport in graphics may lead to better methods in physics and engineering, because light transport has a structure that is similar to radiation and particle transport problems. Section 1.6 discusses these possibilities in detail.

If robust light transport algorithms can be found, it seems inevitable that they will be widely used. This would continue a trend for computer software in general, whereby algorithms that are simpler or more powerful are eventually favored over those designed for efficiency in special cases. We feel that the benefits of accurate light transport simulations will soon outweigh their moderate computational costs.

1.1.2 Assumptions about the transport model

Light transport algorithms do not simulate the behavior of light in every detail, since this is not necessary for most applications.[1] From a graphics standpoint, physical optics is best thought of as a menu of options. For each application, we decide which optical effects are important, and choose an algorithm that can simulate them.

[1] Strictly speaking, it is not even possible, since the laws of physics are not completely known. However, the theory of light and its interaction with matter is one of the best that physics has to offer, and can predict virtually every observed phenomenon with great accuracy [Feynman 1985]. For the purposes of computer graphics, we can assume that these laws are completely understood.

In our work, we generally assume a geometric optics model. Light is emitted, scattered, and absorbed only at surfaces, and travels along straight lines between these surfaces. Thus, we do not allow participating media such as clouds or smoke, or media with a continuously varying index of refraction (e.g. heated air). We also ignore most properties of light that depend on a wave or quantum model for their explanation (e.g. diffraction or fluorescence). In particular, we ignore the possibility of interference between light beams, i.e. light is assumed to be perfectly incoherent.

In normal environments, the effects we have ignored are not very significant. Geometric optics is adequate to model almost everything we see around us, to a high degree of accuracy. For this reason, virtually all light transport algorithms in graphics are based on assumptions similar to those above. Later in this chapter, we will investigate some of the other choices that could have been made (see Sections 1.5 and 1.6).

1.2 Summary of original contributions

Our contributions fall into three areas: new theoretical models, new statistical methods, and new rendering algorithms. We give an overview of each of these areas, and then discuss our results in detail.

The first part of this dissertation investigates the theory of bidirectional light transport algorithms. We have developed light transport models that are simple, mathematically precise, and reveal the structure of the light transport problem in useful ways. In particular we have studied the relationships between different bidirectional solution techniques (e.g. those based on light and importance) under different assumptions about the physical validity of the scene model. These new light transport formulations have led directly to new insights and rendering techniques.

Statistical methods are another vital component of Monte Carlo algorithms. In the process of investigating light transport algorithms, we have developed new general-purpose methods for variance reduction. We isolated these techniques and presented them in an abstract setting, since we believe that they will be useful in other contexts.

Finally, our main contribution has been the development of robust light transport algorithms. The principal advantage of these algorithms is their ability to handle complex illumination efficiently. Because of their Monte Carlo nature, they also support complex scattering models and surface geometries. The combination of these properties allows a wide variety of realistic scenes to be rendered in a reasonable, predictable amount of time, even when there is difficult indirect illumination.

1.2.1 Bidirectional light transport theory in computer graphics

A general linear operator formulation. We present a simple light transport model based on linear operators, extending the work of Arvo [1995]. This new formulation unifies light transport, importance transport, and particle tracing, and concisely summarizes the relationships among them. We do not make any assumptions about the physical validity of the scene model, which gives our framework a richer structure than previous approaches.

New examples of non-symmetric scattering. Certain materials must be treated specially in light transport algorithms, namely those whose *bidirectional scattering distribution function* (BSDF) is not symmetric. We discuss two common examples of this that have not been previously recognized. Specifically, we show that non-symmetric scattering occurs whenever light is refracted, and also whenever shading normals are used. We derive the transformations required to handle these situations correctly in bidirectional algorithms. We also show that if these new transformations are not used, there can be substantial errors and image artifacts.

A reciprocity principle for general materials. It is well known that the reflection of light from physically valid materials is described by a symmetric BSDF. We derive a generalization of this condition that holds for arbitrary materials (i.e. for transmission as well as reflection). We establish this new reciprocity principle using the laws of thermodynamics, in particular Kirchhoff's laws and the principle of detailed balance. We also discuss the historical origins of reciprocity principles, the subtleties involved in their justification, and the conditions under which they are valid.

A self-adjoint operator formulation. Taking advantage of this new reciprocity principle, we propose the first light transport formulation in which the linear operators are *self-adjoint* (symmetric) for all physically valid scenes. We show that this simplifies both the theory and implementation of bidirectional light transport algorithms.

The path integral formulation. Usually the light transport problem is expressed in terms of integral equations or linear operators. Instead, we show how to formulate it as an integration problem over a space of paths. This viewpoint allows new solution techniques to be applied, such as multiple importance sampling, or the Metropolis sampling algorithm.

The inherent limitations of unbiased Monte Carlo methods. We show that certain kinds of transport paths cannot be generated by standard sampling techniques. This implies that the images generated by unbiased Monte Carlo algorithms (such as path tracing) can be missing certain lighting effects. We analyze the conditions under which this occurs, and propose methods for making these path sampling algorithms complete.

1.2.2 General-purpose Monte Carlo techniques

Multiple importance sampling. We describe a new technique for constructing estimators that are robust, i.e. whose variance is low for a broad class of integrands. It is based on the idea of using more than one sampling technique to evaluate an integral, where each technique is designed to sample some feature of the integrand that might otherwise lead to high variance. Our key results are on how to combine the samples: we present combination strategies that are provably close to optimal, compared to any other unbiased method. This leads to low-variance estimators that are useful in a variety of problems in graphics, including distribution ray tracing, multi-pass radiosity algorithms, and bidirectional path tracing.

Efficiency-optimized Russian roulette. Russian roulette is a technique that reduces the average cost of sampling, but increases variance. We propose a new optimization that trades off one property against the other, in order to maximize the efficiency of the resulting estimator. This is particularly useful in the context of visibility tests, where often there are many samples that only make a small contribution.

1.2.3 Robust light transport algorithms

Bidirectional path tracing. We propose a new light transport algorithm based on the idea of using a family of different sampling techniques for paths, and then combining them using multiple importance sampling. Each path is generated by connecting two independently generated subpaths, one starting from the light sources and the other starting from the eye. By varying the lengths of the light and eye subpaths, we obtain a family of different sampling techniques. We show that each technique can efficiently sample different kinds of paths, and that these paths are responsible for different lighting effects in the final image. By combining samples from all of these techniques using multiple importance sampling, a wide range of different lighting effects can be handled efficiently.

We describe the complete set of bidirectional estimators, including the important special cases where the light or eye subpath has at most one vertex. We also discuss extensions for handling ideal specular surfaces, arbitrary path lengths, and efficient visibility testing.

Metropolis light transport. We propose a new Monte Carlo approach to the light transport problem, inspired by the Metropolis sampling method in computational physics. To render an image, we generate a sequence of light transport paths by randomly mutating a single current path (e.g. a mutation might add a new vertex to the path). Each mutation is accepted or rejected with a carefully chosen probability, to ensure that paths are sampled according to the contribution they make to the desired final image. In this way we construct a random walk over the space of transport paths, such that an unbiased image can be formed by simply recording the locations of these paths on the image plane.

This algorithm is unbiased, handles general geometric and scattering models, uses little storage, and can be orders of magnitude more efficient than previous unbiased approaches. It performs especially well on problems that are usually considered difficult, e.g. those involving bright indirect light, small geometric holes, or glossy surfaces. Furthermore, it is competitive with previous unbiased algorithms even on scenes with relatively simple illumination.

The key advantage of the Metropolis approach is that the path space is explored locally, by favoring mutations that make small changes to the current path. This has several consequences. First, the average cost per sample is small (typically only one or two rays). Second, once an important path is found, the nearby paths are explored as well, thus amortizing the expense of finding such paths over many samples. Third, the mutation set is easily extended. By constructing mutations that preserve certain properties of the path (e.g. which light source is used) while changing others, we can exploit various kinds of coherence in the scene. It is often possible to handle difficult lighting problems efficiently by designing a specialized mutation in this way.

To our knowledge, this is the first application of the Metropolis algorithm to transport problems of any kind.

1.3 Thesis organization

The first two chapters consist of introductory and background material. In the rest of Chapter 1, we discuss the advantages and disadvantages of various types of light transport algorithms, we examine the range of optical phenomena that can be simulated by such algorithms, and we compare light transport to similar problems in other fields. In Chapter 2, we give an introduction to Monte Carlo integration, including a survey of the variance reduction techniques that have proven most useful in computer graphics.

The remainder of the dissertation is divided into two parts. In the first part, we describe new theoretical models for bidirectional light transport algorithms. Chapter 3 develops the concepts of radiometry and gives an introduction to the standard light transport equations. It also describes a new measure-theoretic basis for defining radiometric quantities. Chapter 4 presents a new light transport model based on linear operators. This formulation does not make any assumptions about the physical validity of the scene model. Chapter 5 investigates the situations where this model is necessary, i.e. materials whose scattering properties are not symmetric. We give both physical and non-physical examples of such materials, and we derive the techniques needed to handle these materials correctly in bidirectional algorithms.

In Chapter 6, we investigate how the scattering of light from materials is constrained by the laws of physics, and we derive a new reciprocity principle for general materials. In Chapter 7, this principle is used to construct the first light transport framework where light, importance, and particles obey the same transport equations for any physically valid scene. Finally, Chapter 8 describes the path integral framework, which forms the basis of our new light transport algorithms.

The second part of the dissertation is more practical in nature. Chapter 9 describes *multiple importance sampling*, a general tool for reducing the variance of Monte Carlo integration. In Chapter 10, we apply this tool to the path integral framework, to obtain the *bidirectional path tracing* algorithm. Finally, Chapter 11 builds upon the path integral framework in a different way, by combining it with a well-known sampling technique from computational physics to obtain the *Metropolis light transport* algorithm.

1.4 Light transport algorithms

Within the field of computer graphics, many different algorithms have been proposed for solving the light transport problem. In this dissertation, we have chosen to focus on unbiased, view-dependent, Monte Carlo algorithms. We first mention the various kinds of algorithms that have been proposed, and then discuss the choices we have made.

1.4.1 A brief history

Light transport algorithms can be roughly divided into two groups: Monte Carlo methods, and finite element methods.

Monte Carlo methods have been used for neutron transport problems since the 1950's [Albert 1956], and have been studied extensively there [Spanier & Gelbard 1969]. In graphics Monte Carlo methods arose independently, starting with Appel [1968] who computed images using random particle tracing. Whitted [1980] introduced ray tracing (the recursive evaluation of surface appearance), and also suggested the idea of randomly perturbing viewing rays. Cook et al. [1984] implemented this idea and extended it to random sampling of light sources, lenses, and time. This led to the first complete, unbiased Monte Carlo transport algorithm as proposed by Kajiya [1986], who recognized that the problem could be written as an integral equation, and could be evaluated by sampling paths. Since then, many refinements to his *path tracing* technique have been adapted from the particle transport literature [Arvo & Kirk 1990].

There has also been a great deal of work on biased Monte Carlo algorithms, which are often more efficient than path tracing. These include the *irradiance caching* algorithm of Ward et al. [1988], the *density estimation* method of [Shirley et al. 1995], and the *photon map* approach of [Jensen 1995].

Finite element methods for light transport were originally adapted from the radiative heat transfer literature. Goral et al. [1984] introduced these methods to the graphics community, where they are typically known as *radiosity algorithms*. Many improvements have been made to the basic radiosity method, including substructuring [Cohen et al. 1986], progressive refinement [Cohen et al. 1988], hierarchical basis functions [Hanrahan et al. 1991], importance-driven refinement [Smits et al. 1992], discontinuity meshing [Lischinski et al. 1992], wavelet methods [Gortler et al. 1993], and clustering [Smits et al. 1994]. Other extensions include the handling of participating media [Rushmeier & Torrance 1987], and finite element methods for non-diffuse surfaces [Immel et al. 1986, Sillion et al. 1991, Aupperle & Hanrahan 1993, Schröder & Hanrahan 1994].

Methods have also been proposed that combine features of Monte Carlo and finite element approaches. Typically, these take the form of *multi-pass methods*, which combine radiosity and ray tracing passes in order to handle more general scene models [Wallace et al. 1987, Sillion & Puech 1989, Chen et al. 1991]. Another approach is *Monte Carlo radiosity*, where the solution is represented as a linear combination of basis functions (as with finite element methods), but where the coefficients are estimated by tracing random light particles [Shirley 1990b, Pattanaik & Mudur 1993, Pattanaik & Mudur 1995].

1.4.2 Monte Carlo vs. deterministic approaches

At the most basic level, a Monte Carlo algorithm uses random numbers, while a deterministic algorithm does not. However, in practice algorithms often use a mixture of techniques, and are not easily classified. The distinction is further blurred by issues that have nothing to do with random numbers *per se*, but that are often associated with one type of algorithm or the other. We discuss some of these differences below.

First, Monte Carlo algorithms are usually more general. This is a very important issue, since the biggest source of error in light transport calculations is often the scene model itself. A key advantage of Monte Carlo approaches is that virtually any environment can be modeled accurately. With deterministic algorithms, on the other hand, there are often severe restrictions on the allowable geometry (e.g. limited to polygons) and materials (e.g. limited to ideal diffuse reflectors).[2] With these restrictions, it is difficult or impossible to model real environments. To use these methods, we must usually resort to solving a different problem, by modifying the scene model. Any claims about the solution "accuracy" under these circumstances are misleading at best.

Monte Carlo and deterministic approaches are also distinguished by how they access the scene model. Deterministic algorithms usually work with explicit representations of the scene and its properties (e.g. lists of polygons). Thus, they are strongly affected by the size and complexity of the scene representation. On the other hand, Monte Carlo algorithms are based on sampling, which means that the scene model is accessed through a small set of queries (e.g. what is the first surface point intersected by a given ray?). This interface hides the scene complexity behind a layer of abstraction, and means that rendering times are only loosely coupled to the scene representation (for example, the scene complexity may affect the time required for ray casting). In effect, Monte Carlo algorithms can sample the scene to determine the information they actually need, while most deterministic algorithms are designed to examine every detail, whether it is relevant or not.

This is an especially important issue for robustness: ideally, the performance of light transport algorithms should depend only on *what* the scene represents, rather than the details of *how* it is represented. For example, consider a scene illuminated by a square area light source. If this light source is replaced with a 10 by 10 grid of point sources, then the visual results will be nearly identical. However, the performance of many light transport algorithms will be much worse in the second case. Similarly, suppose that we replace the same source by a pair of fluorescent bulbs covered by a translucent panel. In this case the

[2] Even when deterministic algorithms support "general" surfaces and reflection models, their form is often quite limited (e.g. polynomial functions of a prespecified maximum degree). This demands an extra approximation step when modeling the scene, and often this approximation is very bad and/or expensive in some cases (e.g. for glossy surfaces).

entire scene is illuminated indirectly, which will cause problems for many algorithms. Ideally, rendering algorithms should not be sensitive to cosmetic changes of this sort. The same comments apply to geometric complexity: whether an object is represented as a thousand polygons or a million Bezier patches, we would like the rendering times to be as similar as possible. Monte Carlo algorithms at least have the potential to deal with these situations effectively, because they are based on sampling.

The distinction between Monte Carlo and deterministic methods is somewhat blurred by the fact that Monte Carlo algorithms place very weak restrictions on the "randomness" of the numbers they use (e.g. often the only requirement is that these numbers are uniformly distributed). It is usually possible to design fixed sampling patterns which satisfy the same restrictions, and this often leads to better performance (these are called *quasi-Monte Carlo methods* [Niederreiter 1992]). The principle of Monte Carlo methods is not that the samples are truly random, but that random samples *could* be used in their place.

1.4.3 View-dependent vs. view-independent algorithms

The purpose of all light transport algorithms in computer graphics is to produce images, i.e. rectangular arrays of color values, suitable for display on a monitor or printing device. A *view-independent* algorithm is one that computes an intermediate representation of the solution, from which arbitrary views can be generated very quickly. Any other algorithm is *view-dependent*, which can mean one of several things. *Importance-driven* methods compute a solution that is defined globally, but is optimized for a particular view. That is, the solution is detailed in the visible portions of the scene, but it may be very coarse elsewhere. *Multi-pass* methods compute a global solution that is valid for all views, but where the final rendering step to obtain an image is relatively slow (e.g. it requires ray tracing). Finally, *image space* methods compute an image directly from the scene model, without trying to represent the solution everywhere. This category includes Monte Carlo algorithms such as path tracing.

The distinction between view-dependent and view-independent methods raises a number of interesting issues. First, these two types of algorithms generally have different purposes. View-dependent methods are useful for animations, where the scene model can change substantially from one frame to the next. They are also the natural choice for rendering still images. On the other hand, view-independent solutions are useful for interactive applications, such as architectural walkthroughs or computer games.

One problem with “view-independent” algorithms is that they do not make any guarantees about the error for any particular view. Ideally, these algorithms would ensure that every view has a small error. Instead, error is usually measured globally (averaged over the entire scene), which implies that local errors can still be large. This means that if we render an image of a region where the view-independent solution is particularly bad, the results can be completely wrong.

Another problem with view-independent solutions is that they are often more expensive than view-dependent ones, because they compute a representation of the full solution (essentially solving for all views simultaneously). When non-diffuse materials are allowed, this can be a great deal of extra work compared to computing a single view, since the appearance of glossy surfaces changes rapidly with the viewpoint.

Even if only diffuse surfaces are allowed, view-dependent algorithms are often more efficient, since they only need to compute the portion of the solution that we are interested in. For example, if the scene model is complex, and only a small part of it is visible, then it can be much more efficient to compute an image directly. Image space algorithms have the greatest potential here, since importance-driven methods do not scale as well to complex scenes (where it can be very expensive to compute even a coarse solution over the whole domain).

The difference between view-dependent and view-independent algorithms is actually not as large as it might appear at first, since it is often possible to convert a view-dependent algorithm into view-independent one. The similarity is that both types of algorithms compute a finite set of linear measurements of the global solution. For a view-dependent algorithm, these measurements are pixel values: each pixel is defined by integrating the light falling on a small region of the image plane. This is closely related to the view-independent approach, where the solution is usually represented as a linear combination of basis functions. View-dependent algorithms can often be adapted to estimate the coefficients of these basis functions, rather than the pixel values of an image, since they are both defined as linear measurements.

1.4.4 Unbiased vs. consistent Monte Carlo algorithms

A Monte Carlo estimator computes a value $F_N(X_1, \ldots, X_N)$ that is supposed to approximate some unknown quantity Q. Typically, Q is a parameter of a known density function p, and the X_i are random samples from p. The quantity $F_N - Q$ is called the *error*, and its expected value $\beta[F_N] = E[F_N - Q]$ is called the *bias*. The estimator is *unbiased* if $\beta[F_N] = 0$ for all sample sizes N, while it is *consistent* if the error $F_N - Q$ goes to zero with probability one as N approaches infinity [Kalos & Whitlock 1986].

Intuitively, an unbiased estimator computes the correct answer, on average. A biased estimator computes the wrong answer, on average. However, if a biased estimator is also consistent, then the average error can be made arbitrarily small by increasing the sample size.

We argue that unbiased estimators are essential in order for light transport calculations to be robust. This is an important point, since many algorithms used in graphics are merely consistent.

The basic reason for preferring unbiased algorithms is that they make it far easier to estimate the error in a solution. To have any confidence in the computed results, we must have some estimate of this error. For unbiased algorithms, this simply involves computing the sample variance, since any error is guaranteed to show up as random variation among the samples. For algorithms which are merely consistent, however, we must also bound the bias. In general this is very difficult to do; we cannot estimate bias by simply drawing a few more samples. Bias leads to results that are not noisy, but are nevertheless incorrect. In graphics algorithms, this error is often noticeable visually, in the form of discontinuities, excessive blurring, or objectionable surface shading.

Unbiased algorithms are often used to generate reference images, against which other rendering algorithms can be compared. Because unbiased methods make strong guarantees about the kinds of errors that can occur, they are useful for detecting and measuring the artifacts introduced by approximations.[3] For scenes of realistic complexity, unbiased

[3] Improvements in unbiased algorithms may also lead to better approximation techniques. (Similarly, [Arvo 1995] has pointed out that better analytic methods can lead to better Monte Carlo methods.) Our viewpoint is that one should start with an unbiased algorithm, and adopt approximations only where they are clearly necessary (and their effects are well-understood).

algorithms are the only practical way to generate images that we can confidently say are correct.

Other things being equal, it is clear that we should prefer an unbiased algorithm. The conventional wisdom in graphics is that unbiased methods are "too expensive", and that an acceptable image can be achieved in less time by making approximations. However, there is little research to support this claim. While there has been a great deal of work on light transport algorithms in graphics, very little of this has been directed toward unbiased algorithms. In our view, considerably more research is necessary before we can judge their capabilities. One of the goals of this dissertation is to explore what can and cannot be achieved by unbiased methods, to help resolve these questions.

1.5 Models of light and their implications for graphics

Light transport can be studied at many levels of abstraction, ranging from two-dimensional "flatland radiosity" to quantum simulations. It is useful to have a variety of these mathematical models at hand, so that we can select the simplest model that is adequate for each task. As we will see, some optical phenomena have profound implications for algorithm design, while others can be added quite easily. It is this choice about which effects to simulate that distinguishes different classes of rendering algorithms, and that separates light transport in graphics from similar problems in other fields.

In the following sections, we summarize the important optical effects that occur in the real world, and discuss their implications for light transport algorithms. Optical phenomena are grouped according to the least-complicated optical theory that can explain them (geometric, wave, or quantum optics). Each of these theories explains different aspects of the observed behavior of light.

1.5.1 Geometric optics

Geometric optics is essentially the particle theory of light. This model can describe a wide range of optical phenomena, including emission, diffuse and specular reflection, refraction, and absorption. This covers most of what we see in everyday environments, which is why so many rendering algorithms are based on geometric optics.

However, full geometric optics is too complex for most rendering applications. In computer graphics we usually make more restrictive assumptions, to obtain simpler and faster light transport algorithms.

For example, *participating media* are often ignored. In general, light can be emitted, scattered, or absorbed in a three-dimensional medium, such as fog or gelatin. By ignoring these possibilities, all scattering is assumed to happen at surfaces (which are infinitely thin). This also implies that no energy is lost as light travels between surfaces.

In principle, it is easy to include participating media in Monte Carlo algorithms, by simply extending the ray casting procedure to sample the volume scattering and absorption along the ray [Rushmeier 1988]. The main effort required is the implementation of additional geometric primitives. Considerably more work is necessary to implement participating media with finite element approaches, since three-dimensional volumes must be meshed and subdivided, and the interaction with two-dimensional elements must be properly accounted for [Rushmeier & Torrance 1987]. With either approach, it is easier to handle media that only absorb light (no emission or scattering), since this can be handled in the same way as surface occlusion (these media block a fraction of the light traveling on a given ray, rather than all or none).

Geometric optics also allows media that have a *continuously varying refractive index*. This situation occurs when air is heated, for example, leading to shimmering “mirage” effects. In theory, this effect makes the light transport problem much more complicated, since beams of light no longer travel in straight lines between surfaces. Instead, they follow curved trajectories described by the *eiconal equation* [Born & Wolf 1986], which must be integrated in small steps to determine the path of a beam. To check for “visibility” between two points (i.e. the existence of an optical path that connects them), we must solve a difficult optimization problem. Some of these problems can be alleviated by making approximations [Stam & Languenou 1996]. However, since this effect is not important for most graphics models, it is usually just ignored.

Another common assumption is that light is *monochromatic* (i.e. that it has a single frequency). This is usually just a convenience, to simplify the description of algorithms. It is usually straightforward to deal with polychromatic light, by calculating with full spectra rather than monochromatic intensities. Operations on spectra are usually handled through a generic interface, so that different spectral representations can be substituted easily. A large number of representations have been proposed, with various tradeoffs between accuracy and expense [Hall 1989, Peercy 1993]. Sometimes, it is argued that polychromatic light can be handled by simply repeating a monochromatic algorithm at different wavelengths. However, this is rarely a good idea. Many calculations must be repeated separately at each wavelength, and any variations between the results at different wavelengths (for example, the mesh resolution or the location of random samples) can lead to objectionable color artifacts.

Similarly, *transmission* through surfaces is often disallowed. Again, this is usually just a convenience in describing algorithms. Transmission can be handled just like reflection, except that light is scattered to the opposite side of the surface. However, some care must be taken when the refractive index changes from one side to the other, since the radiance of a light beam changes according to the square of refractive index (see Chapter 5). Also, the index of refraction may depend on the frequency of the incident light, leading to the familiar rainbow effect known as *dispersion*.

For some algorithms, *ideal specular scattering* is not supported. This includes reflection by mirrors, and refraction between water and air. This is mainly a problem for algorithms that require an explicit representation of the scattering properties of a surface (e.g. as a polynomial function). In these representations, mirror-like surfaces correspond to Dirac delta distributions, which are not easily handled. If specular surfaces are supported by these algorithms at all, it is often only large, flat mirrors, which can be handled by reflecting the environment around the plane of the mirror, and treating the mirror as a window [Rushmeier 1986, Wallace et al. 1987, Rushmeier & Torrance 1990]. It is relatively easy to support specular surfaces in Monte Carlo algorithms, although this may add considerable variance to the calculations (see Chapter 8).

Finally, some algorithms support only *ideal diffuse* reflection (or transmission). A diffuse surface appears equally bright from all viewing directions; the direction in which a photon is scattered does not depend on how it arrived. This is a serious limitation, since real scenes contain a wide variety of materials, and it is often the variation in their scattering properties that makes an image look interesting or real.

The main advantage of diffuse surfaces is that their appearance depends only on position, rather than position and direction. This reduces a four-dimensional problem to a two-dimensional one, which can obviously lead to simpler algorithms. However, it is usually quite difficult to convert an algorithm the other way, from diffuse surfaces to general materials. Some algorithms handle surfaces that are a linear combination of ideal diffuse and ideal specular, but this is not at all the same as supporting general scattering functions. There are also some algorithms which appear to be general, but where in fact only diffuse surfaces are handled efficiently (e.g. other materials are handled via distribution ray tracing). Claims of generality for these algorithms are misleading, since they do not perform well unless most surfaces are diffuse. For testing generality, it is perfectly reasonable to use a scene with *no* ideal diffuse materials, since these materials do not exist in the real world.

1.5.2 Wave optics

Light can also be regarded as an electromagnetic wave [Born & Wolf 1986]. This model explains all of the phenomena handled by geometric optics, plus a few more. It is not always necessary to simulate the wave model of light to obtain wave effects. For example, polarization can be added quite easily to rendering systems based on geometric optics. In fact, the models of light transport in graphics often combine features from all three optical theories.

One effect exhibited by waves is *diffraction*, which causes light to “bend” slightly around obstacles. While diffraction is rarely noticeable at human scales, it cannot be neglected for small objects (e.g. those which are less than ten wavelengths across). This is an important issue in predicting reflection from rough surfaces, for example by simulating light transport at the microgeometry level [He et al. 1991]. However, it is difficult to incorporate diffraction into most light transport algorithms, since it violates the assumption that light travels in straight lines.

Another important wave effect is *coherence.* Coherence is a relationship between two beams of light, which measures the average correlation between their phases [Born & Wolf 1986]. So far, we have been assuming that light waves are perfectly incoherent, meaning that any two such waves have no phase correlation. The most important property of incoherent beams is that when they are superposed, their intensities add linearly (where intensity means the mean squared amplitude). This agrees with our usual intuition, e.g. two 100 watt bulbs are twice as bright as one 100 watt bulb.

When two beams are partially or fully coherent, their superposition results in *interference*. If there is positive correlation between their phases, it is called constructive interference, otherwise it is destructive interference. When two coherent beams of equal intensity are combined, the resulting intensity can be anywhere from zero to four times as great.[4] This effect is responsible for the light and dark bands in the classic "two-slit experiment" [Born & Wolf 1986].

Interference is important when modeling very small features, such as thin coatings or soap bubbles [Gondek et al. 1994]. Light is reflected back and forth inside the coating, so that the incident light wave is superposed on itself. This leads to interference, since any light beam is perfectly coherent with itself, and there is still partial coherence between two points on the beam that are several wavelengths apart. This applies even to beams from "incoherent" sources, such as incandescent light bulbs.[5]

Interference can be included in light transport algorithms by keeping track of the phases of all light beams [Gondek et al. 1994]. This requires keeping track of the optical length of the path traveled by each beam from the same source, including any coherent reflections or refractions. However, for most applications this additional expense is not justified.

Coherence is also related to *polarization*. Light is a transverse electromagnetic wave, which can be represented as point moving in a two-dimensional plane (this point is the tip of the electric vector, which is always contained in the plane perpendicular to the direction of propagation). Equivalently, we can regard light as the superposition of two independent waves, vibrating at right angles to each other. (Project the function onto two perpendicular vectors, such as the x- and y-axes.) Just as with any waves, these two waves can be partially or fully coherent, or have different amplitudes. If any of these things are true, we say the light is polarized.

[4] This is not an example of non-linear optics (discussed in the next section), since the waves themselves add linearly. However, a wave with double the amplitude corresponds to a fourfold increase in intensity.

[5] Note that the assumption of perfect incoherence in Section 1.1.2 is simply a mathematical abstraction; perfectly incoherent light does not exist in the real world.

Polarization is important for modeling materials such as water or glass, where the scattering properties depend strongly on how the incident light is polarized. Another effect that depends on polarization is *birefringence* (also known as *double refraction*) [Drude 1900]. It occurs in certain kinds of crystals, where the refractive index is different for light polarized parallel or perpendicular to the crystal surface. This has the effect of splitting an incident beam of light into two beams with opposite polarizations, which are refracted in different directions.

Polarization is quite easy to include in most light transport algorithms; the effort is similar to that of adopting a different spectral representation. There are two common representations of polarization: the Jones matrix (appropriate for monochromatic light, which is always completely polarized), and the Stokes matrix (which applies to partially polarized, perfectly incoherent light beams). The general problem of superimposing two partially coherent, partially polarized beams is more difficult; there are no simple representations in general, other than working with an explicit description of the waveforms [Perina 1985].

1.5.3 Quantum optics

Quantum physics offers the most detailed, accurate model of the behavior of light.[6] Some of these effects are not explained by the geometric or wave theories, but are still relevant to computer graphics.

One of these effects is *fluorescence*. This occurs when photons are absorbed by a molecule, and then a new photon is emitted at a different wavelength. This effect is actually quite common in the real world. For example, fluorescent dyes are used commercially to obtain brighter colors; this is why clothing often “glows in the dark” under ultraviolet lights.

Fluorescence is quite easy to add to rendering systems [Glassner 1994], by allowing energy at different wavelengths to interact (in a linear way). If light spectra are represented as vectors (with one coefficient per wavelength), then scattering from a surface can be represented as a matrix. When there is no fluorescence, this matrix is diagonal; otherwise, some

[6] Feynman [1985] has written a very readable account of the basics of this theory, and makes fascinating connections between the macroscopic and quantum behaviors of light.

of the off-diagonal entries will be nonzero.[7]

Another interesting effect is *phosphorescence* [Glassner 1994]. Here photons are absorbed, and re-emitted at a later time (usually at a different wavelength). This effect is not important for most computer graphics applications; however, there are similar problems in other fields where this kind of time-delay reaction is crucial (e.g. the decay of radioactive elements). The implementation of phosphorescence requires that the rendering algorithm integrate the incident light over time, since the current emission of a phosphorescent surface depends on its exposure in the past.

All of the effects we have described so far belong to *linear optics*. Consider an arbitrary optical system, which takes one light beam as input, and produces another light beam as output. The optical system is *linear* if the output wave is a linear function of the input wave; e.g. if we superpose two input waves, the output must be the sum of the outputs we would get if each wave were used alone. This property holds for practically every optical system.

However, with the introduction of lasers, non-linear effects have been discovered. For example, when high-intensity laser light passes through certain crystals, the light that exits the crystal is twice the frequency of the light which enters it. This is known as *frequency doubling* [Bloembergen 1996]. It does not happen with low-intensity light, so this is an example of non-linearity.[8]

There are many other effects whose explanation rests on quantum physics. For example, the photoelectric effect, or the observed spectral distribution of blackbody radiation. Lasers also depend on quantum physics for their explanation. However, these effects are irrelevant for computer graphics. We do not need to simulate blackbody radiation from first principles to include it in our scene models. Similarly, special and general relativity can be ignored for all practical purposes (e.g. the bending of light in a gravitational field).

[7] Glassner [1994] points out that for real materials, the matrix is often triangular. Photons often migrate from higher to lower energies during scattering, but rarely move in the other direction. This is why clothes do not "glow in the dark" when exposed to heat lamps.

[8] Consider a beam of light that is so intense that it heats the receiving surface, until it begins to glow. This effect is non-linear (since with a dim beam of light, the surface will not glow at all). However, this is not what is meant by non-linear optics. The surface temperature depends on the *integral* of the incident light over time (unlike the frequency doubling example). At each instant in time, the system is still linear, since the surface emission does not depend on the current intensity of the incident light.

1.6 Related problems from other fields

Light transport is similar to a variety of problems in physics and engineering. It is important to have a clear understanding of the connections between these problems, since many of the techniques used in graphics were first discovered in other areas. There is still much to learn from other scientific fields, and conversely these fields also have something to learn from computer graphics.

However, the underlying assumptions in other fields are often very different from those in graphics. This can make it difficult to transfer results from one field to another. In fact, some aspects of the light transport problem seem to be unique to computer graphics.

One important difference is the representation of the final output. In computer graphics, the final output always consists of images, and any other representations of the solution are just intermediate steps toward this goal. In physics and engineering, images are not important (except possibly as a visualization aid). Instead, the objective is to compute a set of numerical measurements, or even better, a functional representation of the solution over its entire domain. A full representation of the solution makes it easier to locate design problems (e.g. a leak through the shielding of a nuclear reactor).

Another difference is the way in which the quality of a solution is measured. In other fields, the goal is to compute results that are objectively accurate, according to standard numerical error metrics (e.g. the L_2 norm). In computer graphics, on the other hand, the ultimate error metrics are perceptual (and are thus not easy to define explicitly). Visual artifacts such as discontinuities or Mach bands are very objectionable in graphics, yet they are perfectly acceptable in heat transfer or nuclear engineering problems (as long as the numerical error is satisfactory). Because of this, popular methods in other fields are not always well-suited for graphics applications. In fact, perceptual error has been one of the main forces driving further research on light transport algorithms.

In the remainder of this section, we discuss the light transport problem as it relates to nuclear engineering, radiative heat transfer calculations, radar and acoustic wave scattering, and many-body problems.

1.6.1 Neutron transport

One of the first applications of Monte Carlo methods was the design of nuclear devices. Early Monte Carlo pioneers, such as von Neumann and Ulam, discovered techniques in this context that have now found much wider applicability [Ulam 1987]. Neutron transport problems are natural candidates for Monte Carlo methods, because of the relatively large number of dimensions involved (position, direction, energy, time), and the complexity of the interactions with atomic nuclei.

Light transport has much in common with neutron transport. They are governed by the same underlying equation (the *Boltzmann equation*), which describes the transport of virtually any kind of particles that do not interact with each other.[9] This equation is one of the central aspects of *transport theory,* which studies the transport of generic particles without regard for their physical meaning [Duderstadt & Martin 1979].

However, neutron and light transport differ substantially in emphasis. For example, the simulation of participating media is not important for most applications in computer graphics, whereas it is absolutely essential for neutron transport. Neutrons penetrate much farther into solid objects than photons, so that volume scattering (and volume emission) are the dominant effects. In fact, surface scattering and emission are often completely ignored in these simulations [Spanier & Gelbard 1969].

Another important difference is the interaction between particles at different energy levels. In graphics, fluorescence and phosphorescence are relatively insignificant effects. This means that to a good approximation, photon scattering is *elastic* (its wavelength does not change) and *instantaneous* (there is no significant delay between the arrival and departure of the photon). On the other hand, the scattering of neutrons is inelastic: they generally gain or lose some energy upon collision with a nucleus (an effect similar to fluorescence). Likewise, there is a small delay between the arrival of a neutron, and the scattering or emission of other neutrons (similar to phosphorescence). These delays can substantially affect the outcome of the calculation, and cannot be ignored.

[9] The Boltzmann equation does not model particle transport perfectly, since it is based on assumptions similar to those of geometric optics. For example, it ignores wave effects such as diffraction.

A third major difference is the existence of conservation principles. In graphics, we often rely on conservation of energy: the light scattered from a surface is no greater than the light incident upon it. With neutrons, on the other hand, it is often the objective to avoid this type of conservation. It is possible for a nuclear reaction to be critical or supercritical, in which case the number of neutrons in the environment increases quickly with time. In terms of individual collision events, a single incident neutron may cause several new neutrons to be emitted (by splitting the atomic nucleus). Other kinds of particles may be emitted as well, such as high-energy photons (gamma rays), and it is often necessary to track these particles as well.

Despite these differences, many techniques from the neutron transport literature can be adapted to computer graphics. This is usually quite easy, since light transport is a simpler problem.

There is also some interest in transport algorithms for charged particles, such as electrons. However, an important property of charged particles is that they interact with each other at a distance, by means of the electromagnetic field. Similarly, the path of a charged particle is influenced by fixed electric and magnetic fields, so that these particles follow curved trajectories (similar to photons passing through a medium with a continuously varying refractive index). These features give the transport of charged particles a considerably different flavor, and most light transport algorithms cannot easily be adapted to this purpose.

1.6.2 Heat transfer

Radiative heat transfer is also very similar to light transport. In fact, the only difference is that the photons in heat transfer have longer wavelengths (in the infrared portion of the spectrum). As with neutron transport, however, different aspects of the problem are emphasized.

First, we review the three mechanisms of heat transfer: conduction, convection, and radiation. With *conduction,* energy is exchanged between adjacent vibrating atoms, as they bump into each other. This causes a slow migration of heat away from “hot spots” (e.g. this is what causes the handle of a frying pan to become hot). With *convection*, heat is transferred by the large-scale movements of atoms (e.g. a draft of hot air). Finally, heat can be transferred by the *radiation* of photons, which carry energy almost instantaneously across large distances (e.g. the heat that is felt when standing near a campfire). It is this last mechanism that is similar to light transport.

This brings us to the first important difference from light transport, namely that radiation is only one aspect of heat transfer problem. For many applications, conduction and convection are at least as important. (One indication of this is that the *heat equation* in the applied mathematics and engineering literature often refers only to conduction [Gustafson 1987, Hughes 1987].) In theory, conduction and convection can also affect light transport calculations, if portions of the surrounding environment are so hot that they begin to glow (i.e. emit photons in the visible wavelengths). However, this definitely falls outside the traditional realm of computer graphics.

A second difference is that heat transfer problems are often non-linear. For example, the spectrum of radiation emitted by a hot surface depends on the fourth power of its temperature, and convection is also affected by temperature in complex ways. However, these non-linearities are irrelevant for our purposes, because the *radiative* aspect of heat transfer is always a linear problem. Temperature changes due to conduction, convection, and even radiation are extremely slow compared to the speed of light, so that the system is effectively in radiative equilibrium at all times.

Unlike neutron transport, most heat transfer algorithms are based on the finite element method.[10] There are several reasons for this. First, finite element methods compute a representation of the entire solution (rather than isolated measurements), which makes it easier to locate design problems. Second, a full solution also makes it easier to include the effects of conduction and convection, and to follow the evolution of the system over time. Finally, finite element methods are a standard tool in civil and mechanical engineering, so that it was natural to extend these methods to heat transfer problems.

The heat transfer literature has thus inspired finite element approaches to light transport, just as neutron transport algorithms have inspired Monte Carlo work.

[10] Technically, these are often *boundary element methods* [Siegel & Howell 1992], where the solution is represented only on the boundary of the domain rather than its interior. This is the preferred representation in the absence of convection.

1.6.3 Radar and acoustics problems

The scattering of radio waves is another problem that is similar to light transport. Radio waves are simply another part of the electromagnetic spectrum, but with much longer wavelengths than visible light or radiant heat. Consequently, the wave nature of electromagnetic radiation becomes important: effects such as diffraction and interference cannot be ignored. For this reason, the mathematical models and algorithms for these problems are based on the wave model of light, rather than geometric optics. This yields a totally different set of algorithms and insights.

Radio scattering problems arise in the design of objects that are difficult for radar systems to detect (e.g. military aircraft). Similar problems arise in the design of auditoriums and concert halls, where it is important to predict the scattering of sound waves. The wavelengths of audible sounds are comparable to the dimensions of ordinary objects (ranging approximately from one centimeter to ten meters), so that wave effects cannot be neglected.

At their most basic level, these problems involve solving the *wave equation*, a partial differential equation that describes how waves propagate with time [Strang 1986, Gustafson 1987, Zauderer 1989]. This formulation is extremely general, but for realistic problems it is also difficult and expensive to solve. The problem can be greatly simplified by assuming that all radio sources have a single frequency, and that their intensity does not change with time. This is called the *time-harmonic* version of the problem. Such a system will rapidly converge to an equilibrium state, where the intensity of the electromagnetic field at each point is a sinusoidal function of time. The amplitude and phase of the electromagnetic vibration at each point can be represented by a complex number.

Mathematically, the reduced problem is described by the *Helmholtz equation*, also known as the *reduced wave equation* [Zauderer 1989]. This is a partial differential equation, like the wave equation, except that there is no time dependence (since we are solving for an equilibrium state). Formally, this means that the Helmholtz equation is an elliptic problem, rather than a hyperbolic problem like the wave equation. Elliptic problems require an entirely different set of solution techniques than hyperbolic ones, and are generally easier to solve.

Methods for the scattering of radio and sound waves can be applied directly to the light transport problem; the restriction to a single frequency means that only monochromatic light can be handled (or that each frequency must be simulated independently). This formulation correctly handles diffraction and interference, as well as all of the phenomena handled by geometric optics. This could lead to interesting solution techniques for graphics problems where the wave nature of light is important.

1.6.4 Many-body problems

Efficient algorithms for many-body problems are an important recent influence on computer graphics. The simplest version of this problem involves a set of N particles, each with a different mass. The problem is to determine the gravitational force exerted on each particle by the others. This can be used to simulate the motion of the particles, by integrating their velocity and position over time. The problem can be extended to charged particles, and also to bodies with more complex shapes.

The obvious algorithm for this problem is to compute the $O(N^2)$ pairs of forces, and add them together to find the net force acting on each particle. However, recently several algorithms have been proposed that are far more efficient. These algorithms have complexities of $O(N \log N)$ [Barnes & Hut 1986] or even $O(N)$ [Greengard & Rokhlin 1987, Greengard 1988]. The basic idea is that distant particles can be grouped together, replacing the calculations for many individual particles with a single computation for the group. Because of the $O(1/r^2)$ falloff of gravitational and electric force, these approximations are possible without significant loss of accuracy. Particles are organized into a hierarchical data structure, so that nearby particles can be processed in small groups, while distant particles are handled in large groups.

These techniques were the inspiration for hierarchical light transport algorithms [Hanrahan et al. 1991]. It is easy to see that there is some connection; for example, the intensity of a point light source obeys the same kind of $O(1/r^2)$ falloff law as gravity does. In fact, if we simply replace point masses by point light sources, many-body algorithms can be used to efficiently compute the *fluence rate* due to these light sources at many points simultaneously. (The fluence rate at a point in space is the integral of the incident radiance over all directions, i.e. the total power per unit cross-sectional area that would be received by a tiny spherical light sensor [American National Standards Institute 1986].)

However, computing the fluence at isolated points is not particularly useful for making images. There are substantial differences between light transport and the many-body problem, such as occlusion. Gravity passes through walls, while light does not. Furthermore, the gravitational force is a function only of position, while light intensity (radiance) is a function of position and direction. (This is because a point mass creates the same gravitational force in all directions, while a point light source can radiate different amounts of light in different directions.)

These differences make light transport considerably more complex than the many-body problem, and help to explain why hierarchical algorithms in graphics have not been able to make the same accuracy and performance guarantees that are available for many-body algorithms. The results for many-body algorithms are quite impressive: solutions can be computed with any accuracies comparable to the machine's floating-point resolution, with a time complexity of $O(N)$ [Greengard 1988].[11] It is doubtful that similar results will ever be obtained for realistic light transport problems.

[11] Note that although the force-calculation component of the Greengard algorithm is $O(N)$, there is also a tree building component that can take $O(N \log N)$ time. Similarly, the complexity of the Barnes & Hut [1986] algorithm can be significantly worse than $O(N \log N)$ when the particle distribution is non-uniform [Anderson 1996].

Chapter 2

Monte Carlo Integration

This chapter gives an introduction to Monte Carlo integration. The main goals are to review some basic concepts of probability theory, to define the notation and terminology that we will be using, and to summarize the variance reduction techniques that have proven most useful in computer graphics.

Good references on Monte Carlo methods include Kalos & Whitlock [1986], Hammersley & Handscomb [1964], and Rubinstein [1981]. Sobol' [1994] is a good starting point for those with little background in probability and statistics. Spanier & Gelbard [1969] is the classic reference for Monte Carlo applications to neutron transport problems; Lewis & Miller [1984] is a good source of background information in this area. For quasi-Monte Carlo methods, see Niederreiter [1992], Beck & Chen [1987], and Kuipers & Niederreiter [1974].

2.1 A brief history

Monte Carlo methods originated at the Los Alamos National Laboratory in the early years after World War II. The first electronic computer in the United States had just been completed (the ENIAC), and the scientists at Los Alamos were considering how to use it for the design of thermonuclear weapons (the H-bomb). In late 1946 Stanislaw Ulam suggested the use of random sampling to simulate the flight paths of neutrons, and John von Neumann developed a detailed proposal in early 1947. This led to small-scale simulations whose results were indispensable in completing the project. Metropolis & Ulam [1949] published a paper in 1949 describing their ideas, which sparked to a great deal of research in the 1950's [Meyer 1956]. The name of the Monte Carlo method comes from a city in Monaco, famous for its casinos (as suggested by Nick Metropolis, another Monte Carlo pioneer).

In isolated instances, random sampling had been used much earlier to solve numerical problems [Kalos & Whitlock 1986]. For example, in 1777 the Comte de Buffon performed an experiment in which a needle was dropped many times onto a board marked with equidistant parallel lines. Letting L be the length of the needle and $d > L$ be the distance between the lines, he showed that the probability of the needle intersecting a line is

$$p \;=\; \frac{2L}{\pi d}\,.$$

Many years later, Laplace pointed out that this could be used as a crude means of estimating the value of π.

Similarly, Lord Kelvin used what we would now call a Monte Carlo method to study some aspects of the kinetic theory of gases. His random number generator consisted of drawing slips of paper out of a glass jar. The possibility of bias was a significant concern; he worried that the papers might not be mixed well enough due to static electricity. Another early Monte Carlo experimenter was Student (an alias for W. S. Gosset), who used random sampling as an aid to guessing the form of his famous t-distribution.

An excellent reference on the origins of Monte Carlo methods is the special issue of *Los Alamos Science* published in memory of Stanislaw Ulam [Ulam 1987]. The books by Kalos & Whitlock [1986] and Hammersley & Handscomb [1964] also contain brief histories, including information on the pre-war random sampling experiments described above.

2.2 Quadrature rules for numerical integration

In this section we explain why standard numerical integration techniques do not work very well on high-dimensional domains, especially when the integrand is not smooth.

Consider an integral of the form

$$I \;=\; \int_\Omega f(x)\, d\mu(x)\,, \tag{2.1}$$

where Ω is the domain of integration, $f\,:\,\Omega \to \mathbb{R}$ is a real-valued function, and μ is a measure function on Ω.[1] For now, let the domain be the s-dimensional unit hypercube,

$$\Omega \;=\; [0,1]^s\,,$$

and let the measure function be

$$d\mu(x) \;=\; dx^1 \cdots dx^s\,,$$

where x^j denotes the j-th component of the point $x = (x^1, \ldots, x^s) \in [0,1]^s$.

Integrals of this sort are often approximated using a *quadrature rule*, which is simply a sum of the form

$$\hat{I} \;=\; \sum_{i=1}^{N} w_i\, f(x_i) \tag{2.2}$$

where the weights w_i and sample locations x_i are determined in advance. Common examples of one-dimensional quadrature rules include the *Newton-Cotes rules* (i.e. the midpoint rule, the trapezoid rule, Simpson's rule, and so on), and the *Gauss-Legendre rules* (see Davis & Rabinowitz [1984] for further details). The n-point forms of these rules typically obtain a convergence rate of $O(n^{-r})$ for some integer $r \geq 1$, provided that the integrand has sufficiently many continuous derivatives. For example, the error using Simpson's rule is $O(n^{-4})$, provided that f has at least four continuous derivatives [Davis & Rabinowitz 1984].

Although these quadrature rules typically work very well for one-dimensional integrals, problems occur when extending them to higher dimensions. For example, a common approach is to use *tensor product rules* of the form

$$\hat{I} \;=\; \sum_{i_1=1}^{n} \sum_{i_2=1}^{n} \cdots \sum_{i_s=1}^{n}\; w_{i_1} w_{i_2} \cdots w_{i_s}\, f(x_{i_1}, x_{i_2}, \ldots, x_{i_s})$$

where s is the dimension, and the w_i and x_i are the weights and sample locations for a given

[1] Familiar examples of measures include length, surface area, volume, and solid angle; see Halmos [1950] for an introduction to measure theory.

one-dimensional rule. This method has the same convergence rate as the one-dimensional rule on which it is based (let this be $O(n^{-r})$), however it uses a much larger number of sample points (namely $N = n^s$). Thus in terms of the total number of samples, the convergence rate is only $O(N^{-r/s})$. This implies that the efficiency of tensor product rules diminishes rapidly with dimension, a fact that is often called the *curse of dimensionality* [Niederreiter 1992, p. 2].

The convergence rate can be increased by using a one-dimensional rule with a larger value of r, however this has two problems. First, the total number of samples $N = n^s$ can become impractical in high dimensions, since n increases linearly with r (specifically, $n \geq r/2$). For example, two-point Guass quadrature requires at least 2^s samples, while Simpson's rule requires at least 3^s samples. Second, faster convergence rates require more smoothness in the integrand. For example, if the function f has a discontinuity, then the convergence rate of any one-dimensional quadrature rule is at best $O(n^{-1})$ (assuming that the location of the discontinuity is not known in advance), so that the corresponding tensor product rule converges at a rate no better than $O(N^{-1/s})$.

Of course, not all multidimensional integration rules take the form of tensor products. However, there is an important result which limits the convergence rate of any deterministic quadrature rule, called *Bakhvalov's theorem* [Davis & Rabinowitz 1984, p. 354]. Essentially, it says that given any s-dimensional quadrature rule, there is function f with r continuous and bounded derivatives, for which the error is proportional to $N^{-r/s}$. Specifically, let C^r_M denote the set of functions $f : [0,1]^s \to \mathbb{R}$ such that

$$\left| \frac{\partial^r f}{\partial (x^1)^{a_1} \cdots \partial (x^s)^{a_s}} \right| \;\leq\; M$$

for all $a_1, \ldots, a_s$ with $\sum a_i = r$, recalling that x^j denotes the j-th coordinate of the vector x. Now consider any N-point quadrature rule

$$\hat{I}(f) \;=\; \sum_{i=1}^{N} w_i \, f(x_i)$$

where each x_i is a point in $[0,1]^s$, and suppose that we wish to approximate some integral

$$I(f) \;=\; \int_{[0,1]^s} f(x^1, \ldots, x^s) \, dx^1 \cdots dx^s \, .$$

Then according to Bakhvalov's theorem, there is a function $f \in C^r_M$ such that the error is

$$\left| \hat{I}(f) - I(f) \right| \; > \; k \cdot N^{-r/s} \,,$$

where the constant $k > 0$ depends only on M and r. Thus even if f has a bounded, continuous first derivative, no quadrature rule has an error bound better than $O(N^{-1/s})$.

2.3 A bit of probability theory

Before describing Monte Carlo integration, we review a few concepts from probability and statistics. See Pitman [1993] for an introduction to probability, and Halmos [1950] for an introduction to measure theory. Brief introductions to probability theory can also be found in the Monte Carlo references cited above.

2.3.1 Cumulative distributions and density functions

Recall that the *cumulative distribution function* of a real-valued random variable X is defined as

$$P(x) \;=\; Pr\left\{X \le x\right\} \,,$$

and that the corresponding *probability density function* is

$$p(x) \;=\; \frac{dP}{dx}(x)$$

(also known as the *density function* or *pdf*). This leads to the important relationship

$$Pr\left\{\alpha \le X \le \beta\right\} \;=\; \int_\alpha^\beta p(x)\, dx \;=\; P(\beta) - P(\alpha)\,. \tag{2.3}$$

The corresponding notions for a multidimensional random vector $(X^1, \ldots, X^s)$ are the *joint cumulative distribution function*

$$P(x^1, \ldots, x^s) \;=\; Pr\left\{X^i \le x^i \text{ for all } i = 1, \ldots, s\right\}$$

and the *joint density function*

$$p(x^1, \ldots, x^s) \;=\; \frac{\partial^s P}{\partial x^1 \cdots \partial x^s}(x^1, \ldots, x^s)\,,$$

so that we have the relationship

$$Pr\,\{x \in D\} \;=\; \int_D p(x^1, \ldots, x^s)\, dx^1 \cdots dx^s \qquad (2.4)$$

for any Lebesgue measurable subset $D \subset \mathbb{R}^s$.

More generally, for a random variable X with values in an arbitrary domain Ω, its *probability measure* (also known as a *probability distribution* or *distribution*) is a measure function P such that

$$P(D) \;=\; Pr\,\{X \in D\}$$

for any measurable set $D \subset \Omega$. In particular, a probability measure must satisfy $P(\Omega) = 1$. The corresponding density function p is defined as the *Radon-Nikodym derivative*

$$p(x) \;=\; \frac{dP}{d\mu}(x)\,,$$

which is simply the function p that satisfies

$$P(D) \;=\; \int_D p(x)\, d\mu(x)\,. \qquad (2.5)$$

Thus, the probability that $X \in D$ can be obtained by integrating $p(x)$ over the given region D. This should be compared with equations (2.3) and (2.4), which are simply special cases of the more general relationship (2.5).

Note that the density function p depends on the measure μ used to define it. We will use the notation $p = P_\mu$ to denote the density with respect to a particular measure μ, corresponding to the notation $u_x = \partial u \,/\, \partial x$ that is often used in analysis. This notation will be useful when there are several relevant measure function defined on the same domain Ω (for example, the solid angle and projected solid angle measures that will be described in Chapter 3). See Halmos [1950] for further information on measure spaces and Radon-Nikodym derivatives.

2.3.2 Expected value and variance

The *expected value* or *expectation* of a random variable $Y = f(X)$ is defined as

$$E[Y] \;=\; \int_\Omega f(x)\, p(x)\, d\mu(x)\,, \tag{2.6}$$

while its *variance* is

$$V[Y] \;=\; E\Big[(Y - E[Y])^2\Big]\,. \tag{2.7}$$

We will always assume that expected value and variance of every random variable exist (i.e. the corresponding integral is finite).

From these definitions, it is easy to see that for any constant a we have

$$\begin{aligned} E[a\,Y] &\;=\; a\,E[Y]\,, \\ V[a\,Y] &\;=\; a^2\,V[Y]\,. \end{aligned}$$

The following identity is also useful:

$$E\left[\sum_{i=1}^N Y_i\right] \;=\; \sum_{i=1}^N E[Y_i]\,,$$

which holds for any random variables $Y_1, \ldots, Y_N$. On the other hand, the following identity holds only if the variables Y_i are independent:

$$V\left[\sum_{i=1}^N Y_i\right] \;=\; \sum_{i=1}^N V[Y_i]\,.$$

Notice that from these rules, we can derive a simpler expression for the variance:

$$V[Y] \;=\; E[(Y - E[Y])^2] \;=\; E[Y^2] - E[Y]^2\,.$$

Another useful quantity is the *standard deviation* of a random variable, which is simply the square root of its variance:

$$\sigma[Y] \;=\; \sqrt{V[Y]}\,.$$

This is also known as the *RMS error*.

2.3.3 Conditional and marginal densities

Let $X \in \Omega_1$ and $Y \in \Omega_2$ be a pair of random variables, so that

$$(X, Y) \in \Omega$$

where $\Omega = \Omega_1 \times \Omega_2$. Let P be the joint probability measure of (X, Y), so that $P(D)$ represents the probability that $(X, Y) \in D$ for any measurable subset $D \subset \Omega$. Then the corresponding joint density function $p(x, y)$ satisfies

$$P(D) \;=\; \int_D p(x, y)\, d\mu_1(x)\, d\mu_2(y)\,,$$

where μ_1 and μ_2 are measures on Ω_1 and Ω_2 respectively. Hereafter we will drop the measure function notation, and simply write

$$P(D) \;=\; \int_D p(x, y)\, dx\, dy\,.$$

The *marginal density function* of X is now defined as

$$p(x) \;=\; \int_{\Omega_2} p(x, y)\, dy\,, \tag{2.8}$$

while the *conditional density function* $p(y \,|\, x)$ is defined as

$$p(y \,|\, x) \;=\; p(x, y) \,/\, p(x)\,. \tag{2.9}$$

The marginal density $p(y)$ and conditional density $p(x \mid y)$ are defined in a similar way, leading to the useful identity

$$p(x, y) \;=\; p(y \,|\, x)\, p(x) \;=\; p(x \,|\, y)\, p(y)\,.$$

Another important concept is the *conditional expectation* of a random variable $G = g(X, Y)$, defined as

$$E[G \,|\, x] \;=\; \int_{\Omega_2} g(x, y)\, p(y \,|\, x)\, dy \;=\; \frac{\int g(x, y)\, p(x, y)\, dy}{\int p(x, y)\, dy}\,. \tag{2.10}$$

We will also use the notation $E_Y[G]$ for the conditional expectation, which emphasizes the fact that Y is the random variable whose density function is being integrated.

There is a very useful expression for the variance of G in terms of its conditional expectation and variance, namely

$$V[G] \;=\; E_X V_Y G + V_X E_Y G\,. \tag{2.11}$$

In other words, $V[G]$ is the mean of the conditional variance, plus the variance of the conditional mean. To prove this identity, recall that

$$V[F] \;=\; E[F^2] - E[F]^2\,,$$

and observe that

$$\begin{aligned} E_X V_Y G + V_X E_Y G \;&=\; E_X \left\{ E_Y[G^2] - [E_Y G]^2 \right\} + E_X[E_Y G]^2 - [E_X E_Y G]^2 \\ &=\; E_X E_Y[G^2] - [E_X E_Y G]^2 \\ &=\; V[G]\,. \end{aligned}$$

We will use this identity below to analyze certain variance reduction techniques, including stratified sampling and the use of expected values.

2.4 Basic Monte Carlo integration

The idea of Monte Carlo integration is to evaluate the integral

$$I \;=\; \int_\Omega f(x)\, d\mu(x)$$

using random sampling. In its basic form, this is done by independently sampling N points $X_1, \ldots, X_N$ according to some convenient density function p, and then computing the estimate

$$F_N \;=\; \frac{1}{N} \sum_{i=1}^{N} \frac{f(X_i)}{p(X_i)}\,. \tag{2.12}$$

Here we have used the notation F_N rather than $\hat{I}$ to emphasize that the result is a random variable, and that its properties depend on how many sample points were chosen. Note that this type of estimator was first used in the survey sampling literature (for discrete rather than continuous domains), where it is known as the *Horvitz-Thompson estimator* [Horvitz

& Thompson 1952].

For example, suppose that the domain is $\Omega = [0,1]^s$ and that the samples X_i are chosen independently and uniformly at random. In this case, the estimator (2.12) reduces to

$$F_N \;=\; \frac{1}{N} \sum_{i=1}^{N} f(X_i)\,,$$

which has the same form as a quadrature rule except that the sample locations are random.

It is straightforward to show the estimator F_N gives the correct result on average. Specifically, we have

$$\begin{aligned}
E[F_N] &= E\left[\frac{1}{N} \sum_{i=1}^{N} \frac{f(X_i)}{p(X_i)}\right] \\
&= \frac{1}{N} \sum_{i=1}^{N} \int_\Omega \frac{f(x)}{p(x)}\, p(x)\, d\mu(x) \\
&= \int_\Omega f(x)\, d\mu(x) \\
&= I\,,
\end{aligned}$$

provided that $f(x)/p(x)$ is finite whenever $f(x) \neq 0$.

Advantages of Monte Carlo integration. Monte Carlo integration has the following major advantages. First, it converges at a rate of $O(N^{-1/2})$ in any dimension, regardless of the smoothness of the integrand. This makes it particularly useful in graphics, where we often need to calculate multi-dimensional integrals of discontinuous functions. The convergence rate is discussed in Section 2.4.1 below.

Second, Monte Carlo integration is simple. Only two basic operations are required, namely sampling and point evaluation. This encourages the use of object-oriented *black box* interfaces, which allow great flexibility in the design of Monte Carlo software. In the context of computer graphics, for example, it is straightforward to include effects such motion blur, depth of field, participating media, procedural surfaces, and so on.

Third, Monte Carlo is general. Again, this stems from the fact that it is based on random sampling. Sampling can be used even on domains that do not have a natural correspondence with $[0,1]^s$, and are thus not well-suited to numerical quadrature. As an example of this in graphics, we observe that the light transport problem can be naturally expressed as an integral over the space of all transport paths (Chapter 8). This domain is technically an infinite-dimensional space (which would be difficult to handle with numerical quadrature), but it is straightforward to handle with Monte Carlo.

Finally, Monte Carlo methods are better suited than quadrature methods for integrands with singularities. Importance sampling (see Section 2.5.2) can be applied to handle such integrands effectively, even in situations where there is no analytic transformation to remove the singularity (see the discussion of rejection sampling and the Metropolis method below).

In the remainder of this section, we discuss the convergence rate of Monte Carlo integration, and give a brief review of sampling techniques for random variables. We then discuss the properties of more general kinds of Monte Carlo estimators.

2.4.1 Convergence rates

To determine the convergence rate of Monte Carlo integration, we start by computing the variance of F_N. To simplify the notation let $Y_i = f(X_i)/p(X_i)$, so that

$$F_N \;=\; \frac{1}{N} \sum_{i=1}^{N} Y_i \,.$$

Also let $Y = Y_1$. We then have

$$V[Y] \;=\; E[Y^2] - E[Y]^2 \;=\; \int_\Omega \frac{f^2(x)}{p(x)}\, d\mu(x) \;-\; I^2 \,.$$

Assuming that this quantity is finite, it is easy to check that the variance of $V[F_N]$ decreases linearly with N:

$$V[F_N] \;=\; V\left[\frac{1}{N}\sum_{i=1}^{N} Y_i\right] \;=\; \frac{1}{N^2}\, V\left[\sum_{i=1}^{N} Y_i\right] \;=\; \frac{1}{N^2}\sum_{i=1}^{N} V[Y_i] \;=\; \frac{1}{N}\, V[Y] \qquad (2.13)$$

where we have used $V[a\,Y] \;=\; a^2\, V[Y]$ and the fact that the Y_i are independent samples. Thus the standard deviation is

$$\sigma[F_N] \;=\; \frac{1}{\sqrt{N}}\, \sigma Y \,,$$

which immediately shows that the RMS error converges at a rate of $O(N^{-1/2})$.

It is also possible to obtain probabilitistic bounds on the absolute error, using *Chebychev's inequality*:

$$Pr\left\{ |F - E[F]| \;\geq\; \left(\frac{V[F]}{\delta}\right)^{1/2} \right\} \;\leq\; \delta\,,$$

which holds for any random variable F such that $V[F] < \infty$. Applying this inequality to the variance (2.13), we obtain

$$Pr\left\{ |F_N - I| \;\geq\; N^{-1/2}\left(\frac{V[Y]}{\delta}\right)^{1/2} \right\} \;\leq\; \delta\,.$$

Thus for any fixed threshold δ, the absolute error decreases at the rate $O(N^{-1/2})$.

Tighter bounds on the absolute error can be obtained using the *central limit theorem*, which states that F_N converges to a normal distribution in the limit as $N \to \infty$. Specifically, it states that

$$\lim_{N\to\infty} Pr\left\{ \frac{1}{N}\sum_{i=1}^{N} Y_i - E[Y] \;\leq\; t\,\frac{\sigma[Y]}{\sqrt{N}} \right\} \;=\; \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{t} e^{-x^2/2}\,dx\,,$$

where the expression on the right is the (cumulative) normal distribution. This equation can be rearranged to give

$$Pr\left\{ |F_N - I| \;\geq\; t\,\sigma[F_N] \right\} \;=\; \sqrt{2/\pi}\int_{t}^{\infty} e^{-x^2/2}\,dx\,.$$

The integral on the right decreases very quickly with t; for example when $t = 3$ the right-hand side is approximately 0.003. Thus, there is only about a 0.3% chance that F_N will differ from its mean by more than three standard deviations, provided that N is large enough for the central limit theorem to apply.

Finally, note that Monte Carlo integration will converge even if the variance $V[Y]$ is infinite, provided that the expectation $E[Y]$ exists (although convergence will be slower). This is guaranteed by the *strong law of large numbers*, which states that

$$Pr\left\{ \lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} Y_i \;=\; E[Y] \right\} \;=\; 1\,.$$

2.4.2 Sampling random variables

There are a variety of techniques for sampling random variables, which we briefly review here. Further details can be found in the references given in the introduction.

One method is the *transformation* or *inversion* method. In one dimension, suppose that we want to sample from a density function p. Letting P be the corresponding cumulative distribution function, the inversion method consists of letting $X = P^{-1}(U)$, where U is a uniform random variable on $[0, 1]$. It is easy to verify that X has the required density p. This technique can easily be extended to several dimensions, either by computing marginal and conditional distributions and inverting each dimension separately, or more generally by deriving a transformation $x = g(u)$ with an appropriate Jacobian determinant (such that $|\det(J_g(x))|^{-1} = p(x)$, where J_g denotes the Jacobian of g).

The main advantage of the transformation technique is that it allows samples to be stratified easily, by stratifying the parameter space $[0, 1]^s$ and mapping these samples into Ω (see Section 2.6.1). Another advantage is that the technique has a fixed cost per sample, which can easily be estimated. The main disadvantage is that the density $p(x)$ must be integrated analytically, which is not always possible. It is also preferable for the cumulative distribution to have an analytic inverse, since numerical inversion is typically slower.

A second sampling technique is the *rejection method,* due to von Neumann [Ulam 1987]. The idea is to sample from some convenient density q such that

$$p(x) \ \leq \ M\, q(x)$$

for some constant M. Generally, the samples from q are generated by the transformation method. We then apply the following procedure:

```
function REJECTION-SAMPLING()
  for $i = 1$ to $\infty$
    Sample $X_i$ according to $q$.
    Sample $U_i$ uniformly on $[0, 1]$.
    if $U_i \ \leq \ p(X_i) \,/\, (M\, q(X_i))$
      then return $X_i$
```

It is easy to verify that this procedure generates a sample X whose density function is p.

The main advantage of rejection sampling is that it can be used with any density function, even those that cannot be integrated analytically. However, we still need to be able to integrate some function Mq that is an upper bound for p. Furthermore, this bound should be reasonably tight, since the average number of samples that must be taken before acceptance is M. Thus, the efficiency of rejection sampling can be very low if it is applied naively. Another disadvantage is that it is difficult to apply with stratification: the closest approximation is to stratify the domain of the random vector (X, U), but the resulting stratification is not as good as the transformation method.

A third general sampling technique is the *Metropolis method* (also known as *Markov chain Monte Carlo*), which will be described in Chapter 11. This technique is useful for sampling arbitrary densities on high-dimensional spaces, and has the advantage that the density function does not need to be normalized. The main disadvantage of the Metropolis method is that the samples it generates are not independent; in fact they are highly correlated. Thus, it is most useful when we need to generate a long sequence of samples from the given density p.

Finally, there are various techniques for sampling from specific distributions (see Rubinstein [1981]). For example, if X is the maximum of k independent uniform random variables $U_1, \ldots, U_k$, then X has the density function $p(x) = kx^{k-1}$ (where $0 \leq x \leq 1$). Such “tricks” can be used to sample many of the standard distributions in statistics, such as the normal distribution [Rubinstein 1981].

2.4.3 Estimators and their properties

So far we have only discussed one way to estimate an integral using random samples, namely the standard technique (2.12). However, there are actually a great variety of techniques available, which are encompassed by the concept of a *Monte Carlo estimator*. We review the various properties of estimators and why they are desirable.

The purpose of a Monte Carlo estimator is to approximate the value of some *quantity of interest* Q (also called the *estimand*). Normally we will define Q as the value of a given integral, although more general situations are possible (e.g. Q could be the ratio of two integrals). An *estimator* is then defined to be a function of the form

$$F_N \;=\; F_N(X_1, \ldots, X_N)\,, \tag{2.14}$$

where the X_i are random variables. A particular numerical value of F_N is called an *estimate*. Note that the X_i are not necessarily independent, and can have different distributions.

Note that there are some differences in the standard terminology for computer graphics, as compared to statistics. In statistics, the value of each X_i is called an *observation*, the vector $(X_1, \ldots, X_N)$ is called the *sample*, and N is called the *sample size*. In computer graphics, on the other hand, typically each of the individual X_i is referred to as a sample, and N is the number of samples. We will normally use the graphics conventions.

We now define a number of useful properties of Monte Carlo estimators. The quantity $F_N - Q$ is called the *error*, and its expected value is called the *bias*:

$$\beta[F_N] \;=\; E[F_N - Q]\,. \tag{2.15}$$

An estimator is called *unbiased* if $\beta[F_N] = 0$ for all sample sizes N, or in other words if

$$E[F_N] \;=\; Q \quad \text{for all } N \geq 1\,. \tag{2.16}$$

For example, the random variable

$$F_N \;=\; \frac{1}{N} \sum_{i=1}^{N} \frac{f(X_i)}{p(X_i)}$$

is an unbiased estimator of the integral $I = \int_\Omega f(x)\, d\mu(x)$ (as we saw in Section 2.4).

An estimator is called *consistent* if the error $F_N - Q$ goes to zero with probability one, or in other words if

$$Pr\left\{\lim_{N\to\infty} F_N = Q\right\} \;=\; 1\,. \tag{2.17}$$

For an estimator to be consistent, a sufficient condition is that the bias and variance both go to zero as N is increased:

$$\lim_{N\to\infty} \beta[F_N] \;=\; \lim_{N\to\infty} V[F_N] \;=\; 0\,.$$

In particular, an unbiased estimator is consistent as long as its variance decreases to zero as N goes to infinity.

The main reason for preferring unbiased estimators is that it is easier to estimate the error. Typically our goal is to minimize the *mean squared error* (MSE), defined by

$$MSE[F] \;=\; E[(F-Q)^2] \tag{2.18}$$

(where we have dropped the subscript N). In general, the mean squared error can be rewritten as

$$\begin{aligned} MSE[F] &= E[(F-Q)^2] \\ &= E[(F-E[F])^2] + 2E[F-E[F]](E[F]-Q) + (E[F]-Q)^2 \\ &= V[F] + \beta[F]^2\,, \end{aligned}$$

so that to estimate the error we must have an upper bound on the possible bias. In general, this requires additional knowledge about the estimand Q, and it is often difficult to find a suitable bound.

On the other hand, for unbiased estimators we have $E[F] = Q$, so that the mean squared error is identical to the variance:

$$MSE[F] \;=\; V[F] \;=\; E[(F-E[F])^2]\,.$$

This makes it far easier to obtain error estimates, by simply taking several independent samples. Letting $Y_1, \ldots, Y_N$ be independent samples of an unbiased estimator Y, and letting

$$F_N \;=\; \frac{1}{N}\sum_{i=1}^{N} Y_i$$

as before (which is also an unbiased estimator), then the quantity

$$\hat{V}[F_N] \;=\; \frac{1}{N-1}\left\{\left(\frac{1}{N}\sum_{i=1}^{N} Y_i^2\right) - \left(\frac{1}{N}\sum_{i=1}^{N} Y_i\right)^2\right\}$$

is an unbiased estimator of the variance $V[F_N]$ (see Kalos & Whitlock [1986]). Thus, error estimates are easy to obtain for unbiased estimators.

Notice that by taking many independent samples, the error of an unbiased estimator can be made as small as desired, since

$$V[F_N] \;=\; V[F_1] \,/\, N\,.$$

However, this will also increase the running time by a factor of N. Ideally, we would like to find estimators whose variance and running time are both small. This tradeoff is summarized by the *efficiency* of a Monte Carlo estimator:

$$\epsilon[F] \;=\; \frac{1}{V[F]\,T[F]} \qquad (2.19)$$

where $T[F]$ is the time required to evaluate F. Thus the more efficient an estimator is, the lower the variance that can be obtained in a given fixed running time.

2.5 Variance reduction I: Analytic integration

The design of efficient estimators is a fundamental goal of Monte Carlo research. A wide variety of techniques have been developed, which are often simply called *variance reduction methods*. In the following sections, we describe the variance reduction methods that have proven most useful in computer graphics.[2] These methods can be grouped into several categories, based around four main ideas:

- analytically integrating a function that is similar to the integrand;
- uniformly placing sample points across the integration domain;
- adaptively controlling the sample density based on information gathered during sampling; and
- combining samples from two or more estimators whose values are correlated.

[2] Note that some variance reduction methods are useful only for one-dimensional integrals, or only for smooth integrands (e.g. certain antithetic variates transformations [Hammersley & Handscomb 1964]). Since these situations are usually better handled by numerical quadrature, we do not discuss such methods here.

We start by discussing methods based on analytic integration. There are actually several ways to take advantage of this idea, including *the use of expected values*, *importance sampling*, and *control variates*. These are some of the most powerful and useful methods for computer graphics problems.

Note that many variance reduction methods were first proposed in the survey sampling literature, long before Monte Carlo methods were invented. For example, techniques such as stratified sampling, importance sampling, and control variates were all first used in survey sampling [Cochran 1963].

2.5.1 The use of expected values

Perhaps the most obvious way to reduce variance is to reduce the dimension of the sample space, by integrating analytically with respect to one or more variables of the domain. This idea is commonly referred to as *the use of expected values* or *reducing the dimensionality*. Specifically, it consists of replacing an estimator of the form

$$F \;=\; f(X,Y)\,/\,p(X,Y) \tag{2.20}$$

with one of the form

$$F' \;=\; f'(X)\,/\,p(X)\,, \tag{2.21}$$

where $f'(x)$ and $p(x)$ are defined by

$$\begin{aligned} f'(x) &= \int f(x,y)\,dy \\ p(x) &= \int p(x,y)\,dy\,. \end{aligned}$$

Thus, to apply this technique we must be able to integrate both f and p with respect to y. We also must be able to sample from the marginal density $p(x)$, but this can be done by simply generating (X,Y) as before, and ignoring the value of Y.

The name of this technique comes from the fact that the estimator F' is simply the conditional expected value of F:

$$F' \;=\; E_Y\left[\frac{f(X,Y)}{p(X,Y)}\right]$$

$$\begin{aligned} &= \int \frac{f(X,y)}{p(X,y)}\, p(y \,|\, X)\, dy \\ &= \int \frac{f(X,y)}{p(X,y)}\, \frac{p(X,y)}{\int p(X,y')\, dy'}\, dy \\ &= f(X) \,/\, p(X)\,. \end{aligned}$$

This makes the variance reduction easy to analyze. Recalling the identity

$$V[F] \;=\; E_X V_Y F + V_X E_Y F$$

from equation (2.11), and using the fact that $F' = E_Y F$, we immediately obtain

$$V[F] - V[F'] \;=\; E_X V_Y F\,.$$

This quantity is always non-negative, and represents the component of the variance of F that was caused by the random sampling of Y (as one might expect).

The use of expected values is the preferred variance reduction technique, as long as it is not too expensive to evaluate and sample the analytically integrated quantities. However, note that if expected values are used for only one part of a larger calculation, then variance can actually increase. Spanier & Gelbard [1969] give an example of this in the context of neutron transport problems, by comparing the variance of the *absorption estimator* (which records a sample only when a particle is absorbed) to that of the *collision estimator* (which records the expected value of absorption at each collision along a particle's path). They show that there are conditions where each of these estimators can have lower variance than the other.

2.5.2 Importance sampling

Importance sampling refers to the principle of choosing a density function p that is similar to the integrand f. It is a well-known fact that the best choice is to let $p(x) = cf(x)$, where the constant of proportionality is

$$c \;=\; \frac{1}{\int_\Omega f(y)\, d\mu(y)} \tag{2.22}$$

(to ensure that p integrates to one).[3] This leads to an estimator with zero variance, since

$$F \;=\; \frac{f(X)}{p(X)} \;=\; \frac{1}{c}$$

for all sample points X.

Unfortunately this technique is not practical, since we must already know the value of the desired integral in order to compute the normalization constant c. Nevertheless, by choosing a density function p whose shape is similar to f, variance can be reduced. Typically this is done by discarding or approximating some factors of f in order to obtain a function g that can be integrated analytically, and then letting $p \propto g$. It is also important to choose p such that there is a convenient method of generating samples from it. For low-dimensional integration problems, a useful strategy is to construct a discrete approximation of f (e.g. a piecewise constant or linear function). This can be done either during a separate initialization phase, or adaptively as the algorithm proceeds. The integral of such an approximation can be computed and maintained quite cheaply, and sampling can be done efficiently by means of tree structures or partial sums.

In summary, importance sampling is one of the most useful and powerful techniques of Monte Carlo integration. It is particularly helpful for integrands that have large values on a relatively small part of the domain, e.g. due to singularities.

2.5.3 Control variates

With *control variates,* the idea is to find a function g that can be integrated analytically and is similar to the integrand, and then subtract it. Effectively, the integral is rewritten as

$$I \;=\; \int_\Omega g(x)\, d\mu(x) \;+\; \int_\Omega f(x) - g(x)\, d\mu(x)\,,$$

and then sampled with an estimator of the form

$$F \;=\; \int_\Omega g(x)\, d\mu(x) \;+\; \frac{1}{N} \sum_{i=1}^{N} \frac{f(X_i) - g(X_i)}{p(X_i)}$$

[3] We assume that f is non-negative in this discussion. Otherwise the best choice is to let $p \propto |f|$, however the variance obtained this way is no longer zero [Kalos & Whitlock 1986].

where the value of the first integral is known exactly. (As usual p is the density function from which the X_i are chosen.) This estimator will have a lower variance than the basic estimator (2.12) whenever

$$V\left[\frac{f(X_i) - g(X_i)}{p(X_i)}\right] \;\le\; V\left[\frac{f(X_i)}{p(X_i)}\right] .$$

In particular, notice that if g is proportional to p, then the two estimators differ only by a constant, and their variance is the same. This implies that if g is already being used for importance sampling (up to a constant of proportionality), then it is not helpful to use it as a control variate as well.[4] From another point of view, given some function g that is an approximation to f, we must decide whether to use it as a control variate or as a density function for importance sampling. It is possible to show that either one of these choice could be the best, depending on the particular f and g. In general, if $f - g$ is nearly a constant function, then g should be used as a control variate; while if f/g is nearly constant, then g should be used for importance sampling [Kalos & Whitlock 1986].

As with importance sampling, control variates can be obtained by approximating some factors of f or by constructing a discrete approximation. Since there is no need to generate samples from g, such functions can be slightly easier to construct. However, note that for g to be useful as a control variate, it must take into account all of the significant factors of f. For example, consider an integral of the form $f(x) = f_1(x)\, f_2(x)$, and suppose that $f_1(x)$ represents the reflectivity of a surface at the point x, while $f_2(x)$ represents the incident power per unit area. Without some estimate of the magnitude of f_2, observe that f_1 is virtually useless as a control variate. On the other hand, f_1 can be used for importance sampling without any difficulties.

Control variates have had very few applications in graphics so far (e.g. see Lafortune & Willems [1995a]). One problem with the technique is the possibility of obtaining negative sample values, even for an integrand that is strictly positive. This can lead to large relative errors for integrals whose true value is close to zero (e.g. pixels in the dark regions of an image). On the other hand, the method is straightforward to apply, and can potentially give a modest variance reduction at little cost.

[4] See the discussion under Russian roulette below.

2.6 Variance reduction II: Uniform sample placement

Another important strategy for reducing variance is to ensure that samples are distributed more or less uniformly over the domain. We will examine several techniques for doing this, namely *stratified sampling*, *Latin hypercube sampling*, *orthogonal array sampling*, and *quasi-Monte Carlo methods*.

For these techniques, it is typically assumed that the domain is the s-dimensional unit cube $[0,1]^s$. Other domains can be handled by defining an appropriate transformation of the form $T : [0,1]^s \to \Omega$. Note that by choosing different mappings T, the transformed samples can be given different density functions. This makes it straightforward to apply importance sampling to the techniques described below.[5]

2.6.1 Stratified sampling

The idea of *stratified sampling* is to subdivide the domain Ω into several non-overlapping regions Ω_1, …, Ω_n such that

$$\bigcup_{i=1}^{n} \Omega_i \;=\; \Omega\,.$$

Each region Ω_i is called a *stratum*. A fixed number of samples n_i is then taken within each Ω_i, according to some given density function p_i.

For simplicity, assume that $\Omega = [0,1]^s$ and that p_i is simply the constant function on Ω_i. This leads to an estimate of the form

$$F' \;=\; \sum_{i=1}^{n} v_i\, F_i \tag{2.23}$$

$$\text{where} \qquad F_i \;=\; \frac{1}{n_i} \sum_{j=1}^{n_i} f(X_{i,j})\,. \tag{2.24}$$

Here $v_i = \mu(\Omega_i)$ is the volume of region Ω_i, and each $X_{i,j}$ is an independent sample from

[5] Note that if the desired density $p(x)$ is complex, it may be difficult to find a transformation T that generates it. This can be solved with rejection sampling, but the resulting samples will not be stratified as well.

p_i. The variance of this estimator is

$$V[F'] \;=\; \sum_{i=1}^{n} v_i^2\,\sigma_i^2\,/\,n_i\,, \tag{2.25}$$

where $\sigma_i^2 = V[f(X_{i,j})]$ denotes the variance of f within Ω_i.

To compare this against unstratified sampling, suppose that $n_i = v_i N$, where N is the total number of samples taken. Equation (2.25) then simplifies to

$$V[F'] \;=\; \frac{1}{N}\sum_{i=1}^{n} v_i\,\sigma_i^2\,.$$

On the other hand, the variance of the corresponding unstratified estimator is[6]

$$V[F] \;=\; \frac{1}{N}\left[\sum_{i=1}^{n} v_i\,\sigma_i^2 \;+\; \sum_{i=1}^{n} v_i(\mu_i - I)^2\right]\,, \tag{2.26}$$

where μ_i is the mean value of f in region Ω_i, and I the mean value of f over the whole domain. Since the right-hand sum is always non-negative, stratified sampling can never increase variance.

However, from (2.26) we see that variance is only reduced when the strata have different means; thus, the strata should be chosen to make these means as different as possible. Ideally, this would be achieved by stratifying the *range* of the integrand, by finding strata such that $x_i \in \Omega_i$ implies $x_1 \le x_2 \le \cdots \le x_N$.

Another point of view is to analyze the convergence rate. For functions with a bounded first derivative, the variance of stratified sampling converges at a rate of $O(N^{-1-2/s})$, while if the function is only piecewise continuous then the variance is $O(N^{-1-1/s})$ [Mitchell 1996]. (The convergence rate for the standard deviation is obtained by dividing these exponents by two.) Thus, stratified sampling can increase the convergence rate noticeably in low-dimensional domains, but has little effect in high-dimensional domains.

In summary, stratified sampling is a useful, inexpensive variance reduction technique.

[6] To obtain this result, observe that an unstratified sample in $[0,1]^s$ is equivalent to first choosing a random stratum I_j (according to the discrete probabilities v_i), and then randomly choosing X_j within Ω_{I_j}. From this point of view, X_j is chosen conditionally on I_j. This lets us apply the identity (2.11) to express the variance as a sum of two components, yielding equation (2.26).

It is mainly effective for low-dimensional integration problems where the integrand is reasonably well-behaved. If the dimension is high, or if the integrand has singularities or rapid oscillations in value (e.g. a texture with fine details), then stratified sampling will not help significantly. This is especially true for problems in graphics, where the number of samples taken for each integral is relatively small.

2.6.2 Latin hypercube sampling

Suppose that a total of N samples will be taken. The idea of *Latin hypercube sampling* is to subdivide the domain $[0,1]^s$ into N subintervals along each dimension, and to ensure that one sample lies in each subinterval. This can be done by choosing s permutations π_1, ..., π_s of $\{1,\ldots,N\}$, and letting the sample locations be

$$X_i^j \;=\; \frac{\pi_j(i) - U_{i,j}}{N}\,, \tag{2.27}$$

where X_i^j denotes the j-th coordinate of the sample X_i, and the $U_{i,j}$ are independent and uniformly distributed on $[0,1]$. In two dimensions, the sample pattern corresponds to the occurrences of a single symbol in a *Latin square* (i.e. an $N \times N$ array of N symbols such that no symbol appears twice in the same row or column).

Latin hypercube sampling was first proposed as a Monte Carlo integration technique by McKay et al. [1979]. It is closely related to Latin square sampling methods, which have been used in the design of statistical experiments since at least the 1920's (e.g. in agricultural research [Fisher 1925, Fisher 1926]). Yates [1953] and Patterson [1954] extended these techniques to arbitrary dimensions, and also analyzed their variance-reduction properties (in the context of survey sampling and experimental design). In computer graphics, Latin square sampling was introduced by Shirley [1990a] under the name of *N-rooks sampling* [Shirley 1990a, Shirley 1991].

The first satisfactory variance analysis of Latin hypercube sampling for Monte Carlo integration was given by Stein [1987]. First, we define a function $g(x)$ to be *additive* if it has the form

$$g(x) \;=\; \sum_{j=1}^{s} g_j(x^j)\,, \tag{2.28}$$

where x^j denotes the j-th component of $x \in [0,1]^s$. Next, let f_{add} denote the best additive approximation to f, i.e. the function of the form (2.28) which minimizes the mean squared error

$$\int_\Omega (f_{\text{add}}(x) - f(x))^2 \, d\mu(x) \, .$$

We can then write f as the sum of two components

$$f(x) \;=\; f_{\text{add}}(x) + f_{\text{res}}(x) \, ,$$

where f_{res} is orthogonal to all additive functions, i.e.

$$\int_\Omega f_{\text{res}}(x) \, g(x) \, d\mu(x) \;=\; 0$$

for any additive function g.

Stein [1987] was then able to show that variance of Latin hypercube sampling is

$$V[F'] \;=\; \frac{1}{N} \int_\Omega f_{\text{res}}^2(x) \, d\mu(x) \;+\; o(1/N) \, , \tag{2.29}$$

where $o(1/N)$ denotes a function that decreases faster than $1/N$. This expression should be compared to the variance using N independent samples, which is

$$V[F] \;=\; \frac{1}{N} \left(\int_\Omega f_{\text{res}}^2(x) \, d\mu(x) \;+\; \int_\Omega (f_{\text{add}}(x) - I)^2 \, d\mu(x) \right) \, .$$

The variance in the second case is always larger (for sufficiently large N). Thus Latin hypercube sampling improves the convergence rate for the additive component of the integrand. Furthermore, it is never significantly worse than using independent samples [Owen 1997a]:

$$V[F'] \;\le\; \frac{N}{N-1} \, V[F] \qquad \text{for } N \ge 2 \, .$$

Latin hypercube sampling is easy to implement and works very well for functions that are nearly additive. However, it does not work that well for image sampling, because the samples are not well-stratified in two dimensions. Except in special cases (e.g. pixels with vertical or horizontal edges), it has the same $O(1/N)$ variance that would be obtained with independent samples. This is inferior to stratified sampling, for which the variance is $O(N^{-2})$ for smooth functions and $O(N^{-3/2})$ for piecewise continuous functions.

2.6.3 Orthogonal array sampling

Orthogonal array sampling [Owen 1992, Tang 1993] is an important generalization of Latin hypercube sampling that addresses some of these deficiencies. Rather than stratifying all of the one-dimensional projections of the samples, it stratifies all of the t-dimensional projections for some $t \geq 2$. This increases the rate of convergence for the components of f that depend on t or fewer variables.

An *orthogonal array of strength* t is an $N \times s$ array of symbols, drawn from an alphabet of size b, such that every $N \times t$ submatrix contains the same number of copies of each of the b^t possible rows. (The submatrix is not necessarily contiguous; it can contain any subset of the columns.) If we let λ denote the number of times that each row appears (where λ is known as the *index* of the array), it is clear that $N = \lambda b^t$. The following table gives an example of an orthogonal array whose parameters are $OA(N, s, b, t) = (9, 4, 3, 2)$:

0	0	0	0
0	1	1	2
0	2	2	1
1	0	1	1
1	1	2	0
1	2	0	2
2	0	2	2
2	1	0	1
2	2	1	0

Various methods are known for constructing orthogonal arrays of strength $t = 2$ [Bose 1938, Bose & Bush 1952, Addelman & Kempthorne 1961], strength $t = 3$ [Bose & Bush 1952, Bush 1952], and arbitrary strengths $t \geq 3$ [Bush 1952]. Implementations of these methods are publicly available [Owen 1995a].

Let A be an $N \times s$ orthogonal array of strength t, where the symbols in the array are $\{0, 1, \ldots, b-1\}$. The first step of orthogonal array sampling is to randomize the array, by applying a permutation to the alphabet in each column. That is, we let

$$\hat{A}_{i,j} \;=\; \pi_j(A_{i,j}) \qquad \text{for all } i, j \,,$$

where $\pi_1, \ldots, \pi_s$ are random permutations of the symbols $\{0, \ldots, b-1\}$. It is easy to check that $\hat{A}$ is an orthogonal array with the same parameters (N, s, b, t) as the original array A. This step ensures that each of the b^s possible rows occurs in $\hat{A}$ with equal probability.

Now let the domain be $[0,1]^s$, and consider the family of b^s subcubes obtained by splitting each axis into b intervals of equal size. Each row of $\hat{A}$ can be interpreted as an index into this family of subcubes. The idea of orthogonal array sampling is to take one sample in each of the N subcubes specified by the rows of $\hat{A}$. Specifically, the j-th coordinate of sample X_i is

$$X_i^j \;=\; (\hat{A}_{i,j} + U_{i,j}) \,/\, b$$

where the $U_{i,j}$ are independent uniform samples on $[0,1]$. Because of the randomization step above, it is straightforward to show that each X_i is uniformly distributed in $[0,1]^s$, so that $F_N \;=\; (1/N) \sum_{i=1}^{N} f(X_i)$ is an unbiased estimator of the usual integral I.

To see the advantage of this technique, consider the sample distribution with respect to any t coordinate axes (i.e. project the samples into the subspace spanned by these axes). This subspace can be divided into b^t subcubes by splitting each axis into b intervals. The main property of orthogonal array sampling is that each of these subcubes contains the same number of samples. To see this, observe that the coordinates of the projected samples are specified by a particular $N \times t$ submatrix of the orthogonal array. By the definition of orthogonal arrays, each of the possible b^t rows occurs λ times in this submatrix, so that there will be exactly λ samples in each subcube.

Orthogonal array sampling is clearly a generalization of Latin hypercube sampling. Rather than stratifying the one-dimensional projections of the samples, it stratifies all of the t-dimensional projections simultaneously. (There are $\binom{s}{t}$ such projections in all.)

2.6.3.1 Analysis of variance decompositions

The variance reduction properties of orthogonal array sampling can be analyzed using *continuous analysis of variance (anova) decompositions* [Owen 1994, Owen 1992]. Our description follows [Owen 1992], which in turn is based on [Efron & Stein 1981].

Let $S = \{1, \ldots, s\}$ be the set of all coordinate indices, and let $U \subseteq S$ be any subset of these indices (there are 2^s possible subsets in all). We will use the notation x^U to refer to the set of coordinate variables x^j for $j \in U$. The *anova decomposition* of a given function f can then be written as a sum

$$f(x) \;=\; \sum_{U \subseteq S} f_U(x^U)\,, \tag{2.30}$$

where each function f_U depends only on the variables indexed by U.

The function when $U = \emptyset$ does not depend on any variables, and is called the *grand mean*:

$$I \;=\; f_\emptyset \;=\; \int_{[0,1]^s} f(x)\,dx\,.$$

The other $2^s - 1$ subsets of U are called *sources of variation*. The components of f that depend on just one variable are called the *main effects* and are defined as

$$f_j(x^j) \;=\; \int (f(x) - I) \prod_{i \neq j} dx^i\,.$$

Notice that all of these functions are orthogonal to the constant function $f_\emptyset = I$. Similarly, the *two-factor interactions* are defined by

$$f_{j,k}(x^{j,k}) \;=\; \int \Big(f(x) - I - f_j(x^j) - f_k(x^k) \Big) \prod_{i \neq j,k} dx^i$$

which represent the components of f that depend on two particular variables together. These functions are orthogonal to $f_\emptyset$ and to all the f_j.

In general, f_U is defined by

$$f_U(x^U) \;=\; \int \left(f(x) - \sum_{V \subset U} f_V(x^V) \right) dx^{S-U} \tag{2.31}$$

where the sum is over all proper subsets of U ($V \neq U$). The resulting set of functions is orthogonal, i.e. they satisfy

$$\int f_U(x^U)\, f_V(x^V)\,dx \;=\; 0$$

whenever $U \neq V$. This implies the useful property that

$$\int f^2(x)\,dx \;=\; \sum_{U \subseteq S} \int f_U^2(x^U)\,dx\,,$$

so that the variance of f can be written as

$$\int (f(x) - I)^2 \, dx \;=\; \sum_{|U|>0} \int f_U^2(x^U) \, dx \, .$$

As a particular case of this analysis, the best additive approximation to f is

$$f_{add}(x) = I + \sum_{j=1}^{s} f_j(x^j) \, ,$$

where the residual $f_{res} = f - f_{add}$ is orthogonal to all additive functions. The variance of Latin hypercube sampling can thus be rewritten as

$$\sigma^2_{\text{LH}} \;=\; \frac{1}{N} \sum_{|U|>1} \int f_U^2(x^U) \, dx \;+\; o(1/N) \, ,$$

i.e. the single-variable components of the variance converge at a rate faster than $1/N$.

Orthogonal array sampling generalizes this result; it is possible to show that the variance is [Owen 1992, Owen 1994]

$$\sigma^2_{\text{OA}} \;=\; \frac{1}{N} \sum_{|U|>t} \int f_U^2(x^U) \, dx \;+\; o(1/N) \, ,$$

i.e. the convergence rate is improved with respect to all components of the integrand that depend on t coordinates or less.

The case $t = 2$ is particularly interesting for graphics. For example, if we apply this technique to distribution ray tracing, it ensures that all the two dimensional projections are well stratified (over the pixel, lens aperture, light source, etc). This achieves a similar result to the sampling technique proposed by Cook et al. [1984], except that all combinations of two variables are stratified (including combinations such as the pixel x-coordinate and the aperture x-coordinate, for example).

2.6.3.2 Orthogonal array-based Latin hypercube sampling

Notice that because the t-dimensional margins are well-stratified, the w-dimensional margins are also stratified for any $w < t$. However, the resulting stratification is not as good. For example, in any one-dimensional projectional there will be exactly λb^{t-1} samples in each interval of width $1/b$. This is inferior to Latin hypercube sampling, which places one sample in each interval of width $1/(\lambda b^t)$.

There is a simple modification to orthogonal array sampling that yields the same one-dimensional stratification properties as Latin hypercube sampling. (The result, logically enough, is called *orthogonal array-based Latin hypercube sampling* [Tang 1993].) The idea is to remap the λb^t symbols within each column into a single sequence $\{0, 1, \ldots, \lambda b^t - 1\}$, by mapping the λb^{t-1} identical copies of each symbol m into a random permutation of the symbols

$$\lambda b^{t-1} m, \ \ldots, \ \lambda b^{t-1}(m+1) - 1 \, .$$

This process is repeated for each column separately. Letting $\hat{A}'$ be the modified array, the sample locations are then defined as

$$X_i^j = \frac{\hat{A}'_{i,j} + U_{i,j}}{\lambda b^t} \, .$$

This ensures that the samples are maximally stratified for each one-dimensional projection, as well as for each t-dimensional projection. It is possible to show that this leads to a further reduction in variance [Tang 1993].

This technique is similar to *multi-jittered sampling* [Chiu et al. 1994], which corresponds to the special case where $s = 2$ and $t = 2$.

2.6.4 Quasi-Monte Carlo methods

Quasi-Monte Carlo methods take these ideas a step further, by dispensing with randomness completely. The idea is to distribute the samples as uniformly as possible, by choosing their locations deterministically.

2.6.4.1 Discrepancy

Let $P = \{x_1, \ldots, x_N\}$ be a set of points in $[0, 1]^s$. Typically, the goal of quasi-Monte Carlo methods is minimize the *irregularity of distribution* of the samples with respect to some quantitative measure. One such measure is the *star discrepancy* of P. Let $\mathcal{B}^*$ denote the set of all axis-aligned boxes with one corner at the origin:

$$\mathcal{B}^* \;=\; \{B = [0, u_1] \times \cdots \times [0, u_s] \mid 0 \leq u_i \leq 1 \text{ for all } i\} \,.$$

Ideally, we would like each box B to contain exactly $\lambda(B)N$ of the points in P, where $\lambda(B) \;=\; u_1 \cdots u_s$ is the volume of B. The star discrepancy simply measures how much P deviates from this ideal situation:

$$D_N^*(P) \;=\; \sup_{B \in \mathcal{B}^*} \left| \frac{\#\{P \cap B\}}{N} - \lambda(B) \right| \,, \tag{2.32}$$

where $\#\{P \cap B\}$ denotes the number of points of P that are inside the box B.

Discrepancy measures can also be defined with respect to other sets of shapes (e.g. arbitrary axis aligned boxes, or convex regions [Niederreiter 1992]). For two-dimensional image sampling, it is particularly useful to measure discrepancy with respect to *edges*, by considering the family of shapes obtained by intersecting $[0, 1]^2$ with an arbitrary half-plane [Mitchell 1992]. The relevance of discrepancy to image sampling was first pointed out by Shirley [1991].

The significance of the star discrepancy is that it is closely related to bounds on the integration error. Specifically, the *Koksma-Hlawka inequality* states that

$$\left| \frac{1}{N} \sum_{i=1}^{N} f(x_i) \;-\; \int_{[0,1]^s} f(x)\, dx \right| \;\leq\; V_{HK}(f)\, D_N^*(P) \,,$$

where $V_{HK}(f)$ is the *variation of f in the sense of Hardy and Krause* [Niederreiter 1992]. Thus, the maximum integration error is directly proportional to the discrepancy, provided that the variation $V_{HK}(f)$ is finite. By finding low-discrepancy points sets and sequences, we can ensure that the integration error is small.

It is important to note that for dimensions $s \;\geq\; 2$, the variation $V_{HK}(f)$ is infinite whenever f is discontinuous.[7] This severely limits the usefulness of these bounds in computer graphics, where discontinuities are common. Also note that since $V_{HK}(f)$ is typically

[7] More precisely, $V_{HK}(f) = \infty$ whenever f is discontinuous along a surface that is not perpendicular to one of the s coordinate axes. In general, note that f must be at least s times differentiable in order for $V_{HK}(f)$ to be bounded in terms of the partial derivatives of f. That is, letting M be an upper bound on the magnitude of all partial derivatives of degree at most s, then $V_{HK}(f) \;\leq\; cM$ where the constant c depends only on s [Niederreiter 1992].

harder to evaluate than the original integral, these worst-case bounds are not useful for estimating or bounding the error in practice.

2.6.4.2 Low-discrepancy points sets and sequences

A *low-discrepancy sequence* is an infinite sequence of points $x_1, x_2, \ldots$ such that the star discrepancy is

$$D_N^*(P) \;=\; O\left(\frac{(\log N)^s}{N}\right)$$

for any prefix $P = \{x_1, \ldots, x_N\}$. (Note that P is actually a multiset, i.e. the multiplicity of the elements matters.) This result is achieved by a number of known constructions, and it is widely believed to be the best possible [Niederreiter 1992]. However, it should be noted that the best current lower bound for an arbitrary dimension s is only

$$D_N^*(P) \;\geq\; C(s) \cdot \frac{(\log N)^{s/2}}{N} \,,$$

i.e. there is a significant gap between these bounds.

If we drop the requirement that P is a prefix of an infinite sequence, the discrepancy can be improved slightly. A *low-discrepancy point set* is defined to be a multiset $P = \{x_1, \ldots, x_N\}$ for which

$$D_N^*(P) \;=\; O\left(\frac{(\log N)^{s-1}}{N}\right) \,.$$

(More precisely, this should be the definition of a low-discrepancy point set *construction*, since the bound does not make sense when applied to a single point set P.)

Combining these bounds with the Koksma-Hlawka inequality, the error of quasi-Monte Carlo integration is at most $O((\log N)^{s-1}/N)$ using a low-discrepancy point set, or $O((\log N)^s/N)$ using a prefix of a low-discrepancy sequence.

Note that these bounds are of questionable value unless N is very large, since $(\log N)^s$ is much larger than N for typical values of N and s. In particular, notice that the function $(\log N)^s/N$ is monotonically *increasing* for $N < e^s$ (i.e. the larger the sample size, the worse the error bound). In fact, we should not expect these error bounds to be meaningful until $(\log N)^s < N$ at the very least, since otherwise the error bound is worse than it would be for $N = 2$. To get an idea of how large N must be, consider the case $s = 6$. It is easy to check that $(\log N)^s/N > (\log 2)^s/2$ for all $N < 10^9$, and thus we should not expect meaningful error bounds until N is substantially larger than this.

However, these error bounds are overly pessimistic in practice. Low-discrepancy sequences often give better results than standard Monte Carlo even when N is fairly small, provided that the integrand is reasonably well behaved.

2.6.4.3 Halton sequences and Hammersley points

We now discuss several well-known constructions for low-discrepancy points sets and sequences. In one dimension, the *radical inverse sequence* $x_i = \phi_b(i)$ is obtained by first writing the base-b expansion of i:

$$i \;=\; \sum_{k \geq 0} d_{i,k} b^k \,,$$

and then reflecting the digits around the decimal point:

$$\phi_b(i) \;=\; \sum_{k \geq 0} d_{i,k}\, b^{-1-k} \,.$$

The special case when $b = 2$ is called the *van der Corput sequence,*

$$\frac{1}{2}, \frac{1}{4}, \frac{3}{4}, \frac{1}{8}, \frac{5}{8}, \frac{3}{8}, \ldots \,.$$

The discrepancy of the radical-inverse sequence is $O((\log N)/N)$ in any base b (although the implied constant increases with b).

To obtain a low-discrepancy sequence in several dimensions, we use a different radical inverse sequence in each dimension:

$$x_i \;=\; (\phi_{b_1}(i), \phi_{b_2}(i), \ldots, \phi_{b_s}(i))$$

where the bases b_i are all relatively prime. The classic example of this construction is the *Halton sequence*, where the b_i are chosen to be the first s primes:

$$x_i \;=\; (\phi_2(i), \phi_3(i), \phi_5(i), \ldots, \phi_{p_s}(i)) \,.$$

The Halton sequence has a discrepancy of $O((\log N)^s/N)$.

If the number of sample points N is known in advance, this discrepancy can be improved slightly by using equally spaced points i/N in the first dimension. The result is known as the *Hammersley point set*:

$$x_i \;=\; (i/N, \phi_2(i), \phi_3(i), \ldots, \phi_{p_{s-1}}(i))$$

where p_i denotes the i-th prime as before. The discrepancy of the Hammersley point set is $O((\log N)^{s-1}/N)$.

2.6.4.4 $(\mathbf{t}, \mathbf{m}, \mathbf{s})$-nets and $(\mathbf{t}, \mathbf{s})$-sequences

Although discrepancy is a useful measure of the irregularity of distribution of a set of points, it does not always accurately predict which sequences will work best for numerical integration. Recently there has been a great deal of interest in (t, m, s)-nets and (t, s)-sequences, which define the irregularity of distribution in a slightly different way. Let E be an *elementary interval in the base* b, which is simply an axis-aligned box of the form

$$E \;=\; \prod_{j=1}^{s} \left[\frac{t_j}{b^{k_j}}, \frac{t_j + 1}{b^{k_j}} \right)$$

where the exponents $k_j \;\ge\; 0$ are integers, and $0 \;\le\; t_j \;\le\; b^{k_j} - 1$. In other words, each dimension of the box must be a non-positive power of b, and the box must be aligned to an integer multiple of its size in each dimension. The volume of an elementary interval is clearly

$$\lambda(E) \;=\; b^{-\sum_{j=1}^{s} k_j} \,.$$

A $(0, m, s)$*-net in base* b is now defined to be a point set P of size $N = b^m$, such that every elementary interval of volume $1/b^{-m}$ contains exactly one point of P. This implies that a $(0, m, s)$-net is distributed as evenly as possible with respect to such intervals. For example, suppose that P is $(0, 4, 2)$-net in base 5. Then P would contain $N = 625$ points in the unit square $[0, 1]^2$, such that every elementary interval of size $1 \times 1/625$ contains a point of P. Similarly, all the intervals of size $1/5 \times 1/125$, $1/25 \times 1/25$, $1/125 \times 1/5$, and $1/625 \times 1$ would contain exactly one point of P.

The more general notion of a (t, m, s)*-net* is obtained by relaxing this definition somewhat. Rather than requiring every box of size b^{-m} to contain exactly one point, we require every box of size b^{t-m} to contain exactly b^t points. Clearly, smaller values of t are better. The reason for allowing $t > 0$ is to facilitate the construction of such sequences for more values of b and s. (In particular, $(0, m, s)$-nets for $m \geq 2$ can only exist when $s \leq b + 1$ [Niederreiter 1992].)

A (t, s)*-sequence* is then defined to be an infinite sequence $x_1, x_2, \ldots$ such that for all $m \geq 0$ and $k \geq 0$, the subsequence

$$x_{kb^m+1}, \ \ldots, \ x_{kb^{m+1}}$$

is a (t, m, s)-net in the base b. In particular, every prefix $x_1, \ldots, x_N$ of size $N = b^m$ is a (t, m, s)-net. Explicit constructions of such sequences for various values of b and s have been proposed by Sobol', Faure, Niederreiter, and Tezuka (see Niederreiter [1992] and Tezuka [1995]).

Every (t, s)-sequence is a low-discrepancy sequence, and every (t, m, s)-net is a low-discrepancy points set (provided that t is held fixed while m is increased). Thus these constructions have the same worst-case integration bounds as for the Halton sequences and Hammersley points. However, note that (t, s)-sequences and (t, m, s)-nets often work much better in practice, because the discrepancy is lower by a significant constant factor [Niederreiter 1992].

It is interesting to compare the equidistribution properties of (t, m, s)-nets to orthogonal array sampling. For simplicity let $t = 0$, and let A be an orthogonal array of strength m. Then in the terminology of (t, m, s)-nets, orthogonal array sampling ensures that there is one sample in each elementary interval E of volume $1/b^m$, where E has m sides of length $1/b$ and all other sides of length one. The Latin hypercube extension of Tang [1993] ensures that in addition, there is one sample in each elementary interval E that has one side of length $1/b^m$ and all other of length one. Thus the 1- and m-dimensional projections are maximally stratified. For comparison, the $(0, m, s)$-net not only achieves both of these properties, it also ensures that there is one sample in every other kind of elementary interval of volume $1/b^m$, so that the projections of dimension $2, 3, \ldots, t - 1$ are also stratified as well as possible.

2.6.4.5 Randomly permuted $(\mathbf{t},\mathbf{m},\mathbf{s})$-nets and $(\mathbf{t},\mathbf{s})$-sequences

A significant disadvantage of quasi-Monte Carlo methods is that the sample locations are deterministic. In computer graphics, this leads to significant aliasing artifacts [Mitchell 1992]. It also makes it difficult to compute error estimates, since unlike with Monte Carlo methods we cannot simply take several independent samples.

These difficulties can be resolved by using *randomly permuted* (t,m,s)*-nets and* (t,s)*-sequences* [Owen 1995b] (also called *scrambled nets and sequences*). These are obtained by applying random permutations to the digits of ordinary (t,m,s)-nets and (t,s)-sequences, in such a way that their equidistribution properties are preserved [Owen 1995b]. The idea is straightforward to implement, although its analysis is more involved.

Scrambled nets have several advantages. Most importantly, the resulting estimators are unbiased, since the sample points are uniformly distributed over the domain $[0,1]^s$. This makes it possible to obtain unbiased error estimates by taking several independent random samples (e.g. using different digit permutations of the same original (t,m,s)-net). (See Owen [1997a] for additional discussion of variance estimates.) In the context of computer graphics, scrambled nets also provide a way to eliminate the systematic aliasing artifacts typically encountered with quasi-Monte Carlo integration.

Second, it is possible to show that for smooth functions, scrambled nets lead to a variance of

$$V[\hat{I}] \;=\; O\left(\frac{(\log N)^{s-1}}{N^3}\right) ,$$

and thus an expected error of $O((\log N)^{(s-1)/2}N^{-3/2})$ in probability [Owen 1997b]. This is an improvement over both the Monte Carlo rate of $O(N^{-1/2})$ and the quasi-Monte Carlo rate of $O((\log N)^{s-1}N^{-1})$. In all cases, these bounds apply to a worst-case function f (of sufficient smoothness), but note that the quasi-Monte Carlo rate uses a deterministic set of points while the other bounds are averages over random choices made by the sampling algorithm.

Scrambled nets can improve the variance over ordinary Monte Carlo even when the function f is not smooth [Owen 1997b]. With respect to the analysis of variance decomposition described above, scrambled nets provide the greatest improvement on the components f_U where the number of variables $|U|$ is small. These functions f_U can be smooth even when f itself is not (due to integration over the variables in $S - U$), leading to fast convergence on these components.

2.6.4.6 Discussion

The convergence rates of quasi-Monte Carlo methods are rarely meaningful in computer graphics, due to smoothness requirements on the integrand and the relatively small sample sizes that are typically used. Other problems include the difficulty of estimating the variation $V_{HK}(f)$, and the fact that $(\log N)^{s-1}$ is typically much larger than N in practice. The lack of randomness in quasi-Monte Carlo methods is a distinct disadvantage, since it causes aliasing and precludes error estimation.

Hybrids of Monte Carlo and quasi-Monte Carlo seem promising, such as the scrambled (t, m, s)-nets described above. Although such methods do not necessarily work any better than standard Monte Carlo for discontinuous integrands, at least they are not worse. In particular, they do not introduce aliasing artifacts, and error estimates are available.

Keller [1996, 1997] has applied quasi-Monte Carlo methods to the radiosity problem (a special case of the light transport problem where all surfaces are diffuse). He uses a particle-tracing algorithm (similar to Pattanaik & Mudur [1993]), except that the directions for scattering are determined by a Halton sequence. He has reported a convergence rate that is slightly better than standard Monte Carlo on simple test scenes. The main benefit appears to be due to the sampling of the first four dimensions of each random walk (which control the selection of the initial point on a light source and the direction of emission).

2.7 Variance reduction III: Adaptive sample placement

A third family of variance reduction methods is based on the idea of adaptively controlling the sample density, in order to place more samples where they are most useful (e.g. where the integrand is large or changes rapidly). We discuss two different approaches to doing this. One is *adaptive sampling*, which can introduce bias unless special precautions are taken. The other approach consists of two closely related techniques called *Russian roulette* and *splitting*, which do not introduce bias and are especially useful for light transport problems.

2.7.1 Adaptive sampling

The idea of *adaptive sampling* (also called *sequential sampling*) is to take more samples where the integrand has the most variation. This is done by examining the samples that have been taken so far, and using this information to control the placement of future samples. Typically this involves computing the variance of the samples in a given region, which is then refined by taking more samples if the variance exceeds a given threshold. A number of such techniques have been proposed in graphics for image sampling (for example, see Lee et al. [1985], Purgathofer [1986], Kajiya [1986], [Mitchell 1987], Painter & Sloan [1989]).

Like importance sampling, the goal of adaptive sampling is to concentrate samples where they will do the most good. However, there are two important differences. First, importance sampling attempts to place more samples in regions where the integrand is large, while adaptive sampling attempts to places more samples where the variance is large. (Of course, with adaptive sampling we are free to use other criteria as well.) A second important difference is that with adaptive sampling, the sample density is changed "on the fly" rather than using *a priori* information.

The main disadvantage of adaptive sampling is that it can introduce bias, which in turn can lead to image artifacts. Bias can be avoided using *two-stage sampling* [Kirk & Arvo 1991], which consists of first drawing a small sample of size n from a representative region $R \subset \Omega$, and then using this information to determine the sample size N for the remaining portion $\Omega - R$ of the domain.[8] Although this technique eliminates bias, it also eliminates some of the advantages of adaptive sampling, since it cannot react to unusual samples encountered during the second stage of sampling.

Another problem with adaptive sampling is that it is not very effective for high-dimensional problems. The same problems are encountered as with stratified sampling: there are too many possible dimensions to refine. For example, if we split the region to be refined into two pieces along each axis, there will be 2^s new regions to sample. If most of the sampling error is due to variation along only one or two of these axes, the refinement will be very inefficient.

[8] Alternatively, two samples of size n and N could be drawn over the entire domain, where the first sample is used only to determine the value of N and is then discarded.

2.7.2 Russian roulette and splitting

Russian roulette and *splitting* are two closely related techniques that are often used in particle transport problems. Their purpose is to decrease the sample density where the integrand is small, and increase it where the integrand is large. Unlike adaptive sampling, however, these techniques do not introduce any bias. The applications of these methods in computer graphics have been described by Arvo & Kirk [1990].

Russian roulette. Russian roulette is usually applied to estimators that are a sum of many terms:

$$F \;=\; F_1 + \cdots + F_N\,.$$

For example, F might represent the radiance reflected from a surface along a particular viewing ray, and each F_i might represent the contribution of a particular light source.

The problem with this type of estimator is that typically most of the contributions are very small, and yet all of the F_i are equally expensive to evaluate. The basic idea of Russian roulette is to randomly skip most of the evaluations associated with small contributions, by replacing these F_i with new estimators of the form

$$F_i' \;=\; \begin{cases} \frac{1}{q_i}\,F_i & \text{with probability } q_i\,, \\ 0 & \text{otherwise}\,. \end{cases}$$

The evaluation probability q_i is chosen for each F_i separately, based on some convenient estimate of its contribution. Notice that the estimator F_i' is unbiased whenever F_i is, since

$$\begin{aligned} E[F_i'] \;&=\; q_i \cdot \frac{1}{q_i} E[F_i] + (1 - q_i) \cdot 0 \\ &=\; E[F_i]\,. \end{aligned}$$

Obviously this technique increases variance; it is basically the inverse of the *expected values* method described earlier. Nevertheless, Russian roulette can still increase efficiency, by reducing the average time required to evaluate F.

For example, suppose that each F_i represents the contribution of a particular light source to the radiance reflected from a surface. To reduce the number of visibility tests using Russian roulette, we first compute a tentative contribution t_i for each F_i by assuming that the light source is fully visible. Then a fixed threshold δ is typically chosen, and the probabilities q_i are set to

$$q_i \;=\; \min(1, t_i \,/\, \delta)\,.$$

Thus contributions larger than δ are always evaluated, while smaller contributions are randomly skipped in a way that does not cause bias.

Russian roulette is also used to terminate the random walks that occur particle transport calculations. (This was the original purpose of the method, as introduced by Kahn — see [Hammersley & Handscomb 1964, p. 99].) Similar to the previous example, the idea is to randomly terminate the walks whose estimated contributions are relatively small. That is, given the current walk $\mathbf{x}_0\mathbf{x}_1\cdots\mathbf{x}_k$, the probability of extending it is chosen to be proportional to the estimated contribution that would be obtained by extending the path further, i.e. the contribution of paths of the form $\mathbf{x}_0\cdots\mathbf{x}_{k'}$ where $k' > k$. This has the effect of terminating walks that have entered unproductive regions of the domain. In computer graphics, this technique is used extensively in ray tracing and Monte Carlo light transport calculations.

Splitting. Russian roulette is closely related to *splitting*, a technique in which an estimator F_i is replaced by one of the form

$$F_i' \;=\; \frac{1}{k}\sum_{j=1}^{k} F_{i,j}\,,$$

where the $F_{i,j}$ are independent samples from F_i. As with Russian roulette, the splitting factor k is chosen based on the estimated contribution of the sample F_i. (A larger estimated contribution generally corresponds to a larger value of k.) It is easy to verify that this transformation is unbiased, i.e.

$$E[F_i'] \;=\; E[F_i]\,.$$

In the context of particle transport calculations, this has the effect of splitting a single particle into k new particles which follow independent paths. Each particle is assigned a weight that is a fraction $1/k$ of the weight of the original particle. Typically this technique is applied when a particle enters a high-contribution region of the domain, e.g. if we are trying to measure leakage through a reactor shield, then splitting might be applied to neutrons that have already penetrated most of the way through the shield.

The basic idea behind both of these techniques is the same: given the current state $\mathbf{x}_0\mathbf{x}_1 \cdots \mathbf{x}_k$ of a random walk, we are free to use any function of this state in deciding how many samples of $\mathbf{x}_{k+1}$ will be taken. If we predict that the contribution of the path $\mathbf{x}_0 \cdots \mathbf{x}_{k+1}$ will be low, then most of the time we will take no samples at all; while if the contribution is high, we may decide to take several independent samples. If this is applied at every vertex, the resulting structure is a tree of paths.

In general, Russian roulette and splitting can be applied to any process where each sample is determined by a sequence of random steps. We can use any prefix of this sequence to estimate the importance of the final sample. This is then used to decide whether the current state should be discarded (if the importance is low) or replicated (if the importance is high). Although this idea is superficially similar to adaptive sampling, it does not introduce any bias.

Russian roulette is an indispensable technique in transport calculations, since it allows otherwise infinite random walks to be terminated without bias. Splitting is also useful if it is judiciously applied [Arvo & Kirk 1990]. In combination, these techniques can be very effective at directing sampling effort into the most productive regions of the domain.

2.8 Variance reduction IV: Correlated estimators

The last family of variance reduction methods we will discuss is based on the idea of finding two or more estimators whose values are correlated. So far these methods have not found significant uses in graphics, so our discussion will be brief.

2.8.1 Antithetic variates

The idea of *antithetic variates* is to find two estimators F_1 and F_2 whose values are negatively correlated, and add them. For example, suppose that the desired integral is $\int_0^1 f(x)\,dx$, and consider the estimator

$$F \;=\; (f(U) + f(1-U))\,/\,2$$

where U is uniformly distributed on $[0, 1]$. If the function f is monotonically increasing (or monotonically decreasing), then $f(U)$ and $f(1 - U)$ will be negatively correlated, so that F will have lower variance than if the two samples were independent [Rubinstein 1981, p. 135]. Furthermore, the estimator F is exact whenever the integrand is a linear function (i.e. $f(x) = ax + b$).

This idea can be easily adapted to the domain $[0, 1]^s$, by considering pairs of sample points of the form

$$X_1 = (U_1, \ldots, U_s) \qquad \text{and} \qquad X_2 = (1 - U_1, \ldots, 1 - U_s)\,.$$

Again, this strategy is exact for linear integrands. If more than two samples are desired, the domain can be subdivided into several rectangular regions Ω_i, and a pair of samples of the form above can be taken in each region.

Antithetic variates of this type are most useful for smooth integrands, where f is approximately linear on each subregion Ω_i. For many graphics problems, on the other hand, variance is mainly due to discontinuities and singularities of the integrand. These contributions tend to overwhelm any variance improvements on the smooth regions of the integrand, so that antithetic variates are of limited usefulness.

2.8.2 Regression methods

Regression methods are a more advanced way to take advantage of several correlated estimators. Suppose that we are given several unbiased estimators F_1, ..., F_n for the desired quantity I, and that the F_i are correlated in some way (e.g. because they use different transformations of the same random numbers, as in the antithetic variates example). The idea is to take several samples from each estimator, and apply standard linear regression techniques in order to determine the best estimate for I that takes all sources of correlation into account.

Specifically, the technique works by taking N samples from each estimator (where the j-th samples from F_i is denoted $F_{i,j}$). We then compute the sample means

$$\hat{I}_i = \frac{1}{N} \sum_{j=1}^{N} F_{i,j} \quad \text{for } i = 1, \ldots, n\,,$$

and the sampling variance-covariance matrix $\hat{\mathbf{V}}$, a square $n \times n$ array whose entries are

$$\hat{V}_{i,j} \;=\; \frac{1}{N-1} \sum_{k=1}^{N} (F_{i,k} - \hat{I}_i)\,(F_{j,k} - \hat{I}_j)\,.$$

The final estimate F is then given by

$$F \;=\; (\mathbf{X}^* \, \hat{\mathbf{V}}^{-1} \, \mathbf{X})^{-1} \, \mathbf{X}^* \, \hat{\mathbf{V}}^{-1} \, \hat{\mathbf{I}}\,, \tag{2.33}$$

where $\mathbf{X}^*$ denotes the transpose of $\mathbf{X}$, $\mathbf{X} \;=\; [1 \ldots 1]^*$ is a column vector of length n, and $\hat{\mathbf{I}} = [\hat{I}_1 \ldots \hat{I}_n]^*$ is the column vector of sample means. Equation (2.33) is the standard minimum-variance unbiased linear estimator of the desired mean I, except that we have replaced the true variance-covariance matrix $\mathbf{V}$ by an approximation $\hat{\mathbf{V}}$. Further details can be found in Hammersley & Handscomb [1964].

Note that this technique introduces some bias, due to the fact that the same random samples are used to estimate both the sample means $\hat{I}_i$ and the variance-covariance matrix entries $\hat{V}_{i,j}$ (which are used to weight the $\hat{I}_i$). This bias could be avoided by using different random samples for these two purposes (of course, this would increase the cost).

The main problem with regression methods is in finding a suitable set of correlated estimators. If the integrand has discontinuities or singularities, then simple transformations of the form $f(U)$ and $f(1-U)$ will not produce a significant amount of correlation. Another problem is that this method requires that a substantial number of samples be taken, in order to estimate the covariance matrix with any reasonable accuracy.

Part I

Models for Bidirectional Light Transport in Computer Graphics

Chapter 3

Radiometry and Light Transport

In this chapter, we describe the domains, quantities, and equations that are used for light transport calculations. Many of these concepts have their origins in *radiometry,* a field that studies the measurement of electromagnetic radiation. Radiometry is a natural foundation for graphics, because light is part of the electromagnetic spectrum.

We start by discussing the mathematical representation of the scene model. We then discuss the *phase space* and *trajectory space,* and show how radiometric quantities can be defined in terms of *photon events.* Next we give definitions of the quantities that are needed for light transport calculations, including power, irradiance, radiance, and spectral radiance. We also discuss the concepts of incident and exitant radiance functions.

We then describe how the light transport problem is formulated mathematically. This starts with the definition of the *bidirectional scattering distribution function* (BSDF), which gives a mathematical description of the way that light is scattered by a surface. We show how the BSDF is used to define the basic light transport equations, and we give a brief introduction to adjoint methods and bidirectional algorithms. We also explain why non-symmetric BSDF's require special treatment in bidirectional algorithms, and we define the useful concept of an *adjoint BSDF*. These ideas will be of central importance for the next several chapters.

Appendix 3.A discusses field and surface radiance functions [Arvo 1995], and compares them with the incident and exitant radiance functions that we use instead. Finally, Appendix 3.B gives the details of our measure-theoretic radiometry framework, in which we apply the tools of measure theory to define radiometric concepts more precisely. The main task is to define and use suitable measure functions, extending the work of Arvo [1995, Chapter 2].

A good introduction to radiometry is the book by McCluney [1994]. Other good references include [Nicodemus 1976], [Arvo 1995], [Cohen & Wallace 1993], and [Glassner 1995]. Note that our development is quite different than the standard treatments, due to the emphasis on measure theory.

3.1 Domains and measures

We assume that the scene geometry consists of a finite set of surfaces in $\mathbb{R}^3$, whose union is denoted $\mathcal{M}$. Formally, each surface is a piecewise differentiable two-dimensional manifold, possibly with boundary. For technical reasons, we require each manifold to be a closed set; that is, every manifold M must include its boundary ∂M. This prevents gaps between abutting surfaces (e.g. consider a cube formed from six squares). Note that $\mathcal{M}$ itself is not necessarily a manifold. For example, consider two spheres that touch at a point, or a box sitting on a table.

The surfaces divide $\mathbb{R}^3$ into a number of connected cells, each filled with a non-participating medium with a constant refractive index (i.e. volume absorption, emission, and scattering are not allowed).[1] It is possible that some surfaces do not belong to any cell boundary (e.g., a polygon floating in space).

We define an area measure A on $\mathcal{M}$ in the obvious way,[2] so that $A(D)$ denotes the area

[1] With this convention, all objects are hollow inside; a “solid” object is simply an empty cell with an opaque boundary. This representation is actually used by many rendering systems. Alternatively, a cell could be allowed to contain a perfectly absorbing medium. However, this would require some extra care with definitions, for example when defining the visibility and ray-casting functions used in Chapter 4.

[2] Given that $\mathcal{M}$ is the union of manifolds $\mathcal{M}_1, \ldots, \mathcal{M}_N$, we define $A(D)$ as the sum of the areas $A_i(D \cap \mathcal{M}_i)$, where A_i is the usual area measure on the manifold $\mathcal{M}_i$. The measurable sets $D \subset \mathcal{M}$ are defined by the requirement that all $D \cap \mathcal{M}_i$ are measurable. We also require that the intersection between any pair of surfaces $\mathcal{M}_i$ and $\mathcal{M}_j$ is a set of measure zero. In practice, this means that when the intersection between two surfaces has non-zero area (e.g. a cube sitting on a table), the rendering system must arbitrarily choose one surface over the other. This ensures that almost every point of $\mathcal{M}$ (up to a set of area measure zero) has a unique set of surface properties.

of a region $D \subset \mathcal{M}$. The notation

$$\int_{\mathcal{M}} f(\mathbf{x})\, dA(\mathbf{x})$$

denotes the Lebesgue integral of the function $f : \mathcal{M} \to \mathrm{I\!R}$ with respect to surface area.

Directions are represented as unit-length vectors $\omega \in \mathrm{I\!R}^3$. The set of all directions is denoted $\mathcal{S}^2$, the unit sphere in $\mathrm{I\!R}^3$. Let σ be the usual surface area measure on $\mathcal{S}^2$. Given a set of directions $D \subset \mathcal{S}^2$, the *solid angle* occupied by D is simply $\sigma(D)$. Similarly, the solid angle *subtended* by a surface P from a point $\mathbf{x}$ is determined by projecting P onto the unit sphere centered at $\mathbf{x}$, and computing the measure of the resulting set of directions.

Another useful concept is the *projected solid angle* [Nicodemus 1976, p. 70], which arises in determining the irradiance (power per unit area) received by surface. Given a point $\mathbf{x} \in \mathcal{M}$, let $\mathbf{N}(\mathbf{x})$ be the surface normal at $\mathbf{x}$. Given a set of directions $D \subset \mathcal{S}^2$, the projected solid angle measure $\sigma_{\mathbf{x}}^{\perp}$ is defined by

$$\sigma_{\mathbf{x}}^{\perp}(D) \;=\; \int_D |\omega \cdot \mathbf{N}(\mathbf{x})|\, d\sigma(\omega)\,. \tag{3.1}$$

The factor $\omega \cdot \mathbf{N}(\mathbf{x})$ is often written as $\cos\theta$, where θ is the polar angle of ω (i.e. the angle between ω and the surface normal).

The name *projected solid angle* arises from the following geometric interpretation. Let $T_{\mathcal{M}}(\mathbf{x})$ be the *tangent space* at the point $\mathbf{x}$, i.e. the space of vectors in $\mathrm{I\!R}^3$ that are perpendicular to the surface normal:

$$T_{\mathcal{M}}(\mathbf{x}) \;=\; \{\mathbf{y} \in \mathrm{I\!R}^3 \mid \mathbf{y} \cdot \mathbf{N}(\mathbf{x}) = 0\}\,.$$

(Unlike the more familiar *tangent plane*, the tangent space passes through the origin. Thus it is a linear space rather than an affine one.) The tangent space divides $\mathcal{S}^2$ into two hemispheres, namely the *upward hemisphere*

$$\mathcal{H}_+^2(\mathbf{x}) \;=\; \{\omega \in \mathcal{S}^2 \mid \omega \cdot \mathbf{N}(\mathbf{x}) > 0\} \tag{3.2}$$

and the *downward hemisphere*

$$\mathcal{H}_-^2(\mathbf{x}) \;=\; \{\omega \in \mathcal{S}^2 \mid \omega \cdot \mathbf{N}(\mathbf{x}) < 0\}\,.$$

Now given a set of directions D contained by just one hemisphere, the projected solid angle can be obtained by simply projecting D orthogonally onto the tangent space, and then finding the area of the resulting planar region. For example, suppose that D is the entire upward hemisphere $\mathcal{H}^2_+$. The corresponding projected region is a unit disc, so we have $\sigma^\perp_{\mathbf{x}}(\mathcal{H}^2_+) = \pi$.

3.2 The phase space

Radiometric quantities can be defined within the more general framework of *transport theory*, which studies the motion of particles in an abstract setting. Each particle is characterized by a small number of parameters, which vary as a function of time. Typical particles such as neutrons or gas molecules can be represented by their position and velocity, for a total of 6 degrees of freedom. The state of a system of N particles is then represented as a $6N$-dimensional vector, which can be thought of as a point in the $6N$-dimensional *phase space* containing all possible system states.[3] The evolution of the system over time corresponds to a one-dimensional curve in phase space.

We now consider how this applies to light transport. Under the assumption that light is unpolarized and perfectly incoherent, the state of each photon can be represented by its position $\mathbf{x}$, direction of motion ω, and wavelength λ [Nicodemus 1976, p. 8]. Thus for a system of N photons, the phase space would be $6N$-dimensional.

However, for particles that do not interact with each other (such as photons), it is more useful to let the phase space correspond to the state of a single particle. With this convention, the phase space ψ is only 6-dimensional, and can be expressed as

$$\psi \;=\; \mathbb{R}^3 \times \mathcal{S}^2 \times \mathbb{R}^+\,,$$

where $\mathbb{R}^+$ denotes the positive real numbers (corresponding to the range of allowable wavelengths). A system of N photons is represented as a set of N points in this 6-dimensional space, whose positions vary as a function of time.

Radiometric quantities can then be defined by counting the number of photons in a given

[3] For many problems the natural phase space is not really $6N$-dimensional, since physical laws may cause certain properties of the initial state to be preserved for all time (e.g., the total energy). This restricts the phase space to be a lower-dimensional manifold within the $6N$-dimensional Euclidean space defined above.

region of the phase space, or measuring their density with respect to one or more parameters. The most basic of these quantities is the *photon number* N_p, which simply measures the number of photons in a given phase space region [McCluney 1994, p. 26]. For example, we could count the number of photons in a given spatial volume $\Omega \subset \mathbb{R}^3$ at a fixed time t_0, with no restrictions on the direction or wavelength parameters. This corresponds to the region $\Omega \times \mathcal{S}^2 \times \mathbb{R}^+$ of the phase space ψ.

3.3 The trajectory space and photon events

We generalize the notion of a radiometric measurement further, by considering the time dimension explicitly. If the phase space positions of all photons are graphed over time, we obtain a set of one-dimensional curves in the *trajectory space*

$$\Psi \;=\; \mathbb{R} \times \psi \,,$$

where the first parameter represents time. Radiometric measurements are defined by specifying a set of *photon events* along these curves, and then measuring the distribution of these events in various ways.

A photon event is a single point in the trajectory space Ψ. Some events have natural definitions; for example, each emission, absorption, or scattering event corresponds to a single point along a photon trajectory.[4] Other events can be defined artificially, usually by specifying a *surface* in Ψ that intersects the photon trajectories at a set of points. For example, we could define the events to be the photon states at a particular time t_0. This corresponds to intersecting the trajectories with the plane $t = t_0$ in the trajectory space Ψ. Similarly, given an arbitrary plane P in $\mathbb{R}^3$, we could define a photon event to be a crossing of P, corresponding to an intersection with the surface $\mathbb{R} \times P \times \mathcal{S}^2 \times \mathbb{R}^+$ in trajectory space.

Once the photon events have been defined, we are left with a set of points in the trajectory

[4] In fact, each scattering event corresponds to *two* points along the photon trajectory, since the ω parameter has different values before and after the collision (corresponding to a discontinuity in the trajectory). Similarly, the wavelength parameter λ could change discontinuously in a fluorescent material. Thus, we must distinguish between *in-scattering* and *out-scattering* events, according to whether we measure the photon state before or after the collision. This is the basis for distinguishing between incident and exitant radiance functions, discussed in Section 3.5.

space Ψ. These points may be distributed throughout the whole space Ψ, or they may lie on some lower-dimensional manifold (e.g. if the photon events were defined as an intersection with a surface). To define a radiometric quantity, we then measure the distribution of these events with respect to a suitable geometric measure.

For this purpose, it is convenient to assume that the events are so numerous that their density can be modeled by continuous distributions rather than discrete ones. Rather than counting the photon events in a given region of the trajectory space, for example, we determine their total *radiant energy* Q (measured in joules $[\mathrm{J}]$). We will ignore the discrete nature of photons and assume that Q can take on any non-negative real value. (Note that each photon has an energy of $h\nu$, where h is Planck's constant, and $\nu = 1/\lambda$ is frequency.)

3.4 Radiometric quantities

We now discuss some of the most important radiometric quantities. Each of these is defined by measuring the distribution of energy with respect to one or more parameters. The discussion here is informal; a more detailed development is given in Appendix 3.B.

3.4.1 Power

Radiant power is defined as energy per unit time,

$$\Phi \;=\; \frac{dQ}{dt}\,, \tag{3.3}$$

and is measured in watts $[\mathrm{W} = \mathrm{J} \cdot \mathrm{s}^{-1}]$. For example, this is the quantity used to describe the rate at which energy is emitted or absorbed by a finite surface $S \subset \mathbb{R}^3$.

The notation (3.3) could be written more precisely as

$$\Phi(t) \;=\; \frac{dQ(t)}{dt}\,,$$

which makes it clear that Φ and Q are functions of time. Obviously Q *must* be defined as a function of time, in order for the idea of differentiating it to make sense. In general, this is done by defining $Q(t)$ to measure the energy of the photon events in some region $D(t)$ of trajectory space, where the region $D(t)$ grows with time. For example, suppose that we are counting emission events, and consider the region

$$D(t) \;=\; [0,t] \times S \times \mathcal{S}^2 \times \mathbb{R}^+ \,,$$

where $S \subset \mathbb{R}^3$ is a finite surface. In this case, $Q(t)$ represents the total energy emitted by S over the time interval $[0,t]$, so that $\Phi(t) = dQ(t)/dt$ measures the energy emission per unit time (at each time t).

However, we will usually ignore these subtleties. Most often we are concerned with systems in equilibrium, so that the density of photon events in phase space does not change with time. In this case, the t parameter can be omitted from the notation, as in equation (3.3).

3.4.2 Irradiance

Continuing with our discussion of radiometric quantities, *irradiance* is defined as power per unit surface area:

$$E(\mathbf{x}) \;=\; \frac{d\Phi(\mathbf{x})}{dA(\mathbf{x})}\,, \tag{3.4}$$

with units of $[\mathrm{W} \cdot \mathrm{m}^{-2}]$. It is always defined with respect to a point $\mathbf{x}$ on a surface S (either real or imaginary), with a specific normal $\mathbf{N}(\mathbf{x})$. The term *irradiance* also generally implies the measurement of incident radiation, on one side of the surface only (i.e. light incident from the upward hemisphere $\mathcal{H}^2_+(\mathbf{x})$). When light is leaving the surface, through either emission or scattering, the preferred term is *radiant exitance* (denoted by the symbol M) [Nicodemus 1978, p. 11]. Another common term is *radiosity*, which was introduced by Moon [1936] and popularized in the heat transfer literature (cf. Heckbert [1992]).

3.4.3 Radiance

For light transport calculations, by far the most important quantity is *radiance*, defined by

$$L(\mathbf{x},\omega) \;=\; \frac{d^2\Phi(\mathbf{x},\omega)}{dA^{\perp}_{\omega}(\mathbf{x})\, d\sigma(\omega)}\,, \tag{3.5}$$

where $A_\omega^\perp$ is the *projected area measure*, which measures area on a hypothetical surface perpendicular to ω. That is, to measure the radiance at $(\mathbf{x}, \omega)$, we count the number of photons per unit time passing through a small surface $dA_\omega^\perp(\mathbf{x})$ perpendicular to ω, whose directions are contained in a small solid angle $d\sigma(\omega)$ around ω. Radiance is defined as the limiting ratio of the power $d\Phi$ represented by these photons, divided by the product $dA_\omega^\perp(\mathbf{x})\, d\sigma(\omega)$. The corresponding units are $[\mathrm{W} \cdot \mathrm{m}^{-2} \cdot \mathrm{sr}^{-1}]$.

When measuring the radiance leaving a real surface S, a more convenient equation is given by

$$L(\mathbf{x}, \omega) \;=\; \frac{d^2\Phi(\mathbf{x}, \omega)}{|\omega \cdot \mathbf{N}(\mathbf{x})|\, dA(\mathbf{x})\, d\sigma(\omega)}\,, \tag{3.6}$$

where as before A is the area measure on S, and $\mathbf{N}(\mathbf{x})$ is the surface normal at $\mathbf{x}$. This relates the projected area $dA_\omega^\perp$ to the ordinary area dA, according to[5]

$$dA_\omega^\perp(\mathbf{x}) \;=\; |\omega \cdot \mathbf{N}(\mathbf{x})|\, dA(\mathbf{x})\,. \tag{3.7}$$

Alternatively, the $|\omega \cdot \mathbf{N}(\mathbf{x})|$ factor can be absorbed into the projected solid angle measure defined above, leading to

$$L(\mathbf{x}, \omega) \;=\; \frac{d^2\Phi(\mathbf{x}, \omega)}{dA(\mathbf{x})\, d\sigma_{\mathbf{x}}^\perp(\omega)}\,. \tag{3.8}$$

This is the most useful definition when dealing with radiance on real surfaces, because it uses the natural area measure A.

3.4.4 Spectral radiance

Carrying this one step further, *spectral radiance* L_λ is defined by

$$L_\lambda(\mathbf{x}, \omega, \lambda) \;=\; \frac{d^3\Phi(\mathbf{x}, \omega, \lambda)}{dA(\mathbf{x})\, d\sigma_{\mathbf{x}}^\perp(\omega)\, d\lambda}\,, \tag{3.9}$$

that is, $L_\lambda \;=\; dL/d\lambda$. The units are typically given as $[\mathrm{W} \cdot \mathrm{m}^{-2} \cdot \mathrm{sr}^{-1} \cdot \mathrm{nm}^{-1}]$, where the use of nanometers for wavelength helps to avoid confusion with the spatial variables [Nicodemus 1976, p. 49]. Other spectral quantities can be defined similarly, e.g. spectral

[5] More precisely, the projected area measure is defined by $A_\omega^\perp(D) = \int_D |\omega \cdot \mathbf{N}(\mathbf{x})|\, dA(\mathbf{x})$, where D is an arbitrary region of S.

power is defined by $\Phi_\lambda = d\Phi / d\lambda$.

Spectral radiance is often considered to be the fundamental radiometric quantity, in that many other common quantities can be derived from it. For example, radiance is given by

$$L(\mathbf{x}, \omega) \;=\; \int_0^\infty L_\lambda(\mathbf{x}, \omega, \lambda) \, d\lambda \,,$$

from which irradiance can be obtained by

$$E(\mathbf{x}) \;=\; \int_{\mathcal{H}^2_+(\mathbf{x})} L(\mathbf{x}, \omega) \, d\sigma^\perp_{\mathbf{x}}(\omega) \,.$$

In this dissertation, we will most often deal with spectral radiance L_λ. However, for conciseness we will usually just refer to this as "radiance" and use the symbol L. This is a slight abuse of terminology, but it is common practice in computer graphics.

Many other radiometric quantities have been defined, but we will not need them here. The manual by Nicodemus [1976] is an excellent reference on this topic, although some of the notation has been superceded by the *USA Standard Nomenclature and Definitions for Illuminating Engineering* [American National Standards Institute 1986].

3.5 Incident and exitant radiance functions

A *radiance function* is simply a function whose values correspond to radiance measurements.[6] Most often, we will work with functions of the form

$$L : \mathcal{M} \times \mathcal{S}^2 \to \mathbb{R} \,,$$

where $\mathcal{M}$ is the set of scene surfaces (Section 3.1). Occasionally, radiance functions of the form

$$L : \mathbb{R}^3 \times \mathcal{S}^2 \to \mathbb{R}$$

will also be useful. Note that we allow negative values for $L(\mathbf{x}, \omega)$ (which have no physical meaning), to ensure that the set of all radiance functions is a vector space.

[6] As mentioned in Section 3.4, we will often use the terms *radiance* and *spectral radiance* interchangeably, ignoring the extra λ parameter.

We will distinguish between *incident* and *exitant*[7] radiance functions, according to the interpretation of the ω parameter. An incident function $L_{\rm i}(\mathbf{x}, \omega)$ measures the radiance *arriving* at $\mathbf{x}$ from the direction ω, while an exitant function $L_{\rm o}(\mathbf{x}, \omega)$ measures the radiance *leaving* from $\mathbf{x}$ in the direction ω. In free space, these quantities are related by

$$L_{\rm i}(\mathbf{x}, \omega) \;=\; L_{\rm o}(\mathbf{x}, -\omega)\,. \tag{3.10}$$

However, at surfaces the distinction is more fundamental: $L_{\rm i}$ and $L_{\rm o}$ measure different sets of photon events, corresponding to the photon states just before their arrival at the surface, or just after their departure respectively. The relation between $L_{\rm i}$ and $L_{\rm o}$ can be quite complex, since it depends on the scattering properties of the surface.

The difference between incident and exitant radiance can be understood more precisely in terms of the trajectory space Ψ. Recall that each photon traces out a one-dimensional curve in this space, namely the graph of the function $(\mathbf{x}_i, \omega_i, \lambda_i)(t)$ over all values of t. To measure radiance, we define a photon event to be an intersection of one of these curves with the surface $\mathbb{P} = \mathbb{R} \times \mathcal{M} \times \mathcal{S}^2 \times \mathbb{R}^+$ in trajectory space. Our key observation is that this curve is not continuous at $\mathbb{P}$, since scattered photons instantaneously change their direction and/or wavelength. (A continuous curve would correspond to a photon that passes through $\mathcal{M}$ without any change.) Similarly, the curves for emitted and absorbed photons are discontinuous, since they are defined on only one side of $\mathbb{P}$.

We now observe that $L_{\rm i}$ and $L_{\rm o}$ measure events that are limit points of trajectories on *opposite sides* of the surface $\mathbb{P}$. Each event $(t_i, \mathbf{x}_i, \omega_i, \lambda_i)$ measured by $L_{\rm i}$ is the limit of a trajectory defined for $t < t_i$, while an event measured by $L_{\rm o}$ is the limit of a trajectory defined for $t > t_i$. This gives a simple and precise way to differentiate between incident and exitant radiance.

Note that incident and exitant radiance functions are quite similar to the *field* and *surface radiance functions* proposed by Arvo [1995] (the main difference is that the direction of ω is reversed for field radiance as compared to incident radiance). Appendix 3.A discusses these two approaches and explains the advantages of incident and exitant radiance functions.

[7] Nicodemus prefers the spelling *exitent*, and states that this term was coined by Richmond (cf. [Nicodemus 1976, p. 25]). Our use of *exitant* stems from [Christensen et al. 1993], where the term appears to have been re-invented.

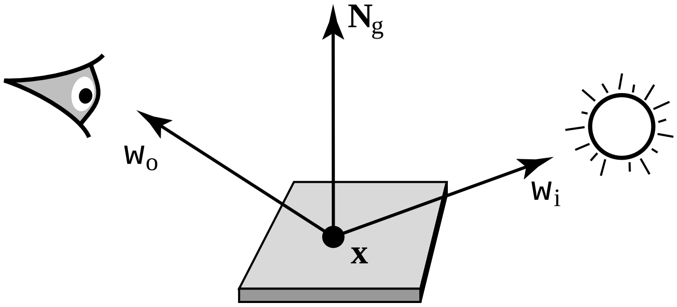


Figure 3.1: Geometry for defining the bidirectional scattering distribution function (BSDF).

3.6 The bidirectional scattering distribution function

The *bidirectional scattering distribution function* (BSDF) is a mathematical description of the light-scattering properties of a surface. Let $\mathbf{x} \in \mathcal{M}$ be a fixed point on the scene surfaces, and consider the radiance leaving $\mathbf{x}$ in a particular direction ω_o (see Figure 3.1). We will denote this $L_\mathrm{o}(\omega_\mathrm{o})$, dropping $\mathbf{x}$ from our notation. In general, the radiance $L_\mathrm{o}(\omega_\mathrm{o})$ depends on the radiance arriving at $\mathbf{x}$ from all directions. For now, we fix a particular direction ω_i, and consider the incident light from an infinitesimal cone around ω_i, where the cone occupies a solid angle of $d\sigma(\omega_\mathrm{i})$. This light strikes the surface at the point $\mathbf{x}$, and generates an irradiance equal to

$$dE(\omega_\mathrm{i}) \;=\; L_\mathrm{i}(\omega_\mathrm{i})\, d\sigma^\perp(\omega_\mathrm{i})\,.$$

The light is then scattered by the surface in all directions; we let $dL_\mathrm{o}(\omega_\mathrm{o})$ represent the contribution made to the radiance leaving in direction ω_o.

It can be observed experimentally that as $dE(\omega_\mathrm{i})$ is increased (by increasing either L_i or $d\sigma(\omega_\mathrm{i})$), there is a proportional increase in the observed radiance $dL_\mathrm{o}(\omega_\mathrm{o})$:

$$dL_\mathrm{o}(\omega_\mathrm{o}) \;\propto\; dE(\omega_\mathrm{i})\,.$$

This corresponds to the fact that light behaves linearly under normal circumstances (recall Section 1.5.3).

The BSDF $f_\mathrm{s}(\omega_\mathrm{i} \!\rightarrow\! \omega_\mathrm{o})$ is now simply defined to be this constant of proportionality:

$$f_\mathrm{s}(\omega_\mathrm{i} \!\rightarrow\! \omega_\mathrm{o}) \;=\; \frac{dL_\mathrm{o}(\omega_\mathrm{o})}{dE(\omega_\mathrm{i})} \;=\; \frac{dL_\mathrm{o}(\omega_\mathrm{o})}{L_\mathrm{i}(\omega_\mathrm{i})\, d\sigma^\perp(\omega_\mathrm{i})}\,. \tag{3.11}$$

In words, $f_\mathrm{s}(\omega_\mathrm{i} \!\rightarrow\! \omega_\mathrm{o})$ is the observed radiance leaving in direction ω_o, per unit of irradiance arriving from ω_i. The notation $\omega_\mathrm{i} \!\rightarrow\! \omega_\mathrm{o}$ symbolizes the direction of light flow.

3.6.1 The scattering equation

By integrating the relationship

$$dL_\mathrm{o}(\omega_\mathrm{o}) \;=\; L_\mathrm{i}(\omega_\mathrm{i})\, f_\mathrm{s}(\omega_\mathrm{i} \!\rightarrow\! \omega_\mathrm{o})\, d\sigma^\perp(\omega_\mathrm{i})$$

over all directions, we can now predict $L_\mathrm{o}(\omega_\mathrm{o})$. This is summarized by the *(surface) scattering equation*,[8]

$$L_\mathrm{o}(\omega_\mathrm{o}) \;=\; \int_{\mathcal{S}^2} L_\mathrm{i}(\omega_\mathrm{i})\, f_\mathrm{s}(\omega_\mathrm{i} \!\rightarrow\! \omega_\mathrm{o})\, d\sigma^\perp(\omega_\mathrm{i})\,. \tag{3.12}$$

This equation can be used to predict the appearance of the surface, given a description of the incident illumination.

3.6.2 The BRDF and BTDF

The BSDF is not a standard concept in radiometry.[9] More typically, the scattered light is subdivided into reflected and transmitted components, which are treated separately. This leads to the definition of the *bidirectional reflectance distribution function* (BRDF), and the *bidirectional transmittance distribution function* (BTDF), denoted f_r and f_t respectively.

The BRDF is obtained by simply restricting f_s to a smaller domain:

$$f_\mathrm{r} : \mathcal{H}^2_\mathrm{i} \times \mathcal{H}^2_\mathrm{r} \rightarrow \mathbb{R}\,,$$

[8] The corresponding equation for one-sided, opaque surfaces is called the *reflectance equation* [Cohen & Wallace 1993, p. 30].

[9] The name appears to have been introduced by Heckbert [Heckbert 1991, p. 26]. Previously, he used the term *bidirectional distribution function* (BDF) [Heckbert 1990], however we feel that this term is more appropriate for a category of such functions, containing the various B*DF's as members.

where $\mathcal{H}^2_{\mathrm{i}}$ and $\mathcal{H}^2_{\mathrm{r}}$ are often called the *incident* and *reflected* hemispheres respectively. In fact, both symbols refer to the same set of directions ($\mathcal{H}^2_{\mathrm{i}} = \mathcal{H}^2_{\mathrm{r}}$), which can be either the upward hemisphere $\mathcal{H}^2_+$, or its complement $\mathcal{H}^2_-$.

The BTDF is defined similarly to the BRDF, by restricting f_{s} to a domain of the form

$$f_{\mathrm{t}} : \mathcal{H}^2_{\mathrm{i}} \times \mathcal{H}^2_{\mathrm{t}} \to \mathbb{R}\,,$$

where the transmitted hemisphere $\mathcal{H}^2_{\mathrm{t}} = -\mathcal{H}^2_{\mathrm{i}}$ is the complement of $\mathcal{H}^2_{\mathrm{i}}$. As before $\mathcal{H}^2_{\mathrm{i}}$ can represent either the upward hemisphere $\mathcal{H}^2_+$, or its complement $\mathcal{H}^2_-$.

Thus, we see that the BSDF is the union of two BRDF's (one for each side of the surface), and two BTDF's (one for light transmitted in each direction). Its main advantage is convenience: we only need to deal with one function, rather than four. The BSDF allows us to write equations that are simple and yet general, capable of describing the scattering from any kind of surface. Surfaces that are purely reflective or transmissive are simply special cases of this formulation. In addition, the BSDF is actually easier to define, since we do not need to specify the hemispherical domains needed by the BRDF and BTDF.

Properties of the BRDF. The BRDF's that describe real surfaces are known to have a number of basic properties. For example, they are *symmetric*:

$$f_{\mathrm{r}}(\omega_{\mathrm{i}} \!\to\! \omega_{\mathrm{o}}) \;=\; f_{\mathrm{r}}(\omega_{\mathrm{o}} \!\to\! \omega_{\mathrm{i}}) \qquad \text{for all } \omega_{\mathrm{i}}, \omega_{\mathrm{o}}\,. \tag{3.13}$$

Because of the symmetry, the notation $f_{\mathrm{r}}(\omega_{\mathrm{i}} \leftrightarrow \omega_{\mathrm{o}})$ is often used. Another property shared by physical BRDF's is *energy conservation*, as embodied by the condition

$$\int_{\mathcal{H}^2_{\mathrm{o}}} f_{\mathrm{r}}(\omega_{\mathrm{i}} \!\to\! \omega_{\mathrm{o}})\, d\sigma^{\perp}(\omega_{\mathrm{o}}) \;\leq\; 1 \qquad \text{for all } \omega_{\mathrm{i}} \in \mathcal{H}^2_{\mathrm{i}}\,. \tag{3.14}$$

Further explanation of BRDF's and their properties can be found in [Nicodemus et al. 1977, p. 5] or [Cohen & Wallace 1993, p. 28].

Note that these simple conditions are unique to reflection, and do not always apply to surfaces that transmit light. Thus, it cannot be assumed that BSDF's or BTDF's satisfy the simple rules above. We will investigate the correct generalization of these properties to arbitrary surfaces in Chapter 6.

3.6.3 Angular parameterizations of the BSDF

It is common to write BSDF's in terms of polar and azimuthal angles, rather than unit direction vectors. We will use this parameterization later in this chapter, to derive the scaling of radiance at a refractive interface (Section 5.2). We show how the two parameterizations are related, and summarize the advantages of the unit vector form.

In the angular parameterization, a direction $\omega \in \mathcal{S}^2$ is represented as a pair of angles (θ, ϕ). The polar angle θ measures the angle between ω and the normal $\mathbf{N}$, while the azimuthal angle ϕ measures the angle between ω and a fixed direction $\mathbf{T}$ lying in the tangent space at $\mathbf{x}$. The angular and vector representations are thus related by

$$\begin{aligned} \cos\theta &= \omega \cdot \mathbf{N}\,, \\ \cos\phi &= \omega \cdot \mathbf{T}\,. \end{aligned}$$

To use this parameterization, we must also know how the angle measures σ and $\sigma^\perp$ are represented. The solid angle σ corresponds to

$$\begin{aligned} d\sigma(\omega) &\equiv \sin\theta\, d\theta\, d\phi \\ &\equiv d!\cos\theta\, d\phi\,, \end{aligned} \tag{3.15}$$

while the projected solid angle $\sigma^\perp$ can be written in a number of forms:

$$\begin{aligned} d\sigma^\perp(\omega) &\equiv |\cos\theta|\, \sin\theta\, d\theta\, d\phi \\ &\equiv |\cos\theta|\, d!\cos\theta\, d\phi \\ &\equiv \sin\theta\, d\mathrm{sin}\,\theta\, d\phi \\ &\equiv (1/2)\, d!\cos^2\theta\, d\phi \\ &\equiv (1/2)\, d\mathrm{sin}^2\,\theta\, d\phi\,. \end{aligned} \tag{3.16}$$

With the angular parameterization, the scattering equation (3.12) thus becomes

$$L_{\mathrm{o}}(\theta_{\mathrm{o}}, \phi_{\mathrm{o}}) = \int_0^{2\pi} \int_0^{\pi} L_{\mathrm{i}}(\theta_{\mathrm{i}}, \phi_{\mathrm{i}})\, f_{\mathrm{s}}(\theta_{\mathrm{i}}, \phi_{\mathrm{i}}, \theta_{\mathrm{o}}, \phi_{\mathrm{o}})\, |\cos\theta_{\mathrm{i}}|\, \sin\theta_{\mathrm{i}}\, d\theta_{\mathrm{i}}\, d\phi_{\mathrm{i}}\,, \tag{3.17}$$

where the other representations (3.16) for the projected solid angle could also be used.

Although the angular representation is common, there are good reasons to prefer the unit vector representation $\omega \in \mathcal{S}^2$. First, note that (θ, ϕ) is a *local* representation of direction, since the angles θ and ϕ depend on the surface normal. When more than one surface point is involved, it is much more convenient to work with direction vectors. Second, the (θ, ϕ) representation creates the impression that many formulas involve trigonometric functions, when in fact they are usually implemented with dot products. Finally, the angular representation depends on an extra parameter (the tangent vector $\mathbf{T}$), which must be chosen arbitrarily since it has no physical significance.

3.7 Introduction to light transport

This section reviews the main concepts of light transport, without getting into too much detail. (These ideas will be defined more precisely in the next chapter, where we reformulate light transport in terms of linear operators.) We discuss the measurement, light transport, and importance transport equations. We also outline the ideas of bidirectional methods for the light transport problem, and explain why they are often the most efficient methods for its solution. Finally, we explain why bidirectional algorithms need to evaluate BSDF's with special care, and we define the useful concept of an *adjoint BSDF*.

3.7.1 The measurement equation

The goal of light transport is to compute a set of real-valued *measurements* $I_1, \ldots, I_M$. For example, in a light transport algorithm that computes an image directly, each measurement I_j represents the value of a single pixel, and M is the number of pixels in the image.

Each measurement corresponds to the output of a hypothetical *sensor* that responds to the radiance $L_{\mathrm{i}}(\mathbf{x}, \omega)$ incident upon it. The response may vary according to the position and direction at which light strikes the sensor; this is characterized by the sensor *responsivity* $W_{\mathrm{e}}(\mathbf{x}, \omega)$. The total response is determined by integrating the product $W_{\mathrm{e}} L_{\mathrm{i}}$, according to

$$I \;=\; \int_{\mathcal{M} \times \mathcal{S}^2} W_{\mathrm{e}}(\mathbf{x}, \omega)\, L_{\mathrm{i}}(\mathbf{x}, \omega)\, dA(\mathbf{x})\, d\sigma^{\perp}_{\mathbf{x}}(\omega)\,. \tag{3.18}$$

This is called the *measurement equation*. Note that there is actually one equation for each measurement I_j, each with a different responsivity function $W_{\mathrm{e}}^{(j)}$ (although we will usually drop the superscript). Also note that we have assumed that the sensors are modeled as part of the scene $\mathcal{M}$, in order that we can integrate over their surface.

3.7.2 The light transport equation

Generally, we are most interested in measuring the *steady-state* or *equilibrium* radiance function for the given scene.[10] It is conventional to solve for the exitant version of this quantity, $L_{\rm o}$, from which the incident radiance $L_{\rm i}$ can be obtained using

$$L_{\rm i}(\mathbf{x}, \omega) \;=\; L_{\rm o}(\mathbf{x}_{\mathcal{M}}(\mathbf{x}, \omega), -\omega)\,.$$

Here $\mathbf{x}_{\mathcal{M}}(\mathbf{x}, \omega)$ is the *ray-casting function*, which returns the first point of $\mathcal{M}$ visible from $\mathbf{x}$ in direction ω.

We can express $L_{\rm o}$ as the sum of *emitted radiance* $L_{\rm e}$, and *scattered radiance* $L_{\rm o,s}$:

$$L_{\rm o} \;=\; L_{\rm e} + L_{\rm o,s}\,.$$

The emitted radiance function $L_{\rm e}(\mathbf{x}, \omega)$ is provided as part of the scene description, and represents all of the light sources in the scene. On the other hand, $L_{\rm o,s}$ is determined using the scattering equation (3.12), according to

$$L_{\rm o,s}(\mathbf{x}, \omega_{\rm o}) \;=\; \int_{\mathcal{S}^2} L_{\rm i}(\mathbf{x}, \omega_{\rm i})\, f_{\rm s}(\mathbf{x}, \omega_{\rm i} \!\rightarrow\! \omega_{\rm o})\, d\sigma^{\perp}_{\mathbf{x}}(\omega_{\rm i})\,.$$

By putting these equations together, we get a complete specification of the light transport problem. The most interesting feature is that $L_{\rm o}$ and $L_{\rm i}$ have been defined in terms of each other; commonly their definitions are combined to obtain

$$L_{\rm o}(\mathbf{x}, \omega_{\rm o}) \;=\; L_{\rm e}(\mathbf{x}, \omega_{\rm o}) + \int_{\mathcal{S}^2} L_{\rm o}(\mathbf{x}_{\mathcal{M}}(\mathbf{x}, \omega_{\rm i}), -\omega_{\rm i})\, f_{\rm s}(\mathbf{x}, \omega_{\rm i} \!\rightarrow\! \omega_{\rm o})\, d\sigma^{\perp}_{\mathbf{x}}(\omega_{\rm i})\,, \qquad (3.19)$$

which is known as the *light transport equation*. Since $L_{\rm i}$ does not appear in this equation, the subscript on $L_{\rm o}$ is usually dropped. The form of this equation naturally leads to recursive solutions (the essence of traditional Monte Carlo methods).

[10] Since light travels so much faster than the everyday objects around us, equilibrium is achieved very quickly after any changes to the environment. Effectively, the world we perceive is always in equilibrium (with respect to light transport).

3.7.3 Importance and adjoint methods

As we have presented them, the transport rules apply to the scattering of light, as emitted by the sources. However, the transport rules can be applied equally well to the sensors, by treating the responsivity $W_{\mathrm{e}}(\mathbf{x}, \omega)$ as an emitted quantity. In this context, $W_{\mathrm{e}}(\mathbf{x}, \omega)$ is called an *emitted importance function*, since W_{e} specifies the "importance" of the light arriving along each ray to the corresponding measurement I.

This idea is the basis of *adjoint* methods, which apply the transport rules to importance rather than radiance. These methods start with the emitted importance $W_{\mathrm{e}}(\mathbf{x}, \omega)$, and solve for the *equilibrium importance function* $W(\mathbf{x}, \omega)$, according to the *importance transport equation*

$$W(\mathbf{x}, \omega) \;=\; W_{\mathrm{e}}(\mathbf{x}, \omega) + \int_{\mathcal{S}^2} W(\mathbf{x}_{\mathcal{M}}(\mathbf{x}, \omega_{\mathrm{i}}), -\omega_{\mathrm{i}})\, f_{\mathrm{s}}(\mathbf{x}, \omega_{\mathrm{o}} \!\rightarrow\! \omega_{\mathrm{i}})\, d\sigma^{\perp}_{\mathbf{x}}(\omega_{\mathrm{i}})\,. \tag{3.20}$$

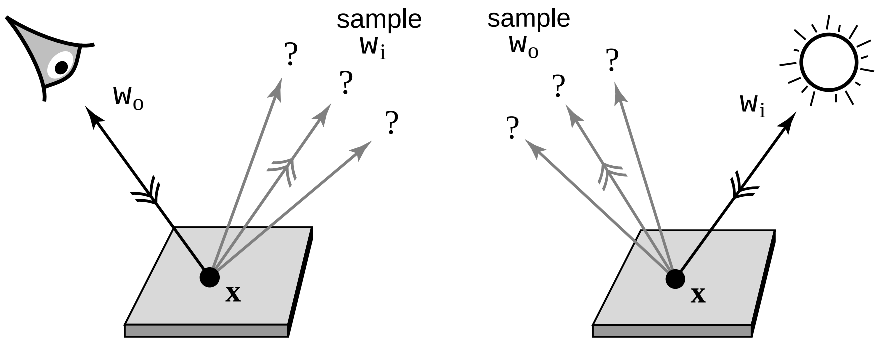
This equation is virtually identical to the light transport equation (3.19), except that the directional arguments to the BSDF have been exchanged.

Given the equilibrium importance W, measurements are computed by integrating the product $W L_{\mathrm{e}}$ (similar to (3.18)). Note that while there is only one equilibrium radiance function, there can be many different equilibrium importance functions (one for each sensor). This is an important difference between direct and adjoint methods.

3.7.4 Bidirectional methods

Many recent algorithms combine features from both of these approaches, leading to *bidirectional* light transport methods. The computation is guided by the viewing information (sensors), as well as the lighting information (sources). This allows these algorithms to be more efficient, since they can do less work in regions that are dark or that are not visible. This concept is similar to certain planning problems in artificial intelligence, where the objective is to get from an initial state to a goal, given some set of possible actions. It is possible to reduce the search complexity by simultaneously working forward from the initial state, and backward from the goal, until the two searches meet somewhere in the middle.

Bidirectional algorithms can appear in a number of different forms. *Importance-driven*



(a) Path tracing **(b)** Particle tracing

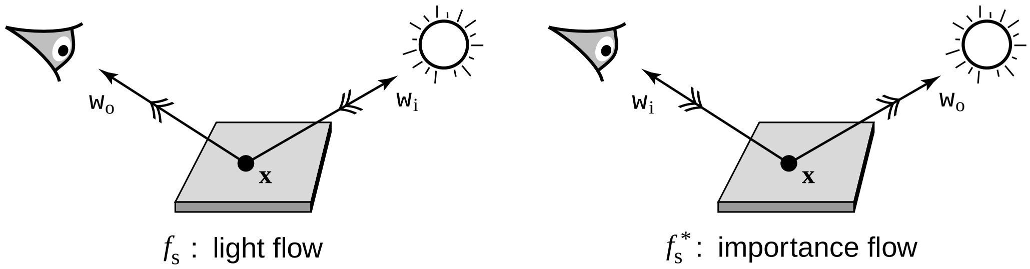
Figure 3.2: Path tracing and particle tracing sample the BSDF in different ways. **(a)** For path tracing, the direction ω_o is given (it points toward the previous vertex on a path leading to a sensor). The path is extended by sampling a direction ω_i according to the BSDF. **(b)** For particle tracing, the direction ω_i is given (pointing along a path toward a light source), and the path is extended by sampling a direction ω_o.

methods use viewing information to guide mesh refinement, by increasing the mesh resolution in regions where the equilibrium importance is high (since these regions have the greatest influence on the desired set of measurements). With Monte Carlo approaches, bidirectional methods often combine *path tracing*, where the transport equation is sampled starting from the sensors, and *particle tracing*, where sampling begins at the light sources.

In one way or another, almost all recent light transport algorithms have taken a bidirectional approach. These include finite element approaches [Smits et al. 1992, Schrӧder & Hanrahan 1994, Christensen et al. 1996], multi-pass methods [Chen et al. 1991, Zimmerman & Shirley 1995], particle tracing algorithms [Heckbert 1990, Pattanaik & Mudur 1995, Shirley et al. 1995, Jensen 1996], and bidirectional path tracing [Lafortune & Willems 1993, Veach & Guibas 1994, Veach & Guibas 1995].

3.7.5 Sampling and evaluation of non-symmetric BSDF’s

Scene models often contain materials whose BSDF is not symmetric, i.e. for which $f_\text{s}(\omega_\text{i} \rightarrow \omega_\text{o}) \neq f_\text{s}(\omega_\text{o} \rightarrow \omega_\text{i})$. Great care must be taken when such materials are used with bidirectional algorithms, because in this case the transport rules for light and importance are



(a) The normal BSDF f_s.

(b) The adjoint BSDF f_s^*.

Figure 3.3: By adopting the convention that ω_i is always the sampled direction, the BSDF f_s and its adjoint f_s^* are used for different purposes. **(a)** The BSDF $f_\mathrm{s}(\omega_\mathrm{i} \to \omega_\mathrm{o})$ is used for radiance evaluation, and to scatter importance particles. **(b)** The adjoint BSDF $f_\mathrm{s}^*(\omega_\mathrm{i} \to \omega_\mathrm{o})$ is used for importance evaluation, and to scatter light particles.

different. Formally, this can be seen by noting that the light transport equation (3.19) and the importance transport equation (3.20) are identical, except that the directional arguments to the BSDF have been exchanged. Thus if the BSDF is not symmetric, then light and importance satisfy different transport equations. From another point of view, recall that the BSDF was defined in terms of light propagation: light flows from the incoming direction ω_i to the outgoing direction ω_o. Thus importance flows from ω_o to ω_i, since it is transported in the opposite direction as light. Similarly, different scattering rules must be used for particle tracing and path tracing to obtain correct results when non-symmetric BSDF's are present (see Figure 3.2). Thus, bidirectional algorithms must take care when evaluating or sampling the BSDF, to ensure that ω_i and ω_o are ordered correctly.

In the next few chapters, we will study non-symmetric BSDF's and their consequences for bidirectional algorithms in detail.

3.7.6 The adjoint BSDF

Given an arbitrary BSDF f_s, the *adjoint BSDF* f_s^* is defined by

$$f_\mathrm{s}^*(\omega_\mathrm{i} \to \omega_\mathrm{o}) \;=\; f_\mathrm{s}(\omega_\mathrm{o} \to \omega_\mathrm{i}) \qquad \text{for all } \omega_\mathrm{i}, \omega_\mathrm{o} \in \mathcal{S}^2 \,. \tag{3.21}$$

The main advantage of the adjoint BSDF is that it lets the importance transport equation (3.20) have the same form as the light transport equation (3.19). Recall that the only difference between these two equations is that the arguments to the BSDF are exchanged ($f_{\mathrm{s}}(\omega_{\mathrm{o}} \to \omega_{\mathrm{i}})$ instead of $f_{\mathrm{s}}(\omega_{\mathrm{i}} \to \omega_{\mathrm{o}})$). By using the adjoint BSDF f_{s}^* in the importance transport equation, this difference is eliminated: the two equations have exactly the same form, but they use different BSDF's (see Figure 3.3.)

The adjoint BSDF also provides a useful convention for sampling. Recall that in path tracing, we sample the BSDF to determine the incident direction ω_{i} (since ω_{o} is given). We extend this idea, by adopting the convention that ω_{i} is *always* the sampled direction during a random walk. We refer to the opposite situation (where ω_{i} is provided, and ω_{o} is sampled) as *sampling the adjoint BSDF*. For example, according to this convention the adjoint BSDF is used to scatter light particles.

We also mention two other techniques that can be used in bidirectional algorithms. The first of these is *importance particle tracing*, in which particles are emitted from the sensors and scattered throughout the environment, in order to obtain a set of samples that represent the equilibrium importance. This process is similar to ordinary particle tracing, except that importance is used instead of light. This implies that importance particles should be scattered using the ordinary BSDF f_{s}. The second technique is *importance evaluation*, in which the equilibrium importance on a ray $(\mathbf{x}, \omega)$ is estimated by recursively sampling the importance transport equation. This is similar to the evaluation of radiance using path tracing, except that the adjoint BSDF f_{s}^* is used instead of f_{s}.

To summarize, the adjoint BSDF is used for importance evaluation and for scattering light particles (i.e. sampling processes that start at a light source), while the normal BSDF is used for radiance evaluation and for scattering importance particles (sampling processes that start at a sensor). These rules will be justified formally in Chapter 4.

Appendix 3.A Field and surface radiance functions

In this appendix we consider the *field* and *surface* radiance functions defined by Arvo [1995, p. 28], and compare them with the incident and exitant radiance functions described in Section 3.5. Basically, field radiance L_{f} is similar to incident radiance L_{i}, while surface radiance L_{s} is similar to exitant radiance L_{o}. The main difference is that L_{f} and L_{s} are defined only in the context of one-sided (reflective) surfaces, which allows them to be defined as two halves of a single radiance distribution L.

To define field and surface radiance precisely, let S be a surface bounding an opaque object, and consider the radiance distribution $L(\mathbf{x}, \omega)$ at a point $\mathbf{x} \in S$. Arvo [1995] observes that since scattering occurs on only one side of S, the direction of ω can be used to distinguish incident photons from exitant ones: if ω is in the upward hemisphere $\mathcal{H}^2_+(\mathbf{x})$, then $L(\mathbf{x}, \omega)$ refers to radiance leaving the surface, and otherwise $L(\mathbf{x}, \omega)$ refers to radiance arriving at the surface. Applying this observation, he proposed that $L(\mathbf{x}, \omega)$ is naturally partitioned into *surface radiance* $L_{\mathrm{s}}(\mathbf{x}, \omega)$ and *field radiance* $L_{\mathrm{f}}(\mathbf{x}, \omega)$, according to whether $\omega \cdot \mathbf{N}(\mathbf{x})$ is positive or negative respectively.

However, there are several important differences between incident/exitant radiance and field/surface radiance. First, the sense of the direction parameter ω is reversed for L_{f} as compared to L_{i}:

$$L_{\mathrm{f}}(\mathbf{x}, \omega) \;=\; L_{\mathrm{i}}(\mathbf{x}, -\omega)\,.$$

The field radiance definition L_{f} would appear to be more natural, since ω corresponds to the direction of travel of the photons. However, the L_{i} definition has two important advantages. At reflective surfaces, it corresponds to the convention assumed by most BRDF formulas, where ω_{i} and ω_{o} both point outward. More significantly, the L_{i} definition causes certain natural transport operators to become self-adjoint (namely the $\mathbf{G}$ and $\mathbf{K}$ operators defined in Section 4.3), which increases the symmetry between the equations governing light and importance transport.

A second difference is that L_{i} and L_{o} are defined for *two-sided* surfaces, e.g. those that allow both reflection and transmission. For these surfaces, ω cannot be used to distinguish between incident and exitant photons, since L_{i} and L_{o} are both defined for all $\omega \in \mathcal{S}^2$. Instead, the two sets of photon events must be distinguished using the time dimension, as we have outlined above.

Finally, field and surface radiance are defined only at surfaces, while incident and exitant radiance are defined in space as well. (The distinction between L_{i} and L_{o} is still useful in this context, since it can be used to define self-adjoint operators for volume scattering.)

Appendix 3.B Measure-theoretic radiometry

Typically, radiometric quantities are defined using "infinitesimals" and limit arguments. Arvo [Arvo 1995, Chapter 2] has taken a different approach, by proposing a set of axioms that correspond to the observable behavior of photons, and then deriving radiometric quantities using the tools of measure theory. His analysis focused on the spatial distribution of steady-state, monochromatic radiation, and led to a measure-theoretic definition of the *phase space density* (defined below). In this section we show how to extend his techniques to a more general class of radiometric quantities: for example, we give measure-theoretic definitions of *spectral radiance* and *spectral radiant sterisent*.

3.B.1 Measure spaces

A *measure space* is a triple $(\mathbb{P}, \mathcal{P}, \varrho)$, where $\mathbb{P}$ is a set (the *underlying set* of the measure space), $\mathcal{P}$ is a collection of subsets of $\mathbb{P}$ (the *measurable sets*), and $\varrho : \mathcal{P} \to [0, \infty]$ is a non-negative, countably additive set function (the *measure function*, or simply the *measure*). The countably additive property means that

$$\varrho\left(\bigcup_{i=1}^{\infty} D_i\right) \;=\; \sum_{i=1}^{\infty} \varrho(D_i)$$

whenever the D_i are mutually disjoint measurable sets.

The measurable sets form a *σ-algebra*, meaning that $\mathcal{P}$ contains $\mathbb{P}$, and is closed under the operations of complementation and countable unions. For technical reasons, $\mathcal{P}$ is generally a proper subset of $2^{\mathbb{P}}$, that is, some sets are not measurable. However, for the measure spaces we are interested in (those constructed as the product of Lebesgue measures), the unmeasurable sets represent pathological situations that can be ignored in practice.

Sometimes, the measures we consider will not be finite; that is, $\varrho(\mathbb{P}) = \infty$. However, they will always have the weaker property of being *σ-finite*, meaning that there is an infinite sequence $D_1, D_2, \ldots$ of measurable sets such that

$$\bigcup_{i=1}^{\infty} D_i \;=\; \mathbb{P}\,,$$

and $\varrho(D_i)$ is finite for all i. That is, a σ-finite measure space is one that can be decomposed into countably many regions, each with finite measure.

3.B.2 The photon event space

To define a radiometric quantity, we first choose an appropriate*photon event space*. This is the subset $\mathbb{P} \subset \Psi$ of the trajectory space containing all possible locations of the photon events we wish to count. Thus, $\mathbb{P}$ depends on the definition of a photon event; by defining photon events in different ways, we will obtain different radiometric quantities. For example, consider the case of*volume emission* (e.g. the light emitted by a fire). Without knowledge of the specific scene geometry, we must assume that a photon could be emitted from any point in$\mathbb{R}^3$, in any direction, at any wavelength, at any time; thus we would set $\mathbb{P} = \Psi$ (the whole trajectory space). On the other hand, if photon events were defined as crossings of a hypothetical surface$S \subset \mathbb{R}^3$, then the photon event space would be

$$\mathbb{P} \;=\; \mathbb{R} \times S \times \mathcal{S}^2 \times \mathbb{R}^+ \,.$$

In this example, $\mathbb{P}$ is a 6-dimensional manifold within the 7-dimensional trajectory spaceΨ.

3.B.3 The geometric measure

Next we define a measure ϱ on the photon event space, called the*geometric measure*, which will be used to measure the density of photon events. It will normally be defined as product of the natural Lebesgue measures on the components of$\mathbb{P}$. For example, in the case of volume emissionϱ is given by

$$\varrho \;=\; l \times v \times \sigma \times l^+ \,,$$

where l and l^+ are the usual length measures on$\mathbb{R}$ and $\mathbb{R}^+$ respectively, and v is the usual volume measure on $\mathbb{R}^3$. Note that this definition also establishes the geometrically measurable sets$\mathcal{P}$, according to the usual rules for product measures [Halmos 1950, p. 140].[11]

3.B.4 The energy measure

To count the photon events in various regions of$\mathbb{P}$, we also define an *energy content function*

$$Q : \mathcal{P} \to [0, \infty] \,.$$

[11] Technically, we work with the *completion* of the product measure, which augments $\mathcal{P}$ to include sets of the form $D \bigtriangleup N$, where $\bigtriangleup$ denotes the symmetric difference of two sets, D is a measurable set, and N is an arbitrary subset of a set of measure zero.

To each measurable set D of the photon event space, it assigns a non-negative real number $Q(D)$ that measures the total energy of the photon events in D. The function Q is assumed to obey the following physically plausible axioms (see [Arvo 1995, p. 19]):

(A1) $Q : \mathcal{P} \to [0, \infty]$

(A2) $Q\left(\bigcup_{i=1}^{\infty} D_i\right) = \sum_{i=1}^{\infty} Q(D_i)$ for mutually disjoint $\{D_i\} \subset \mathcal{P}$

(A3) $\varrho(D) < \infty \implies Q(D) < \infty$

(A4) $\varrho(D) = 0 \implies Q(D) = 0$

Axiom (A1) states that every region contains a non-negative quantity of energy. Axiom (A2) states that Q is countably additive; that is, if we consider a countable set of disjoint regions D_i, the energy contained their union is simply the sum of their individual energies. Together, (A1) and (A2) imply that Q is a non-negative, countably additive set function, so that by definition Q is a measure (on the same measurable sets for which ϱ is defined).

Axiom (A3) states that every region with finite ϱ-measure contains a finite quantity of energy. Intuitively, this says that the energy density is finite everywhere, a concept that will be made more precise below. From a measure-theoretic point of view, it ensures that the σ-finite property of ϱ carries over to Q.

Finally, (A4) states that Q is *continuous* with respect to ϱ, meaning that every set with zero ϱ-measure also has zero Q-measure. This important property allows the “ratio” of two measures to be defined rigorously, as we shall see below.

By translating these axioms into the language of measure theory, we obtain the following theorem (cf. Arvo, Theorem 1 [Arvo 1995, p. 22]):

Theorem 3.1 (Existence of Energy Measures). *Given a photon event space $\mathbb{P}$ with geometric measure ϱ, and an energy content function Q satisfying axioms (A1), (A2), (A3), and (A4), then Q defines a positive σ-finite measure over $\mathbb{P}$, and Q is continuous with respect to ϱ.*

Thus, we will now refer to Q as the *energy measure* on $\mathbb{P}$.

3.B.5 Defining radiometric quantities as a ratio of measures

Loosely speaking, a radiometric quantity can now be defined by measuring the density of Q with respect to ϱ, i.e. the ratio $dQ/d\varrho$ for a region D that becomes arbitrarily small. This idea can be made precise by means of the Radon-Nikodym theorem [Halmos 1950, p. 128].[12]

Theorem 3.2 (Radon-Nikodym). *If $(\mathbb{P}, \mathcal{P}, \varrho)$ is a σ-finite measure space, and if a σ-finite measure Q on $\mathcal{P}$ is continuous with respect to ϱ, then there exists a non-negative, real-valued, ϱ-measurable function f on $\mathbb{P}$ such that*

$$Q(D) \;=\; \int_D f \, d\varrho$$

for every measurable set $D \in \mathcal{P}$. The function f is unique up to a set of ϱ-measure zero.

The function f is called the *Radon-Nikodym derivative* of Q with respect to ϱ, denoted

$$f \;=\; \frac{dQ}{d\varrho}\,. \tag{3.22}$$

This notation emphasizes its similarity with ordinary differentiation, with which it shares many properties.

Using the Radon-Nikodym theorem, we can thus define a function f corresponding to the density of photon events. The meaning of this density obviously depends on how the events are defined. However, we can summarize the fact of its existence as follows (cf. Arvo, Theorem 3 [Arvo 1995, p. 23]):

Theorem 3.3 (Existence of Energy Density). *Given a photon event space $\mathbb{P}$ with geometric measure ϱ, and an energy content function Q satisfying axioms (A1), (A2), (A3), and (A4), then there exists a ϱ-measurable function $f : \mathbb{P} \to (0, \infty)$, which is unique to within a set of ϱ-measure zero, satisfying*

$$Q(D) \;=\; \int_D f \, d\varrho\,,$$

where $D \in \mathcal{P}$ is a measurable subset of $\mathbb{P}$.

3.B.6 Examples of measure-theoretic definitions

We now give several examples showing how these concepts can be applied.

3.B.6.1 Spectral radiant sterisent

Consider again the case of volume emission. Recall from Section 3.B.2 that the photon event space is the whole trajectory space $\mathbb{P} = \Psi$, while the geometric measure is $\varrho = l \times v \times \sigma \times l^+$. By taking

[12] Notice that we have restricted our definition of a measure space to *positive, total* measures, which simplifies the statement of the theorem somewhat.

the derivative $dQ/d\varrho$, we obtain a quantity

$$L_\lambda^* \;=\; \frac{dQ}{d\varrho} \;=\; \frac{dQ}{dl\, dv\, d\sigma\, dl^+}\,.$$

This quantity is called *spectral radiant sterisent* [Nicodemus 1978, p. 55], and has units of power per unit area per unit solid angle per unit wavelength $[\mathrm{W} \cdot \mathrm{m}^{-3} \cdot \mathrm{sr}^{-1} \cdot \mathrm{nm}^{-1}]$. It is used for the measurement of emission, scattering, and absorption within volumes.

3.B.6.2 Spectral phase space density

As another example, consider the events defined by intersecting the photon trajectories with the surface $t = t_0$. This allows us to measure the instantaneous spatial distribution of the photons, a concept that is particularly useful for steady-state systems. This was the situation studied by Arvo [1995], who developed a measure-theoretic *phase space density* for photons distributed in $\mathbb{R}^3 \times \mathcal{S}^2$.

In our framework, the event space for this situation is

$$\mathbb{P} \;=\; \{t_0\} \times \psi\,,$$

where $\{t_0\}$ denotes the set containing the single value t_0, and recalling that ψ is the phase space $\psi = \mathbb{R}^3 \times \mathcal{S}^2 \times \mathbb{R}^+$. The geometric measure ϱ is just the natural measure on the phase space ψ, with a slight technical modification to account for presence of the fixed time t_0:

$$\varrho \;=\; \Lambda_{t_0} \times v \times \sigma \times l^+\,,$$

where $\Lambda_{t_0}(D) = 1$ if $t_0 \in D$ and $\Lambda_{t_0}(D) = 0$ otherwise. Then the quantity

$$u_\lambda \;=\; \frac{dQ}{d\varrho} \;=\; \frac{dQ}{dv\, d\sigma\, dl^+} \tag{3.23}$$

measures the density of energy with respect to volume, direction, and wavelength $[\mathrm{J} \cdot \mathrm{m}^{-3} \cdot \mathrm{sr}^{-1} \cdot \mathrm{nm}^{-1}]$. We call u_λ the *spectral phase space density.* It is similar to the phase space density u described by Arvo [1995], except that we have also taken the derivative with respect to wavelength.

3.B.6.3 Spectral radiance

As a final example, define the photon events as crossings of a surface S. The event space is

$$\mathbb{P} \;=\; \mathbb{R} \times S \times \mathcal{S}^2 \times \mathbb{R}^+\,,$$

with the geometric measure defined by

$$\varrho \;=\; l \times A \times \sigma^{\perp}_{\mathbf{x}} \times l^{+} \,.$$

Notice that $A \times \sigma^{\perp}_{\mathbf{x}}$ is simply the measure that was used to define radiance. Thus the density

$$L_\lambda \;=\; \frac{dQ}{d\varrho} \;=\; \frac{dQ}{dl\, dA\, d\sigma^{\perp}_{\mathbf{x}}\, dl^{+}}$$

corresponds to *spectral radiance* as defined earlier (3.9). Notice that although this definition is valid only on the surface S, the choice of S was arbitrary. Thus we can use this equation to define L_λ anywhere in the trajectory space Ψ.

3.B.7 Discussion: fundamental vs. derived quantities

It is sometimes claimed that spectral radiance is the "fundamental" radiometric quantity, from which all others can be derived. As we have seen, this is not so. All of the three quantities defined in Section 3.B.6 are fundamental, because they measure *different sets of photon events*. It is not possible to obtain one from another by integration. Each quantity must be defined independently, by first specifying the photon events, and then describing their density using a Radon-Nikodym derivative.[13]

There is not even a unique geometric space that we can use, since different kinds of photon events require different geometric measures. In some cases, the measure can be defined on all of Ψ (as with L^*_λ), while in other cases it must be defined on a lower-dimensional subset of Ψ (as with u_λ and L_λ).

Note that many "derived" quantities (i.e. one that is obtained by integrating a fundamental quantity, as we did in Section 3.4 to obtain radiance from spectral radiance) can be interpreted directly as Radon-Nikodym derivatives, by reducing the dimension of the underlying measure space. For example, to interpret radiance as a Radon-Nikodym derivative, we could redefine the trajectory space to be $\mathbb{R} \times \mathbb{R}^3 \times \mathcal{S}^2$ (omitting the wavelength parameter), and then proceed as for spectral radiance

[13] Note that by making additional assumptions, it is often possible to express one fundamental quantity in terms of another. For example, Arvo [1995, p. 26] shows how radiance can be defined in terms of the phase space density u, by assuming that all photons travel at the same speed c. He then observes that radiance and phase space density are related according to $L_\lambda = c\, u_\lambda$, where c is the speed of light. (A similar observation appears in [Milne 1930, p. 76].)

Note that this relationship is only true in a vacuum, since in general photons travel at the speed c/η (where η is the local refractive index, which may vary with position). It is even possible that photons at same point in space will travel at different speeds (i.e. if they have different wavelengths, in a dispersive medium). Thus in general, u_λ and L_λ cannot be derived from each other without additional assumptions, so that we consider both of them to be fundamental quantities.

(Section 3.B.6). This technique can be used to give rigorous meaning to the various derivative notations we used in Section 3.4, such as $E = d^2Q \,/\, (dt\, dA)$.

Chapter 4

A General Operator Formulation of Light Transport

The goal of this chapter is to develop a rigorous theoretical basis for bidirectional light transport algorithms. Current frameworks do not adequately describe the relationships between light and importance transport; between finite element, recursive evaluation, and particle tracing approaches; or between incident and exitant transport quantities, especially when materials with non-symmetric BSDF's are used. As a result, given a bidirectional algorithm that uses some combination of these features, it can be difficult to verify whether it actually solves the original transport equations. This can lead to significant mistakes when bidirectional algorithms are implemented, as we will see in Chapter 5.

To remedy these problems, we need a better theoretical framework for light transport calculations. This theory should clearly state the relationships between the various solution techniques mentioned above, using only a small number of basic concepts. It should also show how these techniques are affected by non-symmetric scattering, and specify a set of rules that allow correct results to be obtained. All components of the framework should be expressed in terms of standard mathematical concepts, and the notation should be concise and yet rigorous.

In this chapter, we develop a light transport framework that addresses these goals. It concisely expresses the relationships between light and importance transport, in both their incident and exitant forms, and also their relationship to particle tracing. The fundamental building blocks used are measures, function spaces, inner products, and linear operators. Our work builds directly on the elegant formulation of Arvo [1995], who considered light transport among reflective surfaces with symmetric BRDF's. We also incorporate ideas from Christensen et al. [1993], Schröder & Hanrahan [1994], and Pattanaik & Mudur [1995].

However, many aspects of our framework are new. Most importantly, we do not make any assumptions about the symmetry of BSDF's. This leads to a framework with a richer structure than previous approaches. There are four distinct transport quantities L_{i}, L_{o}, W_{i}, W_{o}, corresponding to incident/exitant radiance/importance. For each of these quantities, there is a distinct transport operator and measurement equation. All of these are related in a simple way, since they are constructed from just two basic elements: the *scattering* and *propagation* operators, which describe independent aspects of the light transport process. This additional structure actually helps to clarify the relationships among transport quantities, since we can see which relationships are fundamental, and which depend on the symmetry of the BSDF.

There are several other contributions. We characterize particle tracing in a new and more useful way, as a condition on the probability distribution of a set of weighted sample rays. We also introduce the *ray space* abstraction, which simplifies the notation and clarifies the structure of light transport calculations. Finally, we point out that *incident* rather than *field* radiance functions must be used to make certain transport operators self-adjoint.

This chapter is organized as follows. We start by defining the ray space and reviewing some useful properties of functions on ray space. Next, we describe the scattering and propagation operators, and we show how they can be used to represent light transport. We then consider sensors and measurements, and show that the scattering and propagation operators can also be used for importance transport. In Section 4.7, we give a summary of the complete transport framework.

Appendix 4.A considers particle tracing algorithms, and describes a new condition that can be used to verify their correctness. Finally, Appendix 4.B gives an analysis of the inverses, adjoints, and norms of the operators we have defined.

4.1 Ray space

We define the ray space and throughput measure, which together form a natural basis for light transport calculations. We show that it is possible to represent the ray space in more than one way, and we also discuss the advantages of defining the ray space abstractly, as opposed to using an explicit representation of rays.

The *ray space* $\mathcal{R}$ consists of all rays that start at points on the scene surfaces. Formally, $\mathcal{R}$ is the Cartesian product

$$\mathcal{R} \;=\; \mathcal{M} \times \mathcal{S}^2 \,, \tag{4.1}$$

where as usual, $\mathcal{M}$ is the set of surfaces in the scene, and $\mathcal{S}^2$ is the set of all unit direction vectors. The ray $\mathbf{r} = (\mathbf{x}, \omega)$ has origin $\mathbf{x}$ and direction ω. The reason for requiring the origin to lie on a surface is that in the absence of participating media, the radiance along a given ray is constant. Thus instead of representing the radiance at every point in an environment, it is sufficient to represent the radiance leaving surfaces.

The throughput measure. We define a measure μ on $\mathcal{R}$, called the *throughput measure*, that is used to integrate functions on ray space. Consider a small bundle of rays around a central ray $\mathbf{r} = (\mathbf{x}, \omega)$, such that the origins of these rays occupy an area dA, and their directions lie within a solid angle of $d\sigma$. Then the throughput of this small bundle is defined as

$$d\mu(\mathbf{r}) \;=\; d\mu(\mathbf{x}, \omega) \;=\; dA(\mathbf{x})\, d\sigma^{\perp}_{\mathbf{x}}(\omega) \,, \tag{4.2}$$

that is, μ is simply the product of the area and projected solid angle measures. This is known as the *differential form* of the throughput measure. Note that μ is invariant under Euclidean transformations, which makes it unique up to a constant factor [Ambartzumian 1990, p. 51].

To define $\mu(D)$ for a general set of rays $D \subset \mathcal{R}$, we integrate the differential measure (4.2) over the domain D:

$$\mu(D) \;=\; \int_D dA(\mathbf{x})\, d\sigma^{\perp}_{\mathbf{x}}(\omega) \,,$$

which can be written more explicitly as

$$\mu(D) \;=\; \int_{\mathcal{M}} \sigma^{\perp}_{\mathbf{x}}(D_{\mathbf{x}})\, dA(\mathbf{x}) \qquad \text{where} \qquad D_{\mathbf{x}} \;=\; \{\omega \mid (\mathbf{x}, \omega) \in D\} \,.$$

The quantity $\mu(D)$ measures the light-carrying capacity of a bundle of rays D, and corresponds to the classic radiometric concept of *throughput* [Steel 1974, Nicodemus 1976, Cohen & Wallace 1993]. The measure μ is also similar to the usual measure on lines in $\mathbb{R}^3$ (see [Ambartzumian 1990]). However, note that the measures on line space and ray space are not the same, since unlike line space, the ray space $\mathcal{R}$ can contain distinct rays that are colinear (corresponding to lines that intersect $\mathcal{M}$ at more than one point).

The differential form (4.2) of the throughput measure can be written in several alternative forms that are sometimes useful. By expanding the definition (3.1) of projected solid angle, we get

$$\begin{aligned} d\mu(\mathbf{x}, \omega) &= |\omega \cdot \mathbf{N}_{\mathrm{g}}(\mathbf{x})| \, dA(\mathbf{x}) \, d\sigma(\omega) \,, && (4.3) \\ &= dA^{\perp}_{\omega}(\mathbf{x}) \, d\sigma(\omega) \,, && (4.4) \end{aligned}$$

where $A^{\perp}$ is the projected area measure (3.7). All of these definitions are equivalent.

The throughput measure also allows us to define radiance in a simpler and more natural way, namely as power per unit throughput:

$$L(\mathbf{r}) = \frac{d\Phi(\mathbf{r})}{d\mu(\mathbf{r})} \,, \qquad (4.5)$$

It is easy to check that this definition is equivalent to the ones given in Section 3.4.3.

Other representations of ray space. Although we will most often use the representation $\mathbf{r} = (\mathbf{x}, \omega)$ for a ray, it is possible to represent the ray space in other ways. For example, we could define $\mathcal{R}$ as

$$\mathcal{R} = \mathcal{M} \times \mathcal{M} \,, \qquad (4.6)$$

so that each ray is a pair $\mathbf{r} = \mathbf{x} \!\rightarrow\! \mathbf{x}'$ (where the arrow notation denotes the direction of the ray). Notice that there is some redundancy in this representation, since the rays $\mathbf{x} \!\rightarrow\! \mathbf{x}'$ and $\mathbf{x} \!\rightarrow\! \mathbf{x}''$ are equivalent whenever $\mathbf{x}'$ and $\mathbf{x}''$ lie in the same direction from $\mathbf{x}$. However, this redundancy is sometimes useful: for example, it allows us to construct a basis for functions on ray space as a tensor product of bases defined on the scene surfaces. Also notice that with this representation, there is no way to represent light that radiates out to infinity: thus, it is most useful when $\mathcal{M}$ is a closed environment, and we are only interested in light transport between elements of $\mathcal{M}$.

Even when $\mathcal{R}$ is represented in different ways, the throughput measure μ should be understood to have the same meaning. For example, with the representation above, μ is defined by

$$d\mu(\mathbf{x} \to \mathbf{x}') \;=\; V(\mathbf{x} \leftrightarrow \mathbf{x}') \, \frac{\cos(\theta) \, \cos(\theta')}{\|\mathbf{x} - \mathbf{x}'\|^2} \, dA(\mathbf{x}) \, dA(\mathbf{x}') \, . \tag{4.7}$$

Here θ and θ' are the angles between the segment $\mathbf{x} \leftrightarrow \mathbf{x}'$ and the surface normals at $\mathbf{x}$ and $\mathbf{x}'$ respectively, while $V(\mathbf{x} \leftrightarrow \mathbf{x}')$ is the visibility function, which is 1 if $\mathbf{x}$ and $\mathbf{x}'$ are mutually visible and 0 otherwise.[1] As usual, the notation $\mathbf{x} \to \mathbf{x}'$ indicates the direction of a ray, and $f(\mathbf{x} \leftrightarrow \mathbf{x}')$ indicates a symmetric function.

Advantages of the ray space abstraction. There are several reasons to use the abstract representation $\mathbf{r} \in \mathcal{R}$ for rays, rather than writing $(\mathbf{x}, \omega)$ explicitly. First, it clarifies the structure of radiometric formulas, by hiding the details of the ray representation. Second, it emphasizes that the representation is a superficial decision that can easily be changed. Finally, it allows us to define concepts whose meanings do not depend on how the rays are represented, e.g. the throughput measure μ.

4.2 Functions on ray space

The distribution of radiance or importance in a given scene can be represented as a real-valued function on ray space, i.e. a function of the form

$$f : \mathcal{R} \to \mathbb{R} \, .$$

In this section, we study the properties of such functions (e.g. norms and inner products), and review some terminology related to *function spaces* (i.e. collections of functions that all have some specified property). These ideas will be used later to analyze the properties of light transport operators.

[1] That is, $V(\mathbf{x} \leftrightarrow \mathbf{x}') = 1$ if the open line segment between $\mathbf{x}$ and $\mathbf{x}'$ does not intersect $\mathcal{M}$. Note that the visibility factor can be removed from the definition (4.7), by restricting $\mathcal{R}$ to contain only those rays where $V(\mathbf{x} \leftrightarrow \mathbf{x}') = 1$.

Norms. We restrict our attention to the L_p *norms*, which are defined by

$$\|f\|_p \;=\; \left(\int_{\mathcal{R}} |f(\mathbf{r})|^p \, d\mu(\mathbf{r}) \right)^{1/p} , \tag{4.8}$$

where p is a positive integer. In the limit as $p \to \infty$, we obtain the L_∞ *norm*:

$$\|f\|_\infty \;=\; \operatorname*{ess\,sup}_{\mathbf{r} \in \mathcal{R}} |f(\mathbf{r})| , \tag{4.9}$$

where $\operatorname{ess\,sup}$ denotes the *essential supremum*, i.e. the smallest real number m such that $f(\mathbf{r}) \le m$ almost everywhere.[2] The most commonly used norms are the L_1, L_2, and L_∞ norms, which measure the average, root-mean-square (RMS), and maximum absolute values of a function respectively. When the particular norm being used is not important, we will simply write $\|f\|$.

For the purposes of analysis, it is convenient to consider only the functions whose L_p norm is finite. The collection of all such functions (for a given value of p) is called an L_p *space*, which we will denote by $L_p(\mathcal{R})$ (to emphasize the domain $\mathcal{R}$ of these functions). These spaces have desirable analytic properties (which depend on the assumption of finite norms).

There are a variety of terms that are used to describe L_p spaces, corresponding to the various properties that they possess. At the most basic level, they are *vector spaces*, since each space $L_p(\mathcal{R})$ is closed under the operations of addition and scalar multiplication. Vectors spaces are also known as *linear spaces*, and in this context, as *function spaces* (since each element of $L_p(\mathcal{R})$ is a function).

The L_p spaces are also *complete*, meaning that all Cauchy sequences converge[3] (this property is useful for analysis). Thus, $L_p(\mathcal{R})$ is a complete, normed, linear space; in the terminology of functional analysis, this is called a *Banach space*.

[2] *Almost everywhere* means that the rays for which $f(\mathbf{r}) > m$ form a set of measure zero (with respect to the throughput measure μ). Thus according to this definition, the essential supremum ignores values of f that are attained only at isolated points, etc.

[3] A sequence of functions $f_1, f_2, \ldots$ is a *Cauchy sequence* if for any $\epsilon > 0$, there is an index N such that $\|f_i - f_j\| < \epsilon$ for all $i, j > N$. Such a sequence *converges* if there is a function $f \in L_p(\mathcal{R})$ such that $\lim_{N \to \infty} \|f_i - f\| = 0$.

Inner products. Another useful operation is the *inner product* of two functions on ray space, defined by

$$\langle f, g \rangle \;=\; \int_{\mathcal{R}} f(\mathbf{r})\, g(\mathbf{r})\, d\mu(\mathbf{r})\,. \tag{4.10}$$

Notice that the inner product notation is more concise than writing the integral explicitly, and yet it also imparts more information (since it can immediately be recognized as an inner product, rather than some other kind of integral). A linear space $\mathcal{F}$ equipped with an inner product is called an *inner product space*.

Every inner product has an *associated norm* defined by

$$\|f\| \;=\; \langle f, f \rangle^{1/2}\,,$$

which in this case is identical to the L_2 norm. Thus, the space $L_2(\mathcal{R})$, together with the inner product (4.10), is an example of an inner product space that is complete with respect to its associated norm: this is called a *Hilbert space*.

It is also possible to define weighted inner products between functions on ray space, by multiplying the integrand of (4.10) by a positive weighting function $w(\mathbf{r})$. This technique can also be used to define other norms. In this chapter, however, we will only have need for the unweighted versions defined above.

4.3 The scattering and propagation operators

From a physical standpoint, we can consider light transport to be an alternation of two steps. The first is *scattering*, which describes the interaction of photons with surfaces. The other is *propagation*, in which photons travel in straight lines through a fixed medium. Following Arvo et al. [1994], we will define each of these steps as a linear operator acting on radiance functions.

A *linear operator* is simply a linear function $\mathbf{A} : \mathcal{F} \to \mathcal{F}$ whose domain is a vector space $\mathcal{F}$. In our case, $\mathcal{F}$ is a space of radiance functions, as defined above. The notation $\mathbf{A}f$ denotes the application of an operator to a function, whose result is another function.

The local scattering operator. We begin with the *local scattering operator*, defined by[4]

$$(\mathbf{K}h)(\mathbf{x}, \omega_\mathrm{o}) \;=\; \int_{\mathcal{S}^2} f_\mathrm{s}(\mathbf{x}, \omega_\mathrm{i} \!\rightarrow\! \omega_\mathrm{o})\, h(\mathbf{x}, \omega_\mathrm{i})\, d\sigma^\perp_\mathbf{x}(\omega_\mathrm{i})\,. \tag{4.11}$$

When this operator is applied to an incident radiance function L_i, it returns the exitant radiance $L_\mathrm{o} = \mathbf{K} L_\mathrm{i}$ that results from a single scattering operation. Equation (4.11) is similar to the scattering equation (3.12), except that $\mathbf{K}$ operates on entire radiance functions, rather than being restricted to a single point $\mathbf{x}$. It maps one function L into another function $\mathbf{K}L$, where each function is defined over the whole ray space $\mathcal{R}$.

The propagation operator. To define the propagation operator, we first give a more precise definition of the *ray-casting function* $\mathbf{x}_\mathcal{M}(\mathbf{x}, \omega)$ mentioned in Section 3.7. First, let

$$d_\mathcal{M}(\mathbf{x}, \omega) \;=\; \inf\{d > 0 \mid \mathbf{x} + d\omega \in \mathcal{M}\}\,, \tag{4.12}$$

which is called the *boundary distance function* [Arvo 1995, p. 136]. We then define the ray-casting function as

$$\mathbf{x}_\mathcal{M}(\mathbf{x}, \omega) \;=\; \mathbf{x} + d_\mathcal{M}(\mathbf{x}, \omega)\, \omega\,, \tag{4.13}$$

so that $\mathbf{x}_\mathcal{M}(\mathbf{x}, \omega)$ represents the first point of $\mathcal{M}$ that is visible from $\mathbf{x}$ in the direction ω. When the ray $(\mathbf{x}, \omega)$ does not intersect $\mathcal{M}$, we have $d_\mathcal{M}(\mathbf{x}, \omega) = \infty$, and $\mathbf{x}_\mathcal{M}$ is not defined.[5]

The propagation of light in straight lines is now represented by the *geometric* or *propagation operator* $\mathbf{G}$, defined by

$$(\mathbf{G}h)(\mathbf{x}, \omega_\mathrm{i}) \;=\; \begin{cases} h(\mathbf{x}_\mathcal{M}(\mathbf{x}, \omega_\mathrm{i}), -\omega_\mathrm{i}) & \text{if } d_\mathcal{M}(\mathbf{x}, \omega_\mathrm{i}) < \infty\,, \\ 0 & \text{otherwise}\,, \end{cases} \tag{4.14}$$

This operator expresses the incident radiance L_i in terms of the exitant radiance L_o leaving the other surfaces of the environment, according to $L_\mathrm{i} = \mathbf{G} L_\mathrm{o}$.

These definitions of $\mathbf{G}$ and $\mathbf{K}$ are slightly different than those of Arvo [1995]. First, we

[4] Although this definition seems to depend on the particular ray representation $\mathbf{r} = (\mathbf{x}, \omega)$, in fact it can be used with any representation. To do this, simply replace the argument on the left-hand side by a single parameter $\mathbf{r}$, and replace the symbols $\mathbf{x}$ and ω_o by functions $\mathbf{x} = \mathbf{x}(\mathbf{r})$ and $\omega_\mathrm{o} = \omega_\mathrm{o}(\mathbf{r})$ (whose definitions depend on the representation used).

[5] With respect to equation (4.12), we have used the convention that $\inf \emptyset = \infty$, where $\emptyset$ is the empty set.

have considered transmission as well as reflection, by using the BSDF in the definition of $\mathbf{K}$. Second, we have used incident and exitant radiance rather than field and surface radiance (see Section 3.5), so that the direction of ω_{i} is reversed compared to [Arvo 1995]. The main advantage of this convention is that $\mathbf{G}$ and $\mathbf{K}$ are both self-adjoint when f_{s} is symmetric, which greatly increases the symmetry between light and importance transport (as we will see in Section 4.6). On the other hand, the $\mathbf{G}$ and $\mathbf{K}$ defined by Arvo are not self-adjoint. He handles this by introducing an isomorphism $\mathbf{H}$ between surface and field radiance functions, such that $\mathbf{HG}$ and $\mathbf{KH}$ are equivalent to the $\mathbf{G}$ and $\mathbf{K}$ defined here [Arvo 1995, p. 151].

Locality. Notice that to evaluate the radiance scattered along a given ray $(\mathbf{x}, \omega)$, we only need to know the incident radiance at the same point $\mathbf{x}$. In other words, the evaluation of $(\mathbf{K}h)(\mathbf{x}, \omega)$ only requires the evaluation of h on rays of the form $(\mathbf{x}, \omega')$. This property of the scattering operator $\mathbf{K}$ is called *locality*.

In general, we say that a transport operator $\mathbf{A}$ is *local* if the evaluation of $(\mathbf{A}h)(\mathbf{r})$ only requires the evaluation of h on a small set of rays $\mathbf{r}'$. In this sense, the propagation operator $\mathbf{G}$ is also local, since to evaluate $(\mathbf{G}h)(\mathbf{x}, \omega)$ we only need the value of h on a single ray $(\mathbf{x}', -\omega)$. In fact we could say that $\mathbf{G}$ is more local than $\mathbf{K}$, since $(\mathbf{G}h)(\mathbf{r})$ depends on the value of h on a single ray, while $(\mathbf{K}h)(\mathbf{r})$ depends on the value of h on a two-dimensional subset of $\mathcal{R}$.

Locality is important, since it dictates how much of the domain of h must be examined in order to compute $(\mathbf{A}h)(\mathbf{r})$ for a given ray $\mathbf{r}$. This type of locality has been successfully exploited in radiosity calculations, in order to handle textures more efficiently [Gershbein et al. 1994].

4.4 The light transport and solution operators

The composition of the scattering and propagation operators is called the *light transport operator*,

$$\mathbf{T} \;=\; \mathbf{K}\mathbf{G}\,.$$

This operator maps an exitant radiance function L_o into the exitant function $\mathbf{T}L_\mathrm{o}$ that results after a single scattering step. (When there is no ambiguity, we will drop the subscript on exitant functions and simply write L.)

Recall that our goal is to measure the equilibrium radiance L. The condition that must be satisfied in equilibrium is that

$$L \;=\; L_\mathrm{e} + \mathbf{T}L\,, \tag{4.15}$$

where $L_\mathrm{e}(\mathbf{r})$ is the emitted radiance function (specified as part of the scene model). This is called the *light transport equation*. It is simply a reformulation of (3.19), which says that at equilibrium, the exitant radiance must be the sum of emitted and scattered radiance.

The solution operator. Formally, the solution can be obtained by inverting the operator equation (4.15):

$$\begin{aligned} (\mathbf{I} - \mathbf{T})\, L &= L_\mathrm{e} \\ L &= (\mathbf{I} - \mathbf{T})^{-1}\, L_\mathrm{e}\,, \end{aligned}$$

where $\mathbf{I}$ is the identity operator. It is convenient to rewrite this equation in terms of the *solution operator*[6]

$$\mathbf{S} \;=\; (\mathbf{I} - \mathbf{T})^{-1}\,, \tag{4.16}$$

in which case the solution is simply $L \;=\; \mathbf{S}L_\mathrm{e}$.

Conditions for invertibility. These formal manipulations are valid only if the operator $\mathbf{I} - \mathbf{T}$ is invertible. A sufficient condition is that $\|\mathbf{T}\| \;<\; 1$, where $\|\mathbf{T}\|$ is the standard *operator norm*

$$\|\mathbf{T}\| \;=\; \sup_{\|f\| \le 1} \|\mathbf{T}f\|\,, \tag{4.17}$$

[6] Note that $\mathbf{S}$ is closely related to the *resolvent operator* $\mathbf{R}_\lambda$ used in spectral analysis, except that $\mathbf{R}_\lambda$ has a parameter λ, and does not have a universally accepted definition (e.g. compare [Delves & Mohamed 1985, p. 74], [Taylor & Lay 1980, p. 272]). It is also closely related to the "GRDF" of Lafortune & Willems [1994], which is simply a new name for the kernel of the solution operator $\mathbf{S}$.

where the norms on the right are function norms.[7] Given that $\|\mathbf{T}\| < 1$, the inverse of $\mathbf{I} - \mathbf{T}$ exists and is given by

$$\mathbf{S} \;=\; (\mathbf{I} - \mathbf{T})^{-1} \;=\; \sum_{i=0}^{\infty} \mathbf{T}^i \;=\; \mathbf{I} + \mathbf{T} + \mathbf{T}^2 + \cdots \,. \tag{4.18}$$

This is called the *Neumann series* (after C. Neumann, though the method goes back as far as Liouville [Taylor & Lay 1980, p. 191]). This expansion has a physical interpretation when applied to $L = \mathbf{S} L_{\mathrm{e}}$, since

$$L \;=\; L_{\mathrm{e}} + \mathbf{T} L_{\mathrm{e}} + \mathbf{T}^2 L_{\mathrm{e}} + \cdots$$

expresses L as the sum of emitted light, plus light that has been scattered once, twice, etc.

The validity of (4.18) raises the issue of whether $\|\mathbf{T}\| < 1$. In general, this depends on physical assumptions about the scene model, as well as the norm used for radiance functions. We will consider several cases.

For (one-sided) reflective surfaces, Arvo has shown that $\|\mathbf{G}\|_p \le 1$ for any $1 \le p \le \infty$ [Arvo 1995, Theorem 14]. Furthermore, he has shown that $\|\mathbf{K}\|_p \le 1$, as long as all BRDF's in the scene are energy-conserving and symmetric. By making the additional assumption that no surface is perfectly reflective, he obtains $\|\mathbf{K}\|_p < 1$ [Arvo 1995, Theorem 13], and thus

$$\|\mathbf{T}\|_p \;=\; \|\mathbf{K}\mathbf{G}\|_p \;\le\; \|\mathbf{K}\|_p \, \|\mathbf{G}\|_p \;<\; 1\,.$$

In the case of general scattering (i.e. transmission as well as reflection), things are slightly more complicated. Arvo's proof that $\|\mathbf{G}\| \;\le\; 1$ requires some modifications, because it depends on the fact that $\mathbf{G}^2 = \mathbf{I}$ when $\mathcal{M}$ forms an enclosure (which does not hold under the more general assumptions considered here). We will give a different proof below (Appendix 4.B). As for $\mathbf{K}$, it is no longer true that $\|\mathbf{K}\| < 1$. In fact, it is only true that

$$\|\mathbf{K}\| \;<\; \frac{\eta_{\max}^2}{\eta_{\min}^2}\,,$$

[7] Each function norm induces a distinct operator norm. The notation $\|\cdot\|_p$ can mean either the L_p norm on functions (4.8), or the corresponding operator norm, depending on the type of its argument.

where $\eta_{\min}$ and $\eta_{\max}$ denote the minimum and maximum refractive indices in the environment. This corresponds to the fact that radiance can increase during scattering, due to refraction. Putting these facts together, it is possible that $\|\mathbf{T}\|_p > 1$.

However, the condition $\|\mathbf{T}\| < 1$ is not strictly necessary, since the inverse of $\mathbf{I} - \mathbf{T}$ exists whenever the series given by (4.18) converges [Taylor & Lay 1980, p. 192]. A weaker yet sufficient condition for convergence is that $\|\mathbf{T}^k\| < 1$ for some $k \geq 1$.[8] In Section 4.B.3, we will show that this condition is satisfied for any physically valid scene model, and therefore the Neumann series converges (which makes $\mathbf{S}$ well-defined).

4.5 Sensors and measurements

The goal of light transport algorithms is to estimate a finite number of measurements of the equilibrium radiance L. For example, if the algorithm computes an image directly, then the measurements consist of many pixel values $I_1, \ldots, I_M$, where M is the number of pixels in the image. If the algorithm computes a finite-element solution, on the other hand, then the measurements I_j would simply be the basis function coefficients (with one measurement for each basis function).

Each measurement can be thought of as the response of a hypothetical sensor placed somewhere in the scene. For example, we can imagine that each pixel is a small piece of film within a virtual camera, and that the pixel value is proportional to the radiant power that it receives. Of course, most of the time the camera and lens system are not modeled explicitly. However, for any given pixel it is still possible to identify the set of rays in world space that contribute to its value, and assume that there is an imaginary sensor that responds to the radiance along these rays.

The sensor response can vary according to the position and direction of the incident radiance. We will only deal with linear sensors, in which case the response is characterized by a function

$$W_{\mathrm{e}}(\mathbf{x}, \omega) \;=\; \frac{dS(\mathbf{x}, \omega)}{d\Phi(\mathbf{x}, \omega)} \tag{4.19}$$

[8] Note that this condition implies that perfectly reflective mirrors are allowable, as long as it is not possible for light to continue bouncing indefinitely between these mirrors without some light escaping to another (more absorptive) portion of the scene.

that specifies the sensor response per unit of power arriving at $\mathbf{x}$ from direction ω. For real sensors, W_e is called the *flux responsivity* of the sensor [Nicodemus 1978, p. 59]. The corresponding units are $[S \cdot \mathrm{W}^{-1}]$, where S is the unit of sensor response. Depending on the sensor, S could represent a voltage, current, change in photographic film density, deflection of a meter needle, etc.

For the hypothetical sensors used in graphics, W_e is called an *exitant importance function* (we think of the sensor as emitting importance).[9] The corresponding sensor response is unitless, and thus importance has units of $[\mathrm{W}^{-1}]$. However, the symbol S is a convenient reminder that something is being measured. We assume that W_e is defined over the entire ray space $\mathcal{R}$, although it will be zero over most of this domain for typical sensors. In the case where measurements represent pixel values, note that W_e can model arbitrary lens systems used to form the image, as well as any linear filters used for anti-aliasing.

The measurement equation. To compute a measurement, we integrate the response

$$dS(\mathbf{r}) \;=\; W_\mathrm{e}(\mathbf{r})\, d\Phi(\mathbf{r}) \;=\; W_\mathrm{e}(\mathbf{r})\, L_\mathrm{i}(\mathbf{r})\, d\mu(\mathbf{r})$$

for all the incident radiance falling on the sensor. This is summarized by Nicodemus' *measurement equation* [Nicodemus 1978, p. 85], expressed in our notation as

$$I \;=\; \langle W_\mathrm{e}, L_\mathrm{i} \rangle \;=\; \int_{\mathcal{R}} W_\mathrm{e}(\mathbf{r})\, L_\mathrm{i}(\mathbf{r})\, d\mu(\mathbf{r})\,, \tag{4.20}$$

where I is a measurement, W_e is the emitted importance, and L_i is the incident radiance.[10]

Notice that we have defined both L_e and W_e as exitant quantities. This is natural, since it lets us define their values at points *on* the source or sensor. It would not be intuitive to define

[9] This follows the terminology of [Lewins 1965, p. 7, p. 21], where each importance function pertains to a single "meter reading" (measurement). The alternative term *potential function* [Pattanaik & Mudur 1995] is undesirable because it has a well-known, different meaning in physics (a function satisfying Poisson's equation, e.g. the electric or gravitational potential functions).

It is also allowable for an importance function to represent the average of a set of measurements (e.g. the average of all pixel values in an image). This is the case with importance-driven radiosity methods [Smits et al. 1992, p. 275], where the importance function is used only to guide the solution (and the value of the corresponding "measurement" is irrelevant).

[10] Strictly speaking, the measurement equation should also integrate over frequency (since L_i and W_e are spectral quantities, defined separately at each frequency ν). However, to simplify the notation we will usually ignore this detail.

$L_{\rm e}(\mathbf{x}, \omega)$ as the amount of light *arriving* at $\mathbf{x}$ from direction ω, since the actual emission takes place somewhere else. Similarly, it is more natural to define sensor response in terms of radiance arriving at the sensor, rather than radiance leaving points elsewhere in the scene (e.g. as with the "visual potential" proposed by Pattanaik & Mudur [1995]). The definitions of $L_{\rm e}$ and $W_{\rm e}$ as exitant quantities also increases the symmetry between light and importance transport [Christensen et al. 1993], as we discuss below.

Also notice that although we have defined the equilibrium solution $L = \mathbf{S} L_{\rm e}$ as an exitant quantity, the measurement equation (4.20) requires an incident function. This problem can be solved with the $\mathbf{G}$ operator, by using the relationship $L_{\rm i} = \mathbf{G} L$.[11] Each measurement now has the form

$$I \;=\; \langle W_{\rm e}, L_{\rm i} \rangle \;=\; \langle W_{\rm e}, \mathbf{G} L \rangle \;=\; \langle W_{\rm e}, \mathbf{G}\mathbf{S} L_{\rm e} \rangle \,. \tag{4.21}$$

Notice that it is the explicit inclusion of $\mathbf{G}$ in this equation that allows us to use the exitant forms of both $L_{\rm e}$ and $W_{\rm e}$.

4.6 Importance transport via adjoint operators

Adjoint operators are a powerful tool for understanding light transport algorithms. They allow us to evaluate measurements in a variety of ways, which can lead to new insights and rendering algorithms.

The *adjoint* of an operator $\mathbf{H}$ is denoted $\mathbf{H}^*$, and is defined by the property that[12]

$$\langle \mathbf{H}^* f, g \rangle \;=\; \langle f, \mathbf{H} g \rangle \quad \text{for all} \quad f, g\,. \tag{4.22}$$

An operator is *self-adjoint* if $\mathbf{H} = \mathbf{H}^*$. This corresponds to the familiar concept of a symmetric matrix in real linear algebra.

To show how the adjoint can be used, we apply the identity (4.22) to the measurement

[11] In order for $L_{\rm i} = \mathbf{G} L$ to represent the radiance arriving at the sensors, the sensors must be modeled as part of the domain $\mathcal{M}$. For the purposes of this framework, the sensors can be modeled without affecting light transport in the rest of the scene by making them completely transparent.

[12] The adjoint of an operator depends on the inner product used. In this chapter, we always use the inner product (4.10).

equation (4.21), yielding

$$I \;=\; \langle W_{\rm e}, \mathbf{G}\mathbf{S}L_{\rm e}\rangle \;=\; \langle (\mathbf{G}\mathbf{S})^* W_{\rm e}, L_{\rm e}\rangle\,. \tag{4.23}$$

This suggests that we can evaluate I by transporting importance in some way. To determine exactly what this means, we must express the operator $(\mathbf{G}\mathbf{S})^*$ in terms of the known operators $\mathbf{G}$ and $\mathbf{K}$.

We start by examining the adjoints of $\mathbf{G}$ and $\mathbf{K}$. It is relatively straightforward to show that $\mathbf{G} = \mathbf{G}^*$. On the other hand, the adjoint of $\mathbf{K}$ is given by

$$(\mathbf{K}^* h)(\mathbf{x}, \omega_{\rm o}) \;=\; \int_{\mathcal{S}^2} f_{\rm s}^*(\mathbf{x}, \omega_{\rm i}\!\rightarrow\!\omega_{\rm o})\, h(\mathbf{x}, \omega_{\rm i})\, d\sigma_{\mathbf{x}}^{\perp}(\omega_{\rm i}) \tag{4.24}$$

(see Appendix 4.B for proofs of these results). Notice that $\mathbf{K}^*$ is the same as $\mathbf{K}$, except that it uses the adjoint BSDF

$$f_{\rm s}^*(\mathbf{x}, \omega_{\rm i}\!\rightarrow\!\omega_{\rm o}) = f_{\rm s}(\mathbf{x}, \omega_{\rm o}\!\rightarrow\!\omega_{\rm i})\,.$$

For now, let us suppose that $f_{\rm s}$ is symmetric at every point $\mathbf{x} \in \mathcal{M}$, so that $\mathbf{K} = \mathbf{K}^*$. Putting these facts together with standard identities (Appendix 4.B), it is easy to show that

$$(\mathbf{G}\mathbf{S})^* \;=\; \mathbf{G}\mathbf{S}\,, \tag{4.25}$$

i.e. the operator $(\mathbf{G}\mathbf{S})$ is self-adjoint as well.

Thus according to (4.23), measurements can be evaluated using either

$$I \;=\; \langle W_{\rm e}, \mathbf{G}\mathbf{S}L_{\rm e}\rangle \qquad \text{or} \qquad I \;=\; \langle \mathbf{G}\mathbf{S}W_{\rm e}, L_{\rm e}\rangle\,. \tag{4.26}$$

The only difference between these two expressions is that $W_{\rm e}$ and $L_{\rm e}$ have been exchanged.

Importance transport. The significance of this symmetry is that any algorithms that apply to light transport may also be used for importance transport. There is an exact correspondence between the concepts, quantities, and equations in the two cases. In particular, the *equilibrium importance function* is given by $W \;=\; \mathbf{S}W_{\rm e}$, and satisfies the *importance transport equation*

$$W \;=\; W_{\rm e} + \mathbf{T}W\,.$$

Similarly, the relationship $L_{\mathrm{i}} = \mathbf{G} L_{\mathrm{o}}$ for incident radiance becomes $W_{\mathrm{i}} = \mathbf{G} W_{\mathrm{o}}$ for incident importance. Rewriting (4.26) using these definitions, we get the symmetric measurement equations

$$I = \langle W_{\mathrm{e}}, L_{\mathrm{i}} \rangle \qquad \text{or} \qquad I = \langle W_{\mathrm{i}}, L_{\mathrm{e}} \rangle \,.$$

However, if the scene model contains any surface with a non-symmetric BSDF, then $\mathbf{K} \neq \mathbf{K}^*$. This does not affect the light transport operator $\mathbf{T} = \mathbf{K}\mathbf{G}$, which we will rename $\mathbf{T}_L$, but the importance transport operator becomes

$$\mathbf{T}_W = \mathbf{K}^* \mathbf{G}$$

(see Appendix 4.B). This means that in general, light and importance obey different transport equations.

Furthermore, we have not yet considered the transport operators for the corresponding incident quantities, L_{i} and W_{i}. This leads to a multitude of possibilities for evaluating measurements, all with different transport equations. Fortunately, all of these equations share the same general structure, as described in the next section.

4.7 Summary of the operator framework

We consider the four basic transport quantities L_{o}, L_{i}, W_{o}, and W_{i}, corresponding to exitant radiance, incident radiance, exitant importance, and incident importance respectively. The propagation operator $\mathbf{G}$ maps exitant quantities to incident ones, according to the relationships

$$L_{\mathrm{i}} = \mathbf{G} L_{\mathrm{o}} \qquad \text{and} \qquad W_{\mathrm{i}} = \mathbf{G} W_{\mathrm{o}} \,. \tag{4.27}$$

Similarly, the local scattering operator $\mathbf{K}$ maps incident quantities to exitant ones:

$$L_{\mathrm{o}} = \mathbf{K} L_{\mathrm{i}} \qquad \text{and} \qquad W_{\mathrm{o}} = \mathbf{K}^* W_{\mathrm{i}} \,. \tag{4.28}$$

Recall that the operators $\mathbf{K}$ and $\mathbf{K}^*$ differ only in the ordering of the BSDF arguments ω_{i} and ω_{o} (see (4.24)).

By putting these relationships together in various ways, we obtain a different transport operator $\mathbf{T}_X$ for each quantity X, where X is one of L_i, L_o, W_i, or W_o. These operators are summarized in the table below:

	Exitant	Incident
Light	$\mathbf{T}_{L_\mathrm{o}} = \mathbf{KG}$	$\mathbf{T}_{L_\mathrm{i}} = \mathbf{GK}$
Importance	$\mathbf{T}_{W_\mathrm{o}} = \mathbf{K}^*\mathbf{G}$	$\mathbf{T}_{W_\mathrm{i}} = \mathbf{GK}^*$

To solve for the equilibrium value of any of these quantities, we use the transport equation

$$X \;=\; X_\mathrm{e} + \mathbf{T}_X X \,,$$

where X_e is the given emission function for X. The formal solution to this equation is

$$X \;=\; \mathbf{S}_X X_\mathrm{e} \,,$$

where $\mathbf{S}_X \;=\; (\mathbf{I} - \mathbf{T}_X)^{-1}$ is called the *solution operator* for X. Finally, a measurement I can be computed using any of the following expressions:

$$\begin{aligned} I \;&=\; \langle W_\mathrm{e}, L_\mathrm{i} \rangle \;&=\; \langle W_\mathrm{i}, L_\mathrm{e} \rangle \\ &=\; \langle W_\mathrm{e,i}, L_\mathrm{o} \rangle \;&=\; \langle W_\mathrm{o}, L_\mathrm{e,i} \rangle \,. \end{aligned} \tag{4.29}$$

To apply these equations, recall that we are initially given two emission functions, one that describes the emitted radiance, and one that describes the emitted importance (i.e. the sensor responsivity). Most often, both are given as exitant functions (L_e or W_e), but for some problems, the incident form is more natural ($L_\mathrm{e,i}$ or $W_\mathrm{e,i}$). For example, suppose that we wish to project the equilibrium radiance L_o onto a set of orthonormal basis functions B_j (e.g. as with finite element approaches). In this situation, the coefficient of each basis function B_j is given by the inner product $\langle B_j, L_\mathrm{o} \rangle$. Comparing this "measurement" against the templates above (4.29), we see that B_j is considered to be an incident importance function ($W_\mathrm{e,i}$). This is because B_j specifies the response to radiance *leaving* the corresponding surface, rather than radiance arriving at it.

If the emitted radiance and importance are supplied in opposite forms (one incident and one exitant), the equations above can be applied in a straightforward manner by solving for the equilibrium function of one quantity (e.g. L_{o}), and computing an inner product with the emitted function of the other (e.g. $W_{\mathrm{e,i}}$). On the other hand, if two exitant functions are supplied (L_{e} and W_{e}), one of them must be converted to an incident quantity using the relationships $L_{\mathrm{i}} = \mathbf{G}L_{\mathrm{o}}$ or $W_{\mathrm{i}} = \mathbf{G}W_{\mathrm{o}}$(4.27), before a measurement can be computed (e.g. one possibility is $\langle \mathbf{G}W_{\mathrm{e}}, L_{\mathrm{o}} \rangle$). Similarly, if two incident emission functions are provided ($L_{\mathrm{e,i}}$ and $W_{\mathrm{e,i}}$), one of them must be converted to an exitant function using the relationships $L_{\mathrm{o}} = \mathbf{K}L_{\mathrm{i}}$ or $W_{\mathrm{o}} = \mathbf{K}W_{\mathrm{i}}$ (e.g. a measurement of the form $\langle W_{\mathrm{e,i}}, \mathbf{S}_{L_{\mathrm{o}}}\mathbf{K}L_{\mathrm{e,i}} \rangle$).

Together, these equations specify many ways in which measurements can be made. Everything is constructed from only two basic operators, $\mathbf{G}$ and $\mathbf{K}$, which represent independent components of the light transport process. It is clear which relationships are fundamental, and which depend on the assumption of symmetric BSDF's (i.e. $\mathbf{K} = \mathbf{K}^*$). The notation has been chosen to simplify the structure as much as possible, by using the concepts of ray space, measures, inner products, and linear operators.

Previous authors have stated special cases of these results. For example, Christensen et al. [1993] show that L_{o} and W_{o} satisfy the same transport equation, provided that all BSDF's are symmetric. Similarly, Pattanaik & Mudur [1995] show that L_{o} and W_{i} satisfy adjoint transport equations (i.e. $\mathbf{T}_{W_{\mathrm{o}}} = \mathbf{T}^*_{L_{\mathrm{o}}}$), although their arguments are not rigorous. Arvo [1995] derives the adjoints of $\mathbf{G}$ and $\mathbf{K}$ for one-sided, reflective surfaces, but does not discuss their significance. Adjoint relationships have also been used extensively in the field of neutron transport [Spanier & Gelbard 1969, Lewins 1965], but those results apply to volume scattering rather than surfaces.

Appendix 4.A A new characterization of particle tracing

Intuitively, particle tracing consists of following the paths of "photons" as they are emitted, scattered, and eventually absorbed. These *particle histories* are then used to approximate the equilibrium radiance function, either as a discrete set of measurements (e.g. pixel values, basis function coefficients), or in some other way (e.g. density estimation [Shirley et al. 1995]). This simple idea can be made quite sophisticated by applying different estimators to the particle histories, or by sampling the particles in clever ways [Spanier & Gelbard 1969].

Several explanations of particle tracing have been proposed in computer graphics. Most often these methods are justified intuitively, by appealing to the notion that each particle carries a certain amount of "energy" (e.g. [Shirley et al. 1995]). Pattanaik & Mudur [1995] propose a different approach, by interpreting particle tracing as a random walk solution of the importance transport equation. Our goal is to relate particle-based methods to the transport framework of this chapter, and study the conditions that must be satisfied to ensure that particle tracing algorithms are correct.

Our approach. We present a new characterization of particle tracing that addresses these issues. We define a particle tracing algorithm as a method for generating a set of N weighted sample rays

$$(\alpha_i, \mathbf{r}_i) \, ,$$

where each $\mathbf{r}_i$ is a ray, and α_i is its corresponding weight. These samples must be an unbiased representation of the equilibrium radiance L, such that the estimate

$$E\left[\frac{1}{N} \sum_{i=1}^{N} \alpha_i W_{\mathrm{e}}(\mathbf{r}_i) \right] \;=\; \langle W_{\mathrm{e}}, L \rangle \qquad (4.30)$$

holds for any importance function W_{e}. Essentially, this identity states that an arbitrary linear measurement can be estimated by taking a weighted sum over the given set of random sample rays.

Formally, this is a condition on the joint density function $p(\alpha, \mathbf{r})$ of the weighted sample rays:

$$\int_{\mathbb{R}} \alpha \, p(\alpha, \mathbf{r}) \, d\alpha \;=\; L(\mathbf{r}) \, , \qquad (4.31)$$

since this ensures that

$$\begin{aligned} E\left[\frac{1}{N} \sum_{i=1}^{N} \alpha_i \, W_{\mathrm{e}}(\mathbf{r}_i) \right] \;&=\; \int_{\mathcal{R}} \int_{\mathbb{R}} \alpha \, W_{\mathrm{e}}(\mathbf{r}) \, p(\alpha, \mathbf{r}) \, d\alpha \, d\mu(\mathbf{r}) \\ &=\; \int_{\mathcal{R}} W_{\mathrm{e}}(\mathbf{r}) \, L(\mathbf{r}) \, d\mu(\mathbf{r}) \;=\; \langle W_{\mathrm{e}}, L \rangle \, . \end{aligned}$$

Using these ideas, rendering algorithms based on particle tracing can be decomposed into two independent steps. First, there must be a method for generating a set of sample rays that satisfy the conditions above. The simplest way of doing this involves tracing n independent particle histories starting from the light sources, as described below. Second, the algorithm must use these samples in some way to compute the desired set of measurements. In addition to unbiased estimators of the form (4.30), other (biased) possibilities include density estimation [Shirley et al. 1995] and various forms of interpolation (e.g. see [Jensen 1996]).

The conditions (4.30) and (4.31) are a specification of the *interface* between these two components of a rendering algorithm. On the one hand, there are a variety of ways to prove them for specific particle generation schemes. On the other hand, they concisely state the properties that the generated particles possess, i.e. the properties that higher-level rendering algorithms are allowed to depend on. Essentially, these conditions are a rigorous interpretation of the "energy packets" approach: they preserve the idea that particles represent the equilibrium radiance itself, independent of any particular sensor.

Generating the particles. The simplest way to generate a set of weighted ray samples satisfying condition (4.30) is to follow a random walk. This process can be summarized as follows:

1. Choose a random ray $\mathbf{r}_0 = (\mathbf{x}_0, \omega_0)$ starting on a light source, and let its weight be
$$\alpha_0 \;=\; \frac{L_{\mathrm{e}}(\mathbf{r}_0)}{p_0(\mathbf{r}_0)}$$
where $p_0(\mathbf{r})$ is the density from which $\mathbf{r}_0$ was sampled. The initial state of the particle is defined to be $(\alpha_0, \mathbf{r}_0)$.

2. Given the current state $(\alpha_i, \mathbf{r}_i)$, decide whether to continue the random walk. We let q_{i+1} denote the probability with which the random walk is continued (where q_{i+1} depends on the current path in some way). If the walk is terminated (which happens with probability $1 - q_{i+1}$), we let $k = i$ denote its length.

3. Otherwise, let $\mathbf{x}_{i+1}$ be the first intersection point of the ray $\mathbf{r}_i = (\mathbf{x}_i, \omega_i)$ with a surface (see Figure 4.1). Choose a random scattering direction ω_{i+1} according to some density function p_{i+1} that approximates the BSDF there. The particle weight α_{i+1} is then computed from α_i using the formula
$$\alpha_{i+1} \;=\; \alpha_i \, \frac{1}{q_{i+1}} \, \frac{f_{\mathrm{s}}^*(\mathbf{x}_{i+1}, \omega_{i+1} \to -\omega_i)}{p_{i+1}(\omega_{i+1})}\,, \tag{4.32}$$

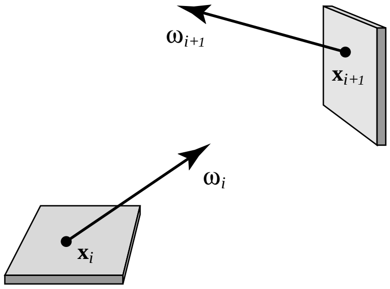


Figure 4.1: A scattering step in particle tracing.

where the density $p_{i+1}(\omega)$ is measured with respect to projected solid angle. (If this density is measured with respect to ordinary solid angle instead, the corresponding update formula is

$$\alpha_{i+1} \;=\; \alpha_i \, \frac{1}{q_{i+1}} \, \frac{f_{\rm s}^*(\mathbf{x}_{i+1}, \omega_{i+1} \to -\omega_i) \, |\omega_{i+1} \cdot \mathbf{N}(\mathbf{x}_{i+1})|}{p_{i+1}(\omega_{i+1})} \,, \tag{4.33}$$

where $\mathbf{N}(\mathbf{x})$ is the normal at $\mathbf{x}$.) Notice that we have used the adjoint BSDF in this expression, according to the sampling conventions of Section 3.7.6.

4. Return to step 2.

This process yields a set of ray samples $\mathbf{r}_0, \ldots \mathbf{r}_k$, where k is the length of the random walk. Each ray $\mathbf{r}_i = (\mathbf{x}_i, \omega_i)$ is assigned the weight

$$\alpha_i \;=\; \frac{L_{\rm e}(\mathbf{x}_0, \omega_0)}{p_0(\mathbf{x}_0, \omega_0)} \prod_{j=0}^{i-1} \frac{1}{q_{j+1}} \, \frac{f_{\rm s}^*(\mathbf{x}_{j+1}, \omega_{j+1} \to -\omega_j)}{p_{j+1}(\omega_{j+1})} \,, \tag{4.34}$$

which was obtained by expanding the recursion (4.32). There are a variety of ways to show that these samples satisfy the desired condition (4.30), either directly from the light transport equation (similar to [Spanier & Gelbard 1969, p. 62]), or from the importance transport equation (as we discuss below).

There are many other possibilities for generating a suitable set of particles, by modifying and extending the basic particle tracing technique. For example, different density functions could be used for sampling, Russian roulette or splitting could be applied, the samples could be resampled to make the weights all equal, and so on. Our new characterization (4.30) applies to all of these possibilities: this makes it clear that the essence of particle tracing is not how the samples are obtained, but what they represent.

Particle tracing as importance transport. For comparison, we summarize the approach of Pattanaik & Mudur [1995] (in our notation, and in a more general form). Given a set of basis functions

$$B_1, \ldots, B_M ,$$

they consider the problem of approximating the equilibrium radiance as a linear combination

$$L_{\mathrm{o}} \approx \sum_{i=1}^{M} I_j \, B_j ,$$

where $I_j = \langle B_j, L_{\mathrm{o}} \rangle$ is the coefficient of the j-th basis function. Notice that since the inner product $\langle B_j, L_{\mathrm{o}} \rangle$ measures the radiance *leaving* the corresponding surface, we can interpret B_j as an incident importance function $W_{\mathrm{e,i}}^{(j)}$, and rewrite the expression for each coefficient I_j as

$$I_j = \langle W_{\mathrm{e,i}}^{(j)}, L_{\mathrm{o}} \rangle . \tag{4.35}$$

To evaluate this expression, they first rewrite it as an importance transport problem:

$$I_j = \langle W_{\mathrm{i}}^{(j)}, L_{\mathrm{e}} \rangle . \tag{4.36}$$

This equation is then evaluated by recursively sampling the importance transport equation. This leads to a random walk that is very similar to the one that we have already described above.

They start by letting $\mathbf{r}_0 = (\mathbf{x}_0, \omega_0)$ be a random ray sampled on a light source, from which I_j can be estimated using

$$I_j = E\left[\frac{W_{\mathrm{i}}^{(j)}(\mathbf{r}_0) \, L_{\mathrm{e}}(\mathbf{r}_0)}{p_0(\mathbf{r}_0)} \right] \tag{4.37}$$

where $p_0(\mathbf{r}_0)$ is the density for sampling $\mathbf{r}_0$. Notice that this expression is just the usual Monte Carlo estimate $f(x)/p(x)$, applied to equation (4.36).

The factors L_{e} and p_0 in (4.37) can easily be evaluated, leaving only the equilibrium importance $W_{\mathrm{i}}^{(j)}(\mathbf{r}_0)$. This factor is evaluated recursively, using the transport equation

$$W_{\mathrm{i}}^{(j)} = W_{\mathrm{e,i}}^{(j)} + \mathbf{G}\mathbf{K}^* W_{\mathrm{i}}^{(j)} . \tag{4.38}$$

Starting with $i = 0$, this is done by casting the ray $\mathbf{r}_i = (\mathbf{x}_i, \omega_i)$ to find the first intersection point $\mathbf{x}_{i+1}$ with a surface. Next, a new ray direction ω_{i+1} is chosen, according to a density function p_{i+1} that approximates the BSDF at $\mathbf{x}_{i+1}$ (recall Figure 4.1). It is then easy to check that $W_{\mathrm{i}}^{(j)}(\mathbf{r}_i)$ can be estimated as

$$\begin{aligned} W_{\mathrm{i}}^{(j)}(\mathbf{r}_i) &= W_{\mathrm{e,i}}^{(j)}(\mathbf{r}_i) + (\mathbf{G}\mathbf{K}^* W_{\mathrm{i}}^{(j)})(\mathbf{r}_i) \\ &= W_{\mathrm{e,i}}^{(j)}(\mathbf{r}_i) + \begin{cases} 0 & \text{if } d(\mathbf{r}_i) = \infty \text{ or } \omega_{i+1} = \Lambda\,, \\ E\left[\dfrac{f_{\mathrm{s}}^*(\mathbf{x}_{i+1}, \omega_{i+1} \to -\omega_i)\, W_{\mathrm{i}}^{(j)}(\mathbf{r}_{i+1})}{q_{i+1}\, p_{i+1}(\omega_{i+1})}\right] \\ & \text{otherwise}\,, \end{cases} \end{aligned} \tag{4.39}$$

where p_{i+1} is measured with respect to projected solid angle. The recursion stops when a ray fails to intersect a surface ($d(\mathbf{r}_i) = \infty$), or when the walk is randomly terminated (as indicated by setting $\mathbf{x}_{i+1} = \Lambda$), which happens with probability $1 - q_{i+1}$.

Discussion. This process is superficially quite similar to particle tracing: it generates a random walk that starts at a light source, is scattered at each surface, and is eventually absorbed. Furthermore, it is theoretically well-founded, since it computes an unbiased estimate of the measurement I.

However, this view of particle tracing also has several disadvantages. First, there is no obvious relationship to the concepts of photons or particle energies. Furthermore, the sampling process seems to depend on which particular measurement I_j we evaluate (since equation (4.38) describes the equilibrium importance for a specific sensor). This would seem to imply that different particle histories are needed to estimate each measurement, which would be very inefficient. In contrast, in practice the random walks are chosen *not* to depend on any particular I_j, since this allows each random walk to be used for all measurements simultaneously. The formal dependence on a particular measurement I_j is rather non-intuitive.

Relationship to particle weights. Finally, observe that the importance transport process generates a set of ray samples $\mathbf{r}_0, \ldots, \mathbf{r}_k$. Furthermore, if we expand the recursion (4.39) and multiply together the factors that weight the emitted importance $W_{\mathrm{e,i}}(\mathbf{r}_i)$, we obtain exactly the same weights α_i that were used for particle tracing (see equation (4.34)). Thus, if we insert this set of weighted ray samples into equation (4.30), we obtain the same estimate for a given measurement I that would have been obtained by recursively sampling the importance transport equation (we have simply rearranged the calculations). Since we have already shown that the importance transport scheme gives an unbiased estimate (for any importance function $W_{\mathrm{e,i}}$), it follows that this set of weighted ray samples satisfies the desired condition (4.30). This validates the expression for the particle weights given earlier.

Appendix 4.B Properties of the operators $\mathbf{G}$, $\mathbf{K}$, $\mathbf{T}$, and $\mathbf{S}$

This appendix gives some formal results on the invertibility, adjoints, and norms of the operators $\mathbf{G}$, $\mathbf{K}$, $\mathbf{T}_X$, and $\mathbf{S}_X$. In some cases, these results are parallel to those of Arvo [1995], who considered the case of purely reflective, one-sided surfaces. However, different proof techniques are required for the general case considered here, and in most cases the specific results are different as well.

The proofs are not difficult; however, it is surprisingly tricky to state the results correctly. There are often exceptions to "intuitively obvious" properties that require careful thought. We have also emphasized some of the more subtle issues that arise in the proofs.

We will need various properties of the scene $\mathcal{M}$, so it is important to clarify exactly what is allowed. Recall that in Section 3.1, we defined $\mathcal{M}$ to be the union of a finite number of closed, piecewise differentiable, two-dimensional manifolds (possibly with boundary). In particular, we allow manifolds to be unbounded (e.g. a plane), to have any number of handles (e.g. a torus), and to have any number of holes (e.g. an annulus). A manifold can even be disconnected; for example, a single manifold may represent several spheres (this could arise naturally as an implicit surface). On the other hand, true fractals are not allowed (since they are not manifolds), and $\mathcal{M}$ must always be representable as a *finite* union of manifolds.[13] Note that despite this restriction, $\mathcal{M}$ may still contain an infinite number of connected surfaces, since one manifold could represent an infinite stack of planes, or an infinite grid of spheres.

4.B.1 Invertibility

Invertibility is an important property for computer vision problems. For example, suppose that we wish to determine the incident radiance L_{i} at a surface point, given the exitant radiance L_{o}. This situation is described by the inverse of the operator that applies to the corresponding graphics problem.

In this section, we show that $\mathbf{G}$ and $\mathbf{K}$ are not invertible in general. We also show that it is possible to construct special scene models where their inverses do exist.[14]

[13] If countable unions were allowed, then $\mathcal{M}$ could be a dense subset of $\mathbb{R}^3$. For example, let $q_1, q_2, \ldots$ be an enumeration of the rational numbers, and define $\mathcal{M}$ as the union of the planes $x = q_i$.

[14] In the case of one-sided, reflective surfaces that form an enclosure, $\mathbf{G}$ is invertible [Arvo 1995]. In fact, $\mathbf{G}$ is its own inverse ($\mathbf{G}^2 = \mathbf{I}$). However, for this to be true, the space of radiance functions f must be defined carefully. In particular, the domain of these functions is not $\mathcal{M} \times \mathcal{S}^2$; it is the set of rays $(\mathbf{x}, \omega)$ where $\mathbf{x} \in \mathcal{M}$ and $\omega \in \mathcal{H}^2_+(\mathbf{x})$ (the upward hemisphere at $\mathbf{x}$).

Recall the definition of the *boundary distance function*

$$d_{\mathcal{M}}(\mathbf{x}, \omega) \;=\; \inf\{d > 0 \mid \mathbf{x} + d\omega \in \mathcal{M}\}\,,$$

and the *ray casting function*

$$\mathbf{x}_{\mathcal{M}}(\mathbf{x}, \omega) \;=\; \mathbf{x} + d_{\mathcal{M}}(\mathbf{x}, \omega)\,\omega\,.$$

For brevity, we will omit the $\mathcal{M}$ subscripts on these functions in this appendix.

We begin with the following definitions:

1. A ray $\mathbf{r}$ is *reversible* if $d(\mathbf{r}) < \infty$.

2. The *reversible ray space* $\tilde{\mathcal{R}}$ is the set of all reversible rays in $\mathcal{R}$.

3. The *reversal map* is a function $M : \tilde{\mathcal{R}} \to \mathcal{R}$ whose value is given by

$$M(\mathbf{x}, \omega) \;=\; (\mathbf{x}(\mathbf{x}, \omega), -\omega)\,.$$

The following lemma implies that the range of M is actually the reversible ray space $\tilde{\mathcal{R}}$ (rather than all of $\mathcal{R}$, as we defined it above), and that furthermore $M : \tilde{\mathcal{R}} \to \tilde{\mathcal{R}}$ is a bijection.

Lemma 4.1. *If $\mathbf{r}$ is reversible, then $M(\mathbf{r})$ is also reversible, and $M(M(\mathbf{r})) = \mathbf{r}$.*

Proof. Let $\mathbf{r} = (\mathbf{x}, \omega)$ and $\mathbf{r}' = M(\mathbf{r})$. Also let $d = d(\mathbf{x}, \omega)$ and $\mathbf{x}' = \mathbf{x} + d\omega$, so that $\mathbf{r}' = (\mathbf{x}', -\omega)$. Now by definition,

$$d(\mathbf{r}') \;=\; \inf\{d > 0 \mid \mathbf{x}' - d\omega \in \mathcal{M}\}\,.$$

Since $\mathbf{x}' - d\omega = \mathbf{x} \in \mathcal{M}$, we have $d(\mathbf{r}') \le d$, and so $\mathbf{r}'$ is reversible.

Now assume that $d(\mathbf{r}') = d'$, where $0 < d' < d$. Then $\mathbf{x}' - d'\omega = \mathbf{x} + (d - d')\omega$ is a point of $\mathcal{M}$, which contradicts the fact that $d(\mathbf{x}, \omega) = d$. ∎

With respect to these definitions, the propagation operator (4.14) is defined by

$$(\mathbf{G}h)(\mathbf{r}) \;=\; \begin{cases} h(M(\mathbf{r})) & \text{for } \mathbf{r} \in \tilde{\mathcal{R}}\,, \\ 0 & \text{otherwise}\,. \end{cases}$$

Theorem 4.2. *Whenever $\mathcal{M}$ is non-empty and bounded, then $\mathbf{G}$ is not invertible. However, $\mathbf{G}$ is invertible for some unbounded scenes.*

Proof. Assuming that $\mathcal{M}$ is non-empty and bounded, we first show that some rays are not reversible, i.e. $\mu(\mathcal{R} - \tilde{\mathcal{R}}) > 0$. (These are the "outward-pointing" rays that do not intersect $\mathcal{M}$.) This may seem obvious, but since there are scenes for which all rays are reversible (see below), we must be a bit careful.

Consider the non-reversible rays $(\mathbf{x}, \omega)$ for a fixed direction $\omega \in \mathcal{S}^2$. The origins of these rays are the maximal points of $\mathcal{M}$ along lines in the direction ω. Thus, there is an exact correspondence between non-reversible rays, and directed lines that intersect $\mathcal{M}$. Given that ω occupies an infinitesimal solid angle $d\sigma(\omega)$, the measure of these rays is

$$A_\omega(\Pi_\omega(\mathcal{M}))\, d\sigma(\omega)\,,$$

where Π_ω denotes orthogonal projection onto the plane perpendicular to ω, and A_ω is the area measure on that plane. Thus, the total measure of the non-reversible rays is exactly

$$\mu(\mathcal{R} - \tilde{\mathcal{R}}) \;=\; \int_{\mathcal{S}^2} A_\omega(\Pi_\omega(\mathcal{M}))\, d\sigma(\omega)\,.$$

This is simply 4π times the average projected surface area of $\mathcal{M}$, which is positive.[15]

Now let L_1, L_2 be any two radiance functions such that

$$L_1(\mathbf{r}) \;=\; L_2(\mathbf{r}) \qquad \text{for } \mathbf{r} \in \tilde{\mathcal{R}}\,.$$

Then clearly $\mathbf{G}L_1 = \mathbf{G}L_2$, no matter what values the L_i have for $r \in \mathcal{R} - \tilde{\mathcal{R}}$. Since $\mu(\mathcal{R} - \tilde{\mathcal{R}}) > 0$, the functions L_1 and L_2 can be distinct, which shows that $\mathbf{G}$ is not invertible in general.

For an example where $\mathbf{G}$ is invertible, let $\mathcal{M}$ be an infinite stack of planes, or an infinite set of concentric spheres. In this case, all rays are reversible (up to a set of μ-measure zero). This can also be achieved with a single infinite surface that is diffeomorphic to a plane. In general, $\mathbf{G}$ is invertible if and only if $\mathcal{M}$ has a non-empty intersection with the interior of every infinite cone in $\mathbb{R}^3$ (or if $\mathcal{M}$ is empty). ∎

Note especially that $\mathbf{G}$ is not its own inverse, so that the relationship $L_\mathrm{i} = \mathbf{G}L_\mathrm{o}$ does not imply $L_\mathrm{o} = \mathbf{G}L_\mathrm{i}$.

[15] Technically, this is also a bit tricky. It is possible make the average projected surface area of $\mathcal{M}$ arbitrarily small, while keeping the same total area, by making $\mathcal{M}$ very convoluted. However, we can use the assumption that $\mathcal{M}$ is a finite union of piecewise differentiable manifolds. This implies that for almost every point $\mathbf{x} \in \mathcal{M}$, we can find a neighborhood $U_\mathbf{x}$ that is arbitrarily close to being a disc (of very small radius). The average projected surface area of any such $U_\mathbf{x}$ is positive (because it is almost a disc), and this is a lower bound on the average projected surface area of $\mathcal{M}$.

Theorem 4.3. *The operator* $\mathbf{K}$ *is not invertible in general. However,* $\mathbf{K}$ *is invertible for some special scene models.*

Proof. Suppose that every point of $\mathcal{M}$ has a constant BSDF (with regard to both reflection and transmission):

$$f_{\mathrm{s}}(\mathbf{x}, \omega_{\mathrm{i}} \to \omega_{\mathrm{o}}) \;=\; g(\mathbf{x}) \qquad \text{for all } \mathbf{x} \in \mathcal{M} \text{ and } \omega_{\mathrm{i}}, \omega_{\mathrm{o}} \in \mathcal{S}^2 \,.$$

In this case, $\mathbf{K}$ reduces to

$$(\mathbf{K}L)(\mathbf{x}, \omega_{\mathrm{o}}) \;=\; \int_{\mathcal{S}^2} g(\mathbf{x})\, L(\mathbf{x}, \omega_{\mathrm{i}})\, d\sigma_{\mathbf{x}}^{\perp}(\omega_{\mathrm{i}}) \,.$$

Since the right-hand side does not depend on ω_{o}, $\mathbf{K}L$ is a function of position only. That is, for every L there is a function $h_0 : \mathcal{M} \to \mathbb{R}$ such that $(\mathbf{K}L)(\mathbf{x}, \omega) = h_0(\mathbf{x})$. However, it is clear that infinitely many distinct functions L map to each such h_0, and thus $\mathbf{K}$ is not invertible.

It is easy to see that this argument still holds if any part of the scene has a diffuse BRDF or BTDF, thus $\mathbf{K}$ is not invertible for most graphics models.

On the other hand, suppose that every point of $\mathcal{M}$ is a mirror. In this case, it is easy to see that $\mathbf{K}$ is an isomorphism. There are also less trivial examples. For example, f_{s} could be chosen so that $\mathbf{K}$ encodes the two-dimensional Fourier transform of the input signal. ∎

With regard to the other operators we have defined, it is easy to see that the transport operators $\mathbf{T}_X$ are not invertible in general, since they are compositions of $\mathbf{K}$ and $\mathbf{G}$. For the operators $\mathbf{I} - \mathbf{T}_X$, on the other hand, invertibility depends on the norms of $\mathbf{K}$ and $\mathbf{G}$ (to be discussed in Section 4.B.3). These operators must be invertible in order for the solution operators $\mathbf{S}_X = (\mathbf{I} - \mathbf{T}_X)^{-1}$ to exist.

4.B.2 Adjoints

We derive the adjoints of $\mathbf{G}$ and $\mathbf{K}$, and use them to prove the relationship

$$\langle W_{\mathrm{e}}, \mathbf{G}\mathbf{S}_{L_{\mathrm{o}}} L_{\mathrm{e}} \rangle \;=\; \langle \mathbf{G}\mathbf{S}_{W_{\mathrm{o}}} W_{\mathrm{e}}, L_{\mathrm{e}} \rangle \,.$$

(The operators $\mathbf{T}_X$ and $\mathbf{S}_X$ are defined in Section 4.7.) Our approach is unique in that we use this identity to *define* the equilibrium importance, according to

$$W_{\mathrm{o}} \;=\; \mathbf{S}_{W_{\mathrm{o}}} W_{\mathrm{e}} \,.$$

From this we derive the transport equation satisfied by $W_{\rm o}$, and we derive similar results for the incident quantities $L_{\rm i}$ and $W_{\rm i}$.

We begin by showing that $\mathbf{G}$ is self-adjoint.[16] The following lemma states that M preserves the measure μ.

Lemma 4.4. *Let μ be the throughput measure (4.2), and let $D \subset \tilde{\mathcal{R}}$ be a measurable set. Then*

$$\mu(M(D)) \;=\; \mu(D)\,,$$

where $M(D) = \{M(\mathbf{x}) \mid \mathbf{x} \in D\}$.

Proof. Since M is a bijection, it is sufficient to show that μ is preserved locally. Let $dA(\mathbf{x}) \times d\sigma(\omega)$ be an infinitesimal neighborhood of the ray $(\mathbf{x}, \omega) \in D$, and define

$$(\mathbf{x}', \omega') \;=\; M(\mathbf{x}, \omega) \;=\; (\mathbf{x}(\mathbf{x}, \omega), -\omega)\,.$$

We must show that $d\mu(\mathbf{x}, \omega) = d\mu(\mathbf{x}', \omega')$. Recall that one expression for μ is given by

$$d\mu(\mathbf{x}, \omega) \;=\; dA^{\perp}_{\omega}(\mathbf{x})\, d\sigma(\omega)\,.$$

We immediately have $d\sigma(\omega') = d\sigma(-\omega) = d\sigma(\omega)$. As for the other factor, recall that $A^{\perp}_{\omega}$ measures projected surface area on a plane perpendicular to ω. By definition of $\mathbf{x}(\mathbf{x}, \omega)$, however, $\mathbf{x}' - \mathbf{x}$ is always parallel to ω. Thus two corresponding areas $dA(\mathbf{x}), dA(\mathbf{x}')$ on $\mathcal{M}$ will always have the same projected area,

$$dA^{\perp}_{\omega}(\mathbf{x}') \;=\; dA^{\perp}_{\omega}(\mathbf{x}(\mathbf{x}, \omega)) \;=\; dA^{\perp}_{\omega}(\mathbf{x})\,. \quad \blacksquare$$

In terms of the composition measure notation (5.29), the preceding lemma states that $\mu \circ M = \mu$ (restricted to reversible rays).

Theorem 4.5. *The operator $\mathbf{G}$ is self-adjoint (for any scene model).*

Proof. Given $f, g \in L_2$, we have

$$\begin{aligned} \langle f, \mathbf{G}g \rangle &= \int_{\tilde{\mathcal{R}}} f(\mathbf{r})\, g(M(\mathbf{r}))\, d\mu(\mathbf{r}) \\ &= \int_{\tilde{\mathcal{R}}} f(\mathbf{r})\, g(M(\mathbf{r}))\, \frac{d\mu(\mathbf{r})}{d(\mu \circ M)}\, d\mu(M(\mathbf{r})) \end{aligned}$$

[16] Our proof is necessarily different than the one in [Arvo 1995], which assumed that $\mathbf{G}^2 = \mathbf{I}$ when $\mathcal{M}$ forms an enclosure.

$$\begin{aligned} &= \int_{\tilde{\mathcal{R}}} f(M(\mathbf{r}'))\, g(\mathbf{r}')\, d\mu(\mathbf{r}') \\ &= \langle \mathbf{G} f, g \rangle\,, \end{aligned}$$

where we have used the fact that M is a bijection, that $\mu \circ M = \mu$, and (5.32). ∎

The following theorem and proof are similar to [Arvo 1995, Theorem 16].

Theorem 4.6. *The adjoint of* $\mathbf{K}$ *is given by*

$$(\mathbf{K}^* h)(\mathbf{x}, \omega_{\mathrm{o}}) \;=\; \int_{\mathcal{S}^2} f_{\mathrm{s}}^*(\mathbf{x}, \omega_{\mathrm{i}} \!\to\! \omega_{\mathrm{o}})\, h(\mathbf{x}, \omega_{\mathrm{i}})\, d\sigma_{\mathbf{x}}^{\perp}(\omega_{\mathrm{i}})\,.$$

In particular, $\mathbf{K}$ *is self-adjoint if and only if* f_{s} *is symmetric for almost every* $\mathbf{x} \in \mathcal{M}$ *(i.e. except on a set of* A*-measure zero).*

Proof. We have

$$\begin{aligned} \langle f, \mathbf{K} g \rangle &= \int_{\mathcal{R}} f(\mathbf{x}, \omega) \int_{\mathcal{S}^2} f_{\mathrm{s}}(\mathbf{x}, \omega' \!\to\! \omega)\, g(\mathbf{x}, \omega')\, d\sigma_{\mathbf{x}}^{\perp}(\omega')\, d\mu(\mathbf{x}, \omega) \\ &= \int_{\mathcal{M}} \int_{\mathcal{S}^2} \int_{\mathcal{S}^2} f(\mathbf{x}, \omega)\, f_{\mathrm{s}}(\mathbf{x}, \omega' \!\to\! \omega)\, g(\mathbf{x}, \omega')\, d\sigma_{\mathbf{x}}^{\perp}(\omega')\, d\sigma_{\mathbf{x}}^{\perp}(\omega)\, dA(\mathbf{x}) \\ &= \int_{\mathcal{M}} \int_{\mathcal{S}^2} \int_{\mathcal{S}^2} f(\mathbf{x}, \omega)\, f_{\mathrm{s}}(\mathbf{x}, \omega' \!\to\! \omega)\, g(\mathbf{x}, \omega')\, d\sigma_{\mathbf{x}}^{\perp}(\omega)\, d\sigma_{\mathbf{x}}^{\perp}(\omega')\, dA(\mathbf{x}) \\ &= \int_{\mathcal{R}} \int_{\mathcal{S}^2} f_{\mathrm{s}}^*(\mathbf{x}, \omega \!\to\! \omega') f(\mathbf{x}, \omega)\, d\sigma_{\mathbf{x}}^{\perp}(\omega)\, g(\mathbf{x}, \omega')\, d\mu(\mathbf{x}, \omega') \\ &= \langle \mathbf{K}^* f, g \rangle\,, \end{aligned}$$

where we have used Fubini's theorem to change the order of integration. ∎

We are now in a position to study importance transport, which usually proceeds by writing down a formula for the equilibrium importance W and verifying that it has the desired properties. We will take the opposite approach, by starting with the fundamental relationship (4.21) for light transport,

$$I \;=\; \langle W_{\mathrm{e}}, \mathbf{G}\mathbf{S}_{L_{\mathrm{o}}} L_{\mathrm{e}} \rangle\,,$$

and then deriving the equations for importance based on the principle that we wish to compute the same value for the measurement I.

We start with the identity

$$I \;=\; \langle (\mathbf{G}\mathbf{S}_{L_{\mathrm{o}}})^* W_{\mathrm{e}}, L_{\mathrm{e}} \rangle\,,$$

which follows from the definition (4.22) of an adjoint operator.

Lemma 4.7. $(\mathbf{G}\mathbf{S}_{L_\text{o}})^* \;=\; \mathbf{G}\mathbf{S}_{W_\text{o}}$, *provided that* $\mathbf{S}_{L_\text{o}}$ *and* $\mathbf{S}_{W_\text{o}}$ *exist.*

Proof. We will make use of the following simple identities, which follow directly from the definition (4.22):

$$\begin{aligned}
\mathbf{I}^* &= \mathbf{I} \\
(\mathbf{A}+\mathbf{B})^* &= \mathbf{A}^* + \mathbf{B}^* \\
(\mathbf{A}\mathbf{B})^* &= \mathbf{B}^*\mathbf{A}^* \\
(\mathbf{A}^{-1})^* &= (\mathbf{A}^*)^{-1}
\end{aligned}$$

Provided that the operators $(\mathbf{I}-\mathbf{K}\mathbf{G})^{-1}$ and $(\mathbf{I}-\mathbf{K}^*\mathbf{G})^{-1}$ exist, we now have

$$\begin{aligned}
(\mathbf{G}\mathbf{S}_{L_\text{o}})^* &= (\mathbf{G}(\mathbf{I}-\mathbf{K}\mathbf{G})^{-1})^* &&= (\mathbf{I}-\mathbf{G}\mathbf{K}^*)^{-1}\mathbf{G} \\
&= \textstyle\sum_{i=0}^{\infty}(\mathbf{G}\mathbf{K}^*)^i\mathbf{G} &&= \textstyle\sum_{i=0}^{\infty}\mathbf{G}(\mathbf{K}^*\mathbf{G})^i \\
&= \mathbf{G}(\mathbf{I}-\mathbf{K}^*\mathbf{G})^{-1} &&= \mathbf{G}\mathbf{S}_{W_\text{o}} \,. \quad \blacksquare
\end{aligned}$$

We have thus proven that $I = \langle W_\text{e}, \mathbf{G}\mathbf{S}_{L_\text{o}} L_\text{e}\rangle = \langle \mathbf{G}\mathbf{S}_{W_\text{o}} W_\text{e}, L_\text{e}\rangle$, which is the basis for the following definition:

Definition 4.8. *The* exitant equilibrium importance function W_o *is defined by*

$$W_\text{o} \;=\; \mathbf{S}_{W_\text{o}} W_\text{e} \,.$$

Theorem 4.9. W_o *satisfies the transport equation* $W_\text{o} \;=\; W_\text{e} + \mathbf{T}_{W_\text{o}} W_\text{o}$, *where* $\mathbf{T}_{W_\text{o}} \;=\; \mathbf{K}^*\mathbf{G}$. *In particular,* L_o *and* W_o *obey the same transport equation when* $\mathbf{K} = \mathbf{K}^*$.

Proof. This follows directly from Lemma 4.7, Definition 4.8, and the definition $\mathbf{S}_{W_\text{o}} \;=\; (\mathbf{I} - \mathbf{K}^*\mathbf{G})^{-1}$ from Section 4.7. $\blacksquare$

Theorem 4.10. *The incident equilibrium quantities* L_i *and* W_i *satisfy*

$$\begin{aligned}
L_\text{i} &= L_\text{e,i} + \mathbf{G}\mathbf{K}L_\text{i} \,, \\
W_\text{i} &= W_\text{e,i} + \mathbf{G}\mathbf{K}^*W_\text{i} \,.
\end{aligned}$$

Proof. We have

$$\begin{aligned} L_{\mathrm{o}} &= L_{\mathrm{e}} + \mathbf{K}\mathbf{G}L_{\mathrm{o}} \\ \mathbf{G}L_{\mathrm{o}} &= \mathbf{G}L_{\mathrm{e}} + \mathbf{G}\mathbf{K}\mathbf{G}L_{\mathrm{o}} \\ L_{\mathrm{i}} &= L_{\mathrm{e,i}} + \mathbf{G}\mathbf{K}L_{\mathrm{i}}\,, \end{aligned}$$

where we have used the definition $L_{\mathrm{i}} = \mathbf{G}L_{\mathrm{o}}$, and we have assumed that emitted radiance is specified as an incident quantity $L_{\mathrm{e,i}}$. A similar relationship holds for W_{i}. ∎

The complete results are summarized in Section 4.7.

4.B.3 Norms

In this section, we prove conditions on the norms of $\mathbf{G}$ and $\mathbf{K}$ that are sufficient to ensure that the various solution operators $\mathbf{S}_X$ are well-defined. This is possible despite the fact that we can have $\|\mathbf{K}\| > 1$ for physically valid scene models.

The following theorem is similar to [Arvo 1995, Theorem 14]. However, our proof holds for two-sided surfaces, and involves only geometric concepts (Arvo's proof requires the principle that radiance is constant along straight lines in free space).

Theorem 4.11. *For any $1 \leq p \leq \infty$, we have $\|\mathbf{G}\|_p \leq 1$. Furthermore, $\|\mathbf{G}\|_p = 1$ unless $\mathcal{M}$ is contained by a plane in $\mathbb{R}^3$, in which case $\|\mathbf{G}\|_p = 0$.*

Proof. For any $1 \leq p < \infty$ and any $L \in L_p$, we have

$$\begin{aligned} \|\mathbf{G}L\|_p &= \left(\int_{\mathcal{R}} |(\mathbf{G}L)(\mathbf{r})|^p \, d\mu(\mathbf{r}) \right)^{1/p} \\ &= \left(\int_{\tilde{\mathcal{R}}} |L(M(\mathbf{r}))|^p \, d\mu(\mathbf{r}) \right)^{1/p} \\ &= \left(\int_{\tilde{\mathcal{R}}} |L(M(\mathbf{r}))|^p \, \frac{d\mu(\mathbf{r})}{d(\mu \circ M)} \, d\mu(M(\mathbf{r})) \right)^{1/p} \\ &= \left(\int_{\tilde{\mathcal{R}}} |L(\mathbf{r}')|^p \, d\mu(\mathbf{r}') \right)^{1/p} \\ &\leq \left(\int_{\mathcal{R}} |L(\mathbf{r}')|^p \, d\mu(\mathbf{r}') \right)^{1/p} \\ &= \|L\|_p\,, \end{aligned}$$

where we have used the fact that M is a measure-preserving bijection on $\tilde{\mathcal{R}}$ (Lemma 4.4). The case $p = \infty$ is similar, but only needs the fact that M is measure-preserving on sets of measure zero. Thus $\|\mathbf{G}\|_p \leq 1$ for all $1 \leq p \leq \infty$.

Now we consider the conditions for which $\|\mathbf{G}\|_p = 1$ exactly. If $\mu(\tilde{\mathcal{R}}) > 0$ (i.e. there are reversible rays), then consider the function L defined by

$$L(\mathbf{r}) \;=\; \begin{cases} 1 & \text{for } \mathbf{r} \in \tilde{\mathcal{R}}\,, \\ 0 & \text{otherwise}\,. \end{cases}$$

Then $\|\mathbf{G}L\|_p = \|L\|_p > 0$, and so $\|\mathbf{G}\|_p = 1$. On the other hand, if $\mu(\tilde{\mathcal{R}}) = 0$, then $\|\mathbf{G}L\|_p = 0$ for all L, and we have $\|\mathbf{G}\|_p = 0$.

Thus we must show that $\mu(\tilde{\mathcal{R}}) = 0$ if and only if $\mathcal{M}$ is contained by a plane in $\mathbb{R}^3$. The "if" direction is clear. For the converse, recall that $\mathcal{M}$ is a union of piecewise differentiable manifolds. Choose points $\mathbf{x}, \mathbf{x}' \in \mathcal{M}$ each lying in the differentiable interior of some manifold, and such that $\mathbf{x}'$ does not lie in the tangent plane at $\mathbf{x}$. (This is possible since $\mathcal{M}$ is not contained by any plane.) Because $\mathcal{M}$ is differentiable at $\mathbf{x}$ and $\mathbf{x}'$, we can choose small disk-shaped regions $dA(\mathbf{x}), dA(\mathbf{x}') \subset \mathcal{M}$ that are arbitrarily close to lying in the tangent planes at $\mathbf{x}$ and $\mathbf{x}'$ respectively. The set of rays leaving $dA(\mathbf{x})$ toward $dA(\mathbf{x}')$ now has positive μ-measure (even if $\mathbf{x}$ happens to lie in the tangent plane of $\mathbf{x}'$). Furthermore, these rays are reversible (even if there are other surfaces between $\mathbf{x}$ and $\mathbf{x}'$). ∎

The following results will be proven in Section 7.B.2. They are stated here for completeness.

Theorem 4.12. *For any physically valid scene, and for any $1 \le p \le \infty$,*

$$\|\mathbf{K}\|_p \;<\; \frac{\eta_{\max}^2}{\eta_{\min}^2}\,,$$

where $\eta_{\min}$ and $\eta_{\max}$ denote the minimum and maximum refractive indices in the environment.

Theorem 4.13. *For any physically valid scene, the solution operators $\mathbf{S}_X$ exist and are well-defined.*

Chapter 5

The Sources of Non-Symmetric Scattering

In this chapter, we study two examples of non-symmetric scattering that have not previously been recognized. Specifically, we show that non-symmetric scattering occurs whenever light is refracted, and also whenever shading normals are used. We show how to handle these situations correctly in bidirectional light transport algorithms, by deriving and using the corresponding adjoint BSDF's.

It is important to note that these sources of non-symmetry are not obvious, and that shading normals and refraction are widely assumed to be described by symmetric BSDF's. We show that this can cause significant problems when bidirectional algorithms are used. For example, it can cause rendered images to have large errors, even when the scene model is physically valid. It can also cause rendering algorithms that are supposedly equivalent to converge to different solutions (whether the scene model is physically valid or not). Finally, it can cause shading artifacts that should not be present, such as brightness discontinuities. These problems can occur whenever a non-symmetric BSDF is used without recognizing it (i.e. when it is handled as though it were symmetric).

We show that there are two distinct situations where non-symmetric BSDF's can arise. First, some scattering models in computer graphics are not physically valid. A good example of this is the use of shading normals (which are commonly applied to make polygonal surfaces look smooth, or to add detail to coarse geometric models). Although shading normals do not have any well-defined physical basis, they are very convenient and useful for many graphics applications. Thus it is important to be able to handle these non-physical materials correctly when bidirectional light transport algorithms are used.

The other situation where non-symmetric BSDF's arise is the refraction of light between two different media. Notice that in this case, the BSDF describes a real physical effect. This implies that even when a scene model is physically valid, it is sometimes necessary to use different transport rules for light and importance (or path tracing and particle tracing) in order for bidirectional algorithms to converge to the correct result.

This chapter is organized as follows. In Section 5.1 we explain why non-symmetric BSDF's are sometimes difficult to recognize, and we describe the significant problems that this can cause. We also discuss several elementary sources of non-symmetric scattering that are well-known in graphics. We then analyze in detail two sources of non-symmetry described above: namely refraction (Section 5.2) and the use of shading normals (Section 5.3). We also present test cases demonstrating the errors that occur in computed images when these BSDF's are not handled correctly.

Another contribution is the idea of Dirac distributions with respect to general measures, which can be used to model specular scattering and transport singularities in general. This concept is described in Appendix 5.A, along with several identities that can be used to manipulate and evaluate them in a consistent way. (Although this idea seems quite basic, we are unable to give a reference for it.)

5.1 Introduction

5.1.1 The problems caused by non-symmetric scattering

We explain the problems that arise when non-symmetric BSDF's are treated as though they were symmetric. This provides some motivation for the rest of this chapter, where we study the various reasons that non-symmetric scattering occurs.

The main problem with non-symmetric BSDF's is that they are sometimes difficult to recognize. Most often this occurs when a scattering model is defined procedurally (rather than by giving the BSDF as an explicit function). For example, refraction is generally implemented as a procedure that maps an incident direction $\omega_{\rm i}$ into a transmitted direction $\omega_{\rm t}$. The BSDF itself is not evaluated or even represented (because as we will see, it is a *Dirac distribution* rather than an ordinary function). Another example of a procedurally defined scattering model is the use of shading normals. With this technique, the true surface normal $\mathbf{N}_{\rm g}$ is replaced by a different vector $\mathbf{N}_{\rm s}$ (the *shading normal*) when the BSDF is evaluated. This can be interpreted as a procedural modification of an existing BSDF.

In these cases, it is easy to use a non-symmetric BSDF without realizing it. When this happens, the same BSDF $f_{\rm s}$ is used in all situations (even those where the adjoint BSDF $f_{\rm s}^*$ should be used instead). This creates problems, because the BSDF is not used consistently. For example, consider a bidirectional algorithm that uses particle tracing in one phase, and path tracing in another. Recall that in order to get correct results with such an algorithm, the adjoint BSDF $f_{\rm s}^*$ must be used during the particle tracing phase (see Section 3.7.5). Therefore, using the ordinary BSDF $f_{\rm s}$ for particle tracing is equivalent to solving a light transport equation that uses the adjoint BSDF $f_{\rm s}^*$. By using the same BSDF $f_{\rm s}$ in both phases, we get results that are almost certainly wrong: they could converge to the solution of a light transport equation that uses $f_{\rm s}$, $f_{\rm s}^*$, or any combination of the two. This has a number of practical consequences:

- Computed images can have substantial errors (even when the scene model is physically correct). The computed radiance values can easily be wrong by a factor of two or more.

- Rendering algorithms that are supposed to be equivalent may in fact converge to different answers. This can happen whether the model is physically valid or not. For example, path tracing might converge to a different result than particle tracing, because particle tracing must use the adjoint BSDF to get results that are consistent with path tracing.

- Computed images can have spurious, visually objectionable artifacts. For example, if the adjoint BSDF for shading normals is not used correctly, there can be false discontinuities in the image reminiscent of flat-shaded polygons. (This will be explained in Section 5.3.)

Note that these errors can occur with all types of bidirectional algorithm. For example, this includes importance-driven finite element methods, multi-pass algorithms, particle tracing approaches, and bidirectional path tracing.[1]

Fortunately, these problems are easy to fix. It is only necessary to recognize non-symmetric BSDF's whenever they exist, and make appropriate use of the corresponding adjoint BSDF.

5.1.2 Elementary sources of non-symmetric scattering

There are several reasons why the BSDF's used in graphics are sometimes not symmetric. One reason is that shading models are sometimes derived empirically, without regard for the laws of physics. Another reason is that shading models are sometimes approximated, to make them faster to evaluate. These sources of non-symmetry are well-known, and they are relatively easy to recognize and handle correctly.

On the other hand, some sources of non-symmetry are not so easily recognized. This category includes non-symmetry due to refraction and the use of shading normals, which are discussed separately in Sections 5.2 and 5.3.

5.1.2.1 Empirical shading models

The most obvious source of non-symmetric scattering is that some shading models are derived empirically, using formulas that are convenient to calculate and happen to give interesting visual results. Probably the best-known example of this is the original Phong model for glossy reflection [Phong 1975].[2] Translating his formula into our terminology, the reflected radiance from a glossy surface is computed according to

$$L_{\mathrm{o}}(\omega_{\mathrm{o}}) \;=\; \int_{\mathcal{H}_{\mathrm{i}}^2} C_{\mathrm{r}}\, \max(0, \omega_{\mathrm{i}} \cdot M_{\mathbf{N}(\mathbf{x})}(\omega_{\mathrm{o}}))^n \, L_{\mathrm{i}}(\omega_{\mathrm{i}})\, d\sigma(\omega_{\mathrm{i}})\,, \tag{5.1}$$

[1] Note that with importance-driven algorithms, the adjoint BSDF is only used to compute importance. Thus if we use the wrong BSDF, it will not cause errors in the solution (provided that importance is only used to guide mesh refinement). However, any error *estimates* that depend on importance will be wrong.

[2] Phong also proposed the use of shading normals, which is another source of non-symmetry.

where C_{r} controls the color and intensity of the glossy highlights, n controls the apparent specularity of the surface, and $M_{\mathbf{N}}(\omega_{\mathrm{o}})$ is the mirror direction (see Section 5.2.1.2).

Although this “shading formula” is symmetric, notice that the integration is with respect to solid angle σ, whereas in the scattering equation (3.12) the projected solid angle $\sigma_{\mathbf{x}}^{\perp}$ is used. Thus when this shading formula is expressed as a BRDF, it has an extra factor of

$$1 \,/\, |\omega_{\mathrm{i}} \cdot \mathbf{N}(\mathbf{x})|$$

that makes it non-symmetric. (This factor is required to cancel the factor of $|\omega_{\mathrm{i}} \cdot \mathbf{N}(\mathbf{x})|$ that is hidden by the projected solid angle notation (3.1).) It is easy to fix the non-symmetry, of course, by changing the definition (5.1) to use integration with respect to projected solid angle.

5.1.2.2 Approximations of symmetric BSDF’s

Non-symmetric BSDF’s can also arise when physically valid scattering models are approximated (to make their computation more efficient). For example, the Cook-Torrance model [Cook & Torrance 1982] has the form

$$f_{\mathrm{s}}(\omega_{\mathrm{i}} \!\rightarrow\! \omega_{\mathrm{o}}) \;=\; \frac{DGF}{\cos\theta_{\mathrm{i}}\cos\theta_{\mathrm{o}}}\,,$$

where D, G, and F are symmetric functions of ω_{i} and ω_{o}. This BSDF is clearly symmetric. However, notice that when it is inserted into the scattering equation (3.12), the factor of $\cos\theta_{\mathrm{i}}$ is canceled by the corresponding factor in the projected solid angle notation. When this formula is implemented in hardware, it is common to throw away the factor of $\cos\theta_{\mathrm{o}}$ as well. This saves a division operation, but destroys the symmetry of the corresponding BSDF. For example, this is the approach taken by the OpenGL specification [OpenGL Architecture Review Board 1992].

5.2 Non-symmetry due to refraction

We show that when light is refracted, the corresponding BSDF is not symmetric. In particular, we show that radiance crossing the interface must be scaled by a factor of $(\eta_{\mathrm{t}}/\eta_{\mathrm{i}})^2$, but

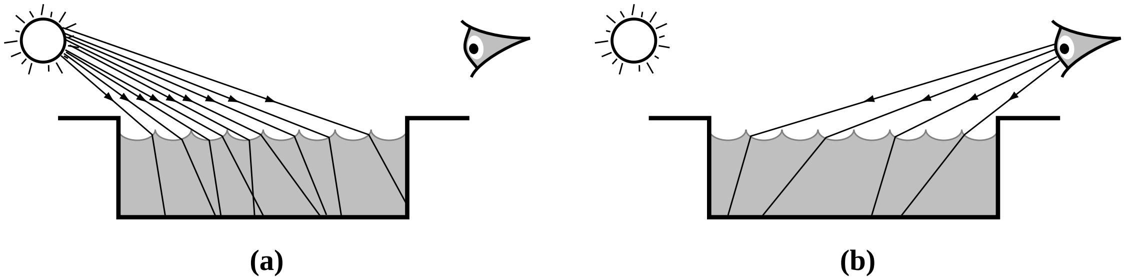


Figure 5.1: Two-pass rendering of caustics on a pool bottom. **(a)** First pass: particles are traced from the light sources, and their distribution on the pool bottom is recorded.**(b)** Second pass: an image is rendered using ray tracing. When rays intersect the pool bottom, the light distribution recorded by the first pass is sampled. To obtain correct results in this example, it is essential to handle particles and viewing rays differently at the air-water interface: radiance is scaled by $(\eta_t/\eta_i)^2$, while particle weights are left unchanged.

that no scaling is required for importance. (For simplicity we will ignore partial reflection in this section; in other words, we assume that all light is transmitted through the interface.) We derive explicit formulas for corresponding BSDF and its adjoint, and we discuss the implications for bidirectional rendering algorithms.

Note that there can be substantial errors if the $(\eta_t/\eta_i)^2$ scaling factor for radiance is ignored. For example, consider a light source shining on a swimming pool with a diffuse bottom and sides (see Figure 5.1). Suppose that particle tracing is used to accumulate the caustic pattern on the bottom of the pool, followed by a ray tracing pass to render the final image. If the radiance of the viewing rays is not scaled at the air-water interface, the caustics in the image will be too bright by a factor of $(\eta_t/\eta_i)^2$ (about 1.78 for water). In particular, the caustics will be brighter than they would be in a path-traced image. On the other hand, if the $(\eta_t/\eta_i)^2$ scaling factor is applied to *both* the viewing rays and the particles, the caustics will be too dim by a factor of $(\eta_t/\eta_i)^2$.

The main point of this section is not that radiance must be scaled when it enters a medium with a different refractive index; this fact is well-known in radiometry and optics, although it does not seem to have been implemented in many graphics systems. Rather, the point is that the adjoint BSDF does *not* involve any such scaling. Thus the BSDF is not symmetric, and so different rules must be used for radiance and importance (or particle tracing and path tracing) in order to obtain correct results.

5.2.1 Background material

We review two results that will be used to derive the BSDF for refraction (and its adjoint). First, we show that when light is refracted, radiance is scaled by a factor of $(\eta_{\mathrm{t}}/rii)^2$ as it crosses the interface. Second, we show how to write the BSDF for reflection from a perfect mirror, using a new notation involving a Dirac distribution with respect to the projected solid angle measure.

5.2.1.1 Radiance scaling at a refractive interface

Intuitively, when light enters a medium with a higher refractive index, the same light energy is squeezed into a smaller volume. To see this, consider a small patch $dA(\mathbf{x})$ that is exposed to uniform radiance over the incident hemisphere $\mathcal{H}^2_{\mathrm{i}}$, and assume that this light is transmitted into a medium with a higher refractive index (Figure 5.2). Then the transmitted light does not fill the entire hemisphere $\mathcal{H}^2_{\mathrm{t}}$, since by Snell's law, the angle of refraction satisfies

$$\sin\theta_{\mathrm{t}} \;<\; \frac{\eta_{\mathrm{i}}}{\eta_{\mathrm{t}}}\,.$$

Thus radiance must increase as light crosses the interface (at least on some subset of the rays), by conservation of energy.

In fact, the incident and transmitted radiance are related by

$$L_{\mathrm{t}} \;=\; \frac{\eta_{\mathrm{t}}^2}{\eta_{\mathrm{i}}^2}\,L_{\mathrm{i}}\,. \tag{5.2}$$

This can be shown using Snell's law (e.g. [Milne 1930, p. 74], [Nicodemus 1963], [Hall 1989, p. 30]). We repeat this argument here, since we will need some of the intermediate results.

Consider a beam of light that strikes small surface patch $dA(\mathbf{x})$, and occupies a solid angle $d\sigma(\omega_{\mathrm{i}})$ (see Figure 5.3). Let ω_{t} be the direction of the refracted beam (determined using Snell's law), and suppose that it occupies a solid angle $d\sigma(\omega_{\mathrm{t}})$. The power carried by

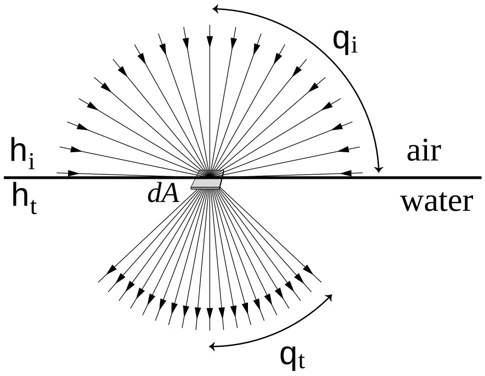


Figure 5.2: When light enters a medium with a higher refractive index, the same light energy is squeezed into a smaller volume. This causes the radiance along each ray to increase.

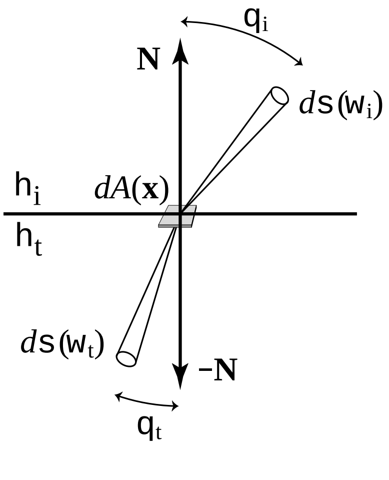


Figure 5.3: Geometry for deriving the $(\eta_t/\eta_i)^2$ scaling of radiance due to refraction.

the incident beam is

$$d\Phi_{\rm i} \;=\; L_{\rm i}\, dA(\mathbf{x})\, d\sigma^{\perp}(\omega_{\rm i})\;,$$

where $L_{\rm i}$ is the radiance of the incident beam (see (3.8)), and similarly

$$d\Phi_{\rm t} \;=\; L_{\rm t}\, dA(\mathbf{x})\, d\sigma^{\perp}(\omega_{\rm t})\;.$$

Thus by conservation of energy,

$$L_{\rm t} \;=\; \frac{d\sigma^{\perp}(\omega_{\rm i})}{d\sigma^{\perp}(\omega_{\rm t})}\, L_{\rm i}\;. \tag{5.3}$$

We now turn to the angular parameterization $\omega \equiv (\theta, \phi)$, for which we have

$$d\sigma^{\perp}(\omega) = \cos\theta \sin\theta\, d\theta\, d\phi$$

(see Section 3.6.3). Using this relationship, we can relate $d\sigma^{\perp}(\omega_{\rm i})$ and $d\sigma^{\perp}(\omega_{\rm t})$ by differentiating Snell's law:

$$\begin{aligned} \eta_{\rm i} \sin\theta_{\rm i} &\;=\; \eta_{\rm t} \sin\theta_{\rm t} \qquad \text{(Snell's law)} \\ \eta_{\rm i} \cos\theta_{\rm i}\, d\theta_{\rm i} &\;=\; \eta_{\rm t} \cos\theta_{\rm t}\, d\theta_{\rm t}\;. \end{aligned}$$

Similarly, the relationship $\phi_{\rm t} = \phi_{\rm i} \pm \pi$ implies

$$d\phi_{\rm i} \;=\; d\phi_{\rm t}\;.$$

Multiplying these three equations together and using (3.16), we get

$$\eta_{\rm i}^2\, d\sigma^{\perp}(\omega_{\rm i}) \;=\; \eta_{\rm t}^2\, d\sigma^{\perp}(\omega_{\rm t})\;, \tag{5.4}$$

which together with (5.3) gives the desired relationship

$$L_{\rm t} \;=\; \frac{\eta_{\rm t}^2}{\eta_{\rm i}^2}\, L_{\rm i}\;.$$

To be precise, this equation only applies to spectral radiance that is measured with respect to frequency (L_ν). For spectral radiance that is measured with respect to wavelength (L_λ), the correct relationship is $L_{\rm t} \;=\; (\eta_{\rm t}/\eta_{\rm i})^3 L_{\rm i}$ (as we will discuss in Section 6.2).

5.2.1.2 The BSDF for specular reflection

We will study in detail the BSDF that describes specular reflection, i.e. a perfect (two-sided) mirror. This BSDF sometimes causes confusion, because it involves Dirac distributions. We will introduce a new, simpler notation for this BSDF, using Dirac distributions defined on the unit sphere. The concepts developed here will be needed below, when we study perfect specular refraction. (Further information on the mirror BSDF can be found in Nicodemus et al. [1977], Cohen & Wallace [1993], and Glassner [1995].)

For a perfect mirror, the desired relationship between L_{i} and L_{o} is that

$$L_{\mathrm{o}}(\omega_{\mathrm{o}}) \;=\; L_{\mathrm{i}}(M_{\mathbf{N}}(\omega_{\mathrm{o}}))\,. \tag{5.5}$$

Here $M_{\mathbf{N}}(\omega_{\mathrm{o}})$ is the *mirror direction*, obtained by reflecting ω_{o} around the normal $\mathbf{N}$. (Algebraically, the mirror direction is defined by $M_{\mathbf{N}}(\omega_{\mathrm{o}}) = 2(\omega_{\mathrm{o}} \cdot \mathbf{N})\mathbf{N} - \omega_{\mathrm{o}}$.)

We would like to find a BSDF that produces the relationship (5.5) when it is inserted into the scattering equation (3.12). We will show how to define this BSDF in terms of a special *Dirac distribution* $\delta_{\sigma^\perp}$, which is defined by the property that

$$\int_{\mathcal{S}^2} f(\omega)\,\delta_{\sigma^\perp}(\omega - \omega')\,d\sigma^\perp(\omega) \;=\; f(\omega')$$

for any function f that is continuous at ω'.

Our notion of a Dirac distribution is slightly more general than the one usually encountered. Often, the Dirac distribution (or *delta function*) is understood to be a “function” $\delta(x)$ defined on $\mathbb{R}$ with the following properties:

1. $\delta(x) = 0$ for all $x \neq 0$.

2. $\int_{\mathbb{R}} \delta(x)\,dx = 1$.

These imply the more useful identity that

$$\int_{\mathbb{R}} f(x)\,\delta(x - x_0)\,dx \;=\; f(x_0)\,, \tag{5.6}$$

provided that f is continuous at x_0.

Our notation simply extends the identity (5.6) to integration on more general domains. Given a domain Ω, a measure μ on Ω, and a function $f : \Omega \to \mathbb{R}$ that is continuous at $\mathbf{x}_0$, the notation $\delta_\mu(\mathbf{x} - \mathbf{x}_0)$ refers to a "function" with the following property:

$$\int_\Omega f(\mathbf{x})\, \delta_\mu(\mathbf{x} - \mathbf{x}_0)\, d\mu(\mathbf{x}) \;=\; f(\mathbf{x}_0)\,. \tag{5.7}$$

The rigorous meaning of this notation is discussed in Appendix 5.A.

Given this background, we can write the BSDF for a perfect mirror as

$$f_{\rm s}(\omega_{\rm i} \!\rightarrow\! \omega_{\rm o}) \;=\; \delta_{\sigma^\perp}(\omega_{\rm i} - M_{\mathbf{N}}(\omega_{\rm o}))\,. \tag{5.8}$$

It is easy to see that this is a correct representation of the mirror BSDF, by inserting these definitions into the scattering equation (3.12) to obtain (5.5).

This BSDF can be written in several other equivalent ways. For example, suppose that we write the scattering equation (3.12) in the form

$$L_{\rm o}(\omega_{\rm o}) \;=\; \int_{\mathcal{S}^2} L_{\rm i}(\omega_{\rm i})\, f_{\rm s}(\omega_{\rm i} \!\rightarrow\! \omega_{\rm o})\, |\omega_{\rm i} \cdot \mathbf{N}|\, d\sigma(\omega_{\rm i}) \tag{5.9}$$

(by expanding the definition of the projected solid angle measure). From this equation, it is clear that the mirror BSDF could also be written as

$$f_{\rm s}(\omega_{\rm i} \!\rightarrow\! \omega_{\rm o}) \;=\; \frac{\delta_\sigma(\omega_{\rm i} - M_{\mathbf{N}}(\omega_{\rm o}))}{|\omega_{\rm i} \cdot \mathbf{N}|}\,. \tag{5.10}$$

Note that expressions containing Dirac distributions must be evaluated with great care. This is particularly true when the measure function associated with the Dirac distribution is different than the measure function used for integration (e.g. suppose that we are given the form (5.8) of the mirror BSDF, together with the form (5.9) of the scattering equation). In Appendix 5.A, we derive several identities that allow such expressions to be evaluated correctly and easily.

5.2.2 The BSDF for refraction

We will write the BSDF for refraction using the Dirac distribution notation developed in Section 5.2.1.2 and Appendix 5.A. It can also be written with ordinary δ-functions, using the (θ, ϕ) parameterization of BSDF's, as we will discuss in Appendix 5.C.

For a fixed point $\mathbf{x} \in \mathcal{M}$, let Ω_R be the set of directions $\omega_{\rm i} \in \mathcal{S}^2$ that are not subject to total internal reflection. We now define a mapping

$$R : \Omega_R \to \Omega_R \,,$$

such that $R(\omega_{\rm i})$ is the transmitted direction corresponding to the incident direction $\omega_{\rm i}$:[3]

$$\omega_{\rm t} \;=\; R(\omega_{\rm i}) \,.$$

Note that R is easily seen to be its own inverse:

$$R(R(\omega)) \;=\; \omega \qquad \text{for all } \omega \in \Omega_R \,. \tag{5.11}$$

Given this mapping, the relationship between $L_{\rm i}$ and $L_{\rm t}$ due to refraction can be expressed as

$$L_{\rm t}(\omega_{\rm t}) \;=\; \frac{\eta_{\rm t}^2}{\eta_{\rm i}^2}\, L_{\rm i}(R(\omega_{\rm t})) \,, \tag{5.12}$$

where we have used the self-inverse property (5.11) to obtain $\omega_{\rm i} = R(\omega_{\rm t})$. The corresponding BSDF is thus

$$f_{\rm s}(\omega_{\rm i} \!\to\! \omega_{\rm t}) \;=\; \frac{\eta_{\rm t}^2}{\eta_{\rm i}^2}\, \delta_{\sigma^\perp}(\omega_{\rm i} - R(\omega_{\rm t})) \,, \tag{5.13}$$

where $\delta_{\sigma^\perp}$ is the Dirac distribution with respect to $\sigma^\perp$, as defined in Section 5.2.1.2. Inserting this BSDF into the scattering equation (3.12), it is easy to check that we get the desired relationship (5.12).

This is the simplest way to write the BSDF from a conceptual point of view; it expresses the desired relationship between $\omega_{\rm i}$ and $\omega_{\rm t}$, and also the fact that radiance is scaled by a factor of $(\eta_{\rm t}/\eta_{\rm i})^2$, with a minimum of extra clutter.

[3] Algebraically, R is defined by

$$R(\theta_{\rm i}, \phi_{\rm i}) \;=\; (\theta_{\rm t}, \phi_{\rm t}) \;=\; (\sin^{-1}(\frac{\eta_{\rm t}}{\eta_{\rm i}} \sin\theta_{\rm i}), \phi_{\rm i} \pm \pi) \,,$$

where $\theta_{\rm t}$ is chosen to lie on the opposite side of the surface as $\theta_{\rm i}$, i.e. $\cos\theta_{\rm i}\,\cos\theta_{\rm t} \le 0$. The symbols $\eta_{\rm i}$ and $\eta_{\rm t}$ denote the refractive indices on the side of the surface containing $\omega_{\rm i}$ and $\omega_{\rm t}$ respectively. (Since $\omega_{\rm i}$ can lie on either side of the surface, this means that $\eta_{\rm i}$ is actually a function of $\theta_{\rm i}$.)

Notice that we have used the angular parameterization $\omega \equiv (\theta, \phi)$ to define R. It is possible to define $R(\omega)$ directly in terms of the unit vector ω, but the result is relatively complicated. The vector form is commonly used in implementations, for example see [Glassner 1989, p. 298]).

5.2.3 The adjoint BSDF for refraction

We now derive the adjoint BSDF for refraction. We will make use of the following general symmetry relationship, which holds for any physically valid BSDF:[4]

$$\frac{f_{\mathrm{s}}(\omega_{\mathrm{i}} \to \omega_{\mathrm{o}})}{\eta_{\mathrm{o}}^2} = \frac{f_{\mathrm{s}}(\omega_{\mathrm{o}} \to \omega_{\mathrm{i}})}{\eta_{\mathrm{i}}^2} \,. \tag{5.14}$$

This fact will be derived in Chapter 6. For now, we can use it to immediately obtain the adjoint BSDF for refraction:

$$\begin{aligned} f_{\mathrm{s}}^*(\omega_{\mathrm{i}} \to \omega_{\mathrm{t}}) &= f_{\mathrm{s}}(\omega_{\mathrm{t}} \to \omega_{\mathrm{i}}) \\ &= (\eta_{\mathrm{i}}/\eta_{\mathrm{t}})^2 f_{\mathrm{s}}(\omega_{\mathrm{i}} \to \omega_{\mathrm{t}}) \\ &= \delta_{\sigma^\perp}(\omega_{\mathrm{i}} - R(\omega_{\mathrm{t}}))\,, \end{aligned} \tag{5.15}$$

where we have used (5.14) and (5.13) in the second and third lines respectively.

The main thing to notice is that the $(\eta_{\mathrm{t}}/\eta_{\mathrm{i}})^2$ factor is *not* present in the adjoint BSDF. Thus, importance and light particles are not scaled when they cross the interface. (Notice that this corresponds to the intuitive idea that light particles carry “power”, since power (unlike radiance) is conserved when light enters a different medium.)

In an implementation, the difference between f_{s} and f_{s}^* must be represented explicitly. It is not possible to evaluate the adjoint BSDF by just exchanging the directional arguments, since there is no way to evaluate the BSDF at all. Specular BSDF’s contain Dirac distributions, which means that the only allowable operation is sampling: there must be an explicit procedure that generates a sample direction and a weight. When the specular BSDF is not symmetric, the direction and/or weight computation for the adjoint is different, and thus there must be two different sampling procedures, or an explicit flag that specifies whether the direct or adjoint BSDF is being sampled.

In Appendix 5.B, we give a different derivation of the adjoint BSDF for refraction. The problem with the derivation given here is that it depends on the laws of physics, by way of the symmetry condition (5.14). Since the adjoint BSDF is a purely mathematical concept,

[4] Equation (5.14) applies to the BSDF’s of a much larger class of surfaces than we consider here, including frosted glass for example. Perfect refraction corresponds to the special case of an optically smooth interface between two dielectric media.

it should be possible to derive it mathematically, and this is what is done in Appendix 5.B.

5.2.4 Results

Figure 5.4 shows a pool of water with small waves, illuminated by two area light sources. This image simulates the results of a two-pass rendering algorithm, consisting of a particle tracing pass followed by a ray tracing pass (Figure 5.1).[5] The waves (and the floor) were modeled using bump mapping.

Figure 5.4(a) shows the correct image, whose computation requires that viewing rays and particles be handled differently at the air-water interface. The radiance along viewing rays is scaled by $(\eta_{\mathrm{t}}/\eta_{\mathrm{i}})^2$ when they cross the interface, but the particle weights are left unchanged.

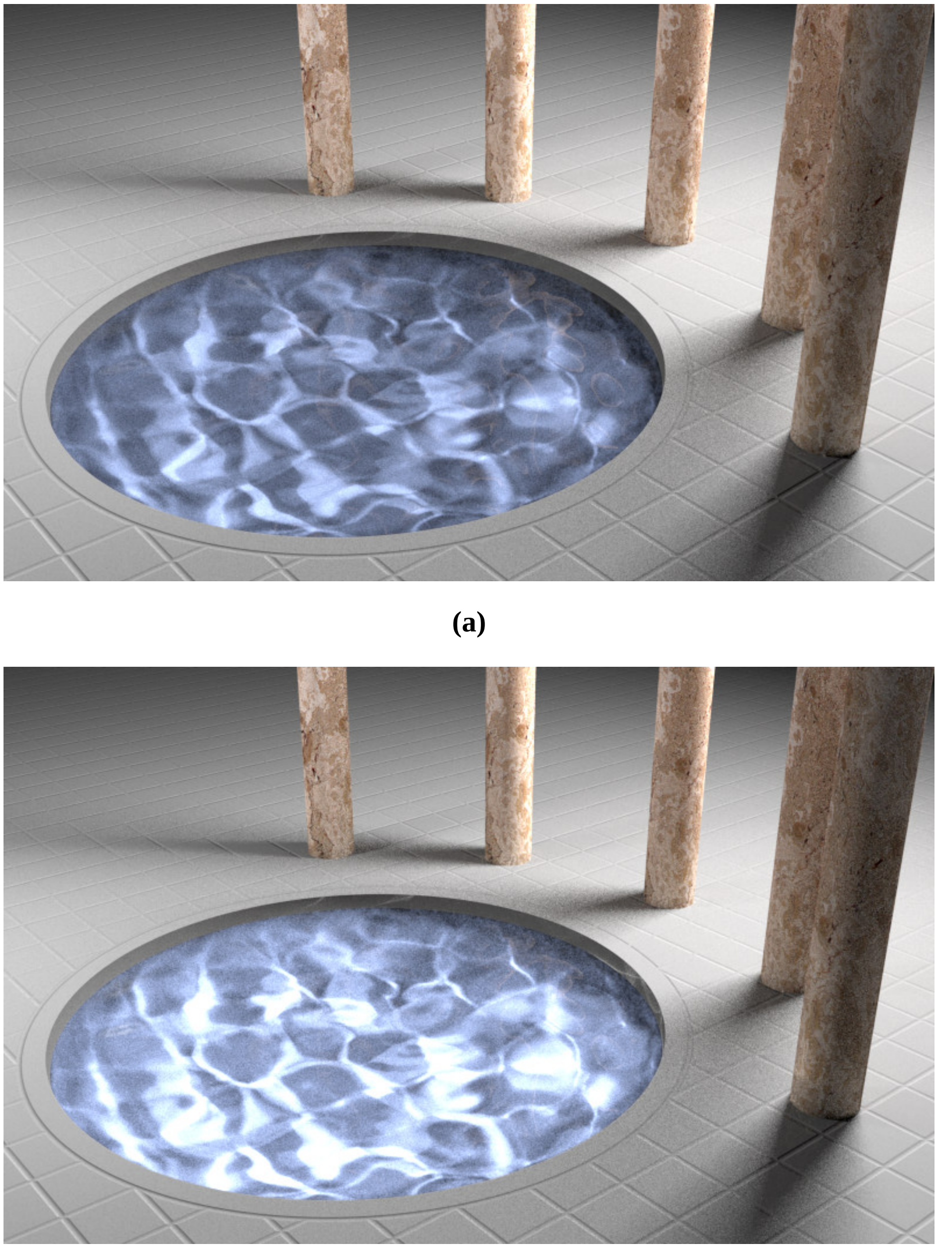
Figure 5.4(b) shows the errors that occur when the BSDF at the air-water interface is assumed to be symmetric, i.e. when the same scattering rules are used for viewing rays and particles (for this image, neither one was scaled). This leads to caustics that are too bright, by a factor of $(\eta_{\mathrm{t}}/\eta_{\mathrm{i}})^2$.

5.2.5 Discussion

Hall [1989] pointed out that radiance should be scaled by $(\eta_{\mathrm{t}}/\eta_{\mathrm{i}})^2$ at a refractive interface, but this fact has been ignored by most ray tracing systems. Our results take this one step further, by showing that the $(\eta_{\mathrm{t}}/\eta_{\mathrm{i}})^2$ scaling should not be applied to importance or light particles. We are not aware of any system (other than ours) that implements different scattering rules for radiance vs. importance or path tracing vs. particle tracing in this way. As we have shown, this is easy to do, and essential for the correctness of bidirectional algorithms.[6]

[5] Both of these images were actually computed using the Metropolis light transport algorithm (Chapter 11), with modifications that simulate the results of two-pass algorithms such as [Shirley et al. 1995, Jensen 1996].

[6] The radiance scaling is not as important for ray tracing or path tracing, since when a path enters and exits a given medium, the two factors cancel out. However, the results of these algorithms will be incorrect when the sources and sensors are in different media (e.g. underwater lights). Here we have assumed that the underwater lights are modeled as direct emitters, rather than as a filament surrounded by a glass shell (since most rendering algorithms would be very inefficient if this representation were used). However, note that errors occur in both cases when bidirectional methods are used (e.g. recall the pool example, for which the source and sensor were in the same medium).



(a)

(b)

Figure 5.4: (a) A pool of water as it would be rendered by a particle tracing algorithm (reference image). **(b)** Incorrect caustics (too bright), caused by assuming that refraction between air and water is modeled by a symmetric BSDF.

Another approach to correctly handling refraction is to find a way to represent it by a symmetric function. This seems plausible, given the existence of the general symmetry condition (5.14). This topic is explored further in Chapter 7, where we derive a framework for light transport in which light, importance, and particles all obey the same scattering rules.

5.3 Non-symmetry due to shading normals

Shading normals are used to change the apparent orientation of a surface without changing its geometry. The mechanism is simple: when the surface is shaded (e.g. using the scattering equation (3.12)), the surface normal $\mathbf{N}(\mathbf{x})$ is replaced by a different, arbitrary direction vector. The new direction is called the *shading normal* $\mathbf{N}_{\mathrm{s}}(\mathbf{x})$, and corresponds to the desired orientation of the surface. To avoid confusion, we will refer to the true surface normal as the *geometric normal* $\mathbf{N}_{\mathrm{g}}(\mathbf{x})$.

Shading normals are useful tool for many graphics applications. For example, their original purpose was to make polygonal surfaces appear more smooth [Phong 1975]. To do this, a *vertex normal* is defined at each vertex of a polygonal mesh. Shading normals are obtained by linearly interpolating the vertex normals across each polygonal face, to give the appearance of a smoothly changing surface orientation. This simple technique is still widely in use today, because computer models of smooth surfaces are usually converted to polygons before they are rendered.

Shading normals are also used for *bump mapping* [Blinn 1978]. This is a technique for adding detail to surfaces that are otherwise smooth and uninteresting. By perturbing the surface normal, it is possible to create the impression of high geometric complexity; for example, a flat rectangle can be given the appearance of a stucco wall.

However, there is a “catch”. As we will explain, shading normals modify the BSDF of the material to which they are applied. (In some sense this is obvious, since shading normals change the surface appearance, and surface appearance is completely determined by the BSDF.) Unfortunately, the modified BSDF does not possess the same properties as the original: in general it is not symmetric, and it does not conserve energy. It should not be surprising that shading normals cause problems, since they do not have any physical basis.

Nevertheless, shading normals are a useful tool for many graphics applications, and we observe that there is still a well-defined set of equations to be solved. By deriving and using the correct adjoint BSDF, we can ensure that different bidirectional light transport algorithms all converge to the same mathematically correct solution. On the other hand, we show that if the non-symmetry due to shading normals is not recognized (i.e. the adjoint BSDF is not used), then different rendering algorithms will converge to different results. This is clearly undesirable.

One way to see the problems caused by shading normals is to observe that some certain calculations still depend on the geometric normal. By changing the normal used in some calculations, but not in others, we get an inconsistent representation of the scene model. In a particle tracing simulation, for example, consider the number of particles received by a given polygon A. Clearly this depends on the geometric normal of A, rather than its shading normal. Similarly, observe that the solid angle subtended by a polygon is not affected by its shading normal. These inconsistencies between geometric and shading normals can cause problems, unless the correct adjoint BSDF is used.

5.3.1 How shading normals modify the BSDF

Let $\mathbf{x} \in \mathcal{M}$ be a fixed point, so that we can omit $\mathbf{x}$ from our notation. When shading normals are not considered, recall that the radiance leaving $\mathbf{x}$ can be evaluated using the scattering equation (5.9),

$$\begin{aligned} L_{\mathrm{o}}(\omega_{\mathrm{o}}) &= \int_{\mathcal{S}^2} L_{\mathrm{i}}(\omega_{\mathrm{i}})\, f_{\mathrm{s},\mathbf{N}_{\mathrm{g}}}(\omega_{\mathrm{i}} \to \omega_{\mathrm{o}})\, d\sigma^{\perp}(\omega_{\mathrm{i}}) \\ &= \int_{\mathcal{S}^2} L_{\mathrm{i}}(\omega_{\mathrm{i}})\, f_{\mathrm{s},\mathbf{N}_{\mathrm{g}}}(\omega_{\mathrm{i}} \to \omega_{\mathrm{o}})\, |\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{g}}|\, d\sigma(\omega_{\mathrm{i}})\,. \end{aligned}$$

With shading normals, however, the following equation is used instead (see Figure 5.5):

$$L_{\mathrm{o}}(\omega_{\mathrm{o}}) = \int_{\mathcal{S}^2} L_{\mathrm{i}}(\omega_{\mathrm{i}})\, f_{\mathrm{s},\mathbf{N}_{\mathrm{s}}}(\omega_{\mathrm{i}} \to \omega_{\mathrm{o}})\, |\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{s}}|\, d\sigma(\omega_{\mathrm{i}})\,, \tag{5.16}$$

i.e. the shading normal is used when evaluating the BSDF and the projected solid angle.

This formula is very effective at changing the apparent orientation of a surface (from $\mathbf{N}_{\mathrm{g}}$ to $\mathbf{N}_{\mathrm{s}}$). However, it does not actually change the surface geometry. Instead, the shading normal should be thought of as a parameter that modifies the BSDF. We can write an explicit

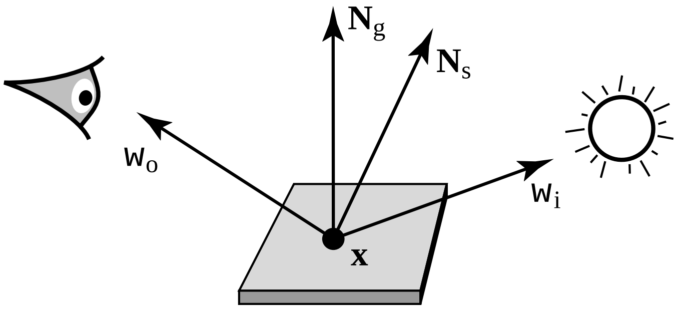


Figure 5.5: Geometry for defining the effect of shading normals.

formula for this modified BSDF, by converting the shading formula (5.16) to the standard form (3.12). In other words, we would like to write it in the form

$$L_\mathrm{o}(\omega_\mathrm{o}) \;=\; \int_{\mathcal{S}^2} L_\mathrm{i}(\omega_\mathrm{i})\, \bar{f}_\mathrm{s}(\omega_\mathrm{i} \!\to\! \omega_\mathrm{o})\, |\omega_\mathrm{i} \cdot \mathbf{N}_\mathrm{g}|\, d\sigma(\omega_\mathrm{i})\,.$$

This can be achieved by letting $\bar{f}_\mathrm{s}$ be the *modified BSDF* given by

$$\bar{f}_\mathrm{s}(\omega_\mathrm{i} \!\to\! \omega_\mathrm{o}) \;=\; f_{\mathrm{s},\mathbf{N}_\mathrm{s}}(\omega_\mathrm{i} \!\to\! \omega_\mathrm{o})\, \frac{|\omega_\mathrm{i} \cdot \mathbf{N}_\mathrm{s}|}{|\omega_\mathrm{i} \cdot \mathbf{N}_\mathrm{g}|}\,. \tag{5.17}$$

It is easy to verify that the formula obtained by using this BSDF is indistinguishable from the original shading formula (5.16) (notice that the two $|\omega_\mathrm{i} \cdot \mathbf{N}_\mathrm{g}|$ factors cancel each other). We have simply interpreted the calculation in a different way.

5.3.2 The adjoint BSDF for shading normals

However, even if the original BSDF was symmetric, the modified BSDF is not. Its adjoint $\bar{f}_\mathrm{s}^*$ is given by

$$\begin{aligned} \bar{f}_\mathrm{s}^*(\omega_\mathrm{i} \!\to\! \omega_\mathrm{o}) \;&=\; \bar{f}_\mathrm{s}(\omega_\mathrm{o} \!\to\! \omega_\mathrm{i}) \\ &=\; f_{\mathrm{s},\mathbf{N}_\mathrm{s}}(\omega_\mathrm{o} \!\to\! \omega_\mathrm{i})\, \frac{|\omega_\mathrm{o} \cdot \mathbf{N}_\mathrm{s}|}{|\omega_\mathrm{o} \cdot \mathbf{N}_\mathrm{g}|}\,. \end{aligned} \tag{5.18}$$

This is the BSDF that must be used for importance transport and particle tracing.

For example, the scattering equation for importance is given by

$$\begin{aligned} W_{\mathrm{o}}(\omega_{\mathrm{o}}) &= \int_{\mathcal{S}^2} W_{\mathrm{i}}(\omega_{\mathrm{i}})\, \bar{f}_{\mathrm{s}}^{*}(\omega_{\mathrm{i}} \to \omega_{\mathrm{o}})\, d\sigma^{\perp}(\omega_{\mathrm{i}}) \\ &= \int_{\mathcal{S}^2} W_{\mathrm{i}}(\omega_{\mathrm{i}})\, f_{\mathrm{s},\mathbf{N}_{\mathrm{s}}}(\omega_{\mathrm{o}} \to \omega_{\mathrm{i}})\, \frac{|\omega_{\mathrm{o}} \cdot \mathbf{N}_{\mathrm{s}}|}{|\omega_{\mathrm{o}} \cdot \mathbf{N}_{\mathrm{g}}|}\, |\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{g}}|\, d\sigma(\omega_{\mathrm{i}})\,. \end{aligned}$$

This will give results that are consistent with the formula (5.16) for radiance evaluation. Notice that the formula for evaluating importance is more complex than the formula for evaluating radiance, because there is no cancelation of the $|\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{g}}|$ factors.

Similarly, recall that the adjoint BSDF is used in particle tracing (see Sections 3.7.5 and 4.A). Given a particle that arrives from direction ω_{o} and is scattered in direction ω_{i}, the particle weight should be multiplied by

$$\begin{aligned} \alpha(\omega_{\mathrm{i}}) &= \frac{\bar{f}_{\mathrm{s}}^{*}(\omega_{\mathrm{i}} \to \omega_{\mathrm{o}})\, |\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{g}}|}{P_{\sigma}(\omega_{\mathrm{i}})} \\ &= \frac{f_{\mathrm{s},\mathbf{N}_{\mathrm{s}}}(\omega_{\mathrm{o}} \to \omega_{\mathrm{i}})}{P_{\sigma}(\omega_{\mathrm{i}})}\, \frac{|\omega_{\mathrm{o}} \cdot \mathbf{N}_{\mathrm{s}}|}{|\omega_{\mathrm{o}} \cdot \mathbf{N}_{\mathrm{g}}|}\, |\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{g}}|\,, \qquad (5.19) \end{aligned}$$

where $P_{\sigma}(\omega_{\mathrm{i}})$ is the density with respect to solid angle for sampling direction ω_{i} (see equations (4.33) and (5.18)). If particles are weighted in this way, the results will be consistent with the desired shading formula (5.16).

5.3.3 Examples of shading normal BSDF's and their adjoints

We show how these results apply to diffuse surfaces (i.e. Lambertian), and perfect specular surfaces (i.e. mirrors).

For a diffuse surface, we will show that the importance sampling techniques needed for ray tracing and particle tracing are different. We start with the constant BRDF:

$$f_{\mathrm{r}}(\omega_{\mathrm{i}} \to \omega_{\mathrm{o}}) = K_{\mathrm{d}}\,.$$

For radiance evaluation (ray tracing), we insert this in (5.16) to obtain

$$L_{\mathrm{o}}(\omega_{\mathrm{o}}) = \int_{\mathcal{S}^2} K_{\mathrm{d}}\, L_{\mathrm{i}}(\omega_{\mathrm{i}})\, |\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{s}}|\, d\sigma(\omega_{\mathrm{i}})\,.$$

For particle tracing, according to (5.19) scattered particle weights should be multiplied by

$$\alpha(\omega_{\rm i}) \;=\; \frac{K_{\rm d}}{p_{\omega_{\rm o}}(\omega_{\rm i})}\,\frac{|\omega_{\rm o}\cdot\mathbf{N}_{\rm s}|}{|\omega_{\rm o}\cdot\mathbf{N}_{\rm g}|}\,|\omega_{\rm i}\cdot\mathbf{N}_{\rm g}|\,,$$

where we have used the convention that particles go from $\omega_{\rm o}$ to $\omega_{\rm i}$.

These equations imply that the importance sampling techniques needed for ray tracing and particle tracing are completely different. In both cases, the task is to sample an appropriate direction $\omega_{\rm i}$, when $\omega_{\rm o}$ is already given. For ray tracing, $\omega_{\rm i}$ should be chosen with probability proportional to $|\omega_{\rm i}\cdot\mathbf{N}_{\rm s}|$, since this is the factor by which incoming radiance is weighted. However, for particles, $\omega_{\rm i}$ should be chosen according to the cosine with the *geometric* normal, $|\omega_{\rm i}\cdot\mathbf{N}_{\rm g}|$. (The weighting factors involving $\omega_{\rm o}$ are irrelevant, since they depend only on the direction the particle arrived from.) The fact that two different density functions are needed for sampling can have important implications for rendering system design (see Section 5.3.4).

As another example, consider a perfect (two-sided) mirror. We will show that reflected particles (or importance) must be weighted by an extra factor of

$$\alpha \;=\; \frac{|\omega_{\rm i}\cdot\mathbf{N}_{\rm g}|}{|\omega_{\rm o}\cdot\mathbf{N}_{\rm g}|}$$

to get correct results.

To show this, recall that the BSDF for a perfect mirror was derived in Section 5.2.1.2 as

$$f_{{\rm s},\mathbf{N}}(\omega_{\rm i}\!\rightarrow\!\omega_{\rm o}) \;=\; \frac{\delta_\sigma(\omega_{\rm i}-M_{\mathbf{N}}(\omega_{\rm o}))}{|\omega_{\rm i}\cdot\mathbf{N}|}\,.$$

To apply equation (5.19) to this BSDF, we must also know the density function $p_{\omega_{\rm o}}(\omega_{\rm i})$ that the reflected particles are sampled from. For a perfect mirror, the direction $\omega_{\rm i}$ is chosen deterministically, as represented by the density function

$$p_{\omega_{\rm o}}(\omega_{\rm i}) \;=\; \delta_\sigma(\omega_{\rm i}-M_{\mathbf{N}}(\omega_{\rm o}))\,.$$

Plugging these into equation (5.19), and noting that $\omega_{\rm i}\cdot\mathbf{N}_{\rm s}=\omega_{\rm o}\cdot\mathbf{N}_{\rm s}$, we get

$$\alpha(\omega_{\rm i}) \;=\; \frac{f_{{\rm s},\mathbf{N}_{\rm s}}(\omega_{\rm o}\!\rightarrow\!\omega_{\rm i})}{p_{\omega_{\rm o}}(\omega_{\rm i})}\,\frac{|\omega_{\rm o}\cdot\mathbf{N}_{\rm s}|}{|\omega_{\rm o}\cdot\mathbf{N}_{\rm g}|}\,|\omega_{\rm i}\cdot\mathbf{N}_{\rm g}|$$

$$
\begin{aligned}
&= \frac{|\omega_\mathrm{o} \cdot \mathbf{N}_\mathrm{s}|}{|\omega_\mathrm{o} \cdot \mathbf{N}_\mathrm{g}|} \frac{|\omega_\mathrm{i} \cdot \mathbf{N}_\mathrm{g}|}{|\omega_\mathrm{i} \cdot \mathbf{N}_\mathrm{s}|} \\
&= \frac{|\omega_\mathrm{i} \cdot \mathbf{N}_\mathrm{g}|}{|\omega_\mathrm{o} \cdot \mathbf{N}_\mathrm{g}|} .
\end{aligned}
$$

As a final example, recall the pool test case of Section 5.2.4. The waves in this scene were modeled using bump mapping, so that the air-water interface involves both refraction and shading normals. This means that both sets of results apply: radiance is scaled by a factor of $(\eta_\mathrm{o}/\eta_\mathrm{i})^2$ when it crosses the interface, while particle weights are scaled by

$$
\frac{|\omega_\mathrm{o} \cdot \mathbf{N}_\mathrm{s}|\, |\omega_\mathrm{i} \cdot \mathbf{N}_\mathrm{g}|}{|\omega_\mathrm{o} \cdot \mathbf{N}_\mathrm{g}|\, |\omega_\mathrm{i} \cdot \mathbf{N}_\mathrm{s}|} ,
$$

where as usual particles go from ω_o to ω_i (since this is a representation of the adjoint BSDF).

5.3.4 Pseudocode for the correct use of shading normals

In Figure 5.6, we give pseudocode to evaluate the factor

$$
K(\omega_\mathrm{i} \to \omega_\mathrm{o}) \;=\; \bar{f}_\mathrm{s}(\omega_\mathrm{i} \to \omega_\mathrm{o})\, |\omega_\mathrm{i} \cdot \mathbf{N}_\mathrm{g}| \tag{5.20}
$$

that appears in the scattering equation (5.9), and also the adjoint factor

$$
K^*(\omega_\mathrm{i} \to \omega_\mathrm{o}) \;=\; \bar{f}_\mathrm{s}^*(\omega_\mathrm{i} \to \omega_\mathrm{o})\, |\omega_\mathrm{i} \cdot \mathbf{N}_\mathrm{g}| \tag{5.21}
$$

that is used for importance evaluation and light particles. We will call these quantities the *scattering kernel* and the *adjoint scattering kernel* respectively. In Figure 5.6, the *adjoint* flag controls which of these is returned.

It may seem redundant to provide both the direct and adjoint kernels (because $\bar{f}_\mathrm{s}^*(\omega_\mathrm{i} \to \omega_\mathrm{o}) = \bar{f}_\mathrm{s}(\omega_\mathrm{o} \to \omega_\mathrm{i})$). However, this is actually quite useful. First, it allows higher-level rendering algorithms to always have the same form, whether they use radiance, importance, or particles. By supplying the appropriate *adjoint* flag, the BSDF is "transposed" appropriately for sampling or evaluation.[7] Second, it allows different density functions to be used for sampling in the direct and adjoint cases, as was shown to be necessary in Section 5.3.3. Given a

[7] In any case, the adjoints of specular BSDF's must always be represented explicitly, as mentioned in Section 5.2.5.

```
function EVAL-KERNEL($\omega_{\mathrm{i}} \to \omega_{\mathrm{o}}$, adjoint)
    assert $\mathbf{N}_{\mathrm{g}} \cdot \mathbf{N}_{\mathrm{s}} \geq 0$        (if not, flip $\mathbf{N}_{\mathrm{g}}$)
    if $(\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{g}})(\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{s}}) \leq 0$ or $(\omega_{\mathrm{o}} \cdot \mathbf{N}_{\mathrm{g}})(\omega_{\mathrm{o}} \cdot \mathbf{N}_{\mathrm{s}}) \leq 0$
        then return 0
    if adjoint
        then return $f_{\mathrm{s},\mathbf{N}_{\mathrm{s}}}(\omega_{\mathrm{o}} \to \omega_{\mathrm{i}})\, |\omega_{\mathrm{o}} \cdot \mathbf{N}_{\mathrm{s}}|\, |\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{g}}| \,/\, |\omega_{\mathrm{o}} \cdot \mathbf{N}_{\mathrm{g}}|$
        else  return $f_{\mathrm{s},\mathbf{N}_{\mathrm{s}}}(\omega_{\mathrm{i}} \to \omega_{\mathrm{o}})\, |\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{s}}|$
```

Figure 5.6: Evaluation of the scattering kernel K when the geometric and shading normals are different. The *adjoint* flag controls whether K or K^* is returned (these are used for ray tracing and particle tracing respectively). The return value includes the factor of $|\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{g}}|$ that is hidden by the projected solid angle notation.

direction ω_{o}, the ideal density function for ω_{i} is proportional to either K or K^* (depending on the *adjoint* flag). These density functions can be attached to the BSDF and returned as a single object during ray casting. Notice that for materials with symmetric BSDF's, we have $K = K^*$ and the *adjoint* flag can be ignored by the material implementation.

5.3.4.1 The prevention of "light leaks"

The pseudocode in Figure 5.6 also shows how to prevent light from "leaking" through the surface [Snyder & Barr 1987]. The problem is that an opaque surface can actually transmit light when shading normals are used. This happens when ω_{i} and ω_{o} lie geometrically on opposite sides of the surface, and yet they are on the same side of the surface according to the shading normal (see Figure 5.7(a)), so that the BSDF is evaluated as though light were "reflected" from one side of the surface to the other.

The simplest way to prevent this is to check that ω_{i} lies on the same side of the surface with respect to both normals, i.e. that $\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{g}}$ and $\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{s}}$ have the same sign. We also perform this test on ω_{o}, and if either test fails, we return a zero value for the BSDF (see Figure 5.6). This technique is effective in preventing the "light leaks" described above. However, it can also cause ordinary surfaces to appear completely black. This happens when the shading normal faces away from the viewing direction, i.e. when ω_{o} lies on opposite sides of the

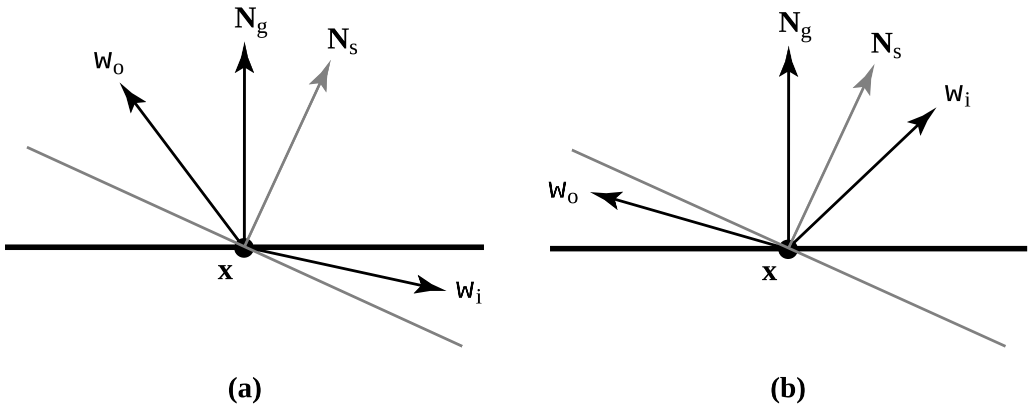


Figure 5.7: **(a)** The directions ω_i and ω_o are on opposite sides of the surface geometrically, yet they are on the same side with respect to the shading normal. Thus if we simply evaluate the BSDF using the shading normal, light can be "reflected" from one side of the surface to the other. The gives the visual impression that light is somehow leaking through the surface. **(b)** The easiest solution to this problem is check ifω_i lies on opposite sides of the surface with respect to $\mathbf{N}_g$ and $\mathbf{N}_s$. A similar test is performed on ω_o, and if either test fails, we return a zero value for the BSDF. However, this creates a new problem: if the test forω_o fails (as shown in the diagram), then the BSDF is zero for allω_i. This leads to sporadic "black patches" on the rendered surface.

surface with respect to N_g and N_s (see Figure 5.7(b)).

We now describe a way to solve both the light leak and black surface problems. To do this, we represent reflection and transmission by separate functions:

$$f_{r,\mathbf{N}} : \mathcal{S}^2 \to \mathcal{S}^2 \qquad \text{and} \qquad f_{t,\mathbf{N}} : \mathcal{S}^2 \to \mathcal{S}^2 \,.$$

Notice that both of these functions are defined for all $\omega_i, \omega_o \in \mathcal{S}^2$, i.e. they can be thought of as *extensions* of the BRDF and BTDF.

To evaluate the BSDF with shading normals, we proceed as follows (see Figure 5.8). If ω_i and ω_o lie on the same (geometric) side of the surface, then f_r is used, and otherwise f_t is used. Then, the chosen function f is evaluated with respect to the shading normal N_s (correcting for the adjoint if necessary).

With respect to this framework, f_r specifies how much light *would* be reflected between any two directions ω_i and ω_o, even if ω_i and ω_o lie on opposite sides of the surface. Similarly,

```
function EVAL-KERNEL-EXTENDED$(\omega_{\rm i} \to \omega_{\rm o}, \textit{adjoint})$
    if $(\omega_{\rm i} \cdot \mathbf{N}_{\rm g})(\omega_{\rm o} \cdot \mathbf{N}_{\rm g}) \;\le\; 0$
        then $f \leftarrow f_{\rm r}$
        else $f \leftarrow f_{\rm t}$
    if adjoint
        then return $f_{\mathbf{N}_{\rm s}}(\omega_{\rm o} \to \omega_{\rm i})\, |\omega_{\rm o} \cdot \mathbf{N}_{\rm s}|\, |\omega_{\rm i} \cdot \mathbf{N}_{\rm g}| \,/\, |\omega_{\rm o} \cdot \mathbf{N}_{\rm g}|$
        else return $f_{\mathbf{N}_{\rm s}}(\omega_{\rm i} \to \omega_{\rm o})\, |\omega_{\rm i} \cdot \mathbf{N}_{\rm s}|$
```

Figure 5.8: This pseudocode shows a different way to prevent light from leaking through a surface, by extending the BRDF and BTDF to be functions defined for all directions. We use the extended BRDF when $\omega_{\rm i}$ and $\omega_{\rm o}$ lie on the same side of the surface (with respect to the geometric normal), and otherwise we use the extended BTDF.

$f_{\rm t}$ is extended to describe transmission between directions on the same side of the surface. This extra information is used only when the shading and geometric normals give conflicting information.

For example, a diffuse surface would be represented by

$$f_{\rm r}(\omega_{\rm i} \to \omega_{\rm o}) \;=\; K_{\rm d} \qquad \text{and} \qquad f_{\rm t}(\omega_{\rm i} \to \omega_{\rm o}) \;=\; 0\,,$$

for all $\omega_{\rm i}, \omega_{\rm o} \in \mathcal{S}^2$. With these definitions, no light will leak through the surface in Figure 5.7(a), and yet the surface will not appear to be black in Figure 5.7(b).

Many other BRDF's, such as those based on microfacet theory, can naturally be extended to a function defined over all directions. Thus, this idea can be applied quite generally to solve the problem of light leaks. However, it is important to note that these are not the only artifacts associated with shading normals; see [Snyder & Barr 1987] for further examples.

5.3.5 Shading normals violate conservation of energy

In Section 6.3, we will show that any energy-conserving BSDF must satisfy

$$\int_{\mathcal{S}^2} f_{\rm s}(\omega_{\rm i} \to \omega_{\rm o})\, d\sigma^{\perp}(\omega_{\rm o}) \;\le\; 1 \qquad \text{for all } \omega_{\rm i}\,.$$

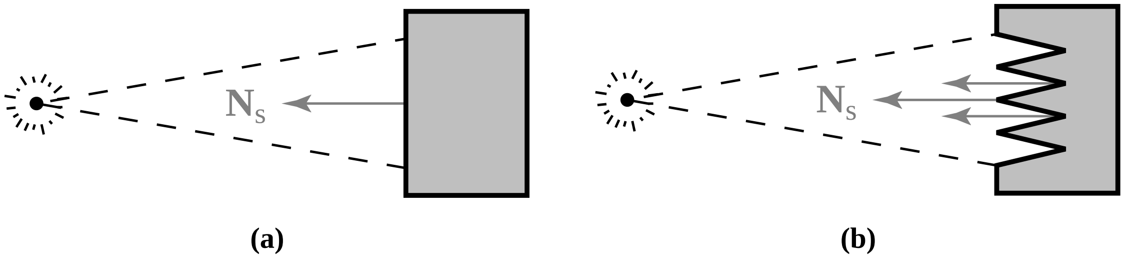


Figure 5.9: **(a)** A flat, diffuse surface facing toward a point light source, with$\mathbf{N}_\mathrm{s} = \mathbf{N}_\mathrm{g}$. The surface is assumed to not to absorb any light, so that the incident and reflected power is the same. **(b)** A ridged surface with shading normals that point toward the light. It receives the same power as (a), but reflects far more due to its larger surface area.

When shading normals are used, this condition applies to the modified BSDF $\bar{f}_\mathrm{s}$ defined by (5.17), leading to

$$\int_{\mathcal{S}^2} f_{\mathrm{s},\mathbf{N}_\mathrm{s}}(\omega_\mathrm{i} \to \omega_\mathrm{o}) \, \frac{|\omega_\mathrm{i} \cdot \mathbf{N}_\mathrm{s}|}{|\omega_\mathrm{i} \cdot \mathbf{N}_\mathrm{g}|} \, d\sigma^\perp(\omega_\mathrm{o}) \;\leq\; 1 \qquad \text{for all } \omega_\mathrm{i} \, .$$

However, if $\mathbf{N}_\mathrm{g}$ and $\mathbf{N}_\mathrm{s}$ are different, then the factor $|\omega_\mathrm{i} \cdot \mathbf{N}_\mathrm{s}| \,/\, |\omega_\mathrm{i} \cdot \mathbf{N}_\mathrm{g}|$ can be arbitrarily large (by choosing ω_i nearly perpendicular to $\mathbf{N}_\mathrm{g}$, but not $\mathbf{N}_\mathrm{s}$). Notice that this factor can be taken outside the integral, since it does not depend on ω_o. Thus energy is not conserved (for some values of ω_i).

For intuition about this, consider Figure 5.9(a), which shows a point light source shining on a flat, perfectly reflective, diffuse surface. To determine whether energy is conserved, we compare the power received by the surface to the power that is reflected. In Figure 5.9(a), these two quantities are equal.

In Figure 5.9(b), the surface is covered with steep ridges, but the shading normals point toward the light source as though the surface were flat. This surface receives the same total power as (a), since it occupies the same solid angle with respect to the light source. It also has the same apparent brightness as (a) at every point, because it has the same shading normal. In other words, the reflected radiance at every point and in every direction is the same in both cases, so that (b) reflects the same power per unit area as (a). However, the total surface area of (b) is much larger than (a). Thus, surface (b) reflects far more power than it receives.

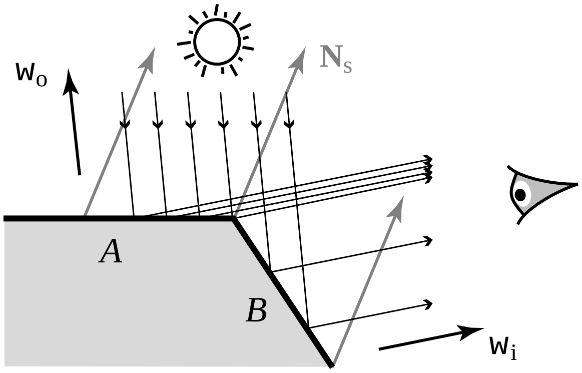


Figure 5.10: Light of uniform intensity (represented by equally spaced particles) arrives from direction ω_o at the polygonal surface shown. The shading normal $\mathbf{N}_s$ is continuous across the boundary between polygons A and B, implying that the shading should be continuous as well. However, suppose that an image is computed by estimating the apparent particle density from some other directionω_i. This density is discontinuous at the boundary, as shown; to get continuous shading, the particles must be weighted according to equation (5.19).

This lack of energy conservation has an important consequence for particle tracing algorithms: namely, that sometimes particle weights will increase during a scattering operation. This is especially important for algorithms that use unweighted particles (e.g. density estimation [Shirley et al. 1995]), since *splitting* of particles may be required. That is, rather than multiplying the current particle's weight by α, we replace it by $\lfloor\alpha\rfloor$ new particles, plus an extra particle with probability $\alpha - \lfloor\alpha\rfloor$.

5.3.6 Shading normals can cause brightness discontinuities

It is very important to use the adjoint BSDF $\bar{f}_s^*$ in particle tracing algorithms. Otherwise, there can be noticeable artifacts in the shading of polygonal meshes.

Consider Figure 5.10. Light of uniform intensity is arriving from direction ω_o at the polygonal surface shown. The shading normal $\mathbf{N}_s$ is continuous (in fact, constant) across the boundary between polygons A and B, implying that the shading of the mesh should appear smooth (no matter what rendering algorithm is used).

However, suppose that a particle tracing algorithm is used, and that an image is computed directly by making a dot for each particle collision at the corresponding point in the image. (This is an example of an *image space rendering algorithm,* as discussed in Section 1.4.3.) We let $\omega_{\rm o}$ denote the direction that particles arrive from, while $\omega_{\rm i}$ denotes the direction toward the viewer, following our convention that $\omega_{\rm i}$ is always the sampled direction in a random walk (Section 3.7.5).

We will show that the computed brightness of the polygons A and B is not the same, implying that there is a discontinuity at the boundary between A and B. To see this, observe that the brightness of the image regions corresponding to A and B is proportional to the number of dots per pixel made there. In turn, this is proportional to the apparent density of particles on A and B, measured perpendicular to the viewing direction $\omega_{\rm i}$.

To compute this apparent density, first note that the geometric normals of A and B are different, so that fewer particles per unit area are received by B than by A. It is easy to show that the particle densities on the polygons A and B are in the ratio

$$\frac{E_A}{E_B} \;=\; \frac{|\omega_{\rm o} \cdot \mathbf{N}_{\rm g}(A)|}{|\omega_{\rm o} \cdot \mathbf{N}_{\rm g}(B)|}\,.$$

(This is also the ratio of the irradiances on A and B.) From this, we can now compute the apparent particle density as seen from the viewpoint. This is simply the density of particles on the surface, divided by $|\omega_{\rm i} \cdot \mathbf{N}_{\rm g}|$. (Observe that if $|\omega_{\rm i} \cdot \mathbf{N}_{\rm g}|$ is small, then we are looking at the surface edge-on, and thus the particles will appear much more dense.)

Putting this all together, the image brightnesses of A and B are in the ratio

$$\frac{I_A}{I_B} \;=\; \frac{|\omega_{\rm o} \cdot \mathbf{N}_{\rm g}(A)|}{|\omega_{\rm o} \cdot \mathbf{N}_{\rm g}(B)|}\,\frac{|\omega_{\rm i} \cdot \mathbf{N}_{\rm g}(B)|}{|\omega_{\rm i} \cdot \mathbf{N}_{\rm g}(A)|}\,, \tag{5.22}$$

and so there is a discontinuity in the image brightness at the boundary between A and B.

Our original goal was that this boundary should appear smooth (since $\mathbf{N}_{\rm s}$ is continuous there). We will show that if the particles are weighted according to the adjoint BSDF $\bar{f}_{\rm s}^*$ (as they should be), this will be achieved. Referring to (5.19), the particles striking A are weighted by a factor of

$$\alpha_A \;=\; \frac{|\omega_{\rm i} \cdot \mathbf{N}_{\rm g}(A)|}{|\omega_{\rm o} \cdot \mathbf{N}_{\rm g}(A)|}\,,$$

where we have ignored weighting factors that are the same for particles on A and B. A similar weighting factor applies to particles striking B, and the ratio of these weights is

$$\frac{\alpha_A}{\alpha_B} \;=\; \frac{|\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{g}}(A)|}{|\omega_{\mathrm{o}} \cdot \mathbf{N}_{\mathrm{g}}(A)|} \, \frac{|\omega_{\mathrm{o}} \cdot \mathbf{N}_{\mathrm{g}}(B)|}{|\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{g}}(B)|} \, .$$

Comparing this with (5.22), we see that the change in particle weight exactly compensates for the change in particle density, resulting in smooth shading across the boundary.

Particle tracing techniques are not often used to compute direct illumination, as we have supposed here, because other techniques are usually more efficient. However, it is quite common that particle tracing is used to render at least *some* component of the lighting on visible surfaces; for example, particle tracing is often used to render caustics. If the adjoint BSDF is not used for these particles, there will be false discontinuities in the image as we have outlined above.

5.3.7 Results

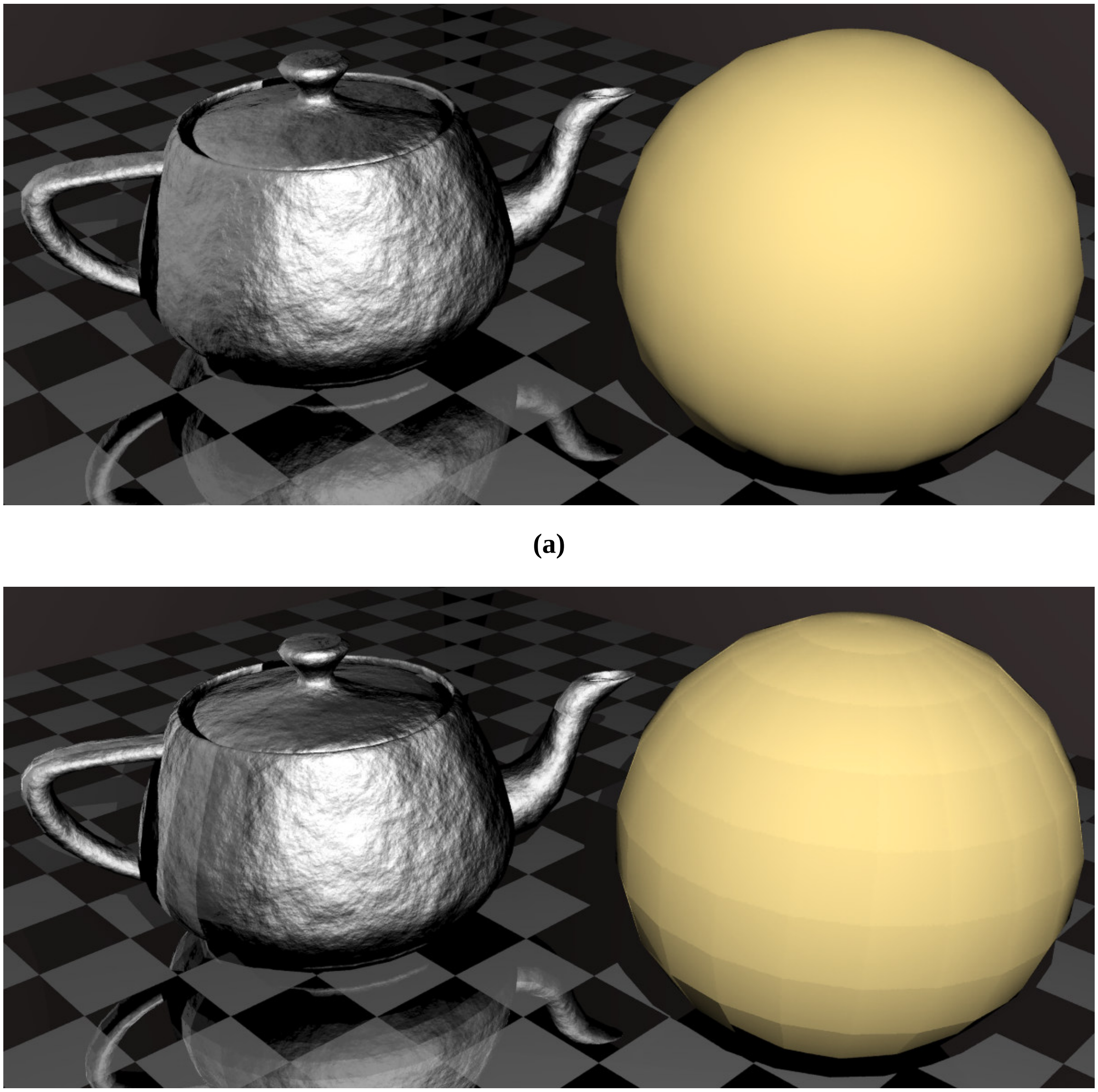
Figure 5.11 shows a bump-mapped teapot, and a polygonalized sphere with smooth shading normals. The images are simulations of a particle tracing algorithm: for each particle that strikes a surface, a dot is made at the corresponding point in the image, where the dot intensity is proportional to how much light is reflected toward the viewer. Figure 5.11(a) shows the correct result (using the adjoint BSDF), while Figure 5.11(b) shows what happens if particles are scattered just like viewing rays (i.e. if the non-symmetry caused by shading normals is not recognized). Both images use the same shading normals; the flat-shaded appearance of Figure 5.11(b) is an example of the shading artifacts described in Section 5.3.6.

5.3.8 Alternatives to shading normals

One way to avoid the problems associated with shading normals is to simply not use them. After all, they are not physically plausible. However, they are almost too useful to give up, both for approximating smooth surfaces with polygonal meshes, and for adding apparent surface detail without increasing geometric complexity.

At first, it might appear that some problems can be avoided by using the shading formula:

$$L_{\mathrm{o}}(\omega_{\mathrm{o}}) \;=\; \int_{\mathcal{S}^2} L_{\mathrm{i}}(\omega_{\mathrm{i}}) \, f_{\mathrm{s},\mathbf{N}_{\mathrm{s}}}(\omega_{\mathrm{i}} \!\rightarrow\! \omega_{\mathrm{o}}) \, |\omega_{\mathrm{i}} \cdot \mathbf{N}_{\mathrm{g}}| \, d\sigma(\omega_{\mathrm{i}}) \, , \qquad (5.23)$$



(a)

(b)

Figure 5.11: (a) Bump mapping and Phong interpolation, reference image. Shows direct lighting as it would be computed by particle tracing. **(b)** The same model, with errors caused by assuming that shading normals do not affect the symmetry of BSDF's.

where we have used $|\omega_{\rm i} \cdot \mathbf{N}_{\rm g}|$ instead of $|\omega_{\rm i} \cdot \mathbf{N}_{\rm s}|$. This method preserves the symmetry of $f_{\rm s}$ (assuming that it was symmetric in the first place), and will often conserve energy (but not always). However, the results obtained with this formula are not very useful. For example, consider a diffuse surface. The proposed shading formula (5.23) computes the *same surface appearance* for all values of $\mathbf{N}_{\rm s}$, because the BRDF $f_{\rm r}$ is a constant. Thus, it is impossible to make a polygonal surface look smooth, or make a flat surface look bumpy with this formula.

Similarly, consider a perfect mirror. Using (5.23) and the representation (5.10) of the mirror BSDF, we see that the radiance reflected by the mirror would be

$$L_{\rm o}(\omega_{\rm o}) \;=\; \frac{|M_{\mathbf{N}_{\rm s}}(\omega_{\rm o}) \cdot \mathbf{N}_{\rm g}|}{|\omega_{\rm o} \cdot \mathbf{N}_{\rm s}|}\, L_{\rm i}(M_{\mathbf{N}_{\rm s}}(\omega_{\rm o}))\,.$$

The weighting factor in this equation causes the reflectivity of the mirror to change with $\mathbf{N}_{\rm s}$, varying in the range $0 < \rho < 2$.[8] Again, this formula does not achieve what we would expect with shading normals, since it changes the reflectivity of the surface as well as the direction of reflection.

Other BSDF's produce similarly strange effects when used with (5.23), but they do not create the appearance of a changing surface orientation (as shading normals do). Thus the usefulness of (5.23) is quite limited. It seems far better to just use traditional shading normals, and accept the fact that their use corresponds to a non-symmetric BSDF.

Another possibility is to look for new BSDF models that serve the same purpose as shading normals, and yet are symmetric and energy-conserving. This is an interesting area for future research. Perhaps it could be accomplished with a microfacet shading model [Torrance & Sparrow 1967, Glassner 1995], where the distribution of microfacets is not symmetric about the surface normal.

However, it is important to realize that this kind of approach will never replace shading normals. One of the big advantages of shading normals is that they can be applied to *any* BSDF, while a microfacet approach would obviously be limited to a particular scattering model. Second, shading normals are designed to be as effective as possible at changing the apparent surface orientation. The results achieved using any symmetric, energy-conserving

[8] Because it is possible that $\rho > 1$, this is an example of a BSDF which was originally energy conserving, but not when formula (5.23) is used.

approach will necessarily be less convincing. Note that it is impossible to duplicate the surface appearance achieved by shading normals, since if two shading formulas *always* produce the same surface appearance, then they are represented by the same BSDF.

Appendix 5.A Dirac distributions for general measures

Our goal in this section is to give a rigorous definition of the notation $\delta_\mu(\mathbf{x} - \mathbf{x}_0)$ that was used to define the mirror BSDF (Section 5.2.1.2). This concept of a Dirac distribution with respect to a general measure is apparently new (although it seems quite basic), and we have found it to be a useful tool for many problems in graphics. We show how these distributions fit into the standard framework, and we also derive several identities that give them a consistent meaning as they are manipulated during calculations.

5.A.1 Linear functionals and distributions

Although the notation $\delta(x)$ looks like a function, it is properly called a *generalized function* or *distribution.* A rigorous theory of distributions was first developed by Laurent Schwartz in the late 1940's and early 1950's [Schwartz 1966]. However, physicists had been using similar ideas well before that; for example, Dirac introduced his famous "delta function" in 1925 [Lützen 1982].

To define distributions rigorously would take us too far afield, but we will at least summarize the basic concepts. Further information can be found in [Rudin 1973, Al-Gwaiz 1992].

First, the notation

$$\int_{\mathbb{R}} f(x)\, \delta(x - x_0)\, dx \tag{5.24}$$

should be thought of as purely symbolic (there is nothing being integrated in the traditional sense). Rather, this notation defines a mapping that takes a continuous function f, and yields a real number $f(x_0)$. The mapping is called a *linear functional*, and we will denote it by Λ_{x_0}.

Formally, a linear functional is a linear operator $\Lambda : X \to \mathcal{F}$ from a vector space X onto its scalar field $\mathcal{F}$ [Taylor & Lay 1980, p. 31]. In the example (5.24), the vector space is the set $C(\mathbb{R})$ of all continuous functions $f : \mathbb{R} \to \mathbb{R}$, with the usual operations of addition and scalar multiplication, and the scalars are simply the real numbers ($\mathcal{F} = \mathbb{R}$). The functional $\Lambda_{x_0} : C(\mathbb{R}) \to \mathbb{R}$ is defined by

$$\Lambda_{x_0}(f) \;=\; f(x_0)\,.$$

Thus, the notation (5.24) is just a long way of writing $\Lambda_{x_0}(f)$.

The mapping Λ_{x_0} is actually a special kind of functional called a *distribution.* Distributions have many desirable properties: they are infinitely differentiable, they obey the usual formal rules of calculus, they are equipped with many convergence theorems, and furthermore every continuous function is a distribution [Rudin 1973, p. 135]. To achieve these wonderful properties, however certain technical restrictions must be made (see Schwartz [1966] and Rudin [1973, p. 137,141]).

5.A.2 General Dirac distributions

Returning to our notation for general Dirac distributions, the expression

$$\int_\Omega f(\mathbf{x})\,\delta_\mu(\mathbf{x}-\mathbf{x}_0)\,d\mu(\mathbf{x}) \tag{5.25}$$

represents a distribution $\Lambda_{\mathbf{x}_0}$ acting on a function $f : \Omega \to \mathbb{R}$, where $\Lambda_{\mathbf{x}_0}$ is defined by[9]

$$\Lambda_{\mathbf{x}_0}(f) \;=\; f(\mathbf{x}_0)\,. \tag{5.26}$$

At this point, the notation $\delta_\mu(\mathbf{x}-\mathbf{x}_0)$ may seem rather odd, because the measure μ does not appear anywhere in the definition of $\Lambda_{\mathbf{x}_0}$. That is, given a different measure μ' on the domain Ω, the expression

$$\int_\Omega f(\mathbf{x})\,\delta_{\mu'}(\mathbf{x}-\mathbf{x}_0)\,d\mu'(\mathbf{x})$$

denotes exactly the same distribution $\Lambda_{\mathbf{x}_0}$ that we defined above. However, the point is that we have defined the meaning of the notation

$$\int_\Omega f(\mathbf{x})\,\delta_\mu(\mathbf{x}-\mathbf{x}_0)\,d\mu'(\mathbf{x}) \tag{5.27}$$

only when the measures μ and μ' are the same. The subscript on δ_μ is a reminder of this, since it is possible to get meaningless results if an expression such as (5.27) is evaluated carelessly. For example, if we take definition (5.8) of the mirror BSDF, and substitute it in the expanded version (5.9) of the scattering equation, we obtain

$$L_\mathrm{o}(\omega_\mathrm{o}) \;=\; \int_{\mathcal{S}^2} L_\mathrm{i}(\omega_\mathrm{i})\,\delta_{\sigma^\perp}(\omega_\mathrm{i}-M_\mathbf{N}(\omega_\mathrm{o}))\,|\omega_\mathrm{i}\cdot\mathbf{N}|\,d\sigma(\omega_\mathrm{i})\,. \tag{5.28}$$

If it were not for the subscript on δ, we might apply the identity (5.7) to obtain

$$L_\mathrm{o}(\omega_\mathrm{o}) \;=\; L_\mathrm{i}(M_\mathbf{N}(\omega_\mathrm{o}))\,|\omega_\mathrm{o}\cdot\mathbf{N}|\,,$$

which is incorrect.

[9] The fact that $\Lambda_{\mathbf{x}_0}$ is a distribution, and not merely a functional, is because the formula $\Lambda_\nu(f) = \int_\Omega f\,d\nu$ defines a distribution whenever ν is a σ-finite positive measure [Rudin 1973, p. 143]. In our case, the measure ν is defined by $\nu(D) = 1$ if $\mathbf{x}_0 \in D$, and $\nu(D) = 0$ otherwise.

5.A.3 Identities for evaluating general Dirac distributions

For future reference, we give several identities that are useful for evaluating expressions containing general Dirac distributions. We can summarize these as follows:

(D1) $\delta_\mu(\mathbf{x}_0 - \mathbf{x}) \;=\; \delta_\mu(\mathbf{x} - \mathbf{x}_0)$

(D2) $\delta_\mu(\mathbf{x} - \mathbf{x}_0) \;=\; \dfrac{d\mu'}{d\mu}(\mathbf{x}_0)\, \delta_{\mu'}(\mathbf{x} - \mathbf{x}_0)$

(D3) $\delta_\mu(\beta(\mathbf{x}) - \beta(\mathbf{x}_0)) \;=\; \delta_{\mu\circ\beta}(\mathbf{x} - \mathbf{x}_0)$

For property (D3), β denotes a bijective function $\beta : \Omega \to \Omega$, and $\mu \circ \beta$ is the *composition measure* defined by

$$(\mu \circ \beta)(D) \;=\; \mu(\beta(D))\,, \tag{5.29}$$

where $\beta(D) = \{\beta(\mathbf{x}) \mid \mathbf{x} \in D\}$. Property (D3) may look more familiar when it is specialized to the case of ordinary Dirac distributions on the real line, yielding

$$\delta(f(x) - f(x_0)) \;=\; \frac{1}{|f'(x_0)|}\, \delta(x - x_0)\,, \tag{5.30}$$

where f is a bijective function that is differentiable at x_0.

Note that all three of these properties are actually *definitions*, whose purpose is to extend the notation (5.25) in a consistent way. The definitions are designed to be compatible with the usual rules of calculus, so that we may formally apply them as though Dirac distributions were ordinary functions.

Property (D1). This defines the meaning of the notation $\delta_\mu(\mathbf{x}_0 - \mathbf{x})$. Note that the expressions $\mathbf{x} - \mathbf{x}_0$ and $\mathbf{x}_0 - \mathbf{x}$ are purely symbolic; they do not imply that subtraction is defined on the domain Ω.

Property (D2). This definition gives a consistent meaning to expressions of the form

$$\int_\Omega f(\mathbf{x})\, \delta_\mu(\mathbf{x} - \mathbf{x}_0)\, d\mu'(\mathbf{x})\,, \tag{5.31}$$

where the measures μ and μ' are different. One way to evaluate an expression of this kind is to change the integration measure, a concept similar to a change of variables. To do this, we require that μ and μ' are continuous with respect to each other, i.e. they have the same sets of measure zero. This guarantees the existence of the Radon-Nikodym derivative $d\mu/d\mu'$ (see Theorem 3.2), and allows us to switch from one integration measure to the other using

$$\int_\Omega f(\mathbf{x})\, d\mu(\mathbf{x}) \;=\; \int_\Omega f(\mathbf{x})\, \frac{d\mu}{d\mu'}(\mathbf{x})\, d\mu'(\mathbf{x}) \tag{5.32}$$

[Rudin 1987, p. 23]. For example, we could use this relationship to evaluate (5.28) correctly, by first substituting

$$|\omega_\mathrm{i} \cdot \mathbf{N}| \;=\; \frac{d\sigma^\perp(\omega_\mathrm{i})}{d\sigma(\omega_\mathrm{i})}\,,$$

and then applying (5.32) and (5.7) to get the right answer.

Definition (D2) allows us to evaluate (5.31) in another way, by changing the *distribution* rather than the integration measure. To obtain this identity, we rewrite (5.31) as

$$\begin{aligned} \int_\Omega f(\mathbf{x})\, \delta_\mu(\mathbf{x}-\mathbf{x}_0)\, d\mu'(\mathbf{x}) \;&=\; \int_\Omega f(\mathbf{x})\, \delta_\mu(\mathbf{x}-\mathbf{x}_0)\, \frac{d\mu'}{d\mu}(\mathbf{x})\, d\mu(\mathbf{x}) \\ &=\; \frac{d\mu'}{d\mu}(\mathbf{x}_0) \int_\Omega f(\mathbf{x})\, \delta_\mu(\mathbf{x}-\mathbf{x}_0)\, d\mu(\mathbf{x}) \\ &=\; \frac{d\mu'}{d\mu}(\mathbf{x}_0)\, f(\mathbf{x}_0)\,. \end{aligned} \tag{5.33}$$

We now simply observe that if the substitution (D2) is made on the left-hand side of (5.33), the same result is obtained. Thus, the definition (D2) is consistent. (Note that we have not previously defined the meaning of expressions such as (5.31).)

Property (D3). The definition gives a consistent meaning to the notation

$$\int_\Omega f(\mathbf{x})\, \delta_\mu(\beta(\mathbf{x}) - \beta(\mathbf{x}_0))\, d\mu(\mathbf{x})\,, \tag{5.34}$$

where β be a bijective function $\beta : \Omega \to \Omega$. Our goal is to define this in such a way that δ_μ can be treated as an ordinary function.

To do this, we make the definition

$$\delta_\mu(\beta(\mathbf{x}) - \beta(\mathbf{x}_0))\, \frac{d(\mu \circ \beta)}{d\mu}(\mathbf{x}_0) \;=\; \delta_\mu(\mathbf{x}-\mathbf{x}_0)\,, \tag{5.35}$$

where $\mu \circ \beta$ is the composition measure (5.29). We require that $\mu \circ \beta$ is continuous with respect to μ, so that the Radon-Nikodym derivative exists in (5.35). Notice that property (D3) can be obtained from (5.35) by applying property (D2).

To show that definition (5.35) is consistent, we evaluate

$$\begin{aligned}
&\int_\Omega f(\mathbf{x})\, \delta_\mu(\beta(\mathbf{x}) - \beta(\mathbf{x}_0))\, \frac{d(\mu \circ \beta)}{d\mu}(\mathbf{x}_0)\, d\mu(\mathbf{x}) \\
&\qquad = \int_\Omega f(\mathbf{x})\, \delta_\mu(\beta(\mathbf{x}) - \beta(\mathbf{x}_0))\, \frac{d\mu(\beta(\mathbf{x}))}{d\mu(\mathbf{x})}\, d\mu(\mathbf{x}) \\
&\qquad = \int_\Omega f(\mathbf{x})\, \delta_\mu(\beta(\mathbf{x}) - \beta(\mathbf{x}_0))\, d\mu(\beta(\mathbf{x})) \\
&\qquad = \int_\Omega f(\beta^{-1}(\mathbf{x}'))\, \delta_\mu(\mathbf{x}' - \beta(\mathbf{x}_0))\, d\mu(\mathbf{x}') \\
&\qquad = f(\beta^{-1}(\beta(\mathbf{x}_0))) \\
&\qquad = f(\mathbf{x}_0) \\
&\qquad = \int_\Omega f(\mathbf{x})\, \delta_\mu(\mathbf{x} - \mathbf{x}_0)\, d\mu(\mathbf{x})\,,
\end{aligned}$$

where we have defined $\mathbf{x}' = \beta(\mathbf{x})$ and used the fact the β is a bijection. Comparing the first and last lines, we get (5.35).

Another form of this identity is sometimes useful, when we are given an expression of the form $\delta_\mu(\beta(\mathbf{x}) - \mathbf{x}_0')$. By letting $\mathbf{x}_0 = \beta^{-1}(\mathbf{x}_0')$ in (5.35), we obtain

$$\delta_\mu(\beta(\mathbf{x}) - \mathbf{x}_0') \;=\; \left[\frac{d(\mu \circ \beta)}{d\mu}(\beta^{-1}(\mathbf{x}_0'))\right]^{-1} \delta_\mu(\mathbf{x} - \beta^{-1}(\mathbf{x}_0'))\,. \tag{5.36}$$

Appendix 5.B Derivation of the adjoint BSDF for refraction

We derive the adjoint BSDF for refraction (5.15). Unlike the derivation in Section 5.2.3, this one does not depend on physical laws.

Recall from Section 5.2.3 that refraction is described by the following BSDF:

$$f_{\rm s}(\omega_{\rm i} \to \omega_{\rm t}) \;=\; \frac{\eta_{\rm t}^2}{\eta_{\rm i}^2}\, \delta_{\sigma^\perp}(\omega_{\rm i} - R(\omega_{\rm t}))\,.$$

From the definition (3.21) of the adjoint BSDF, we can immediately write it as

$$\begin{aligned} f_{\rm s}^*(\omega_{\rm i} \to \omega_{\rm t}) &= f_{\rm s}(\omega_{\rm t} \to \omega_{\rm i}) \\ &= \frac{\eta_{\rm i}^2}{\eta_{\rm t}^2}\, \delta_{\sigma^\perp}(\omega_{\rm t} - R(\omega_{\rm i}))\,, \end{aligned}$$

where $\eta_{\rm i}$ and $\eta_{\rm t}$ have been exchanged because they are functions of$\omega_{\rm i}$ and $\omega_{\rm t}$ respectively.

This is a valid expression for the adjoint BSDF, but it is certainly not obvious that it is equivalent to the expression (5.15) given in Section 5.2.3. To show that it is, we first observe that although$R$ is a bijection, it does not preserve the measure$\sigma^\perp$, since from (5.4) we have

$$\frac{d\sigma^\perp(R(\omega_{\rm i}))}{d\sigma^\perp(\omega_{\rm i})} \;=\; \frac{d\sigma^\perp(\omega_{\rm t})}{d\sigma^\perp(\omega_{\rm i})} \;=\; \frac{\eta_{\rm i}^2}{\eta_{\rm t}^2}\,.$$

Thus, we can apply the identity (5.36) to get

$$\begin{aligned} f_{\rm s}^*(\omega_{\rm i} \to \omega_{\rm t}) &= \frac{\eta_{\rm i}^2}{\eta_{\rm t}^2}\, \delta_{\sigma^\perp}(R(\omega_{\rm i}) - \omega_{\rm t}) \\ &= \frac{\eta_{\rm i}^2}{\eta_{\rm t}^2} \left[\frac{d(\sigma^\perp \circ R)}{d\sigma^\perp}(R^{-1}(\omega_{\rm t})) \right]^{-1} \delta_{\sigma^\perp}(\omega_{\rm i} - R^{-1}(\omega_{\rm t})) \\ &= \frac{\eta_{\rm i}^2}{\eta_{\rm t}^2} \left[\frac{d\sigma^\perp(\omega_{\rm t})}{d\sigma^\perp(\omega_{\rm i})} \right]^{-1} \delta_{\sigma^\perp}(\omega_{\rm i} - R(\omega_{\rm t})) \\ &= \delta_{\sigma^\perp}(\omega_{\rm i} - R(\omega_{\rm t}))\,, \end{aligned}$$

which agrees with the expression (5.15) that we obtained before.

Appendix 5.C Angular forms of perfect specular BSDF's

Although we prefer to use general Dirac distributions to represent specular BSDF's, it may be helpful to see how reflection and refraction can be represented using ordinaryδ-functions. We will use the identities from Section 3.6.3 and Appendix 5.A.

5.C.1 The BSDF for a mirror

Starting with mirrors, recall that the desired relationship betweenL_{i} and L_{o} is

$$L_{\mathrm{o}}(\theta_{\mathrm{o}}, \phi_{\mathrm{o}}) \;=\; L_{\mathrm{i}}(\theta_{\mathrm{o}}, \phi_{\mathrm{o}} \pm \pi)\,.$$

The corresponding BSDF is thus

$$f_{\mathrm{s}}(\theta_{\mathrm{i}}, \phi_{\mathrm{i}}, \theta_{\mathrm{o}}, \phi_{\mathrm{o}}) \;=\; \frac{\delta(\theta_{\mathrm{i}} - \theta_{\mathrm{o}})\,\delta(\phi_{\mathrm{i}} - (\phi_{\mathrm{o}} \pm \pi))}{|\cos\theta_{\mathrm{i}}|\,\sin\theta_{\mathrm{i}}}\,.$$

To verify this, simply substitute f_{s} in the scattering equation (3.17).

By using different expressions (3.16) for the projected solid angle, we can also write the mirror BSDF as

$$\begin{aligned} f_{\mathrm{r}}(\theta_{\mathrm{i}}, \phi_{\mathrm{i}}, \theta_{\mathrm{o}}, \phi_{\mathrm{o}}) \;&=\; \frac{\delta(\cos\theta_{\mathrm{i}} - \cos\theta_{\mathrm{o}})\,\delta(\phi_{\mathrm{i}} - (\phi_{\mathrm{o}} \pm \pi))}{\cos\theta_{\mathrm{i}}} \\ &=\; 2\,\delta(\sin^2\theta_{\mathrm{i}} - \sin^2\theta_{\mathrm{o}})\,\delta(\phi_{\mathrm{i}} - (\phi_{\mathrm{o}} \pm \pi))\,, \end{aligned} \tag{5.37}$$

where the last expression is valid only for one-sided mirrors (since there are two solutions forθ_{i} in the range $0 \le \theta_{\mathrm{i}} \le \pi$). These forms of the mirror BRDF were given in [Nicodemus et al. 1977, p. 44] and [Cohen & Wallace 1993, p. 31].

5.C.2 The BSDF for refraction

The BSDF for refraction can also be written in terms of the angles(θ, ϕ). This form is given by

$$f_{\mathrm{s}}(\theta_{\mathrm{i}}, \phi_{\mathrm{i}}, \theta_{\mathrm{t}}, \phi_{\mathrm{t}}) \;=\; \frac{\eta_{\mathrm{t}}^2}{\eta_{\mathrm{i}}^2}\,2\,\delta(\sin^2\theta_{\mathrm{i}} - \frac{\eta_{\mathrm{t}}^2}{\eta_{\mathrm{i}}^2}\sin^2\theta_{\mathrm{t}})\,\delta(\phi_{\mathrm{i}} - (\phi_{\mathrm{t}} \pm \pi))\,. \tag{5.38}$$

(compare with (5.8)). Strictly speaking, this only represents the BTDF for light flowing in one direction (rather than the full BSDF), since equation (5.38) has two solutions forθ_{i} in the range $0 \le \theta_{\mathrm{i}} \le \pi$.

The adjoint BSDF for refraction is given by

$$\begin{aligned} f_{\mathrm{s}}^{*}(\theta_{\mathrm{i}}, \phi_{\mathrm{i}}, \theta_{\mathrm{t}}, \phi_{\mathrm{t}}) &= \frac{\eta_{\mathrm{i}}^2}{\eta_{\mathrm{t}}^2}\, 2\, \delta(\sin^2\theta_{\mathrm{t}} - \frac{\eta_{\mathrm{i}}^2}{\eta_{\mathrm{t}}^2} \sin^2\theta_{\mathrm{i}})\, \delta(\phi_{\mathrm{t}} - (\phi_{\mathrm{i}} \pm \pi)) \\ &= 2\, \delta(\sin^2\theta_{\mathrm{i}} - \frac{\eta_{\mathrm{t}}^2}{\eta_{\mathrm{i}}^2} \sin^2\theta_{\mathrm{t}})\, \delta(\phi_{\mathrm{i}} - (\phi_{\mathrm{t}} \pm \pi))\,, \end{aligned}$$

where we have used (5.30).

Chapter 6

Reciprocity and Conservation Laws for General BSDF's

In this chapter, we derive a new reciprocity principle that holds for materials that transmit as well as reflect light.[1] According to this principle, the BSDF of any physically valid material must satisfy

$$\frac{f_{\mathrm{s}}(\omega_{\mathrm{i}} \to \omega_{\mathrm{o}})}{\eta_{\mathrm{o}}^2} \;=\; \frac{f_{\mathrm{s}}(\omega_{\mathrm{o}} \to \omega_{\mathrm{i}})}{\eta_{\mathrm{i}}^2}\,, \tag{6.1}$$

where η_{i} and η_{o} are the refractive indices of the materials containing ω_{i} and ω_{o} respectively. This is a generalization of the well-known condition for reflective materials, which states that the corresponding BRDF must be symmetric:

$$f_{\mathrm{r}}(\omega_{\mathrm{i}} \to \omega_{\mathrm{o}}) \;=\; f_{\mathrm{r}}(\omega_{\mathrm{o}} \to \omega_{\mathrm{i}})\,.$$

We also investigate how light scattering is constrained by the law of conservation of energy, and we derive a simple condition that must be satisfied by any BSDF that is energy-conserving.

These conditions are important for two reasons. First, they provide a convenient test of the plausibility of BSDF models in computer graphics. Second, they provide a minimal set

[1] A *reciprocity principle* is a statement that expresses some form of symmetry in the laws governing a physical system. Such principles have been proposed throughout physics and chemistry, and are often stated as a pair of hypothetical experiments whose outcomes are supposed to be the same.

of facts that can be assumed by any physically valid rendering system. Along these lines, in Chapter 7 we use the reciprocity condition mentioned above to derive a framework for which light, importance, and particles all obey the same transport rules (for any physically valid scene). This can make many rendering algorithms significantly easier to implement.

The main goal of this chapter is to derive the new reciprocity principle (6.1). Given an arbitrary material, we analyze its scattering properties when it is placed within an *isothermal enclosure* (i.e. one where all objects have the same temperature, and no heat is lost to the external environment). For such a system, which is said to be in *thermodynamic equilibrium*, the exchange of light energy between various parts of the enclosure is highly constrained by the laws of thermodynamics. This allows us to derive the reciprocity condition (6.1) from only two basic principles, namely *Kirchhoff's equilibrium radiance law*, and the *principle of detailed balance*. Note that even though our analysis takes place within an isothermal enclosure, the resulting reciprocity principle is valid generally (since the BSDF of a material is an inherent property).

We also discuss the historical origins of reciprocity principles. One of the first physicists to study these ideas was Helmholtz, who proposed a famous principle concerning the propagation of light through an optical system. However, it is important to note that Helmholtz himself did not make any statement that would imply the symmetry of BRDF's. As we will see, his reciprocity principle only applies to reflection from mirrors (rather than arbitrary materials), and thus it does not have any direct implications for the symmetry of general BRDF's.

We also discuss the subtleties that arise in rigorously justifying such principles. For example, we explain why the symmetry of BRDF's cannot be derived directly from the second law of thermodynamics, or from the principle of *time reversal invariance*.

This chapter is organized as follows. Section 6.1 describes the second law of thermodynamics, Kirchhoff's laws, and the principle of detailed balance. Section 6.2 shows how these ideas can be put together in a "thought experiment" to prove the desired reciprocity condition (6.1). Section 6.3 derives a separate condition to ensure that BSDF's are energy-conserving.

In the appendices we examine the history of reciprocity principles, and also their limitations. Appendix 6.A describes the Helmholtz reciprocity principle, and explains why it does not have any implications regarding the symmetry of BRDF's. Appendix 6.B describes a different reciprocity principle due to Lord Rayleigh, who appears to be the first person to state a principle for reflection from arbitrary surfaces. Appendix 6.C considers the principle of time reversal invariance, and explains why observable light scattering processes are *irreversible* in general. Finally, Appendix 6.D investigates the limitations of these reciprocity principles, by describing two situations in which they fail: namely, in the presence of absorbing media or external magnetic fields.

6.1 Thermodynamics, Kirchhoff's laws, and detailed balance

Consider an enclosure containing various kinds of matter, which is completely insulated from its surrounding environment. Eventually, the contents will reach a uniform temperature, and the system is said to be in *thermodynamic equilibrium*. At equilibrium, each portion of matter will be emitting, scattering, and absorbing energy at various wavelengths (e.g. in the thermal, visible, and ultraviolet portions of the spectrum), in a manner that depends on both the local material properties and the surrounding radiation field. Thus, energy is constantly being exchanged among different regions of the enclosure, but in such a way that the temperature everywhere remains constant.

In this section, we explain two basic facts about systems in thermodynamic equilibrium, which will be used to derive the reciprocity condition (6.1). These facts are:

1. The radiance in an isothermal enclosure is *uniform*, i.e. it is the same for all positions and directions. More precisely, it depends only on the temperature of the enclosure and the local refractive index, such that

$$\frac{L_\nu(\mathbf{x}, \omega, \nu)}{\eta(\mathbf{x}, \omega, \nu)^2}$$

is constant throughout the enclosure (this is called *Kirchhoff's equilibrium radiance law*). Here $(\mathbf{x}, \omega)$ is a ray, ν is a frequency, $L_\nu(\mathbf{x}, \omega, \nu)$ is the spectral radiance for this ray and frequency, and $\eta(\mathbf{x}, \omega, \nu)$ is the refractive index of the medium that surrounds this ray. (The reason that η is a function of ω is to handle the case when $\mathbf{x}$ is on the boundary between two different media.)

2. For every process that transfers energy from one part of an isothermal system to another, there is a reverse process that transfers energy at the same rate in the opposite direction. This is known as the *principle of detailed balance*. For example, this principle states that for a system in thermodynamic equilibrium, the rates of emission and absorption for any given surface are equal.

We now explain these concepts in more detail. We first discuss the second law of thermodynamics, followed by Kirchhoff's equilibrium radiance law, and finally the principle of detailed balance. Our discussion of these ideas is based mainly on the excellent summary of [Milne 1930]; more detailed information can be found in Drude [1900], Siegel & Howell [1992], and [de Groot & Mazur 1962].

The second law of thermodynamics. Using the second law of thermodynamics, it is possible to derive important facts about the distribution of light energy in an isothermal enclosure. According to this principle, no ideal experiment can produce a temperature difference within the enclosure unless the experiment does work or modifies the external environment. For example, suppose that we divide the enclosure into two compartments separated by a surface S. Furthermore, suppose that S is transparent to light in a particular frequency band $[\nu_1, \nu_2]$, but reflects light at all other frequencies. Then by the second law, the rate of energy flow across this surface must be the same in both directions. Otherwise, the net flow would produce a temperature difference between the two sides of S, which could then be used to perform work.

The second law can be stated more precisely in terms of *entropy*. Entropy measures the amount of energy that can be transferred from one system to another, in the form of work. For a given system with a fixed energy, the entropy can range from zero to some maximum: if it is zero, then all of the energy in the system can be converted into work; while if it is at a maximum, then no work can be done at all. With respect to this concept, the second law states that the entropy of a closed, insulated system can never decrease, unless work is performed on it from some external source (see [de Groot & Mazur 1962, p. 20] for further details). Thus, once a system has reached thermodynamic equilibrium (the state of maximum entropy), it will remain in equilibrium, even if we perform ideal experiments such as adding barriers or mirrors, changing the locations of objects, etc.

Kirchhoff's laws. By proposing ideal experiments of this kind, Gustav Kirchoff was able to derive many interesting facts about the radiation in an isothermal enclosure [Milne 1930, p. 79], which are collectively known as *Kirchhoff's laws*.[2] Of these facts, we will need only his *equilibrium radiance law* mentioned above, which states that the quantity

$$\frac{L_\nu(\mathbf{x},\omega,\nu)}{\eta(\mathbf{x},\omega,\nu)^2} \tag{6.2}$$

is constant throughout the enclosure.[3] For example, if the objects in the enclosure are surrounded by a single medium, such as air, then this law states that the observed spectral radiance for all positions and directions will be the same. This is true even though the objects within the enclosure may have very different emission, scattering, and absorption properties. In fact, the observed spectral radiance depends only on temperature; given any two enclosures with different contents, but at the same temperature, the observed spectral radiance in these enclosures will be the same.

If the objects in the enclosure are surrounded by several different media, the spectral radiance will be proportional to η^2, as indicated by equation (6.2). This is one of the key facts that we will need to derive the reciprocity condition for general BSDF's.

Detailed balance. The other fact we need is the principle of detailed balance, which asserts that for a system in thermodynamic equilibrium, every detailed process that we choose to consider has a reverse process, and that the rates of these two processes are equal [van de Hulst 1980, p. 17]. For example, this principle asserts that in an isothermal enclosure, the

[2] In the heat transfer literature, *Kirchhoff's law* generally refers to one of these facts in particular, namely that the emissivity and absorptivity of real materials are the same [Siegel & Howell 1992, p. 66]. This was derived by Kirchhoff as a consequence of his equilibrium radiance law. Note that these results are not related to Kirchhoff's laws for electric circuits, which he proposed much earlier in 1845. Also note that the invariance of L/η^2 is often falsely attributed to Clausius (cf. Drude [1900, p.504]).

[3] Strictly speaking, this law is true only when some material in the enclosure is capable of emitting or absorbing radiation at the given frequency ν [Milne 1930, p. 80]. We can ensure that this is always true by assuming that the enclosure contains a *black body* (which absorbs and emits radiation at all frequencies).

emission and absorption rates of every surface are equal. This principle also applies to scattering, as we will see in the next section.

Detailed balance has been formulated and proven for any system that possesses time reversal invariance, both in classical systems and in quantum mechanics [de Groot & Mazur 1962], [van Kampen 1954], [Wigner 1954]. *Time reversal invariance* is one of the basic principles of physics, which states if the time variable is negated in all formulas and equations, then the laws of physics at their most microscopic level are unchanged (see Appendix 6.C). The only significant restriction of detailed balance is that for it to be valid, there must not be any external magnetic fields [de Groot 1963].

6.2 A reciprocity principle for general BSDF's

By combining Kirchhoff's equilibrium radiance law with the principle of detailed balance, we derive a reciprocity principle that holds for arbitrary physically valid materials. We also show that this principle cannot be derived from the second law alone.

Consider a small area $dA(\mathbf{x})$ within an isothermal enclosure (see Figure 6.1). We assume that $\mathbf{x}$ lies either on an opaque surface, or on the boundary between two non-absorbing media. We also assume that no external magnetic fields are present, so that the principle of detailed balance applies.

Consider the light that arrives from a small cone of directions $d\sigma(\omega_\mathrm{i})$, and is scattered toward another cone $d\sigma(\omega_\mathrm{o})$, where ω_i and ω_o can lie on either side of the surface. The scattering can be of any type: reflection or transmission, specular or non-specular. According to the definition of the BSDF (3.11), the power scattered from ω_i to ω_o is

$$\begin{aligned} d\Phi_1 &= L_\mathrm{o}(\omega_\mathrm{o})\, dA(\mathbf{x})\, d\sigma^\perp(\omega_\mathrm{o}) \\ &= L_\mathrm{i}(\omega_\mathrm{i})\, d\sigma^\perp(\omega_\mathrm{i})\, f_\mathrm{s}(\omega_\mathrm{i} \!\to\! \omega_\mathrm{o})\, dA(\mathbf{x})\, d\sigma^\perp(\omega_\mathrm{o})\,. \end{aligned}$$

On the other hand, the power scattered from ω_o to ω_i is

$$d\Phi_2 = L_\mathrm{i}(\omega_\mathrm{o})\, d\sigma^\perp(\omega_\mathrm{o})\, f_\mathrm{s}(\omega_\mathrm{o} \!\to\! \omega_\mathrm{i})\, dA(\mathbf{x})\, d\sigma^\perp(\omega_\mathrm{i})\,.$$

By the principle of detailed balance, the rates of scattering in these two directions are equal.

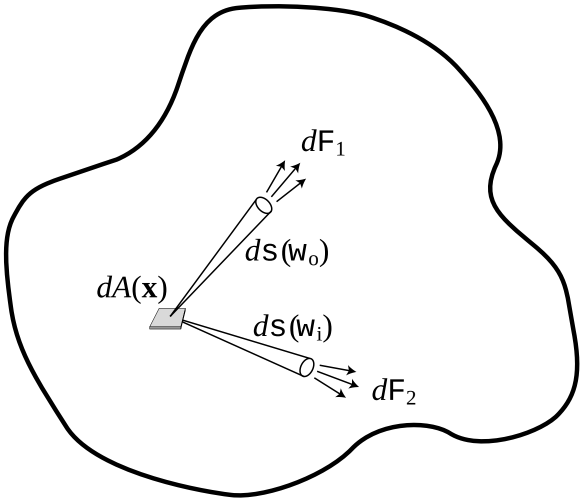


Figure 6.1: To prove a reciprocity condition for general BSDF's, we consider the light energy scattered between two directionsω_i and ω_o at a point $\mathbf{x}$ in an isothermal enclosure. By the principle of detailed balance, the rates of scattering fromω_i to ω_o and from ω_o to ω_i are equal ($d\Phi_1 = d\Phi_2$), while by Kirchhoff's equilibrium radiance law, the incident radiance from each direction is proportional to the refractive index squared. Putting these facts together, we get the desired reciprocity condition (6.1).

Thus we have $d\Phi_1 = d\Phi_2$, so that

$$L_\mathrm{i}(\omega_\mathrm{i})\, f_\mathrm{s}(\omega_\mathrm{i} \!\rightarrow\! \omega_\mathrm{o}) \;=\; L_\mathrm{i}(\omega_\mathrm{o})\, f_\mathrm{s}(\omega_\mathrm{o} \!\rightarrow\! \omega_\mathrm{i})\,.$$

Next, we consider the incident radiance values, $L_\mathrm{i}(\omega_\mathrm{i})$ and $L_\mathrm{i}(\omega_\mathrm{o})$. According to Kirchhoff's equilibrium radiance law, L_i/η^2 is constant throughout the enclosure, so that

$$\frac{L_\mathrm{i}(\omega_\mathrm{i})}{\eta_\mathrm{i}^2} \;=\; \frac{L_\mathrm{i}(\omega_\mathrm{o})}{\eta_\mathrm{o}^2}\,.$$

Putting these two facts together, we get the following result for physically valid BSDF's:

Theorem 6.1. *Let f_s be the BSDF for a physically valid surface, which is either the boundary of an opaque object or the interface between two non-absorbing media. Provided that there are no external magnetic fields, then*

$$\frac{f_\mathrm{s}(\omega_\mathrm{i} \!\rightarrow\! \omega_\mathrm{o})}{\eta_\mathrm{o}^2} \;=\; \frac{f_\mathrm{s}(\omega_\mathrm{o} \!\rightarrow\! \omega_\mathrm{i})}{\eta_\mathrm{i}^2}\,, \tag{6.3}$$

where $\eta_{\mathrm{o}} = \eta(\omega_{\mathrm{o}})$ is a function of ω_{o}, and similarly for η_{i}. ■

This condition is clearly a generalization of the usual symmetry condition for BRDF's. The most significant change concerns BSDF's that describe the interface between two different refractive media. For this case, the ratio of f_{s} to f_{s}^* is $(\eta_{\mathrm{o}}/\eta_{\mathrm{i}})^2$, so that radiance and importance are scaled differently when they are transmitted through the interface. This law is not limited to perfect specular refraction; it also includes diffusely transmitting materials, such as frosted glass.

Note that although this relationship was derived in an isothermal enclosure, it is valid in general. The BSDF is an inherent property of the surface, and does not change simply because the surrounding environment is isothermal. Also note that the enclosure does not have to be at a high temperature for this argument to hold, since even at ordinary temperatures, there is a small amount of thermal radiation in the visible wavelengths.

Insufficiency of the second law. Returning to the simpler case of opaque materials (BRDF's), it is sometimes claimed that the reciprocity condition for these materials can be derived directly from the second law of thermodynamics.[4] We show that this is false, by giving an example of a BRDF which is not symmetric, but where this lack of symmetry cannot be detected by any ideal experiment in an isothermal enclosure.

We consider a hypothetical surface that is similar to a mirror. For an ordinary mirror, light is reflected from the incident direction ω_{i} to the mirror direction ω_{o}, where the mirror direction is obtained by rotating ω_{i} by 180 degrees about the surface normal. We consider a new BRDF that modifies this rule: the mirror vector is obtained by rotating the incident vector by only 90 degrees about the surface normal, in a clockwise direction. Clearly, this new BRDF is not symmetric.

However, the new BRDF and the original mirror BRDF are *indistinguishable* in an isothermal enclosure. The reason is that the incident radiance is guaranteed to be uniform, and both of these BRDF's will map a uniform incident radiance function into a uniform exitant radiance function. Thus, there is no ideal experiment in an isothermal enclosure that

[4] For example, the BRDF reciprocity argument of Siegel & Howell [1992, p. 73] appears to depend only on the second law. However, their argument is flawed. To fix it, they require the principle of detailed balance; in which case their proof could be simplified by neglecting the transport path labeled $dA_1\, dA_3$ in their figure.

can distinguish between these two situations. This is why the principle of detailed balance is necessary; it provides more detailed information about how the incident and exitant radiance functions are related, by considering the energies traveling in opposite directions along the same path.

Reciprocity for spectral radiance. To be precise, the reciprocity condition (6.3) applies to spectral radiance (rather than radiance), and in fact it applies only to spectral radiance that is measured with respect to frequency (L_ν). When spectral radiance is measured with respect to wavelength (L_λ), this condition must be modified, since light undergoes a change in wavelength when it is transmitted into a different medium. In particular, the incident and transmitted wavelengths are related by

$$\lambda_{\mathrm{t}} \;=\; (\eta_{\mathrm{i}}/\eta_{\mathrm{t}})\,\lambda_{\mathrm{i}}\,,$$

i.e. the wavelength is smaller in media with a higher refractive index. Notice that according to this equation, the product $\lambda\eta = \lambda_0$ is constant across the interface, where λ_0 is the wavelength of light in a vacuum.

For spectral radiance with respect to wavelength, Kirchhoff's equilibrium radiance law now states that the quantity

$$\frac{L_\lambda(\mathbf{x}, \omega, \lambda_0/\eta)}{\eta(\mathbf{x}, \omega, \lambda_0)^3}$$

is constant throughout the enclosure. This equation applies separately at each wavelength λ_0. The factor of η^3 instead of η^2 occurs because L_λ is defined as a derivative with respect to wavelength (see [Nicodemus 1976, p. 51]). Effectively, when light enters a medium of higher refractive index, the same light energy is squeezed into a smaller band of wavelengths, which causes the spectral radiance to increase proportionately.

Applying this version of Kirchhoff's equilibrium radiance law, the reciprocity condition for BSDF's becomes

$$\frac{f_{\mathrm{s},\lambda}(\omega_{\mathrm{i}} \to \omega_{\mathrm{o}}, \lambda_0/\eta_{\mathrm{i}})}{\eta_{\mathrm{o}}(\lambda_0)^3} \;=\; \frac{f_{\mathrm{s},\lambda}(\omega_{\mathrm{o}} \to \omega_{\mathrm{i}}, \lambda_0/\eta_{\mathrm{o}})}{\eta_{\mathrm{i}}(\lambda_0)^3}\,, \qquad (6.4)$$

where the wavelength parameter of $f_{\mathrm{s},\lambda}(\omega_{\mathrm{i}} \to \omega_{\mathrm{o}}, \lambda)$ refers to the incident light, and λ_0 is the wavelength in a vacuum.

6.3 Conservation of energy

We show that the following condition is implied by conservation of energy:

Theorem 6.2. *If $f_{\rm s}$ is the BSDF for a physically valid surface, which is either the boundary of an opaque object or the interface between two non-absorbing media, then*

$$\int_{\mathcal{S}^2} f_{\rm s}(\omega_{\rm i} \to \omega_{\rm o}) \, d\sigma^\perp(\omega_{\rm o}) \;\le\; 1 \qquad \textit{for all} \quad \omega_{\rm i} \in \mathcal{S}^2 \,. \tag{6.5}$$

This is very similar to the energy-conservation condition for BRDF's, which was mentioned in Section 3.6.2.

Proof. Equation (6.5) can be proven from the relations

$$\begin{aligned} E &= \int_{\mathcal{S}^2} L_{\rm i}(\omega) \, d\sigma^\perp(\omega) \\ L_{\rm o}(\omega_{\rm o}) &= \int_{\mathcal{S}^2} L_{\rm i}(\omega) \, f_{\rm s}(\omega \to \omega_{\rm o}) \, d\sigma^\perp(\omega) \\ M &= \int_{\mathcal{S}^2} L_{\rm o}(\omega) \, d\sigma^\perp(\omega) \,, \end{aligned}$$

where E denotes the irradiance (i.e. the incident power per unit area), and M denotes the radiant exitance (i.e. the scattered power per unit area, see Section 3.4). By conservation of energy, we require that $M \le E$ for all possible incident radiance functions $L_{\rm i}$; that is, the surface should never scatter more light than it receives.

To obtain the desired condition (6.5), we fix a particular direction $\omega_{\rm i}$, and consider the incident radiance distribution $L_{\rm i}(\omega) = \delta_{\sigma^\perp}(\omega - \omega_{\rm i})$, i.e. we let the incident power be concentrated in a single direction $\omega_{\rm i}$.[5] With this choice of $L_{\rm i}$, we obtain

$$\begin{aligned} E &= \textstyle\int_{\mathcal{S}^2} L_{\rm i}(\omega) \, d\sigma^\perp(\omega) &&= 1 \\ L_{\rm o}(\omega_{\rm o}) &= \textstyle\int_{\mathcal{S}^2} L_{\rm i}(\omega) \, f_{\rm s}(\omega \to \omega_{\rm o}) \, d\sigma^\perp(\omega) &&= f_{\rm s}(\omega_{\rm i} \to \omega_{\rm o}) \\ M &= \textstyle\int_{\mathcal{S}^2} L_{\rm o}(\omega) \, d\sigma^\perp(\omega) &&= \textstyle\int_{\mathcal{S}^2} f_{\rm s}(\omega_{\rm i} \to \omega_{\rm o}) \, d\sigma^\perp(\omega_{\rm o}) \,, \end{aligned}$$

from which the requirement that $M \le E$ gives the desired result. ∎

[5] Alternatively, we could use a sequence of radiance functions that approximate $L_{\rm i}$, to avoid the issue of whether $L_{\rm i}$ is allowed to be a Dirac distribution.

Appendix 6.A Helmholtz reciprocity

In the graphics and radiometry literature, the symmetry of BRDF's is often attributed to the Helmholtz reciprocity principle. Apparently this notion first arose in the radiometry literature (see the references in Section 6.A.3), and migrated to graphics through the work of Nicodemus (e.g. see [Nicodemus et al. 1977, p. 40], [Nicodemus 1965, p. 769]).

In this section, we examine the original statement of Helmholtz reciprocity, and show that it does not imply the symmetry of BRDF's. Helmholtz stated his principle only for classical optical systems (consisting of mirrors and lenses), and thus with regard to the reflection of light from surfaces, his principle applies only to mirrors. He does not mention non-specular reflection of any sort (e.g. diffuse or glossy surfaces). (Of course, we would not expect Helmholtz to mention BRDF's in any case, since the concept of a BRDF was not invented at that time.)

6.A.1 Summary of the principle

The Helmholtz reciprocity principle is found in his famous treatise on physiological optics, first published in 1856 [von Helmholtz 1856, p. 231]. This three-volume work concerns human vision: the anatomy of the eye, the mechanisms of sensation, and the interpretation of those sensations. With regard to optics, Helmholtz' main concern was to analyze the properties of the eye within the framework of classical geometric optics.

In this context, Helmholtz proposed the following reciprocity principle for beams traveling through an optical system (i.e. a collection of mirrors, lenses, prisms, etc). Suppose that a beam of light A undergoes any number of reflections or refractions, eventually giving rise (among others) to a beam B whose power is a fraction f of beam A. Then on reversing the path of the light, an incident ray B' will give rise (among others) to a beam A' whose power is the same fraction f of beam B'.[6] In other words, the path of a light beam is always reversible, and furthermore the relative power loss is the same for propagation in both directions.

The main point is that the only type of reflection considered by Helmholtz is specular reflection from mirrors. Thus, his principle does not have any direct implications for general BRDF's (or BSDF's).

Note that Helmholtz reciprocity can easily be extended to materials that are composed of many

[6] Our paraphrasing follows that of Chandrasekhar [1960, p. 176].

small mirrors. By considering the limit as these mirrors become very small, a variety of interesting materials can be obtained (this is the basic idea behind *microfacet reflection models* [Torrance & Sparrow 1967, Cook & Torrance 1982]). However, note that this approach is not adequate to prove a reciprocity principle for real materials, since it only applies to a particular *model* for reflection from surfaces. That is, real surfaces are not necessarily composed of microfacets, so this type of argument cannot be used to make statements about the properties of real BRDF's. (Microfacet models were only proposed as a model to explain reflection from metals, and in any case the microfacets are generally so small that geometric optics is not applicable: diffraction theory must be used instead [He et al. 1991].)

6.A.2 Helmholtz' original statement

We now examine the original statement of Helmholtz reciprocity[7], to explain in more detail why it applies only to specular reflection. Note that the following quotation is for polarized light, which makes it slightly more complicated. With respect to the paraphrasing of the previous section, it assumes that beam A is polarized in a given plane P_A, and that we only measure the component of beam B that is polarized in a given plane P_B. Helmholtz' principle then states that the same fraction of power is lost for a beam traveling in either direction:

> Suppose light proceeds by any path whatever from a point A to another point B, undergoing any number of reflections or refractions *en route.* Consider a pair of rectangular planes a_1 and a_2 whose line of intersection is along the initial path of the ray at A; and another pair of rectangular planes b_1 and b_2 intersecting along the path of the ray when it comes to B. The components of the vibrations of the aether particles in these two pairs of planes may be imagined. Now suppose that a certain amount of light J leaving the point A in the given direction is polarised in the plane a_1, and that of this light the amount K arrives at the point B polarised in the plane b_1; then it can be proved that, when the light returns over the same path, and the quantity of light J polarised in the plane b_1 proceeds from the point B, the amount of this light that arrives at the point A polarised in the plane a_1 will be equal to K.
>
> Apparently the above proposition is true no matter what happens to the light in the way of single or double refraction, reflection, absorption, ordinary dispersion, and diffraction, provided that there is no change of its refrangibility, and provided it does not traverse any magnetic medium that affects the position of the plane of polarisation, as Faraday found to be the case.

[7] Translated from the German [von Helmholtz 1856, p.231]. A somewhat shorter statement of this principle appears in [von Helmholtz 1903, Section 42, p. 158], but the apparent meaning is the same.

It is absolutely clear that Helmholtz did not intend this principle to be applied to diffuse reflection. In the cited reference, he consistently uses the word*reflection* to mean only specular (mirror-like) reflection. For example, in the discussion leading up to the statement of the principle above [von Helmholtz 1856, p. 230], Helmholtz states and proves a similar theorem where it is obvious that only specular reflection is considered. The very phrase*reflections and refractions* implies specularity, since otherwise the theorem would be stated in terms of reflection and*transmission*.

Also, the principle refers to the*amounts* of light leaving A and arriving at B. In the terminology of the day, an "amount" of light referred to total flux or power.[8] Thus, the principle does not even make sense for diffuse reflection: the power arriving on beam$B$ would always be zero, since only an infinitesimal quantity of power is reflected by a diffuse surface in any particular direction. In order to make sense for non-specular reflection, the law would need to relate the power at$A$ to a *differential* quantity at B, such as irradiance. (This is exactly what was done by Lord Rayleigh, in the reciprocity principle discussed below.)

Another important fact is that Helmholtz reciprocity is not always valid, as we will discuss in Appendix 6.D). Interestingly, Helmholtz did not provide a proof of his principle, claiming that "anybody who is at all familiar with the laws of optics can easily prove it for himself" [von Helmholtz 1856, p. 231].

6.A.3 Further reading on Helmholtz reciprocity

This section gives a sampling of the various sources of information available concerning reciprocity principles. It is by no means exhaustive.

First, there are references that interpret the Helmholtz reciprocity principle correctly, in the limited sense discussed above (beams propagating through an optical system, rather than general scattering from surfaces). These include Planck [1914, p. 49], von Fragstein [1955], Chandrasekhar [1960, p. 176], and Born & Wolf [1986, p. 381].

Second, there are sources in the radiometry literature that claim (in passing) that Helmholtz reciprocity implies the symmetry of physically valid BRDF's. These include McNicholas [1928], de Vos [1954], de la Perrelle et al. [1963], Nicodemus [1965, p. 769], and Nicodemus et al. [1977, p. 40].

[8] It is important that Helmholtz stated his law in terms of power, rather than radiance, since this way his law is valid even when A and B lie in media with different refractive indices. If it were stated in terms of radiance (which Helmholtz calls "brightness"), there would need to be an η^2 scaling factor as discussed in Section 5.2. Helmholtz was aware of this scaling factor [von Helmholtz 1856, p. 233], and thus phrased his law to make it as general as possible.

Another worthwhile reference is Minnaert [1941], who recognized that the original statement of Helmholtz reciprocity does not apply to general scattering, but shows how the statement can be reinterpreted to have greater generality.

Third, there are general reciprocity principles in the physics literature regarding the scattering of electromagnetic waves. These include Kerr [1987], Saxon [1955], and de Hoop [1960]. These principles are unrelated to Helmholtz', and require additional physical assumptions for their validity.

This brings up an important point, which is that many reciprocity principles in optics are derived by *starting* with a relationship of the sort that we want to prove (i.e. an assumption that is equivalent to the symmetry of BRDF's). For example, several reciprocity principles for volume scattering are proven in [Case 1957], by assuming that the phase function is symmetric (bottom p. 653). By making additional assumptions of this sort (e.g. that all scattering particles have random orientations), it is possible to derive a wide variety of reciprocity principles in optics [van de Hulst 1957, Chapter 5], [Hovenier 1969], [van de Hulst 1980, Chapter 3]. It would be an easy mistake to derive a reciprocity principle for BSDF's by starting with results such as these.

Appendix 6.B Lord Rayleigh's reciprocity principle

In 1900, Lord Rayleigh stated a reciprocity principle for non-specular reflection, and apparently he was the first to do so.[9] Essentially, this principle asserts that real materials have symmetric BRDF's. The original statement is as follows:

> Suppose that in any direction (i) and at any distance r from a small surface (S) reflecting in *any manner* there be situated a radiant point (A) of given intensity, and consider the intensity of the reflected vibrations at any point B situated in direction ϵ and at distance r' from S. The theorem is to the effect that the intensity is the same as it would be at A if the radiant point were transferred to B. [*Footnote:* I have not thought it necessary to enter into questions connected with polarization, but a more particular statement could easily be made.]

Translated into modern terminology, we are given a small reflective surface, exposed to a small light source and a small *irradiance sensor* (which measures the power per unit area falling on a square facing toward the reflective surface). His principle states that if the positions of the source and sensor are exchanged, the measured irradiance will be the same. This implies that the corresponding BRDF must be symmetric, as may easily be verified.

Observe that Rayleigh's principle is merely a statement of fact; no proof was given in terms of more basic physical laws. Although it was claimed as a consequence of "a fundamental principle of reciprocity, of such generality that escape from it is difficult" (to be found in his *Theory of Sound* [Rayleigh 1877, Sec. 109, p. 154]), the methods used there are not rigorous by modern standards, and are not explicitly related to light. Furthermore, they require symmetry assumptions about the underlying system (e.g. see [Rayleigh 1877, Sec. 103a, p. 139]) that seem no more justifiable than assuming the symmetry of the BRDF in the first place.

[9] This observation was made by Chandrasekhar [Chandrasekhar 1960, p. 177]. Rayleigh's statement of reciprocity can be found in a short letter to the *Philosophical Magazine* [Rayleigh 1900, p. 324] (reprinted in [Rayleigh 1964, p. 480]).

Appendix 6.C Time reversal invariance and the irreversibility of light scattering

The symmetry of BRDF's is sometimes attributed to a physical law known as*time reversal invariance*. In this section, we explain why such claims are incorrect. Time reversal invariance does not have any direct consequences for the symmetry of BRDF's, because most observable light scattering processes are *irreversible*.

We first explain the principle of time reversal invariance, which applies the laws of physics at a microscopic level (e.g. interactions between individual particles). Next, we explain why observable processes are usually *irreversible*, even though they are governed by microscopically reversible laws. Finally, we explain how this applies to light scattering: we show that observable light scattering processes are almost always irreversible, so that time reversal invariance does not have any direct implications for the symmetry of BRDF's.

Time reversal. It is known that the fundamental laws of physics are invariant under the operation of *time reversal*, in which the time variable is negated in formulas and equations. More precisely, this principle should be called*motion reversal invariance*, since it asserts that if the motions of all particles and waves in a system are reversed, then they will retrace their former paths [de Groot & Mazur 1962, p. 35]. This principle holds in any physical system, as long as there are no external magnetic fields; otherwise, the direction of the field must be reversed along with the wave and particle motions, in order for time reversal invariance to hold ([de Groot & Mazur 1962, p. 38], [de Groot 1963]).[10]

This principle can be stated more precisely as follows. Let$A$ and B be any two *microscopic states* of the given system, where each state completely specifies the attributes of all particles and waves. We let $p(A, B, t)$ denote the *transition function* for this system, i.e. the probability density that if the system is in state A, it will evolve to state B over a time interval of length t. (Note that according to quantum mechanics, the universe is not deterministic; thus, we can only compute the *probability* with which the system evolves from state to state.) Finally, given a state$X$, we let $-X$ denote the state obtained by motion reversal of all particles and waves (including the reversal of magnetic fields, if necessary). Given these definitions, the principle of time reversal invariance then

[10] Technically, there are some known exceptions to time reversal invariance, however these involve nuclear interactions and are not significant for optics [*Brittanica Online* 1996].

states that

$$p(A, B, t) \;=\; p(-B, -A, t) \qquad \text{for all } A,\, B, \text{ and } t$$

(see [de Groot 1963]).

Irreversible processes. It is important to realize that time reversibility applies only at a microscopic level, and that most physically observable processes are*irreversible*. As a simple example of an irreversible process, consider a box that is divided into two compartments, one containing vacuum and another filled with air. If a hole is made in the separating wall, air will rush from one side to the other until the pressures on both sides are the same. This process is irreversible, since if the motions of all the particles are reversed, they will not revert to their original configuration.

How can we explain this paradox, given that the underlying physical laws are time reversible? Briefly, the reason is that observable states and microscopic states are not in one-to-one correspondence. In fact, each observable state X' can be realized in a large number of different microscopic ways, all of which are indistinguishable with respect to measurable properties (such as pressure or temperature). This idea is closely related to the concept of*entropy*: letting W denote the number of ways that an observable state X' can be realized, its entropy is given by $S = k \ln W$, where k is the Boltzmann constant. Thus, states with higher entropy can be realized in a greater number of microscopic ways.

Given these facts, irreversible processes can arise as follows. Consider a discrete system where there are only 100 microscopic states X_1, …, X_{100}, and the transition probabilities between them are all equal: $p(X_i, X_j) = p(X_j, X_i)$ for all i and j. We suppose that motion reversal is simply the identity operation, i.e. $-X_i = X_i$ (recalling that $-X$ denotes motion reversal). Clearly this system is microscopically time reversible, since

$$p(A, B) \;=\; p(-B, -A) \qquad \text{for all } A \text{ and } B\,.$$

However, now suppose that the system has only two observable states A' and B', which correspond to 1 and 99 microscopic states respectively. It is easy to verify that $p(A', B') = 99/100$, while $p(B', A') = 1/100$. Thus from an observable point of view, the system is not time reversible: if the system moves from A' to B', and motion reversal is applied to the microscopic state underlying B', then the system is far more likely to move to another microscopic state of B', than it is to return to the original microscopic state underlying A'. The general reason for this behavior is that B' corresponds to a much larger number of microscopic states than A': that is, it has a higher entropy.

In a real thermodynamic system, any measurable increase in entropy corresponds to a huge increase in the number of equivalent microscopic states. For all practical purposes, the probability of returning to the original observable state is zero, and thus any process which increases entropy is said to be *irreversible*process. This is the essence of the second law of thermodynamics: given a closed, isolated system, it will always move from less probable to more probable observable states. (These are only the basic ideas behind irreversible processes; for more rigorous arguments, see van Kampen [1954], de Groot & Mazur [1962], or de Groot [1963].)

Irreversibility of light scattering. As we mentioned, the symmetry of BRDF's is sometimes attributed directly to the time reversibility of physical laws. This is incorrect, because the scattering of light at an ordinary surface is irreversible. (Here we are referring to the*observable* behavior of light scattering, which corresponds to an average behavior over many indistinguishable microscopic states.) There are two reasons for this: first, when a light beam strikes a surface, some of the energy is absorbed (and converted into heat). Motion reversal of all photons and other particles will not convert this heat back into light. Second, the incident beam will generally be scattered in many directions (e.g. by a diffuse surface); and if the direction of this scattered light is reversed to form an incident distribution, it does not recreate the original beam. Both of these situations represent an increase in entropy, and are not reversible.[11] Thus, time reversal invariance does not have any direct implications for ordinary BRDF's, where some light is absorbed and/or scattered in multiple directions.

Light scattering is only reversible at a perfect mirror (if such a thing could be constructed), or at an optically smooth interface between two dielectric materials, as pointed out by Stokes in 1849 (see [Lekner 1987, p. 36] or [Knittl 1962]). It is occasionally claimed that Maxwell's equations themselves are time reversible, but this is true only in special cases. Obviously Maxwell's equations are not time reversible in general, since they describe phenomena such as absorption.[12]

Although time reversal invariance is not useful to us directly, recall that it underlies the principle of detailed balance. Since detailed balance holds even for irreversible processes, it can be applied to light scattering, as we did in Section 6.2. The main limitation of detailed balance (as compared to time reversal) is that it only holds for systems in thermodynamic equilibrium.

[11] See [Jones 1953] for an intuitive discussion of the irreversibility of light scattering; however, note that this paper has a few technical errors.

[12] Although Maxwell's equations are not invariant under the operation of time reversal, they do have other symmetry properties. This has been studied by Šantavý [1961], who describes an operation closely related to time reversal under which Maxwell's equations are indeed invariant.

Appendix 6.D Exceptions to reciprocity

When we derived the reciprocity condition (6.3) for BSDF's, we needed two important assumptions: that there are no external magnetic fields, and that there are no absorbing media.[13] In this section, we give some insight into why these assumptions are necessary, by showing what goes wrong when they are violated.

First, we discuss magnetic fields, which cause problems for polarized light. Then we discuss absorbing media, which at first appear to violate not only the reciprocity condition (6.3), but also the principles of detailed balance and conservation of energy. We explain the apparent contradictions, and we also derive a reciprocity condition that applies to absorbing media (in Section 6.D.2.3).

6.D.1 Magnetic fields and the Faraday effect

We show how magnetic fields can cause reciprocity principles to fail. This includes both Helmholtz reciprocity, and the reciprocity condition (6.3) for general BSDF's.

The source of these problems is the *Faraday effect*, which states that when plane-polarized light propagates within an external magnetic field, the plane of polarization is rotated. For example, consider a polarized beam of light that passes through an electromagnet. According to the Faraday effect, the plane of polarization will rotate in the same direction as the current flow in the magnet. This rotation does not depend on the direction of light propagation, but only on the magnetic field: thus, if the same beam is reflected back and forth through the magnet, the rotation increases each time. This obviously represents an exception to the Helmholtz reciprocity principle, as it was stated for polarized light, and Helmholtz himself was aware of this (see the quotation in Section 6.A.2).

Lord Rayleigh's light trap. As a more dramatic example of how reciprocity can fail, Lord Rayleigh proposed the following *light trap*. Consider a horizontal cylinder filled with a magnetic medium,[14] together with an external field such that polarized light passing through the cylinder is rotated by 45 degrees. Now suppose that a polarizer is placed at either end of the cylinder, oriented so that their planes of polarization are 45 degrees apart. In this situation, light passing in one direction through the cylinder is completely blocked by the second polarizer, while light traveling the

[13] An *absorbing medium* is one that absorbs some of the light energy passing through it, so that the intensity of a light beam decreases with distance.

[14] Note that the Faraday effect only occurs in substances that are *magnetically active*. Oxygen, hydrogen, and water are all magnetically active to some degree [Born & Wolf 1986, p. 3], although the Faraday effect is strongest in substances such as carbon bisulphide [Drude 1900, Chapter 7].

other direction is transmitted. Thus, a source of light at one end of the cylinder$A$ would be visible at the other end B, while a source at B would not be visible from A.

Using this idea, we can show the existence of BSDF's that do not obey the reciprocity condition (6.3) in the presence of a magnetic field. For example, consider a thin film of magnetized iron. When plane-polarized light passes through this film, its plane of polarization is rotated slightly [Drude 1900, p. 451]. Now suppose that the iron is coated with polarizing films on both sides; this will produce an effect similar to the "light trap", i.e. the transmissivity of the surface will be different for light traveling through it in opposite directions. The same can be achieved for reflective surfaces, for example by coating a magnetized mirror with certain kinds of optical crystals [Drude 1900].

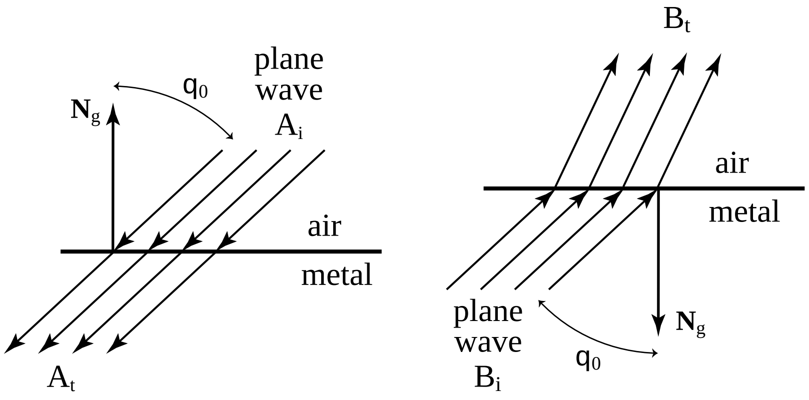
6.D.2 Transmission between absorbing media

Reciprocity also fails when light is transmitted into*absorbing media*. For these media, the radiance of a light beam decreases exponentially with the distance traveled. The absorption is due to electrical conduction, which transforms light energy into electron vibrations (which then appear as heat). The medium may be only slightly absorbing, as with imperfect dielectric materials, or it may be a conductor (metals), in which case light is virtually extinguished after propagating only a few wavelengths.

For absorbing media, there are two separate ways in which reciprocity fails [von Fragstein 1955]. We give a brief introduction to them here, and provide more detail in the following sections.

First, the path of a light beam is not always reversible. For example, consider a light wave that is transmitted from air into metal (Figure 6.2). For some metals, there exists an non-zero angle of incidence where the transmitted beam does not change its direction (i.e. it is not refracted), and yet light beams that go in the opposite direction from metal into air are refracted for all non-zero incident angles. For other metals, the reverse is true: beams are always refracted upon entering the metal, but for beams exiting the metal, there is a non-zero angle where the direction of propagation does not change. Note that these situations do not happen in the familiar case of non-absorbing media, where light beams are refracted for all non-zero angles of incidence, and the path of a light beam is always reversible.

The second effect concerns the*transmissivity* of the interface between two media, i.e. the fraction of incident power that is transmitted through the surface. For absorbing media, a larger fraction of light can be transmitted in one direction than the other. Letting$\tau_{i,j}$ denote the transmissivity from



(a) Air to metal **(b)** Metal to air

Figure 6.2: When absorbing media such as metals are present, the path of a light beam is not always reversible. For example, when a light beamA_i is transmitted from air into some metals, there is a non-zero angle of incidence θ_0 for which the beam does not change its direction of propagation (Figure (a)). However, a beam of lightB_i traveling in the reverse direction (from metal into air) is refracted at the surface, and follows a different path (Figure (b)).

medium i to medium j, the transmissivities in opposite directions are related by

$$\frac{\tau_{1,2}}{\tau_{2,1}} = \frac{1+\kappa_1^2}{1+\kappa_2^2},$$

where κ_i is the *attenuation index* for medium i (which measures the rate of light absorption in that medium).[15] Furthermore, the reflectivity and transmissivity at such an interface can sum to more than one (which cannot happen with non-absorbing media).

At first sight, these properties appear to violate the principles of detailed balance and conservation of energy, respectively. However, this is not the case. In the following sections, we explain these apparent contradictions, and we also derive a more general reciprocity condition for BSDF's that holds even when there are absorbing media.

[15] The attenuation index is defined so that when a light wave travels a single wavelength λ, its amplitude is reduced by a factor of $e^{-2\pi\kappa}$. This is not the same as the *absorption coefficient* σ_a used in the volume rendering and radiation transport literature, which measures the rate of absorption per unit length. The two quantities are related by $\sigma_\mathrm{a} = 4\pi\kappa/\lambda$ [Born & Wolf 1986, p. 614]. Adding further to the confusion, κ is often called the *extinction coefficient*, which is the same name given in the transport literature to the sum of the absorption and scattering coefficients.

6.D.2.1 Non-reversibility of optical paths

We have stated that the path of a light beam between air and metal is not always reversible. To explain this, consider a homogeneous plane wave A_{i} traveling through the air (i.e. an infinitely wide beam, propagating in a given direction). Suppose that this wave strikes the planar boundary of an absorbing medium, where the angle of incidence is $\theta(A_{\mathrm{i}}) = \theta_0$ (see Figure 6.2(a)).

In [von Fragstein 1955], it is shown that for some metals, there is a non-zero value of θ_0 for which the incoming wave will not be refracted (i.e. $\theta(A_{\mathrm{t}}) = \theta(A_{\mathrm{i}}) = \theta_0$). On the other hand, a homogeneous wave B_{i} traveling in the reverse direction *will* be refracted; that is, if it strikes the boundary at an angle $\theta(B_{\mathrm{i}}) = \theta_0$ from inside the metal, it will exit at an angle $\theta(B_{\mathrm{t}})$ that is different from θ_0 (Figure 6.2(b)).

At first sight, this appears to contradict the principle of detailed balance. At thermodynamic equilibrium, we must have the same energy flowing both ways between any given pair of directions; thus, it would seem that if the wave A_{i} is not refracted, then the wave B_{i} should not be refracted as well. (Otherwise, power arriving from the given direction ω_{i} would be scattered to $-\omega_{\mathrm{i}}$, but not vice versa.)

The crucial observation is that the two situations we have considered are actually *not* the reverse of each other. To obtain the refraction results above, the waves A_{i} and B_{i} must both be *homogeneous*, i.e. their amplitude must be constant along each wavefront [von Fragstein 1955]. However, when the wave A_{i} is refracted into metal, the result A_{t} is not a homogeneous wave: the wavefronts are perpendicular to the direction of propagation $\theta(A_{\mathrm{t}}) = \theta_0$, while the surfaces of constant amplitude are parallel to the boundary between the two media [Born & Wolf 1986, p. 616]. This happens because each point on a given wavefront has traveled a different distance through the absorbing medium, and the amplitude of the wave falls off according to the distance traveled.

Because of this, the irreversibility of optical paths between absorbing media is a bit misleading. The situation considered by von Fragstein is not a true reversal of the optical path, because he assumes that the incident wave is homogeneous in both directions. Suppose that instead, we let B_{i} be an inhomogeneous wave of the same type as A_{t}, where the wavefronts are perpendicular to the direction θ_0, but the surfaces of constant amplitude are parallel to the boundary. It is possible to show that this yields a transmitted wave B_{t} of the same form as A_{i}: a homogeneous wave, propagating in the desired direction θ_0. Thus, the requirements of detailed balance are satisfied.

To see that this is true, consider the following experiment. Suppose that the metal forms a thin layer, with air on both sides, and consider a homogeneous wave A_{i} that is incident at the angle θ_0. This wave enters the metal, where it is refracted into an inhomogeneous wave A_{t} traveling in the same direction. This wave propagates to the far side of the metal layer, where we will rename it B_{i}, and is then transmitted into air yielding a wave B_t. This wave B_t must clearly be homogeneous, since all parts of the wave have traversed the same thickness of metal. Furthermore, from Snell's law it is straightforward to show that the waves A_i and B_t have the same direction of propagation [Born & Wolf 1986, p. 629]. Thus, the second refraction (which acts on an inhomogeneous wave B_i) exactly reverses the action of the first, as we claimed above.

6.D.2.2 Apparent non-conservation of energy

Next, we turn to the transmissivity of the interface between absorbing media. We have claimed that the reflectivity and transmissivity at such an interface can sum to more than one, which appears to violate conservation of energy. This can be explained in terms of interference between the incident and reflected light waves.

In particular, consider a planar boundary between air and metal, where the metal has an attenuation index of κ. Suppose that an incident wave A_i strikes the boundary from within the metal, giving rise to a reflected wave A_r and a transmitted wave A_t. Then according to von Fragstein [1950], the transmissivity satisfies

$$\tau \;=\; (1+\kappa^2)(1-\rho)\,,$$

where the reflectivity ρ can be determined from the Fresnel laws [Born & Wolf 1986, p. 628]. Note that the factor $1+\kappa^2$ can be rather large; e.g. for silver it is approximately 400 [von Fragstein 1950, p. 65]. Thus, the amount of transmitted light can be much larger than it would be if $\rho+\tau=1$.

To understand this, we must examine the definitions of reflectivity and transmissivity. They measure the power of the reflected and transmitted waves, as compared to the incident wave:

$$\rho \;=\; \frac{\Phi(A_\mathrm{r})}{\Phi(A_\mathrm{i})}\,, \qquad \tau \;=\; \frac{\Phi(A_\mathrm{t})}{\Phi(A_\mathrm{i})}\,.$$

However, the key observation is that the incident and reflected waves are propagating in the same medium, and that these two waves can *interfere* with each other. The power carried toward the boundary by the combined wave $A_\mathrm{i} + A_\mathrm{r}$ can be either more or less than the intuitively expected value $\Phi(A_\mathrm{i}) - \Phi(A_\mathrm{r})$.

For transmission from metal to air, the combined wave $A_\mathrm{i} + A_\mathrm{r}$ carries more power toward the boundary than expected. This can be shown from Maxwell's equations, where the additional energy appears as *mixed product* terms in the Poynting vector [von Fragstein 1950]. In the one-dimensional case [Salzberg 1948], the power carried by the combined wave can be written in the form

$$(1/2)\,\mathrm{Re}(E_{\mathrm{i}}H_{\mathrm{i}}^{*}) - (1/2)\,\mathrm{Re}(E_{\mathrm{r}}H_{\mathrm{r}}^{*}) - (1/2)\,\mathrm{Re}(E_{\mathrm{i}}H_{\mathrm{r}}^{*} - E_{\mathrm{r}}H_{\mathrm{i}}^{*})\,,$$

where E and H denote the complex amplitudes of the electric and magnetic components of the wave, respectively, and * denotes complex conjugation. The first two terms denote the power of the incident and reflected waves, while the third term measures the additional energy flow due to interference.

Effectively, the interference between incident and reflected waves causes there to be less absorption in the metal near the boundary [von Fragstein 1955]. That is, for a wave propagating far inside the metal, absorption will occur at the usual rate (as determined byκ). However, as the wave approaches the boundary, the rate of absorption becomes smaller, due to interference from the reflected wave. Thus, when the wave finally exits from the metal, it will have much more power than it would if the reduced absorption were not taken into account.

6.D.2.3 A reciprocity condition for BSDF's with absorbing media

We derive a reciprocity condition for BSDF's that applies even when absorbing media are present. This requires only one small change to the argument in Section 6.2.

Recall that for a system in thermodynamic equilibrium, where only non-absorbing media are present, that the quantity L/η^2 is constant throughout the enclosure. When absorbing media are present, this must be modified: it is possible to show that the quantity

$$\frac{L(\mathbf{x},\omega)\,(1+\kappa^2)}{\eta^2}$$

is constant throughout the enclosure [von Fragstein 1950, Tingwaldt 1952], where$L$, κ, and η are parameterized by frequency ν. Thus according to this formula, the equilibrium radiance is smaller in an absorbing medium than in a non-absorbing one.

By repeating the argument of Section 6.2, we can now show that an arbitrary, physically valid BSDF must satisfy

$$\frac{f_{\mathrm{s}}(\omega_{\mathrm{i}}\!\to\!\omega_{\mathrm{o}})\,(1+\kappa_{\mathrm{o}}^2)}{\eta_{\mathrm{o}}^2} \;=\; \frac{f_{\mathrm{s}}(\omega_{\mathrm{o}}\!\to\!\omega_{\mathrm{i}})\,(1+\kappa_{\mathrm{i}}^2)}{\eta_{\mathrm{i}}^2}\,, \tag{6.6}$$

where all quantities are parameterized by frequencyν. This is clearly a generalization of the condition (6.3) given for non-absorbing media.

To put this into perspective, however, the difference between (6.6) and (6.3) is utterly insignificant for the typical participating media used in graphics, because the attenuation indices are so small.

For example, consider a medium that is so dense that 99% of the incident light is absorbed after a distance of one millimeter. This corresponds to an attenuation index of only $\kappa = 0.00037$ (for light with a wavelength of 500nm), so that the $(1 + \kappa^2)$ change in transmissivity is inappreciable.

Also, note that the participating media in graphics are often not true absorbing media, but instead consist of small particles (e.g. clouds, fog, smoke). These materials are described by *scattering theory* [van de Hulst 1957], rather than the theory of absorbing media described here. (In a true absorbing medium, the particles must be of negligible size compared to the wavelength: for example, an iodine solution, or a cloud of chlorine gas.)

6.D.2.4 Discussion

Given these bizarre examples, it is clear that absorbing media cannot be described with familiar optical concepts. The idea of independent waves propagating and reflecting, each with its own power, is simply meaningless in an absorbing medium [Salzberg 1948]. For example, consider the standard Fresnel formulas for reflection and transmission between absorbing media [Born & Wolf 1986, p. 628]. According to these formulas, there is non-zero reflection even at a ficticious interface between two *identical* media; furthermore, the corresponding transmissivity is greater than one. To handle such situations correctly, it is necessary to work with explicit wave descriptions (e.g. monochromatic waves described by their phase and amplitude), rather than with secondary concepts such as power.

It is reassuring to note that the strange effects we have described are restricted to the absorbing media themselves. For example, consider a wave A_{i} that is transmitted from air, through a metal film, and then back into air to yield a wave A_{t}. It can be shown that the optical path is reversible, and that the transmissivity of the film is the same in both directions [Lekner 1987]. This holds even if the film consists of many layers of absorbing and non-absorbing media (known as a *striated medium*).

Chapter 7

A Self-Adjoint Operator Formulation of Light Transport

As we have mentioned, it is very convenient for implementations to use the same scattering rules for light, importance, and particles. This is desirable for theoretical work as well, so that we may avoid the use of adjoint operators. However, the existing light transport models in graphics fail to achieve this, even when the scene model is physically valid. To obtain symmetry, the typical solution is to limit the scene to reflective surfaces (or more generally, to require that all media have the same refractive index). This is a major restriction, since it disallows materials such as glass and water, which occur frequently in graphics models.

In this chapter, we develop a framework where light, importance, and particles obey the same scattering rules, for any physically valid scene. Technically, this requires us to define operators that are *self-adjoint*, so that the same operators apply in all situations. The major issue, of course, is how to deal with transmission between media with different indices of refraction. The solution turns out to very simple and practical, and also reveals interesting connections with classical geometric optics.

7.1 Concepts of the new framework

We would like to find a framework where light and importance always obey the same transport equation, even when there are media with different refractive indices. We show how to achieve this by modifying the framework of Chapter 4. In this section, we discuss only the formal changes that are required, leaving interpretation and discussion for later. We assume throughout this chapter that only physically valid materials are used in the scene model.

The problem. The problem with the framework that we described in Chapter 4 is that sometimes the local scattering operator $\mathbf{K}$ is not self-adjoint. In this case, the light and importance transport operators are different, since they are given by $\mathbf{T}_L = \mathbf{K}\mathbf{G}$ and $\mathbf{T}_W = \mathbf{K}^*\mathbf{G}$ respectively.

To fix this, recall that $\mathbf{K}$ is defined by

$$(\mathbf{K}L)(\mathbf{x}, \omega_\mathrm{o}) \;=\; \int_{\mathcal{S}^2} f_\mathrm{s}(\mathbf{x}, \omega_\mathrm{i} \!\rightarrow\! \omega_\mathrm{o})\, L(\mathbf{x}, \omega_\mathrm{i})\, d\sigma^\perp_\mathbf{x}(\omega_\mathrm{i})\;,$$

and that $\mathbf{K} = \mathbf{K}^*$ whenever f_s is symmetric. From Section 6.2, we also know that

$$\frac{f_\mathrm{s}(\omega_\mathrm{i} \!\rightarrow\! \omega_\mathrm{o})}{\eta_\mathrm{o}^2} \;=\; \frac{f_\mathrm{s}(\omega_\mathrm{o} \!\rightarrow\! \omega_\mathrm{i})}{\eta_\mathrm{i}^2} \tag{7.1}$$

for any physically valid BSDF; that is, $f_\mathrm{s}(\omega_\mathrm{i} \!\rightarrow\! \omega_\mathrm{o}) / \eta_\mathrm{o}^2$ is a symmetric function. Thus, if we could change the definition of $\mathbf{K}$ to use this symmetric function $f_\mathrm{s}/\eta_\mathrm{o}^2$, rather than f_s itself, then we would have $\mathbf{K} = \mathbf{K}^*$ for any physically valid scene model.

The solution. There is a very simple way to achieve this. The idea is to define a new solid angle measure σ', which *replaces* the usual measure σ in all radiometric quantities and definitions. We call the new measure *basic solid angle,* and it is defined by[1]

$$d\sigma'_\mathbf{x}(\omega) \;=\; \eta^2(\mathbf{x}, \omega)\, d\sigma(\omega)\;, \tag{7.2}$$

where $\eta(\mathbf{x}, \omega)$ is the refractive index of the medium that is adjacent to $\mathbf{x}$ in the direction ω. Note that unlike the usual solid angle measure σ, the basic solid angle measure $\sigma'_\mathbf{x}$ is

[1] As always, when measures are defined using "infinitesimals", it should be understood as shorthand for a formal definition involving integration. For example, a more precise definition of equation (7.2) would be

$$\sigma'_\mathbf{x}(D) \;=\; \int_D \eta^2(\mathbf{x}, \omega)\, d\sigma(\omega)\;,$$

where $D \subset \mathcal{S}^2$ is a σ-measurable set of directions.

a function of position, since its value depends on the refractive indices of the surrounding media.

By mechanically substituting this new measure into all of our old definitions, we obtain a framework with very desirable symmetry properties. The symbols for the new quantities are obtained by appending a prime symbol (e.g. L'), while their names are obtained by prefixing the word *basic* (e.g. basic radiance). This naming convention is justified by the unique invariance properties of these quantities, which will be studied further in Appendix 7.A. It also extends the terminology of Nicodemus, who first introduced the ideas of *basic radiance* and *basic throughput* [Nicodemus 1976]. By replacing the solid angle measure as outlined above, we obtain these concepts along with a variety of new ones (which generally differ from their original definitions by a factor of η^2):

- The *basic projected solid angle* measure ($\sigma^{\perp}_{\mathbf{x}}{}'$) on $\mathcal{S}^2$:

$$\begin{aligned} d\sigma^{\perp}_{\mathbf{x}}{}'(\omega) &= |\omega \cdot \mathbf{N}(\mathbf{x})|\, d\sigma'_{\mathbf{x}}(\omega) \\ &= \eta^2(\mathbf{x},\omega)\, d\sigma^{\perp}_{\mathbf{x}}(\omega)\,. \end{aligned} \tag{7.3}$$

- The *basic throughput* measure (μ') on the ray space $\mathcal{R}$:

$$\begin{aligned} d\mu'(\mathbf{x},\omega) &= dA(\mathbf{x})\, d\sigma^{\perp}_{\mathbf{x}}{}'(\omega) \\ &= \eta^2(\mathbf{x},\omega)\, dA(\mathbf{x})\, d\sigma^{\perp}_{\mathbf{x}}(\omega)\,, \\ \text{or in other words,} \qquad d\mu'(\mathbf{r}) &= \eta^2(\mathbf{r})\, d\mu(\mathbf{r})\,. \end{aligned} \tag{7.4}$$

- *Basic radiance* (L'):

$$L'(\mathbf{r}) = \frac{d\Phi(\mathbf{r})}{d\mu'(\mathbf{r})} = \frac{L(\mathbf{r})}{\eta^2(\mathbf{r})}\,. \tag{7.5}$$

  Basic spectral radiance (L'_ν) is defined in a similar way.

- The *basic inner product* on $L_2(\mathcal{R})$:

$$\begin{aligned} \langle f, g\rangle' &= \int_{\mathcal{R}} f(\mathbf{r})\, g(\mathbf{r})\, d\mu'(\mathbf{r}) \\ &= \int_{\mathcal{R}} f(\mathbf{r})\, g(\mathbf{r})\, \eta^2(\mathbf{r})\, d\mu(\mathbf{r})\,. \end{aligned} \tag{7.6}$$

This inner product will be used to define adjoint operators in this chapter, i.e. two operators $\mathbf{H}$ and $\mathbf{H}^*$ are adjoint if $\langle \mathbf{H}^* f, g\rangle' \;=\; \langle f, \mathbf{H} g\rangle'$ for all f and g.

- The *basic BSDF* (f_{s}'):

$$f_{\mathrm{s}}'(\mathbf{x}, \omega_{\mathrm{i}} \!\rightarrow\! \omega_{\mathrm{o}}) \;=\; \frac{dL_{\mathrm{o}}'(\mathbf{x}, \omega_{\mathrm{o}})}{dE(\mathbf{x}, \omega_{\mathrm{i}})} \;=\; \frac{f_{\mathrm{s}}(\mathbf{x}, \omega_{\mathrm{i}} \!\rightarrow\! \omega_{\mathrm{o}})}{\eta_{\mathrm{o}}^2}\,, \tag{7.7}$$

  where we have once again used the convention that $\eta_{\mathrm{o}} = \eta(\mathbf{x}, \omega_{\mathrm{o}})$.

Note that we have only redefined quantities whose definitions depend on solid angle. All other quantities (e.g. irradiance) are left unchanged.

These new quantities have interesting symmetry properties. Most importantly, the basic BSDF of any physically valid material is guaranteed to be symmetric:

$$f_{\mathrm{s}}'(\mathbf{x}, \omega_{\mathrm{i}} \!\rightarrow\! \omega_{\mathrm{o}}) \;=\; f_{\mathrm{s}}'(\mathbf{x}, \omega_{\mathrm{o}} \!\rightarrow\! \omega_{\mathrm{i}})\,.$$

This follows directly from the general reciprocity principle (7.1) that was proven in Chapter 6. (As a special case of this, Appendix 7.C derives the basic BSDF for perfect specular refraction and shows how to express it in a symmetric form.)

The other quantities defined above also have interesting symmetry properties, some of which take the form of *optical invariants* (a notion from classical geometric optics). These properties are discussed in Appendix 7.A.

7.2 The self-adjoint operator formulation

We now show how to put these concepts together into a framework of self-adjoint transport operators. The main idea is to use *basic radiance* (L') for all light transport calculations, while for importance transport the standard definitions are used.[2] As we will see, this leads to the desired symmetry properties, because basic radiance and importance satisfy the same transport equation. Furthermore, this framework computes exactly the same value for every measurement as before.

[2] Recall that importance has units of $[S \cdot \mathrm{W}^{-1}]$, and thus it is not affected by the new solid angle measure.

We now give the details of our framework. Measurements are computed using the basic inner product (7.6):

$$I \;=\; \langle W_{\mathrm{e}}, L'_{\mathrm{i}} \rangle' \,. \tag{7.8}$$

This equation gives the same results as the original measurement equation (4.20), since

$$\langle W_{\mathrm{e}}, L'_{\mathrm{i}} \rangle' \;=\; \int_{\mathcal{R}} W_{\mathrm{e}}(\mathbf{r})\, \frac{L_{\mathrm{i}}(\mathbf{r})}{\eta^2(\mathbf{r})} \left[\eta^2(\mathbf{r})\, d\mu'(\mathbf{r})\right] \;=\; \langle W_{\mathrm{e}}, L_{\mathrm{i}} \rangle \,.$$

The propagation operator $\mathbf{G}$ is unchanged; however, we define the new *basic local scattering operator* ($\mathbf{K}'$) by

$$(\mathbf{K}'h)(\mathbf{x}, \omega_{\mathrm{o}}) \;=\; \int_{\mathcal{S}^2} f'_{\mathrm{s}}(\mathbf{x}, \omega_{\mathrm{i}} \!\to\! \omega_{\mathrm{o}})\, h(\mathbf{x}, \omega_{\mathrm{i}})\, d\sigma^{\perp\prime}_{\mathbf{x}}(\omega_{\mathrm{i}}) \,. \tag{7.9}$$

It is helpful to expand the basic scattering equation $L'_{\mathrm{o}} = \mathbf{K}' L'_{\mathrm{i}}$, to see how the old and new quantities are related:

$$\frac{L_{\mathrm{o}}}{\eta_{\mathrm{o}}^2} \;=\; \int_{\mathcal{S}^2} \frac{f_{\mathrm{s}}(\mathbf{x}, \omega_{\mathrm{i}} \!\to\! \omega_{\mathrm{o}})}{\eta_{\mathrm{o}}^2}\, \frac{L_{\mathrm{i}}(\mathbf{x}, \omega_{\mathrm{i}})}{\eta_{\mathrm{i}}^2} \left[\eta_{\mathrm{i}}^2\, d\sigma^{\perp}_{\mathbf{x}}(\omega_{\mathrm{i}})\right] \,. \tag{7.10}$$

We see that the η_{i}^2 and η_{o}^2 factors are handled consistently. The most important difference is that the BSDF has been replaced by a symmetric quantity (the basic BSDF f'_{s}). Because of this, it is straightforward to check that the scattering operator $\mathbf{K}'$ is self-adjoint (see Appendix 7.B for details). Also notice that when there is only a single medium at $\mathbf{x}$, then $\mathbf{K}'$ is identical to the original operator $\mathbf{K}$ (since the hidden factor of η_{o}^2 in f'_{s} cancels the hidden factor of η_{i}^2 in $\sigma^{\perp\prime}$).

Light and importance transport operators. We define the transport operators $\mathbf{T}_X$ and solution operators $\mathbf{S}_X$ in the same way as before (see Section 4.7). These can be summarized as follows:

	Exitant	Incident
Light	$\mathbf{T}_{L'_{\mathrm{o}}} = \mathbf{K}'\mathbf{G}$	$\mathbf{T}_{L'_{\mathrm{i}}} = \mathbf{G}\mathbf{K}'$
Importance	$\mathbf{T}_{W_{\mathrm{o}}} = \mathbf{K}'^{*}\mathbf{G}$	$\mathbf{T}_{W_{\mathrm{i}}} = \mathbf{G}\mathbf{K}'^{*}$

However, because $\mathbf{G} = \mathbf{G}^*$ and $\mathbf{K}' = \mathbf{K}'^*$ for physically valid scenes, these definitions simplify to

$$\mathbf{T}_{L'_\mathrm{o}} = \mathbf{T}_{W_\mathrm{o}} = \mathbf{K}'\mathbf{G}\,,$$

$$\mathbf{T}_{L'_\mathrm{i}} = \mathbf{T}_{W_\mathrm{i}} = \mathbf{G}\mathbf{K}'\,.$$

Thus, basic radiance and importance obey the same transport equation.

We should emphasize that only the light transport operators have been changed in this framework; the importance transport operators have the same definitions as before. This is not immediately obvious, since we originally defined $\mathbf{T}_{W_\mathrm{o}} = \mathbf{K}^*\mathbf{G}$, and now we have defined $\mathbf{T}_{W_\mathrm{o}} = \mathbf{K}'^*\mathbf{G}$. However, $\mathbf{K}^*$ and $\mathbf{K}'^*$ are actually the same operator (as will be shown in Appendix 7.B). Because of this fact (which holds in all environments, physically valid or not), we continue to use the same symbols $\mathbf{T}_{W_\mathrm{i}}$ and $\mathbf{T}_{W_\mathrm{o}}$ for the importance transport operators.

To summarize, the main idea of the self-adjoint framework is to use basic radiance rather than radiance for light transport calculations, and to compensate for this by including a factor of η^2 in the measurement equation. With these simple changes, light, importance, and particles can be scattered and propagated in the same way. Further details of the framework are described in Appendix 7.B.

7.3 Consequences for implementations

We show how this framework affects the implementation of path tracing and bidirectional rendering algorithms. It is actually very simple to use the self-adjoint framework, since no scaling factors are required for transmission between different media, and the same scattering rules apply to light, importance, and particles.

Consider the structure of an ordinary path tracing algorithm. The calculation starts at the viewpoint, where a particular pixel value $I_j = \langle W_\mathrm{e}^{(j)}, L_\mathrm{o} \rangle$ is estimated by sampling a ray that contributes to this integral. The initial ray lies in a medium with some refractive index η_1. We then proceed by following a path backward, through a sequence of media with indices $\eta_2, \ldots, \eta_k$, until finally a light source is reached, and the emitted radiance L_e is computed. With a standard framework (e.g. that of Chapter 4), a scaling factor of η_i^2/η_{i+1}^2 is required between each pair of media i and $i + 1$ to account for the change in radiance.

With the self-adjoint framework, we obtain the same result in a simpler way. Starting again at the viewpoint, we sample a ray $\mathbf{r}$ to estimate the basic inner product $I_j = \langle W_{\mathrm{e}}^{(j)}, L'_{\mathrm{o}} \rangle'$. Since $\mathbf{r}$ lies in medium η_1, the weight for this ray has an extra factor of η_1^2 compared to the standard path tracing implementation. However, we now evaluate the *basic* radiance along this ray, which means that no special scaling factors are required as we follow a path backward through media $\eta_2, \ldots, \eta_k$. When the path finally reaches a light source, we must divide its emitted radiance by η_k^2 to obtain basic radiance.[3] Thus the combined scaling factor for this path is η_1^2 / η_k^2, which is identical to the product of all the scaling factors above.

With bidirectional algorithms, some of the calculations are carried out by propagating information forward from the light sources. For example, consider the "pool of water" scene from Section 5.2. Suppose that a particle tracing pass is used to accumulate the caustics on the pool bottom in a view-independent form (e.g. a texture map), which is then rendered using a ray tracing pass. With the self-adjoint framework, the particle tracing pass does not require any changes. The ray tracing pass is similar to the path tracing algorithm described above, except that now the "emission function" consists of a texture map on the pool bottom, which must be expressed in the form of basic radiance before it is used. This is done looking up the irradiance value in the texture map, and dividing it by the η^2 value of the surrounding medium (i.e. water).

Similarly, the self-adjoint framework can be used with algorithms such as density estimation [Shirley et al. 1995], the photon map [Jensen 1996], and bidirectional path tracing, by making changes of a similar nature.

We should mention that it is also possible to obtain a symmetric transport framework by working with the quantities L/η and W/η (rather than L and W), and computing measurements using the ordinary inner product. With this convention, light and importance are both scaled by the same factor of $\eta_{\mathrm{t}} / \eta_{\mathrm{i}}$ when they enter a new medium. However, this scheme only gives correct results when all sources and sensors are located in media whose refractive index is $\eta = 1$.

[3] Alternatively, the emission from light sources can be expressed using basic radiance in the first place.

Appendix 7.A Classical optical invariants

The quantities defined in Section 7.1 have several important symmetry properties. Some of these correspond to the classical notion of an optical invariant, a topic that we explore here.

Optical invariants are defined within the framework of classical geometric optics, which studies the formation of images by systems of mirrors and lenses. An *optical invariant* is a quantity that preserved by such systems, that is, a numerical measurement that has the same value for any real object and its image.

7.A.1 The Smith-Helmholtz invariant

Of the classical optical invariants, the most famous is the *Smith-Helmholtz* or *Lagrange* invariant, which was first stated by Smith in his *Compleat System of Opticks* (Cambridge, 1738), and subsequently rediscovered by Lagrange (1803) and von Helmholtz [1856, p. 74].

Consider a lens system that has rotational symmetry about the lens axis, so that it can be represented by a planar diagram (see Figure 7.1). Given some object and its corresponding image, the Smith-Helmholtz invariant states that

$$\eta h \alpha \;=\; \eta' h' \alpha' \,, \tag{7.11}$$

where η is the refractive index of the medium containing the object, h is the object height, and α is the angle over which light is radiated from the object toward the lens system. The quantities η', h', and α' denote the corresponding quantities for the image (where α' is now the angle over which light is received from the lens system, at a given point of the image). The ratio h'/h is called the *linear magnification* of the lens system, while α'/α is called the *angular magnification*; equation (7.11) shows that these quantities are related in a simple way.

7.A.2 The invariance of basic throughput

Another classical invariant is *basic throughput* [Nicodemus 1976, p. 37], which is also known as *etendue* [Steel 1974]. This quantity has already been defined (Section 7.1), but we repeat its definition here:

$$\mu'(D) \;=\; \int_D \eta^2(\mathbf{r})\, d\mu(\mathbf{r})\,,$$

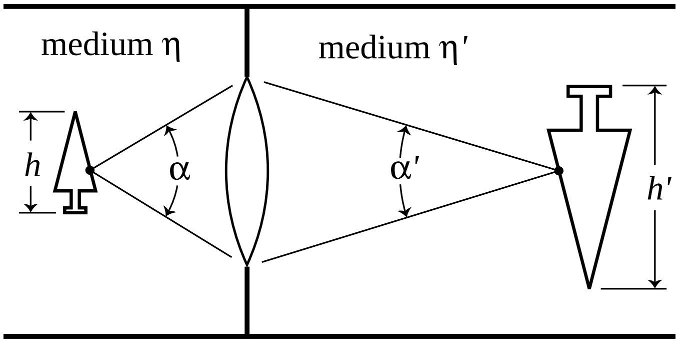


Figure 7.1: The Smith-Helmholtz invariant relates the geometry of an object and its corresponding image, according to the equation $\eta h \alpha = \eta' h' \alpha'$.

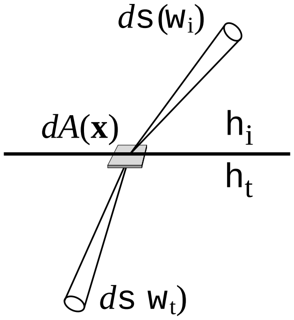


Figure 7.2: When a beam of light is refracted, its basic throughput is preserved.

where $D \subset \mathcal{R}$ is a set of rays. With respect to classical geometric optics, D would represent the beam of rays that leave an object toward the lens system, eventually forming an image. The invariance of this quantity implies that as this light beam propagates through an optical system, its basic throughput μ' is preserved.

An example: perfect specular refraction. We show how this invariance can be proven, for the special case of perfect specular refraction. Consider a beam of light that strikes small surface patch $dA(\mathbf{x})$, occupying a solid angle of $d\sigma(\omega_{\rm i})$ (see Figure 7.2). Let $\omega_{\rm t}$ be the direction of the refracted beam, which occupies a solid angle of $d\sigma(\omega_{\rm t})$. In Section 5.2.1.1, we have already shown that the incident and transmitted beams are related by

$$\eta_{\mathrm{i}}^2 \, d\sigma^{\perp}(\omega_{\mathrm{i}}) \;=\; \eta_{\mathrm{t}}^2 \, d\sigma^{\perp}(\omega_{\mathrm{t}}) \,, \tag{7.12}$$

recalling that $\sigma^{\perp}$ denotes the projected solid angle. Since these beams travel through the same area $dA(\mathbf{x})$ on the surface, we thus have

$$\begin{aligned} \eta_{\mathrm{i}}^2 \, dA(\mathbf{x}) \, d\sigma^{\perp}(\omega_{\mathrm{i}}) &= \eta_{\mathrm{t}}^2 \, dA(\mathbf{x}) \, d\sigma^{\perp}(\omega_{\mathrm{t}}) \\ \Longrightarrow \qquad \eta_{\mathrm{i}}^2 \, d\mu(\mathbf{r}_{\mathrm{i}}) &= \eta_{\mathrm{t}}^2 \, d\mu(\mathbf{r}_{\mathrm{t}}) \\ \Longrightarrow \qquad d\mu'(\mathbf{r}_{\mathrm{i}}) &= d\mu'(\mathbf{r}_{\mathrm{t}}) \,, \end{aligned} \tag{7.13}$$

where $\mathbf{r}_{\mathrm{i}} = (\mathbf{x}, \omega_{\mathrm{i}})$ and $\mathbf{r}_{\mathrm{t}} = (\mathbf{x}, \omega_{\mathrm{t}})$ represent the incident and transmitted rays. By integrating this relationship, we can show that basic throughput μ' is invariant for an arbitrary set of rays D that strike the surface. This affirms the invariance of μ' in the special case of refraction. It is also straightforward to show that basic throughput is preserved when light is reflected, or when it propagates through a constant medium (see Appendix 7.B). Using more advanced techniques, it is possible to show that basic throughput is actually preserved in any system that obeys the laws of geometric optics [Nicodemus 1963].

Note that the Smith-Helmholtz invariant can be derived as a special case of this law. To see this, observe that for rotationally symmetric lens systems, the area dA of an object is proportional to y^2, while the solid angle $d\sigma$ over which light radiates is proportional to α^2. Thus the Smith-Helmholtz invariant follows immediately from (7.13).[4]

7.A.3 The invariance of basic radiance

If we assume that each light beam follows a single path through an optical system (i.e. partial reflection is not allowed), and that there are no losses due to absorption, we can also show the invariance of *basic radiance* [Nicodemus 1976, p. 26]. That is, as a beam of light propagates through an optical system, its basic radiance $L' = L/\eta^2$ is preserved (this is known as *Abbe's law* [Keitz 1971, p. 195]).

The invariance of basic radiance can be derived directly from its definition,

$$L'(\mathbf{r}) \;=\; \frac{d\Phi(\mathbf{r})}{d\mu'(\mathbf{r})} \,.$$

[4] Note that the Smith-Helmholtz equation (7.11) is strictly valid only for infinitesimally small objects that are aligned with the optical axis.

That is, as a beam of light propagates through an optical system, its power$d\Phi$ and its basic throughput $d\mu'$ are both preserved (by conservation of energy, and the invariance of basic throughput). Thus, basic radiance is invariant as well. This can be shown under very general conditions by using thermodynamic principles [Liebes 1969].

Basic spectral radiance L_ν/η^2 is an optical invariant as well, when it is parameterized by frequency. However, if spectral radiance is parameterized by wavelength, thenL_λ/η^3 is invariant instead [Nicodemus 1976, p. 52], since wavelengths (unlike frequencies) are modified at the interface (see Section 6.2).

The other "basic" quantities we have defined also possess symmetry properties, however they do not take the form of optical invariants. For example, the basic BSDF is symmetric, but does not correspond to any property that is preserved by beams propagating through an optical system. Similarly, equation (7.12) implies that the basic projected solid angle is preserved at a refractive interface ($d\sigma^{\perp\prime}(\omega_{\mathrm{i}}) = d\sigma^{\perp\prime}(\omega_{\mathrm{t}})$).

Appendix 7.B Properties of the new operators

We consider the adjoints and norms of the operators defined in this chapter. (The invertibility properties are unchanged from Section 4.B.1.)

7.B.1 Adjoints

Recall that adjoint operators in this chapter are defined with respect to the basic inner product, i.e. two operators $\mathbf{H}$ and $\mathbf{H}^*$ are adjoint if $\langle \mathbf{H}^* f, g\rangle' = \langle f, \mathbf{H} g\rangle'$ for all f and g.

Lemma 7.1. *The reversal map M preserves the measure μ'. In other words, $\mu'(M(D)) = \mu'(D)$ for any measurable set $D \subset \tilde{\mathcal{R}}$.*

The proof depends on the fact that the refractive indices of $\mathbf{r}$ and $M(\mathbf{r})$ are always equal. It is otherwise similar to the proof of Lemma 4.4.

Theorem 7.2. *The operator $\mathbf{G}$ is self-adjoint (with respect to the basic inner product).*

The proof is similar to Theorem 4.5, but requires the preceding lemma.

Theorem 7.3. *The adjoint of $\mathbf{K}'$ is given by*

$$(\mathbf{K}'^* h)(\mathbf{x}, \omega_{\mathrm{o}}) \;=\; \int_{\mathcal{S}^2} f_{\mathrm{s}}'^*(\mathbf{x}, \omega_{\mathrm{i}} \!\rightarrow\! \omega_{\mathrm{o}})\, h(\mathbf{x}, \omega_{\mathrm{i}})\, d\sigma_{\mathbf{x}}^{\perp\prime}(\omega_{\mathrm{i}})\,.$$

In particular, $\mathbf{K}'$ is self-adjoint for any physically valid scene model.

The proof is similar to Theorem 4.6. The last statement follows from the fact that $f_{\mathrm{s}}' = f_{\mathrm{s}}'^*$ for physically valid scenes (7.1).

Corollary 7.4. *The operators $\mathbf{K}'^*$ and $\mathbf{K}^*$ are the same (for all scenes).*

Proof. We have

$$\begin{aligned}
(\mathbf{K}'^* h)(\mathbf{x}, \omega_{\mathrm{o}}) &= \int_{\mathcal{S}^2} f_{\mathrm{s}}'^*(\mathbf{x}, \omega_{\mathrm{i}} \!\rightarrow\! \omega_{\mathrm{o}})\, h(\mathbf{x}, \omega_{\mathrm{i}})\, d\sigma_{\mathbf{x}}^{\perp\prime}(\omega_{\mathrm{i}}) \\
&= \int_{\mathcal{S}^2} f_{\mathrm{s}}'(\mathbf{x}, \omega_{\mathrm{o}} \!\rightarrow\! \omega_{\mathrm{i}})\, h(\mathbf{x}, \omega_{\mathrm{i}})\, d\sigma_{\mathbf{x}}^{\perp\prime}(\omega_{\mathrm{i}}) \\
&= \int_{\mathcal{S}^2} \frac{f_{\mathrm{s}}(\mathbf{x}, \omega_{\mathrm{o}} \!\rightarrow\! \omega_{\mathrm{i}})}{\eta_{\mathrm{i}}^2}\, h(\mathbf{x}, \omega_{\mathrm{i}})\, \left[\eta_{\mathrm{i}}^2\, d\sigma_{\mathbf{x}}^{\perp}(\omega_{\mathrm{i}})\right] \\
&= \int_{\mathcal{S}^2} f_{\mathrm{s}}(\mathbf{x}, \omega_{\mathrm{o}} \!\rightarrow\! \omega_{\mathrm{i}})\, h(\mathbf{x}, \omega_{\mathrm{i}})\, d\sigma_{\mathbf{x}}^{\perp}(\omega_{\mathrm{i}}) \\
&= (\mathbf{K}^* h)(\mathbf{x}, \omega_{\mathrm{o}})\,. \quad \blacksquare
\end{aligned}$$

Thus importance obeys the same scattering rules in both frameworks. In physically valid scenes, we also have $\mathbf{K}' = \mathbf{K}'^*$, so that $\mathbf{K}'$, $\mathbf{K}'^*$, and $\mathbf{K}^*$ are all the same operator.

7.B.2 Norms

Since the new operators are defined using the basic throughput measureμ', it will be convenient to define a new set of L_p norms, denoted $\|\cdot\|'_p$:

$$\|f\|'_p \;=\; \left(\int_{\mathcal{R}} |f(\mathbf{r})|^p \, d\mu'(\mathbf{r}) \right)^{\frac{1}{p}} . \tag{7.14}$$

(The norm $\|\cdot\|_\infty$ defined by equation (4.9) is not affected by this change, but we will relabel it$\|\cdot\|'_\infty$ for consistency.)

The old and new norms are always within a constant factor of each other, as stated by the following lemma:

Lemma 7.5. *Let* $\eta_{\min}$ *and* $\eta_{\max}$ *denote the minimum and maximum refractive indices in the given scene. Then for any* $1 \le p \le \infty$ *and any* $f \in L_p(\mathcal{R})$*, we have*

$$(\eta^2_{\min})^{1/p} \|f\|_p \;\le\; \|f\|'_p \;\le\; (\eta^2_{\max})^{1/p} \|f\|_p \,. \tag{7.15}$$

Furthermore if $\mathbf{H}$ *is any bounded operator on* $L_p(\mathcal{R})$*, then*

$$\|\mathbf{H}\|_p \;\le\; \left(\frac{\eta^2_{\max}}{\eta^2_{\min}} \right)^{\frac{1}{p}} \|\mathbf{H}\|'_p \,. \tag{7.16}$$

The proofs follow directly from the corresponding definitions.

As a corollary, note that the space $L_p(\mathcal{R})$ contains the same functions when it is defined using either of the norms $\|\cdot\|_p$ or $\|\cdot\|'_p$, since the two norms are always within a constant factor. Thus we can refer to $L_p(\mathcal{R})$ in either case without ambiguity.

Theorem 7.6. $\|\mathbf{K}'\|'_p < 1$ *for any physically valid scene, and for any*$1 \le p \le \infty$.

Proof. The proof is very similar to [Arvo 1995, Appendix A.8]. First, we consider the case$p = 1$. To bound the operator norm $\|\mathbf{K}'\|'_1$, we must find a number m such that

$$\|\mathbf{K}'h\|'_1 \;\le\; m\,\|h\|'_1 \qquad \text{for any function} \quad h \in L_1(\mathcal{R}) \,.$$

To obtain such a bound, we compute

$$\begin{aligned}
\|\mathbf{K}'h\|_1' &= \int_{\mathcal{R}} (\mathbf{K}'h)(\mathbf{r})\, d\mu'(\mathbf{r}) \\
&= \int_{\mathcal{S}^2} \int_{\mathcal{M}} (\mathbf{K}'h)(\mathbf{x}, \omega_{\mathrm{o}})\, dA(\mathbf{x})\, d\sigma_{\mathbf{x}}^{\perp\prime}(\omega_{\mathrm{o}}) \\
&= \int_{\mathcal{S}^2} \int_{\mathcal{M}} \int_{\mathcal{S}^2} f_{\mathrm{s}}'(\mathbf{x}, \omega_{\mathrm{i}} \!\to\! \omega_{\mathrm{o}})\, h(\mathbf{x}, \omega_{\mathrm{i}})\, d\sigma_{\mathbf{x}}^{\perp\prime}(\omega_{\mathrm{i}})\, dA(\mathbf{x})\, d\sigma_{\mathbf{x}}^{\perp\prime}(\omega_{\mathrm{o}}) \\
&= \int_{\mathcal{M}} \int_{\mathcal{S}^2} \left[\int_{\mathcal{S}^2} f_{\mathrm{s}}'(\mathbf{x}, \omega_{\mathrm{i}} \!\to\! \omega_{\mathrm{o}})\, d\sigma_{\mathbf{x}}^{\perp\prime}(\omega_{\mathrm{o}}) \right] h(\mathbf{x}, \omega_{\mathrm{i}})\, d\sigma_{\mathbf{x}}^{\perp\prime}(\omega_{\mathrm{i}})\, dA(\mathbf{x})
\end{aligned}$$

(where we have dropped the absolute value signs, since all quantities are positive).

To obtain an upper bound on this expression, we let m denote the maximum value attained by the bracketed quantity over the entire domain of the outer integrals:

$$m \;=\; \operatorname*{ess\,sup}_{(\mathbf{x}, \omega_{\mathrm{i}}) \in \mathcal{R}} \int_{\mathcal{S}^2} f_{\mathrm{s}}'(\mathbf{x}, \omega_{\mathrm{i}} \!\to\! \omega_{\mathrm{o}})\, d\sigma_{\mathbf{x}}^{\perp\prime}(\omega_{\mathrm{o}})\,.$$

We thus have

$$\|\mathbf{K}'h\|_1' \;\le\; m \int_{\mathcal{M}} \int_{\mathcal{S}^2} h(\mathbf{x}, \omega_{\mathrm{i}})\, d\sigma^{\perp\prime}(\omega_{\mathrm{i}})\, dA(\mathbf{x}) \;=\; m\, \|h\|_1'\,,$$

so that m is an upper bound on the operator norm $\|\mathbf{K}'\|_1'$.

To better understand the meaning of this bound, we re-express it in terms of the ordinary BSDF f_{s}:

$$\begin{aligned}
m &= \operatorname*{ess\,sup}_{(\mathbf{x}, \omega_{\mathrm{i}}) \in \mathcal{R}} \int_{\mathcal{S}^2} f_{\mathrm{s}}'(\mathbf{x}, \omega_{\mathrm{i}} \!\to\! \omega_{\mathrm{o}})\, d\sigma_{\mathbf{x}}^{\perp\prime}(\omega_{\mathrm{o}}) \qquad (7.17) \\
&= \operatorname*{ess\,sup}_{(\mathbf{x}, \omega_{\mathrm{i}}) \in \mathcal{R}} \int_{\mathcal{S}^2} \frac{f_{\mathrm{s}}(\mathbf{x}, \omega_{\mathrm{i}} \!\to\! \omega_{\mathrm{o}})}{\eta^2(\mathbf{x}, \omega_{\mathrm{o}})} \left[\eta^2(\mathbf{x}, \omega_{\mathrm{o}})\, d\sigma_{\mathbf{x}}^{\perp}(\omega_{\mathrm{o}}) \right] \\
&= \operatorname*{ess\,sup}_{(\mathbf{x}, \omega_{\mathrm{i}}) \in \mathcal{R}} \int_{\mathcal{S}^2} f_{\mathrm{s}}(\mathbf{x}, \omega_{\mathrm{i}} \!\to\! \omega_{\mathrm{o}})\, d\sigma_{\mathbf{x}}^{\perp}(\omega_{\mathrm{o}})\,.
\end{aligned}$$

Comparing this to the BSDF energy-conservation condition (6.5) derived in Chapter 6, we see that if the scene model uses only physically valid BSDF's, then we are guaranteed that $m \le 1$. Furthermore, real materials will always have at least a small amount of absorption, so that for physically valid scenes we may assume that $m < 1$.[5] This establishes the theorem in the case $p = 1$.

[5] Even for situations such as total internal reflection, or reflection from metals at grazing angles, there will always be some absorption due to tiny imperfections and impurities in the materials.

The case $p = \infty$ is very similar; it is straightforward to show that

$$\|\mathbf{K}'\|'_\infty \quad \leq \quad \operatorname*{ess\,sup}_{(\mathbf{x},\omega_\mathrm{i})\in\mathcal{R}} \int_{\mathcal{S}^2} f'_\mathrm{s}(\mathbf{x},\omega_\mathrm{o}\!\rightarrow\!\omega_\mathrm{i})\, d\sigma^{\perp}_{\mathbf{x}}{}'(\omega_\mathrm{o})$$

.

Notice that this expression is identical to (7.17), except that the directional arguments to the basic BSDF f'_s have been exchanged. Since f'_s is guaranteed to be symmetric for physically valid scenes, we obtain the same bound as for$p = 1$, namely

$$\|\mathbf{K}'\|'_\infty \;\leq\; m \;<\; 1\,.$$

For values of p with $1 < p < \infty$, we use the fact that

$$\|\mathbf{K}'\|'_p \;\leq\; \max\{\|\mathbf{K}'\|'_1, \|\mathbf{K}'\|'_\infty\}\,.$$

This was shown by Arvo [1995, Theorems 12 and 13], whose results apply to any operator of the form $\mathbf{K}$ or $\mathbf{K}'$. ■

From this result and the bound (7.16), we obtain the following:

Corollary 7.7. *For any physically valid scene, and for any*$1 \leq p \leq \infty$*,*

$$\|\mathbf{K}\|_p \;<\; \frac{\eta^2_\mathrm{max}}{\eta^2_\mathrm{min}}\,,$$

where η_min and η_max denote the minimum and maximum refractive indices in the environment.

This was previously stated as Theorem 4.12.

Finally, we can put these results together to show that the solution operators$\mathbf{S}_X$ are well defined, i.e. that the operators $(\mathbf{I} - \mathbf{T}_X)$ are invertible.

Theorem 7.8. *For any physically valid scene, the solution operators*$\mathbf{S}_X$ *exist and are well-defined, where X is any of*L'_i*,* L'_o*,* L_i*,* L_o*,* W_i*, or* W_o*.*

This was previously stated as Theorem 4.13.

Proof. When X is one of L'_i, L'_o, W_i, or W_o, then $\mathbf{T}_X$ is a composition of $\mathbf{K}'$ and $\mathbf{G}$. Therefore

$$\|\mathbf{T}_X\|' \;\leq\; \|\mathbf{K}'\|'\,\|\mathbf{G}\|' \;<\; 1\,,$$

and thus $(\mathbf{I} - \mathbf{T}_X)$ is invertible.

For the cases $X = L_{\rm i}$ and $X = L_{\rm o}$, it is sufficient to show that $\|\mathbf{T}_X^k\| < 1$ for some integer $k \geq 1$. To do this, observe that since $\|\mathbf{T}_X\|' < 1$, there is some integer k such that

$$\|\mathbf{T}_X^k\|' < (\eta_{\rm min}/\eta_{\rm max})^2 \,.$$

Applying the relationship (7.16) between the operator norms$\|\cdot\|$ and $\|\cdot\|'$, we obtain the desired result. ∎

Appendix 7.C The basic BSDF for refraction

In Section 5.2.2 we showed that the BSDF for perfect specular refraction is

$$f_{\mathrm{s}}(\omega_{\mathrm{i}} \to \omega_{\mathrm{t}}) \;=\; \frac{\eta_{\mathrm{t}}^2}{\eta_{\mathrm{i}}^2}\, \delta_{\sigma^\perp}(\omega_{\mathrm{i}} - R(\omega_{\mathrm{t}}))$$

where R is the refraction mapping, and δ_μ is the Dirac distribution with respect to the given measure μ. Thus the corresponding basic BSDF is

$$f_{\mathrm{s}}'(\omega_{\mathrm{i}} \to \omega_{\mathrm{t}}) \;=\; \frac{1}{\eta_{\mathrm{t}}^2} f_{\mathrm{s}}(\omega_{\mathrm{i}} \to \omega_{\mathrm{t}}) \;=\; \frac{1}{\eta_{\mathrm{i}}^2}\, \delta_{\sigma^\perp}(\omega_{\mathrm{i}} - R(\omega_{\mathrm{t}}))\,. \tag{7.18}$$

We have given two separate arguments showing that this quantity is symmetric: the first was a direct mathematical derivation (Appendix 5.B), while the second was based on a general reciprocity principle (Chapter 6).

In this appendix, we show how the basic BSDF for refraction can be rewritten to make its symmetry more obvious. The idea is to rewrite it as a Dirac distribution with respect to the basic projected solid angle measure ($\sigma^{\perp\prime}$). Using the relationship (7.3) between basic and ordinary projected solid angle, and the identity (5.35) between Dirac distributions with respect to different measures, we have

$$f_{\mathrm{s}}'(\omega_{\mathrm{i}} \to \omega_{\mathrm{t}}) \;=\; \frac{1}{\eta_{\mathrm{i}}^2}\, \delta_{\sigma^\perp}(\omega_{\mathrm{i}} - R(\omega_{\mathrm{t}})) \;=\; \delta_{\sigma^{\perp\prime}}(\omega_{\mathrm{i}} - R(\omega_{\mathrm{t}}))\,.$$

The symmetry of this quantity can then be expressed as

$$\delta_{\sigma^{\perp\prime}}(\omega_{\mathrm{i}} - R(\omega_{\mathrm{t}})) \;=\; \delta_{\sigma^{\perp\prime}}(\omega_{\mathrm{t}} - R(\omega_{\mathrm{i}}))\,,$$

which follows from the fact that the mapping R is a bijection, and that it preserves the basic projected solid angle measure (see Section 5.2 and equation (5.36)).

With respect to the angular parameterization (θ, ϕ), the basic BSDF for refraction can be written as

$$f_{\mathrm{s}}'(\theta_{\mathrm{i}}, \phi_{\mathrm{i}}, \theta_{\mathrm{t}}, \phi_{\mathrm{t}}) \;=\; 2\,\delta(\eta_{\mathrm{i}}^2 \sin^2\theta_{\mathrm{i}} - \eta_{\mathrm{t}}^2 \sin^2\theta_{\mathrm{t}})\, \delta(\phi_{\mathrm{i}} - (\phi_{\mathrm{t}} \pm \pi))\,,$$

which follows from equations (5.38), (7.7), and (5.30). In this form, the symmetry of the basic BSDF is clear.

Chapter 8

A Path Integral Formulation of Light Transport

In this chapter, we show how to transform the light transport problem into an integration problem. This *path integral formulation* expresses each measurement in the form of a simple integral (rather than as the solution to an integral equation or operator equation, as with the other formulations we have described). More precisely, each measurement I_j is written in the form

$$I_j \; = \; \int_{\Omega} f_j(\bar{x}) \, d\mu(\bar{x}) \, ,$$

where Ω is the set of *transport paths* of all lengths, μ is a measure on this space of paths, and f_j is called the *measurement contribution function* (to be defined below).

The path integral model has several benefits. The main advantage is that by reducing light transport to an integration problem, it allows general-purpose integration methods to be applied. For example, we will show how light transport problems can be solved more robustly using *multiple importance sampling* (Chapter 9), an integration method that allows several different sampling strategies to be efficiently combined.

The path integral model also leads to new techniques for sampling paths. The problem with models based on integral equations is that they only describe scattering from one surface at a time. This leads to light transport algorithms that construct paths incrementally, by recursive sampling of the integral equation. The path integral model takes a more global view, which has led directly to techniques such as bidirectional path tracing (Chapter 10) and the Metropolis light transport algorithm (Chapter 11). These new techniques can only be properly understood within the path integral framework.

Finally, the path integral model is a useful tool for understanding the limitations of unbiased Monte Carlo algorithms. It provides a natural way to classify transport paths, and to identify those that cannot be sampled by certain kinds of techniques.

This chapter is organized as follows. First, we review the *three-point form* of the light transport equations, and show how to transform them into an integral over paths. We then discuss the advantages of the path integral model in more detail, and show how it can be used to construct unbiased Monte Carlo estimators. Finally, introduce the idea of *full-path regular expressions* (extending a notation of Heckbert [1990]), and discuss the limitations of path sampling approaches to light transport.

In Appendix 8.A, we describe several other ways that the path integral model can be formulated, by introducing new measures on the space of paths. These measures have natural physical interpretations whose meanings are described.

8.1 The three-point form of the transport equations

We show how to rewrite the transport equations to eliminate the directional variables $\omega_{\rm i}, \omega_{\rm o}$. This first step is to write the equilibrium radiance in the form $L(\mathbf{x} \to \mathbf{x}')$, where $\mathbf{x}, \mathbf{x}' \in \mathcal{M}$ are points on the scene surfaces. In terms of the function $L(\mathbf{x}, \omega)$ we have been using up until now, we define

$$L(\mathbf{x} \to \mathbf{x}') \;=\; L(\mathbf{x}, \omega)$$

where $\omega \;=\; \widehat{\mathbf{x}' - \mathbf{x}}$ is the unit-length vector pointing from $\mathbf{x}$ to $\mathbf{x}'$. (This representation of the ray space $\mathcal{R}$ was described in Section 4.1; recall that it has some redundancy, since $L(\mathbf{x} \to \mathbf{x}') = L(\mathbf{x} \to \mathbf{x}'')$ whenever $\mathbf{x}'$ and $\mathbf{x}''$ lie in the same direction from $\mathbf{x}$.)

Similarly, we write the BSDF as a function of the form

$$f_{\rm s}(\mathbf{x} \to \mathbf{x}' \to \mathbf{x}'') \;=\; f_{\rm s}(\mathbf{x}', \omega_{\rm i} \to \omega_{\rm o})\,,$$

where $\omega_{\rm i} = \widehat{\mathbf{x} - \mathbf{x}'}$ and $\omega_{\rm o} = \widehat{\mathbf{x}'' - \mathbf{x}'}$. The arrow notation $\mathbf{x} \to \mathbf{x}'$ symbolizes the direction

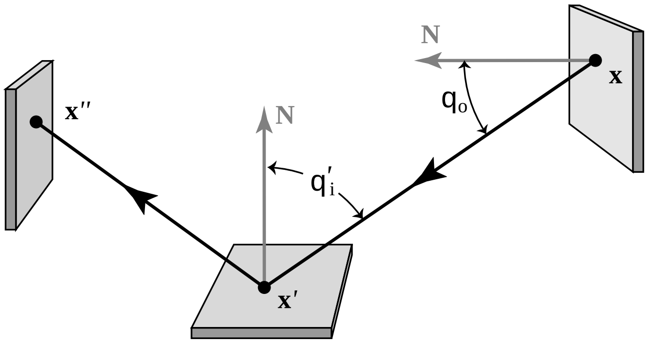


Figure 8.1: Geometry for the light transport equation in three-point form.

of light flow.

The *three-point form* of the light transport equation can now be written as

$$L(\mathbf{x}' \to \mathbf{x}'') \;=\; L_{\mathrm{e}}(\mathbf{x}' \to \mathbf{x}'') + \int_{\mathcal{M}} L(\mathbf{x} \to \mathbf{x}')\, f_{\mathrm{s}}(\mathbf{x} \to \mathbf{x}' \to \mathbf{x}'')\, G(\mathbf{x} \leftrightarrow \mathbf{x}')\, dA(\mathbf{x}) \qquad (8.1)$$

(see Figure 8.1). This is simply a reformulation of the original version of the light transport equation (3.19) that we have already described. As before, $\mathcal{M}$ is the union of all scene surfaces, A is the area measure on $\mathcal{M}$, and L_{e} is the emitted radiance function. The function G represents the change of variables from the original integration measure $d\sigma^{\perp}$ to the new integration measure dA, which are related by

$$d\sigma^{\perp}_{\mathbf{x}'}(\omega_{\mathrm{i}}) \;=\; d\sigma^{\perp}_{\mathbf{x}'}(\widehat{\mathbf{x} - \mathbf{x}'}) \;=\; G(\mathbf{x} \leftrightarrow \mathbf{x}')\, dA(\mathbf{x})\,, \qquad (8.2)$$

where

$$G(\mathbf{x} \leftrightarrow \mathbf{x}') \;=\; V(\mathbf{x} \leftrightarrow \mathbf{x}')\, \frac{|\cos(\theta_{\mathrm{o}})\; \cos(\theta'_{\mathrm{i}})|}{\|\mathbf{x} - \mathbf{x}'\|^2}\,. \qquad (8.3)$$

Here θ_{o} and θ'_{i} are the angles between the segment $\mathbf{x} \leftrightarrow \mathbf{x}'$ and the surface normals at $\mathbf{x}$ and $\mathbf{x}'$ respectively, while $V(\mathbf{x} \leftrightarrow \mathbf{x}') = 1$ if $\mathbf{x}$ and $\mathbf{x}'$ are mutually visible and is zero otherwise.

We also use the change of variables (8.2) to rewrite the original measurement equation (3.18) as

$$I_j \;=\; \int_{\mathcal{M} \times \mathcal{M}} W_{\mathrm{e}}^{(j)}(\mathbf{x} \to \mathbf{x}')\, L(\mathbf{x} \to \mathbf{x}')\, G(\mathbf{x} \leftrightarrow \mathbf{x}')\, dA(\mathbf{x})\, dA(\mathbf{x}')\,, \qquad (8.4)$$

where as usual, the notation $\mathbf{x} \to \mathbf{x}'$ indicates the direction of light flow. In particular, $W_{\text{e}}^{(j)}(\mathbf{x} \to \mathbf{x}')$ represents the importance that is emitted from $\mathbf{x}'$ toward $\mathbf{x}$ (opposite to the arrow notation). This is, we define $W_{\text{e}}^{(j)}(\mathbf{x} \to \mathbf{x}') = W_{\text{e}}^{(j)}(\mathbf{x}', \omega)$, where $\omega = \widehat{\mathbf{x} - \mathbf{x}'}$.[1]

8.2 The path integral formulation

In this section, we first define the components of the path integral formulation: the integration domain, measure, and integrand. Next, we discuss the advantages of this formulation. Finally, we show how to use the path integral framework in Monte Carlo algorithms, and in particular how to calculate the probability densities with which paths are sampled.

Recall that our goal is to express each measurement in the form

$$I_j \;=\; \int_\Omega f_j(\bar{x})\, d\mu(\bar{x})\,. \tag{8.5}$$

To do this, let Ω_k represent the paths of length k, i.e. the set of paths of the form

$$\bar{x} \;=\; \mathbf{x}_0\, \mathbf{x}_1\, \ldots\, \mathbf{x}_k\,,$$

where $1 \le k < \infty$ and $\mathbf{x}_i \in \mathcal{M}$ for each i. We define a measure μ_k on this set of paths, called the *area-product measure,* according to

$$\mu_k(D) \;=\; \int_D dA(\mathbf{x}_0)\, \cdots\, dA(\mathbf{x}_k)\,,$$

where $D \subset \Omega_k$ is a set of paths. Formally, μ_k is a product measure [Halmos 1950]; we could also have written its definition as

$$\begin{aligned} d\mu_k(\mathbf{x}_0 \ldots \mathbf{x}_k) &\;=\; dA(\mathbf{x}_0)\, \cdots\, dA(\mathbf{x}_k)\,, \\ \text{or} \qquad \mu_k &\;=\; \underbrace{A \times \cdots \times A}_{k \text{ times}}\,. \end{aligned}$$

[1] Notice that the visibility factor $V(\mathbf{x} \leftrightarrow \mathbf{x}')$ hidden in the function G is essential, since $L(\mathbf{x} \to \mathbf{x}')$ refers to the radiance leaving $\mathbf{x}$, while $W_{\text{e}}^{(j)}(\mathbf{x} \to \mathbf{x}')$ applies to the radiance arriving at $\mathbf{x}'$. To put this another way, L and $W_{\text{e}}^{(j)}$ are both exitant quantities, since $W_{\text{e}}^{(j)}$ specifies the importance leaving $\mathbf{x}'$, rather than the importance arriving at $\mathbf{x}$.

Next, we define the *path space* Ω as

$$\Omega \;=\; \bigcup_{k=1}^{\infty} \Omega_k \,,$$

i.e. Ω represents the set of paths of all finite lengths. We extend the area-product measure μ to this space in the natural way, by letting

$$\mu(D) \;=\; \sum_{k=1}^{\infty} \mu_k(D \cap \Omega_k)\,. \tag{8.6}$$

That is, the measure of a set of paths is simply the sum of the measures of the paths of each length.[2]

To complete the definition of the path integral formulation (8.5), we must define the integrand f_j. To do this, we start with the measurement equation (8.4), and recursively expand the transport equation (8.1) to obtain

$$\begin{aligned} I_j \;&=\; \sum_{k=1}^{\infty} \int_{\mathcal{M}^{k+1}} L_{\mathrm{e}}(\mathbf{x}_0 \!\to\! \mathbf{x}_1)\, G(\mathbf{x}_0 \!\leftrightarrow\! \mathbf{x}_1) \prod_{i=1}^{k-1} f_{\mathrm{s}}(\mathbf{x}_{i-1} \!\to\! \mathbf{x}_i \!\to\! \mathbf{x}_{i+1})\, G(\mathbf{x}_i \!\leftrightarrow\! \mathbf{x}_{i+1}) \\ &\qquad\qquad \cdot W_{\mathrm{e}}^{(j)}(\mathbf{x}_{k-1} \!\to\! \mathbf{x}_k)\; dA(\mathbf{x}_0) \cdots dA(\mathbf{x}_k) \qquad (8.7) \\ &=\; \int_{\mathcal{M}^2} L_{\mathrm{e}}(\mathbf{x}_0 \!\to\! \mathbf{x}_1)\, G(\mathbf{x}_0 \!\leftrightarrow\! \mathbf{x}_1)\, W_{\mathrm{e}}^{(j)}(\mathbf{x}_0 \!\to\! \mathbf{x}_1)\; dA(\mathbf{x}_0)\, dA(\mathbf{x}_1) \\ &+\; \int_{\mathcal{M}^3} L_{\mathrm{e}}(\mathbf{x}_0 \!\to\! \mathbf{x}_1)\, G(\mathbf{x}_0 \!\leftrightarrow\! \mathbf{x}_1)\, f_{\mathrm{s}}(\mathbf{x}_0 \!\to\! \mathbf{x}_1 \!\to\! \mathbf{x}_2)\, G(\mathbf{x}_1 \!\leftrightarrow\! \mathbf{x}_2) \\ &\qquad\qquad \cdot W_{\mathrm{e}}^{(j)}(\mathbf{x}_1 \!\to\! \mathbf{x}_2)\; dA(\mathbf{x}_0)\, dA(\mathbf{x}_1)\, dA(\mathbf{x}_2) \\ &+\; \cdots\,. \end{aligned}$$

The integrand f_j is defined for each path length k separately, by extracting the appropriate term from the expansion (8.7). For example, given a path $\bar{x} = \mathbf{x}_0\mathbf{x}_1\mathbf{x}_2\mathbf{x}_3$, we have

$$\begin{aligned} f_j(\bar{x}) \;&=\; L_{\mathrm{e}}(\mathbf{x}_0 \!\to\! \mathbf{x}_1)\, G(\mathbf{x}_0 \!\leftrightarrow\! \mathbf{x}_1)\, f_{\mathrm{s}}(\mathbf{x}_0 \!\to\! \mathbf{x}_1 \!\to\! \mathbf{x}_2) \\ &\qquad \cdot G(\mathbf{x}_1 \!\leftrightarrow\! \mathbf{x}_2)\, f_{\mathrm{s}}(\mathbf{x}_1 \!\to\! \mathbf{x}_2 \!\to\! \mathbf{x}_3)\, G(\mathbf{x}_2 \!\leftrightarrow\! \mathbf{x}_3)\, W_{\mathrm{e}}^{(j)}(\mathbf{x}_2 \!\to\! \mathbf{x}_3) \end{aligned}$$

(see Figure 8.2). This function f_j is called the *measurement contribution function.*

[2] This measure on paths is similar to that of Spanier & Gelbard [1969, p. 85]. However, in our case the path space Ω does not include any infinite-length paths. This makes it easy to verify that (8.6) is in fact a measure, directly from the axioms [Halmos 1950].

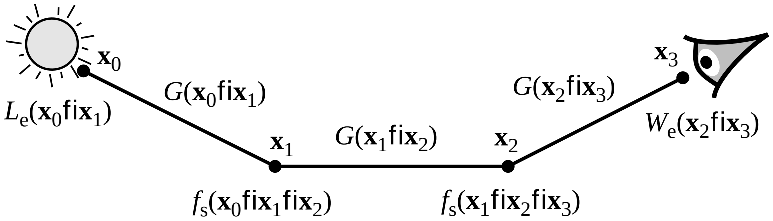

$\mathbf{x}_0$
$L_e(\mathbf{x}_0\text{fi}\mathbf{x}_1)$
$G(\mathbf{x}_0\text{fi}\mathbf{x}_1)$
$\mathbf{x}_1$
$G(\mathbf{x}_1\text{fi}\mathbf{x}_2)$
$\mathbf{x}_2$
$G(\mathbf{x}_2\text{fi}\mathbf{x}_3)$
$\mathbf{x}_3$
$W_e(\mathbf{x}_2\text{fi}\mathbf{x}_3)$
$f_s(\mathbf{x}_0\text{fi}\mathbf{x}_1\text{fi}\mathbf{x}_2)$
$f_s(\mathbf{x}_1\text{fi}\mathbf{x}_2\text{fi}\mathbf{x}_3)$

Figure 8.2: The measurement contribution function f_j is a product of many factors (shown for a path of length 3).

We have now defined all the terms of path integral model (8.5): the integration domain, integrand, and measure. There is nothing particularly complicated about this transformation; we have just expanded and rearranged the transport equations. The most significant aspect is that we have removed the sum over different path lengths, and replaced it with a single integral over an abstract measure space of paths.

8.2.1 Advantages of the path integral formulation

The path integral formulation has several advantages. First, the expression for each measurement has the form of an integral (as opposed to some other mathematical object). This allows us to derive new rendering algorithms by applying general-purpose integration techniques, such as multiple importance sampling (Chapter 9).

Second, the path integral model has a much simpler structure: a single expression defines the value of each measurement. In contrast, the integral equation approach requires two equations (the light transport and measurement equations), one of which is defined recursively. With the path integral approach, there are no adjoint equations, no intermediate quantities such as light or importance, and no need to choose between these alternatives. Measurements are defined and computed directly, by organizing the calculations around a *geometric* primitive (the path), rather than radiometric quantities.

By dealing with whole paths rather than rays, the path integral framework also provides a more explicit and complete description of light transport. Each path specifies the emission, scattering, and measurement events along a complete photon trajectory. On the other hand, integral equations describe the scattering events in isolation, by specifying the interaction of light with each surface separately.

This has practical consequences for sampling paths: the natural strategy for solving an integral equation is to sample the equation recursively, leading to paths that are built starting entirely from the lens, or entirely from a light source (depending on whether the light transport equation or its adjoint is sampled). With the path integral approach, on the other hand, it is possible to construct paths in arbitrary ways, e.g. by starting with a vertex in the middle, and building the path outwards in both directions. This leads directly to sampling strategies such as bidirectional path tracing (Chapter 10), and the Metropolis algorithm (Chapter 11).

Furthermore, the path integral approach gives a convenient framework for computing probability densities on paths (as described in the next section). This allows us to easily compare the probabilities with which a given path is sampled by different techniques. This is an essential prerequisite for the use of the multiple importance sampling and Metropolis techniques.

8.2.2 Applying the path integral formulation

In this section, we explain how the path integral framework can be used in Monte Carlo algorithms. We first show how measurements can be estimated, by randomly generating transport paths $\bar{X}$, and computing an estimate of the form $f_j(\bar{X})/p(\bar{X})$. This requires the evaluation of the probability density $p(\bar{X})$ with which each path was sampled. We consider how to do this within the framework of *local path sampling*, which is general enough to describe virtually all unbiased path sampling algorithms that are used in practice.

Our goal is to estimate the path integral

$$I_j \;=\; \int_\Omega f_j(\bar{x})\, d\mu(\bar{x})$$

for each measurement I_j. To do this, the natural Monte Carlo strategy is to first sample a random path $\bar{X}$ according to some chosen density function p, and then compute an estimate of the form

$$I_j \;\approx\; \frac{f_j(\bar{X})}{p(\bar{X})}\,. \tag{8.8}$$

This is an unbiased estimate of the measurement I_j, since its expected value is

$$\begin{aligned} E\left[\frac{f_j(\bar{X})}{p(\bar{X})}\right] &= \int_\Omega \frac{f_j(\bar{x})}{p(\bar{x})}\, p(\bar{x})\, d\mu(\bar{x}) \\ &= \int_\Omega f_j(\bar{x})\, d\mu(\bar{x}) \\ &= I_j\,, \end{aligned} \tag{8.9}$$

where we have assumed that p is measured with respect to the area-product measure μ, in order for the first line of this equation to hold.

To apply this strategy, we must be able to evaluate the functions f_j and p for the given path $\bar{X}$. An explicit formula for the measurement contribution function f_j has already been given; thus, the main question is how to evaluate the probability density $p(\bar{X})$. Obviously, this depends not only on the particular path $\bar{X}$, but also on how this path was generated. For example, one way to generate paths is with ordinary path tracing: the vertex $\mathbf{x}_k$ is chosen on the lens, and subsequent vertices $\mathbf{x}_{k-1}$, . . ., $\mathbf{x}_1$ are generated by following random bounces backward, until eventually we connect the path to a random vertex $\mathbf{x}_0$ on a light source. The probability $p(\bar{X})$ depends on all of the random choices made during this process, as we will discuss in more detail below.

8.2.2.1 Local path sampling

We will concentrate on a particular family of methods for generating paths, called *local path sampling algorithms*. These methods generate vertices one at a time, based on local information at existing vertices (such as the BSDF). There are three basic mechanisms that can be used to construct paths in this framework:

- A vertex can be chosen according some *a priori* distribution over the scene surfaces. For example, this can be used to sample a vertex on a light source, with a probability density proportional to the radiant exitance (i.e. the power per unit area emitted over the light source). Similarly, this technique can be used to sample the initial vertex on a finite-aperture lens. It can also be used to sample intermediate vertices along the path, e.g. to sample a vertex on a window between two adjacent rooms.

- The second method for generating a vertex is to sample a direction according to a locally defined probability distribution at an existing vertex $\mathbf{x}$, and then cast a ray to find the first surface intersection $\mathbf{x}'$ (which becomes the new vertex). For example, this is what happens when the BSDF at an existing vertex is sampled (or an approximation to the BSDF). This mechanism can also used to sample a direction for emission, once a vertex on a light source has been chosen.
- The third mechanism for path sampling is to connect two existing vertices, by checking the visibility between them. In effect, this step verifies the existence of an *edge* between two vertices, rather than generating a new vertex.

By combining these three simple techniques, it is possible to sample paths in a great variety of ways. Subpaths can be built up starting from the light sources, the lens, or from an arbitrary scene surface. These subpaths can then be joined together to create a full path from a light source to the lens. This local sampling framework is general enough to accommodate virtually all path sampling techniques that are used in practice.[3]

8.2.2.2 Computing the path probabilities

In this section, we describe how to compute the probability density $p(\bar{x})$ for sampling a given path $\bar{x}$. As mentioned above (equation (8.9)), we wish to compute the probability density with respect to the area-product measure μ, that is:

$$p(\bar{x}) \;=\; \frac{dP}{d\mu}(\bar{x})\,.$$

Given a path $\bar{x} \;=\; \mathbf{x}_0 \ldots \mathbf{x}_k$, this expands to

$$\begin{aligned} p(\bar{x}) \;&=\; \frac{dP}{d\mu}(\mathbf{x}_0 \ldots \mathbf{x}_k) \\ &=\; \prod_{i=0}^{k} \frac{dP}{dA}(\mathbf{x}_i)\,. \end{aligned}$$

[3] As an example of a non-local sampling technique, suppose that the location of a new vertex is computed by solving an algebraic equation involving two or more existing vertices. For example, this could be used to determine the point $\mathbf{y}$ on a curved mirror that reflects light from a given vertex $\mathbf{x}$ to another vertex $\mathbf{x}'$. This is not allowed in the local path sampling framework, since the position of $\mathbf{y}$ depends on more than one existing vertex. This type of non-local sampling will be discussed further in Section 8.3.4.

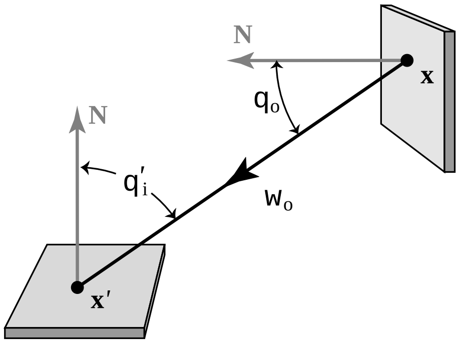


Figure 8.3: Geometry for converting between area and directional probabilities.

Thus to evaluate $p(\bar{X})$, we must compute the probability per unit area (dP/dA) with which each vertex $\mathbf{x}_i$ was generated, and multiply them together.

We now consider how to compute the probability for sampling a given vertex. According to the local path sampling model, each vertex $\mathbf{x}_i$ can be generated according to one of two methods: either $\mathbf{x}_i$ is sampled from a distribution over the scene surfaces (in which the probability density $dP/dA(\mathbf{x}_i)$ can be computed directly), or else it is generated by casting a ray from an existing vertex, in a randomly chosen direction.

To calculate the density in the latter case, let $\mathbf{x}$ be the existing vertex, and let $\mathbf{x}' = \mathbf{x}_i$ be the new vertex. We assume that $\mathbf{x}'$ was generated by casting a ray from $\mathbf{x}$ in the direction ω_o, where

$$\omega_o \;=\; \widehat{\mathbf{x}' - \mathbf{x}}$$

(see Figure 8.3). We are also given the probability density $p(\omega_o)$ with which ω_o was chosen (measured with respect to solid angle). To compute the density $p(\mathbf{x}')$ with respect to surface area, we must express it in terms of the given density $p(\omega_o)$. These two densities are related by

$$\begin{aligned} \frac{dP}{dA}(\mathbf{x}') \;&=\; \frac{dP}{d\sigma}(\omega_o)\,\frac{d\sigma(\omega_o)}{dA(\mathbf{x}')} \\ \Longrightarrow \qquad p(\mathbf{x}') \;&=\; p(\omega_o)\left(\frac{|\cos(\theta'_i)|}{\|\mathbf{x} - \mathbf{x}'\|^2}\right) \qquad (8.10) \end{aligned}$$

(see Figure 8.3). The parenthesized expression is the solid angle subtended at $\mathbf{x}$ per unit of surface area at $\mathbf{x}'$.

Using these rules, it is straightforward to compute the probability density $p(\bar{x})$ for the whole path. We simply consider the vertices in the order that they were generated, and multiply together the densities dP/dA for each vertex (converting from directional to area probabilities as necessary). There are few restrictions on how the paths are generated: starting from the lens (as with path tracing), starting from the lights (as with particle tracing), or a combination of both (as with bidirectional path tracing). Paths can also be constructed starting from the middle, by sampling vertices according to predefined distributions over the scene surfaces: this could be useful in difficult geometric settings, e.g. to generate transport paths that pass through a known small portal.

In the path integral framework, all of these possibilities are handled in the same way. They are viewed as different sampling strategies for the measurement equation (8.5), leading to different probability distributions on the space of paths. They are unified under one simple equation, namely the estimate $f_j(\bar{X})/p(\bar{X})$.

Densities with respect to projected solid angle. In many cases, it is more natural and convenient to represent directional distributions as densities with respect to projected solid angle $\sigma^\perp$ (rather than ordinary solid angle σ). We summarize the equations here for future reference.

Given an existing vertex $\mathbf{x}$ (Figure 8.3), let $p(\omega_\mathrm{o})$ and $p^\perp(\omega_\mathrm{o})$ be the probability densities with respect to ordinary and projected solid angle respectively for sampling the given direction ω_o. These two densities are related by

$$\begin{array}{rrcl} & \dfrac{dP}{d\sigma^\perp}(\omega_\mathrm{o}) & = & \dfrac{dP}{d\sigma}(\omega_\mathrm{o})\, \dfrac{d\sigma(\omega_\mathrm{o})}{d\sigma^\perp(\omega_\mathrm{o})} \\ \Longrightarrow \qquad & p^\perp(\omega_\mathrm{o}) & = & p(\omega_\mathrm{o})\, \dfrac{1}{\cos(\theta_\mathrm{o})}\,, \end{array} \tag{8.11}$$

where we have used the relationship

$$d\sigma^\perp(\omega_\mathrm{o}) \;=\; |\omega_\mathrm{o} \cdot \mathbf{N}(\mathbf{x})|\, d\sigma(\omega_\mathrm{o})\,.$$

Putting this together with equation (8.10), we can convert between densities with respect to projected solid angle and densities with respect to surface area using

$$\begin{aligned} p(\mathbf{x}') &= p^{\perp}(\omega_{\mathrm{o}}) \, \frac{|\cos(\theta_{\mathrm{o}}) \, \cos(\theta'_{\mathrm{i}})|}{\|\mathbf{x} - \mathbf{x}'\|^2} \\ &= p^{\perp}(\widehat{\mathbf{x}' - \mathbf{x}}) \, G(\mathbf{x} \leftrightarrow \mathbf{x}') \, , \end{aligned}$$

where G is the geometric factor (8.2).[4] Notice that this conversion factor is symmetric, unlike the conversion factor (8.10) for densities with respect to ordinary solid angle.

8.3 The limitations of path sampling

Although algorithms based on path sampling tend to be simple and general, they do have limits. For example, if point light sources and perfect mirrors are allowed, then there are some types of transport paths that cannot be sampled at all. Images computed by path sampling algorithms will be missing the contributions made by these paths. As a typical example of this problem, consider a scene where a point light source reflects off a mirror, creating caustics on a diffuse surface. Although algorithms such as bidirectional path tracing are capable of rendering these caustics when viewed directly, they will fail if the caustics are viewed indirectly through a second mirror. (The indirectly viewed caustics will simply be missing from the image.)

More generally, there are some light transport problems that are provably difficult for any algorithm. In this regard, it has been shown that some ray tracing problems are *undecidable*, i.e. they cannot be solved on a Turing machine [Reif et al. 1994]. These examples are not physically realizable, since they rely on perfect mirrors and infinite geometric precision. However, we can expect that as the geometry and materials of the input scene approach a provably difficult configuration, any light transport algorithm will perform very badly.

Our goals in this section are more practical. We are mainly concerned with the limitations of *local* path sampling algorithms, as described in Section 8.2.2.1. For this type of algorithm, problems are caused not only by mirrors and point sources, but also by refraction,

[4] Note that the visibility term $V(\mathbf{x} \leftrightarrow \mathbf{x}')$ hidden in G is required only when the visibility between $\mathbf{x}$ and $\mathbf{x}'$ is not known.

perfectly anisotropic surfaces, parallel light sources, pinhole lenses, and orthogonal viewing projections. Our goal is to determine which combinations of these features can cause local path sampling algorithms to fail.

We start by reviewing Heckbert's regular expression notation for paths. Next, we show how to extend this notation to describe the properties of light sources and sensors, in order to allow features such as point light sources and orthographic lenses to be represented in a compact and consistent way. We then give a criterion for determining which types of paths cannot be generated by local path sampling. Finally, we consider some ways to lift this restriction using non-local sampling methods.

8.3.1 Heckbert's regular expression notation for paths

Heckbert [1990] introduced a useful notation for classifying paths by means of regular expressions. Originally, it was used to describe the capabilities of multi-pass global illumination algorithms, e.g. algorithms that combine radiosity and ray tracing. In this context, it was assumed that all BSDF's can be written as a linear combination of an ideal diffuse component and an ideal specular component. For example, a typical surface might reflect 50% of the incident light diffusely, reflect 10% in a mirror-like fashion, and absorb the rest.

Paths are then described using regular expressions of the form[5]

$$L\,(S|D)^*\,E\,.$$

Each symbol represents one vertex of a path: L denotes the first vertex of the path, which lies on a light source, while E denotes the last vertex (the camera position or "eye"). The remaining vertices are classified as S or D, according to whether the light was reflected by the specular or diffuse component of the surface respectively. Note that the symbols S and D represent the type of the *scattering event* at each vertex, not the type of the surface, since the surface itself is allowed to be a combination of specular and diffuse.

[5] In regular expressions, X^+ denotes one or more occurrences of X, X^* denotes zero or more occurrences of X, $X|Y$ denotes a choice between X or Y, ϵ denotes the empty string, and parentheses are used for grouping.

Definitions for general materials. This notation is easily extended to scenes with general materials, by redefining the symbols S and D appropriately. We show how to make these definitions rigorously, by relating them to the BSDF.

Let $\bar{x} = \mathbf{x}_0 \ldots \mathbf{x}_k$ be a path, and consider the scattering event at a vertex $\mathbf{x}_i$ (where $0 < i < k$). For general materials, we let the symbol D represent any scattering event where the BSDF is finite, i.e. where

$$f_{\mathrm{s}}(\mathbf{x}_{i-1} \to \mathbf{x}_i \to \mathbf{x}_{i+1}) < \infty .$$

All other scattering events (where the BSDF is not finite) are denoted by the symbol S. This category includes not only pure specular reflection and refraction, where light is scattered in a zero-dimensional set of directions, but also pure anisotropic scattering, where light is scattered in a one-dimensional set of directions (similar to the reflection properties of brushed aluminum). These possibilities will be discussed in more detail below.

8.3.2 Full-path regular expressions

Heckbert's notation describes only the scattering events along a path. We show how to extend these regular expressions in a natural way, to describe the properties of light sources and sensors as well.

Each light source is classified according to a two-letter combination, of the form $(S|D)\,(S|D)$. The first letter represents the surface area of the light source: D denotes a finite-area source, while S denotes a source with zero area (e.g. a point or linear source). The second letter represents the directional properties of the emission: D denotes emission over a finite solid angle, while S denotes emission over a set of angles with measure zero. Thus, a point light source that radiates light in all directions would be denoted by the regular expression LSD. Note that unlike Heckbert's notation, the symbol L does not represent a real vertex; it is simply a placeholder that indicates the ordering of vertices (i.e. the fact that the first vertex is on a light source rather than a sensor).

Similarly, to represent the properties of the sensor we use a suffix of the form

$$(S|D)\,(S|D)\,E\,.$$

LDD	a diffusely emitting sphere
LDS	sunlight shining through a window, where the window itself is modeled as the light source
LSD	a point spotlight
LSS	a laser beam
DDE	a finite-aperture lens
SDE	an orthographic projection lens (where the image plane located within the scene, rather than at infinity)
DSE	a pinhole lens
SSE	an idealized spot meter (which measures radiance along a single given ray)

Table 8.1: Examples of regular expressions that approximate various kinds of real light sources and sensors (e.g. by treating the sun as a point at infinity, etc.)

The first letter represents the directional sensitivity of the sensor, i.e. whether it is sensitive to light over a finite solid angle (D), or to light that arrives from a set of directions with measure zero (S). The second letter represents the surface area of the sensor, with the same conventions used for the first letter of the light source classification.

Table 8.1 gives some examples of light sources and lens models which are good approximations to the various letter combinations (e.g. if we treat the sun as a point source at infinity).

Combining this notation for light sources and sensors with Heckbert's notation for scattering events, an entire path is thus described by a regular expression such as

$$L\,D\,D\,S^*\,D\,D\,E\,.$$

This example represents a path that starts on an ordinary area light source, is scattered by zero or more specular surfaces, and terminates at an ordinary finite-aperture lens. This extended notation is called a *full-path regular expression*.

The main advantage of full-path expressions is that they give a compact way to describe the paths generated by specific sampling strategies. For this purpose, it is essential to specify the properties of the light source and sensor, since some strategies do not work for sources or sensors with zero area, or those that emit or measure light over a zero solid angle. (For example, “pure” path tracing cannot handle point light sources, since they will never be intersected by a path that is randomly generated starting from the lens.) We will make extensive use of full-path expressions to describe the sampling strategies of bidirectional path tracing and Metropolis light transport, and also to investigate the limitations of local path sampling.

Formal definitions of the full-path notation. Full-path regular expressions can be defined more rigorously in the following way. First, we show how to split the emitted radiance function L_{e} into a product of two factors $L_{\mathrm{e}}^{(0)}$ and $L_{\mathrm{e}}^{(1)}$, which represent the spatial and directional components of the emission respectively. The factor $L_{\mathrm{e}}^{(0)}$ is defined by

$$L_{\mathrm{e}}^{(0)}(\mathbf{x}) \;=\; \int_{\mathcal{S}^2} L_{\mathrm{e}}(\mathbf{x}, \omega)\, d\sigma^{\perp}(\omega)\,, \tag{8.12}$$

and represents the *radiant exitance* (emitted power per unit area) associated with a point $\mathbf{x}$ on a light source. The second factor $L_{\mathrm{e}}^{(1)}$ is given by

$$L_{\mathrm{e}}^{(1)}(\mathbf{x}, \omega) \;=\; L_{\mathrm{e}}(\mathbf{x}, \omega) / L_{\mathrm{e}}^{(0)}(\mathbf{x})\,, \tag{8.13}$$

and represents the directional distribution of the emitted radiance at $\mathbf{x}$. These factors correspond to the fact that sampling for emission is naturally subdivided into two steps, consisting of first choosing a point on a light source, and then a direction for the emitted ray. Notice that by definition,

$$\int_{\mathcal{S}^2} L_{\mathrm{e}}^{(1)}(\mathbf{x}, \omega)\, d\sigma^{\perp}(\omega) \;=\; 1\,,$$

so that $L_{\mathrm{e}}^{(1)}$ is simply the probability density function for ω, for a given choice of $\mathbf{x}$.

With these definitions, the light source notation LXY has the following meaning:

$$X \;=\; \begin{cases} D & \text{if } L_{\mathrm{e}}^{(0)}(\mathbf{x}_0) < \infty \\ S & \text{otherwise}\,, \end{cases}$$

$$Y \;=\; \begin{cases} D & \text{if } L_{\mathrm{e}}^{(1)}(\mathbf{x}_0 \!\rightarrow\! \mathbf{x}_1) < \infty \\ S & \text{otherwise}\,. \end{cases}$$

Likewise, we can rigorously define the meaning of the notation YXE for sensors. This is done by splitting the emitted importance function W_{e} into a product of two factors $W_{\mathrm{e}}^{(0)}$ and $W_{\mathrm{e}}^{(1)}$, and making a definition similar to the one for LXY.

Thus far, we have only distinguished between light that is emitted or scattered in a two-dimensional set of directions (D), vs. all other cases (S). It is sometimes useful to classify the S vertices further, according to whether light is scattered in a zero- or one-dimensional set of directions (S_0 vs. S_1). This extended notation is discussed in Appendix 8.B, and can be used to describe the properties of light sources, sensors, and materials more precisely.

Note that Langer & Zucker [1997] have independently proposed a classification system for light sources that is similar to the one described here. However, they do not attempt to give a general definition of their classification scheme, they do not develop any notation for it, and they do not consider the classification of sensors or scattering events.

8.3.2.1 Interpreting sources and sensors as scattering events

The definitions above are somewhat cumbersome to use, because sources and sensors are treated as special cases. In other words, the first two $(S|D)$ symbols and the last two $(S|D)$ symbols of each path cannot be handled in the same way as the rest, since they represent emission and measurement rather than scattering. It would be easier to reason about these regular expressions if the S and D symbols had a consistent meaning.

In this section, we show how the S and D symbols describing light sources and sensors can be interpreted as "scattering events" in a natural way. To do this, we introduce an imaginary vertex at each end of the path, and extend the definition of the BSDF to describe light transport to and from these imaginary vertices. With these changes, all of the symbols in a full-path regular expression have a consistent interpretation, so that the special cases associated with sources and sensors can be avoided.

The conversion from emission to scattering is described in two steps. We first consider the directional component of the emission, and then the spatial component.

Scattering events at $\mathbf{x}_0$ and $\mathbf{x}_k$. We show how the directional components of the emission functions L_{e} and W_{e} can be interpreted as scattering at the vertices $\mathbf{x}_0$ and $\mathbf{x}_k$. To do this, we introduce two imaginary vertices Ψ_L and Ψ_W, which become the new path endpoints.

A complete path thus has the form

$$\Psi_L\, \mathbf{x}_0\, \mathbf{x}_1\, \ldots\, \mathbf{x}_k\, \Psi_W\,,$$

where the vertices Ψ_L and Ψ_W always occur at positions $\mathbf{x}_{-1}$ and $\mathbf{x}_{k+1}$ respectively.

We regard the vertex Ψ_L as the source of all light, while Ψ_W is the source of all importance. That is, rather than allowing surfaces to emit light directly, we assume that emission occurs only at the vertex Ψ_L. Light is emitted along imaginary rays of the form $\Psi_L \to \mathbf{x}$, and is then scattered at $\mathbf{x}$ into physical rays of the form $\mathbf{x} \to \mathbf{x}'$. This process is defined so that we obtain the same results as the original emission function L_{e}. Similarly, all sensor measurements are made at the point Ψ_W. This corresponds to the following symbolic definitions:

$$\begin{aligned}
L_{\mathrm{e}}(\Psi_L \!\to\! \mathbf{x}) &= L_{\mathrm{e}}^{(0)}(\mathbf{x})\,, \\
f_{\mathrm{s}}(\Psi_L \!\to\! \mathbf{x} \!\to\! \mathbf{x}') &= L_{\mathrm{e}}^{(1)}(\mathbf{x} \!\to\! \mathbf{x}')\,, \\
f_{\mathrm{s}}(\mathbf{x}' \!\to\! \mathbf{x} \!\to\! \Psi_W) &= W_{\mathrm{e}}^{(1)}(\mathbf{x}' \!\to\! \mathbf{x})\,, \\
W_{\mathrm{e}}(\mathbf{x} \!\to\! \Psi_W) &= W_{\mathrm{e}}^{(0)}(\mathbf{x})\,,
\end{aligned}$$

where $L_{\mathrm{e}}^{(i)}$ and $W_{\mathrm{e}}^{(i)}$ are the spatial and directional components of emission (8.12, 8.13).

Scattering events at Ψ_L and Ψ_W. We now show how the spatial components of emission can be interpreted as scattering at the imaginary vertices Ψ_L and Ψ_W. To do this, we assume that the emitted light is initially concentrated on the single imaginary ray $\Psi_L \to \Psi_L$. This light is scattered at Ψ_L, to obtain a distribution along rays of the form $\Psi_L \to \mathbf{x}$. We then proceed as before (with a second scattering step at $\mathbf{x}$), to obtain emission along physical rays $\mathbf{x} \!\to\! \mathbf{x}'$. Similarly, measurements are handled by scattering light from rays of the form $\mathbf{x} \!\to\! \Psi_W$ into the single ray $\Psi_W \!\to\! \Psi_W$, where the actual measurement takes place.

This idea corresponds to the following symbolic definitions. First we define Φ_L and Φ_W to represent the total power and the total importance emitted over all surfaces of the scene:

$$\begin{aligned}
\Phi_L &= \int_{\mathcal{M}} L_{\mathrm{e}}^{(0)}(\mathbf{x})\, dA(\mathbf{x})\,, \\
\Phi_W &= \int_{\mathcal{M}} W_{\mathrm{e}}^{(0)}(\mathbf{x})\, dA(\mathbf{x})\,.
\end{aligned}$$

Next, we change the emission functions so that light and importance are emitted on a single imaginary ray:

$$\begin{aligned} L_{\mathrm{e}}(\Psi_L \!\rightarrow\! \Psi_L) &= \Phi_L \,, \\ W_{\mathrm{e}}(\Psi_W \!\rightarrow\! \Psi_W) &= \Phi_W \,. \end{aligned}$$

Finally, we extend the BSDF to scatter this light and importance along rays of the form $\Psi_L \!\rightarrow\! \mathbf{x}$ and $\mathbf{x} \!\rightarrow\! \Psi_W$ respectively:

$$\begin{aligned} f_{\mathrm{s}}(\Psi_L \!\rightarrow\! \Psi_L \!\rightarrow\! \mathbf{x}) &= L_{\mathrm{e}}^{(0)}(\mathbf{x}) \,/\, \Phi_L \,, \\ f_{\mathrm{s}}(\mathbf{x} \!\rightarrow\! \Psi_W \!\rightarrow\! \Psi_W) &= W_{\mathrm{e}}^{(0)}(\mathbf{x}) \,/\, \Phi_W \,. \end{aligned}$$

Notice that these BSDF's are normalized to integrate to one, so that there is a natural correspondence with scattering.

With these conventions, every S and D symbol corresponds to a unique scattering event at some vertex of the full path $\mathbf{x}_{-1} \ldots \mathbf{x}_{k+1}$. Furthermore, these symbols have a consistent meaning. Given any vertex $\mathbf{x}_i$ of a path, the symbol D means that the BSDF at that vertex is finite (so that energy is spread over a two-dimensional set of adjacent vertices), while S means that the BSDF is not finite (in which case power is distributed to a zero- or one-dimensional set of adjacent vertices). This consistency will be useful as we study the limitations of local path sampling below.

8.3.3 The limitations of local path sampling

In this section, we show that local sampling strategies can only generate paths that contain the substring DD. Any path that does not contain this substring cannot be sampled, and the contributions of these paths will be missing from any computed images. Examples of paths that cannot be sampled are shown in Table 8.2.

We start by consider specular vertices, and the constraints that they impose on path sampling. Next, we show that paths can be sampled by local sampling strategies if and only if they contain the substring DD. Finally, we discuss the significance of these results.

Lemma 8.1. *Let $\bar{x}$ be any path generated by a local sampling algorithm, for which the measurement contribution function $f_j(\bar{x})$ is non-zero. If this path contains a specular vertex $\mathbf{x}_i$,*

$L\,S\,D\,S\,D\,S\,E$	a point light source reflected in a mirror, viewed with a pinhole lens
$L\,D\,S\,S\,D\,S\,D\,E$	caustics from a parallel light source, viewed with an orthographic lens
$L\,S\,D\,S\,D\,S\,D\,S\,E$	caustics from a point light source, viewed indirectly through a mirror with a pinhole lens

Table 8.2: Examples of path types that cannot be generated by local sampling algorithms.

then one of the adjacent vertices $\mathbf{x}_{i+1}$ or $\mathbf{x}_{i-1}$ was necessarily generated by sampling the BSDF at $\mathbf{x}_i$.

Proof. For any fixed positions of $\mathbf{x}_i$ and $\mathbf{x}_{i-1}$, consider the positions of $\mathbf{x}_{i+1}$ for which

$$f_{\mathrm{s}}(\mathbf{x}_{i-1} \to \mathbf{x}_i \to \mathbf{x}_{i+1}) \;=\; \infty\,,$$

i.e. for which $\mathbf{x}_i$ is a specular vertex. By definition, the possible locations of $\mathbf{x}_{i+1}$ form a set of area measure zero, since they subtend a zero solid angle at $\mathbf{x}_i$. Similarly, if we fix the positions of $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$, the possible locations of $\mathbf{x}_{i-1}$ for which $\mathbf{x}_i$ is a specular vertex form a set of measure zero.

Thus, if the vertices $\mathbf{x}_{i-1}$ and $\mathbf{x}_{i+1}$ are generated independently by the local sampling algorithm, then $\mathbf{x}_i$ has type D with probability one. Thus if $\mathbf{x}_i$ has type S, then one of these two vertices must be generated by sampling the BSDF at $\mathbf{x}_i$ (since this is the only other alternative that is allowed within the framework of local path sampling). ■

It is easy to extend this result to the case where several specular vertices are adjacent.

Corollary 8.2. *Let $\bar{x}$ be a path as described above, and suppose that $\bar{x}$ contains a subpath $\mathbf{x}_i \ldots \mathbf{x}_j$ of the form DS^+D. Then one of the endpoints $\mathbf{x}_i$ or $\mathbf{x}_j$ must be generated by sampling the BSDF of the adjacent S-vertex (that is, either $\mathbf{x}_{i+1}$ or $\mathbf{x}_{j-1}$).* ■

We are now ready to consider the sampling of full paths.

Theorem 8.3. *Let $\bar{x}$ be a path generated by a local sampling algorithm for which the measurement contribution function is non-zero. Then $\bar{x}$ necessarily has the form*

$$L\,(S|D)^*\,D\,D\,(S|D)^*\,E\,,$$

i.e. it must contain the substring DD. Furthermore, it is possible to generate any path of this form using local sampling strategies.

Proof. If $\bar{x}$ does not contain the substring DD, then it has the form

$$L\,(D|\epsilon)\,S^+\,(DS^+)^*\,(D|\epsilon)\,E\,.$$

This path has n specular substrings of the form S^+, but only $n-1$ vertices of type D separating them.[6] Thus according to the corollary above, one of these D vertices must be generated by sampling the BSDF of *both* adjacent specular vertices (which is not possible). In effect, there are not enough D vertices to allow this path to be sampled by local techniques.

Conversely, let $\bar{x}$ be a path that contains an edge $\mathbf{x}_i\mathbf{x}_{i+1}$ of the form DD. Then this path can be generated by at least one local sampling strategy: namely, by generating the subpath $\mathbf{x}_0 \ldots \mathbf{x}_i$ starting from a light source, and the subpath $\mathbf{x}_{i+1} \ldots \mathbf{x}_k$ starting from the lens. ∎

Thus, the DD condition is necessary and sufficient for local path sampling. Of course, specific algorithms may have more restrictive requirements. With ordinary path tracing, for example, all vertices are generated starting from the camera lens, except for the vertex $\mathbf{x}_0$ which is chosen directly on the surface of a light source. This implies that ordinary path tracing can only sample paths of the form

$$L\,(S|D)\,D\,D\,(S|D)^+\,E\,.$$

These results are significant for two reasons. First, it is very common for graphics systems to support point light sources and perfect mirrors, even though these are mathematical idealizations that do not physically exist. If scenes are modeled that use these primitives, then some lighting effects will simply be missing from the computed images. Second, even

[6] The symbol following L and the symbol preceding E do not count, because they are not sampled: they represent the fixed, imaginary vertices Ψ_L and Ψ_W.

if we disallow these features (e.g. by disallowing point and parallel light sources, so that every path starts with the prefix LDD), we should expect that path sampling algorithms will perform badly as the scene model approaches a difficult configuration. In this case, the contributions from the difficult paths will not be missing; however, they will be sampled with high variance, leading to noisy regions in resulting images.

8.3.4 Approaches to non-local sampling

We outline several approaches for handling paths that cannot be sampled locally. The easiest solution is to not allow these paths in the first place, by placing mild restrictions on the scene model. For example, any of the following strategies are sufficient:

- Allow only (ordinary) area light sources, so that all paths start with LDD.
- Allow only finite-aperture lenses, so that all paths end with DDE.
- Do not allow perfectly specular surfaces.

These strategies ensure that path sampling algorithms will produce unbiased results, although there can still be high variance in limiting cases as discussed above.

A second approach is to use a more sophisticated path sampling strategy. We first introduce some new terminology.

Chains and chain separators. Given a path, we divide its edges into a sequence of *chains* as follows. A vertex is called a *chain separator* if it has type D, or if it is one of the special vertices Ψ_L or Ψ_W. A *chain* is now defined to be a maximal subpath bounded by chain separators (not including the symbols L and E, which do not correspond to any vertex). For example, the path

$$L\,D\,S\,D\,D\,S\,S\,D\,S\,E$$

consists of four chains. The first chain is DSD, consisting of the imaginary edge from Ψ_L to $\mathbf{x}_0$, and the real edge from $\mathbf{x}_0$ to $\mathbf{x}_1$. The second chain is DD (the edge $\mathbf{x}_1\mathbf{x}_2$), the third is $DSSD$ (three edges connecting $\mathbf{x}_2$ to $\mathbf{x}_5$), and the last chain is DS, an imaginary edge from $\mathbf{x}_5$ to Ψ_W. Notice that each chain separator vertex is shared between two chains (except for the special vertices Ψ_L and Ψ_W).

Connectors. We can extend the class of paths that can be sampled by implementing methods that generate *connecting chains*. That is, given two vertices $\mathbf{x}$ and $\mathbf{x}'$ of type D, we would like to generate a chain of zero or more specular vertices that connect them. Strategies that do this are called *connectors*. The simplest connector consists of joining the two vertices with an edge, by checking the visibility between them. This yields a chain of the form DD.

Another simple form of connector can be used with planar mirrors, by computing the point $\mathbf{y}$ on the mirror that reflects light from $\mathbf{x}$ to $\mathbf{x}'$. If such a point $\mathbf{y}$ does not exist, or if either of the segments $\mathbf{x}\mathbf{y}$ or $\mathbf{y}\mathbf{x}'$ is occluded, then the connection attempt fails. Otherwise, we have generated a connecting chain of the form DSD. This is similar to the idea of "virtual worlds" and "virtual light sources" used in radiosity and elsewhere [Rushmeier 1986, Wallace et al. 1987, Ward 1994].

Connectors can also be used to handle parallel light sources (LDS) and orthogonal viewing projections (SDE) in a simple way. For example, a connecting chain between a real vertex $\mathbf{x}$ and the imaginary vertex Ψ_L can be generated by projecting $\mathbf{x}$ onto the surface of the light source along the direction of emission.

The general case is closely related to the problem of computing illumination from curved reflectors [Mitchell & Hanrahan 1992]. The connecting chains problem can be equivalently stated as follows: given a point source at $\mathbf{x}$, what is the irradiance received at $\mathbf{x}'$ over specular paths? Light flows from $\mathbf{x}$ to $\mathbf{x}'$ along paths of stationary optical length, also known as *Fermat paths*. In general, there are a countable set of such paths, and they can be found by solving an optimization problem [Mitchell & Hanrahan 1992]. Once a path has been found, the irradiance received at $\mathbf{x}'$ along that path can be determined by keeping track of the shape of the wavefront as light is reflected, refracted, and propagated, and computing the Gaussian curvature of the wavefront at $\mathbf{x}'$.

In our case, we seek an algorithm that can either generate all such paths (in which case their contributions are summed), or one that can generate a single path at random (in which case there must be a non-zero probability of generating each candidate path, and this probability must be explicitly computable). This would make it possible to generate paths of any type in an unbiased Monte Carlo algorithm.

Although it seems unlikely that the general case will ever be practical, these ideas are still useful for handling planar mirrors, short sequences of such mirrors, or simple curved surfaces. With more sophisticated geometric search techniques, it may eventually be possible to handle moderately large numbers of specular surfaces in this way with reasonable efficiency.

Appendix 8.A Other measures on path space

We describe several new measures on the path space μ. These include the *measurement contribution measure*, the *power throughput measure*, the *scattering throughput measure*, and the *geometric throughput measure*. Each of these measures has a natural physical significance, which is described. We also show that it is possible to base the path integral framework on any of these measures (rather than using the area-product measure μ). To avoid confusion, we will use the symbol μ^{a} for the area-product measure throughout this appendix.

The measurement contribution measure. The most important of these new measures is the *measurement contribution measure*, defined by

$$\mu_j^{\mathrm{m}}(D) \;=\; \int_D f_j(\bar{x})\, \mu^{\mathrm{a}}(\bar{x})\,. \tag{8.14}$$

This equation combines f_j and μ^{a} into a single measure μ_j^{m}, with the following physical significance: $\mu_j^{\mathrm{m}}(D)$ represents the portion of measurement I_j that is due to light flowing on the given set of paths D. In particular, the value of I_j itself is given by

$$I_j \;=\; \mu_j^{\mathrm{m}}(\Omega)\,,$$

i.e. I_j is the measure of the whole path space. The units of $\mu_j^{\mathrm{m}}(D)$ are [S] (the unit of sensor response).

This measure μ_j^{m} is actually the fundamental component of our path integral framework. It is more basic than the measurement contribution function f_j, since f_j implicitly depends on the measure used for integration (i.e. the area-product measure μ^{a}). By choosing different integration measures (e.g. the ones we define below), we can obtain any number of different but equivalent "measurement contribution functions". In contrast, the meaning of μ_j^{m} does not depend on details such as these.

The main reason for working with the function f_j (rather than the measure μ_j^{m}) is so that Monte Carlo estimators can be written as a ratio of functions, rather than as Radon-Nikodym derivatives. For example, the estimator $f_j(\bar{X})/p(\bar{X})$ corresponds to the Radon-Nikodym derivative

$$\frac{d\mu_j^{\mathrm{m}}}{dP}(\bar{X})\,.$$

Although this may be an improvement from the standpoint of purism (since it avoids any reference to the auxiliary measure μ^{a}), it is undesirable from a practical standpoint. It makes use of the Radon-Nikodym derivative (which is unfamiliar to many in graphics), and leaves us with a rather abstract expression with no clear recipe for computing its value. This is why we have emphasized the formulation of Section 8.2, where μ_j^{m} is split into a function f_j and a measure μ^{a}, and where the measure is made as simple as possible.

The power throughput measure. We now consider another interesting measure called the *power throughput measure* (μ^{p}), which is obtained from the previous measure by omitting the importance function $W_{\mathrm{e}}^{(j)}$. Explicitly, it is defined for paths of length k by

$$\begin{aligned}\mu_k^{\mathrm{p}}(D) \;&=\; \int_D L_{\mathrm{e}}(\mathbf{x}_0 \!\rightarrow\! \mathbf{x}_1)\, G(\mathbf{x}_0 \!\leftrightarrow\! \mathbf{x}_1)\, f_{\mathrm{s}}(\mathbf{x}_0 \!\rightarrow\! \mathbf{x}_1 \!\rightarrow\! \mathbf{x}_2)\, G(\mathbf{x}_1 \!\leftrightarrow\! \mathbf{x}_2) \,\cdots \\ &\qquad\qquad \cdots\; f_{\mathrm{s}}(\mathbf{x}_{k-2} \!\rightarrow\! \mathbf{x}_{k-1} \!\rightarrow\! \mathbf{x}_k)\, G(\mathbf{x}_{k-1} \!\leftrightarrow\! \mathbf{x}_k)\; dA(\mathbf{x}_0) \cdots dA(\mathbf{x}_k)\,,\end{aligned} \tag{8.15}$$

where $D \subset \Omega_k$, and then extended to a measure μ^{p} over the whole path space by the same technique we used for the area-product measure (8.6).

Physically, $\mu^{\mathrm{p}}(D)$ represents the power that is carried by a set of paths D (units: [W]). A nice property of this measure is that it is independent of any sensor: there is only one measure for the whole scene, rather than one per sensor (as with μ_k^{m}). It can still be used to evaluate measurements, however, using the relationship

$$I_j \;=\; \int_\Omega W_{\mathrm{e}}^{(j)}(\mathbf{x}_{k-1} \!\rightarrow\! \mathbf{x}_k)\, d\mu^{\mathrm{p}}(\bar{x})\,.$$

This equation shows that I_j can be split into a function and a measure in more than one way. In this case, we have moved almost all the factors of f_j into the integration measure, leaving only $W_{\mathrm{e}}^{(j)}$ as the "measurement contribution function".

The scattering throughput measure. Next, we discuss the *scattering throughput measure* μ^{s}. The value $\mu^{\mathrm{s}}(D)$ represents the power-carrying capacity of a set of paths D, in the following sense: if a uniform radiance L_{e} is emitted along the first segment of each path in D, then the power carried by these paths and received by surfaces at the path endpoints will be

$$L_{\mathrm{e}}\, \mu^{\mathrm{s}}(D)\,.$$

The definition of μ^{s} is identical to the previous measure (8.15), except that the emitted radiance function L_{e} is omitted (as well as the importance function $W_{\mathrm{e}}^{(j)}$). A nice property of this measure is that it depends only on the scene geometry and materials, not on the light sources or sensors. The units of $\mu^{\mathrm{s}}(D)$ are $[\mathrm{m}^2 \cdot \mathrm{sr}]$.

The geometric throughput measure. Finally, we consider the *geometric throughput measure* μ^{g}, which measures the geometric "size" of a set of paths. To do this, we start with the expression for the scattering throughput μ^{s}, and set all of the BSDF factors to the constant value

$$f_{\mathrm{s}}(\mathbf{x}_{i-1} \to \mathbf{x}_i \to \mathbf{x}_{i+1}) \;=\; \frac{1}{2\pi} \,.$$

Physically, this corresponds to a scene where the surfaces scatter light in all directions uniformly; the value $1/(2\pi)$ ensures that f_{s} is energy-preserving (see Section 6.3).[7] With this modification to the scattering throughput measure μ^{s}, any differences in the power-carrying capacity of different path sets are due entirely to their geometry.

Explicitly, the geometric throughput measure μ^{g} is defined at each path length k by

$$\mu_k^{\mathrm{g}}(D) \;=\; \left(\frac{1}{2\pi}\right)^{k-1} \int_D G(\mathbf{x}_0 \leftrightarrow \mathbf{x}_1) \,\cdots\, G(\mathbf{x}_{k-1} \leftrightarrow \mathbf{x}_k)\; dA(\mathbf{x}_0) \cdots dA(\mathbf{x}_k) \,, \tag{8.16}$$

and extended to a measure μ^{g} over the whole path space as before. The term *geometric throughput measure* is particularly appropriate for μ^{g}, since it is a natural extension of the throughput measure μ defined on the space of rays (see Section 4.1): these two measures are identical for paths of length one. The units of μ^{g} are the same as the previous measure, namely $[\mathrm{m}^2 \cdot \mathrm{sr}]$.

Notice that μ^{g} has several properties that we should expect of a geometric measure on paths. First, it does not encode any preference for directional scattering at surfaces (since this is a property of materials rather than geometry). Second, in general the measure μ^{g} is not finite, even for scenes with finite surface area.[8] This corresponds to the fact that there is no geometric reason for light energy to diminish as it propagates over long paths.

In fact, by comparing the scattering and geometric throughput measures, it is possible to determine whether the power-carrying capacity of a given set of paths is limited primarily by materials or geometry. A suitable quantitative measure of this is the ratio

$$\mu^{\mathrm{s}}(D) \,/\, \mu^{\mathrm{g}}(D) \,.$$

[7] This type of surface has the same radiance when viewed from all directions, on both sides of the surface. In an environment where only reflection is allowed, i.e. where all surfaces are one-sided, the BRDF would be $f_{\mathrm{r}} = 1/\pi$ instead.

[8] If the scene has finite area, then $\mu_k^{\mathrm{g}}(\Omega_k)$ will be finite for each path length k. However, when we take the union Ω over all path lengths, the resulting space has infinite geometric measure.

The area-product measure. Finally, we return to the area-product measure μ^{a}. The chief advantage of this measure is that it is simple. This makes it easy to compute the probabilities of various sampling techniques with respect to this measure, so that we may compare them. Like the geometric throughput measure μ^{g}, the area-product measure is in general not finite.

Appendix 8.B Subclassification of specular vertices

Specular vertices can be subclassified into two categories, according to whether light is scattered into a zero- or one-dimensional set of directions. We distinguish between these possibilities with the symbols S_0 and S_1. This notation allows the properties of sources, sensors, and materials to be specified more precisely.

We first consider light sources, which are represented by a string of the formLXY. The first symbol X represents the physical extent of the light source, so thatS_0 denotes a point source, while S_1 denotes a linear, ring, or other one-dimensional source. The second symbol$Y$ represents the set of directions over which light is emitted. The symbolS_0 denotes emission in a discrete set of directions, while S_1 denotes emission into a plane or other one-dimensional set. A similar classification applies to sensors, which are represented by a string of the formYXE. Several examples are given in Table 8.3.

For scattering events, S_0 denotes a surface that scatters light from an incoming directionω_{i} into a discrete of directions (e.g. a mirror or a window). The symbolS_1 denotes a surface such as an ideal anisotropic reflector, where light from an incoming directionω_{i} is scattered into a one-dimensional set of outgoing rays.

For example, the full-path regular expression

$$L\, S_1\, D\, S_0^*\, S_0\, D\, E$$

represents a path where light is emitted from a linear source, bounces off zero or more mirrors, and then is measured by a camera with an orthographic lens.

Formal definitions of S_0, S_1, and D. For completeness, we give formal definitions of these symbols. Consider a scattering event at a vertex$\mathbf{x}_i$. As we have already mentioned, this vertex has type D is the BSDF at $\mathbf{x}_i$ is finite:

$$f_{\mathrm{s}}(\mathbf{x}_i, \omega_{\mathrm{i}} \!\rightarrow\! \omega_{\mathrm{o}}) \;<\; \infty\,,$$

where ω_{i} and ω_{o} are the directions toward $\mathbf{x}_{i-1}$ and $\mathbf{x}_{i+1}$ respectively.

The scattering event at $\mathbf{x}_i$ is defined to be S_0 whenever the BSDF behaves locally like a two-dimensional Dirac distribution (as was used to define the BSDF for mirror reflection in

LS_0D	a uniform point source, point spotlight, etc.
LS_0S_1	emission from a point into a planar fan or sheet
LS_0S_0	an idealized laser beam
LS_1D	a typical linear or ring source
LS_1S_1	an area light source in “flatland” [Heckbert 1990]
LDS_0	sunshine through a window
DS_0E	a typical pinhole lens model
DS_1E	a pinhole lens with motion blur due to movement of the camera (in a static scene)
S_0DE	an orthographic viewing projection
S_0S_0E	an idealized spot meter

Table 8.3: Examples of regular expressions for light sources and sensors, where the specular components have been subclassified into zero- and one-dimensional components.

Section 5.2.1.2). More precisely, this happens when there is a constant$\epsilon > 0$ such that

$$\int_D f_{\mathrm{s}}(\mathbf{x}_i, \omega \to \omega_{\mathrm{o}}) \, d\sigma^\perp(\omega) \;\geq\; \epsilon$$

for every open set $D \subset \mathcal{S}^2$ that contains ω_{i}.

Finally, a vertex is defined to be S_1 if it is not S_0 or D. It is straightforward to extend these definitions to the classification of light sources and sensors, using the functions$L_{\mathrm{e}}^{(i)}$ and $W_{\mathrm{e}}^{(i)}$ defined in Section 8.3.2.

Part II

Robust Light Transport Algorithms

Chapter 9

Multiple Importance Sampling

We introduce a technique called *multiple importance sampling* that can greatly increase the reliability and efficiency of Monte Carlo integration. It is based on the idea of using more than one sampling technique to evaluate a given integral, and combining the sample values in a provably good way.

Our motivation is that most numerical integration problems in computer graphics are “difficult”, i.e. the integrands are discontinuous, high-dimensional, and/or singular. Given a problem of this type, we would like to design a sampling strategy that gives a low-variance estimate of the integral. This is complicated by the fact that the integrand usually depends on parameters whose values are not known at the time an integration strategy is designed (e.g. material properties, the scene geometry, etc.) It is difficult to design a sampling strategy that works reliably in this situation, since the integrand can take on a wide variety of different shapes as these parameters vary.

In this chapter, we explore the general problem of constructing low-variance estimators by combining samples from several different sampling techniques. We do not construct new sampling techniques — we assume that these are given to us. Instead, we look for better ways to combine the samples, by computing weighted combinations of the sample values. We show that there is a large class of unbiased estimators of this type, which can be parameterized by a set of weighting functions. Our goal is to find an estimator with minimum variance, by choosing these weighting functions appropriately.

A good solution to this problem turns out to be surprisingly simple. We show how to combine samples from several techniques in a way that is provably good, both theoretically and practically. This allows us to construct Monte Carlo estimators that have low variance for a broad class of integrands — we call such estimators *robust*. The significance of our methods is not that we can take several bad sampling techniques and concoct a good one out of them, but rather that we can take several potentially good techniques and combine them so that the strengths of each are preserved.

This chapter is organized as follows. We start with an extended example to motivate our variance reduction techniques (Section 9.1). Specifically, we consider the problem of computing the appearance of a glossy surface illuminated by an area light source. Next, in Section 9.2 we explain the multiple importance sampling framework. Several models for taking and combining the sampling are described, and we present theoretical results showing that these techniques are provably close to optimal (proofs may be found in Appendix 9.A). In Section 9.3, we show that these techniques work well in practice, by presenting images and numerical measurements for two specific applications: the glossy highlights problem mentioned above, and the "final gather" pass that is used in some multi-pass algorithms. Finally, Section 9.4 discusses of a number of tradeoffs and open issues related to our work.

9.1 Application: glossy highlights from area light sources

We have chosen a problem from distribution ray tracing to illustrate our techniques. Given a glossy surface illuminated by an area light source, the goal is to determine its appearance. These "glossy highlights" are commonly evaluated in one of two ways: either by sampling the light source, or sampling the BSDF. We show that each method works very well in some situations, but fails in others. Obviously, we would prefer a sampling strategy that works well all the time. Later in this chapter, we will show how multiple importance sampling can be applied to solve this problem.

9.1.1 The glossy highlights problem

Consider an area light source S that illuminates a nearby glossy surface (see Figure 9.1). The goal is to determine the appearance of this surface, i.e. to evaluate the radiance $L_{\mathrm{o}}(\mathbf{x}', \omega_{\mathrm{o}}')$

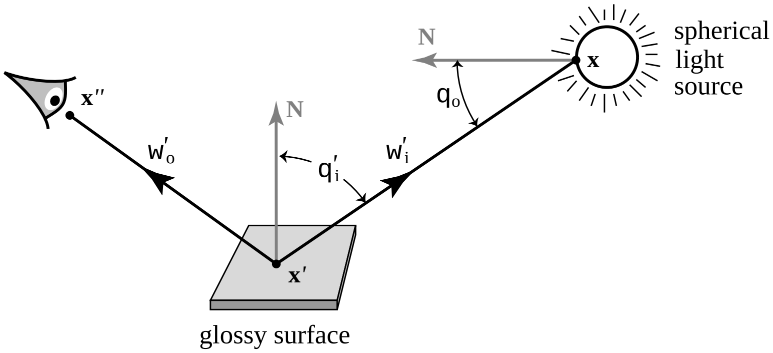


Figure 9.1: Geometry for the glossy highlights computation. The radiance for each viewing ray is obtained by integrating the light that is emitted by the source, and reflected from the glossy surface toward the eye.

that leaves the surface toward the eye. Mathematically, this is determined by the scattering equation (3.12):

$$L_{\mathrm{o}}(\mathbf{x}', \omega_{\mathrm{o}}') \;=\; \int_{\mathcal{S}^2} f_{\mathrm{s}}(\mathbf{x}', \omega_{\mathrm{i}}' \!\to\! \omega_{\mathrm{o}}')\, L_{\mathrm{e,i}}(\mathbf{x}', \omega_{\mathrm{i}}')\, d\sigma^{\perp}(\omega_{\mathrm{i}}')\,, \tag{9.1}$$

where $L_{\mathrm{e,i}}$ represents the incident radiance due to the area light source S.

We will examine a family of integration problems of this form, obtained by varying the size of the light source and the glossiness of the surface. In particular, we consider spherical light sources of varying radii, and glossy materials that have a *surface roughness parameter* (r) that determines how sharp or fuzzy the reflections are. Smooth surfaces ($r = 0$) correspond to highly polished, mirror-like reflections, while rough surfaces ($r = 1$) correspond to diffuse reflection. It is possible to simulate a variety of surface finishes by using intermediate roughness values in the range $0 < r < 1$.

9.1.2 Two sampling strategies

There are two common strategies for Monte Carlo evaluation of the scattering equation (9.1), which we call *sampling the BSDF* and *sampling the light source*. The results of these techniques are demonstrated in Figure 9.2(a) and Figure 9.2(b) respectively, over a range of different light source sizes and surface finishes. We will first describe these two strategies, and then examine why each one has high variance in some situations.

Sampling the BSDF. To sample the BSDF, an incident direction $\omega'_{\rm i}$ is randomly chosen according to a predetermined density $p(\omega'_{\rm i})$. Normally, this density is chosen to be proportional to the BSDF (or some convenient approximation), i.e.

$$p(\omega'_{\rm i}) \;\propto\; f_{\rm s}(\mathbf{x}', \omega'_{\rm i} \!\to\! \omega'_{\rm o})\,,$$

where p is measured with respect to projected solid angle. To estimate the scattering equation (9.1), an estimate of the usual form

$$L_{\rm o}(\mathbf{x}', \omega'_{\rm o}) \;\approx\; \frac{f_{\rm s}(\mathbf{x}', \omega'_{\rm i} \!\to\! \omega'_{\rm o})\, L_{\rm e,i}(\mathbf{x}', \omega'_{\rm i})}{p(\omega'_{\rm i})}$$

is used. The emitted radiance $L_{\rm e,i}(\mathbf{x}', \omega'_{\rm i})$ is evaluated by casting a ray to find the corresponding point on the light source. Note that some rays may miss the light source S, in which case they do not contribute to the highlight calculation. The image in Figure 9.2(a) was computed using this strategy.

Sampling the light source. To explain the other strategy, we first rewrite the scattering equation as an integral over the surface of the light source:

$$L_{\rm o}(\mathbf{x}' \!\to\! \mathbf{x}'') \;=\; \int_{\mathcal{M}} f_{\rm s}(\mathbf{x} \!\to\! \mathbf{x}' \!\to\! \mathbf{x}'')\, L_{\rm e}(\mathbf{x} \!\to\! \mathbf{x}')\, G(\mathbf{x} \!\leftrightarrow\! \mathbf{x}')\, dA(\mathbf{x})\,. \tag{9.2}$$

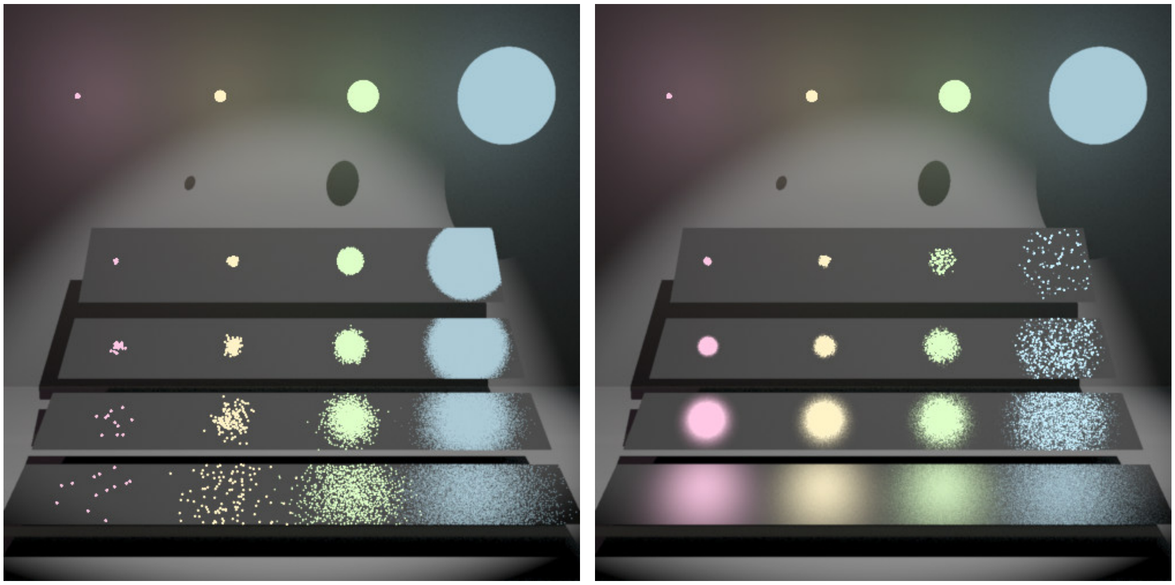
This is called the *three-point form* of the scattering equation (previously described in Section 8.1). The function G represents the change of variables from $d\sigma^{\perp}(\omega'_{\rm i})$ to $dA(\mathbf{x})$, and is given by

$$G(\mathbf{x} \!\leftrightarrow\! \mathbf{x}') \;=\; V(\mathbf{x} \!\leftrightarrow\! \mathbf{x}')\, \frac{|\cos(\theta_{\rm o})\; \cos(\theta'_{\rm i})|}{\|\mathbf{x} - \mathbf{x}'\|^2}$$

(see Figure 9.1).

The strategy of sampling the light source now proceeds as follows. First, a point $\mathbf{x}$ on the light source S is randomly chosen according to a predetermined density $p(\mathbf{x})$, and then a standard Monte Carlo estimate of the form

$$L_{\rm o}(\mathbf{x}' \!\to\! \mathbf{x}'') \;\approx\; \frac{L_{\rm e}(\mathbf{x} \!\to\! \mathbf{x}')\, G(\mathbf{x} \!\leftrightarrow\! \mathbf{x}')}{p(\mathbf{x})}\, f_{\rm s}(\mathbf{x} \!\to\! \mathbf{x}' \!\to\! \mathbf{x}'')$$



(a) Sampling the BSDF

(b) Sampling the light sources

Figure 9.2: A comparison of two sampling techniques for glossy highlights from area light sources. There are four spherical light sources of varying radii and color, plus a spotlight overhead. All spherical light sources emit the same total power. There are also four shiny rectangular plates, each one tilted so that we see the reflected light sources. The plates have varying degrees of surface roughness, which controls how sharp or fuzzy the reflections are.

Given a viewing ray that strikes a glossy surface (see Figure 9.1), images (a) and (b) use different sampling techniques for the highlight calculation. Both images are 500 by 500 pixels.

(a) Incident directions ω'_{i} are chosen with probability proportional to the BSDF $f_{\mathrm{s}}(\mathbf{x}', \omega'_{\mathrm{i}} \!\rightarrow\! \omega'_{\mathrm{o}})$, using $n_1 = 4$ samples per pixel. We call this strategy *sampling the BSDF.*

(b) Sample points $\mathbf{x}$ are randomly chosen on each light source S, using $n_2 = 4$ samples per pixel (per light source). The samples are uniformly distributed within the solid angle subtended by S at the current point $\mathbf{x}'$. We call this strategy *sampling the light source.*

The glossy BSDF used in these images is a symmetric, energy-conserving variation of the Phong model. The Phong exponent is $n = (1/r) - 1$, where r is the surface roughness parameter mentioned above, and $0 \leq r \leq 1$. The glossy surfaces also have a small diffuse component. Similar effects would occur with other glossy BSDF's.

is used. The image in Figure 9.2(b) was computed with this type of strategy, where samples were chosen according to the density

$$p(\mathbf{x}) \;\propto\; L_{\rm e}(\mathbf{x}\!\to\!\mathbf{x}')\,\frac{|\cos(\theta_{\rm o})|}{\|\mathbf{x}-\mathbf{x}'\|^2}$$

(measured with respect to surface area). With this strategy, the sample points $\mathbf{x}$ are uniformly distributed within the solid angle subtended by the light source at the current point $\mathbf{x}'$. (See Shirley et al. [1996] for further details on light source sampling strategies.)

9.1.3 Comparing the two strategies

One of these sampling strategies can have a much lower variance than the other, depending on the size of the light source and the surface roughness parameter. For example, if the light source is small and the material is relatively diffuse, then sampling the light source gives far better results than sampling the BSDF (compare the lower left portions of the images in Figure 9.2). On the other hand, if the light source is large and the material is highly polished, then sampling the BSDF is far superior (compare the upper right portions of Figure 9.2).

In both these cases, high variance is caused by inadequate sampling where the integrand is large. To understand this, notice that the integrand in the scattering equation (9.2) is a product of various factors — the BSDF $f_{\rm s}$, the emitted radiance $L_{\rm e}$, and several geometric quantities. The ideal density function for sampling would be proportional to the product of all of these factors, according to the principle that the variance is zero when $p(x) \propto f(x)$ (see Chapter 2).

However, neither sampling strategy takes all of these factors into account. For example, the light source sampling strategy does not consider the BSDF of the glossy surface. Thus when the BSDF has a large influence on the overall shape of the integrand (e.g. when it is a narrow, peaked function), then sampling the light source leads to high variance. On the other hand, the BSDF sampling strategy does not consider the emitted radiance function $L_{\rm e}$. Thus it leads to high variance when the emission function dominates the shape of the integrand (e.g. when the light source is very small). As a consequence of these two effects, neither sampling strategy is effective over the entire range of light source geometries and surface finishes.

It is important to realize that both strategies are importance sampling techniques aimed at generating sample points on the same domain. This domain can be modeled as either a set of directions, as in equation (9.1), or a set of surface points, as in equation (9.2). For example, the BSDF sampling strategy can be expressed as a distribution over the surface of the light source, using the relationship

$$p(\mathbf{x}) \;=\; p(\omega_{\mathrm{i}}') \, \frac{d\sigma^{\perp}(\omega_{\mathrm{i}}')}{dA(\mathbf{x})} \;=\; p(\omega_{\mathrm{i}}') \, \frac{|\cos(\theta_{\mathrm{o}}) \, \cos(\theta_{\mathrm{i}}')|}{\|\mathbf{x} - \mathbf{x}'\|^2} \tag{9.3}$$

(as discussed in Section 8.2.2.2). This formula makes it possible to convert a directional density into an area density, so that we can express the two sampling strategies as different probability distributions on the same domain.

9.1.4 Discussion

There are many problems in graphics that are similar to the glossy highlights example, where a large number of integrals of a specific form must be evaluated. The integrands generally have a known structure (e.g. $f(x) = f_1(x) f_2(x) + f_3(x)$), but they also depend on various parameters of the scene model (e.g. the surface roughness and light source geometry in the example above). This makes it difficult to design an adequate sampling strategy, since the parameter values are not known in advance. Furthermore, different integrals may have different parameter values even within the same scene (e.g. they may change from pixel to pixel).

The main issue is that we would like low-variance results for the entire range of parameter values, i.e. for all of the potential integrands that are obtained as these parameters vary. Unfortunately, it is often difficult to achieve this. The problem is that the integrand is usually a sum or product of many different factors, and is too complicated to sample from directly. Instead, samples are chosen from a density function that is proportional to some subset of the factors (e.g. the BSDF sampling strategy outlined above). This can lead to high variance when one of the unconsidered factors has a large effect on the integrand.

We propose a new strategy for this kind of integration problem, called *multiple importance sampling*. It is based on the idea of taking samples using several different techniques, designed to sample different features of the integrand. For example, suppose that the integrand has the form

$$f \;=\; (f_1 + f_2)\, f_3\,.$$

If the functions f_i are simple enough to be sampled directly, then the density functions $p_i \propto f_i$ would all be good candidates for sampling. Similarly, if the integrand is a product

$$f \;=\; f_1\, f_2 \,\cdots\, f_k\,,$$

then several different density functions p_i could be chosen, each proportional to the product of a different set of f_i. In this way, it is often possible to find a set of importance sampling techniques that cover the various factors that can cause high variance.

Our main concern in this chapter is not how to construct a suitable set of sampling techniques, or even how to determine the number of samples that should be taken from each one. Instead, we consider the problem of how these samples should be combined, once they have been taken. We will show how to do this in a way that is unbiased, and with a variance is provably close to optimal.

In the glossy highlights problem, for example, we propose taking samples using both the BSDF and light source sampling strategies. We then show how these samples can be automatically combined to obtain low-variance results over the entire range of surface roughness and light source parameters. (For a preview of our results on this test case, see Figure 9.8.)

9.2 Multiple importance sampling

In this section, we show how Monte Carlo integration can be made more robust by using more than one sampling technique to evaluate the same integral. Our main results are on how to combine the samples: we propose strategies that are provably good compared to any other unbiased method. This makes it possible to construct estimators that have low variance for a broad class of integrands.

We start by describing a general model for combining samples from multiple techniques, called the *multi-sample model*. Using this model, any unbiased method of combining the samples can be represented as a set of weighting functions. This gives us a large space of possible combination strategies to explore, and a uniform way to represent them.

We then present a provably good strategy for combining the samples, which we call the *balance heuristic*. We show that this method gives a variance that is smaller than any other unbiased combination strategy, to within a small additive term. The method is simple and practical, and can make Monte Carlo calculations significantly more robust. We also propose several other combination strategies, which are basically refinements of the balance heuristic: they retain its provably good behavior in general, but are designed to have lower variance in a common special case. For this reason, they are often preferable to the balance heuristic in practice.

We conclude by considering a different model for how the samples are taken and combined, called the *one-sample model*. Under this model, the integral is estimated by choosing one of the n sampling techniques at random, and then taking a single sample from it. Again we consider how to minimize variance by weighting the samples, and we show that for this model the balance heuristic is optimal.

9.2.1 The multi-sample model

In order to prove anything about our methods, there must be a precise model for how the samples are taken and combined. For most of this chapter, we will use the *multi-sample model* described below. This model allows any unbiased combination strategy to be encoded as a set of weighting functions.

We consider the evaluation of an integral

$$\int_\Omega f(x)\, d\mu(x)\,,$$

where the domain Ω, the function $f : \Omega \to \mathbb{R}$, and the measure μ are all given. We are also given a set of n different sampling techniques on the domain Ω, whose corresponding density functions are labeled p_1, . . ., p_n. We assume that only the following operations are available:

- Given any point $x \in \Omega$, $f(x)$ and $p_i(x)$ can be evaluated.

- It is possible to generate a sample X distributed according to any of the p_i.

To estimate the integral, several samples are generated using each of the given techniques. We let n_i denote the number of samples from p_i, where $n_i \geq 1$, and we let $N = \sum n_i$ denote the total number of samples. We assume that the number of samples from each technique is fixed in advance, before any samples are taken. (We do not consider the problem of how to allocate samples among the techniques; this is an interesting problem in itself, which will be discussed further in Section 9.4.2.) The samples from technique i are denoted $X_{i,j}$, for $j = 1, \ldots, n_i$. All samples are assumed to be independent, i.e. new random bits are generated to control the selection of each one.

9.2.1.1 The multi-sample estimator

We now examine how the samples $X_{i,j}$ can be used to estimate the desired integral. Our goal is generality: given any unbiased way of combining the samples, there should be a way to represent it. To do this, we consider estimators that allow the samples to be weighted differently, depending on which technique p_i they were sampled from. Each estimator has an associated set of weighting functions w_1, . . ., w_n which give the weight $w_i(x)$ for each sample x drawn from p_i. The *multi-sample estimator* is then given by

$$F \;=\; \sum_{i=1}^{n} \frac{1}{n_i} \sum_{j=1}^{n_i} w_i(X_{i,j}) \, \frac{f(X_{i,j})}{p_i(X_{i,j})} \,. \tag{9.4}$$

This formula can be thought of as a weighted sum of the estimators $f(X_{i,j})/p_i(X_{i,j})$ that would be obtained by using each sampling technique p_i on its own. Notice that the weights are not constant, but can vary as a function of the sample point $X_{i,j}$.

For this estimate to be unbiased, the weighting functions w_i must satisfy the following two conditions:

(W1) $\sum_{i=1}^{n} w_i(x) \;=\; 1$ whenever $f(x) \neq 0$, and

(W2) $w_i(x) = 0$ whenever $p_i(x) = 0$.

These conditions imply the following corollary: at any point where $f(x) \neq 0$, at least one of the $p_i(x)$ must be positive (i.e., at least one sampling technique must be able to generate samples there). Thus on the other hand, it is not necessary for every p_i to sample the whole domain; it is allowable for some of the p_i to be specialized sampling techniques that concentrate on specific regions of the integrand.[1]

Given that (W1) and (W2) hold, the following lemma states that F is unbiased:

Lemma 9.1. *Let F be any estimator of the form (9.4), where $n_i \geq 1$ for all i, and the weighting functions w_i satisfy conditions (W1) and (W2). Then*

$$E[F] \;=\; \int_\Omega f(x)\, d\mu(x)\,.$$

Proof.

$$\begin{aligned} E[F] \;&=\; \sum_{i=1}^{n} \frac{1}{n_i} \sum_{j=1}^{n_i} \int_\Omega \frac{w_i(x)\, f(x)}{p_i(x)}\, p_i(x)\, d\mu(x) \\ &=\; \int_\Omega \sum_{i=1}^{n} w_i(x)\, f(x)\, d\mu(x) \\ &=\; \int_\Omega f(x)\, d\mu(x)\,. \quad \blacksquare \end{aligned}$$

The remainder of this section is devoted to showing the generality of the multi-sample model. We show that by choosing the weighting functions appropriately, it is possible to represent virtually any unbiased combination strategy. To make this more concrete, we first give some examples of possible strategies, and show how to represent them by weighting functions. We then show how the multi-sample estimator can be rewritten in a different form that makes its generality more obvious. This leads up to Section 9.2.2, where we will describe a new combination strategy that has provably good performance compared to all strategies that the multi-sample model can represent.

9.2.1.2 Examples of weighting functions

Suppose that there are three sampling techniques p_1, p_2, and p_3, and that a single sample $X_{i,1}$ is taken from each one ($n_1 = n_2 = n_3 = 1$). First, consider the case where the weighting

[1] If f is allowed to contain Dirac distributions, note that (W2) should be modified to state that $w_i(x) = 0$ whenever $f(x)/p_i(x)$ is not finite. To relate this to graphics, consider a mirror which also reflects some light diffusely. The modified (W2) states that samples from the diffuse component cannot be used to estimate the specular contribution, since this corresponds to the situation where $f(x)$ contains a Dirac distribution $\delta(\mathbf{x} - \mathbf{x}_0)$, but $p(x)$ does not.)

functions are constant over the whole domain Ω. This leads to the estimator

$$F \;=\; w_1 \frac{f(X_{1,1})}{p_1(X_{1,1})} + w_2 \frac{f(X_{2,1})}{p_2(X_{2,1})} + w_3 \frac{f(X_{3,1})}{p_3(X_{3,1})}\,,$$

where the w_i sum to one. This estimator is simply a weighted combination of the estimators $F_i \;=\; f(X_{i,1}) \,/\, p_i(X_{i,1})$ that would be obtained by using each of the sampling techniques alone. Unfortunately, this combination strategy does not work very well: if any of the given sampling techniques is bad (i.e. the corresponding estimator F_i has high variance), then F will have high variance as well, since

$$V[F] \;=\; w_1 V[F_1] + w_2 V[F_2] + w_3 V[F_3]\,.$$

Another possible combination strategy is to partition the domain among the sampling techniques. To do this, the integral is written in the form

$$\int_\Omega f(x)\,d\mu(x) \;=\; \sum_{i=1}^{n} \int_{\Omega_i} f(x)\,d\mu(x)\,,$$

where the Ω_i are non-overlapping regions whose union is Ω. The integral is then estimated in each region Ω_i separately, using samples from just one technique p_i. In terms of weighting functions, this is represented by letting

$$w_i(x) \;=\; \begin{cases} 1 & \text{if } x \in \Omega_i\,, \\ 0 & \text{otherwise}\,. \end{cases}$$

This combination strategy is used a great deal in computer graphics; however, sometimes it does not work very well due to the simple partitioning rules that are used. For example, it is common to evaluate the scattering equation by dividing the scene into light source regions and non-light-source regions, which are sampled using different techniques (e.g. sampling $L_{\rm e}$ vs. sampling the BSDF). Depending on the geometry and materials of the scene, this fixed partitioning can lead to a much higher variance than necessary (as we saw in the glossy highlights example).

Another combination technique that is often used in graphics is to write the integrand as a sum $f = \sum g_i$, and use a different sampling technique to estimate the contribution of each g_i. For example, this occurs when the BSDF is split into diffuse, glossy, and specular components, whose contributions are estimated separately (by sampling from density functions $p_i \propto g_i$). As before, it is straightforward to represent this strategy as a set of weighting functions.

9.2.1.3 Generality of the multi-sample model

The generality of this model can be seen more easily by rewriting the multi-sample estimator (9.4) in the form

$$F \;=\; \sum_{i=1}^{n} \sum_{j=1}^{n_i} C_i(X_{i,j})\,, \tag{9.5}$$

where $C_i(X_{i,j})$ is the called the *sample contribution* for $X_{i,j}$. The functions C_i are arbitrary, except that in order for F to be unbiased they must satisfy

$$\sum_{i=1}^{n} n_i\, C_i(x)\, p_i(x) \;=\; f(x) \tag{9.6}$$

at each point $x \in \Omega$. In this form, it is clear that the multi-sample model can represent any unbiased combination strategy, subject only to the assumptions that all samples are taken independently, and that our knowledge of f and p_i is limited to point evaluation. (This forces the estimator to be unbiased at each point x independently, as expressed by condition (9.6).)

To see that this formulation of the multi-sample model is equivalent to the original one, we simply let

$$C_i(x) \;=\; \frac{w_i(x)\, f(x)}{n_i\, p_i(x)}\,. \tag{9.7}$$

It is easy to verify that if the weighting functions w_i satisfy conditions (W1) and (W2), then the corresponding contributions C_i satisfy (9.6), and vice versa. The main reason for preferring the w_i formulation is that the corresponding conditions are easier to satisfy.

9.2.2 The balance heuristic

The multi-sample model gives us a large space of unbiased estimators to explore, and a uniform way to represent them (as a set of weighting functions). Our goal is now to find the estimator F with minimum variance, by choosing the w_i appropriately.

We will show that the following weighting functions are a good choice:

$$\hat{w}_i(x) \;=\; \frac{n_i\, p_i(x)}{\sum_k \, n_k\, p_k(x)}\,. \tag{9.8}$$

We call this strategy the *balance heuristic*.[2] The key feature of the balance heuristic is that no other combination strategy is much better, as stated by the following theorem:

Theorem 9.2. *Let f, n_i, and p_i be given, for $i = 1, \ldots, n$. Let F be any unbiased estimator of the form (9.4), and let $\hat{F}$ be the estimator that uses the weighting functions $\hat{w}_i$ (the balance heuristic). Then*

$$V[\hat{F}] - V[F] \;\le\; \left(\frac{1}{\min_i n_i} \;-\; \frac{1}{\sum_i n_i} \right) \mu^2\,, \tag{9.9}$$

where $\mu = E[F] = E[\hat{F}]$ is the quantity to be estimated. (A proof is given in Appendix 9.A.)

According to this result, no other combination strategy can significantly improve upon the balance heuristic. That is, suppose that we let F^* denote the *best possible* combination strategy for a particular problem (i.e. for a given choice of the f, p_i, and n_i). In general, we have no way of knowing what this strategy is: for example, suppose that one of the p_i is exactly proportional to f, so that the best strategy is to ignore any samples taken with the other techniques, and use only the samples from p_i. We cannot hope to discover this fact from a practical point of view, since our knowledge of f and p_i is limited to point sampling and evaluation. Nevertheless, even compared to this unknown optimal strategy F^*, the balance heuristic is almost as good: its variance is worse by at most the term on the right-hand side of (9.9).

To give some intuition about this upper bound on the “variance gap”, suppose that there are just two sampling techniques, and that $n_1 = n_2 = 4$ samples are taken from each one. In this case, the variance of the balance heuristic is optimal to within an additive term of $\mu^2/8$. In familiar graphics terms, this corresponds to the variance obtained by sending 8 shadow

[2] The name refers to the fact that the sample contributions are “balanced” so that they are the same for all techniques i:

$$C_i(x) \;=\; \frac{\hat{w}_i(x)\, f(x)}{n_i\, p_i(x)} \;=\; \frac{f(x)}{\sum_k \, n_k\, p_k(x)}\,.$$

That is, the contribution $C_i(X_{i,j})$ of a sample $X_{i,j}$ does not depend on which technique i generated it.

rays to an area light source that is 50% occluded. Furthermore, notice that the variance gap goes to zero as the number of samples from each technique is increased. On the other hand, if a poor combination strategy is used then the variance can be larger than optimal by an arbitrary amount. This is essentially what we observed in the glossy highlights images of Figure 9.2: if the wrong samples are used to estimate the integral, the variance can be tens or hundreds of times larger than μ^2.

Furthermore, the balance heuristic is practical to implement. The main requirement for evaluating the weighting functions $\hat{w}_i$ is that given any point x, we must be able to evaluate the probability densities $p_k(x)$ for all k. This situation is different than for the usual estimator $f(X)/p(X)$, where it is only necessary to evaluate $p(X)$ for sample points generated using p. The balance heuristic requires slightly more than this: given a sample $X_{i,j}$ generated using technique p_i, we also need to evaluate the probabilities $p_k(X_{i,j})$ with which all of the *other* $n-1$ techniques generate that sample point. It is usually straightforward to do this; it is just a matter of reorganizing the routines that compute probabilities, and expressing all densities with respect to the same measure.

For example, consider the glossy highlights problem of Section 9.1. To evaluate the weighting function $\hat{w}_i$ for each sample point x, we compute the probability density for generating x using both sampling techniques. Thus if x was generated by sampling the light source, then we also compute the probability density for generating the same point x by sampling the BSDF (as discussed in Section 9.1.3). Note that the cost of computing these extra probabilities is insignificant compared to the other calculations involved, such as ray casting; details will be given in Section 9.3.

9.2.2.1 A simple interpretation of the balance heuristic

By writing the balance heuristic in a different form, we will show that it is actually a very natural way to combine samples from multiple techniques.

To do this, we insert the weighting functions $\hat{w}_i$ into the multi-sample estimator (9.4), yielding

$$\hat{F} \;=\; \sum_{i=1}^{n} \frac{1}{n_i} \sum_{j=1}^{n_i} \left(\frac{n_i\, p_i(X_{i,j})}{\sum_k \, n_k\, p_k(X_{i,j})} \right) \frac{f(X_{i,j})}{p_i(X_{i,j})}$$

$$
\begin{aligned}
&= \sum_{i=1}^{n} \sum_{j=1}^{n_i} \frac{f(X_{i,j})}{\sum_k n_k\, p_k(X_{i,j})} \\
&= \frac{1}{N} \sum_{i=1}^{n} \sum_{j=1}^{n_i} \frac{f(X_{i,j})}{\sum_k c_k\, p_k(X_{i,j})}\,, \qquad (9.10)
\end{aligned}
$$

where $N = \sum_i n_i$ is the total number of samples, and $c_k = n_k/N$ is the fraction of samples from p_k.

In this form, the balance heuristic corresponds to a standard Monte Carlo estimator of the form f/p. This can be seen more easily by rewriting the denominator of (9.10) as

$$
\hat{p}(x) \;=\; \sum_{k=1}^{n} c_k\, p_k(x)\,, \qquad (9.11)
$$

which we call the *combined sample density*. The quantity $\hat{p}(x)$ represents the probability density for sampling the given point x, averaged over the entire sequence of N samples.[3]

Thus, the balance heuristic is natural way to combine the samples. It has the form of a standard Monte Carlo estimator, where the denominator $\hat{p}$ represents the average distribution of the whole group of samples to which it is applied. Pseudocode for this estimator is given in Figure 9.3. However, it is important to realize that the main advantage of this estimator is not that it is simple or standard, but that it has provably good performance compared to other combination strategies. This is the reason that we introduced the more complex formulation in terms of weighting functions, so that we could compare it against a family of other techniques.

9.2.3 Improved combination strategies

Although the balance heuristic is a good combination strategy, there is still some room for improvement (within the bounds given by Theorem 9.2). In this section, we discuss two families of estimators that have lower variance than the balance heuristic in a common special case. These estimators are unbiased, and like the balance heuristic, they are provably good compared to all other combination strategies.

[3] More precisely, it is the density of a random variable X that is equal to each $X_{i,j}$ with probability $1/N$.

```
function BALANCE-HEURISTIC()
    $N \leftarrow \sum_{i=1}^{n} n_i$
    for $i \leftarrow 1$ to $n$
        for $j \leftarrow 1$ to $n_i$
            $X \leftarrow$ TAKESAMPLE$(p_i)$
            $\hat{p} \leftarrow \sum_{k=1}^{n} (n_k / N)\, p_k(X)$
            $F \leftarrow F + f(X)/\hat{p}$
    return $F/N$
```

Figure 9.3: Pseudocode for the balance heuristic estimator.

We start by applying the balance heuristic to the glossy highlights problem of Section 9.1. We show that it leads to more variance than necessary in exactly those cases where the original sampling techniques did very well, e.g. where sampling the light source gave a low-variance result. The problem is that the additional variance due to the balance heuristic is additive: this is not significant when the optimal estimator already has substantial variance, but it is noticeable compared to an optimal estimator whose variance is very low.

We thus consider how to improve the performance of the balance heuristic on *low-variance problems*, i.e. those for which one of the given sampling techniques is an excellent match for the integrand. We show that the balance heuristic can be improved in this case by modifying its weighting functions slightly. In particular, we show that it is desirable to *sharpen* these weighting functions, by decreasing weights that are close to zero, and increasing weights that are close to one. We propose two general strategies for doing this, which we call the *cutoff* and *power* heuristics. The balance heuristic can be obtained as a limiting case of both these families of estimators.

Finally, we give some theoretical results showing that these new combination strategies are provably close to optimal. Thus, they are never much worse than the balance heuristic, but for low-variance problems they can be noticeably better. Later in this chapter, we will describe numerical tests that verify these results (Section 9.3). Based on these experiments, we have found that one strategy in particular is a good choice in practice: namely, the power

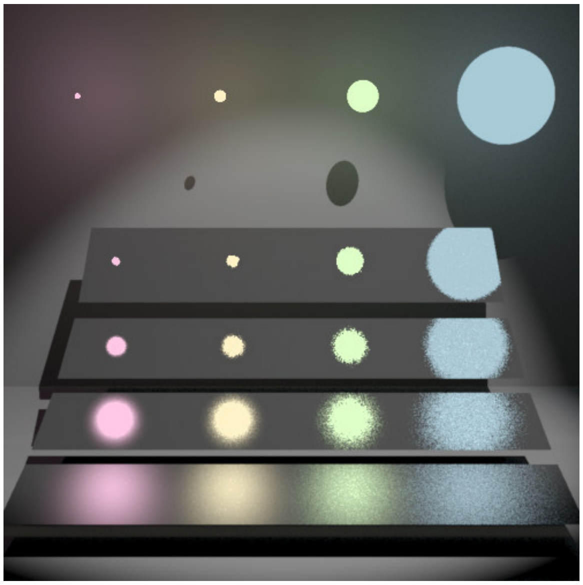


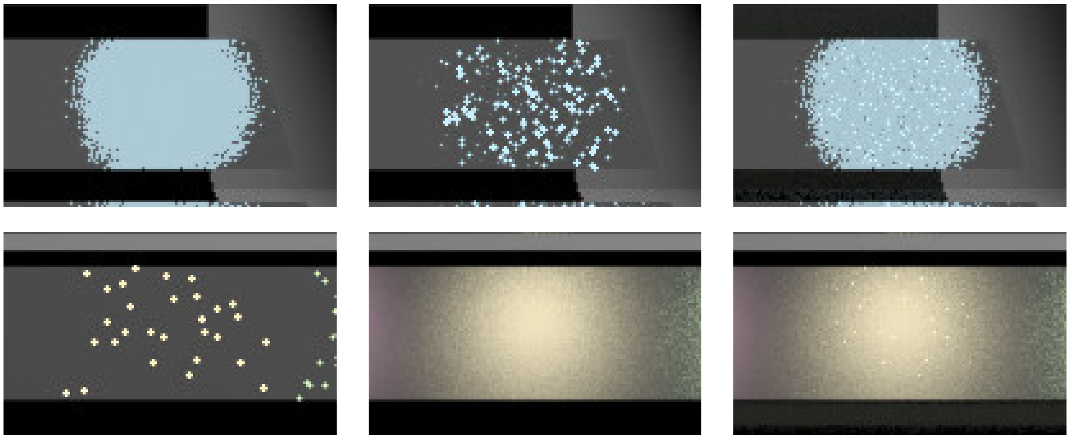
Figure 9.4: This image was rendered using both the BSDF sampling strategy and the light source sampling strategy. The samples are exactly the same as those for Figure 9.2(a) and (b), except that here the two kinds of samples are combined using the balance heuristic. This leads to a strategy that is effective over the entire range of glossy surfaces and light source geometries.

heuristic with the exponent $\beta = 2$.

9.2.3.1 Low-variance problems: examples and analysis

Figure 9.4 shows the balance heuristic applied to glossy highlights problem of Section 9.1. This image combines the samples from Figure 9.2(a) and (b), which used the BSDF and light source sampling strategies respectively. By combining both kinds of samples, we obtain a strategy that works well over the entire range of surface finishes and light source geometries.

In some regions of the image, however, the balance heuristic does not work quite as well as the best of the given sampling techniques. Figure 9.5 demonstrates this, by comparing the balance heuristic against images that use the BSDF or light source samples alone. Columns (a), (b), and (c) show close-ups of the images in Figure 9.2(a), Figure 9.2(b), and Figure 9.4 respectively. To make the differences more obvious, these images were computed using



(a) Sampling the BSDF **(b)** Sampling the lights **(c)** The balance heuristic

Figure 9.5: These images show close-ups of the glossy highlights test scene, computed by **(a)** sampling the BSDF, **(b)** sampling the light sources, and **(c)** the balance heuristic. Notice that although the balance heuristic works much better than one of the two techniques in each region, it does not work quite as well as the other. These images were computed with one sample per pixel from each technique ($n_1 = n_2 = 1$), as opposed to the four samples per pixel used in Figures 9.2 and 9.4, in order to reveal the noise differences more clearly.

only one sample per pixel (as opposed to the four samples per pixel used in the source images.) It is clear that although the balance heuristic works far better in each region than the technique whose variance is high, it has some additional noise compared to the technique whose variance is low.

The test cases in Figure 9.5 are examples of *low-variance problems*, which occur when one of the given sampling techniques p_i is an extremely good match for the integrand f. In this situation it is possible to construct an estimator whose variance is nearly zero, by taking samples using p_i and applying the standard estimate f/p_i. The balance heuristic can be noticeably worse than the results obtained in this way, because Theorem 9.2 only states that the variance of the balance heuristic is optimal to within an additive extra term. Even though this extra variance is guaranteed to be small on an absolute scale, it can still be noticeable compared to an optimal variance that is practically zero (especially if only a few samples are taken).

Unfortunately, there is no way to reliably detect this situation under the point sampling

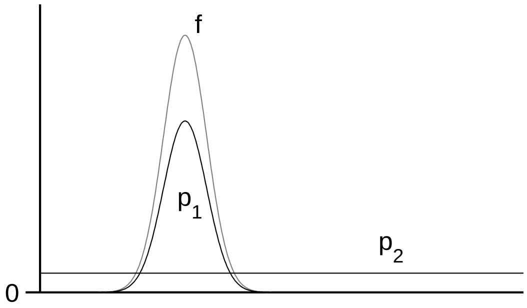


Figure 9.6: Two density functions for sampling a simple integrand.

assumptions of the multi-sample model. Instead, our strategy is to take samples using all of the given techniques, and compute weighting functions that automatically assign low weights to any irrelevant samples. In the case where one of the p_i is a good match for f, the ideal result would be to compute weighting functions such that $w_i(x) = 1$ over the whole domain, while all of the other w_j are zero. This would achieve the same end result as using p_i alone, at the expense of taking several unnecessary samples from the other p_j. However, extra sampling is unavoidable if we do not know in advance which of the given sampling techniques will work best.

We now consider how the balance heuristic can be improved, so that it performs better on low-variance problems. To do this, we study the simple test case of Figure 9.6, which shows an integrand f and two density functions p_1 and p_2 to be used for importance sampling. The density function p_1 is proportional to f, while p_2 is a constant function. For this situation, the optimal weighting functions are obviously

$$\begin{aligned} w_1^*(x) &\equiv 1\,, \\ w_2^*(x) &\equiv 0\,, \end{aligned}$$

since this would give an estimator F^* whose variance is zero.

The balance heuristic weighting functions $\hat{w}_i$ are different than the optimal ones above, and thus the balance heuristic will lead to additional variance. We now examine where this extra variance comes from, to see how it can be reduced. We start by dividing the domain

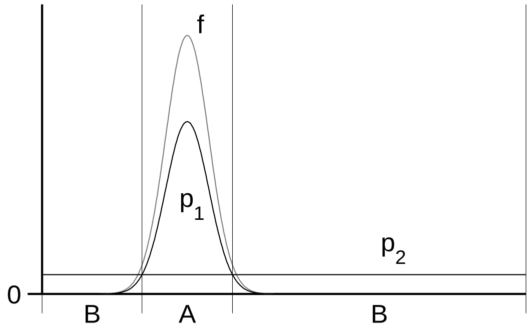


Figure 9.7: The integration domain is divided into two regions A and B. Region A represents the set of points where $p_1 > p_2$, while region B represents the points where $p_2 > p_1$. The weights computed by the balance heuristic are considered in each region separately.

into two regions A and B, as shown in Figure 9.7. Region A represents the set of points where $p_1 > p_2$, while region B represents the points where $p_2 > p_1$. We will consider the weights computed by the balance heuristic in each region separately. To simplify the discussion, we assume that $n_1 = n_2 = 1$ (i.e. a single sample is taken using each technique, and their contributions are summed).

First consider the sample from p_1, which is likely to occur in the central part of region A. Since p_1 is much larger than p_2 in this region, the sample weight $\hat{w}_1 = p_1/(p_1+p_2)$ will be close to one. This agrees with the optimal weighting function $w_1^* = 1$, as desired.

Similarly, the sample from p_2 is likely to occur in region B, where its weight $\hat{w}_2 = p_2/(p_1 + p_2)$ is close to one. Nevertheless, the contribution of this sample will be small, since the integrand f is nearly zero in region B. Therefore this situation is also close to the optimal one, in which the samples from p_2 are ignored.

However, there are two effects that lead to additional variance. First, the sample from p_1 sometimes occurs near the boundaries of region A (or even in region B), where its weight $\hat{w}_1 = p_1/(p_1+p_2)$ is significantly smaller than one. In this case, the sample makes a contribution that is noticeably smaller than the optimal value f/p_1. (Recall that p_1 is proportional to f, so that f/p_1 is the desired value μ of the integral.) In Figure 9.5, this effect shows up as occasional pixels that are darker than they should be (e.g. in the top image of column (c)).

The second problem is that the sample from p_2 sometimes occurs in region A. When this happens, its weight $\hat{w}_2 = p_2/(p_1 + p_2)$ is small. However, the contribution made by this sample is

$$\hat{w}_2 \frac{f}{p_2} = \frac{p_2}{p_1 + p_2} \frac{f}{p_2} = \frac{f}{p_1 + p_2},$$

which is approximately equal to $f/p_1 = \mu$ in this region. Since it is likely that the sample from p_1 also lies in region A (contributing another μ toward the estimate), this leads to a total estimate of approximately 2μ. In Figure 9.5(c), this effect shows up as occasional pixels that are approximately twice as bright as their neighbors.[4]

Thus, the additional noise of the balance heuristic can be attributed to two problems. First, some of the samples from p_1 have weights that are significantly smaller than one: this happens near the boundary of region A, where p_1 and p_2 have comparable magnitude. (Very few of these samples will occur in the region where $p_1 \ll p_2$, simply because p_1 is very small there.) The second problem is that some samples from p_2 make contributions of noticeable size (i.e. a significant fraction of μ). Most of these samples have small weights, because they occur in region A where $p_1 > p_2$. Some samples will also occur in the region where p_1 and p_2 have comparable magnitude; however, the samples where $p_2 \gg p_1$ do not cause any problems, since the sample contribution $f/(p_1 + p_2)$ is negligible there.

9.2.3.2 Better strategies for low-variance problems

We now present two families of combination strategies that have better performance on low-variance problems. These strategies are variations of the balance heuristic, where the weighting functions have been *sharpened* by making large weights closer to one and small weights closer to zero. This idea is effective at reducing both sources of variance described above.

The basic observation is that most samples from p_1 occur in region A, where $p_1 > p_2$. We would like all of these samples to have the optimal weight $w_1^* = 1$. Since the balance heuristic already assigns these samples a weight $\hat{w}_1 = p_1/(p_1 + p_2)$ that is greater than $1/2$, we can get closer to the optimal weighting functions by applying the sharpening strategy mentioned above. For example, one way to do this would be to set $w_1 = 1$ (and $w_2 = 0$)

[4] Note that this situation is entirely different than the "spikes" of Figure 9.5(a) and (b), which are caused by sample contributions that are hundreds of times larger than the desired mean value.

whenever $\hat{w}_1 > 1/2$.

Similarly, this idea can reduce the variance caused by samples from p_2 in region A. The optimal weight for these samples is $w_2^* = 0$, while the balance heuristic assigns them a weight $\hat{w}_2 < 1/2$, so that sharpening the weighting functions is once again an effective strategy.[5]

We now describe two different combination strategies that implement this sharpening idea, called the *cutoff heuristic* and the *power heuristic*. Each of these is actually a family of strategies, controlled by an additional parameter. For convenience in describing them, we will drop the x argument on the functions w_i and p_i, and define a new symbol q_i as the product $q_i = n_i p_i$. For example, in this notation the balance heuristic would be written as

$$\hat{w}_i = \frac{q_i}{\sum_k q_k}\,.$$

The cutoff heuristic. The *cutoff heuristic* modifies the weighting functions by discarding samples with low weight, according to a cutoff threshold $\alpha \in [0,1]$:

$$w_i = \begin{cases} 0 & \text{if } q_i < \alpha\, q_{\max} \\ \dfrac{q_i}{\sum_k \{q_k \mid q_k \geq \alpha\, q_{\max}\}} & \text{otherwise} \end{cases} \tag{9.12}$$

where $q_{\max} = \max_k q_k$. The threshold α determines how small q_i must be (compared to $q_{\max}$) before it is thrown away.

The power heuristic. The *power heuristic* modifies the weighting functions in a different way, by raising all of the weights to an exponent β, and then renormalizing:

$$w_i = \frac{q_i^\beta}{\sum_k q_k^\beta}\,. \tag{9.13}$$

[5] Note that sharpening the weighting functions is not a perfect solution for low-variance problems, since it does not address the extra variance due to samples in region B (where $p_2 > p_1$). In this region, sharpening the weighting functions has the effect of decreasing w_1 and increasing w_2, which is opposite to what is desired. The number of samples affected in this way is relatively small, however, under the assumption that most samples from p_1 occur where $p_1 \gg p_2$.

We have found the exponent $\beta = 2$ to be a reasonable value. With this choice, the sample contribution $(w_i\, f)/(n_i\, p_i)$ is proportional to p_i, so that it decreases gradually as p_i becomes smaller relative to the other p_k. (Compare this with the balance heuristic, where a sample at a given point x always makes the same contribution, no matter which sampling technique generated it.)

Notice that the balance heuristic can be obtained as a limiting case of both strategies (when $\alpha = 0$ or $\beta = 1$). These two strategies also share another limiting case, obtained by setting $\alpha = 1$ or $\beta = \infty$. This special case is called the *maximum heuristic*:

The maximum heuristic. The maximum heuristic partitions the domain into n regions, according to which function q_i is largest at each point x:

$$w_i \;=\; \begin{cases} 1 & \text{if } q_i = q_{\max} \\ 0 & \text{otherwise}\,. \end{cases} \tag{9.14}$$

In other words, samples from p_i are used to estimate the integral only in the region Ω_i where $w_i = 1$. The maximum heuristic does not work as well as the other strategies in practice; intuitively, this is because too many samples are thrown away. However, it gives some insight into the other combination strategies, and has an elegant structure.

9.2.3.3 Variance bounds

The advantage of these strategies is reduced variance when one of the p_i is a good match for f. Their performance is otherwise similar to the balance heuristic; it is possible to show they are never much worse. In particular, we have the following worst-case bounds:

Theorem 9.3. *Let f, n_i, and p_i be given, for $i = 1, \ldots, n$. Let F be any unbiased estimator of the form (9.4), and let F' be one of the estimators described above. Then the variance of F' satisfies a bound of the form*

$$V[F'] \;\le\; c\,V[F] \,+\, \left(\frac{1}{\min_i n_i} \,-\, \frac{1}{\sum_i n_i}\right) \mu^2\,,$$

where the constant c is given by the following table:

Cutoff heuristic (with threshold α)	$c \;=\; 1 + \alpha\,(n-1)$
Power heuristic (with exponent β)	$c = 1 + (1/\beta)^{1/\beta}\,((n-1)(1-1/\beta))^{1-1/\beta}$
Power heuristic (with exponent $\beta = 2$)	$c \;=\; (1/2)\,(1+\sqrt{n})$

In particular, these bounds hold when F' is compared against the unknown, optimal estimator F^*. A proof of this theorem in given in Appendix 9.A. However, the true test of these strategies is how they perform on practical problems; measurements along these lines are presented in Section 9.3.1.

9.2.4 The one-sample model

We conclude by considering a different model for how the samples are taken and combined, called the *one-sample model*. Under this model, the integral is estimated by choosing one of the n sampling techniques at random, and then taking a single sample from it. Again we consider how to minimize variance by weighting the samples, and we show that for this model the balance heuristic is optimal: no other combination technique has smaller variance.

Let p_1, . . ., p_n be the density functions for the n given sampling techniques. To generate a sample, one of the density functions p_i is chosen at random according to a given set of probabilities c_1, . . ., c_n (which sum to one). A single sample is then taken from the chosen technique. This sampling model is often used in graphics: for example, it describes algorithms such as path tracing, where sampling the BSDF may require a random choice between different techniques for the diffuse, glossy, and specular components.

As before, we consider a family of unbiased estimators for the given integral $\int_\Omega f(x)\,d\mu(x)$, where each estimator is represented by a set of weighting functions w_1, . . ., w_n. The process of choosing a sampling technique, taking a sample, and computing a weighted estimate is then expressed by the *one-sample estimator*

$$F \;=\; \frac{w_I(X_I)\,f(X_I)}{c_I\,p_I(X_I)}\,, \tag{9.15}$$

where $I \in \{1, \ldots, n\}$ is a random variable distributed according to the probabilities c_i, and

X_I is a sample from the corresponding technique p_I. This estimator is unbiased under the same conditions on the w_i discussed in Section 9.2.1.

We now consider how to choose the weighting functions w_i, to minimize the variance of the resulting estimator. We can show that for this model, the balance heuristic is optimal:

Theorem 9.4. *Let f, c_i, and p_i be given, for $i = 1, \ldots, n$. Let F be any unbiased estimator of the form (9.15), and let $\hat{F}$ be the corresponding estimator that uses the balance heuristic weighting functions (9.8). Then*

$$V[\hat{F}] \;\leq\; V[F]\,.$$

(A proof is given in Appendix 9.A.) Thus, for this sampling model the improved combination strategies of Section 9.2.3 are unnecessary.

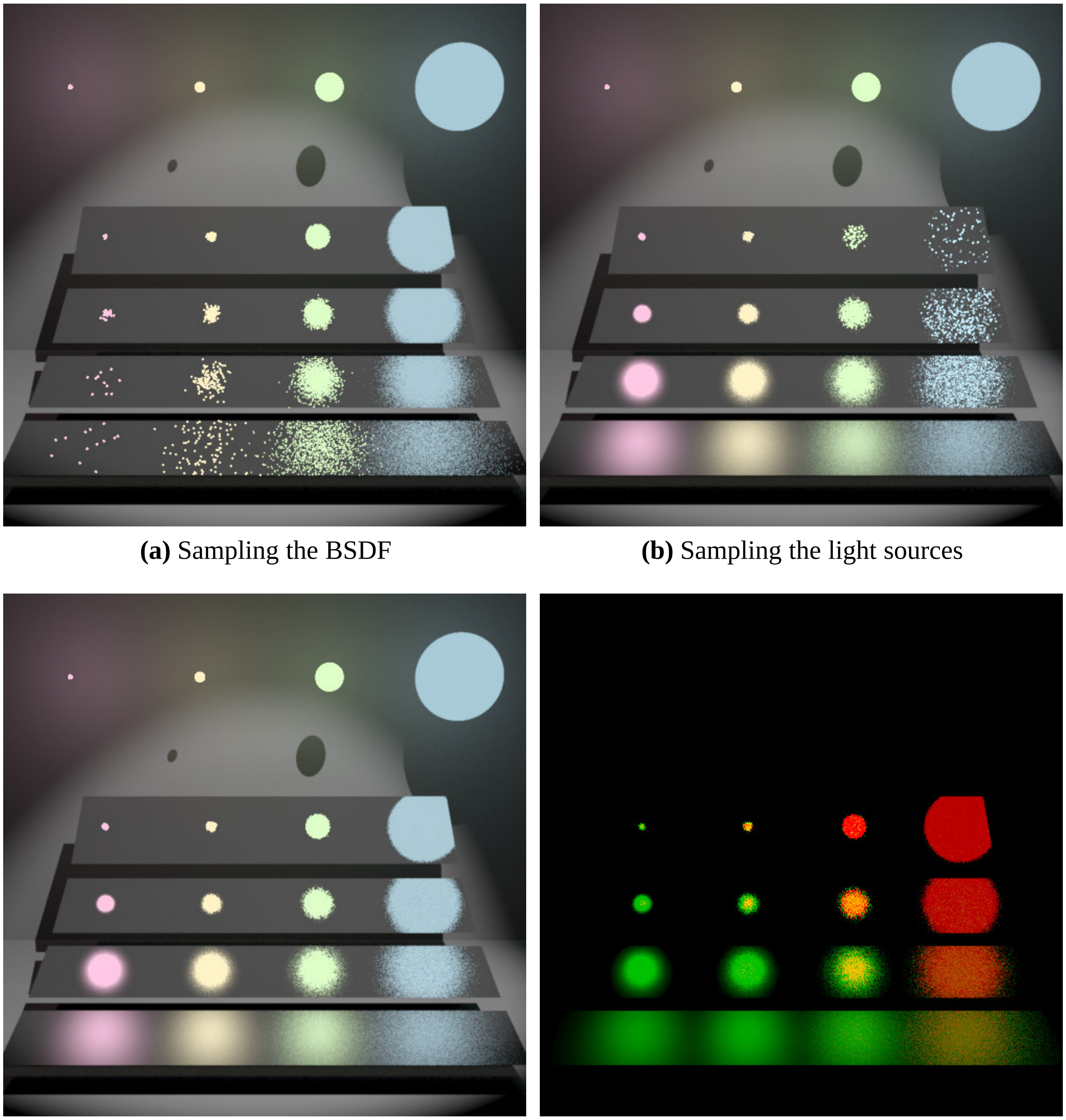
9.3 Results

In this section, we show how multiple importance sampling can be applied to two important application areas: distribution ray tracing (in particular, the glossy highlights problem from Section 9.1), and the *final gather* pass of certain light transport algorithms. (In the next chapter we will describe a more advanced example of our techniques, namely bidirectional path tracing.)

9.3.1 The glossy highlights problem

Our first test is the computation of glossy highlights from area light sources (previously described in Section 9.1). As can be seen in Figure 9.8(a) and (b), sampling the BSDF works well for sharp reflections of large light sources, while sampling the light source works well for fuzzy reflections of small light sources. In Figure 9.8(c), we have used the power heuristic with $\beta = 2$ to combine both kinds of samples. This method works very well for all light source/BSDF combinations. Figure 9.8(d) is a visualization of the weighting functions that were used to compute this image.

To compare the various combination strategies (the balance, cutoff, power, and maximum heuristics), we have measured the variance numerically as a function of the surface



(a) Sampling the BSDF

(b) Sampling the light sources

(c) The power heuristic with $\beta = 2$.

(d) The weights used by the power heuristic.

Figure 9.8: Multiple importance sampling applied to the glossy highlights problem.**(a)** and **(b)** are the images from Figure 9.2, computed by sampling the BSDF and sampling the light sources respectively. **(c)** was computed by combining the samples from (a) and (b) using the power heuristic with $\beta = 2$. Finally, **(d)** is a false-color image showing the weights used to compute (c). Red represents sampling of the BSDF, while green represents sampling of the light sources. Yellow indicates that both types of samples are assigned a significant weight.

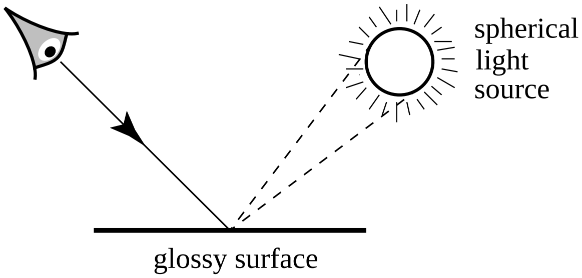
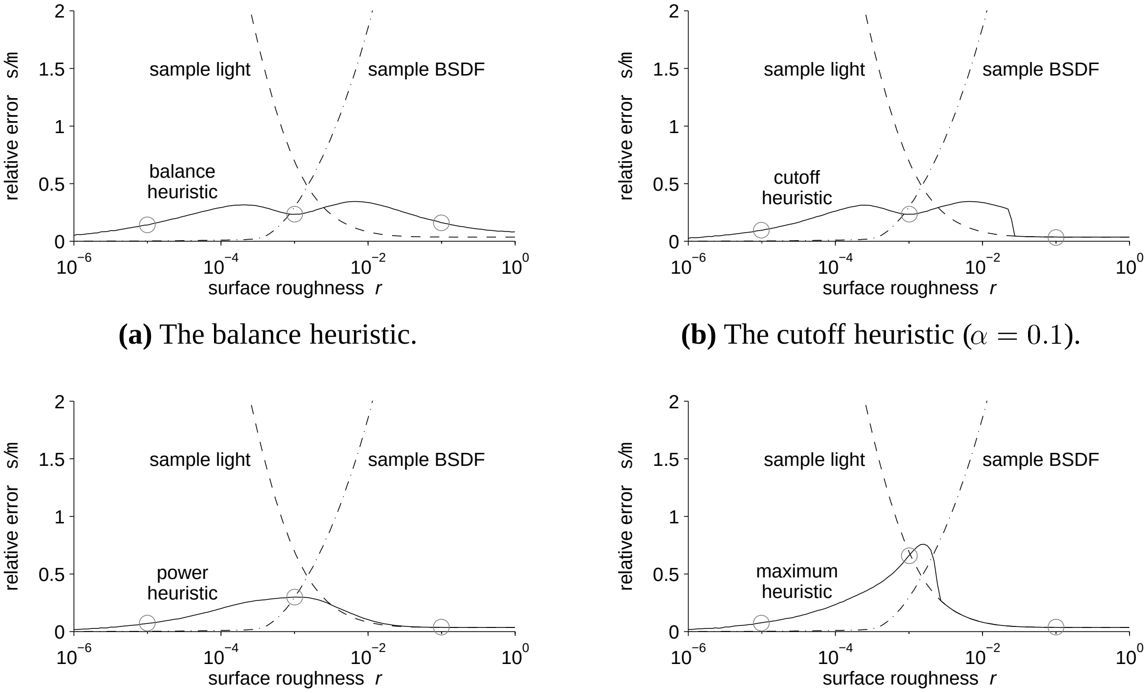


Figure 9.9: A scale diagram of the scene model used to measure the variance of the glossy highlights calculation. The glossy surface is illuminated by a single spherical light source, so that a blurred reflection of the light source is visible from the camera position. Variance was measured by taking 100,000 samples along the viewing ray shown, which intersects the center of the blurred reflection at an angle of 45 degrees. This calculation was repeated for approximately 100 different values of the surface roughness parameter$r$ (which controls how sharp or fuzzy the reflections are), in order to measure the variance as a function of surface roughness. The light source occupies a solid angle of 0.063 radians.

roughness parameter r. Figure 9.9 shows the test setup, and the results are summarized in Figure 9.10. Three curves are shown in each graph: two of them correspond to the BSDF and light source sampling techniques, while the third corresponds to the combination strategy being tested (i.e. the balance, cutoff, power, or maximum heuristic). Each graph plots the relative error σ/μ as a function of r, where σ is the standard deviation of a single sample, and μ is the mean.

Notice that all four combination strategies yield a variance that is close to the minimum of the two other curves (on an absolute scale). This is in accordance with Theorem 9.2, which guarantees that the variance σ^2 of the balance heuristic is within $\mu^2/2$ of the variance obtained when either of the given sampling techniques is used on its own. The plots in Figure 9.10(a) are well within this bound.

At the extremes of the roughness axis there are significant differences among the various combination strategies. As expected, the balance heuristic (a) performs worst at the extremes, since the other strategies were specifically designed to have better performance in this case (i.e. the case when one of the given sampling techniques is an excellent match for the integrand). The power heuristic (c) with $\beta = 2$ works especially well over the entire range of roughness values.



(a) The balance heuristic.

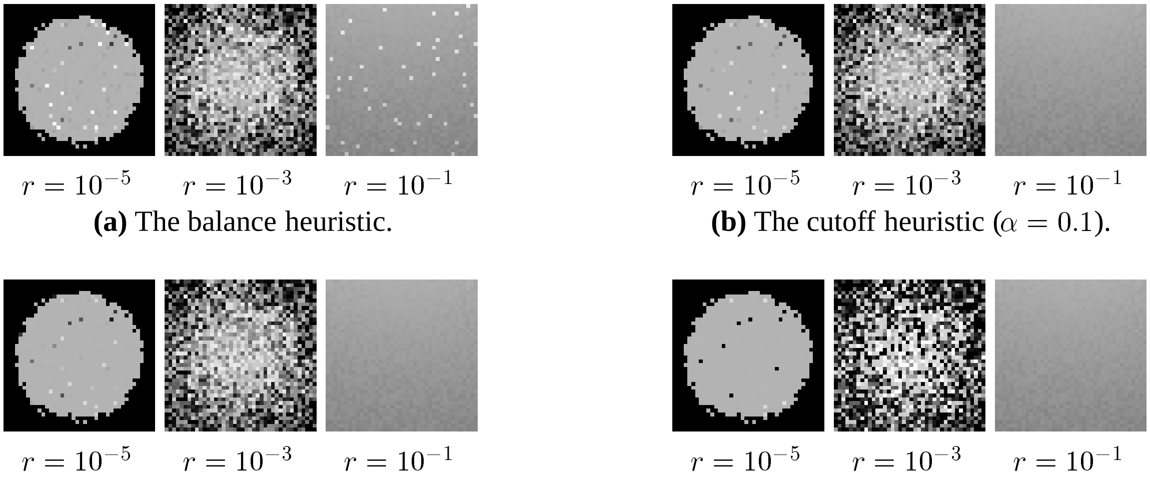
(b) The cutoff heuristic ($\alpha = 0.1$).

(c) The power heuristic ($\beta = 2$).

(d) The maximum heuristic.

Figure 9.10: Variance measurements for the glossy highlights problem using different combination strategies. Each graph plots the relative errorσ/μ as a function of the surface roughness parameter r (where σ^2 represents the variance of a single sample, andμ is the mean). A fixed size, spherical light source was used (as shown in Figure 9.9). The three curves in each graph correspond to sampling the BSDF, sampling the light source, and a weighted combination of both sample types using the (a) balance, (b) cutoff, (c) power, and (d) maximum heuristics. (The three small circles on each graph are explained in Figure 9.11.)

Figure 9.11 shows how these numerical measurements translate into actual image noise. Each image shows a glossy reflection of a spherical light source, using the same test setup as for the graphs (see Figure 9.9). The three images in each group were computed using different parameter values (namely $r = 10^{-5}$, $r = 10^{-3}$, and $r = 10^{-1}$), which causes the reflected light source to be blurred by varying amounts. The noise levels in these images should be compared against the corresponding circled variance measurements in the graphs of Figure 9.10. Notice that the cutoff, power, and maximum heuristics substantially reduce the noise at the extremes of the roughness axis.



(a) The balance heuristic.

(b) The cutoff heuristic ($\alpha = 0.1$).

(c) The power heuristic ($\beta = 2$).

(d) The maximum heuristic.

Figure 9.11: Each of these test images corresponds to one of the circled points on the variance curves of Figure 9.10. Their purpose is to compare the different combination strategies visually, by showing how the numerical variance measurements translate into actual image noise. Each image shows a glossy reflection of a spherical light source, as shown in Figure 9.9 (the same test setup used for the graphs). The three images in each group were computed using different values of the surface roughness parameter$r$ (with one sample per pixel, box filtered), which causes the reflected light source to be blurred by varying amounts (the sharpest reflections are on the left). The noise levels in these images should be compared against the corresponding circled variance measurements shown in Figure 9.10. Notice in particular that the improved weighting strategies (b), (c), and (d) give much better results when $r = 10^{-1}$, and significantly better results when$r = 10^{-5}$.

In all cases, the additional cost of multiple importance sampling was small. The total time spent evaluating probabilities and weighting functions in these tests was less than 5%. For scenes of realistic complexity, the overhead would be even smaller (as a fraction of the total computation time).

We have also made measurements of the cutoff and power heuristics using other values of α and β (which represent the cutoff threshold and the exponent, respectively). In fact, the graphs in Figure 9.10 already give results for three values of α and β each, since the balance and maximum heuristics are limiting cases of the other two strategies. Specifically, the cutoff heuristic for $\alpha = 0$, $\alpha = 0.1$, and $\alpha = 1$ is represented by graphs (a), (b), and (d), while the power heuristic for $\beta = 1$, $\beta = 2$, and $\beta = \infty$ is represented by graphs (a), (c), and (d). The graphs we have obtained at other parameter values are not significantly different than what would be obtained by interpolating these results.

Related work. Shirley & Wang [1992] have also compared BRDF and light source sampling techniques for the glossy highlights problem. They analyze a specific Phong-like BRDF and a specific light source sampling method, and derive an expression for when to switch from one to the other (as a function of the Phong exponent, and the solid angle occupied by the light source). Their methods work well, but they apply only to this particular BSDF and sampling technique. In contrast, our methods work for arbitrary BSDF's and sampling techniques, and can combine samples from any number of techniques.

9.3.2 The final gather problem

In this section we consider a simple test case motivated by multi-pass light transport algorithms. These algorithms typically compute an approximate solution using the finite element method, followed by one or more ray tracing passes to replace parts of the solution that are poorly approximated or missing. For example, some radiosity algorithms use a *local pass* or *final gather* to recompute the basis function coefficients more accurately.

We examine a variation called *per-pixel final gather*. The idea is to compute an approximate radiosity solution, and then use it to illuminate the visible surfaces during a ray tracing pass [Rushmeier 1988, Chen et al. 1991]. Essentially, this type of final gather is equivalent to ray tracing with many area light sources (one for each patch, or one for each link in a hierarchical solution). That is, we would like to evaluate the scattering equation (9.2) where L_{e} is given by the initial radiosity solution.

As with the glossy highlights example, there are two common sampling techniques. The brightest patches are typically reclassified as "light sources" [Chen et al. 1991], and are sampled using direct lighting techniques. For example, this might consist of choosing one sample for each light source patch, distributed according to the emitted power per unit area. The remaining patches are handling by sampling the BSDF at the point intersected by the viewing ray, and casting rays out into the scene. If any ray hits a light source patch, the contribution of that ray is set to zero (to avoid counting the light source patches twice). Within

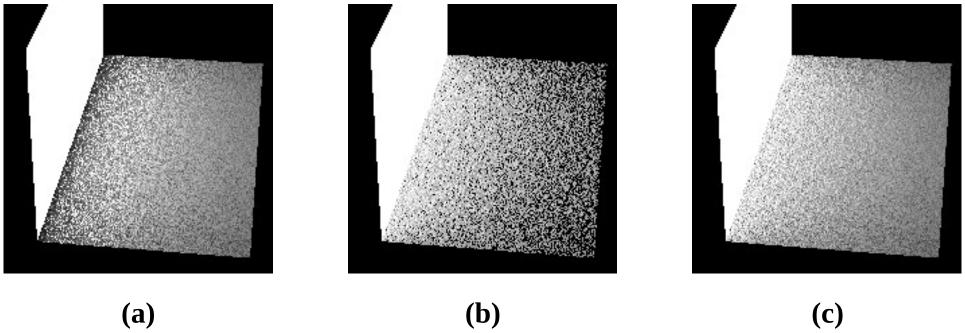


Figure 9.12: A simple test scene consisting of one area light source (i.e. a bright patch, in the radiosity context), and an adjacent diffuse surface. The images were computed by **(a)** sampling the light source according to emitted power, using $n_1 = 3$ samples per pixel, **(b)** sampling the BSDF with respect to the projected solid angle measure, using $n_2 = 6$ samples per pixel, and **(c)** a weighted combination of samples from (a) and (b) using the power heuristic with $\beta = 2$.

our framework for combining sampling techniques, this is clearly a partitioning of the integration domain into two regions.

Given some classification of patches into light sources and non-light sources, we consider alternative ways of combining the two types of samples. To test our combination strategies, we used the extremely simple test scene of Figure 9.12, which consists of a single area light source and an adjacent diffuse surface. Image (a) was computed by sampling the light source according to emitted power, while image (b) was computed by sampling the BSDF and casting rays out into the scene. Twice as many samples were taken in image (b) than (a); in practice this ratio would be substantially higher (i.e. the number of directional samples, compared to the number of samples for any one light source).

Notice that the sampling technique in Figure 9.12(a) does not work well for points near the light source, since this technique does not take into account the $1/r^2$ distance term of the scattering equation (9.2). On the other hand Figure 9.12(b) does not work well for points far away from the light source, where the light subtends a small solid angle. In Figure 9.12(c), the power heuristic is used to combine samples from (a) and (b). As expected, this method performs well at all distances. Although (c) uses more samples (the sum of (a) and (b)), this still is a valid comparison with the partitioning approach described above (which also uses

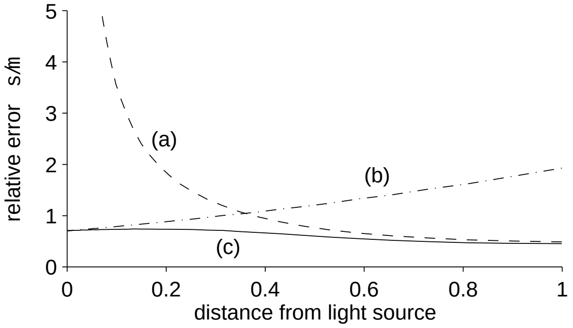


Figure 9.13: A plot of the relative error σ/μ, as a function of the distance from the light source. Three curves are shown, corresponding to the three images of Figure 9.12. The curves have been normalized to show the variance when $n_1 = 1$ and $n_2 = 2$ (the same ratio of samples used in Figure 9.12).

both kinds of samples).

Variance measurements for these experiments are plotted in Figure 9.13. There are three curves, corresponding to the three images of Figure 9.12. Each curve plots the relative error σ/μ as a function of the distance from the light source. Notice that the combined curve (c) always lies below the other two curves, indicating that both kinds of samples are being used effectively. Also, notice that unlike Figure 9.10, the variance curves do not approach zero at the extremes of the distance axis (not even as the distance d goes to infinity). This implies that neither of the given sampling techniques is an excellent match for the integrand, so that the balance, cutoff, power, and maximum heuristics all perform similarly on this problem. This is why we have only shown one graph, rather than four.

9.4 Discussion

There are several important issues that we have not yet discussed.

We start by considering how multiple importance sampling is related to the classical Monte Carlo techniques of importance sampling and stratified sampling. We show that it unifies and extends these ideas within a single sampling model. Next, we consider the problem of choosing the n_i, i.e. how to allocate a fixed number of samples among the given sampling techniques. We argue that this decision is not nearly as important as choosing the weighting functions appropriately. Finally, we discuss some special issues that arise in direct lighting problems.

9.4.1 Relationship to classical Monte Carlo techniques

Multiple importance sampling can be viewed as a generalization of both importance sampling and stratified sampling. It extends importance sampling to the case where more than one sampling technique is used, while it extends stratified sampling to the case where the strata are allowed to overlap each other. From the latter point of view, multiple importance sampling consists of taking one or more samples in each of n given regions Ω_i. These regions do not need to be disjoint; the only requirement is that their union must cover the portion of the domain where f is non-zero.

This generalization of stratified sampling is useful, especially when the integrand is a sum of several quantities. A good example in graphics is the BSDF, which is often written as a sum of diffuse, glossy, and specular components (for reflection and/or transmission). The process of taking one or more samples from each component is essentially a form of stratified sampling, where the strata overlap.

When stratified sampling is generalized in this way, however, there is more than one way to compute an unbiased estimate of the integral (since when two strata overlap, samples from either or both strata can be used). To address this, multiple importance sampling assigns an explicit representation to each possible unbiased estimator (as a set of weighting functions w_i). Furthermore it provides a reasonable way to select one of these estimators, by showing that certain estimators perform well compared to all the rest.

9.4.2 Allocation of samples among the techniques

In this section, we consider how to choose the number of samples that are taken using each technique p_i. We show that this decision is not as important as it might seem at first: no strategy is that much better than that of simply setting all the n_i equal.

To see this, suppose that a total of N samples will be taken, and that these samples must be allocated among the n sampling techniques. Let F be an estimator that allocates these samples in any way desired (provided that $\sum_i n_i = N$), and uses any weighting functions desired (provided that F is unbiased). On the other hand, let $\hat{F}$ be the estimator that takes an equal number of samples from each p_i, and combines them using the balance heuristic. Then it is straightforward to show that

$$V[\hat{F}] \;\le\; n\, V[F] + \frac{n-1}{N}\, \mu^2$$

where as usual, $\mu = E[F]$ is the quantity to be estimated (see Theorem 9.5 in Appendix 9.A for a proof).

According to this result, changing the n_i can improve the variance by at most a factor of n, plus a small additive term. In contrast, a poor choice of the w_i can increase variance by an arbitrary amount. Thus, the sample allocation is not as important as choosing a good combination strategy.

Furthermore, the sample allocation is often controlled by other factors, so that the optimal sample allocation is irrelevant. For example, consider the glossy highlights problem. In a distribution ray tracer, the samples used to estimate the glossy highlights are also used for other purposes: e.g. the light source samples are used to estimate the diffuse shading of the surface, while the BSDF samples are used to compute glossy reflections of ordinary, non-light-source objects. Often these other purposes will dictate the number of samples taken, so that the sample allocation for the glossy highlights calculation cannot be chosen arbitrarily. On the other hand, by computing an appropriate weighted combination of the samples that need to be taken anyway, we can reduce the variance of the highlight calculation essentially for free.

Similarly, the sample allocation is also constrained in bidirectional path tracing. In this case, it is for efficiency reasons: it is more efficient to take one sample from all the techniques at once, rather than taking different numbers of samples using each strategy. (This will be discussed further in Chapter 10.)

9.4.3 Issues for direct lighting problems

The glossy highlights and final gather test cases are both examples of direct lighting problems. They differ only in the terms of the scattering equation that cause high variance: in the case of glossy highlights, it was the BSDF and the emission function L_{e}, while for the final gather problem it was the $1/r^2$ distance factor.

Although there are more sophisticated techniques for direct lighting that take into account more factors of the scattering equation [Shirley et al. 1996], it is still useful to combine several kinds of samples. There are several reasons for this. First, sophisticated sampling strategies are generally designed for a specific light source geometry (e.g. the light source must be a triangle or a sphere). Second, they are often expensive: for example, taking a sample may involve numerical inversion of a function. Third, none of these strategies is perfect: there are always some factors of the scattering equation that are not included in the approximation (e.g. virtually all direct lighting strategies do not consider the BSDF or visibility factors). Thus, in parts of the scene where these unconsidered factors are dominant, it can be more efficient to use a simpler technique such as sampling the BSDF. Thus, combining samples from two or more techniques can make direct lighting calculations more robust.

9.5 Conclusions and recommendations

As we have shown, multiple importance sampling can substantially reduce the variance of Monte Carlo rendering calculations. These techniques are practical, and the additional cost is small — less than 5% of the time in our tests was spent evaluating probabilities and weighting functions. There are also good theoretical reasons to use these methods, since we have shown strong bounds on their performance relative to all other combination strategies.

For most Monte Carlo problems, the balance heuristic is an excellent choice for a combination strategy: it has the best theoretical bounds, and is the simplest to implement. The additional variance term of $(1/\min_i n_i - 1/N)\,\mu^2$ is not an issue for integration problems of reasonable complexity, because it is unlikely that any of the given density functions p_i will be an excellent match for f. Under these circumstances, even the optimal combination F^* has considerable variance, so that the maximum improvement that can be obtained by using some other strategy instead of the balance heuristic is a small fraction of the total.

On the other hand, if it is possible that the given integral is a low-variance problem (i.e. one of the p_i is good match for f), then the power heuristic with $\beta = 2$ is an excellent choice. It performs similarly to the balance heuristic overall, but gives better results on low-variance problems (which is exactly the case where better performance is most noticeable). Direct lighting calculations are a good example of where this optimization is useful.

In effect, multiple importance sampling provides a new viewpoint on Monte Carlo integration. Unlike ordinary importance sampling, where the goal is to find a single “perfect” sampling technique, here the goal is to find a set of techniques that *cover* the important features of the integrand. It does not matter if there are a few bad sampling techniques as well — some effort will be wasted in sampling them, but the results will not be significantly affected. Thus, multiple importance sampling gives a recipe for making Monte Carlo software more reliable: whenever there is some situation that is not handled well, then we can simply add another sampling technique designed for that situation alone. We believe that there are many applications that could benefit from this approach, both in computer graphics and elsewhere.

Appendix 9.A Proofs

Proof of Theorem 9.2 (from p. 264). Let $F_{i,j}$ be the random variable

$$F_{i,j} \;=\; \frac{w_i(X_{i,j})\, f(X_{i,j})}{p_i(X_{i,j})}\,,$$

and let μ_i be its expected value

$$\begin{aligned} \mu_i &= E[F_{i,j}] \\ &= \int_\Omega w_i(x)\, f(x)\, d\mu(x) \end{aligned}$$

(which does not depend on j). We can then write the variance of F as

$$\begin{aligned} V[F] &= V\left[\sum_{i=1}^{n} \frac{1}{n_i} \sum_{j=1}^{n_i} F_{i,j}\right] \\ &= \sum_{i=1}^{n} \frac{1}{n_i^2} \sum_{j=1}^{n_i} V[F_{i,j}] \\ &= \left(\sum_{i=1}^{n} \frac{1}{n_i^2} \sum_{j=1}^{n_i} E[F_{i,j}^2]\right) - \left(\sum_{i=1}^{n} \frac{1}{n_i^2} \sum_{j=1}^{n_i} E[F_{i,j}]^2\right) \\ &= \left(\sum_{i=1}^{n} \frac{1}{n_i^2} \sum_{j=1}^{n_i} \int_\Omega \frac{w_i^2(x)\, f^2(x)}{p_i^2(x)}\, p_i(x)\, d\mu(x)\right) - \left(\sum_{i=1}^{n} \frac{1}{n_i^2}\, n_i\, \mu_i^2\right) \\ &= \left(\int_\Omega \sum_{i=1}^{n} \frac{w_i^2(x)\, f^2(x)}{n_i\, p_i(x)}\, d\mu(x)\right) - \left(\sum_{i=1}^{n} \frac{1}{n_i}\, \mu_i^2\right). \end{aligned} \tag{9.16}$$

Notice that there are no covariance terms, because the $X_{i,j}$ are sampled independently.

We will bound the two parenthesized expressions separately. To minimize the first expression

$$\int_\Omega \sum_{i=1}^{n} \frac{w_i^2(x)\, f^2(x)}{n_i\, p_i(x)}\, d\mu(x)\,, \tag{9.17}$$

it is sufficient to minimize the integrand at each point x separately. Noting that $f^2(x)$ is a constant and dropping x from our notation, we must minimize

$$\sum_{i=1}^{n} \frac{w_i^2}{n_i\, p_i}$$

subject to the condition $\sum_i w_i = 1$. Using the method of Lagrange multipliers, the minimum value is attained when all $n+1$ partial derivatives of the expression

$$\sum_i \frac{w_i^2}{n_i\,p_i} + \lambda\left(\sum_i w_i - 1\right)$$

are zero. This yields n equations of the form $-2\,w_i \;=\; n_i\,p_i\,\lambda$, together with constraint $\sum_i w_i = 1$. The solution of these equations is

$$\hat{w}_i \;=\; \frac{n_i\,p_i}{\sum_k n_k\,p_k}$$

(the balance heuristic). Thus no other combination strategy can make the first variance term of (9.16) any smaller.

We now consider the second variance term of (9.16), namely

$$\sum_{i=1}^{n} \frac{1}{n_i}\,\mu_i^2\,.$$

We will prove an upper bound of $(1/\min_i n_i)\,\mu^2$ and a lower bound of $(1/\sum_i n_i)\,\mu^2$, such that these bounds hold for any functions w_i. (Recall that $\mu = E[F]$ is the quantity to be estimated.) Combining this with the previous result, we immediately obtain the theorem.

For the upper bound, we have

$$\sum_i \frac{1}{n_i}\,\mu_i^2 \;\le\; \frac{1}{\min_i n_i}\sum_i \mu_i^2 \;\le\; \frac{1}{\min_i n_i}\left(\sum_i \mu_i\right)^2 \;=\; \frac{1}{\min_i n_i}\,\mu^2\,,$$

where the second inequality holds because all the μ_i are non-negative.

For the lower bound, we minimize $\sum_i \mu_i^2/n_i$ subject to the constraint $\sum_i \mu_i \;=\; \mu$. Using the method of Lagrange multipliers, the minimum is attained when all $n+1$ partial derivatives of the expression

$$\sum_i \frac{\mu_i^2}{n_i} + \lambda\left(\sum_i \mu_i - \mu\right)$$

are zero. This yields $n+1$ equations whose solution is $\mu_i \;=\; (n_i/\sum_k n_k)\,\mu$, so that the minimum value of the second variance term of (9.16) is

$$\sum_i \frac{1}{n_i}\left(\frac{n_i}{\sum_k n_k}\,\mu\right)^2 \;=\; \frac{1}{\sum_k n_k}\,\mu^2$$

as desired. ∎

Proof of Theorem 9.3 (from p. 274). According to the arguments of the previous theorem, it is sufficient to prove a bound of the form

$$\sum_i \frac{w_i^2(x)\, f^2(x)}{n_i\, p_i(x)} \;\le\; c \sum_i \frac{\hat{w}_i^2(x)\, f^2(x)}{n_i\, p_i(x)}$$

at each point x, where the w_i are the weighting functions given by one of the heuristics of Theorem 9.3, and the $\hat{w}_i$ are given by the balance heuristic. Dropping the argument$x$, letting $q_i = n_i p_i$, and substituting the definition

$$\hat{w}_i = \frac{q_i}{\sum_k q_k}\,,$$

we must show that

$$\sum_i \frac{w_i^2}{q_i} \;\le\; c \sum_i \frac{1}{q_i} \left(\frac{q_i}{\sum_k q_k} \right)^2 \;=\; \frac{c}{\sum_k q_k}\,. \tag{9.18}$$

For the cutoff heuristic, we have

$$\begin{aligned}\sum_i \frac{w_i^2}{q_i} \;&=\; \sum_{i \mid q_i \ge \alpha\, q_{\max}} \frac{1}{q_i} \left(\frac{q_i}{\sum_{k \mid q_k \ge \alpha\, q_{\max}} q_k} \right)^2 \\ &=\; \frac{1}{\sum_{i \mid q_i \ge \alpha\, q_{\max}} q_i}\,.\end{aligned}$$

Thus according to (9.18), we must find a value of$c$ such that

$$\begin{aligned} & \frac{1}{\sum_{i \mid q_i \ge \alpha\, q_{\max}} q_i} \;\le\; \frac{c}{\sum_k q_k} \\ \iff\quad & c \sum_{i \mid q_i \ge \alpha\, q_{\max}} q_i \;\ge\; \sum_k q_k \\ \iff\quad & (c-1) \sum_{i \mid q_i \ge \alpha\, q_{\max}} q_i \;\ge\; \sum_k q_k - \sum_{i \mid q_i \ge \alpha\, q_{\max}} q_i \\ \iff\quad & c - 1 \;\ge\; \frac{\sum_{i \mid q_i < \alpha\, q_{\max}} q_i}{\sum_{i \mid q_i \ge \alpha\, q_{\max}} q_i}\,. \end{aligned}$$

To find a value of c for which this is true, it is sufficient to find an upper bound for the right-hand side. Examining the numerator and denominator, we have

$$\frac{\sum_{i \mid q_i < \alpha\, q_{\max}} q_i}{\sum_{i \mid q_i \ge \alpha\, q_{\max}} q_i} \;\le\; \frac{(n-1)\, \alpha\, q_{\max}}{q_{\max}} \;=\; \alpha\,(n-1)\,.$$

Thus the variance claim is true whenever$c \;\ge\; 1 + \alpha\,(n-1)$, as desired.

Next, we consider the power heuristic with the exponent$\beta \;=\; 2$. Starting with the inequality

(9.18), we have

$$\sum_i \frac{w_i^2}{q_i} \;=\; \sum_i \frac{1}{q_i} \left(\frac{q_i^2}{\sum_k q_k^2} \right)^2 \;=\; \frac{\sum_i q_i^3}{\left(\sum_k q_k^2\right)^2} \,. \tag{9.19}$$

Thus we must find a value of c such that

$$\begin{aligned} & \frac{\sum_i q_i^3}{\left(\sum_k q_k^2\right)^2} \;\le\; \frac{c}{\sum_k q_k} \\ \Longleftrightarrow \qquad & \left(\sum_i q_i\right) \left(\sum_i q_i^3\right) \;\le\; c \left(\sum_k q_k^2\right)^2 \,. \end{aligned} \tag{9.20}$$

Notice that this inequality is unchanged if all the q_i are scaled by a constant factor. Thus without loss of generality we can assume that

$$\sum_i q_i^2 \;=\; \sum_i q_i \,, \tag{9.21}$$

so that our goal reduces to finding a value of c such that

$$c \;\ge\; \left(\sum_i q_i^3\right) \,/\, \left(\sum_i q_i^2\right) \,.$$

We proceed as before, by finding an upper bound for the right-hand side. Without loss of generality, let q_1 be the largest of the q_i. Observing that

$$\left(\sum_i q_i^3\right) \,/\, \left(\sum_i q_i^2\right) \;\le\; \max_i q_i \;=\; q_1 \,,$$

it is sufficient to find an upper bound for q_1. According to (9.21), we have

$$q_1^2 - q_1 \;=\; \sum_{i=2}^{n} q_i - q_i^2 \,.$$

Letting S denote the quantity on the right-hand side, we have $S \;\le\; (1/4)\,(n{-}1)$, since the maximum value of $q_i - q_i^2$ is attained when $q_i = 1/2$. Thus using the quadratic formula, we have

$$\begin{aligned} & q_1^2 - q_1 && \le \;\; (1/4)\,(n-1) \\ \Longrightarrow \qquad & q_1 && \le \;\; (1/2)\,(1 + \sqrt{(-1)^2 + 4\,(1/4)\,(n-1)}) \\ & && = \;\; (1/2)\,(1+\sqrt{n})\,. \end{aligned}$$

Thus, the original inequality (9.18) is true for any value of c larger than this.

For an exponent in the range $1 \le \beta \le \infty$, the argument is similar. We find that

$$\sum_i \frac{w_i^2}{q_i} \;=\; \left(\sum_i q_i^{2\beta-1}\right) / \left(\sum_k q_k^\beta\right)^2$$

(compare this with (9.19)), and we must find a value of c for which

$$\left(\sum_i q_i\right)\left(\sum_i q_i^{2\beta-1}\right) \;\le\; c\left(\sum_k q_k^\beta\right)^2$$

(compare with (9.20)). By scaling all the q_i by a constant factor, we can assume without loss of generality that

$$\sum_i q_i^\beta \;=\; \sum_i q_i\,, \tag{9.22}$$

so that we must find a value of c that satisfies

$$c \;\ge\; \frac{\sum_i q_i^{2\beta-1}}{\sum_i q_i^\beta}\,.$$

Letting q_1 be the largest of the q_i, a trivial upper bound for the right-hand side is $q_1^{\beta-1}$. Our strategy will be to find an upper bound for this quantity, in terms of β and n.

Defining

$$S \;=\; \sum_{i=2}^n q_i - q_i^\beta \tag{9.23}$$

and using the restriction (9.22), we have

$$\begin{array}{rrcl} & q_1^\beta - q_1 &=& S \\ \Longrightarrow & q_1^{\beta-1} &=& 1 + S/q_1\,. \end{array} \tag{9.24}$$

To find an upper bound for the right-hand side, we must find an upper bound for S, and a lower bound for q_1. For q_1, we have

$$\begin{array}{rrcl} & q_1^\beta &=& q_1 + S \\ \Longrightarrow & q_1^\beta &\ge& S \\ \Longrightarrow & q_1 &\ge& S^{1/\beta}\,, \end{array}$$

and inserting this in (9.24) yields

$$q_1^{\beta-1} \;\le\; 1 + S^{1-1/\beta}\,. \tag{9.25}$$

Now to find an upper bound for S, from (9.23) we have

$$S \;\le\; (n-1)\sup_{x\ge 0}(x - x^\beta)\,. \tag{9.26}$$

The maximum value of $f(x) = x - x^\beta$ occurs when $f'(x) = 0$, yielding

$$\begin{aligned} 1 - \beta x^{\beta-1} &= 0 \\ \Longrightarrow \qquad x &= (1/\beta)^{1/(\beta-1)} \,. \end{aligned}$$

Substituting this in (9.26), we obtain an upper bound for$S$:

$$\begin{aligned} S &\le (n-1)\left((1/\beta)^{1/(\beta-1)} - (1/\beta)^{\beta/(\beta-1)}\right) \\ &= (n-1)\,(1/\beta)^{1/(\beta-1)}\,(1 - 1/\beta) \,. \end{aligned}$$

Finally, we combine this with (9.25) to obtain an upper bound for$q_1^{\beta-1}$:

$$\begin{aligned} q_1^{\beta-1} &\le 1 + S^{1-1/\beta} \\ &\le 1 + \left[(n-1)\,(1/\beta)^{1/(\beta-1)}\,(1-1/\beta)\right]^{(\beta-1)/\beta} \\ &= 1 + (1/\beta)^{1/\beta}\,((n-1)(1-1/\beta))^{1-1/\beta} \end{aligned}$$

as desired.

Notice that for the case $\beta = 2$, this argument gives a bound of

$$c = (1/2)\,(2 + \sqrt{n-1}) \,,$$

which is slightly larger than the bound of$c = (1/2)\,(1 + \sqrt{n})$ previously shown. ■

Tightness of the bounds. For the cutoff heuristic, the constant$c$ cannot be reduced for any value of α. (To see this, let $q_1 = 1$, and let $q_i = \alpha - \epsilon$ for all $i = 2, \ldots, n$, where $\epsilon > 0$ can be made as small as desired.)

For the power heuristic, the given bounds are tight when$\beta = 1$ and $\beta = \infty$ (corresponding to the balance and maximum heuristics respectively, and yielding the constants$c = 1$ and $c = n$. For other values of β, the bounds are not tight. However, they are not as loose as might be expected, considering the simplifications that were made to obtain them. For example, let$q_1 = 1 + \sqrt{n}$, and $q_i = 1$ for $i = 2, \ldots, n$. Substituting these values into the defining equation (9.20) for$c$, we obtain

$$c = (1/4)\,(3 + \sqrt{n}) \,.$$

Thus, the bounds $c = (1/2)\,(1 + \sqrt{n})$ and $c = (1/2)\,(2 + \sqrt{n-1})$ proven above cannot be reduced by more than a factor of two.

Proof of Theorem 9.4 (from p. 276). The variance of F is

$$V[F] = E[F^2] - E[F]^2 .$$

Since $E[F]^2 = \mu^2$ is the same for all unbiased estimators, it is enough to show that the balance heuristic minimizes the second moment $E[F^2]$. We have

$$\begin{aligned} E[F^2] &= \sum_{i=1}^{n} c_i \int_\Omega \frac{w_i^2(x)\, f^2(x)}{c_i^2\, p_i^2(x)}\, p_i(x)\, d\mu(x) \\ &= \int_\Omega \sum_{i=1}^{n} \frac{w_i^2(x)\, f^2(x)}{c_i\, p_i(x)}\, d\mu(x) . \end{aligned}$$

Except for the substitution of c_i for n_i, this expression is identical to the second moment term (9.17) that was minimized in the proof of Theorem 9.2. Thus, the balance heuristic minimizes $E[F^2]$, and we are done. ∎

The following theorem concerns the allocation of samples among the given sampling techniques. Before stating it, we first rewrite the multi-sample estimator (9.4) to allow for the possibility that some n_i are zero:

$$F = \sum_{i=1}^{n} \sum_{j=1}^{n_i} \frac{w_i(X_{i,j})\, f(X_{i,j})}{n_i\, p_i(X_{i,j})} , \tag{9.27}$$

where $n_i \geq 0$ for all i. The possibility that $n_i = 0$ also requires a modification to condition (W2) for F to be unbiased:

(W2') $w_i(x) = 0$ whenever $n_i p_i(x) = 0$.

We now have the following theorem (which was informally summarized in Section 9.4.2):

Theorem 9.5. *Let f, p_1, …, p_n, and the total number of samples N be given, where $N = kn$ for some integer k. Let F be any unbiased estimator of the form (9.27), and let $\hat{F}$ be the corresponding estimator that uses the weighting functions*

$$\hat{w}_i(x) = \frac{n_i\, p_i(x)}{\sum_k n_k\, p_k(x)}$$

(the balance heuristic), and takes an equal number of samples from each p_i. Then

$$V[\hat{F}] \;\leq\; n\, V[F] \;+\; \frac{n-1}{N}\, \mu^2 ,$$

where $\mu = E[F]$ is the quantity to be estimated.

Proof. Given any unbiased estimator F, let F^+ be the estimator that uses the same weighting functions F ($w_i^+ = w_i$), but takes an equal number of samples using each sampling technique ($n_i^+ = N/n$). We will show that $V[F^+] \le n\,V[F]$. Starting with equation (9.16) for $V[F]$, we have

$$\begin{aligned}
V[F] &= \int_\Omega \sum_{i=1}^{n} \frac{w_i^2(x)\,f^2(x)}{n_i\,p_i(x)}\,d\mu(x) \;-\; \sum_{i=1}^{n} \frac{1}{n_i}\,\mu_i^2 \\
&= \sum_{i=1}^{n} \frac{1}{n_i} \left(\int_\Omega \frac{w_i^2(x)\,f^2(x)}{p_i(x)}\,d\mu(x) \;-\; \mu_i^2 \right) \\
&\ge \sum_{i=1}^{n} \frac{1}{N} \left(\int_\Omega \frac{w_i^2(x)\,f^2(x)}{p_i(x)}\,d\mu(x) \;-\; \mu_i^2 \right) \\
&= \frac{1}{n} \sum_{i=1}^{n} \frac{1}{N/n} \left(\int_\Omega \frac{w_i^2(x)\,f^2(x)}{p_i(x)}\,d\mu(x) \;-\; \mu_i^2 \right) \\
&= \frac{1}{n}\,V[F^+]\,.
\end{aligned}$$

We now compare the variance of F^+ to the variance of $\hat{F}$. These two estimators take the same number of samples from each p_i, so that we can apply Theorem 9.2:

$$\begin{aligned}
V[\hat{F}] &\le V[F^+] \;+\; \left(\frac{1}{\min_i n_i^+} - \frac{1}{\sum_i n_i^+} \right) \mu^2 \\
&\le n\,V[F] \;+\; \left(\frac{1}{N/n} - \frac{1}{N} \right) \mu^2 \\
&= n\,V[F] + \frac{n-1}{N}\,\mu^2\,. \quad \blacksquare
\end{aligned}$$

Chapter 10

Bidirectional Path Tracing

In this chapter, we describe a new light transport algorithm called *bidirectional path tracing*. This algorithm is a direct combination of the ideas in the last two chapters: namely, expressing light transport as an integration problem, and then applying more than one importance sampling technique to evaluate it. The resulting algorithm handles arbitrary geometry and materials, is relatively simple to implement, and can handle indirect lighting problems far more efficiently and robustly than ordinary path tracing.

To sample each transport path, we generate one subpath starting from a light source, a second subpath starting from the eye, and join them together. By varying the number of vertices generated from each side, we obtain a family of sampling techniques for paths of all lengths. Each sampling technique has a different probability distribution over the space of paths, and takes into account a different subset of the factors of the integrand (i.e. the measurement contribution function). Samples from all of these techniques are then combined using multiple importance sampling.

This chapter is organized as follows. We start in Section 10.1 with an overview of the bidirectional path tracing algorithm. This is followed by a more detailed mathematical description in Section 10.2, where we derive explicit formulas for the sample contributions. Section 10.3 then discusses the issues that arise when implementing the algorithm, including how to generate the subpaths and evaluate their contributions efficiently, how to handle specular materials, and how to implement the important special cases where the light or eye subpath contains less than two vertices. In Section 10.4 we describe an important optimization to reduce the number of visibility tests, using a new technique called *efficiency-optimized Russian roulette*. Section 10.5 then presents some results and measurements of the algorithm, Section 10.6 compares our algorithm to other related work in this area, and Section 10.7 summarizes our conclusions.

10.1 Overview

Recall that according to the path integral framework of Chapter 8, each measurement can be written in the form

$$I_j \;=\; \int_\Omega f_j(\bar{x})\, d\mu(\bar{x})\,, \tag{10.1}$$

where $\bar{x} = \mathbf{x}_0 \ldots \mathbf{x}_k$ is a path, Ω is the set of such paths (of any length), μ is the area-product measure $d\mu(\bar{x}) \;=\; dA(\mathbf{x}_0) \;\cdots\; dA(\mathbf{x}_k)$, and f_j is the measurement contribution function

$$\begin{aligned} f_j(\bar{x}) \;&=\; L_{\mathrm{e}}(\mathbf{x}_0 \!\to\! \mathbf{x}_1)\, G(\mathbf{x}_0 \!\leftrightarrow\! \mathbf{x}_1)\, W_{\mathrm{e}}^{(j)}(\mathbf{x}_{k-1} \!\to\! \mathbf{x}_k) \\ &\quad\cdot \prod_{i=1}^{k-1} f_{\mathrm{s}}(\mathbf{x}_{i-1} \!\to\! \mathbf{x}_i \!\to\! \mathbf{x}_{i+1})\, G(\mathbf{x}_i \!\leftrightarrow\! \mathbf{x}_{i+1})\,. \end{aligned} \tag{10.2}$$

Bidirectional path tracing consists of a family of different importance sampling techniques for this integral. Each technique samples a path by connecting two independently generated pieces, one starting from the light sources, and the other from the eye. For example, in Figure 10.1 the *light subpath* $\mathbf{x}_0\mathbf{x}_1$ is constructed by choosing a random point $\mathbf{x}_0$ on a light source, followed by casting a ray in a random direction to find $\mathbf{x}_1$. The *eye subpath* $\mathbf{x}_2\mathbf{x}_3\mathbf{x}_4$ is constructed by a similar process starting from a random point $\mathbf{x}_4$ on the camera lens. A complete transport path is formed by concatenating these two pieces. (Note that the integrand may be zero on this path, e.g. if $\mathbf{x}_1$ and $\mathbf{x}_2$ are not mutually visible.)

By varying the number of vertices in the light and eye subpaths, we obtain a family of sampling techniques. Each technique generates paths of a specific length k, by randomly generating a light subpath with s vertices, randomly generating an eye subpath with t vertices, and concatenating them (where $k \;=\; s + t - 1$). It is important to note that there is more than one sampling technique for each path length: in fact, for a given length k it is easy to see that there are $k + 2$ different sampling techniques (by letting $s = 0, \ldots, k + 1$).

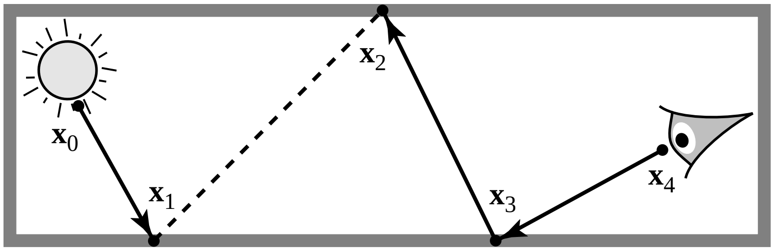
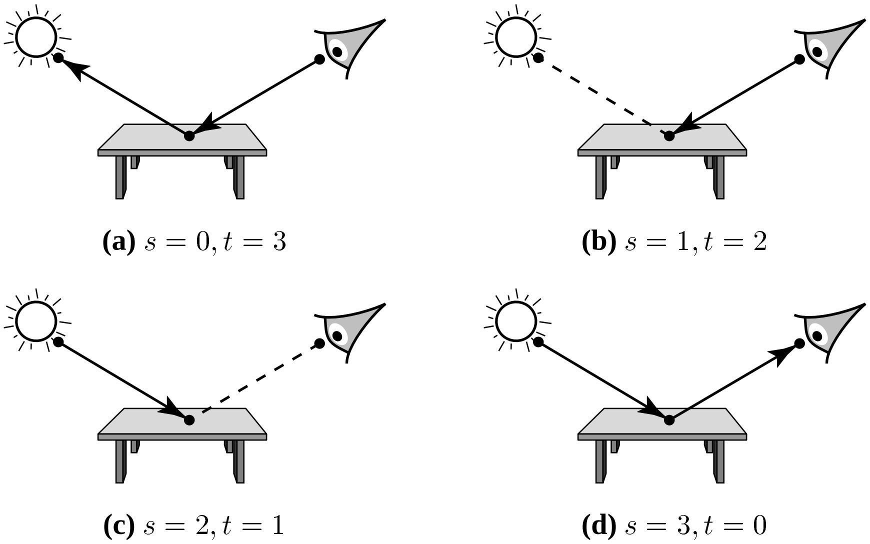


Figure 10.1: A transport path from a light source to the camera lens, created by concatenating two separately generated pieces.



(a) $s = 0, t = 3$

(b) $s = 1, t = 2$

(c) $s = 2, t = 1$

(d) $s = 3, t = 0$

Figure 10.2: The four bidirectional sampling techniques for paths of length $k = 2$. Intuitively, they can be described as **(a)** Monte Carlo path tracing with no special handling of light sources, **(b)** Monte Carlo path tracing with a direct lighting calculation, **(c)** tracing photons from the light sources and recording an image sample whenever a photon hits a visible surface, and **(d)** tracing photons and recording an image sample only when photons hit the camera lens. Note that technique (a) can only be used with an area light source, while technique (d) can only be used with a finite-aperture lens.

These techniques generate different probability distributions on the space of paths, which makes them useful for sampling different kinds of effects. For example, although technique (b) works well under most circumstances (for paths of length two), technique (a) can be superior if the table is very glossy or specular. Similarly, techniques (c) or (d) can have the lowest variance if the light source is highly directional.

Figure 10.2 illustrates the four bidirectional sampling techniques for paths of length $k = 2$.

The reason that these techniques are useful is that they correspond to different density functions $p_{s,t}$ on the space of paths. All of these density functions are good candidates for importance sampling, because they take into account different factors of the measurement contribution function f_j (as we will explain below). In practical terms, this means that each technique can efficiently sample a different set of lighting effects.

To take advantage of this, bidirectional path tracing generates samples using all of the techniques $p_{s,t}$ and combines them using multiple importance sampling. Specifically, the following estimate is computed for each measurement I_j:

$$F \;=\; \sum_{s \ge 0} \sum_{t \ge 0} \, w_{s,t}(\bar{x}_{s,t}) \, \frac{f_j(\bar{x}_{s,t})}{p_{s,t}(\bar{x}_{s,t})} \, . \tag{10.3}$$

Here $\bar{x}_{s,t}$ is a path generated according to the density function $p_{s,t}$, and the weighting functions $w_{s,t}$ represent the combination strategy being used (which is assumed to be one of the provably good strategies in Chapter 9, such as the balance heuristic). By combining samples from all the bidirectional techniques in this way, a wide variety of scenes and lighting effects can be handled well.

Efficiently generating the samples. So far, we have assumed that all the paths $\bar{x}_{s,t}$ are sampled independently, by generating a separate light and eye subpath for each one. However, in practice it is important to make the sampling more efficient. This is achieved by generating the samples in groups. For each group, we first generate a light subpath

$$\mathbf{y}_0 \ldots \mathbf{y}_{n_L - 1}$$

with n_L vertices, and an eye subpath

$$\mathbf{z}_{n_E - 1} \ldots \mathbf{z}_0$$

with n_E vertices (where $\mathbf{y}_0$ is a point on a light source, and $\mathbf{z}_0$ is a point on the camera lens). The length of each subpath is determined randomly, by defining a probability for terminating the subpath at each vertex (details are given in Section 10.3.3). We can then take samples from a whole group of techniques $p_{s,t}$ at once, by simply joining each prefix of the light subpath to each suffix of the eye subpath. The sample from $p_{s,t}$ is taken to be

$$\bar{x}_{s,t} \;=\; \mathbf{y}_0 \ldots \mathbf{y}_{s-1}\,\mathbf{z}_{t-1} \ldots \mathbf{z}_0\,,$$

which is a path with $s+t$ vertices and $k = s+t-1$ edges (where $0 \le s \le n_L$, $0 \le t \le n_E$, and $k \ge 1$). The vertices $\mathbf{y}_{s-1}$ and $\mathbf{z}_{t-1}$ are called the *connecting vertices*, and the edge between them is the *connecting edge*.

The contributions of all the samples $\bar{x}_{s,t}$ are then computed and summed according to the multi-sample estimator (10.3). In order to evaluate the contribution of each path, the visibility of the connecting edge must be tested (except when $s = 0$ or $t = 0$). If the connecting edge is obstructed, or if the BSDF at either connecting vertex does not scatter any light toward the other, then the contribution for that path is zero. (The following section gives further details.)

There is an important detail that we have not mentioned yet. Notice that we have modeled the multi-sample estimator (10.3) as a sum over an infinite number of samples, one from each bidirectional technique $p_{s,t}$. We did this because of the way that multiple importance sampling was defined: it assumes that an integer number of samples $n_{s,t}$ is taken from each sampling technique, so in this case we set $n_{s,t} = 1$ for all s, t. (Note that if we placed an upper bound on the allowable values of s and t, the result would be biased.) Of course, the strategy above does not take a sample from all of the techniques $p_{s,t}$, since there are an infinite number of them. However, notice that there is always some finite *probability* of taking a sample from each technique, no matter how large s and t are. This is because for any given values of s and t, there is some probability of generating a light subpath with $n_L \ge s$ and an eye subpath with $n_E \ge t$ (since there lengths are chosen randomly).

Formally, we can show how this corresponds to the multi-sample model as follows. First we introduce the notion of an *empty path* ϵ, which is defined to have a contribution of zero. We then re-interpret the strategy above to be method for sampling all of the techniques $p_{s,t}$ simultaneously, by defining the sample from $p_{s,t}$ to be $\bar{x}_{s,t} = \epsilon$ whenever $s > n_L$ or $t > n_E$. In other words, although the estimator (10.3) is formally a combination of samples from an infinite number of techniques, in fact all but a finite number of them will be the empty path ϵ on each evaluation, so that their contributions can be ignored. Another way of interpreting this is to say that the density functions $p_{s,t}$ are allowed to integrate to less than one, since any remaining probability can be assigned to the empty path. (Notice that having an infinite number of sampling techniques does not cause any problems when computing the weights $w_{s,t}(\bar{x}_{s,t})$, since there are only $k+2$ sampling techniques that can generate paths of any given length k.)

10.2 Mathematical formulation

In this section we derive the formulas for determining the contribution of each sample, and we show how to organize the calculations so that they can be done efficiently.

Letting $\bar{x}_{s,t}$ be the sample from technique $p_{s,t}$, we must evaluate its contribution

$$C_{s,t} \;\equiv\; w_{s,t}(\bar{x}_{s,t})\,\frac{f_j(\bar{x}_{s,t})}{p_{s,t}(\bar{x}_{s,t})}$$

to the estimator (10.3), which can be rewritten as

$$F \;=\; \sum_{s \geq 0} \sum_{t \geq 0} C_{s,t}\,.$$

We will evaluate this contribution in several stages. First, we define the *unweighted contribution $C^*_{s,t}$* as

$$C^*_{s,t} \;\equiv\; \frac{f_j(\bar{x}_{s,t})}{p_{s,t}(\bar{x}_{s,t})}\,.$$

We will show how to write this as a product

$$C^*_{s,t} \;=\; \alpha^L_s\, c_{s,t}\, \alpha^E_t\,,$$

where the factor α^L_s depends only on the light subpath, α^E_t depends only on the eye subpath, and $c_{s,t}$ depends only on the connecting edge $\mathbf{y}_{s-1}\mathbf{z}_{t-1}$. The weighted contribution then has the form

$$C_{s,t} \;=\; w_{s,t}\, C^*_{s,t}\,,$$

where $w_{s,t}$ depends on the probabilities with which all the other sampling techniques generate the given path $\bar{x}_{s,t}$.

We now discuss how to compute these factors in detail.

The density $\mathbf{p_{s,t}}$. We start by showing how to compute the probability density

$$p_{s,t} \;\equiv\; p_{s,t}(\bar{x}_{s,t})$$

with which the path $\bar{x}_{s,t}$ was generated. As previously discussed in Chapter 8.2, this is simply the product of the densities $P_A(\mathbf{x}_i)$ with which the individual vertices are generated (measured with respect to surface area). The vertex $\mathbf{y}_0$ is chosen directly on the surface of a light source, so that $P_A(\mathbf{y}_0)$ can be computed directly (and similarly for $\mathbf{z}_0$).

The remaining vertices $\mathbf{y}_i$ are chosen by sampling a direction and casting a ray from the current subpath endpoint $\mathbf{y}_{i-1}$. We let $P_{\sigma^\perp}(\mathbf{y}_{i-1} \to \mathbf{y}_i)$ denote the density for choosing the direction from $\mathbf{y}_{i-1}$ to $\mathbf{y}_i$, measured with respect to projected solid angle.[1] Now the density $P_A(\mathbf{y}_i)$ for choosing vertex $\mathbf{y}_i$ is simply

$$P_A(\mathbf{y}_i) \;=\; P_{\sigma^\perp}(\mathbf{y}_{i-1} \!\to\! \mathbf{y}_i)\, G(\mathbf{y}_{i-1} \!\leftrightarrow\! \mathbf{y}_i)$$

recalling that

$$G(\mathbf{x} \!\leftrightarrow\! \mathbf{x}') \;=\; V(\mathbf{x} \!\leftrightarrow\! \mathbf{x}')\, \frac{|\cos(\theta_{\mathrm{o}})\ \cos(\theta'_{\mathrm{i}})|}{\|\mathbf{x} - \mathbf{x}'\|^2}$$

(see Section 8.2.2.2 for further details).

We define symbols p_i^L and p_i^E to represent the probabilities for generating the first i vertices of the light and eye subpaths respectively. These are defined by

$$\begin{aligned} p_0^L &= 1\,, \\ p_1^L &= P_A(\mathbf{y}_0)\,, \\ p_i^L &= P_{\sigma^\perp}(\mathbf{y}_{i-2} \!\to\! \mathbf{y}_{i-1})\, G(\mathbf{y}_{i-2} \!\leftrightarrow\! \mathbf{y}_{i-1})\, p_{i-1}^L \qquad \text{for } i \geq 2\,, \end{aligned}$$

and similarly

$$\begin{aligned} p_0^E &= 1\,, \\ p_1^E &= P_A(\mathbf{z}_0)\,, \\ p_i^E &= P_{\sigma^\perp}(\mathbf{z}_{i-2} \!\to\! \mathbf{z}_{i-1})\, G(\mathbf{z}_{i-2} \!\leftrightarrow\! \mathbf{z}_{i-1})\, p_{i-1}^E \qquad \text{for } i \geq 2\,. \end{aligned}$$

[1] More precisely, it should be written as $P_{\sigma^\perp}(\mathbf{y}_{i-1} \to \mathbf{y}_i \mid \mathbf{y}_{i-2}, \mathbf{y}_{i-1})$, since the probability is conditional on the locations of the previous two vertices in the subpath.

Using these symbols, the density for generating the path $\bar{x}_{s,t} \;=\; \mathbf{y}_0 \ldots \mathbf{y}_{s-1}\, \mathbf{z}_{t-1} \ldots \mathbf{z}_0$ is simply

$$p_{s,t}(\bar{x}_{s,t}) \;=\; p^L_s\, p^E_t \,. \tag{10.4}$$

The unweighted contribution $\mathbf{C}^*_{\mathbf{s,t}}$. Next, we consider the unweighted contribution

$$C^*_{s,t} \;\equiv\; \frac{f_j(\bar{x}_{s,t})}{p_{s,t}(\bar{x}_{s,t})} \,. \tag{10.5}$$

To calculate this quantity efficiently, we precompute the weights α^L_i and α^E_i given below. These weights consist of all the factors of the definition (10.5) that can be computed using the first i vertices of the light and eye subpaths respectively. Specifically, we have

$$\begin{aligned}
\alpha^L_0 &= 1\,, \\
\alpha^L_1 &= \frac{L^{(0)}_{\mathrm{e}}(\mathbf{x}_0)}{P_A(\mathbf{y}_0)}\,, \\
\alpha^L_i &= \frac{f_{\mathrm{s}}(\mathbf{y}_{i-3} \to \mathbf{y}_{i-2} \to \mathbf{y}_{i-1})}{P_{\sigma^\perp}(\mathbf{y}_{i-2} \to \mathbf{y}_{i-1})}\, \alpha^L_{i-1} \qquad \text{for } i \ge 2\,,
\end{aligned} \tag{10.6}$$

and similarly

$$\begin{aligned}
\alpha^E_0 &= 1\,, \\
\alpha^E_1 &= \frac{W^{(0)}_{\mathrm{e}}(\mathbf{z}_0)}{P_A(\mathbf{z}_0)}\,, \\
\alpha^E_i &= \frac{f_{\mathrm{s}}(\mathbf{z}_{i-1} \to \mathbf{z}_{i-2} \to \mathbf{z}_{i-3})}{P_{\sigma^\perp}(\mathbf{z}_{i-2} \to \mathbf{z}_{i-1})}\, \alpha^E_{i-1} \qquad \text{for } i \ge 2\,.
\end{aligned} \tag{10.7}$$

Here we have used the conventions previously described in Section 8.3.2: the emitted radiance is split into a product

$$L_{\mathrm{e}}(\mathbf{y}_0 \to \mathbf{y}_1) \;=\; L^{(0)}_{\mathrm{e}}(\mathbf{y}_0)\, L^{(1)}_{\mathrm{e}}(\mathbf{y}_0 \to \mathbf{y}_1)\,,$$

where $L^{(0)}_{\mathrm{e}}$ and $L^{(1)}_{\mathrm{e}}$ represents the spatial and directional components of L_{e} respectively, and we define $f_{\mathrm{s}}(\mathbf{y}_{-1} \to \mathbf{y}_0 \to \mathbf{y}_1) \;\equiv\; L^{(1)}_{\mathrm{e}}(\mathbf{y}_0 \to \mathbf{y}_1)$. The quantities $W^{(0)}_{\mathrm{e}}$ and $f_{\mathrm{s}}(\mathbf{z}_1 \to \mathbf{z}_0 \to \mathbf{z}_{-1}) \;\equiv\; W^{(1)}_{\mathrm{e}}$ are defined similarly. The purpose of this convention is to reduce the number of special cases that need to be considered, by interpreting the directional component of emission as a BSDF. Also, notice that the geometry factors $G(\mathbf{x} \leftrightarrow \mathbf{x}')$ do not appear in the formulas for α_i^L and α_i^E, because these factors occur in both the numerator and denominator of (10.5) (see the definitions of p_i^L and p_i^E).

As mentioned above, the unweighted contribution can now be computed as

$$C^*_{s,t} \;=\; \alpha_s^L \, c_{s,t} \, \alpha_t^E \,, \tag{10.8}$$

where $c_{s,t}$ consists of the remaining factors of the integrand f_j that are not included in the precomputed weights. Examining the definitions of f_j, α_i^L, and α_i^E, we obtain

$$\begin{aligned} c_{0,t} &= L_{\mathrm{e}}(\mathbf{z}_{t-1} \!\rightarrow\! \mathbf{z}_{t-2}) \,, \\ c_{s,0} &= W_{\mathrm{e}}(\mathbf{y}_{s-2} \!\rightarrow\! \mathbf{y}_{s-1}) \,, \quad \text{and} \\ c_{s,t} &= f_{\mathrm{s}}(\mathbf{y}_{s-2} \!\rightarrow\! \mathbf{y}_{s-1} \!\rightarrow\! \mathbf{z}_{t-1}) \, G(\mathbf{y}_{s-1} \!\leftrightarrow\! \mathbf{z}_{t-1}) \, f_{\mathrm{s}}(\mathbf{y}_{s-1} \!\rightarrow\! \mathbf{z}_{t-1} \!\rightarrow\! \mathbf{z}_{t-2}) \\ &\qquad\qquad\qquad\qquad\qquad\qquad \text{for } s, t > 0 \,. \end{aligned}$$

Note that the factor $G(\mathbf{y}_{s-1} \leftrightarrow \mathbf{z}_{t-1})$ includes a visibility test (for the case $s, t > 0$), which is the most expensive aspect of the evaluation.

The weighting function $\mathbf{w_{s,t}}$. Finally we consider how to evaluate

$$w_{s,t} \;\equiv\; w_{s,t}(\bar{x}_{s,t}) \,,$$

whose value depends on the probability densities with which $\bar{x}$ is generated by all of the $s + t + 1$ possible sampling techniques for paths of this length. We define p_i as the density for generating $\bar{x}_{s,t}$ using a light subpath with i vertices, and an eye subpath with $s + t - i$ vertices:

$$p_i \;=\; p_{i,(s+t)-i}(\bar{x}_{s,t}) \qquad \text{for } i = 0, \ldots, s+t \,.$$

In particular, p_s is the probability with which the given path was actually generated, while $p_0 \ldots p_{s-1}$ and $p_{s+1} \ldots p_{s+t}$ represent all the other ways that this path *could* have been generated.

The evaluation of the p_i can be simplified by observing that their values only matter up to an overall scale factor. For example, if the samples are combined using the power heuristic with $\beta = 2$, we must compute

$$w_{s,t} \;=\; \frac{p_s^2}{\sum_i p_i^2} \;=\; \frac{1}{\sum_i (p_i/p_s)^2}\,.$$

The same is true for all the other combination strategies of Chapter 9. Thus we can arbitrarily set $p_s = 1$, and compute the values of the other p_i relative to p_s.

To do this, we consider the ratio p_{i+1}/p_i. It will be convenient to ignore the distinction between vertices in the light and eye subpaths, and to write the path $\bar{x}_{s,t}$ as

$$\bar{x} \;=\; \mathbf{x}_0 \ldots \mathbf{x}_k$$

where $k = s+t-1$. In this notation, the only difference between p_i and p_{i+1} lies in how the vertex $\mathbf{x}_i$ is generated: for p_i, it is generated as part of the eye subpath $\mathbf{x}_i \ldots \mathbf{x}_k$, while for p_{i+1} it is generated as part of the light subpath $\mathbf{x}_0 \ldots \mathbf{x}_i$. All other vertices of $\bar{x}$ are generated with the same probability by both techniques. Thus, the ratio of p_{i+1} to p_i is

$$\begin{aligned}
\frac{p_1}{p_0} &= \frac{P_A(\mathbf{x}_0)}{P_{\sigma^\perp}(\mathbf{x}_1 \rightarrow \mathbf{x}_0)\, G(\mathbf{x}_1 \leftrightarrow \mathbf{x}_0)}\,, \\
\frac{p_{i+1}}{p_i} &= \frac{P_{\sigma^\perp}(\mathbf{x}_{i-1} \rightarrow \mathbf{x}_i)\, G(\mathbf{x}_{i-1} \leftrightarrow \mathbf{x}_i)}{P_{\sigma^\perp}(\mathbf{x}_{i+1} \rightarrow \mathbf{x}_i)\, G(\mathbf{x}_{i+1} \leftrightarrow \mathbf{x}_i)} \qquad \text{for } 0 < i < k\,, \qquad (10.9) \\
\frac{p_{k+1}}{p_k} &= \frac{P_{\sigma^\perp}(\mathbf{x}_{k-1} \rightarrow \mathbf{x}_k)\, G(\mathbf{x}_{k-1} \leftrightarrow \mathbf{x}_k)}{P_A(\mathbf{x}_k)}\,.
\end{aligned}$$

This equation can be applied repeatedly starting with p_s to find p_{s+1}, . . ., p_{k+1}. Similarly, the reciprocal ratio p_i/p_{i+1} can be used to compute p_{s-1}, . . ., p_0.

Once the p_i have been calculated, it is straightforward to compute $w_{s,t}$ according to the combination strategy being used. The final weighted sample contribution is then

$$\begin{aligned}
C_{s,t} &= w_{s,t}\, C^*_{s,t} \\
&= w_{s,t}\, \alpha^L_s\, c_{s,t}\, \alpha^E_t\,.
\end{aligned}$$

Note that the samples in each group are dependent (since they are all generated from the same light and eye subpath). However this does not significantly affect the results, since the correlation between them goes to zero as we increase the number of independent sample groups for each measurement. For example, if N independent light and eye subpaths are used, then all of the samples from each $p_{s,t}$ are independent, and each sample from $p_{s,t}$ is correlated with only one of the N samples from any other given technique $p_{s',t'}$. From this fact it is easy to show that the variance results of Chapter 9 are not affected by more than a factor of $(N-1)/N$ due to the correlation between samples in each group.

10.3 Implementation issues

This section describes several aspects of our implementation. We start by explaining how the image is sampled and filtered. Next we describe how the light and eye subpaths are generated. This includes a summary of the information that is precomputed and stored with each subpath (in order to evaluate the sample contributions more efficiently), and the methods used to determine the length of each subpath. Following this, we describe how to implement the important special cases where the light or eye subpath has at most one vertex. Finally, we consider how to handle specular surfaces correctly, and we consider several situations where the weighting functions $w_{s,t}$ cannot be computed exactly (so that approximations must be used).

10.3.1 Image sampling and filtering

So far, our discussion of bidirectional path tracing could be applied to any kind of measurements I_j. Here we discuss the special issues that arise when computing an image (as opposed to some other set of measurements).

Overall, the image sampling of bidirectional path tracing is similar to ray tracing or path tracing. The camera and lens model determine a mapping from rays in world space onto the *image plane*. This mapping is used to define an *image function* I such that $I(u, v)$ is proportional to the irradiance on the image plane at the point (u, v).[2] Each pixel value I_j is

[2] Strictly speaking, the units of $I(u, v)$ are *sensor response per unit area* $[\mathrm{S} \cdot \mathrm{m}^{-2}]$ rather than irradiance. (When $I(u, v)$ is integrated, the resulting pixel values have units of sensor response [S] rather than power.)

then defined as a weighted average

$$I_j \;=\; \iint_D h_j(u,v)\, I(u,v)\, du\, dv\,,$$

where D is the image region, and h_j is the *filter function* for pixel j (which integrates to one). In general, the filter functions are all translated copies of one another, and each one is zero except on a small subset of D.

To estimate the values of all the pixels $I_1, \ldots, I_M$, a large number of sample points are chosen across the image region. We do this by taking a fixed number of stratified samples per pixel (e.g. to take $n = 25$ samples, the nominal rectangle corresponding to each pixel would be subdivided into a 5 by 5 grid). Each sample can contribute to the value of several pixels, since the filter functions h_j generally overlap one another. Specifically, the pixel values are estimated using[3]

$$I_j \;\approx\; \frac{\sum_{i=1}^{N} h_j(u_i, v_i)\, I(u_i, v_i)}{\sum_{i=1}^{N} h_j(u_i, v_i)}\,, \tag{10.11}$$

where $N = nM$ is the total number of samples. This equation can be evaluated efficiently by storing the current value of the numerator and denominator of (10.11) at each pixel, and accumulating samples as they are taken. Note that each sample (u_i, v_i) contributes to only a few nearby pixels (because of the filter functions h_j), and that it is not necessary to store the samples themselves.

10.3.2 Estimation of the image function

The image function $I(u, v)$ is estimated using bidirectional path tracing. The initial vertex of the light subpath is chosen according to the emitted power at each surface point, while the remaining vertices are chosen by sampling from the BSDF (or some convenient approximation). Sampling the camera lens is slightly trickier: the vertex $\mathbf{z}_0$ can be chosen anywhere

[3] Note that this estimate is slightly biased. The corresponding unbiased estimate is simply

$$I_j \;=\; E\left[(|D|\,/\,N)\, \textstyle\sum_{i=1}^{N} h_j(u_i, v_i)\, I(u_i, v_i)\right]\,, \tag{10.10}$$

where $|D|$ is the area of the image region D. However, equation (10.11) typically gives better results (a smaller mean-squared error) because it compensates for random variations in the sum of the filter weights.

on the lens surface, but the direction $\mathbf{z}_0 \to \mathbf{z}_1$ is then uniquely determined by the given point (u, v) on the image plane (since there is only one direction at $\mathbf{z}_0$ that is mapped to the point (u, v) by the lens).[4] Note that the density $P_{\sigma^\perp}(\mathbf{z}_0 \to \mathbf{z}_1)$ is determined by the fact that (u, v) is uniformly distributed over the image region.

After generating the light and eye subpaths, we consider all possible connections between them as described above. In order to do this efficiently, we cache information about the vertices in each subpath. The vertex itself is stored in the form of a special *Event* object that has methods for sampling and evaluating the BSDF, and for evaluating the probability with which a given direction is sampled (according to a built-in sampling strategy associated with each BSDF). The vertices $\mathbf{y}_0$ and $\mathbf{z}_0$ are also stored in this form, so that the distribution of emitted radiance and importance at these vertices can be queried using the same methods.

Other per-vertex information includes the cumulative subpath weights α_i^L and α_i^E defined above, the geometric factors $G(\mathbf{x}_{i-1} \leftrightarrow \mathbf{x}_i)$, and the probability densities $P_{\sigma^\perp}(\mathbf{x}_i \to \mathbf{x}_{i-1})$ and $P_{\sigma^\perp}(\mathbf{x}_i \to \mathbf{x}_{i+1})$ for sampling the adjacent subpath vertices on either side. The latter three fields are used in equation (10.9) to efficiently evaluate the probabilities p_i with which a given path is sampled using all the other possible techniques.

When information about the subpaths is cached, then the work required to evaluate the contributions $C_{s,t}$ is minimal (except for the visibility test, if necessary). The only quantities that need to be evaluated are those associated with the connecting edge $\mathbf{y}_{s-1}\mathbf{z}_{t-1}$ (since this edge is not part of either subpath).

10.3.3 Determining the subpath lengths

To control the lengths of the light and eye subpaths (n_L and n_E), we define a probability for the subpath to be terminated or *absorbed* after each vertex is generated. We let q_i denote the probability that the subpath is continued past vertex $\mathbf{x}_i$, while $1 - q_i$ is the probability that the subpath is terminated. This is a form of *Russian roulette* (Chapter 2).

[4] For real lens models [Kolb et al. 1995], it is difficult to determine the direction $\mathbf{z}_0 \to \mathbf{z}_1$ once the point $\mathbf{z}_0$ has already been chosen, since this requires us to find a chain of specular refractions that connects two given points on opposite side of the lens (i.e. $\mathbf{z}_0$ and the point (u, v) on the film plane). A better approach in this case is to generate $\mathbf{z}_0$ and $\mathbf{z}_0 \to \mathbf{z}_1$ together, by starting on the film plane at (u, v) and tracing a ray toward the exit pupil.

In our implementation, we set $q_i = 1$ for the first few vertices of each subpath, to avoid any extra variance on short subpaths (which typically make the largest contribution to the image). After that, q_i is determined by first sampling a candidate direction $\mathbf{x}_i \to \mathbf{x}_{i+1}$, and then letting

$$q_i \;=\; \min\{1, \frac{f_{\mathrm{s}}(\mathbf{x}_{i-1} \to \mathbf{x}_i \to \mathbf{x}_{i+1})}{P^*_{\sigma^\perp}(\mathbf{x}_i \to \mathbf{x}_{i+1})}\}\,,$$

where $P^*_{\sigma^\perp}$ is density function used for sampling the direction $\mathbf{x}_i \to \mathbf{x}_{i+1}$. Notice that if $P^*_{\sigma^\perp}(\mathbf{x}_i \to \mathbf{x}_{i+1})$ is proportional to the BSDF, then q_i is simply the albedo of the material, i.e. the fraction of energy that is scattered rather than absorbed for the given incident direction.

This procedure does not require any modification to the formulas for the sample contributions described in Section 10.2. However, it is important to realize that the final probability density for sampling each direction is now a product:

$$P_{\sigma^\perp}(\mathbf{x}_i \to \mathbf{x}_{i+1}) \;=\; q_i\, P^*_{\sigma^\perp}(\mathbf{x}_i \to \mathbf{x}_{i+1})\,.$$

The density $P_{\sigma^\perp}(\mathbf{x}_i \to \mathbf{x}_{i+1})$ can integrate to less than one, since there is a discrete probability associated with terminating the subpath at $\mathbf{x}_i$.

10.3.4 Special cases for short subpaths

Subpaths with less than two vertices require special treatment for various reasons. The most important issues are: taking advantage of direct lighting calculations when the light subpath has only one vertex, and allowing samples to contribute to any pixel of the image in the cases when the eye subpath has zero or one vertices. In addition, the cases when the light or eye subpath is empty require special handling since no visibility test is needed.

10.3.4.1 Zero light subpath vertices ($\mathrm{s} = 0$)

These samples occur when the eye subpath randomly intersects a light source. For this to occur, the light sources must be modeled as part of the scene (so that it is possible for a ray to intersect them). We also require the ability to determine whether the current eye subpath endpoint $\mathbf{z}_{t-1}$ is on a light source, and to evaluate the emitted radiance along the ray $\mathbf{z}_{t-1} \to \mathbf{z}_{t-2}$. In order to evaluate the combination weight $w_{0,t}$, we must also compute the probability densities for generating the point $\mathbf{z}_{t-1}$ and the direction $\mathbf{z}_{t-1} \to \mathbf{z}_{t-2}$ by sampling the light sources (in order to compute the densities p_i with which the other sampling techniques generate this path).

The $s = 0$ sampling technique is very important for the rendering of certain lighting effects. These include: directly visible light sources; lights that are seen by reflection or refraction in a specular surface; caustics due to large area light sources; and caustics that are viewed indirectly through a specular surface.

A nice thing about this sampling technique is that no visibility test is required. Thus its contributions are cheap to evaluate, compared to the other $C_{s,t}$. In our implementation, we accumulate these contributions as the eye subpath is being generated.

10.3.4.2 One light subpath vertex (s $= 1$)

This sampling technique connects a given eye subpath $\mathbf{z}_{t-1} \ldots \mathbf{z}_0$ to a randomly chosen point on the light sources. Recall that in the basic algorithm, this point is simply the first vertex $\mathbf{y}_0$ of the light subpath (which was chosen according to emitted power). However, the variance of these samples can be greatly reduced by choosing the vertex using special direct lighting techniques. That is, we simply ignore the vertex $\mathbf{y}_0$, and connect the eye subpath to a new vertex $\mathbf{y}_0^{\mathrm{d}}$ chosen using a more sophisticated method (such as those described by Shirley et al. [1996]). This strategy is applied to each eye subpath suffix $\mathbf{z}_{t-1} \ldots \mathbf{z}_0$ separately, by choosing a different light source vertex for each one.

This optimization is very important for direct illumination (i.e. paths of length two), since it allows the same low-variance lighting techniques used in ray tracing to be applied. It is also an important optimization for longer paths; this corresponds to standard path tracing, where each vertex of the path is connected to a point on the light source. A direct lighting strategy is essentially an importance sampling technique that chooses a light source vertex $\mathbf{y}_0^{\mathrm{d}}$ according to how much it contributes to the illuminated point $\mathbf{z}_{t-1}$ (or some approximation of this distribution).

This strategy requires some changes in the way that sample contributions are evaluated:

- The unweighted contribution $C^*_{1,t}$ is computed using the density $P_A^{\mathrm{d}}(\mathbf{y}_0^{\mathrm{d}})$ with which the light vertex $\mathbf{y}_0^{\mathrm{d}}$ was chosen. This calculation is identical to standard path tracing.

- The evaluation of the combination weight $w_{1,t}$ is slightly trickier, because the direct lighting strategy does not affect the sampling of light subpaths with two or more vertices. Thus we must evaluate the density with which $\mathbf{y}_0^{\mathrm{d}}$ is sampled according to emitted power; this is used to compute the probabilities p_i for sampling the current path using the other possible techniques.

- The direct lighting strategy also affects the combinations weights for paths where $s \neq 1$. The correct probabilities p_i can be found by computing them as usual, and then multiplying the density for p_1 by $P_A^{\mathrm{d}}(\mathbf{x}_0) \, / \, P_A(\mathbf{x}_0)$. Here $P_A(\mathbf{x}_0)$ is the density for generating $\mathbf{x}_0$ according to emitted power, and $P_A^{\mathrm{d}}(\mathbf{x}_0)$ is the density for generating $\mathbf{x}_0$ using direct lighting for the point $\mathbf{x}_1$.

It is also possible to use a direct lighting strategy that takes more than one sample, e.g. a strategy that iterates over the light sources taking a few samples from each one. This is equivalent to using more than one sampling technique to generate these paths; the samples are simply combined as usual according to the rules of multiple importance sampling.

10.3.4.3 One eye subpath vertex (t $= 1$)

These samples are generated by connecting each light subpath prefix $\mathbf{y}_0 \ldots \mathbf{y}_{s-1}$ to the vertex $\mathbf{z}_0$ on the camera lens. These samples are important for rendering caustics (especially those from small or point light sources), some forms of direct illumination, and a variety of other lighting effects.

The main issue with this technique is that the samples it generates can lie anywhere in the image, not just at the current point (u, v). One way to handle this is to discard samples that do not contribute to the current measurement I_j. However, this is inefficient; much more information can be obtained by letting these samples contribute to any pixel of the image.

To implement this, we allocate a separate image to record the contributions of paths where 0 or 1 vertices are generated from the eye. We call this the *light image,* as opposed to the *eye image* that holds the contributions of paths where $t \geq 2$ eye subpath vertices are used.

To accumulate each sample, we first determine the point (u', v') on the image plane that corresponds to the ray $\mathbf{y}_{s-1} \!\rightarrow\! \mathbf{z}_0$. We then compute the contribution $C_{s,1}$ of this sample as usual, and record it at the location (u', v'). This is done by finding all of the pixels whose filter value $h_j(u', v')$ is non-zero, and updating the pixel values I_j^L of the light image using

$$I_j^L \;\leftarrow\; I_j^L \;+\; h_j(u', v')\, C_{s,1}\,.$$

Note that the estimate $I(u, v)$ at the current image point is not affected by this calculation. Also note that it is not necessary to store the light and eye images in memory (although this is what is done in our implementation). The eye image can be sampled and written to disk in scanline order, while the light image can be handled by repeatedly accumulating a fixed number of samples in memory, sorting them in scanline order, and merging them with an image on disk.

When the algorithm has finished, the final estimate for each pixel has the form

$$I_j \;\approx\; (|D| \,/\, N)\, I_j^L \;+\; I_j^E\,,$$

where $|D|$ is the area of the image region, N is the total number of bidirectional samples that were taken, and I_j^E is the estimate for pixel j from the eye image (sampled and filtered as described in Section 10.3.1). Note that the eye and light images are filtered differently: the eye image is normalized at each pixel by dividing by the sum of the filter weights, while the light image is not (see equations (10.11) and (10.10) respectively). Thus the final pixel values of the light image are determined by the sample density as well as the sample values; more samples per pixel correspond to a brighter image.

Note that to evaluate the contribution $C_{s,1}$ of each sample, we must evaluate the importance emitted from $\mathbf{z}_0$ toward $\mathbf{y}_{s-1}$ (or more precisely, the directional component $W_{\mathrm{e}}^{(1)}$ of the importance). The function $W_{\mathrm{e}}^{(1)}$ is defined so that

$$\int_D W_{\mathrm{e}}^{(1)}(\mathbf{z}_0, \omega)\, d\sigma^{\perp}(\omega)$$

is equal to the fraction of the image region covered by the points (u, v) that are mapped by the lens to directions $\omega \in D$. It is important to realize that this function is not uniform for most lens models in graphics, since pixels near the center of the image correspond to a set of rays whose projected solid angle is larger than for pixels near the image boundary.

10.3.4.4 Zero eye subpath vertices (t $= 0$)

These samples occur when the light subpath randomly intersects the camera lens. Because the camera lens is a relatively small target, these samples do not contribute significantly for most scenes. On the other hand, these samples are very cheap to evaluate (because no visibility test is required), and can sometimes make the computation more robust. For example, this can be an effective sampling strategy for rendering specular reflections of small or highly directional light sources.

To implement this method, the lens surface must have a physical representation in the scene (so that it can be intersected by a ray). In particular, this sampling technique cannot be used for pinhole lens models. As with the case for $t = 1$ eye subpath vertices, samples can contribute to any pixel of the image. The image point (u', v') is determined from the ray $\mathbf{y}_{s-2} \to \mathbf{y}_{s-1}$, and samples are accumulated and filtered in the light image as before.

10.3.5 Handling specular surfaces

Specular surfaces require careful treatment, because the BSDF and the density functions used for importance sampling both contain Dirac distributions. This is not a problem when computing the weights α_i^L and α_i^E, since the ratio

$$\frac{f_{\mathrm{s}}(\mathbf{x}_{i-3} \to \mathbf{x}_{i-2} \to \mathbf{x}_{i-1})}{P_{\sigma^\perp}(\mathbf{x}_{i-2} \to \mathbf{x}_{i-1})}$$

of equation (10.6) is well-defined. Although this ratio cannot be directly evaluated (since the numerator and denominator both contain a Dirac distribution), it can be returned as a “weight” when the specular component of the BSDF is sampled.

Similarly, specular surfaces do not cause any problems when computing the unweighted contribution $C^*_{s,t}$ that connects the eye and light subpaths. The specular components of the BSDF’s can simply be ignored when computing the factor

$$c_{s,t} \;=\; f_{\mathrm{s}}(\mathbf{y}_{s-2} \to \mathbf{y}_{s-1} \to \mathbf{z}_{t-1})\, G(\mathbf{y}_{s-1} \leftrightarrow \mathbf{z}_{t-1})\, f_{\mathrm{s}}(\mathbf{y}_{s-1} \to \mathbf{z}_{t-1} \to \mathbf{z}_{t-2})\,,$$

since there is a zero probability that these BSDF’s will have a non-zero specular component in the direction of the given connecting edge.[5]

On the other hand, specular surfaces require careful treatment when computing the weights $w_{s,t}$ for multiple importance sampling. To compute the densities p_i for the other possible ways of sampling this path, we must evaluate expressions of the form

$$\frac{p_{i+1}}{p_i} = \frac{P_{\sigma^\perp}(\mathbf{x}_{i-1} \to \mathbf{x}_i)\, G(\mathbf{x}_{i-1} \leftrightarrow \mathbf{x}_i)}{P_{\sigma^\perp}(\mathbf{x}_{i+1} \to \mathbf{x}_i)\, G(\mathbf{x}_{i+1} \leftrightarrow \mathbf{x}_i)} \tag{10.12}$$

(see equation (10.9)). In this case the denominator may contain a Dirac distribution that is not matched by a corresponding factor in the numerator.

We handle this problem by introducing a *specular* flag for each vertex. If the flag is true, it means that the BSDF and sampling probabilities at this vertex are represented only up to an unspecified constant of proportionality. That is, the cached values of the BSDF $f_{\mathrm{s}}(\mathbf{x}_{i-1} \to \mathbf{x}_i \to \mathbf{x}_{i+1})$ and the probability densities $P_{\sigma^\perp}(\mathbf{x}_i \to \mathbf{x}_{i-1})$ and $P_{\sigma^\perp}(\mathbf{x}_i \to \mathbf{x}_{i+1})$ are all considered to be coefficients for a single Dirac distribution δ that is shared between them.[6] When applying equation (10.12), we use only the coefficients, and simply keep track of the fact that the corresponding density also contains a Dirac distribution.

Specifically, consider a path whose connecting edge is $\mathbf{x}_{s-1}\mathbf{x}_s$. We start with the nominal probability $p_s = 1$, and compute the relative values of the other p_i by applying (10.12) repeatedly. It is easy to check that a specular vertex at $\mathbf{x}_j$ causes a Dirac distribution to appear in the denominator of p_j and p_{j+1}, so that these probabilities are effectively zero. (Notice that these densities correspond to the sampling techniques where $\mathbf{x}_j$ is a connecting vertex.) However, these are the only p_i that are affected, since for other values of i the Dirac distributions in $P_{\sigma^\perp}(\mathbf{x}_j \to \mathbf{x}_{j-1})$ and $P_{\sigma^\perp}(\mathbf{x}_j \to \mathbf{x}_{j+1})$ are canceled by each other.

The end result is particularly simple: we first compute all of the p_i exactly as we would for non-specular vertices, without regard for the fact the some of the densities are actually coefficients for Dirac distributions. Then for every vertex where $\mathbf{x}_j$ is specular, we set p_j and

[5] Even if the connecting edge happened to have a direction for which one of the BSDF's is specular (a set of measure zero), the value of the BSDF is infinite and cannot be represented as a real number. Thus we choose to ignore such paths (by assigning them a weight $\alpha_{s,t} = 0$), and instead we account for them using one of the other sampling techniques.

[6] From another point of view, we can say that BSDF and probability densities are expressed with respect to a different *measure function*, one that assigns a positive measure to the discrete direction $\mathbf{x}_{i-2} \to \mathbf{x}_{i-1}$.

p_{j+1} to zero (since these probabilities include a symbolic Dirac distribution in the denominator). Note that these techniques apply equally well to the case of perfectly anisotropic reflection, where light from a given direction is scattered into a one-dimensional set of outgoing directions. In this case, the unspecified constant of proportionality associated with the *specular* flag is a one-dimensional Dirac distribution.

10.3.6 Approximating the weighting functions

Up until now, we have assumed that the densities p_i for sampling the current path using other techniques can be computed exactly (as required to evaluate the weight $w_{s,t}$). However, there are some situation where it is difficult or impossible to do this; examples will be given below. In these situations, the solution is to replace the true densities p_i with approximations $\hat{p}_i$ when evaluating the weights. As long as these approximations are reasonably good, the optimality properties of the combination strategy being used will not be significantly affected. But even if the approximations are bad, the results will at least be unbiased, since the weighting functions sum to one for any values of the $\hat{p}_i$.[7] We now discuss the reasons that approximations are sometimes necessary.

Adaptive sampling is one reason that the exact densities can be difficult to compute.[8] For example, suppose that adaptive sampling is used on the image plane, to take more samples where the measured variance is high. In this case, it is impossible to compare the densities for sampling techniques where $t \geq 2$ eye vertices are used to those where $t \leq 1$, since the densities for $t \geq 2$ depend on the eventual distribution of samples over the image plane (which has not yet been determined). A suitable approximation in this case is to assume that the density of samples is uniform across the image.

Similarly there are some direct lighting strategies where approximations are necessary, because the strategy makes random choices that cannot be determined from the final light source vertex $\mathbf{y}_0^{\mathrm{d}}$. For example, consider the following strategy [Shirley et al. 1996]. First,

[7] Note that to avoid bias, the unweighted contribution $C^*_{s,t}$ must always be evaluated exactly; this part of the calculation is required for any unbiased Monte Carlo algorithm. The evaluation of $C^*_{s,t}$ should never be a problem, since all the random choices that were used to generate the current path are explicitly available (including random choices that are cannot be determined from the resulting path itself).

[8] Note that adaptive sampling can introduce bias, unless two-stage sampling is used [Kirk & Arvo 1991].

a candidate vertex $\mathbf{x}_i$ is generated on each light source S_i. Next we compute the contribution that each vertex $\mathbf{x}_i$ makes to the illuminated point $\mathbf{z}_{t-1}$, under the assumption that the corresponding visibility ray is not obstructed. Finally, we choose one of the candidates $\mathbf{x}_i$ at random according to its contribution, and return it as the light source vertex $\mathbf{y}_0^{\mathrm{d}}$. The problem with this strategy is that given an arbitrary point $\mathbf{x}$ on a light source, it is very difficult to evaluate the probability density $P_A^{\mathrm{d}}(\mathbf{x})$ with which $\mathbf{x}$ is sampled. This is because the sampling procedure makes random choices that are not reflected in the final result $\mathbf{y}_0^{\mathrm{d}}$: namely, the locations of the other candidate points $\mathbf{x}_i$, which are generated and then discarded. To evaluate the density exactly would require analytic integration over the all possible locations of the $\mathbf{x}_i$. A suitable approximation in this case is to use the conditional probability $A^{\perp}(\mathbf{x}_i \mid S_i)$, i.e. the density for sampling $\mathbf{x}_i$ given that the light source S_i has already been chosen.

10.4 Reducing the number of visibility tests

To make bidirectional path tracing more efficient, it is important to reduce the number of visibility tests. The basic algorithm assumes that all of the $O(n_L n_E)$ contributions are evaluated; however, typically most of these contributions are so small that a visibility test is not justified. In this section, we develop a new technique called *efficiency-optimized Russian roulette* that is an effective solution to this problem. We start with an introduction to ordinary Russian roulette and a discussion of its shortcomings. Next, we describe efficiency-optimized Russian roulette as a general technique. Finally we describe the issues that arise when applying this technique to bidirectional path tracing.

We consider the following abstract version of the visibility testing problem. Suppose that we must repeatedly evaluate an estimator of the form

$$F \;=\; C_1 + \cdots + C_n \,,$$

where the number of contributions n is a random variable. We assume that each contribution C_i can be written as the product of a *tentative contribution* t_i, and a *visibility factor* v_i (which is either 0 or 1).

The number of visibility tests can be reduced using *Russian roulette*. We define the *roulette probability* q_i to be the probability of testing the visibility factor v_i. Each contribution then has the form

$$C_i \;=\; \begin{cases} (1/q_i)\, v_i\, t_i & \text{with probability } q_i\,, \\ 0 & \text{otherwise}\,. \end{cases}$$

It is easy to verify that $E[C_i] \;=\; E[v_i\, t_i]$, i.e. this estimator is unbiased.

The main question, of course, is how to choose the roulette probabilities q_i. Typically this is done by choosing a fixed *roulette threshold* δ, and defining

$$q_i \;=\; \min(1, t_i \,/\, \delta)\,.$$

Thus contributions larger than δ are always evaluated, while smaller contributions are randomly skipped in a way that does not cause bias.

This approach is not very satisfying, however, because the threshold δ is chosen arbitrarily. If the threshold is chosen too high, then there will be a substantial amount of extra variance (due to visibility tests that are randomly skipped), while if the threshold is too low, then many unnecessary visibility tests will be performed (leading to computation times that are longer than necessary). Russian roulette thus involves a tradeoff, where the reduction in computation time must be balanced against the corresponding increase in variance.

10.4.1 Efficiency-optimized Russian roulette

In this section, we show how to choose the roulette probabilities q_i so as to maximize the efficiency of the resulting estimator F. Recall that *efficiency* is defined as

$$\epsilon \;=\; \frac{1}{\sigma^2\, T}\,,$$

where σ^2 is the variance of the given estimator, and T is the average computation time required to evaluate it. We assume the computation time is simply proportional to the number of rays that are cast (n). Note that n includes all types of rays, not just visibility rays; e.g. for bidirectional path tracing, it includes the rays that are used to generate the light and eye subpaths.

To begin, we consider the effect that q_i has on the variance and cost of F. For the variance, we return to the definition

$$C_i \;=\; \begin{cases} (1/q_i)\, v_i\, t_i & \text{with probability } q_i\,, \\ 0 & \text{otherwise}\,. \end{cases}$$

We can treat t_i as a fixed quantity (since we are only interested in the additional variance relative to the case $q_i = 1$), and we can also assume that $v_i = 1$ (a conservative assumption, since if $v_i = 0$ then Russian roulette does not add any variance at all). The additional variance due to Russian roulette can then be written as

$$\begin{aligned} V[C_i] &= E[C_i^2] - E[C_i]^2 \\ &= \Big[q_i\, (t_i \,/\, q_i)^2 \;+\; (1 - q_i)\, 0\Big] \;-\; t_i^2 \\ &= t_i^2\, (1/q_i - 1)\,. \end{aligned}$$

As for the cost, it is easy to see that the number of rays is reduced by $1 - q_i$ on average.

Next, we examine how this affects the overall efficiency of F. Here we make an important assumption: namely, that F is sampled repeatedly, so that estimates of its average variance σ_0^2 and average sample cost n_0 can be computed. Then according to the discussion above, the modified efficiency due to q_i can be estimated as

$$\epsilon \;=\; \frac{1}{\Big[\sigma_0^2 + t_i^2\, (1/q_i - 1)\Big] \;\cdot\; (n_0 - (1 - q_i))}\,. \tag{10.13}$$

The optimal value of q_i is found by taking the derivative of this expression and setting it equal to zero. After some manipulation, this yields

$$q_i \;=\; t_i \,/\, \sqrt{(\sigma_0^2 - t_i^2)/(n_0 - 1)}\,.$$

Conveniently, this equation has the same form that is usually used for Russian roulette calculations, where the tentative contribution is compared against a given threshold δ. Since q_i is limited to the range $(0, 1]$, the optimal value is

$$\begin{aligned} q_i &= \min(1, t_i/\delta) \\ \text{where} \qquad \delta &= \sqrt{(\sigma_0^2 - t_i^2)/(n_0 - 1)}\,. \end{aligned} \tag{10.14}$$

However, this choice of the threshold δ has two undesirable properties. First, its value depends on the current tentative contribution t_i, so that it must be recalculated for every sample. Second, there is the possibility that an unusually large sample will have $t_i^2 > \sigma_0^2$, in which case the formula for δ does not make sense (although by returning to the original expression (10.13), it is easy to verify that the optimal choice in this case is $q_i = 1$).

To avoid these problems, we look for a fixed threshold δ^* that has the same transition point at which $q_i = 1$. It is easy to check that $q_i \geq 1$ if and only if $t_i^2 \geq \sigma_0^2/n_0$. Thus, the fixed threshold

$$\delta^* = \sqrt{\sigma_0^2/n_0}$$

leads to Russian roulette being applied to the same set of contributions as the original threshold (10.14).[9] Notice that δ^* is simply the estimated standard deviation per ray.

Summary. Efficiency-optimized Russian roulette consists of the following steps. Given an estimator F that is sampled a number of times, we keep track of its average variance σ_0^2 and average ray count n_0. Before each sample is taken we compute the threshold

$$\delta^* = \sqrt{\sigma_0^2/n_0}\,,$$

and apply this threshold to all of the individual contributions t_i that require a visibility test. The roulette probability q_i is given by

$$q_i = \min(1, t_i/\delta)\,.$$

Note that this technique does not maximize efficiency in a precise mathematical sense, since we have made several assumptions in our derivation. Rather, it should be interpreted as a heuristic that is *guided* by mathematical analysis; its purpose is to provide theoretical insight about parameter values that would otherwise be chosen in an *ad hoc* manner.

[9] The roulette probabilities will be slightly different for $q_i < 1$; it is easy to check that δ^* results in values of q_i that are slightly larger, by a factor between 1 and $\sqrt{n_0/(n_0-1)}$. Thus, visibility is tested slightly more often using the fixed threshold δ^* than the original threshold δ.

10.4.2 Implementation

The main requirement for implementing this technique is that we must be able to estimate the average variance and cost of each sample (i.e. σ_0^2 and n_0). This is complicated by the fact that the mean, variance, and sample cost can vary substantially over the image plane. It is not sufficient to simply compute the variance of all the samples taken so far, since the average variance of samples over the whole image plane does not reflect the variance at any particular pixel. For example, suppose that the left half of the current image is white, and the right half is black. The variance at most pixels might well be zero, and yet the estimated variance will be large if all the image samples are combined.

Ideally, we would like σ_0^2 and n_0 to estimate the variance and sample cost within the current pixel. This could be done by taking samples in random order over the whole image plane, and storing the location and value of each sample. We could then estimate σ_0^2 and n_0 at a given point (u, v) by computing the sample variance and average cost of the nearest N_0 samples.

In our implementation, we use a simpler approach. The image is sampled in scanline order, and we estimate σ_0^2 and n_0 using the last N_0 samples (for some fixed value of N_0). Typically we let N_0 be the number of samples per pixel; this ensures that all variance and cost estimates are made using samples from either the current pixel or the one before. (To ensure that the previous pixel is always nearby, scanlines are rendered in alternating directions. Alternatively, the pixels could be traversed according to a space-filling curve.)

The calculation of σ_0^2 and n_0 can be implemented efficiently as follows. Let n_j be the number of rays cast for the j-th sample, and let F_j be its value. We then simply maintain partial sums of n_j, F_j, and F_j^2 for the last N_0 samples, and set the Russian roulette threshold for the current sample to

$$\delta \;=\; \sqrt{\sigma_0^2 \,/\, n_0} \;=\; \sqrt{\frac{\sum F_j^2 - (1/N_0)\ \left(\sum F_j\right)^2}{\sum n_j}}\,,$$

where the sums are over the most recent N_0 samples only. (Note that the variance calculation is not numerically stable in this form, but we have not found this to be a problem.) It is most efficient to update these sums incrementally, by adding the values for the current sample j and subtracting the values for sample $j - N_0$. For this purpose, the last N_0 values of F_j and

n_j are kept in an array. We have found the overhead of these calculations to be negligible compared to ray casting.

An alternative would be to compute a *running average* of each quantity. This is done using the update formula

$$S_j \;=\; \alpha\, x_j \;+\; (1-\alpha)\, S_{j-1}\,,$$

where α is a small real number that determines how quickly the influence of each sample drops off with time. (This technique is also known as *exponential smoothing*.)

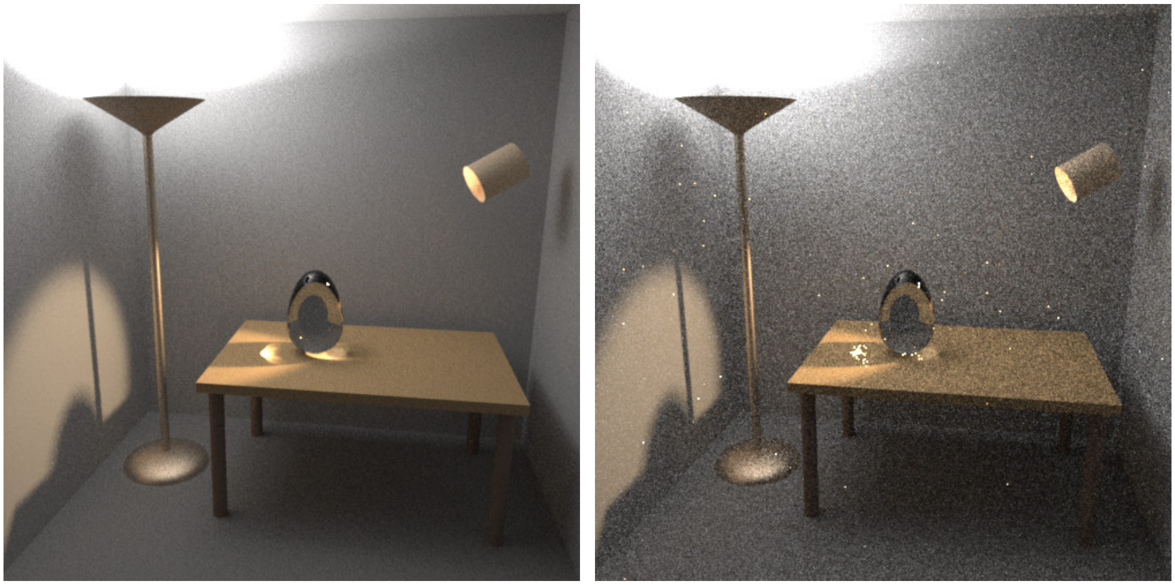
10.5 Results

We have compared bidirectional path tracing against ordinary path tracing using the test scene shown in Figure 10.3. The scene contains a floor lamp, a spotlight, a table, and a large glass egg. Observe that diffuse, glossy, and pure specular surfaces are all present, and that most of the room is illuminated indirectly.

Figure 10.3(a) was created by sampling paths up to length $k = 5$ using bidirectional path tracing, and combining the sampling techniques $p_{s,t}$ using the power heuristic with $\beta = 2$ (see Chapter 9). The image is 500 by 500 with 25 samples per pixel. Observe the caustics on the table, both directly from the spotlight and indirectly from light reflected on the ceiling. The unusual caustic pattern to the left is caused by the square shape of the spotlight's emitting surface.

For comparison, Figure 10.3(b) was computed using standard path tracing with 56 samples per pixel (the same computation time as Figure 10.3(a)). Each path was generated starting from the eye, and direct lighting calculations were used to calculate the contribution at each vertex. Russian roulette was applied to reduce the number of visibility tests. Caustics were rendered using paths that randomly intersected the light sources themselves, since these paths would otherwise not be accounted for. (Direct lighting calculations cannot be used for paths where a light source shines directly on a specular surface.)

Recall that bidirectional path tracing computes a weighted sum of the contributions made by each sampling technique $p_{s,t}$. Figure 10.4 is a visualization of how much each of these techniques contributed toward the final image in Figure 10.3(a). Each row r shows



(a) Bidirectional path tracing with 25 samples per pixel

(b) Standard path tracing with 56 samples per pixel (the same computation time as (a))

Figure 10.3: A comparison of bidirectional and standard path tracing. The test scene contains a spotlight, a floor lamp, a table, and a large glass egg. Image**(a)** was computed with bidirectional path tracing, using the power heuristic with$\beta = 2$ to combine the samples for each path length. The image is 500 by 500 with 25 samples per pixel. Image**(b)** was computed using standard path tracing in the same amount of time (using 56 samples per pixel).

the sampling techniques for a particular path length $k = r + 1$ (for example, the top row shows the sampling techniques for paths of length two). The position of each image in its row indicates how the paths were generated: the s-th image from the left corresponds to paths with s light source vertices (and similarly, the t-th image from the right of each row corresponds to paths with t eye subpath vertices). Notice that the complete set of sampling techniques $p_{s,t}$ is not shown; paths of length $k = 1$ are not shown because the light sources are not directly visible, and paths with zero eye or light subpath vertices are not shown because these images are virtually black (i.e. their weighted contributions are very small for this particular scene). Thus, the full set of images (for paths up to length 5) would have one more image on the left and right side of each row, plus an extra row of three images on the top of the pyramid. (Even though these images are not shown, their contributions are included in Figure 10.3(a).)

The main thing to notice about these images is that different sampling techniques account for different lighting effects in the final image. This implies that most paths are sampled much more efficiently by one technique than the others. For example, consider the image in the middle of the second row of Figure 10.4, corresponding to the sampling technique $p_{2,2}$ (the full-size image is shown in Figure 10.5(a)). These paths were generated by sampling two vertices starting from the eye, and two vertices starting from a light source. Overall, this image is brighter than the other images in its row, which implies that samples from this technique make a larger contribution in general. Yet observe that the glass egg is completely black, and that the inside of the spot light looks at though it were turned off. This implies that the paths responsible for these effects were sampled more efficiently (i.e. with higher probability) by the other two sampling techniques in that row.

As paths get longer and more sampling techniques are used, the effects become much more interesting. For example, consider the rightmost image of the bottom row in Figure 10.4 (enlarged in Figure 10.5(b)), which corresponds to paths with five light vertices and one eye vertex ($p_{5,1}$). Observe the caustics from the spotlight (especially the long “horns” stretching to the right), which are due to internal reflections inside the glass egg. This sampling technique also captures paths that are somehow associated with the corners of the room (where there is a $1/r^2$ singularity in the integrand), and paths along the silhouette edges of the floor lamp’s glossy surfaces. Notice that it would be very difficult to take all of these factors into account if we needed to manually partition paths among the sampling techniques; multiple importance sampling is absolutely essential in order to make bidirectional path tracing work well.

It is also interesting to observe that the middle images of each row in Figure 10.4 are brighter than the rest. This implies that for the majority of paths, the best sampling strategy is to generate an equal number of vertices from both sides. This can be understood in terms of the diffusing properties of light scattering, i.e. the fact that although the emitted radiance is quite concentrated, each scattering step spreads the energy more evenly throughout the scene. The same can be said for the emitted importance function; thus by taking several steps from the light sources and the eye, we have a bigger “target” when attempting to connect the two subpaths.

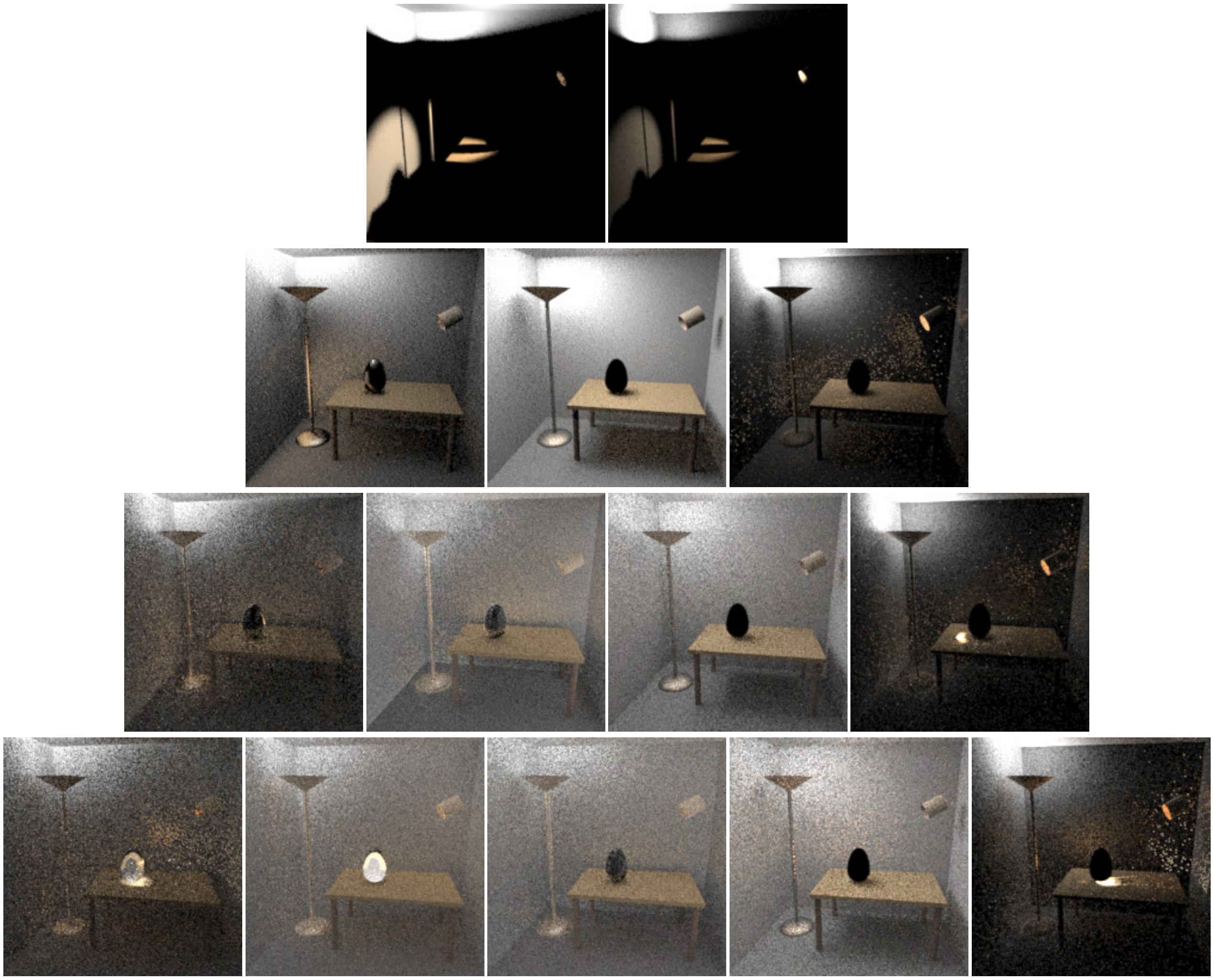
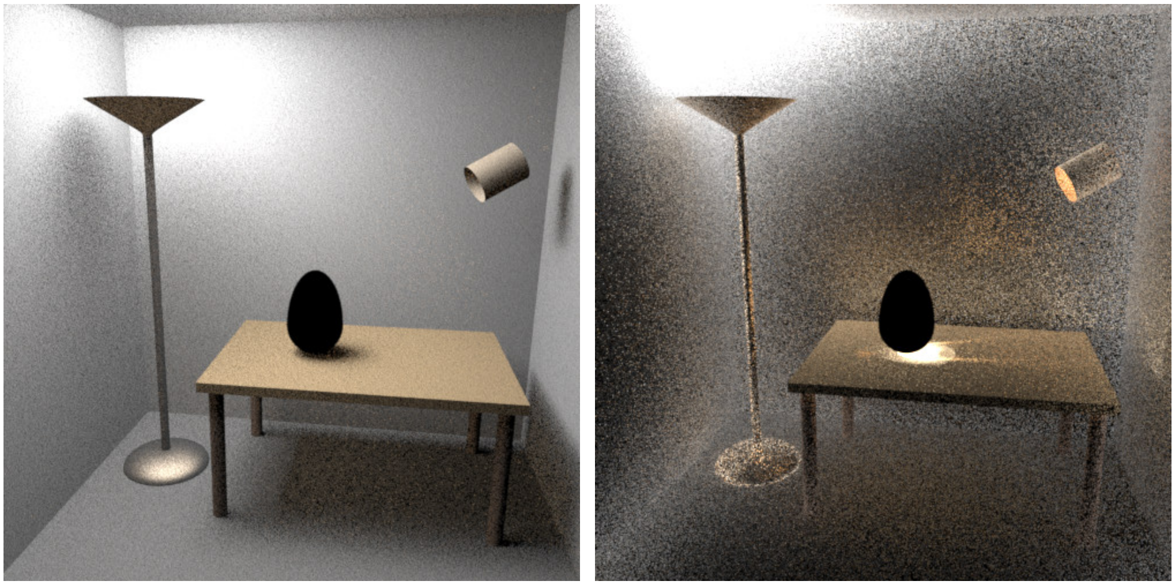


Figure 10.4: This figures shows the weighted contribution that each bidirectional sampling technique $p_{s,t}$ makes to Figure 10.3(a). Each row r shows the contributions of the sampling techniques for a particular path length $k = r + 1$. The position of each image in its row indicates how the paths were generated: the s-th image from the left in each row uses s light subpath vertices, while the t-th image from the right uses t eye subpath vertices. (For example, the top right image uses $s = 2$ light vertices and $t = 1$ eye vertex, while the bottom left image uses $s = 1$ light vertex and $t = 5$ eye vertices.) Note that these images have been over-exposed so that their details can be seen; specifically, the images in row r were over-exposed by r f-stops. The images were made by simply recording the contributions $C_{s,t}$ in a different image for each value of s and t.



(a) Two light vertices, two eye vertices ($p_{2,2}$). **(b)** Five light vertices, one eye vertex ($p_{5,1}$).

Figure 10.5: These are full-size images showing the weighted contributions to Figure 10.3(a) that are due to samples from two particular techniques ($p_{2,2}$ and $p_{5,1}$). These are enlarged versions of the images in Figure 10.4, where $p_{2,2}$ is the middle image of the second row, and $p_{5,1}$ is the rightmost image of the bottom row.

10.6 Comparison with related work

A similar bidirectional path tracing algorithm has been described independently by Lafortune & Willems [1993, 1994]. This section compares the two frameworks in detail, and discusses some possible extensions of the algorithms.

The most important difference between our algorithm and Lafortune's is that the samples are combined using a provably good strategy. This requires a substantially different theoretical basis for the algorithm, in order that multiple importance sampling can be applied. In particular, the path integral formulation of Chapter 8 makes two essential steps: it expresses light transport in the form of an integration problem, and it provides a well-defined basis for comparing the probabilities with which different sampling techniques generate the same path. On the other hand, Lafortune formulates bidirectional path tracing as a recursive evaluation of the *global reflectance distribution function* (GRDF).[10] This is certainly a valid theoretical basis for bidirectional path tracing; however, it does not express the problem in the form needed for multiple importance sampling.

Another difference is that our framework includes several important estimators that are missing from Lafortune's. These include the estimators where zero or one vertices are generated from the eye, and also the naive path tracing estimator where zero vertices are generated from the light source. These estimators are very important for generating caustics and other "difficult" transport paths, and help to make the calculations more robust. We have found that the estimator with one eye vertex ($t = 1$) is surprisingly useful for low-variance rendering in general (it is essentially a particle tracing technique where samples are recorded directly in the image). Also note that although Lafortune describes the estimator with one light vertex ($s = 1$), his framework does not allow the use of direct lighting techniques. This optimization is very important for making bidirectional path tracing competitive with standard path tracing on "normal" scenes, i.e. those where most surfaces are directly lit.

More generally, the two frameworks have a different conception of what bidirectional path tracing is. Lafortune describes it as a specific technique for generating a path from the eye, a path from the light sources, and connecting all pairs of vertices via shadow rays. On the other hand, we view bidirectional path tracing as a family of sampling techniques for paths. The samples from each technique can be generated in any way desired; the specific strategy of connecting every prefix of a light subpath to every suffix of an eye subpath is simply an optimization that allows these samples to be generated more efficiently. Any other desired method of generating the paths could be used instead, e.g. by connecting several different eye subpaths to the same light subpath, or by maintaining a "pool" of eye and light subpaths and making random connections between them, or by generating the paths in more than two pieces (by sampling one or more pieces starting from the middle).

A minor difference between the two frameworks is that Lafortune assumes that light sources are sampled according to emitted power, and that materials are sampled according to the BSDF (exactly). Our formulation of bidirectional path tracing allows the use of arbitrary probability distributions to choose each vertex. The direct lighting strategy applied

[10] The "GRDF" is simply a new name for the kernel of the solution operator $\mathbf{S}$ defined by equation (4.16).

to the case $s = 1$ is a simple example of why this is useful. Other possibilities include: selecting certain scene objects for extra sampling (e.g. portals between adjacent rooms, or small specular objects); using non-local sampling techniques to generate chains of specular vertices (see Section 8.3.4); or using an approximate radiance/importance solution to increase the sample densities in bright/important regions of the scene. Bidirectional path tracing is designed to be used in conjunction with these other sampling techniques, not to replace them.

Another minor difference is that our development is in terms of general linear measurements I_j, rather being limited to pixel estimates only. This means that bidirectional path tracing could be used to compute a view-independent solution, where the equilibrium radiance function L is represented as a linear combination of basis functions $\{B_1, \ldots, B_M\}$.[11] Each measurement I_j is simply the coefficient of B_j, and is defined by

$$I_j \;=\; \langle W_{\mathrm{e}}^{(j)}, L \rangle$$

where $W_{\mathrm{e}}^{(j)} = \tilde{B}_j$ is the corresponding dual basis function.[12] In this situation, each "eye subpath" starts from a surface of the scene rather than the camera lens. By using a fixed number of eye subpaths for each basis function, we can ensure that every coefficient receives at least some minimum number of samples. This bidirectional approach is an unexplored alternative to particle tracing for view-independent solutions, and may help to solve the problem of surface patches that do not receive enough particles. (Note that particle tracing itself corresponds to the case where $t = 0$, and is included as a special case of this framework.)

Lafortune & Willems [1995b] has described an alternative approach to reducing the number of visibility tests. His methods are based on standard Russian roulette and do not attempt to maximize efficiency. We have not made a detailed numerical comparison of the two approaches.

[11] Typically this representation is practical only when most surfaces are diffuse, so that the directional dependence of $L(\mathbf{x}, \omega)$ does not need to be represented.

[12] The dual basis functions satisfy $\langle \tilde{B}_i, B_j \rangle = 1$ when $i = j$, and $\langle \tilde{B}_i, B_j \rangle = 0$ otherwise. For example, when $\{B_1, \ldots, B_M\}$ is an orthonormal basis, then $\tilde{B}_j = B_j$.

10.7 Conclusions

Bidirectional path tracing is an effective rendering algorithm for many kinds of indoor scenes, with or without strong indirect lighting. By using a range of different sampling techniques that take into account different factors of the integrand, it can render a wide variety of lighting effects efficiently and robustly. The algorithm is unbiased, and supports the same range of geometry and materials as standard path tracing.

It is possible to construct scenes where bidirectional path tracing improves on the variance of standard path tracing by an arbitrary amount. To do so, it suffices to increase the intensity of the indirect illumination. In the test case of Figure 10.3, for example, the variance of path tracing increases dramatically as we reduce the size of the directly illuminated area on the ceiling, while bidirectional path tracing is relatively unaffected.

On the other hand, one weakness of the basic bidirectional path tracing algorithm is that there is no intelligent sampling of the light sources. For example, if we were to simulate the lighting in a single room of a large building, most of the light subpaths would start on a light source in a room far from the portion of the scene being rendered, and thus would not contribute. This suggests the idea of sampling light sources according to some estimate of their indirect lighting contribution. Note that methods have already been developed to accelerate the *direct* lighting component when there are many lights, for example by recording information in a spatial subdivision [Shirley et al. 1996]. However, these methods do not help with choosing the initial vertex of a light subpath. In general, we would like to choose a light source that is nearby physically, but is not necessarily directly visible to the viewer.

Similarly, bidirectional path tracing is not suitable for outdoor scenes, or for scenes where the light sources and the viewer are separated by difficult geometry (e.g. a door slightly ajar). In these cases the independently chosen eye and light subpaths will probably not be visible to each other.

Finally, note that bidirectional path tracing can miss the contributions of some paths if point light sources and perfectly specular surfaces are allowed. (This is true of standard path tracing as well.) For example, the algorithm is not capable of rendering caustics from a point source, when viewed indirectly through a mirror using a pinhole lens. This is because bidirectional path tracing is based on local path sampling techniques and thus it is will miss the contributions of paths that do not contain two adjacent non-specular vertices (see Section 8.3.3). However, recall that such paths cannot exist if a finite-aperture lens is used, or if only area light sources are used, or if there are no perfectly specular surfaces in the given scene. Thus bidirectional path tracing is unbiased for all physically valid scene models.

Chapter 11

Metropolis Light Transport

We propose a new Monte Carlo algorithm for solving the light transport problem, called *Metropolis light transport* (MLT). It is inspired by the Metropolis sampling method from computational physics, which is often used for difficult sampling problems in high-dimensional spaces. We show how the Metropolis method can be combined with the path integral framework of Chapter 8, in order to obtain an effective importance algorithm for the space of paths.

Paths are sampled according to the contribution they make to the ideal image, by means of a random walk through path space. Starting with a single seed path, we generate a sequence of light transport paths by applying random mutations (e.g. adding a new vertex to the current path). Each mutation is accepted or rejected with a carefully chosen probability, to ensure that paths are sampled according to the contribution they make to the ideal image. This image is then estimated by sampling many paths, and recording their locations on the image plane.

The resulting algorithm is unbiased, handles general geometric and scattering models, uses little storage, and can be orders of magnitude more efficient than previous unbiased approaches. It performs especially well on problems that are usually considered difficult, e.g. those involving bright indirect light, small geometric holes, or glossy surfaces. Furthermore, it is competitive with previous unbiased algorithms even for scenes with relatively simple illumination.

We start with a high-level overview of the MLT algorithm in Section 11.1, and then we describe its components in detail. Section 11.2 summarizes the classical Metropolis sampling algorithm, as developed in computational physics. Section 11.3 shows how to combine this idea with the path integral framework of Chapter 8, to yield an effective light transport algorithm. Section 11.4 discusses the properties that a good mutation strategy should have, and describes the strategies that we have implemented. In Section 11.5, we describe several refinements to the basic algorithm that can make it work more efficiently. Results are presented in Section 11.6, followed by conclusions and suggested extensions in Section 11.7. To our knowledge, this is the first application of the Metropolis method to transport problems of any kind.

11.1 Overview of the MLT algorithm

To make an image, we sample paths from the light sources to the lens. Each path $\bar{x}$ is a sequence $\mathbf{x}_0 \mathbf{x}_1 \ldots \mathbf{x}_k$ of points on the scene surfaces, where $k \geq 1$ is the length of the path (the number of edges). The numbering of the vertices along the path follows the direction of light flow.

We will show how to define a function f on paths, together with a measure μ, such that $\int_D f(\bar{x})\, d\mu(\bar{x})$ represents the power flowing from the light sources to the image plane along a set of paths D. We call f the *image contribution function*, since $f(\bar{x})$ is proportional to the contribution made to the image by light flowing along $\bar{x}$. (It is closely related to the *measurement contribution function* f_j (described in Chapter 8), which specifies how much each path contributes to a given pixel value.)

Our overall strategy is to sample paths with probability proportional to f, and record the distribution of paths over the image plane. To do this, we generate a sequence of paths $\bar{X}_0$, $\bar{X}_1$, . . ., $\bar{X}_N$, where each $\bar{X}_i$ is obtained by a random mutation to the path $\bar{X}_{i-1}$. The mutations can have almost any desired form, and typically involve adding, deleting, or replacing a small number of vertices on the current path.

However, each mutation has a chance of being rejected, depending on the relative contributions of the old and new paths. For example, if the new path passes through a wall, the mutation will be rejected (by setting $\bar{X}_i = \bar{X}_{i-1}$). The Metropolis framework gives a recipe for determining the acceptance probability for each mutation, such that in the limit

```
function METROPOLIS-LIGHT-TRANSPORT()
    $\bar{x} \leftarrow$ INITIALPATH()
    image $\leftarrow$ { array of zeros }
    for $i \leftarrow 1$ to N
        $\bar{y} \leftarrow$ MUTATE($\bar{x}$)
        $a \leftarrow$ ACCEPTPROB($\bar{x} \rightarrow \bar{y}$)
        if RANDOM() $< a$
            then $\bar{x} \leftarrow \bar{y}$
        RECORDSAMPLE(image, $\bar{x}$)
    return image
```

Figure 11.1: Pseudocode for the Metropolis light transport algorithm.

the sampled paths $\bar{X}_i$ are distributed according to f (this is the *stationary distribution* of the random walk).

As each path is sampled, we update the current image (which is stored in memory as a two-dimensional array of pixel values). To do this, we find the image location (u, v) corresponding to each path sample $\bar{X}_i$, and update the values of those pixels whose filter support contains (u, v). All samples are weighted equally; the light and dark regions of the final image are caused by differences in the number of samples recorded there.[1]

The basic structure of the MLT algorithm is summarized in Figure 11.1. We start with an image of zeros, and a single path $\bar{x}$ that contributes to the desired image. We then repeatedly propose a mutation to the current path, randomly decide whether or not to accept it (according to a carefully chosen probability), and update the image with a sample at the new path location.

The key advantage of the Metropolis approach is that the path space can be explored locally, by favoring mutations that make small changes to the current path. This has several consequences. First, the average cost per sample is small (typically only one or two rays).

[1] At least, this is true of the basic algorithm; in Section 11.5, we describe optimizations that allow the samples to be weighted differently.

Second, once an important path is found, the nearby paths are explored as well, thus amortizing the expense of finding such paths over many samples. Third, the mutation set is easily extended. By constructing mutations that preserve certain properties of the path (e.g. which light source is used) while changing others, we can exploit various kinds of coherence in the scene. It is often possible to handle difficult lighting problems efficiently by designing a specialized mutation in this way.

In the remainder of this chapter, we will describe the MLT algorithm in more detail.

11.2 The Metropolis sampling algorithm

In 1953, Metropolis, Rosenbluth, Rosenbluth, Teller, and Teller introduced an algorithm for handling difficult sampling problems in computational physics [Metropolis et al. 1953]. It was originally used to predict the material properties of liquids, but has since been applied to many areas of physics and chemistry.

The method works as follows (our discussion is based on Kalos & Whitlock [1986]). We are given a state space Ω, and a non-negative function $f : \Omega \to \mathbb{R}^+$. We are also given some initial state $\bar{X}_0 \in \Omega$. The goal is to generate a random walk $\bar{X}_0$, $\bar{X}_1$, ... such that $\bar{X}_i$ is eventually distributed proportionally to f, no matter which state $\bar{X}_0$ we start with. Unlike most sampling methods, the Metropolis algorithm does not require that f must integrate to one.

Each sample $\bar{X}_i$ is obtained by making a random change to $\bar{X}_{i-1}$ (in our case, these are the path mutations). This type of random walk, where $\bar{X}_i$ depends only on $\bar{X}_{i-1}$, is called a *Markov chain*. We let $K(\bar{x} \to \bar{y})$ denote the probability density of going to state $\bar{y}$, given that we are currently in state $\bar{x}$. This is called the *transition function*, and satisfies the condition

$$\int_\Omega K(\bar{x} \to \bar{y}) \, d\mu(\bar{y}) \;=\; 1 \quad \text{for all} \quad \bar{x} \in \Omega \, .$$

11.2.1 The stationary distribution

Each $\bar{X}_i$ is a random variable with some density function p_i, which is determined from p_{i-1} by

$$p_i(\bar{x}) \;=\; \int_\Omega K(\bar{y} \to \bar{x}) \, p_{i-1}(\bar{y}) \, d\mu(\bar{y}) \, . \tag{11.1}$$

With mild conditions on K (discussed further in Section 11.4.1), the p_i will converge to a unique density function p^*, called the ***stationary distribution***. Note that p^* does not depend on the initial state $\bar{X}_0$.

To give a simple example of this idea, consider a state space consisting of n^2 vertices arranged in an $n \times n$ grid. Each vertex is connected to its four neighbors by edges, where the edges "wrap" from left to right and top to bottom as necessary (i.e. with the topology of a torus). A transition consists of randomly moving from the current vertex $\bar{x}$ to one of the neighboring vertices $\bar{y}$ with a probability of $1/5$ each, and otherwise staying at vertex $\bar{x}$.

Suppose that we start at an arbitrary vertex $\bar{X}_0 = \bar{x}_0$, so that $p_0(\bar{x}) = 1$ for $\bar{x} = \bar{x}_0$, and $p_0(\bar{x}) = 0$ otherwise. Then after one transition, $\bar{X}_1$ is distributed with equal probability among $\bar{x}_0$ and its four neighbors. Similarly, $\bar{X}_2$ is randomly distributed among 13 vertices (although not with equal probability). If this process is continued, eventually p_i converges to a fixed density function p^*, which necessarily satisfies

$$p^*(\bar{x}) \;=\; \sum_{\bar{y}} K(\bar{y} \!\rightarrow\! \bar{x})\, p^*(\bar{y})\,.$$

For this example, p^* is the uniform density $p^*(\bar{x}) = 1/n^2$.

11.2.2 Detailed balance

In a typical physical system, the transition function K is determined by the physical laws governing the system. Given some arbitrary initial state, the system then evolves towards equilibrium through transitions governed by K.

The Metropolis algorithm works in the opposite direction. The idea is to invent or construct a transition function K whose resulting stationary distribution will be proportional to the given f, and which will converge to f as quickly as possible. The technique is simple, and has an intuitive physical interpretation called *detailed balance*.

Given $\bar{X}_{i-1}$, we obtain $\bar{X}_i$ as follows. First, we choose a tentative sample $\bar{X}'_i$, which can be done in almost any way desired. This is represented by the *tentative transition function* T, where $T(\bar{x} \!\rightarrow\! \bar{y})$ gives the probability density that $\bar{X}'_i = \bar{y}$ given that $\bar{X}_{i-1} = \bar{x}$.

The tentative sample is then either accepted or rejected, according to an acceptance probability $a(\bar{x} \to \bar{y})$ which will be defined below. That is, we let

$$\bar{X}_i \;=\; \begin{cases} \bar{X}'_i & \text{with probability } a(\bar{X}_{i-1} \to \bar{X}'_i)\,, \\ \bar{X}_{i-1} & \text{otherwise}\,. \end{cases} \tag{11.2}$$

To see how to set $a(\bar{x} \to \bar{y})$, suppose that we have already reached equilibrium, i.e. p_{i-1} is proportional to f. We must define $K(\bar{x} \to \bar{y})$ such that the equilibrium is maintained. To do this, consider the density of transitions between any two states $\bar{x}$ and $\bar{y}$. From $\bar{x}$ to $\bar{y}$, the transition density is proportional to $f(\bar{x})\, T(\bar{x} \to \bar{y})\, a(\bar{x} \to \bar{y})$, and a similar statement holds for the transition density from $\bar{y}$ to $\bar{x}$. To maintain equilibrium, it is sufficient that these densities be equal:

$$f(\bar{x})\, T(\bar{x} \to \bar{y})\, a(\bar{x} \to \bar{y}) \;=\; f(\bar{y})\, T(\bar{y} \to \bar{x})\, a(\bar{y} \to \bar{x})\,, \tag{11.3}$$

a condition known as *detailed balance*. We can verify that if $p_{i-1} \propto f$ and condition (11.3) holds, then equilibrium is preserved:

$$\begin{aligned} p_i(\bar{x}) \;&=\; p_{i-1}(\bar{x}) \left[1 - \int_\Omega T(\bar{x} \to \bar{y})\, a(\bar{x} \to \bar{y})\, d\mu(\bar{y})\right] + \int_\Omega p_{i-1}(\bar{y})\, T(\bar{y} \to \bar{x})\, a(\bar{y} \to \bar{x})\, d\mu(\bar{y}) \\ &=\; p_{i-1}(\bar{x}) + \int_\Omega \left[p_{i-1}(\bar{x})\, T(\bar{x} \to \bar{y})\, a(\bar{x} \to \bar{y}) \;-\; p_{i-1}(\bar{y})\, T(\bar{y} \to \bar{x})\, a(\bar{y} \to \bar{x})\right]\, d\mu(\bar{y}) \\ &=\; p_{i-1}(\bar{x})\,. \end{aligned}$$

Thus the unique equilibrium distribution must be proportional to f.

11.2.3 The acceptance probability

Recall that f is given, and T was chosen arbitrarily. Thus, equation (11.3) is a condition on the ratio $a(\bar{x} \to \bar{y}) / a(\bar{y} \to \bar{x})$. In order to reach equilibrium as quickly as possible, the best strategy is to make $a(\bar{x} \to \bar{y})$ and $a(\bar{y} \to \bar{x})$ as large as possible [Peskun 1973], which is achieved by letting

$$a(\bar{x} \to \bar{y}) \;=\; \min \left\{ 1, \frac{f(\bar{y})\, T(\bar{y} \to \bar{x})}{f(\bar{x})\, T(\bar{x} \to \bar{y})} \right\}\,. \tag{11.4}$$

According to this rule, transitions in one direction are always accepted, while in the other direction they are sometimes rejected, such that the expected number of moves each way is the same.

11.2.4 Comparison with genetic algorithms

The Metropolis method differs from genetic algorithms [Goldberg 1989] in several ways. First, they have different purposes: genetic algorithms are intended for optimization problems, while the Metropolis method is intended for sampling problems (there is no search for an optimum value). Genetic algorithms work with a population of individuals, while Metropolis stores only a single current state. Finally, genetic algorithms have much more freedom in choosing the allowable mutations, since they do not need to compute the conditional probability of their actions.

Beyer & Lange [1994] have applied genetic algorithms to the problem of integrating radiance over a hemisphere. They start with a population of rays (actually directional samples), which are evolved to improve their distribution with respect to the incident radiance at a particular surface point. However, their methods do not seem to lead to a feasible light transport algorithm.

11.3 Theoretical formulation of Metropolis light transport

To complete the MLT algorithm outlined in Section 11.1, there are several tasks. First, we must formulate the light transport problem so that it fits the Metropolis framework. Second, we must show how to avoid *start-up bias*, a problem that affects many Metropolis applications. Most importantly, we must design a suitable set of mutations on paths, such that the Metropolis method will work efficiently. In this section we deal with the first two problems, by showing how the Metropolis method can be adapted to estimate all of the pixel values of an image simultaneously and without bias.

Recall that according to the path integral framework of Chapter 8, each measurement I_j can be expressed in the form

$$I_j \;=\; \int_\Omega f_j(\bar{x})\, d\mu(\bar{x})\,,$$

where Ω is the set of all transport paths, μ is the area-product measure, and f_j is the measurement contribution function. In our case, the measurements I_j are pixel values. This implies that each integrand f_j has the form

$$f_j(\bar{x}) \;=\; h_j(\bar{x})\, f(\bar{x})\,, \tag{11.5}$$

where h_j represents the filter function for pixel j, and f represents all the other factors of f_j (which are the same for all pixels). In physical terms, $\int_D f(\bar{x})\, d\mu(\bar{x})$ represents the radiant power received by the image region of the image plane along a set D of paths.[2] Note that h_j depends only on the last edge $\mathbf{x}_{k-1}\mathbf{x}_k$ of the path, which we call the *lens edge*.

An image can now be computed by sampling N paths $\bar{X}_i$ according to some density function p, and using the identity

$$I_j \;=\; E\left[\frac{1}{N}\sum_{i=1}^{N} \frac{h_j(\bar{X}_i)\, f(\bar{X}_i)}{p(\bar{X}_i)}\right]\,. \tag{11.6}$$

Notice that if we could take samples according to the density function $p = (1/b)\, f$ (where b is the normalization constant $\int_\Omega f(\bar{x})\, d\mu(\bar{x})$), the estimate for each pixel would simply be

$$I_j \;=\; E\left[\frac{1}{N}\sum_{i=1}^{N} b\, h_j(\bar{X}_i)\right]\,.$$

This equation can be evaluated efficiently for all pixels at once, since each path contributes to only a few pixel values.

This approach requires the evaluation of b, and the ability to sample from a density function proportional to f. Both of these are hard problems. For the second part, the Metropolis algorithm will help; however, the samples $\bar{X}_i$ will have the desired distribution only in the limit as $i \to \infty$. In typical Metropolis applications, this is handled by starting in some fixed initial state $\bar{X}_0$, and discarding the first k samples until the random walk has approximately converged to the equilibrium distribution. However, it is often difficult to know how large k should be. If it is too small, then the samples will be strongly influenced by the choice of the initial path $\bar{X}_0$, which will bias the results (this is called *start-up bias*).

[2] We define $f(\bar{x})$ to be zero for paths that do not contribute to any pixel value (so that we do not waste any samples there).

11.3.1 Eliminating start-up bias

We show how the MLT algorithm can be initialized to avoid start-up bias. The idea is to start the walk in a random initial state $\bar{X}_0$, which is sampled from some convenient density function p_0 on paths (we use bidirectional path tracing for this purpose). To compensate for the fact that p_0 is not the desired equilibrium distribution $p^* \;=\; (1/b)\, f$, the sample $\bar{X}_0$ is assigned a weight:

$$W_0 \;=\; f(\bar{X}_0) \,/\, p_0(\bar{X}_0)\,.$$

Thus after one sample, the estimate for pixel j is $W_0\, h_j(\bar{X}_0)$ (see equation (11.6). All of these quantities are computable since $\bar{X}_0$ is known.

Additional samples $\bar{X}_1$, $\bar{X}_2$, ..., $\bar{X}_N$ are generated by mutating $\bar{X}_0$ according to the Metropolis algorithm (using f as the target density). Each of the $\bar{X}_i$ has a different density function p_i, which only approaches the stationary distribution $p^* \;=\; (1/b)\, f$ as $i \rightarrow \infty$. To avoid bias, however, it is sufficient to assign these samples the same weight $W_i \;=\; W_0$ as the original sample, and use the following estimate for pixel j:

$$I_j \;=\; E\left[\frac{1}{N} \sum_{i=1}^{N} W_i\, h_j(\bar{X}_i) \right]\,. \tag{11.7}$$

We give a proof that this estimate is unbiased in Appendix 11.A. However, the following explanation may give some additional insight. Recall that the initial path is a random variable, so that the expected value in (11.7) is an average over all possible values of $\bar{X}_0$. Thus, consider a large group of initial paths $\bar{X}_{0,j}$ obtained by sampling p_0 many times. If p_0 is the stationary distribution $(1/b)\, f$, and all the paths are weighted equally, then this group of paths is in equilibrium: the distribution of paths does not change as mutations are applied. Now suppose that we again sample a large group of initial paths, this time from an arbitrary density function p_0, and that we assign each path the weight $f(\bar{X}_{0,j})/p_0(\bar{X}_{0,j})$. Even though this does not give the desired distribution of *paths*, the distribution of *weight* is proportional to the desired equilibrium f. The equilibrium is preserved as the paths are mutated (just as in the first case), which leads to an unbiased estimate of I_j.

This technique for removing start-up bias is not specific to light transport. However, it requires the existence of an alternative sampling method p_0, which is difficult to obtain in some cases. (Often the reason for using the Metropolis method in the first place is the lack of suitable alternatives.)

11.3.2 Initialization

In practice, initializing the MLT algorithm with a single seed path does not work well. If we generate only one path $\bar{X}_0$ (e.g. using bidirectional path tracing), it is likely that $W_0 = 0$ (for example, the path may go through a wall). Since all subsequent samples use the same weight $W_i = W_0$, this would lead to a completely black final image. Conversely, the initial weight W_0 on other runs may be much larger than expected. This does not contradict the fact that the algorithm is unbiased, since bias refers only to the *expected* value on a particular run.

The obvious solution is to run n copies of the algorithm in parallel (with different random initial paths), and accumulate all the samples into one image. The strategy we have implemented has two phases. First we sample a moderately large number of paths $\bar{X}_{0,1}$, …, $\bar{X}_{0,n}$, and let $W_{0,1}$, …, $W_{0,n}$ be the corresponding weights. We then select a representative sample of n' of these paths (where n' is much smaller than n), and assign them equal weights. (The reasons for doing this are discussed below.) These paths are used as independent seeds for the Metropolis phase of the algorithm.

Specifically, each representative path $\bar{X}'_{0,i}$ is chosen from among the initial paths $\bar{X}_{0,j}$ according to discrete probabilities that are proportional to $W_{0,j}$. All of these paths $\bar{X}'_{0,i}$ are assigned the same weight:

$$W'_{0,i} \;=\; \frac{1}{n} \sum_{j=1}^{n} W_{0,j} \,.$$

It is straightforward to show that this resampling procedure is unbiased.[3]

The value of n is determined indirectly, by generating a fixed number of eye and light subpaths (e.g. 10 000 pairs), and considering all the ways to link the vertices of each pair. Note that it is not necessary to save all of these paths in order to apply the resampling step; they can be regenerated by restarting the random number generator with the same seed.

[3] The resampling can be optimized slightly by choosing the new paths with equal spacing in the cumulative weight distribution of the $\bar{X}_{0,j}$; this ensures that the same path is not selected twice, unless its weight is at least a fraction $1/n'$ of the total.

It is often reasonable to choose $n' = 1$ (i.e. to initialize the Metropolis algorithm with a single representative seed path). In this case, the purpose of sampling n paths in the first phase is to estimate the mean value of W_0, which determines the absolute image brightness.[4] If the image is desired only up to a constant scale factor, then the first phase can be terminated as soon as a single path with $f(\bar{x}) > 0$ is found. The main reasons for retaining more than one seed path (i.e. for choosing $n' > 1$) are to implement convergence tests (see below) or lens subpath mutations (see Section 11.4.4).

Effectively, we have separated the image computation into two subproblems. The initialization phase estimates the overall image brightness, while the Metropolis phase determines the relative pixel intensities across the image. The effort spent on each phase can be decided independently. In practice, however, the initialization phase is a negligible part of the total computation time. (Observe that even if the algorithm is initialized using $100\,000$ bidirectional samples, this would represent less than one sample per pixel for an image of reasonable size.)

11.3.3 Convergence tests

Another reason to run several copies of the algorithm in parallel is that it facilitates convergence testing. (We cannot apply the usual variance tests to the samples generated by a single run of the Metropolis algorithm, since consecutive samples are highly correlated.)

To test for convergence, the Metropolis phase can be started with n' independent seed paths, whose contributions to the image are recorded separately (in the form of n' separate images). This is done only for a small representative fraction of the pixels, since it would be too expensive to maintain many copies of a large image. For each such pixel, we thus have available n' independent, unbiased samples of its true value. (Each sample value changes as the algorithm proceeds, since it depends on how many path mutations have contributed to the specified pixel of a particular test image.) The sample variance of these pixels can then be tested periodically, until the results are within prespecified bounds. Notice that unlike most graphics problems, the number of independent samples per pixel remains constant (at

[4] More precisely, $E[W_0] = \int f = b$, which represents the total power falling on the image region of the film plane.

n') as the algorithm proceeds — it is the *values* of the samples that change.

If the radiance values that contribute to a given pixel can be bounded in advance, more advanced convergence techniques could in theory be applied. In particular Dagum et al. [1995] have proposed an algorithm that can estimate the expected value of a random variable Z to within a factor of $(1 + \epsilon)$ with a guaranteed probability of at least $1 - \delta$. They assume only that Z is bounded within a known range $[0, M]$. Furthermore, the number of independent samples used by their algorithm is proven to optimal for every given ϵ, δ, and Z to within a constant factor. In the case of the Metropolis light transport, observe that an arbitrary number of independent samples can be generated by restarting the algorithm with new seed paths. However, once again it seems impractical to apply this technique to every pixel of an image.

These convergence testing procedures add a small amount of bias, but this is inevitable for any technique that makes guarantees about the quality of its results. Note that the first technique we described bounds the sample variance of the test pixels, while the second technique bounds the actual error. Also note that unbiased techniques such as two-stage adaptive sampling [Kirk & Arvo 1991] do not make any guarantees about the final image quality, due to the possibility of outlying samples during the second stage of sampling.

Finally, note that in all of our tests the number of mutations was specified manually, both to eliminate bias and so that we would have explicit control over the computation time.

11.3.4 Spectral sampling

Our discussion so far has been limited to monochrome images, but the modifications for color are straightforward.

We represent BSDF's and light sources as point-sampled spectra (although it would be easy to use some other representation). Given a path, we compute the energy delivered to the lens at each of the sampled frequencies. The resulting spectrum is then converted to a tristimulus color value (we use RGB) before it is accumulated in the current image.

The image contribution function f is redefined to compute the luminance of the corresponding path spectrum. This implies that path samples will be distributed according to the luminance of the ideal image, and that the luminance of every filtered image sample will be the same (irrespective of its color). Effectively, each color component c_i is sampled with an estimator of the form c_i/p, where p is proportional to the luminance.

Since the human eye is substantially more sensitive to luminance differences than other color variations, this choice helps to minimize the apparent noise.[5]

11.4 Good mutation strategies

The main disadvantage of the Metropolis method is that consecutive samples are correlated, which leads to higher variance than we would get with independent samples. This can happen either because the proposed mutations to the paths are very small, or because too many mutations are rejected.

Correlation can be minimized by choosing a suitable set of path mutations. We first consider some of the properties that these mutations should have, in order to minimize the error in the final image. Then we describe three specific mutation strategies that we have implemented, namely *bidirectional mutations*, *perturbations*, and *lens subpath mutations*. These strategies are designed to satisfy different subsets of the goals mentioned below; our implementation uses a mixture of all three (as we will discuss in Section 11.4.5).

11.4.1 Desirable mutation properties

In this section, we describe the properties that a good mutation strategy should have. These are the main factors that need to be considered when a mutation strategy is designed.

High acceptance probability. If the acceptance probability $a(\bar{x} \to \bar{y})$ is very small on the average, there will be long path sequences of the form $\bar{x}$, $\bar{x}$, $\ldots$, $\bar{x}$ due to rejections. This leads to many samples at the same point on the image plane, and appears as noise.

[5] Another way to handle color is to have a separate run for each frequency. However, this is inefficient (we get less information from each path) and leads to unnecessary color noise. Note that it is *not* necessary to have a separate run at each wavelength in order to handle dispersion (i.e. a refractive index that varies with wavelength). It can be handled perfectly well in the model described above, by randomly sampling a spectral band only when a dispersive material is actually encountered (and using a weight of the usual form f/p).

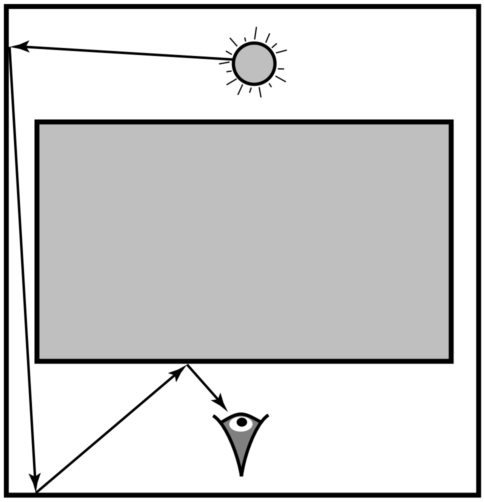


Figure 11.2: If only additions and deletions of a single vertex are allowed, then paths cannot mutate from one side of the barrier to the other.

Large changes to the path. Even if the acceptance probability for most mutations is high, samples will still be highly correlated if the proposed path mutations are too small. It is important to propose mutations that make substantial changes to the current path, such as increasing the path length, or replacing a specular bounce with a diffuse one.

Ergodicity. If the allowable mutations are too restricted, it is possible for the random walk to get "stuck" in some subregion of the path space (i.e. one where the integral of f is less than b). To see how this can happen, consider Figure 11.2, and suppose that we only allow mutations that add or delete a single vertex. In this case, there is no way for the path to mutate from one side of the barrier to the other, and we will miss part of the path space.

Technically, we want to ensure that the random walk converges to an *ergodic* state. This means that no matter how $\bar{X}_0$ is chosen, it converges to the same stationary distribution p^*. To do this, it is sufficient to ensure that $T(\bar{x} \to \bar{y}) > 0$ for every pair of states $\bar{x}$, $\bar{y}$ with $f(\bar{x}) > 0$ and $f(\bar{y}) > 0$. In our implementation, this is always true (see Section 11.4.2).

Changes to the image location. To minimize correlation between the sample locations on the image plane, it is desirable for mutations to change the lens edge $\mathbf{x}_{k-1}\mathbf{x}_k$. Mutations to other portions of the path do not provide information about the path distribution over the image plane, which is what we are most interested in.

Stratification. Another potential weakness of the Metropolis approach is the random distribution of samples across the image plane. This is commonly known as the “balls in bins” effect: if we randomly throw n balls into n bins, we cannot expect one ball per bin. (Many bins may be empty, while the fullest bin is likely to contain $\Theta(\log n)$ balls.) In an image, this unevenness in the distribution produces noise.

For some kinds of mutations, this effect is difficult to avoid. However, it is worthwhile to consider mutations for which some form of stratification is possible.

Low cost. It is also desirable that mutations be inexpensive. Generally, this is measured by the number of rays cast, since the other costs are relatively small.

We now consider some specific mutation strategies that address these goals. Note that the Metropolis framework allows us greater freedom than standard Monte Carlo algorithms in designing sampling strategies. This is because we only need to compute the conditional probability $T(\bar{x} \to \bar{y})$ of each mutation: in other words, the mutation strategy is allowed to depend on the current path.

11.4.2 Bidirectional mutations

Bidirectional mutations are the foundation of the MLT algorithm. They are responsible for making large changes to the path, such as modifying its length. The basic idea is simple: we choose a subpath of the current path $\bar{x}$, and replace it with a different subpath. We divide this into several steps.

First, the subpath to delete is chosen. Given the current path $\bar{x} = \mathbf{x}_0 \ldots \mathbf{x}_k$, we assign a probability $p_{\mathrm{d}}[l, m]$ to the deletion of each subpath $\mathbf{x}_l \ldots \mathbf{x}_m$. The endpoints of this subpath are not included, so that $\mathbf{x}_l \ldots \mathbf{x}_m$ consists of $m - l$ edges and $m - l - 1$ vertices (with indices satisfying $-1 \leq l < m \leq k + 1$).

In our implementation, the deletion probability $p_{\mathrm{d}}[l, m]$ is a product two factors. The first factor $p_{\mathrm{d},1}$ depends only on the subpath length (i.e. the number of edges); its purpose is to favor the deletion of short subpaths. (These are less expensive to replace, and yield mutations that are more likely to be accepted, since they make a smaller change to the current path). The purpose of the second factor $p_{\mathrm{d},2}$ is to avoid mutations with low acceptance probabilities; it will be described in Section 11.5.

The density function $p_{\mathrm{d}}[l, m]$ is normalized and sampled to determine the deleted subpath. At this point, $\bar{x}$ has been split into two (possibly empty) pieces $\mathbf{x}_0 \ldots \mathbf{x}_l$ and $\mathbf{x}_m \ldots \mathbf{x}_k$. To complete the mutation, we must generate a new subpath that connects these two pieces.

We start by choosing the number of vertices l' and m' to be added to each side. This is done in two steps: first, we choose the new subpath length, $k_{\mathrm{a}} = l' + m' + 1$. It is desirable that the old and new subpath lengths be similar, since this will tend to increase the acceptance probability (i.e. it represents a smaller change to the path). Thus we choose k_{a} according to a discrete distribution $p_{\mathrm{a},1}$ which assigns a high probability to keeping the total path length the same. Then, we choose specific values for l' and m' (subject to the condition $l' + m' + 1 = k_{\mathrm{a}}$), according to another discrete distribution $p_{\mathrm{a},2}$ that assigns equal probability to each candidate value of l'. For convenience, we let $p_{\mathrm{a}}[l', m']$ denote the product of $p_{\mathrm{a},1}$ and $p_{\mathrm{a},2}$.

To sample the new vertices, we add them one at a time to the appropriate subpath. This involves first sampling a direction according to the BSDF at the current subpath endpoint (or a convenient approximation, if sampling from the exact BSDF is difficult), followed by casting a ray to find the first surface intersected. An initially empty subpath is handled by choosing a random point on a light source or the lens as appropriate.

Finally, we join the new subpaths together, by testing the visibility between their endpoints. If the path is obstructed, the mutation is immediately rejected. This also happens if any of the ray casting operations failed to intersect a surface.

Notice that there is a non-zero probability of throwing away the entire path, and generating a new one from scratch. This automatically ensures the ergodicity condition (Section 11.4.1), so that the algorithm can never get “stuck” forever in a small subregion of the path space. (However, if the mutations are poorly chosen then the algorithm might get stuck for a long finite time.)

Parameter values. The following values have provided reasonable results on our test cases. For the probability $p_{\mathrm{d},1}[k_{\mathrm{d}}]$ of deleting a subpath of length $k_{\mathrm{d}} = m - l$, we use

$p_{\mathrm{d},1}[1] = 0.25$, $p_{\mathrm{d},1}[2] = 0.5$, and $p_{\mathrm{d},1}[k_\mathrm{d}] = 2^{-k_\mathrm{d}}$ for $k_\mathrm{d} \geq 3$. For the probability $p_{\mathrm{a},1}[k_\mathrm{a}]$ of adding a subpath of length k_a, we use $p_{\mathrm{a},1}[k_\mathrm{d}] = 0.5$, $p_{\mathrm{a},1}[k_\mathrm{d} \pm 1] = 0.15$, and $p_{\mathrm{a},1}[k_\mathrm{d} \pm j] = 0.2(2^{-j})$ for $j \geq 2$.

11.4.2.1 Evaluation of the acceptance probability.

Observe that the acceptance probability $a(\bar{x} \to \bar{y})$ from (11.4) can be written as the ratio

$$a(\bar{x} \to \bar{y}) \;=\; \frac{R(\bar{x} \to \bar{y})}{R(\bar{y} \to \bar{x})}\,, \qquad \text{where} \qquad R(\bar{x} \to \bar{y}) \;=\; \frac{f(\bar{y})}{T(\bar{x} \to \bar{y})}\,. \tag{11.8}$$

The form of $R(\bar{x} \to \bar{y})$ is very similar to the sample value $f(\bar{y})/p(\bar{y})$ that is computed by standard Monte Carlo algorithms; we have simply replaced an absolute probability $p(\bar{y})$ by a conditional probability $T(\bar{x} \to \bar{y})$.

Specifically, $T(\bar{x} \to \bar{y})$ is the product of the discrete probability $p_\mathrm{d}[l, m]$ for deleting the subpath $\mathbf{x}_l \ldots \mathbf{x}_m$, and the probability density for generating the $l' + m'$ new vertices of $\bar{y}$. To calculate the latter, we must take into account all $l' + m' + 1$ ways that the new vertices can be split between subpaths generated from $\mathbf{x}_l$ and $\mathbf{x}_m$. (Although these vertices were generated by a particular choice of l', the probability $T(\bar{x} \to \bar{y})$ must take into account all of these ways of going from state $\bar{x}$ to $\bar{y}$.) Note that the unchanged portions of $\bar{x}$ do not contribute to the calculation of $T(\bar{x} \to \bar{y})$. It is also convenient to ignore the factors of $f(\bar{x})$ and $f(\bar{y})$ that are shared between the paths, since this does not change the result.

An example. Let $\bar{x}$ be a path $\mathbf{x}_0\mathbf{x}_1\mathbf{x}_2\mathbf{x}_3$, and suppose that the random mutation step has deleted the edge $\mathbf{x}_1\mathbf{x}_2$ (see Figure 11.3). It is replaced by new vertex $\mathbf{z}_1$ by casting a ray from $\mathbf{x}_1$, so that the new path is

$$\bar{y} \;=\; \mathbf{x}_0\,\mathbf{x}_1\,\mathbf{z}_1\,\mathbf{x}_2\,\mathbf{x}_3\,.$$

This corresponds to the random choices $l = 1$, $m = 2$, $l' = 1$, $m' = 0$.

Let $P_{\sigma^\perp}(\mathbf{x} \to \mathbf{x}')$ denote the probability density of sampling the direction from $\mathbf{x}$ to $\mathbf{x}'$, measured with respect to projected solid angle.[6] Then the probability density of sampling

[6] Recall that if $P_\sigma(\mathbf{x} \to \mathbf{x}')$ is the density with respect to ordinary solid angle, then $P_{\sigma^\perp} = P_\sigma \,/\, |\cos(\theta_\mathrm{o})|$, where θ_o is the angle between $\mathbf{x} \to \mathbf{x}'$ and the surface normal at $\mathbf{x}$.

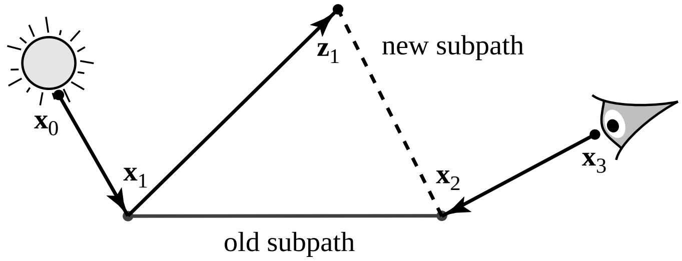


Figure 11.3: A simple example of a bidirectional mutation. The original path $\bar{x} = \mathbf{x}_0\mathbf{x}_1\mathbf{x}_2\mathbf{x}_3$ is modified by deleting the edge $\mathbf{x}_1\mathbf{x}_2$ and replacing it with a new vertex $\mathbf{z}_1$. The new vertex is generated by sampling a direction at $\mathbf{x}_1$ (according to the BSDF) and casting a ray. This yields a mutated path $\bar{y} = \mathbf{x}_0\mathbf{x}_1\mathbf{z}_1\mathbf{x}_2\mathbf{x}_3$.

the vertex $\mathbf{x}'$ (measured with respect to surface area) is given by $P_{\sigma^\perp}(\mathbf{x} \to \mathbf{x}')\, G(\mathbf{x} \leftrightarrow \mathbf{x}')$.

We now have all of the information necessary to compute $R(\bar{x} \to \bar{y})$. From definition (8.7), the numerator is

$$\begin{aligned} f(\bar{y}) \;&=\; f_\mathrm{s}(\mathbf{x}_0 \to \mathbf{x}_1 \to \mathbf{z}_1)\, G(\mathbf{x}_1 \leftrightarrow \mathbf{z}_1)\, f_\mathrm{s}(\mathbf{x}_1 \to \mathbf{z}_1 \to \mathbf{x}_2) \\ &\qquad \cdot G(\mathbf{z}_1 \leftrightarrow \mathbf{x}_2)\, f_\mathrm{s}(\mathbf{z}_1 \to \mathbf{x}_2 \to \mathbf{x}_3)\,, \end{aligned}$$

where the factors shared between $R(\bar{x} \to \bar{y})$ and $R(\bar{y} \to \bar{x})$ have been omitted. The denominator is

$$\begin{aligned} T(\bar{x} \to \bar{y}) \;&=\; p_\mathrm{d}[1,2]\, \Big\{ p_\mathrm{a}[1,0]\, P_{\sigma^\perp}(\mathbf{x}_1 \to \mathbf{z}_1)\, G(\mathbf{x}_1 \leftrightarrow \mathbf{z}_1) \\ &\qquad + p_\mathrm{a}[0,1]\, P_{\sigma^\perp}(\mathbf{x}_2 \to \mathbf{z}_1)\, G(\mathbf{x}_2 \leftrightarrow \mathbf{z}_1) \Big\}\,. \end{aligned}$$

In a similar way, we find that the factor $R(\bar{y} \to \bar{x})$ for the mutation in the reverse direction is given by

$$R(\bar{y} \to \bar{x}) \;=\; \frac{f_\mathrm{s}(\mathbf{x}_0 \to \mathbf{x}_1 \to \mathbf{x}_2)\, G(\mathbf{x}_1 \leftrightarrow \mathbf{x}_2)\, f_\mathrm{s}(\mathbf{x}_1 \to \mathbf{x}_2 \to \mathbf{x}_3)}{p_\mathrm{d}[1,3]\, p_\mathrm{a}[0,0]}\,,$$

where p_d and p_a now refer to the path $\bar{y}$.

Implementation. We now describe how to compute the acceptance probability for bidirectional mutations in general form, and we also discuss how to implement this calculation efficiently.

Let $\bar{x} = \mathbf{x}_0 \ldots \mathbf{x}_k$ be the old path, let $\mathbf{x}_l \ldots \mathbf{x}_m$ be the deleted subpath, and let $\mathbf{z}_1 \ldots \mathbf{z}_{k_\mathrm{a}-1}$ be the vertices of the new subpath. This yields a mutated path $\bar{y}$ of the form

$$\begin{aligned} \bar{y} &= \mathbf{y}_0 \ldots \mathbf{y}_{k'} \\ &= \mathbf{x}_0 \ldots \mathbf{x}_l \, \mathbf{z}_1 \ldots \mathbf{z}_{k_\mathrm{a}-1} \, \mathbf{x}_m \ldots \mathbf{x}_k \, , \end{aligned}$$

where $k' = k - k_\mathrm{d} + k_\mathrm{a}$ is the length of the new path $\bar{y}$. (Recall that $k_\mathrm{d} = m - l$ and $k_\mathrm{a} = l' + m' + 1$ represent the number of edges in the old and new subpaths respectively.

Rather than evaluating the ratio $R(\bar{x} \to \bar{y})$ as we did in the example above, it is more convenient to evaluate its reciprocal:[7]

$$Q(\bar{x} \to \bar{y}) \;=\; \frac{1}{R(\bar{x} \to \bar{y})} \;=\; \frac{T(\bar{x} \to \bar{y})}{f(\bar{y})} \, . \tag{11.9}$$

This quantity can be evaluated efficiently using the same techniques that were developed for bidirectional path tracing in Chapter 10. In particular, suppose that we split $\bar{y}$ into two pieces, using the i-th edge of the new subpath as the connecting edge. In other words, consider the light subpath

$$\mathbf{y}_0 \ldots \mathbf{y}_{l+i-1} \;=\; \mathbf{x}_0 \ldots \mathbf{x}_l \, \mathbf{z}_1 \ldots \mathbf{z}_{i-1} \, ,$$

and the eye subpath

$$\mathbf{y}_{l+i} \ldots \mathbf{y}_{k'} \;=\; \mathbf{z}_i \ldots \mathbf{z}_{k_\mathrm{a}-1} \mathbf{x}_m \ldots \mathbf{x}_k \, ,$$

where $1 \le i \le k_\mathrm{a}$. These subpaths have $s = l + i$ and $t = (k' + 1) - (l + i)$ vertices respectively. Now let C_i^bd be the unweighted contribution from bidirectional tracing that would be computed in this situation:

$$C_i^\mathrm{bd} \;=\; C^*_{s,t} \, ,$$

[7] The quantity $Q(\bar{x} \to \bar{y})$ has an interesting interpretation: it is simply the probability density of sampling the path $\bar{y}$, measured with respect to the *image contribution measure* defined by $\mu^\mathrm{i}(D) = \int_D f(\bar{x}) \, \mu(\bar{x})$. This measure μ^i is closely related to the *measurement contribution measure* μ_j^m defined in Appendix 8.A, except that it corresponds to the contribution made by a set of paths D to the entire image rather than to an individual measurement I_j.

where $C^*_{s,t}$ has already been defined in equation (10.5) . The value of $Q(\bar{x} \to \bar{y})$ can then be expressed as

$$Q(\bar{x} \to \bar{y}) \;=\; p_{\mathrm{d}}[l, m] \sum_{i=1}^{k_{\mathrm{a}}} \frac{p_{\mathrm{a}}[i-1, k_{\mathrm{a}}-i]}{C^{\mathrm{bd}}_i} \,. \tag{11.10}$$

To evaluate this sum efficiently we first compute the unweighted bidirectional contribution $C^{\mathrm{bd}}_{l'+1}$ (corresponding to the way the path was actually generated, using l' new light vertices and m' new eye vertices). This is done using the weights α^L_s, α^E_t and the connecting factor $c_{s,t}$ defined in Chapter 10. If the contribution $C^{\mathrm{bd}}_{l'+1}$ evaluates to zero (for example if the visibility test fails), then the mutation is immediately rejected. Otherwise, we compute the reciprocal value $1/C^{\mathrm{bd}}_{l'+1}$, and find the values of the other factors $1/C^{\mathrm{bd}}_i$ by iteratively applying the relationship (10.9) given in Chapter 10. This calculation is just a simple loop and can be done very efficiently.

11.4.3 Perturbations

There are some lighting situations where bidirectional mutations will almost always be rejected. This happens when there are small regions of the path space in which paths contribute much more than average. This can be caused by caustics, difficult visibility (e.g. a small hole), or by concave corners where two surfaces meet (a form of singularity in the integrand). The problem is that bidirectional mutations are relatively large, and so they usually attempt to mutate the path outside the high-contribution region.

One way to increase the acceptance probability is to use smaller mutations. The principle is that nearby paths will make similar contributions to the image, and so the acceptance probability will be high. Thus, rather than having many rejections, we can explore the other nearby paths that also have a high contribution.

Our solution is to choose a subpath of the current path, and move the vertices slightly. We call this type of mutation a *perturbation*. While the idea can be applied to arbitrary subpaths, our main interest is in perturbations that include the lens edge $\mathbf{x}_{k-1}\mathbf{x}_k$ (since other changes do not help to prevent long sample sequences at the same image point). We have implemented two specific kinds of perturbations that change the lens edge, termed *lens perturbations* and *caustic perturbations* (see Figure 11.4). These are described below.

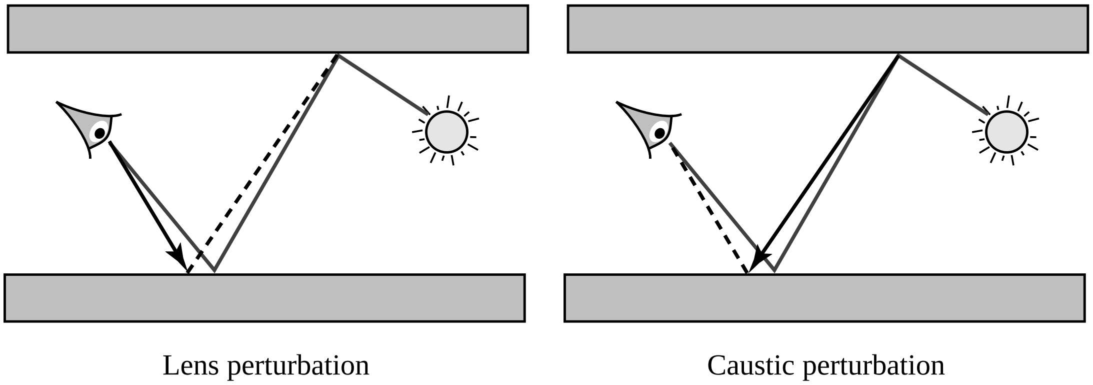


Figure 11.4: The lens edge can be perturbed by regenerating it from either side: we call these *lens perturbations* and *caustic perturbations*.

Lens perturbations. We delete a subpath $\mathbf{x}_m \ldots \mathbf{x}_k$ of the form $(L|D)DS^*E$ (where the symbols S, D, E, and L stand for specular, non-specular, lens, and light vertices respectively).[8] This is called the *lens subpath*, and consists of $k-m$ edges and $k-m-1$ vertices (the vertex $\mathbf{x}_m$ is not included). Note that we require both $\mathbf{x}_m$ and $\mathbf{x}_{m+1}$ to be non-specular, since otherwise any perturbation would result in a path $\bar{y}$ for which $f(\bar{y}) = 0$.

To replace the lens subpath, we perturb the image location of the old subpath by moving it a random distance R in a random direction ϕ on the image plane. The angle ϕ is chosen uniformly, while R is exponentially distributed between two values r_1 and r_2:

$$R \;=\; r_2 \, \exp(-\ln(r_2/r_1)\, U) \,, \tag{11.11}$$

where U is uniformly distributed on $[0, 1]$.

We then cast a ray at the new image location, and extend the subpath through additional specular bounces to be the same length as the original. The mode of scattering at each specular bounce is preserved (i.e. specular reflection or transmission), rather than making new random choices. (If the perturbation moves a vertex from a specular to a non-specular material, then the mutation is immediately rejected.) This allows us to efficiently sample rare

[8] This is Heckbert's regular expression notation, as described in Section 8.3.1. We have not used the full-path notation of Section 8.3.2, although we assume that the light source has type $L(S|D)D$ and the lens has type $D(S|D)E$ with respect to the classifications introduced there.

combinations of events, e.g. specular reflection from a surface where 99% of the light is transmitted. This is important when only some of these combinations contribute to the image: for example, consider a scene model containing a glass window, where the environment beyond the window is dark. In this case, only reflections from the window will contribute significantly to the image.

The calculation of $a(\bar{x} \to \bar{y})$ is similar to the bidirectional case. The main difference is the method used to select a sample point on the image plane (i.e. equation (11.11) is used, rather than choosing a point uniformly at random within the image region).

Caustic perturbations. Lens perturbations are not possible in some situations; the most notable example occurs when computing caustics. These paths have the form LS^+DE, which is not acceptable for lens perturbations.

Fortunately there is another way to perturb these paths, or in fact any path with a suffix $\mathbf{x}_m \ldots \mathbf{x}_k$ of the form $(D|L)S^*DE$ (see Figure 11.5). To do this, we generate a new subpath starting from the vertex $\mathbf{x}_m$. The direction of the segment $\mathbf{x}_m \to \mathbf{x}_{m+1}$ is perturbed by a random amount (θ, ϕ), where the $\theta = 0$ axis corresponds to the direction of the original ray. As before, the angle ϕ is chosen uniformly, while θ is exponentially distributed between two values θ_1 and θ_2:

$$\theta \;=\; \theta_2 \, \exp(-\ln(\theta_2/\theta_1) \, U) \,,$$

where U is uniformly distributed on $[0, 1]$. The technique is otherwise similar to lens perturbations, i.e. the new subpath is extended to the same length as the original, and the mode of scattering at each bounce is preserved.

Multi-chain perturbations. Neither of the above can handle paths with a suffix of the form $(D|L)DS^+DS^+E$, i.e. caustics seen through a specular surface. This can be handled by perturbing the path through more than one specular chain. A lens perturbation is used for the first chain DS^+E, and a new direction is chosen for the first edge of each subsequent chain DS^+D by perturbing the direction of the corresponding edge in the original subpath (using the same method described for caustic perturbations). Figure 11.6 shows an example of a situation where this technique is useful.

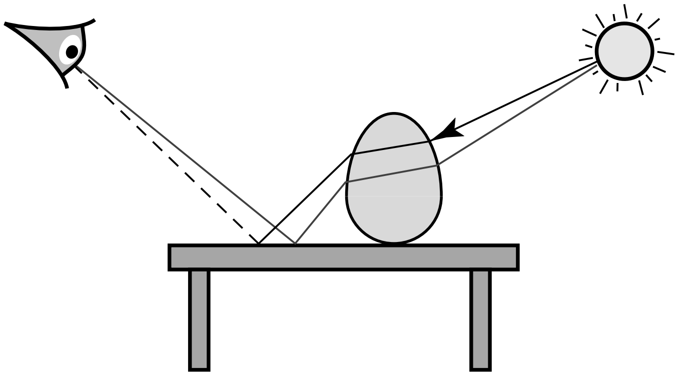


Figure 11.5: A caustic perturbation. A new path is generated by perturbing the direction of the ray from the light source by a small amount, and then tracing the perturbed ray through the same sequence of specular reflections and refractions as the original path.

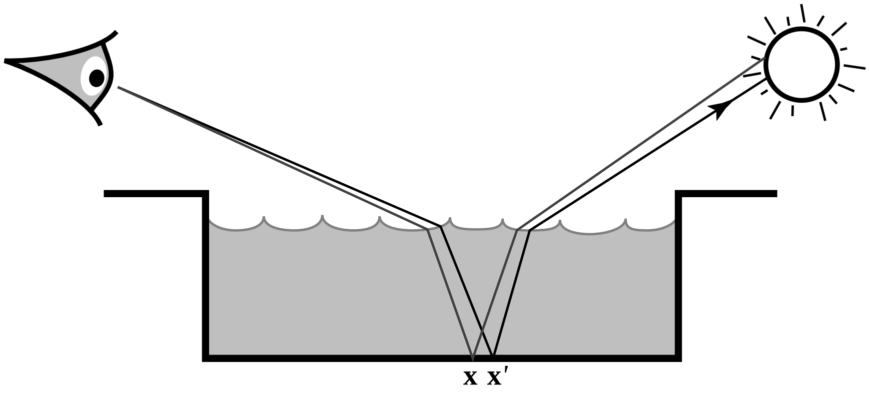


Figure 11.6: Using a two-chain perturbation to sample caustics in a pool of water. First, the lens edge is perturbed to generate a point $\mathbf{x}'$ on the pool bottom. Then, the direction from original point $\mathbf{x}$ toward the light source is perturbed, and a ray is cast from $\mathbf{x}'$ in this direction.

Parameter values. For lens perturbations, the image resolution is a guide to the useful range of values. We use a minimum perturbation size of $r_1 = 0.1$ pixels, while r_2 is chosen such that the perturbation region is 5% of the image area. For caustic perturbations, we also make use of the image resolution. Specifically, the maximum perturbation angle is defined as

$$\theta_2 \;=\; \theta(r_2)\,\frac{\|\mathbf{x}_k - \mathbf{x}_{k-1}\|}{\sum_{i=m+1}^{k-1} \|\mathbf{x}_i - \mathbf{x}_{i-1}\|}\,,$$

where $\mathbf{x}_m \ldots \mathbf{x}_k$ is the perturbed subpath, and $\theta(r)$ is the angle through which the ray $\mathbf{x}_k \to \mathbf{x}_{k-1}$ needs to be perturbed to change the image location by a distance of r pixels. A similar rule defines θ_1 in terms of r_1. The purpose of these formulas is to ensure that caustic perturbations change the image location by an amount that is similar to that used for lens perturbations.

Finally, for multi-chain perturbations, we use $\theta_1 = 0.0001$ radians and $\theta_2 = 0.1$ radians. The image resolution cannot be used as a guide here, so the range of useful perturbation values is larger. Note that in our experiments, we have not found the MLT algorithm to be particularly sensitive to any of these values.

11.4.4 Lens subpath mutations

We now describe *lens subpath mutations*, whose goal is to stratify the samples over the image plane, and also to reduce the cost of sampling by re-using subpaths. Each mutation consists of deleting the lens subpath of the current path, and replacing it with a new one. (As before, the lens subpath has the form $(L|D)S^*E$.) The lens subpaths are stratified across the image plane, such that every pixel receives the same number of proposed lens subpath mutations.

We briefly describe one way to do this. We initialize the algorithm with n' independent seed paths (Section 11.3), which are mutated in a rotating sequence. At all times, we also store a current lens subpath $\bar{x}_\mathrm{e}$. A lens subpath mutation consists of deleting the lens subpath of the current path $\bar{x}$, and replacing it with $\bar{x}_\mathrm{e}$. This happens whenever a lens subpath mutation is selected for the current path (as opposed to a perturbation or bidirectional mutation). After the lens subpath $\bar{x}_\mathrm{e}$ has been re-used a fixed number of times n_e, it is discarded and a new one is generated. We chose $n' \gg n_\mathrm{e}$, to prevent the same lens subpath from being used more than once on the same path.

Each lens subpath $\bar{x}_{\mathrm{e}}$ is generated by casting a ray through a random point on the image plane, and following zero or more specular bounces until a non-specular vertex is found. (At a material with specular and non-specular components, we randomly choose between them.) To stratify the samples on the image plane, we maintain a tally of the number of lens subpaths that have been generated at each pixel. When generating a new subpath, we choose a random pixel and increment its tally. If that pixel already has its quota of lens subpaths, we search for a non-full pixel using the concept of a *rover* (named after a similar idea in certain memory management schemes). The rover is simply an index into a pseudo-random ordering of the image pixels, such that every pixel appears exactly once.[9] If the randomly chosen pixel from the first step is full, we check the pixel corresponding to the rover, and if necessary we visit additional pixels in pseudo-random order until a non-full one is found. Note that we also control the distribution of samples within each pixel, by computing a Poisson minimum-disc pattern and tiling it over the image plane.

The acceptance probability $a(\bar{x} \to \bar{y})$ is computed in a similar way to the bidirectional case, except that the new subpath can be generated in only one way. (Subpath re-use does not influence the calculation.)

11.4.5 Selecting between mutation types

At each step, we assign a probability to each of the three mutation types. This discrete distribution is sampled to determine which kind of mutation is applied to the current path.

We have found that it is important to make the probabilities relatively balanced. This is because the mutation types are designed to satisfy different goals, and it is difficult to predict in advance which types will be the most successful. The overall goal is to make mutations that are as large as possible, while still having a reasonable chance of acceptance. This can be achieved by randomly choosing between mutations of different sizes, so that there is a good chance of trying an appropriate mutation for any given path.

These observation are similar to those of multiple importance sampling (Chapter 9). We would like a set of mutations that cover all the possibilities, even though we may not (and

[9] The low-order bits of a linear congruential generator can be used for this purpose.

need not) know the optimum way to choose among them for a given path. It is perfectly fine to include mutations that are designed for special situations, and that result in rejections most of the time. This increases the cost of sampling by only a small amount, and yet it can increase robustness considerably.

11.5 Refinements

This section describes a number of general techniques that improve the efficiency of MLT.

Direct lighting. We use standard techniques for direct lighting (e.g. see Shirley et al. [1996]), rather than the Metropolis algorithm. In most cases, these standard methods give better results at lower cost, since the Metropolis samples are not as well-stratified across the image plane (Section 11.4.1). By excluding direct lighting paths from the Metropolis calculation, we can apply more effort to the indirect lighting.

This optimization is easy to implement; it can be done as part of the lens subpath mutation strategy, which already generates a fixed number of subpaths at each pixel. To compute the direct lighting, we perform a standard ray tracing calculation as each lens subpath is generated (independent of the current MLT path). These contributions are accumulated in the same image as the Metropolis samples.[10] We also need to remove the direct lighting paths from the Metropolis portion of the algorithm, but this is easy: when a mutation generates a direct lighting path, we simply reject it. An even better approach is to modify the mutation strategies themselves, in order to avoid generating these paths in the first place.

Finally, note that if the lighting is especially difficult (e.g. due to visibility), then the direct lighting "optimization" may be a disadvantage. For example, imagine a large building with many rooms and lights, but where only one room is visible. Unless the direct lighting strategy does a good job of excluding all the unimportant lights, then MLT can be substantially more efficient.

[10] To do this, we must know in advance how many direct lighting samples there will be at each pixel; adaptive sampling of the image plane is not allowed.

Use of expected values. For each proposed mutation, there is a probability $a(\bar{x} \to \bar{y})$ of accumulating an image sample at $\bar{y}$, and a probability $1 - a(\bar{x} \to \bar{y})$ of accumulating a sample at $\bar{x}$. We can make this more efficient by always accumulating a sample at both locations, weighted by the corresponding probability. Effectively, this optimization replaces a random variable by its expected value (see [Kalos & Whitlock 1986, p. 105]). This is especially useful for sampling the dim regions of the image, which would otherwise receive very few samples. Note that this optimization does not affect the random walk itself; each transition is accepted or rejected in the same way as before.

Two-stage MLT. For images with large brightness variations, the MLT algorithm can spend most of its time sampling the brightest regions. This is undesirable, since it means that brighter pixels are estimated with a higher relative accuracy. Specifically, the variance of pixel j is proportional to I_j, the standard error is proportional to $\sqrt{I_j}$, and the relative error is proportional to $1/\sqrt{I_j}$. As a first approximation, it would be better for the relative errors at all the pixels to be the same (because the human eye is sensitive to contrast differences). To achieve this, we would like an algorithm that generates approximately the same number of samples at every pixel (with a sample value that varies according to the brightness of the ideal image).

The MLT algorithm can easily be modified to approach this goal, by precomputing a test image I_0 at a low sampling density. Then rather than sampling according to the image contribution function f, we sample according to

$$f'(\bar{x}) \;=\; f(\bar{x}) \,/\, I_0(\bar{x})\,, \tag{11.12}$$

where $I_0(\bar{x})$ depends only on the image location of $\bar{x}$. This function f' is used instead of f everywhere in the MLT algorithm, including the computation of the paths weights W_0 during initialization. To compensate for this, each MLT sample value is multiplied by $I_0(\bar{x})$ just before it is accumulated in the image.

The end result is that the MLT sample values are no longer constant across the image; instead, they vary according to the test image I_0. This does not introduce any bias; it simply means that the bright parts of the image are estimated using a smaller number of samples with larger values.[11]

This optimization is mainly useful for images where the range of intensities is very large. Note that the brightest regions of an image are often light sources or directly lit surfaces, in which case handling the direct lighting separately will solve most of the problem.

Importance sampling for mutation probabilities. We describe a technique that can increase the efficiency of MLT substantially, by increasing the average acceptance probability $a(\bar{x} \to \bar{y})$. The idea is to implement a form of importance sampling with respect to $a(\bar{x} \to \bar{y})$ when deciding which mutation to attempt, by weighting each possible mutation according to the probability with which the deleted subpath can be regenerated. (This is the factor $p_{\mathrm{d},2}$ mentioned in Section 11.4.2.)

Let $\bar{x} = \mathbf{x}_0 \ldots \mathbf{x}_k$ be the current path, and consider a mutation that deletes the subpath $\mathbf{x}_l \ldots \mathbf{x}_m$. The insight is that given only the deleted subpath, it is already possible to compute some of the factors in the acceptance probability $a(\bar{x} \to \bar{y})$. In particular, from equation (11.8) we see that $a(\bar{x} \to \bar{y})$ is proportional to

$$Q(\bar{y} \to \bar{x}) \ = \ 1 \,/\, R(\bar{y} \to \bar{x}) \,,$$

and from equation (11.10) we see that given only the path $\bar{x}$, it is possible to compute all the components of $Q(\bar{y} \to \bar{x})$ except for the discrete probabilities p_{d} and p_{a}. (These probabilities depend on the path $\bar{y}$, which has not been generated yet). If we simply set these unknown quantities to one, we obtain

$$p_{\mathrm{d},2} \ = \ \sum_{i=1}^{k_{\mathrm{a}}} (1/C_i^{\mathrm{bd}}) \,, \tag{11.13}$$

where i refers to the i-th edge of the deleted subpath $\mathbf{x}_l \ldots \mathbf{x}_m$, and C_i^{bd} is the unweighted contribution defined below equation (11.10).

This quantity is proportional to a subset of the factors in the acceptance probability $a(\bar{x} \to \bar{y})$. Thus by weighting the discrete probabilities for each mutation type by this factor, we can avoid mutations that are unlikely to be accepted. With bidirectional mutations,

[11] Note that if not enough samples are used to create the test image, then some pixels will be zero (which is not allowed by the estimate (11.12)). This problem can be solved by filtering the test image before it is used. The simplest approach is to extract the brightest parts of the test image, and weight the other pixels uniformly.

for example, this factor is applied to each of the $O(k^2)$ possibilities for the deleted subpath $\mathbf{x}_l \ldots \mathbf{x}_m$. The computation can be made more efficient by approximating $p_{\mathrm{d},2}$ even further. For example, equation (11.13) can be evaluated for many mutations in parallel by replacing the sum of the $1/C_i^{\mathrm{bd}}$ by their maximum.

11.6 Results

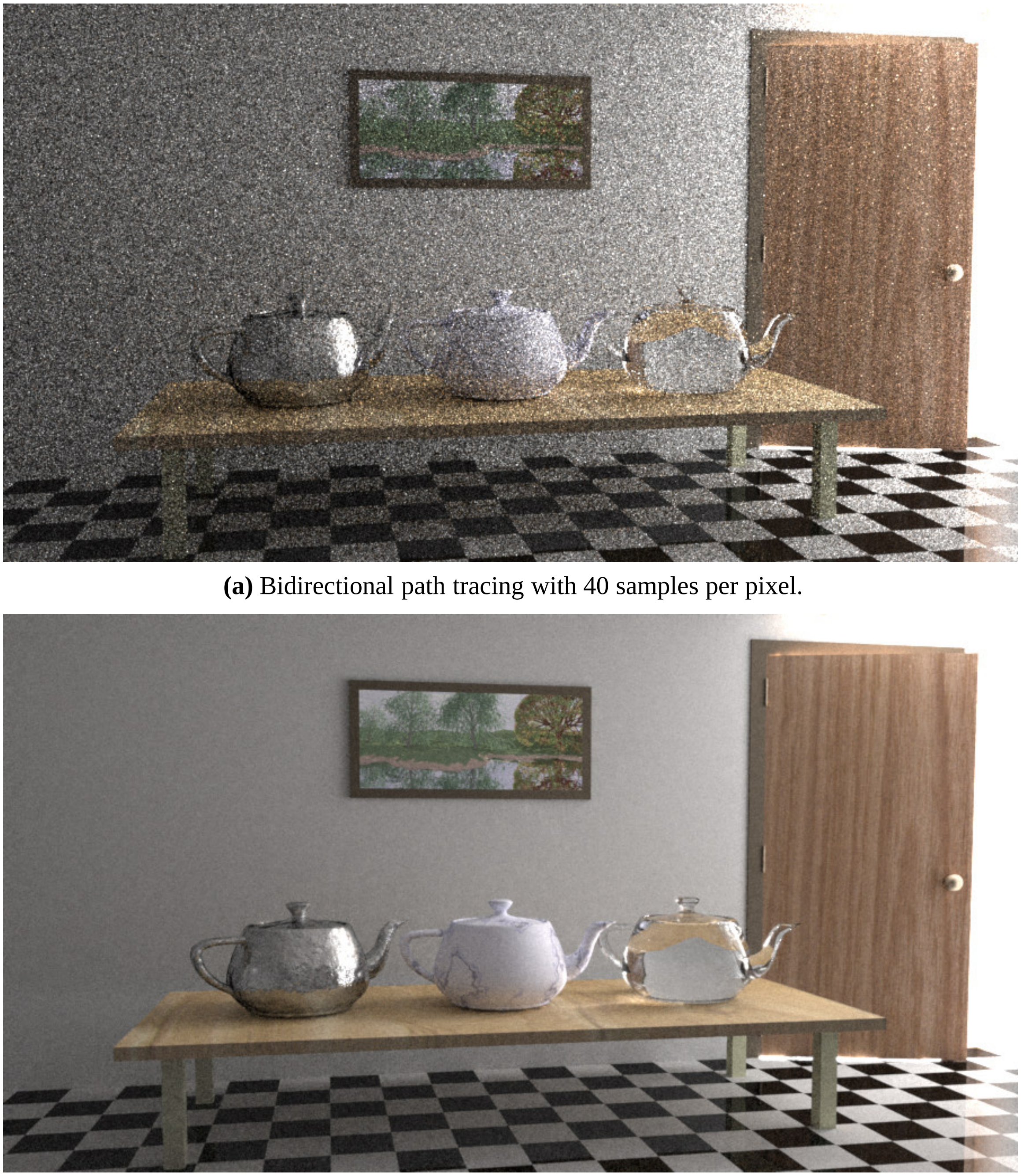
We have rendered test images that compare Metropolis light transport with classical and bidirectional path tracing. Our path tracing implementations support efficient direct lighting calculations, importance-sampled BSDF's, Russian roulette on shadow rays, and several other optimizations.

Figure 11.7 shows a test scene with difficult indirect lighting. All of the light in this scene comes through a slightly open doorway, which lets through about 0.1% of the light in the adjacent room. The light source is a diffuse ceiling panel at the far end of that room (which is quite large), so that most of the light coming through the doorway has already bounced several times.

For equal computation times, Metropolis light transport gives far better results than bidirectional path tracing. Notice the details that would be difficult to obtain with many light transport algorithms: contact shadows, caustics under the glass teapot, light reflected by the white tiles under the door, and the brighter strip along the back of the floor (due to the narrow gap between the table and the wall). This scene contains diffuse, glossy, and specular surfaces, and the wall is untextured to clearly reveal the noise levels.

For this scene, MLT gains efficiency from its ability to change only part of the current path. The portion of the path through the doorway can be preserved and re-used for many mutations, until it is successfully mutated into a different path through the doorway. Note that perturbations are not essential to make this process efficient, since the path through the doorway needs to change only infrequently.

Figure 11.8 compares MLT against bidirectional path tracing for a scene with strong indirect illumination and caustics. Both methods give similar results in the top row of images (where indirect lighting from the floor lamp dominates). However, MLT performs much better as we zoom into the caustic, due to its ability to generate new paths by perturbing



(a) Bidirectional path tracing with 40 samples per pixel.

(b) Metropolis light transport with 250 mutations per pixel [the same computation time as (a)].

Figure 11.7: All of the light in this scene comes through a slightly open doorway, which lets through about 0.1% of the light in the adjacent room. The MLT algorithm is able to generate paths efficiently by always preserving a path segment that goes through the small opening between the rooms. The images are 900 by 500 pixels, and include paths up to length 10.

existing paths. The image quality degrades with magnification (for the same computation time), but only slowly. This is due to the fact that the average mutation cost goes up as we zoom into the caustic (since each successful perturbation requires at least four ray-casting operations). Once the caustic fills the entire image, the image quality remains virtually constant.[12]

Notice the streaky appearance of the noise at the highest magnification. This is due to caustic perturbations: each ray from the spotlight is perturbed within a narrow cone; however, the lens maps this cone of directions into an elongated shape. The streaks are due to long strings of caustic mutations that were not broken by successful mutations of some other kind.

Even in the top row of images, there are slight differences between the two methods. The MLT algorithm leads to lower noise in the bright regions of the image, while the bidirectional algorithm gives lower noise in the dim regions. This is what we would expect, since the number of Metropolis samples varies according to the pixel brightness, while the number of bidirectional samples per pixel is constant.

Figure 11.9 shows another difficult lighting situation: caustics on the bottom of a small pool, seen indirectly through the ripples on the water surface. Path tracing does not work well in this case, because when a path strikes the bottom of the pool, a reflected direction is sampled according to the BRDF. Only a very small number of these paths contribute to the image, because the light source occupies about 1% of the hemisphere of directions above the pool.[13] (Bidirectional path tracing does not help for these paths, because they can be generated only starting from the eye.) As in the previous example, perturbations are the key to sampling these caustics efficiently. However, for this scene it is multi-chain rather than caustic perturbations that are important (recall Figure 11.6). One interesting feature of MLT is that it obtains these results without special handling of the light sources or specular surfaces — see Mitchell & Hanrahan [1992] or Collins [1995] for good examples of what

[12] Note that the according to the rules for caustic perturbations described in Section 11.4.3, the average perturbation angle decreases with linearly with the magnification. This implies that the average perturbation size is constant when measured in image pixels.

[13] Note that the brightness of the caustic is proportional to the solid angle occupied by the light source, as seen from the bottom of the pool. Thus in regions where the caustics are dim, the chance of a ray hitting the light source is actually much less than one percent.

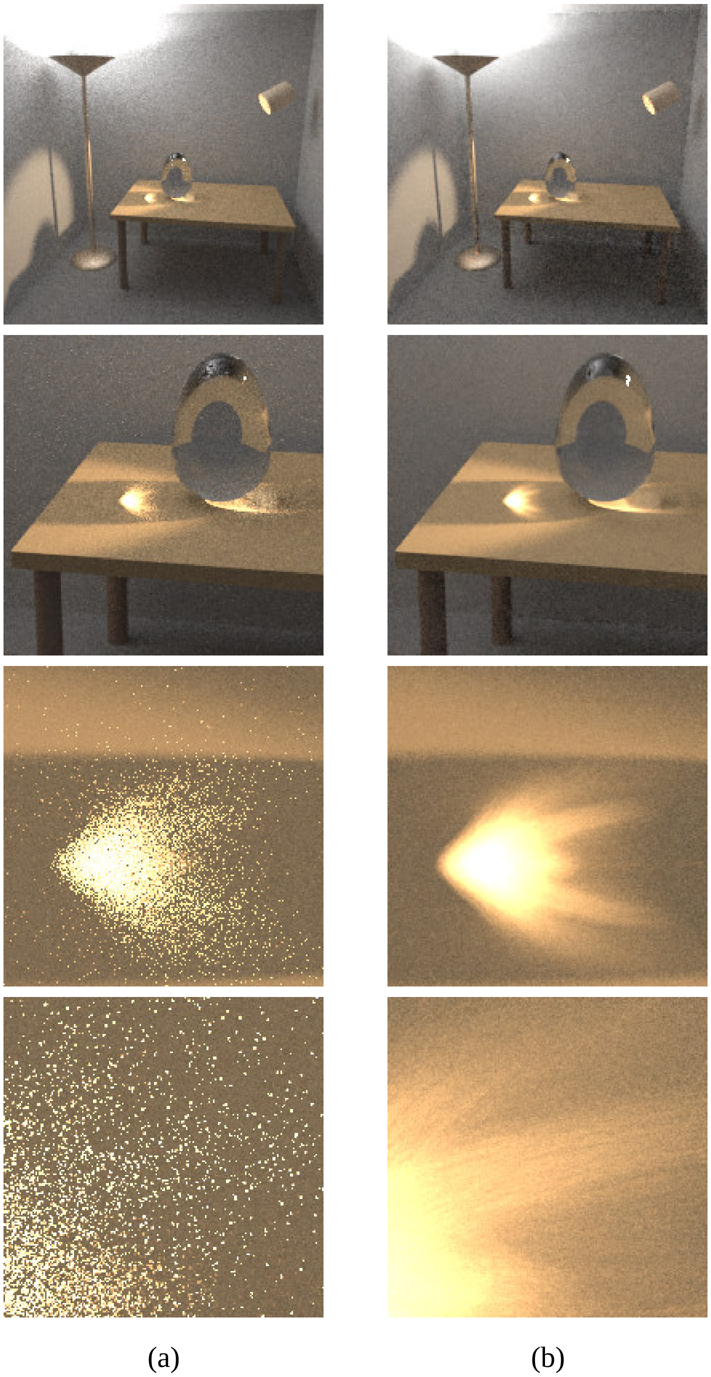
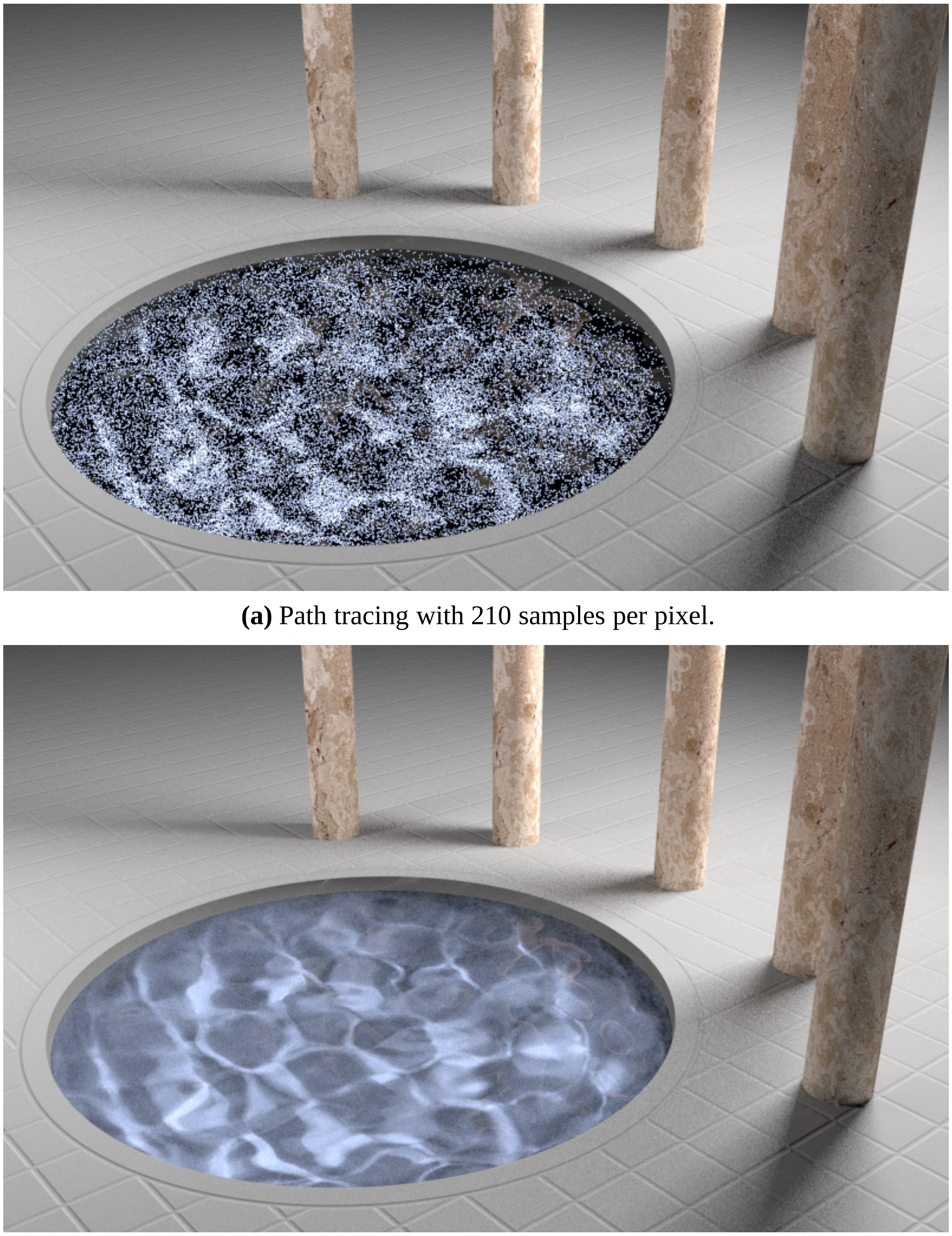


Figure 11.8: These images show caustics formed by a spotlight shining on a glass egg. Column (a) was computed using bidirectional path tracing with 25 samples per pixel, while (b) uses Metropolis light transport with the same number of ray queries (varying between 120 and 200 mutations per pixel). The solutions include paths up to length 7, and the images are 200 by 200 pixels.



(a) Path tracing with 210 samples per pixel.

(b) Metropolis light transport with 100 mutations per pixel [the same computation time as (a)].

Figure 11.9: Caustics in a pool of water, viewed indirectly through the ripples on the surface. It is difficult for unbiased Monte Carlo algorithms to find the important transport paths, since they must be generated starting from the lens, and the light source only occupies about 1% of the hemisphere as seen from the pool bottom (which is curved). The MLT algorithm samples these paths efficiently by means of perturbations. The images are 800 by 500 pixels.

Test Case	PT vs. MLT			BPT vs. MLT		
	l_1	l_2	l_∞	l_1	l_2	l_∞
Figure 11.7 (door)	7.7	11.7	40.0	5.2	4.9	13.2
Figure 11.8 (egg, top image)	2.4	4.8	21.4	0.9	2.1	13.7
Figure 11.9 (pool)	3.2	4.7	5.0	4.2	6.5	6.1

Table 11.1: This table shows numerical error measurements for path tracing (PT) and bidirectional path tracing (BPT) relative to Metropolis light transport (MLT), for the same computation time. The entries in the table were determined as follows. For each test image, we computed the relative error $e_j = (\tilde{I}_j - I_j)/I_j$ at each pixel, where $\tilde{I}_j$ corresponds to the algorithm being measured, and I_j is the value from a reference solution. Next, we computed the l_1, l_2, and l_∞ norms of the resulting array of errors e_j. Finally, we divided the error norms for path tracing and bidirectional path tracing by the corresponding error norm for MLT, to obtain the normalized results shown in the table above. Note that the gain in efficiency of MLT over the other algorithms is proportional to the square of the table entries.

can be achieved if this restriction is lifted.

We have also made numerical measurements in order to compare the performance of the various algorithms on each test scene. To do this, we first computed images using path tracing (PT), bidirectional path tracing (BPT), and Metropolis light transport (MLT), with the same computation time in each case. Next, we computed the relative error $e_j = (\tilde{I}_j - I_j)/I_j$ at each pixel, where $\tilde{I}_j$ corresponds to the algorithm being measured, and I_j is the value from a reference solution (created using bidirectional path tracing with a large number of samples, at a lower image resolution). We then computed the l_1, l_2, and l_∞ norms of the resulting array of errors e_j, and divided the error norms for PT and BPT by the corresponding error norm for MLT. This yielded the results shown in Table 11.1.

Note that the efficiency gain of MLT over the other methods is proportional to the *square* of the table entries, since the error obtained using path tracing and bidirectional path tracing decreases according to the square root of the number of samples. For example, the RMS relative error in the three-teapots image of Figure 11.7(a) is 4.9 times higher than in Figure 11.7(b), which implies that approximately 25 times more bidirectional path tracing samples would be required to achieve the same error levels as MLT. Even in the topmost images of Figure 11.8 (for which bidirectional path tracing is well-suited), notice that the results of

MLT are competitive.

For comparison, we consider the techniques proposed by Jensen [1995] and Lafortune & Willems [1995a] for sampling difficult paths more efficiently. Basically, their idea is to build an approximate representation of the radiance in a scene, and use it to modify the directional sampling of the basic path tracing algorithm. The radiance information can be collected either with a particle tracing prepass [Jensen 1995], or by adaptively recording it in a spatial subdivision as the algorithm proceeds [Lafortune & Willems 1995a]. However, these techniques have several problems, including insufficient directional resolution to be able to sample concentrated indirect lighting efficiently, and substantial space and time requirements. In any case, the best variance reductions that have been reported are in the range of 50% to 70% (relative to standard path tracing), as opposed to the reductions of 96% to 99% reported in Table 11.1. (Similar ideas have also been applied to particle tracing algorithms [Pattanaik & Mudur 1995, Dutre & Willems 1995], with similar results.)

In our tests, the computation times were approximately 4 hours for the each image in Figure 11.7 (the door ajar), 15 minutes for the images in Figure 11.8 (the glass egg), and 2.5 hours for the images in Figure 11.9 (the pool), where all times were measured on a 190 MHz MIPS R10000 processor. The memory requirements are modest: we only store the scene model, the current image, and a single path (or a small number of paths, if the mutation technique in Section 11.4.4 is used). For high-resolution images, memory usage could be reduced further by collecting the samples in batches, sorting them in scanline order, and applying them to an image on disk.

11.7 Conclusions

We have presented a novel approach to global illumination problems, by showing how to adapt the Metropolis sampling method to light transport. Our algorithm starts from a few seed light transport paths and applies a sequence of random mutations to them. In the steady state, the resulting Markov chain visits each path with a probability proportional to that path's contribution to the image. The MLT algorithm is notable for its generality and simplicity. A single control structure can be used with different mutation strategies to handle a variety of difficult lighting situations. In addition, the MLT algorithm needs little memory, and always computes an unbiased result.

The MLT algorithm offers interesting new possibilities for adaptive sampling without bias, since the mutation strategy is allowed to depend on the current path. For example, consider the strategy of replacing the light source vertex $\mathbf{x}_0$ with a new randomly sampled position on the same light source. This is potentially a simple, effective strategy for handling scenes with many lights: once an important light source is found, the MLT algorithm can efficiently generate many samples from it. (More generally, mutations could be proposed to nearby light sources by constructing a spatial subdivision.) This is clearly a form of adaptive sampling, since more samples are taken in regions nearby existing good samples. Unlike with standard Monte Carlo algorithms, however, no bias is introduced.

This also raises interesting possibilities for handling specular surfaces. For example, we could try a strategy similar to that above: when mutating a subpath containing a specular vertex, generate a new vertex on the same specular object. If only a small fraction of the specular surfaces in the scene made a large contribution to the image, this would provide a means of sampling them efficiently. Note that this technique is more powerful than simply flagging specular surfaces for extra sampling, since we do not need to assign an *a priori* probability to the sampling of each surface. This is important when a large number of specular surfaces are present, since in the MLT case the sampling efficiency is not affected once an important surface has been found.

The MLT framework could also be an advantage for techniques that generate specular vertices deterministically. In particular, recall the idea of generating a chain of specular vertices connecting two given points (as mentioned in Section 8.3.4). A simple example is that given two points $\mathbf{x}_1$ and $\mathbf{x}_3$ and a planar mirror, we might calculate the point $\mathbf{x}_2$ on the mirror that reflects light between them. (Note that it is also possible to handle non-planar surfaces, or sequences of such surfaces, using techniques described by Mitchell & Hanrahan [1992].) However, these analytic techniques have problems when there are many specular surfaces, since each possible surface and sequence of surfaces must be checked separately for a solution.

The MLT framework helps to solve the combinatorial aspect of this problem. Once an important specular chain is found, a new chain could be generated by simply perturbing one of its endpoints, and then regenerating the intermediate vertices using the same sequence of specular surfaces. For example, this could be used to efficiently sample caustics seen indirectly through specular reflectors, even a point light source is used, and when there are possibly many specular surfaces in the scene. On the other hand, recall that we cannot hope to solve all such problems efficiently, since provably difficult configurations of mirrors do exist [Reif et al. 1994].

The MLT algorithm can also be extended in other ways. For example, with modest changes we could use it to compute view-independent radiance solutions, by letting the I_j be the basis function coefficients, and defining $f(\bar{x}) = \sum_j f_j(\bar{x})$. We could also use MLT to render a sequences of images (as in animation), by sampling the entire space-time of paths at once (thus, a mutation might try to perturb a path forward or backward in time). Another interesting problem is to determine the optimal settings for the various parameters used by the algorithm. The values we use have not been extensively tuned, so that further efficiency improvements may be possible. Genetic algorithms may be useful in this regard, to optimize the parameter settings on a suite of test images. We hope to address some of these refinements and extensions in the future.

Appendix 11.A Proof of Unbiased Initialization

In this appendix, we show that the estimate

$$I_j \;=\; E\left[\frac{1}{N}\sum_{i=1}^{N} W_i\, h_j(\bar{X}_i)\right]$$

is unbiased (see Section 11.3.1). To do this, we show that the following*weighted equilibrium condition* is satisfied at each step of the random walk:

$$\int_{\mathbb{R}} w\, p_i(w,\bar{x})\, dw \;=\; f(\bar{x})\,, \qquad (11.14)$$

where p_i is the joint density function of the$i$-th weighted sample $(W_i, \bar{X}_i)$. This is a sufficient condition for the above estimate to be unbiased, since

$$\begin{aligned} E\Big[W_i\, h_j(\bar{X}_i)\Big] \;&=\; \int_{\Omega}\int_{\mathbb{R}} w\, h_j(\bar{x})\, p_i(w,\bar{x})\, dw\, d\mu(\bar{x}) \\ &=\; \int_{\Omega} h_j(\bar{x})\, f(\bar{x})\, d\mu(\bar{x}) \\ &=\; I_j\,. \end{aligned}$$

To show that the weighted equilibrium condition holds for all samples$(W_i, \bar{X}_i)$, we proceed by induction. For $i = 0$, we have

$$p_0(w,\bar{x}) \;=\; \delta(w - \frac{f(\bar{x})}{p_0(\bar{x})})\, p_0(\bar{x})\,,$$

where $\delta(w - w_0)$ is a Dirac distribution, corresponding to the fact thatW_0 is chosen as a deterministic function of $\bar{X}_0$ rather than by random sampling. It is easy to verify thatp_0 satisfies condition (11.14).

Next we verify that the Metropolis algorithm preserves the weighted equilibrium condition from one sample to the next. Since the mutations set$W_i = W_{i-1}$, the first part of equation (11.4) is still true when $p_j(\bar{x})$ is replaced by $p_j(w,\bar{x})$:

$$\begin{aligned} p_i(w,\bar{x}) \;&=\; p_{i-1}(w,\bar{x}) \,+\, \int_{\Omega}\Big\{ p_{i-1}(w,\bar{x})\, T(\bar{x}\!\to\!\bar{y})\, a(\bar{x}\!\to\!\bar{y}) \\ &\qquad\qquad -\, p_{i-1}(w,\bar{y})\, T(\bar{y}\!\to\!\bar{x})\, a(\bar{y}\!\to\!\bar{x})\Big\}\, d\mu(\bar{y})\,. \end{aligned}$$

Multiplying both sides by w and integrating, we obtain

$$\begin{aligned} \textstyle\int_{\mathbb{R}} w\, p_i(w,\bar{x})\, dw \;&=\; \textstyle\int_{\mathbb{R}} w\, p_{i-1}(w,\bar{x})\, dw \,+\, \displaystyle\int_{\Omega}\Big\{ [\textstyle\int_{\mathbb{R}} w\, p_{i-1}(w,\bar{x})\, dw]\; T(\bar{x}\!\to\!\bar{y})\, a(\bar{x}\!\to\!\bar{y}) \\ &\qquad -\, [\textstyle\int_{\mathbb{R}} w\, p_{i-1}(w,\bar{y})\, dw]\; T(\bar{y}\!\to\!\bar{x})\, a(\bar{y}\!\to\!\bar{x})\Big\}\, d\mu(\bar{y}) \end{aligned}$$

$$
\begin{aligned}
&= \textstyle\int_{\mathbb{R}} w \, p_{i-1}(w, \bar{x}) \, dw \;+\; \int_{\Omega} \Big\{ f(\bar{x}) \, T(\bar{x} \!\to\! \bar{y}) \, a(\bar{x} \!\to\! \bar{y}) \\
&\qquad\qquad\qquad\qquad\qquad\quad - f(\bar{y}) \, T(\bar{y} \!\to\! \bar{x}) \, a(\bar{y} \!\to\! \bar{x}) \Big\} \, d\mu(\bar{y}) \\
&= \textstyle\int_{\mathbb{R}} w \, p_{i-1}(w, \bar{x}) \, dw \\
&= f(\bar{x}) \, ,
\end{aligned}
$$

where we have used the detailed balance condition (11.3). Thus every mutation step preserves the weighted equilibrium condition (11.14). ■

It is interesting to note that even though the random walk is always in weighted equilibrium, the distributions of paths and weights change at each step. In particular, the path distribution is initially given by some arbitrary density function $p_0(\bar{x})$, and converges toward the stationary distribution $p^*(\bar{x})$. Similarly, the weight distribution $p_i(w \mid \bar{x})$ at a given point $\bar{x}$ starts out as a Dirac distribution

$$p_0(w \,|\, \bar{x}) \;=\; \delta(w - \frac{f(\bar{x})}{p_0(\bar{x})}) \, ,$$

and gradually evolves toward an equilibrium $p^*(w \mid \bar{x})$. Furthermore this equilibrium does not depend on $\bar{x}$, since

$$p^*(w \,|\, \bar{x}) \, p^*(\bar{x}) \;\equiv\; p^*(\bar{x} \,|\, w) \, p^*(w) \, ,$$

and the density functions $p^*(\bar{x} \,|\, w)$ and $p^*(\bar{x})$ are equal (i.e. the paths at each weight evolve toward the same equilibrium, since the transition rules do not depend on weight). Thus we have

$$p^*(w \,|\, \bar{x}) \;=\; p^*(w) \;=\; p_0(w) \, ,$$

observing that the marginal weight density $p_0(w)$ does not change with time (recall that $W_i = W_{i-1}$).

The net effect is that the path and weight distributions may start far from equilibrium, and gradually converge toward it. However, this is done in such a way that the weighted equilibrium condition (11.14) is initially satisfied, and preserved at every step. Thus we can obtain unbiased results immediately, rather than waiting for the path and weight distributions to converge separately.

Chapter 12

Conclusion

We conclude by summarizing the main results of this dissertation, in somewhat more detail than they were described in Chapter 1.

12.1 Bidirectional light transport theory

In the first part of this thesis, we have investigated the theoretical basis of bidirectional light transport algorithms. We proposed two different linear operator formulations of light transport, based on different sets of assumptions. First we considered the general case, where no assumptions are made about the physical validity of the scattering models used. In this case, we cannot rely on the properties of light transport in the real world: for example, energy might not be conserved. Nevertheless, there is still a well-defined mathematical problem to be solved (with mild restrictions discussed in Chapter 4), and we describe the manipulations that are necessary to ensure that algorithms based on radiance transport, importance transport, light particles, and importance particles all converge to the same mathematically correct solution. We have given a detailed analysis of the framework, including the norms, inverses, and adjoints of the various transport operators. We have also given explicit rules for handling all the various combinations of incident and exitant quantities.

We have shown that the above model is useful whenever the scene contains materials whose bidirectional scattering distribution function (BSDF) is not symmetric. There are two distinct situations where this can arise. First, some scattering models in computer graphics are not physically valid. It is important to be able to handle non-physical materials correctly, since they are sometimes very convenient. As a particular example we consider the use of shading normals, which are commonly applied to make polygonal surfaces look smooth or to add detail to coarse geometric models. We show that shading normals modify BSDF's and make them non-symmetric. However, using the linear operator formulation above we show that it is still possible to handle shading normals correctly and consistently in bidirectional algorithms, by using the correct adjoint BSDF (which we derive).

We also show that non-symmetric BSDF's can arise even for materials that are physically valid. This occurs whenever light is transmitted between two substances with different indices of refraction. Again we show how to handle this situation correctly within the general framework above, by deriving the adjoint BSDF for refraction. The use of this adjoint BSDF is necessary to ensure that bidirectional algorithms will converge to correct results.

However, when all materials in the scene model are physically valid, we have shown that there is a much better way to formulate the light transport problem. This formulation is based on a new reciprocity principle that holds for materials that transmit as well as reflect light. In particular, for physically valid materials we have shown that it is not the BSDF $f_{\mathrm{s}}(\omega_{\mathrm{i}} \to \omega_{\mathrm{o}})$ that is symmetric, but instead the quantity $f_{\mathrm{s}}(\omega_{\mathrm{i}} \to \omega_{\mathrm{o}}) / \eta_{\mathrm{o}}^2$ (where η_{o} is the refractive index of the medium containing ω_{o}). We establish this principle using the laws of thermodynamics, in particular Kirchhoff's laws and the principle of detailed balance. These laws hold for systems in thermodynamic equilibrium, but the resulting reciprocity principle is valid generally. We have investigated the historical origins of such principles, including Helmholtz and Rayleigh reciprocity, and clarified the important point that Helmholtz himself did not make any statement that would imply the symmetry of BRDF's. We have also discussed the subtle issues that arise in justifying such principles: the roles of thermodynamic equilibrium, time reversal invariance, and detailed balance. Finally, we have considered the conditions under which reciprocity does not hold, i.e. in the presence of absorbing media or external magnetic fields.

Taking advantage of this reciprocity principle, we have proposed a new light transport model where the transport operators are symmetric (self-adjoint) for any physically valid scene model. This symmetry simplifies both the theory of light transport algorithms (by eliminating the need for adjoint operators), and also their implementation (since the same transport rules apply to light and importance, or to path tracing and particle tracing). Furthermore, the transport quantities in the new model are optical invariants, which creates interesting connections to classical geometric optics. The modifications relative to the previous formulation are straightforward, and simply involve scaling the various transport quantities by the square of the refractive index of the surrounding medium. We have also provided a detailed analysis of the norms, inverses, and adjoints of the new operators.

We have proposed a third theoretical model for light transport, where each measurement is expressed as an integral over a space of paths (rather than as the solution to an integral equation or linear operator equation). The main advantage of this approach is its simple abstract form: by reducing light transport to a set of integrals, it allows general-purpose integration and sampling techniques to be applied (such as multiple importance sampling, or the Metropolis method). It is also useful from a conceptual point of view, since this formulation makes it clear that paths can be sampled in virtually any way desired, not just by recursively sampling a transport equation. We have described a variety of natural measures on paths with well-defined physical meanings, and we have developed an extended regular expression notation for paths that describes the properties of sources and sensors as well as the scattering properties at intermediate vertices. We have used this model to analyze the capabilities of unbiased Monte Carlo sampling algorithms, and we have shown that there are certain kinds of paths that cannot be generated by standard sampling techniques. This implies that certain lighting effects will be missing from the images generated using these techniques. We have analyzed the conditions under which this occurs, and we have proposed methods for making these path sampling algorithms complete.

12.2 General-purpose Monte Carlo techniques

The second area of this dissertation concerns new general-purpose techniques for Monte Carlo integration. Our main contribution in this area is *multiple importance sampling*, a method for combining several different sampling techniques for the same integral in order to obtain low-variance estimators for a broad class of integrands. We started by proposing a general model for combining samples from different techniques, called the *multi-sample* *model*. Using this model, we showed that any unbiased combination strategy could be represented as a set of weighting functions. This gave us a large space of possible combination strategies to explore, and a uniform way to represent them. We then proposed a specific combination strategy called the *balance heuristic*, and we proved that the variance obtained using this strategy is optimal to within a small additive term. We then proposed several other combination strategies, which are basically refinements of the balance heuristic: their variance is also provably close to optimal, but they give better results in certain important special cases.

We tested these methods on a variety of integration problems in computer graphics, and we found that multiple importance sampling can reduce variance substantially at little extra cost. The method is simple and practical to implement, and can make Monte Carlo calculations significantly more robust.

We have also proposed a new technique called *efficiency-optimized Russian roulette*. We started by showing that the variance of Russian roulette can be analyzed as a function of its threshold parameter (whose value is usually chosen in an *ad hoc* manner). We then described a technique for choosing the value of this parameter in order to maximize the efficiency of the resulting estimator. The main application of this technique in graphics is to reduce the number of visibility tests in rendering problems.

12.3 Robust light transport algorithms

We have shown how these theories and techniques can be applied to the construction of robust Monte Carlo light transport algorithms. The first algorithm we described was *bidirectional path tracing*, which is based on the path integral framework: it generates paths using a family of different importance sampling techniques, and then combines them using multiple importance sampling. Specifically, each path is constructed by concatenating two subpaths, one generated starting from a light source and another generated starting from the camera. We have shown that each such technique can efficiently sample a different set of paths, and that these paths are responsible for different lighting effects in the final image. By combining samples from all the techniques, we can efficiently render scenes under a wide variety of illumination conditions.

In addition to describing the mathematical basis of the method, we also discussed the implementation issues in detail. This includes how to sample and filter of the image, how to generate the paths and evaluate their contributions efficiently, and how to implement the important special cases where the light or eye subpath contains at most one vertex. We have also described the extensions required to handle ideal specular surfaces, and we have shown how efficiency-optimized Russian roulette can be used to reduce the number of visibility tests.

Bidirectional path tracing is unbiased, straightforward to implement, and supports the same range of geometry and materials as standard path tracing. It is an effective rendering algorithm for many kinds of indoor scenes, and is particularly useful for scene models with concentrated indirect lighting. On the other hand, the main weakness of the algorithm is that the light and eye subpaths are generated independently. This makes it unsuitable for outdoor environments, or scenes with many light sources, or scenes with constricted geometry between the light sources and the viewer.

Finally, we have introduced a new algorithm called *Metropolis light transport*. This method is also based on the path integral framework, but it samples paths in a different way. Specifically, it uses the Metropolis sampling algorithm, which generates a sequence of paths by following a random walk through path space. Each path is generated from the previous one by proposing a random mutation. This mutation is then either accepted or rejected with a carefully chosen probability, in order that the probability density of sampling each path is proportional to the contribution it makes to the desired final image. The resulting algorithm is unbiased, handles general geometric and scattering models, and can be far more efficient than previous algorithms on scenes with complex illumination. Furthermore, it is competitive with previous unbiased algorithms even for scenes whose lighting is relatively simple.

To derive this method we first proposed a slight modification to the path integral framework that allows paths to be sampled across the entire image (rather than within each pixel separately). We showed that the Metropolis algorithm can then be used to determine the relative pixel intensities across the image, while the overall image brightness needs to be determined during a separate initialization phase. We addressed the issue of start-up bias during initialization (a common problem with Metropolis applications), and showed that in the case of light transport this bias can be eliminated completely. For the Metropolis phase of the algorithm, we proposed a set of criteria for designing path mutations in order to minimize the error in the final image. We have also described three different mutation strategies we have implemented that partially satisfy these goals, namely *bidirectional mutations*, *lens subpath mutations*, and *perturbations*. Finally, we described several refinements to the basic algorithm that improve its performance in practice.

The main advantage of Metropolis light transport is its ability to handle complex illumination efficiently, by exploring the space of paths that actually contribute to the image. Unlike bidirectional path tracing, it can also handle problems where only a small fraction of the emitted light in the scene reaches the viewer (e.g. due to difficult visibility). Furthermore, since it is a Monte Carlo algorithm it can support complex geometry and materials efficiently. We feel that the ability to handle complex geometry, materials, and illumination is an important goal, since light transport algorithms need to produce reliable, consistent results over the widest possible range of real environments if they are ever going to be widely used.

Bibliography

Addelman, S. & Kempthorne, O. [1961]. Some main-effect plans and orthogonal arrays of strength two, *The Annals of Mathematical Statistics* **32**: 1167–1176.

Al-Gwaiz, M. A. [1992]. *Theory of Distributions*, Marcel Dekker, New York.

Albert, G. E. [1956]. A general theory of stochastic estimates of the Neumann series for solution of certain Fredholm integral equations and related series, *in* M. A. Meyer (ed.), *Symposium on Monte Carlo Methods*, John Wiley & Sons, New York, pp. 37–46.

Ambartzumian, R. V. [1990]. *Factorization Calculus and Geometric Probability*, Cambridge University Press, New York.

American National Standards Institute [1986]. ANSI standard nomenclature and definitions for illuminating engineering,, *ANSI/IES RP-16-1986*, Illuminating Engineering Society, 345 East 47th Street, New York, NY 10017.

Anderson, R. J. [1996]. Tree data structures for n-body simulation, *Proceedings of the 37th Annual Symposium on Foundations of Computer Science*, pp. 224–233.

Appel, A. [1968]. Some techniques for shading machine renderings of solids, *AFIPS 1968 Spring Joint Computer Conference*, Vol. 32, pp. 37–45.

Arvo, J. [1995]. *Analytic Methods for Simulated Light Transport*, PhD thesis, Yale University.

Arvo, J. & Kirk, D. [1990]. Particle transport and image synthesis, *Computer Graphics (SIGGRAPH 90 Proceedings)* **24**(4): 63–66.

Arvo, J., Torrance, K. & Smits, B. [1994]. A framework for the analysis of error in global illumination algorithms, *SIGGRAPH 94 Proceedings*, ACM Press, pp. 75–84.

Aupperle, L. & Hanrahan, P. [1993]. A hierarchical illumination algorithm for surfaces with glossy reflection, *SIGGRAPH 93 Proceedings*, pp. 155–162.

Barnes, J. & Hut, P. [1986]. A hierarchical $O(N \log N)$ force-calculation algorithm, *Nature* **324**(4): 446–449.

Beck, J. & Chen, W. W. L. [1987]. *Irregularities of Distribution*, Cambridge University Press, New York.

Beyer, M. & Lange, B. [1994]. Rayvolution: An evolutionary ray tracing algorithm, *Eurographics Rendering Workshop 1994 Proceedings*, pp. 137–146. Also in *Photorealistic Rendering Techniques*, Springer-Verlag, New York, 1995.

Blinn, J. F. [1978]. Simulation of wrinkled surfaces, *Computer Graphics (SIGGRAPH 78 Proceedings)*, Vol. 12, pp. 286–292.

Bloembergen, N. [1996]. *Nonlinear Optics*, fourth ed., World Scientific, River Edge, New Jersey.

Born, M. & Wolf, E. [1986]. *Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light*, sixth (corrected) ed., Pergamon Press, New York.

Bose, R. C. [1938]. On the application of Galois fields to the problem of the construction of Hyper-Graeco-Latin squares, *Sankhya* **3**: 323–338.

Bose, R. C. & Bush, K. A. [1952]. Orthogonal arrays of strength two and three, *The Annals of Mathematical Statistics* **23**: 508–524.

Brittanica Online [1996]. URL `http://www.eb.com`. “The first encyclopedia on the Internet”.

Bush, K. A. [1952]. Orthogonal arrays of index unity, *The Annals of Mathematical Statistics* **23**: 426–434.

Case, K. M. [1957]. Transfer problems and the reciprocity principle, *Reviews of Modern Physics* **29**(4): 651–658.

Chandrasekhar, S. [1960]. *Radiative Transfer*, Dover Publications, New York.

Chen, S. E., Rushmeier, H. E., Miller, G. & Turner, D. [1991]. A progressive multi-pass method for global illumination, *Computer Graphics (SIGGRAPH 91 Proceedings)*, Vol. 25, pp. 165–174.

Chiu, K., Shirley, P. & Wang, C. [1994]. Multi-jittered sampling, *in* P. Heckbert (ed.), *Graphics Gems IV*, Academic Press, Boston, pp. 370–374.

Christensen, P. H., Salesin, D. H. & DeRose, T. D. [1993]. A continuous adjoint formulation for radiance transport, *Fourth Eurographics Workshop on Rendering Proceedings*, pp. 95–104.

Christensen, P. H., Stollnitz, E. J., Salesin, D. H. & DeRose, T. D. [1996]. Global illumination of glossy environments using wavelets and importance, *ACM Transactions on Graphics* **15**: 37–71.

Cochran, W. G. [1963]. *Sampling Techniques*, second ed., John Wiley & Sons, New York.

Cohen, M. F., Chen, S. E., Wallace, J. R. & Greenberg, D. P. [1988]. A progressive refinement approach to fast radiosity image generation, *Computer Graphics (SIGGRAPH 88 Proceedings)*, Vol. 22, pp. 75–84.

Cohen, M. F. & Wallace, J. R. [1993]. *Radiosity and Realistic Image Synthesis*, Academic Press Professional, San Diego, CA.

Cohen, M., Greenberg, D. P., Immel, D. S. & Brock, P. J. [1986]. An efficient radiosity approach for realistic image synthesis, *IEEE Computer Graphics and Applications* **6**(3): 26–35.

Collins, S. [1995]. Reconstruction of indirect illumination from area luminaires, *Rendering Techniques '95*, pp. 274–283. Also in *Eurographics Rendering Workshop 1996 Proceedings* (June 1996).

Cook, R. L., Porter, T. & Carpenter, L. [1984]. Distributed ray tracing, *Computer Graphics (SIGGRAPH 84 Proceedings)* **18**(3): 137–145.

Cook, R. L. & Torrance, K. E. [1982]. A reflectance model for computer graphics, *ACM Transactions on Graphics* **1**(1): 7–24.

Dagum, P., Karp, R., Luby, M. & Ross, S. [1995]. An optimal algorithm for Monte Carlo estimation, *Proceedings of the 36th Annual Symposium on Foundations of Computer Science*, IEEE, pp. 142–149.

Davis, P. J. & Rabinowitz, P. [1984]. *Methods of Numerical Integration*, second ed., Academic Press, New York.

de Groot, S. R. [1963]. On the development of nonequilibrium thermodynamics, *Journal of Mathematical Physics* **4**(2): 147–153.

de Groot, S. R. & Mazur, P. [1962]. *Non-equilibrium Thermodynamics*, North-Holland, Amsterdam.

de Hoop, A. T. [1960]. A reciprocity theorem for the electromagnetic field scattered by an obstacle, *Applied Scientific Research, Section B* **8**: 135–141.

de la Perrelle, E. T., Moss, T. S. & Herbert, H. [1963]. The measurements of absorptivity and reflectivity, *Infrared Physics* **3**: 35–43.

de Vos, J. C. [1954]. Evaluation of the quality of a blackbody, *Physica* **20**: 669–673.

Delves, L. M. & Mohamed, J. L. [1985]. *Computational Methods for Integral Equations*, Cambridge University Press, New York.

Drude, P. [1900]. *The Theory of Optics*, Dover Publications, New York. This Dover edition published in 1959. Translated from the German by C. R. Mann and R. A. Millikan.

Duderstadt, J. J. & Martin, W. R. [1979]. *Transport Theory*, John Wiley & Sons, New York.

Dutre, P. & Willems, Y. D. [1995]. Potential-driven Monte Carlo particle tracing for diffuse environments with adaptive probability density functions, *Rendering Techniques*

'95, pp. 306–315. Also in *Eurographics Rendering Workshop 1996 Proceedings* (June 1996).

Efron, B. & Stein, C. [1981]. The jackknife estimate of variance, *The Annals of Statistics* **9**: 586–596.

Feynman, R. [1985]. *QED: The Strange Theory of Light and Matter*, Princeton University Press, Princeton, New Jersey. Reprinted in 1988 with corrections.

Fisher, R. A. [1925]. *Statistical Methods for Research Workers*, Oliver and Boyd, Edinburgh. Fourteenth edition also published by Hafner, New York (1973).

Fisher, R. A. [1926]. The arrangement of field experiments, *Journal of the Ministry of Agriculture* **33**: 503–513.

Gershbein, R., Schröder, P. & Hanrahan, P. [1994]. Textures and radiosity: Controlling emission and reflection with texture maps, *SIGGRAPH 94 Proceedings*, ACM Press, pp. 51–58.

Glassner, A. [1994]. A model for fluorescence and phosphorescence, *Eurographics Rendering Workshop 1994 Proceedings*, pp. 57–68. Also in *Photorealistic Rendering Techniques*, Springer-Verlag, New York, 1995.

Glassner, A. [1995]. *Principles of Digital Image Synthesis*, Morgan Kaufmann, New York.

Glassner, A. (ed.) [1989]. *An Introduction to Ray Tracing*, Academic Press.

Goldberg, D. E. [1989]. *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, Massachusetts.

Gondek, J. S., Meyer, G. W. & Newman, J. G. [1994]. Wavelength dependent reflectance functions, *SIGGRAPH 94 Proceedings*, pp. 213–220.

Goral, C. M., Torrance, K. E., Greenberg, D. P. & Battaile, B. [1984]. Modelling the interaction of light between diffuse surfaces, *Computer Graphics (SIGGRAPH 84 Proceedings)*, Vol. 18, pp. 212–222.

Gortler, S. J., Schröder, P., Cohen, M. F. & Hanrahan, P. [1993]. Wavelet radiosity, *SIGGRAPH 93 Proceedings*, pp. 221–230.

Greengard, L. [1988]. *The Rapid Evaluation of Potential Fields in Particle Systems*, MIT Press, Cambridge, Massachusetts.

Greengard, L. & Rokhlin, V. [1987]. A fast algorithm for particle simulations, *Journal of Computational Physics* **73**: 325–348.

Gustafson, K. E. [1987]. *Introduction to Partial Differential Equations and Hilbert Space Methods*, second ed., John Wiley & Sons, New York.

Hall, R. [1989]. *Illumination and Color in Computer Generated Imagery*, Springer-Verlag, New York.

Halmos, P. R. [1950]. *Measure Theory*, Van Nostrand, New York.

Hammersley, J. M. & Handscomb, D. C. [1964]. *Monte Carlo Methods*, Chapman and Hall, New York.

Hanrahan, P., Salzman, D. & Aupperle, L. [1991]. A rapid hierarchical radiosity algorithm, *Computer Graphics (SIGGRAPH 91 Proceedings)*, Vol. 25, pp. 197–206.

He, X. D., Torrance, K. E., Sillion, F. X. & Greenberg, D. P. [1991]. A comprehensive physical model for light reflection, *Computer Graphics (SIGGRAPH 91 Proceedings)*, Vol. 25, pp. 175–186.

Heckbert, P. S. [1990]. Adaptive radiosity textures for bidirectional ray tracing, *Computer Graphics (SIGGRAPH 90 Proceedings)*, Vol. 24, pp. 145–154.

Heckbert, P. S. [1991]. *Simulating Global Illumination Using Adaptive Meshing*, PhD thesis, University of California, Berkeley.

Heckbert, P. S. [1992]. Introduction to global illumination, *SIGGRAPH 92 Global Illumination course notes*, Vol. 18.

Horvitz, D. G. & Thompson, D. J. [1952]. A generalization of sampling without replacement from a finite universe, *Journal of American Statistical Association* **47**: 663–685.

Hovenier, J. W. [1969]. Symmetry relationships for scattering of polarized light in a slab of randomly oriented particles, *Journal of the Atmospheric Sciences* **26**: 488–499.

Hughes, T. J. R. [1987]. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*, Prentice-Hall, Englewood Cliffs, New Jersey.

Immel, D. S., Cohen, M. F. & Greenberg, D. P. [1986]. A radiosity method for non-diffuse environments, *Computer Graphics (SIGGRAPH 86 Proceedings)*, Vol. 20, pp. 133–142.

Jensen, H. W. [1995]. Importance driven path tracing using the photon map, *Eurographics Rendering Workshop 1995*, Eurographics.

Jensen, H. W. [1996]. Global illumination using photon maps, *Eurographics Rendering Workshop 1996 Proceedings*, pp. 22–31. Also in *Rendering Techniques '96*, Springer-Verlag, New York, 1996.

Jones, R. C. [1953]. On reversibility and irreversibility in optics, *Journal of the Optical Society of America* **43**(2): 138–144.

Kajiya, J. T. [1986]. The rendering equation, *Computer Graphics (SIGGRAPH 86 Proceedings)*, Vol. 20, pp. 143–150.

Kalos, M. H. & Whitlock, P. A. [1986]. *Monte Carlo Methods, Volume I: Basics*, John Wiley & Sons, New York.

Keitz, H. A. E. [1971]. *Light Calculations and Measurements*, The Macmillan Company, New York.

Keller, A. [1996]. Quasi-Monte Carlo radiosity, *Eurographics Rendering Workshop 1996 Proceedings*. Also in *Rendering Techniques '96*, Springer-Verlag, New York, 1996.

Keller, A. [1997]. Instant radiosity, *SIGGRAPH 97 Proceedings*, Addison-Wesley, pp. 49–56.

Kerr, D. E. [1987]. Application of the lorentz reciprocity theorem to scattering, *Propagation of Short Radio Waves*, revised ed., Vol. 24 of *IEE Electromagnetic Waves Series*, Peter Peregrinus Ltd., London, chapter Appendix A, pp. 693–698. First edition appeared as Volume 13 of the MIT Radiation Laboratory Series, McGraw-Hill, New York, 1951.

Kirk, D. B. & Arvo, J. [1991]. Unbiased sampling techniques for image synthesis, *Computer Graphics (SIGGRAPH 91 Proceedings)*, Vol. 25, pp. 153–156.

Knittl, Z. [1962]. The principle of reversibility and thin film optics, *Optica Acta* **9**: 33–45.

Kolb, C., Hanrahan, P. & Mitchell, D. [1995]. A realistic camera model for computer graphics, *SIGGRAPH 95 Proceedings*, Addison-Wesley, pp. 317–324.

Kuipers, L. & Niederreiter, H. [1974]. *Uniform Distribution of Sequences*, John Wiley & Sons, New York.

Lafortune, E. P. & Willems, Y. D. [1993]. Bi-directional path tracing, *CompuGraphics Proceedings*, Alvor, Portugal, pp. 145–153.

Lafortune, E. P. & Willems, Y. D. [1994]. A theoretical framework for physically based rendering, *Computer Graphics Forum* **13**(2): 97–107.

Lafortune, E. P. & Willems, Y. D. [1995a]. A 5D tree to reduce the variance of Monte Carlo ray tracing, *Rendering Techniques '95*, pp. 11–20. Also in *Eurographics Rendering Workshop 1996 Proceedings* (June 1996).

Lafortune, E. P. & Willems, Y. D. [1995b]. Reducing the number of shadow rays in bidirectional path tracing, *Winter School of Computer Graphics 1995*. held at University of West Bohemia, Plzen, Czech Republic, 14-18 February 1995.

Langer, M. S. & Zucker, S. W. [1997]. What is a light source?, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 172–178.

Lee, M. E., Redner, R. A. & Uselton, S. P. [1985]. Statistically optimized sampling for distributed ray tracing, *Computer Graphics (SIGGRAPH 85 Proceedings)*, Vol. 19, pp. 61–67.

Lekner, J. [1987]. *Theory of Reflection of Electromagnetic and Particle Waves*, Martinus Nijhoff Publishers, Dordrecht, The Netherlands. Also distributed by Kluwer Academic Publishers, Hingham, Massachusetts.

Lewins, J. [1965]. *Importance, The Adjoint Function: The Physical Basis of Variational and Perturbation Theory in Transport and Diffusion Problems*, Pergamon Press, New York.

Lewis, E. E. & Miller, Jr., W. F. [1984]. *Computational Methods of Neutron Transport*, John Wiley & Sons, New York.

Liebes, Jr., S. [1969]. Brightness—on the ray invariance of B/n^2, *American Journal of Physics* **37**(9): 932–934.

Lischinski, D., Tampieri, F. & Greenberg, D. P. [1992]. Discontinuity meshing for accurate radiosity, *IEEE Computer Graphics and Applications* **12**(6): 25–39.

Lützen, J. [1982]. *The Prehistory of the Theory of Distributions*, Vol. 7 of *Studies in the History of Mathematics and Physical Sciences*, Springer-Verlag, New York.

McCluney, R. [1994]. *Introduction to Radiometry and Photometry*, Artech House, Inc., Boston.

McKay, M. D., Beckman, R. J. & Conover, W. J. [1979]. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, *Technometrics* **21**: 239–245.

McNicholas, H. J. [1928]. Absolute methods in reflectometry, *Journal of Research of the National Bureau of Standards* **1**: 29–37.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. & Teller, E. [1953]. Equations of state calculations by fast computing machines, *Journal of Chemical Physics* **21**: 1087–1091.

Metropolis, N. & Ulam, S. [1949]. The monte-carlo method, *Journal of American Statistical Association* **44**: 335–341.

Meyer, M. A. (ed.) [1956]. *Symposium on Monte Carlo Methods*, John Wiley & Sons, New York.

Milne, E. A. [1930]. Thermodynamics of the stars, *in* D. H. Menzel (ed.), *Selected Papers on the Transfer of Radiation*, Dover Publications, New York, pp. 77–269. This Dover edition published in 1966. Reprinted from the *Handbuch der Astrophysik*, Volume 3, Part I, 1930.

Minnaert, M. [1941]. The reciprocity principle in lunar photometry, *Astrophysical Journal* **93**: 403–410.

Mitchell, D. P. [1987]. Generating antialiased images at low sampling densities, *Computer Graphics (SIGGRAPH 87 Proceedings)*, Vol. 21, pp. 65–72.

Mitchell, D. P. [1992]. Ray tracing and irregularities of distribution, *Third Eurographics Workshop on Rendering Proceedings*, pp. 61–69.

Mitchell, D. P. [1996]. Consequences of stratified sampling in computer graphics, *SIGGRAPH 96 Proceedings*, Addison-Wesley, pp. 277–280.

Mitchell, D. P. & Hanrahan, P. [1992]. Illumination from curved reflectors, *Computer Graphics (SIGGRAPH 92 Proceedings)*, Vol. 26, pp. 283–291.

Moon, P. [1936]. *The Scientific Basis of Illuminating Engineering*, McGraw-Hill, New York.

Nicodemus, F. E. [1963]. Radiance, *American Journal of Physics* **31**(5): 368–377.

Nicodemus, F. E. [1965]. Directional reflectance and emissivity of an opaque surface, *Applied Optics* **4**(7): 767–773.

Nicodemus, F. E. (ed.) [1976]. *Self-Study Manual on Optical Radiation Measurements: Part I—Concepts, Chapters 1 to 3*, Technical Note 910-1, National Bureau of Standards (US).

Nicodemus, F. E. (ed.) [1978]. *Self-Study Manual on Optical Radiation Measurements: Part I—Concepts, Chapters 4 and 5*, Technical Note 910-2, National Bureau of Standards (US).

Nicodemus, F. E. et al. [1977]. Geometric considerations and nomenclature for reflectance, *Monograph 161*, National Bureau of Standards (US).

Niederreiter, H. [1992]. *Random Number Generation and Quasi-Monte Carlo Methods*, Society for Industrial and Applied Mathematics, Philadelphia.

OpenGL Architecture Review Board [1992]. *OpenGL Reference Manual*, Addison-Wesley, Reading, Massachusetts.

Owen, A. B. [1992]. Orthogonal arrays for computer experiments, integration and visualization, *Statistica Sinica* **2**: 439–452.

Owen, A. B. [1994]. Lattice sampling revisited: Monte carlo variance of means over randomized orthogonal arrays, *The Annals of Statistics* **22**: 930–945.

Owen, A. B. [1995a]. Programs to construct and manipulate orthogonal arrays. Available at `http://stat.stanford.edu/reports/owen/oa/`.

Owen, A. B. [1995b]. Randomly permuted (t, m, s)-nets and (t, s)-sequences, *in* H. Niederreiter & P. J.-S. Shiue (eds.), *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*, Springer-Verlag, New York, pp. 299–317.

Owen, A. B. [1997a]. Monte carlo variance of scrambled net quadrature, *SIAM Journal on Numerical Analysis* **34**: 1884–1910.

Owen, A. B. [1997b]. Scrambled net variance for integrals of smooth functions, *The Annals of Statistics* **25**(4): 1541–1562.

Painter, J. & Sloan, K. [1989]. Antialiased ray tracing by adaptive progressive refinement, *Computer Graphics (SIGGRAPH 89 Proceedings)*, Vol. 23, pp. 281–288.

Pattanaik, S. N. & Mudur, S. P. [1993]. Efficient potential equation solutions for global illumination computation, *Computer and Graphics* **17**(4): 387–396.

Pattanaik, S. N. & Mudur, S. P. [1995]. Adjoint equations and random walks for illumination computation, *ACM Transactions on Graphics* **14**: 77–102.

Patterson, H. D. [1954]. The errors of lattice sampling, *Journal of the Royal Statistical Sociey, Series B* **16**: 140–149.

Peercy, M. S. [1993]. Linear color representations for full spectral rendering, *SIGGRAPH 93 Proceedings*, pp. 191–198.

Perina, J. [1985]. *Coherence of Light*, second ed., Kluwer Academic Publishers, Boston.

Peskun, P. H. [1973]. Optimum Monte-Carlo sampling using Markov chains, *Biometrika* **60**(3): 607–612.

Phong, B. T. [1975]. Illumination for computer generated pictures, *Communications of the ACM* **18**(6): 311–317.

Pitman, J. [1993]. *Probability*, Springer-Verlag, New York.

Planck, M. [1914]. *The Theory of Heat Radiation*, second ed., P. Blakiston's Son and Co., Philadelphia. First edition published in 1906. Translated from the second German edition (1913) by Morton Masius. Reprinted in *The History of Modern Physics*, Volume 11, Tomash Publishers/American Institute of Physics, 1989.

Purgathofer, W. [1986]. A statistical method for adaptive stochastic sampling, *Eurographics '86*, North-Holland, pp. 145–152.

Rayleigh, John William Strutt, B. [1877]. *The Theory of Sound, Volume I*, second ed., Dover Publications, New York. This Dover edition published in 1945.

Rayleigh, John William Strutt, B. [1900]. On the law of reciprocity in diffuse reflexion, *Philosophical Magazine* **49**: 324–325.

Rayleigh, John William Strutt, B. [1964]. *Scientific Papers, Volume IV*, Dover Publications, New York.

Reif, J. H., Tygar, J. D. & Yoshida, A. [1994]. Computability and complexity of ray tracing, *Discrete and Computational Geometry* **11**: 265–287.

Rubinstein, R. Y. [1981]. *Simulation and the Monte Carlo Method*, John Wiley & Sons, New York.

Rudin, W. [1973]. *Functional Analysis*, first ed., McGraw-Hill, New York.

Rudin, W. [1987]. *Real and Complex Analysis*, third ed., McGraw-Hill, New York.

Rushmeier, H. E. [1986]. *Extending the radiosity method to transmitting and specularly reflecting surfaces*, Master's thesis, Program of Computer Graphics, Cornell Univ.

Rushmeier, H. E. [1988]. *Realistic Image Synthesis for Scenes with Radiatively Participating Media*, Ph.d. thesis, Cornell University.

Rushmeier, H. E. & Torrance, K. E. [1987]. The zonal method for calculating light intensities in the presence of a participating medium, *Computer Graphics (SIGGRAPH 87 Proceedings)*, Vol. 21, pp. 293–302.

Rushmeier, H. E. & Torrance, K. E. [1990]. Extending the radiosity method to include specularly reflecting and translucent materials, *ACM Transactions on Graphics* **9**(1): 1–27.

Salzberg, B. [1948]. A note on the significance of power reflection, *American Journal of Physics* **16**: 444–446.

Saxon, D. S. [1955]. Tensor scattering matrix for the electromagnetic field, *Physical Review* **100**(6): 1771–1775.

Schröder, P. & Hanrahan, P. [1994]. Wavelet methods for radiance computations, *Eurographics Rendering Workshop 1994 Proceedings*, pp. 303–311. Also in *Photorealistic Rendering Techniques*, Springer-Verlag, New York, 1995.

Schwartz, L. [1966]. *Théorie des Distributions*, Hermann, Paris. Includes Volume I (1950) and Volume II (1951), with corrections and two supplementary chapters.

Shirley, P. [1990a]. *Physically Based Lighting Calculations for Computer Graphics*, PhD thesis, University of Illinois at Urbana-Champaign.

Shirley, P. [1990b]. A ray tracing method for illumination calculation in diffuse-specular scenes, *Graphics Interface ‘90 Proceedings*, pp. 205–212.

Shirley, P. [1991]. Discrepancy as a quality measure for sample distributions,*Eurographics 91 Proceedings*, pp. 183–194.

Shirley, P., Wade, B., Hubbard, P. M., Zareski, D., Walter, B. & Greenberg, D. P. [1995]. Global illumination via density-estimation, *Eurographics Rendering Workshop 1995 Proceedings*, pp. 219–230. Also in *Rendering Techniques ’95*, Springer-Verlag, New York, 1995.

Shirley, P. & Wang, C. [1992]. Distribution ray tracing: Theory and practice, *Third Eurographics Workshop on Rendering Proceedings*, pp. 33–43.

Shirley, P., Wang, C. & Zimmerman, K. [1996]. Monte Carlo methods for direct lighting calculations, *ACM Transactions on Graphics* **15**(1): 1–36.

Siegel, R. & Howell, J. R. [1992]. *Thermal Radiation Heat Transfer*, third ed., Hemisphere Publishing Corp., Washington, D.C.

Sillion, F. X., Arvo, J. R., Westin, S. H. & Greenberg, D. P. [1991]. A global illumination solution for general reflectance distributions, *Computer Graphics (SIGGRAPH 91 Proceedings)*, Vol. 25, pp. 187–196.

Sillion, F. X. & Puech, C. [1989]. A general two-pass method integrating specular and diffuse reflection, *Computer Graphics (SIGGRAPH 89 Proceedings)*, Vol. 23, pp. 335–344.

Smits, B., Arvo, J. & Greenberg, D. [1994]. A clustering algorithm for radiosity in complex environments, *SIGGRAPH 94 Proceedings*, pp. 435–442.

Smits, B. E., Arvo, J. R. & Salesin, D. H. [1992]. An importance-driven radiosity algorithm, *Computer Graphics (SIGGRAPH 92 Proceedings)*, Vol. 26, pp. 273–282.

Snyder, J. M. & Barr, A. H. [1987]. Ray tracing complex models containing surface tessellations, *Computer Graphics (SIGGRAPH 87 Proceedings),* Vol. 21, pp. 119–128.

Sobol', I. M. [1994]. *A Primer for the Monte Carlo Method,* CRC Press, Boca Raton, Florida. Translated from the fourth Russian edition (1985).

Spanier, J. & Gelbard, E. M. [1969]. *Monte Carlo Principles and Neutron Transport Problems,* Addison-Wesley, Reading, Massachusetts.

Stam, J. & Languenou, E. [1996]. Ray tracing in non-constant media, *Eurographics Rendering Workshop 1996 Proceedings.* Also in *Rendering Techniques '96,* Springer-Verlag, New York, 1996.

Steel, W. H. [1974]. Luminosity, throughput, or etendue?, *Applied Optics* **13**(4): 704–705.

Stein, M. [1987]. Large sample properties of simulations using latin hypercube sampling, *Technometrics* **29**: 143–151.

Strang, G. [1986]. *Introduction to Applied Mathematics,* Wellesley-Cambridge Press, Wellesley, Massachusetts.

Tang, B. [1993]. Orthogonal array-based latin hypercubes, *Journal of American Statistical Association* **88**: 1392–1397.

Taylor, A. E. & Lay, D. C. [1980]. *Introduction to Functional Analysis,* second ed., John Wiley & Sons, New York.

Tezuka, S. [1995]. *Uniform Random Numbers: Theory and Practice,* Kluwer Academic Publishers, Boston.

Tingwaldt, C. P. [1952]. Über das helmholtzsche reziprozitätsgesetz in der optik, *Optik* **9**: 248–253.

Torrance, K. E. & Sparrow, E. M. [1967]. Theory for off–specular reflection from roughened surfaces, *Journal of the Optical Society of America* **57**(9): 1105–1114.

Ulam, S. M. [1987]. *Stanislaw Ulam, 1909–1984*, Los Alamos National Laboratory, Los Alamos, New Mexico. A special issue of *Los Alamos Science* (no. 15).

van de Hulst, H. C. [1957]. *Light Scattering by Small Particles*, John Wiley & Sons, New York.

van de Hulst, H. C. [1980]. *Multiple Light Scattering: Tables, Formulas, and Applications, Volume I*, Academic Press, New York.

van Kampen, N. G. [1954]. Quantum statistics of irreversible processes, *Physica* **20**: 603–622.

Veach, E. [1996]. Non-symmetric scattering in light transport algorithms, *Eurographics Rendering Workshop 1996 Proceedings*. Also in *Rendering Techniques '96*, Springer-Verlag, New York, 1996.

Veach, E. & Guibas, L. [1994]. Bidirectional estimators for light transport, *Eurographics Rendering Workshop 1994 Proceedings*, pp. 147–162. Also in *Photorealistic Rendering Techniques*, Springer-Verlag, New York, 1995.

Veach, E. & Guibas, L. J. [1995]. Optimally combining sampling techniques for Monte Carlo rendering, *SIGGRAPH 95 Proceedings*, Addison-Wesley, pp. 419–428.

Veach, E. & Guibas, L. J. [1997]. Metropolis light transport, *SIGGRAPH 97 Proceedings*, Addison-Wesley, pp. 65–76.

von Fragstein, C. [1950]. Energieübergang an der grenze zweier absorbierender medien mit einer anwendung auf die wärmestrahlung in absorbierenden körpern, *Annelen der Physik* **7**(6): 63–72.

von Fragstein, C. [1955]. Ist eine lichtbewegung stets umkehrbar?, *Optica Acta* **2**(1): 16–22.

von Helmholtz, H. [1856]. *Helmholtz's Treatise on Physiological Optics*, Vol. 1, Dover Publications, New York. This Dover edition published in 1962, James P. C. Southall, Ed. Translated from the third German edition (1909), which includes the entire verbatim text of the first edition (1856).

von Helmholtz, H. [1903]. *Vorlesungen über Theorie Der Wärme*, Vol. 6 of *Vorlesungen über theoretische Physik*, J. A. Barth, Leipzig.

Šantavý, I. [1961]. On the reversibility of light beams in conducting media, *Optica Acta* **8**: 301–307.

Wallace, J. R., Cohen, M. F. & Greenberg, D. P. [1987]. A two-pass solution to the rendering equation: A synthesis of ray tracing and radiosity methods, *Computer Graphics (SIGGRAPH 87 Proceedings)*, Vol. 21, pp. 311–320.

Ward, G. J. [1994]. The RADIANCE lighting simulation and rendering system, *SIGGRAPH 94 Proceedings*, pp. 459–472.

Ward, G. J., Rubinstein, F. M. & Clear, R. D. [1988]. A ray tracing solution for diffuse interreflection, *Computer Graphics (SIGGRAPH 88 Proceedings)*, Vol. 22, pp. 85–92.

Whitted, T. [1980]. An improved illumination model for shaded display, *Communications of the ACM* **32**(6): 343–349.

Wigner, E. P. [1954]. Derivations of Onsager's reciprocal relations, *Journal of Chemical Physics* **22**(11): 1912–1915.

Yates, F. [1953]. *Sampling Methods for Censuses and Surveys*, second ed., Griffin, London. Fourth edition also published by Macmillan, New York (1981).

Zauderer, E. [1989]. *Partial Differential Equations of Applied Mathematics*, second ed., John Wiley & Sons, New York.

Zimmerman, K. & Shirley, P. [1995]. A two-pass realistic image synthesis method for complex scenes, *Eurographics Rendering Workshop 1995 Proceedings*. Also in *Rendering Techniques '95*, Springer-Verlag, New York, 1995.

Index