A

Monte Carlo Integration

THE techniques developed in this dissertation are all Monte Carlo methods. Monte Carlo methods are numerical techniques which rely on random sampling to *approximate* their results. Monte Carlo integration applies this process to the numerical estimation of integrals. In this appendix we review the fundamental concepts of Monte Carlo integration upon which our methods are based. From this discussion we will see why Monte Carlo methods are a particularly attractive choice for the multidimensional integration problems common in computer graphics. Good references for Monte Carlo integration in the context of computer graphics include Pharr and Humphreys [2004], Dutré et al. [2006], and Veach [1997].

The term "Monte Carlo methods" originated at the Los Alamos National Laboratory in the late 1940s during the development of the atomic bomb [Metropolis and Ulam, 1949]. Not surprisingly, the development of these methods also corresponds with the invention of the first electronic computers, which greatly accelerated the computation of repetitive numerical tasks. Metropolis [1987] provides a detailed account of the origins of the Monte Carlo method.

Las Vegas algorithms are another class of method which rely on randomization to compute their results. However, in contrast to Las Vegas algorithms, which always produce the exact or correct solution, the accuracy of Monte Carlo methods can only be analyzed from a statistical viewpoint. Because of this, we first review some basic principles from probability theory before formally describing Monte Carlo integration.

A.1 Probability Background

In order to define Monte Carlo integration, we start by reviewing some basic ideas from probability.

A.1.1 Random Variables

A *random variable* X is a function that maps outcomes of a random process to numbers. A random variable can be either discrete (e.g., the roll of a six-sided die where a fixed set of outcomes is possible, $X = \{1,2,3,4,5,6\}$), or continuous (e.g., a person's height, which can take on real values $\mathbb{R}$). In computer graphics we more commonly deal with continuous random variables which take on values over ranges of continuous domains (e.g., the real numbers $\mathbb{R}$ or the sphere of directions Ω).

A.1.2 Cumulative Distributions and Density Functions

The *cumulative distribution function*, or CDF, of a random variable X is the probability that a value chosen from the variable's distribution is less than or equal to some thresold x:

$$cdf(x) = Pr\{X \leq x\}. \tag{A.1}$$

The corresponding *probability density function*, or PDF, is the derivative of the CDF:

$$pdf(x) = \frac{d}{dx} cdf(x). \tag{A.2}$$

CDFs are always monotonically increasing, which means that the PDF is always non-negative. An important relationship arises from the above two equations, which allows us to compute the probability that a random variable lies within an interval:

$$Pr\{a \leq X \leq b\} = \int_a^b pdf(x)\, dx. \tag{A.3}$$

From this expression it is clear that the PDF must always integrate to one over the full extent of its domain.

A.1.3 Expected Values and Variance

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

$$E[Y] = \int_{\mu(x)} f(x)\, pdf(x)\, d\mu(x), \tag{A.4}$$

while its *variance* is

$$\sigma^2[Y] = E\left[(Y - E[Y])^2\right], \tag{A.5}$$

where σ, the *standard deviation*, is the square root of the variance. From these definitions it is easy to show that for any constant a,

$$E[a\,Y] = a\,E[Y], \tag{A.6}$$

$$\sigma^2[a\,Y] = a^2\,\sigma^2[Y]. \tag{A.7}$$

Furthermore, the expected value of a sum of random variables Y_i is the sum of their expected values:

$$E\left[\sum_i Y_i\right] = \sum_i E[Y_i]. \tag{A.8}$$

From these properties it is possible to derive a simpler expression for the variance:

$$\sigma^2[Y] = E\left[Y^2\right] - E[Y]^2. \tag{A.9}$$

Additionally, if the random variables are *uncorrelated*, a summation property also holds for the variance[1]:

$$\sigma^2\left[\sum_i Y_i\right] = \sum_i \sigma^2[Y_i]. \tag{A.10}$$

A.2 The Monte Carlo Estimator

The Basic Estimator. Monte Carlo integration uses random sampling of a function to numerically compute an estimate of its integral. Suppose that we want to integrate the one-dimensional function $f(x)$ from a to b:

$$F = \int_a^b f(x)\,dx. \tag{A.11}$$

We can approximate this integral by averaging samples of the function f at uniform random points within the interval. Given a set of N uniform random variables $X_i \in [a,b)$ with a corresponding PDF of $1/(b-a)$, the Monte Carlo estimator for computing F is

$$\left\langle F^N \right\rangle = (b-a)\frac{1}{N-1}\sum_{i=0}^{N} f(X_i), \tag{A.12}$$

The random variable $X_i \in [a,b)$ can be constructed by warping a canonical random number uniformly distributed between zero and one, $\xi_i \in [0,1)$:

$$X_i = a + \xi_i(b-a). \tag{A.13}$$

Using this construction, we can expand the estimator to:

$$\left\langle F^N \right\rangle = (b-a)\frac{1}{N}\sum_{i=0}^{N-1} f\left(a+\xi_i(b-a)\right). \tag{A.14}$$

[1] This property is often made with the stronger condition that the variables are *independent*; however, it suffices for them to be uncorrelated.

Since $\langle F^N \rangle$ is a function of X_i, it is itself a random variable. We use this notation to clarify that $\langle F^N \rangle$ is an approximation of F using N samples.

Intuitively, the Monte Carlo estimator in Equation A.12 computes the mean value of the function $f(x)$ over the interval a to b, and then multiplies this mean by the length of the interval $(b-a)$. By moving $(b-a)$ into the summation, the estimator can be thought of as choosing a height at a random evaluation of the function and averaging a set of rectangular areas computed by multiplying this height by the interval length $(b-a)$. These two interpretations are illustrated in Figure A.1.

A.2.1 Expected Value and Convergence

It is easy to show that the expected value of $\langle F^N \rangle$ is in fact F:

$$\begin{aligned}
E\left[\langle F^N \rangle\right] &= E\left[(b-a)\frac{1}{N}\sum_{i=0}^{N-1} f(X_i)\right], \\
&= (b-a)\frac{1}{N}\sum_{i=0}^{N-1} E\left[f(X_i)\right], && \text{from Equations A.8 and A.6} \\
&= (b-a)\frac{1}{N}\sum_{i=0}^{N-1} \int_a^b f(x)\,pdf(x)\,dx, && \text{from Equation A.4} \\
&= \frac{1}{N}\sum_{i=0}^{N-1} \int_a^b f(x)\,dx, && \text{since } pdf(x) = 1/(b-a) \\
&= \int_a^b f(x)\,dx, \\
&= F. && \text{(A.15)}
\end{aligned}$$

Furthermore, as we increase the number of samples N, the estimator $\langle F^N \rangle$ becomes a closer and closer approximation of F. Due to the *Strong Law of Large Numbers*, in the limit we can guarantee that we have the exact solution:

$$Pr\left\{\lim_{N\to\infty} \langle F^N \rangle = F\right\} = 1. \tag{A.16}$$

In practice we are interested in knowing just how quickly this estimate converges to a

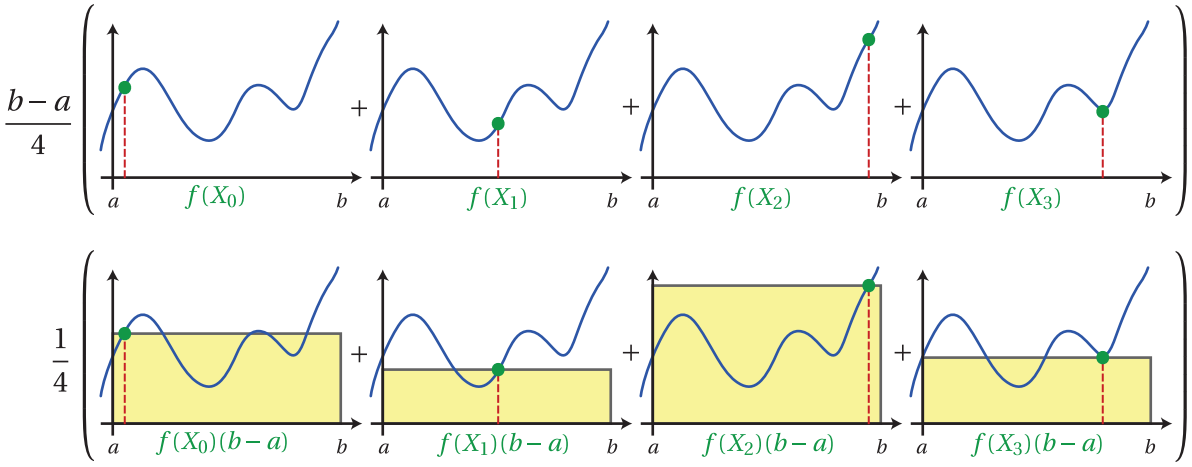

Figure A.1: An illustration of the two interpretations of the basic Monte Carlo estimator in Equation A.12 using four samples: computing the mean value, or height, of the function and multiplying by the interval length (top), or computing the average of several rectangular areas (bottom).

sufficiently accurate solution. This can be analyzed by determining the *convergence rate* of the estimators variance. In the next section we will show that the standard deviation is proportional to:

$$\sigma\left[\langle F^N \rangle\right] \propto \frac{1}{\sqrt{N}}. \tag{A.17}$$

Unfortunately, this means that we must quadruple the number of samples in order to reduce the error by half!

Standard integration techniques exist which converge much faster in one dimension; however, these techniques suffer from the *curse of dimensionality*, where the convergence rate becomes *exponentially* worse with increased dimensions. The basic Monte Carlo estimator above can easily be extended to multiple dimensions, and, in contrast to deterministic quadrature techniques, the convergence rate for Monte Carlo is independent of the number of dimensions in the integral. This makes Monte Carlo integration the only practical technique for many high dimensional integration problems, such as those encountered when computing global illumination.

A.2.2 Multidimensional Integration

Monte Carlo integration can be generalized to use random variables drawn from arbitrary PDFs and to compute multidimensional integrals, such as

$$F = \int_{\mu(x)} f(x)\, d\mu(x), \tag{A.18}$$

with the following modification to Equation A.12:

$$\left\langle F^N \right\rangle = \frac{1}{N} \sum_{i=0}^{N-1} \frac{f(X_i)}{pdf(X_i)}. \tag{A.19}$$

It is similarly easy to show that this generalized estimator also has the correct expected value:

$$\begin{aligned}
E\left[\left\langle F^N \right\rangle\right] &= E\left[\frac{1}{N} \sum_{i=0}^{N-1} \frac{f(X_i)}{pdf(X_i)}\right], \\
&= \frac{1}{N} \sum_{i=0}^{N-1} E\left[\frac{f(X_i)}{pdf(X_i)}\right], \\
&= \frac{1}{N} \sum_{i=0}^{N-1} \int_\Omega \frac{f(x)}{pdf(x)} pdf(x)\, dx, \\
&= \frac{1}{N} \sum_{i=0}^{N-1} \int_\Omega f(x)\, dx, \\
&= \int_\Omega f(x)\, dx, \\
&= F.
\end{aligned} \tag{A.20}$$

In addition to the convergence rate, a secondary benefit of Monte Carlo integration over traditional numerical integration techniques is the ease of extending it to multiple dimensions. Deterministic quadrature techniques require using N^d samples for a d-dimensional integral. In contrast, Monte Carlo techniques provide the freedom of choosing any arbitrary number of samples.

As mentioned previously, the Monte Carlo estimator has a constant $O(\sqrt{N})$ convergence rate in any dimension. However, in many situations it is possible to do much better. The efficiency of Monte Carlo integration can be significantly improved using a variety of techniques, which we

discuss in the next section.

A.3 Variance Reduction

Sources of Variance. In order to improve the quality of Monte Carlo integration we need to reduce variance. Since the samples in Monte Carlo integration are independent, using Equation A.10, the variance of $\langle F^N \rangle$ can be simplified to:

$$\begin{aligned}
\sigma^2 \left[\langle F^N \rangle \right] &= \sigma^2 \left[\frac{1}{N} \sum_{i=0}^{N-1} \frac{f(X_i)}{pdf(X_i)} \right] \\
&= \frac{1}{N^2} \sum_{i=0}^{N-1} \sigma^2 \left[\frac{f(X_i)}{pdf(X_i)} \right] \\
&= \frac{1}{N^2} \sum_{i=0}^{N-1} \sigma^2 \left[Y_i \right] \\
&= \frac{1}{N} \sigma^2 \left[Y \right], \qquad \text{(A.21)}
\end{aligned}$$

and hence,

$$\sigma \left[\langle F^N \rangle \right] = \frac{1}{\sqrt{N}} \sigma \left[Y \right], \qquad \text{(A.22)}$$

where $Y_i = f(X_i)/pdf(X_i)$ and Y represents the evaluation of any specific Y_i, e.g., $Y = Y_1$.

This derivation proves our earlier statement that the standard deviation converges with $O(\sqrt{N})$. Moreover, this expression shows that by reducing the variance of each Y_i we can reduce the overall variance of $\langle F^N \rangle$.

Variance-reduction techniques try to make each Y_i as constant as possible in order to reduce the overall error of the estimator. A significant amount of research has been put into this area, leading to a number of complementary techniques. Most of these methods rely on exploiting some previous knowledge of the function being integrated.

A.3.1 Importance Sampling

Importance sampling reduces variance by observing that we have the freedom to choose the PDF used during integration. By choosing samples from a distribution $pdf(x)$, which has a *similar shape* as the function $f(x)$ being integrated, variance is reduced. Intuitively, importance sampling attempts to place more samples where the contribution of the integrand is high, or "important." If we can properly guess the important regions during integration, the variance of the standard Monte Carlo estimator can be significantly reduced.

The Perfect Estimator

To demonstrate the effect of importance sampling, consider a PDF which is exactly proportional to the function being integrated, $pdf(x) = c\,f(x)$ for some normalization constant c. Since c is a constant, if we apply this PDF to the Monte Carlo estimator in Equation A.19, each sample X_i would have the same value,

$$Y_i = \frac{f(X_i)}{pdf(X_i)} = \frac{\cancel{f(X_i)}}{c\cancel{f(X_i)}} = \frac{1}{c}. \tag{A.23}$$

By using this PDF we have succeeded at reducing the variance of each Y_i. In fact, since each Y_i returns the same value, the overall variance is zero!

Since the PDF must integrate to one, it is easy to derive the value of c:

$$c = \frac{1}{\int f(x)\,dx}. \tag{A.24}$$

This unfortunately shows us that determining the normalization constant c in the PDF involves solving the integral we are interested in estimating in the first place. This best case is therefore not a realistic situation. In practice, all validly constructed PDFs will still produce a convergence rate of $\sigma \propto \sqrt{N}$; however, by choosing a PDF that is similar to $f(x)$, we can make the variance arbitrarily low.

The only restriction on the PDF, in order to maintain the correct expected value from Equation A.20, is that it must be non-zero everywhere where the function $f(x)$ is non-zero.

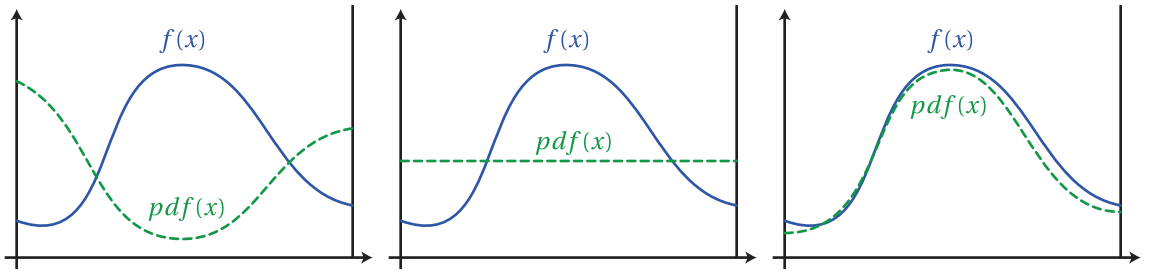

Figure A.2: Comparison of three probability density functions. The PDF on the right provides variance reduction over the uniform PDF in the center. However, using the PDF on the left would significantly increase variance over simple uniform sampling.

Importance sampling should not be used carelessly however. Though we can decrease the variance by using a "good" importance function, we can also make the variance arbitrarily large by choosing an importance function that increases the variance of Y. Figure A.2 shows a uniform PDF, as well as examples of PDFs which reduce and increase the overall variance.

Importance Sampling Complex Functions

Most of the time, the integrand $f(x)$ is very complicated, and we cannot guess its full behavior ahead of time. However, we may know something about its general structure. For instance, the integrand function $f(x)$ may in fact be the combination of more than one function, e.g., $f(x) = g(x)\,h(x)$. In these situations, it may not be possible to create a PDF exactly proportional to $f(x)$, but, if we know one of the functions in advance, we may be able to construct a PDF proportional to a portion of $f(x)$, e.g., $pdf_g(x) \propto g(x)$. In this situation, the Monte Carlo estimator simplifies to:

$$\begin{aligned}\langle F^N \rangle &= \frac{1}{N}\sum_{i=0}^{N-1}\frac{f(X_i)}{pdf_g(X_i)},\\ &= \frac{1}{N}\sum_{i=0}^{N-1}\frac{g(X_i)\,h(X_i)}{c\,g(X_i)},\\ &= \frac{1}{cN}\sum_{i=0}^{N-1} h(X_i).\end{aligned} \tag{A.25}$$

Since $h(x)$ will in general be smoother than the full $g(x)\,h(x)$, the variance of the Monte Carlo estimator is reduced.

Multiple Importance Sampling

What if we could also create a PDF which is proportional to $h(x)$, $pdf_h \propto h(x)$? Should we sample according to pdf_h or pdf_g? As it turns out, we don't have to choose. We can benefit from distributing samples independently from each PDF and combining the results using *multiple importance sampling* [Veach and Guibas, 1995; Veach, 1997].

A.3.2 Control Variates

Control variates is another variance-reduction technique which relies on some prior knowledge of the behavior of f. The idea behind control variates is to find a function g which can be analytically integrated and subtract it from the integral of f:

$$\begin{aligned} F &= \int_a^b f(x)\,dx, \\ &= \int_a^b f(x) - g(x)\,dx + \int_a^b g(x)\,dx, \\ &= \int_a^b f(x) - g(x)\,dx + G. \end{aligned} \tag{A.26}$$

We can then apply Monte Carlo integration to the modified integrand $f(x) - g(x)$:

$$\left\langle F_c^N \right\rangle = \left(\frac{1}{N} \sum_{i=0}^{N-1} \frac{f(X_i) - g(X_i)}{pdf(X_i)} \right) + G, \tag{A.27}$$

where the value of the integral G is known exactly.

If $f(X_i)$ and $g(X_i)$ are correlated, then the variance of $\left\langle F^N \right\rangle$ is reduced. For control variates, g should be chosen so that $f - g$ is nearly constant. In contrast, with importance sampling we tried to choose a PDF, $pdf = g$, so that f/g was nearly constant.

As with importance sampling, we can obtain optimal results by choosing a g such that

$$Y_i = \frac{f(X_i) - g(X_i)}{pdf(X_i)} \tag{A.28}$$

is a constant. Unfortunately, this again requires knowledge of the true integral F.

Another way to think about control variates is by rewriting Equation A.27 as

$$\begin{aligned}\left\langle F_c^N \right\rangle &= \left(\frac{1}{N} \sum_{i=0}^{N-1} \frac{f(X_i)}{pdf(X_i)} \right) + \left(G - \frac{1}{N} \sum_{i=0}^{N-1} \frac{g(X_i)}{pdf(X_i)} \right), \\ &= \left\langle F^N \right\rangle + G - \left\langle G^N \right\rangle, \end{aligned} \tag{A.29}$$

where the same set of random samples X_i are used in both summations. Using this formulation, control variates can be interpreted as computing the difference between the exact integral G and its Monte Carlo approximation $\left\langle G^N \right\rangle$, and then adding this difference to our original Monte Carlo estimator $\left\langle F^N \right\rangle$. The idea is to detect what type of errors are introduced during Monte Carlo sampling of a known integral. If g and f are correlated, then we can expect the same errors to be present in the integral of f, so we can remove them.

A.3.3 Uniform Sample Placement

Importance sampling probabilistically places more samples where the function is deemed more important. The problem with this is that it does not eliminate sample "clumping," as the PDF only dictates the expected number, not the actual number, of samples in any given region. Intuitively, sample clumping is wasteful, since a sample that is very close to another does not provide much new information about the function being integrated. In order to obtain the best solution, we want to maximize the amount of information we gain from each sample.

Stratified Sampling

One powerful variance-reduction technique that addresses this problem is called stratified sampling. Stratified sampling works by splitting up the original integral into a sum of integrals over sub-domains. In its simplest form, stratified sampling divides the domain $[a,b]$ into N sub-domains (or *strata*) and places a random sample within each of these intervals. If using a

uniform PDF, with $\xi_i \in [0,1)$, this can be expressed as

$$\begin{aligned}\left\langle F_s^N \right\rangle &= \frac{(b-a)}{N} \sum_{i=0}^{N-1} f\left(X_i\right), \\ &= \frac{(b-a)}{N} \sum_{i=0}^{N-1} f\left(a + \tfrac{i+\xi_i}{N}(b-a)\right) \\ &= \frac{(b-a)}{N} \sum_{i=0}^{N-1} f\left(a + \xi_i^N (b-a)\right). \end{aligned} \tag{A.30}$$

Compare this estimator to the expanded basic estimator in Equation A.14. Most of the time, stratification can be added by simply replacing a canonical random number $\xi_i \in [0,1)$ with a sequence of stratified random numbers in the same range $\xi_i^N = \frac{i+\xi_i}{N}$.

It can be shown that stratified sampling can never result in higher variance than pure random sampling. In fact, stratified sampling is asymptotically better, since the error reduces linearly with the number of samples, $\sigma \propto 1/N$. Therefore, whenever possible, stratified sampling *should* be used.

Stratified sampling does place more restrictions on the sampling process. This can become problematic when, for instance, the number of samples is not known in advance. Furthermore, though it is possible to extend stratified sampling to higher dimensions, in general, N^d strata would need to be created for a d dimensional domain. This explosion of samples can make fine-grained control of simulation times difficult.

Comparison to Deterministic Quadrature. Stratified sampling incorporates ideas from deterministic quadrature, while still retaining the statistical properties of Monte Carlo integration. It is informative to compare Equation A.30 to a simple Riemann summation of the integral:

$$F = \int_a^b f(x)\, dx \approx \sum_{i=0}^{N-1} f(x_i)\, \Delta x, \tag{A.31}$$

$$\approx \sum_{i=0}^{N-1} f\left(a + \tfrac{i}{N}(b-a)\right) \frac{(b-a)}{N}, \tag{A.32}$$

$$\approx \frac{(b-a)}{N} \sum_{i=0}^{N-1} f\left(a + \tfrac{i}{N}(b-a)\right). \tag{A.33}$$

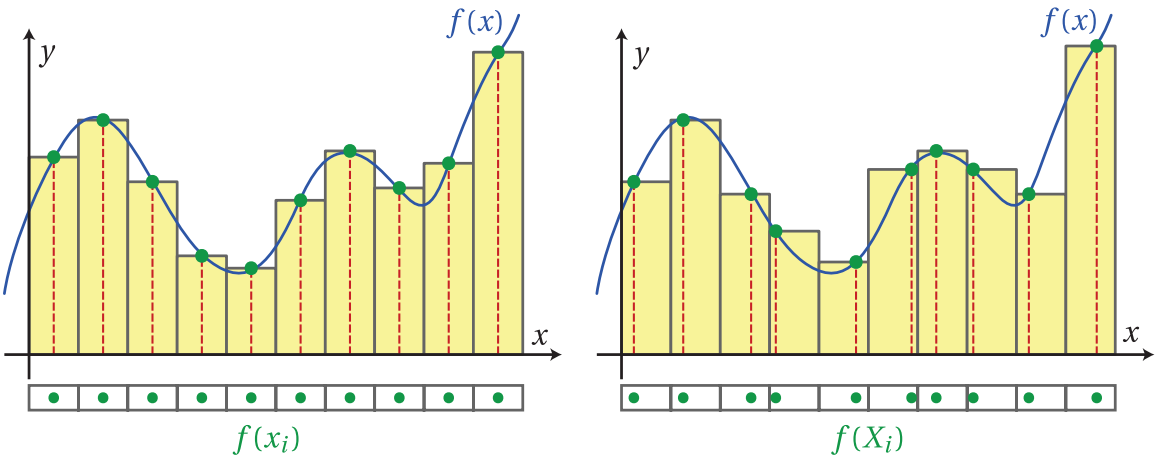

Figure A.3: Deterministic quadrature techniques such as a Riemann summation (left) sample the function at *regular* intervals. Conceptually, stratified Monte Carlo integration (right) performs a very similar summation, but instead evaluates the function at a *random* location within each of the strata.

In effect, the only difference between the Riemann sum and the stratified estimator in Equation A.30 is that Riemann summation evaluates the function at deterministic locations within the domain (the start of each stratum in the above example) whereas stratified Monte Carlo places the samples randomly within each stratum. This distinction is illustrated in Figure A.3.

Other Approaches

There is a wide range of other methods to more uniformly distribute samples. We illustrate some of these approaches in Figure A.4. Other stratification approaches developed include N-Rooks [Shirley, 1990] or Latin hypercube sampling [McKay et al., 1979], multi-jittered [Chiu et al., 1994] and orthogonal array sampling [Owen, 1992], and multi-stage N-Rooks sampling [Wang and Sung, 1999]. Veach [1997] provides a good overview of these approaches.

Several advanced techniques go beyond analyzing just the variance of the Monte Carlo estimator. One approach is to measure the quality of the sample distribution based on frequency analysis of the variance. This analysis shows that, to minimize variance while avoiding aliasing artifacts beyond the Nyquist limit, a sample distribution conforming to a blue-noise frequency spectrum is desirable [Cook, 1986; Mitchell, 1991]. In fact, this distribution of samples has been observed in the placement of photoreceptors in the eye of the rhesus monkey [Yellott, 1983].

Another technique is to compute the *discrepancy* [Shirley, 1991] of the points, which

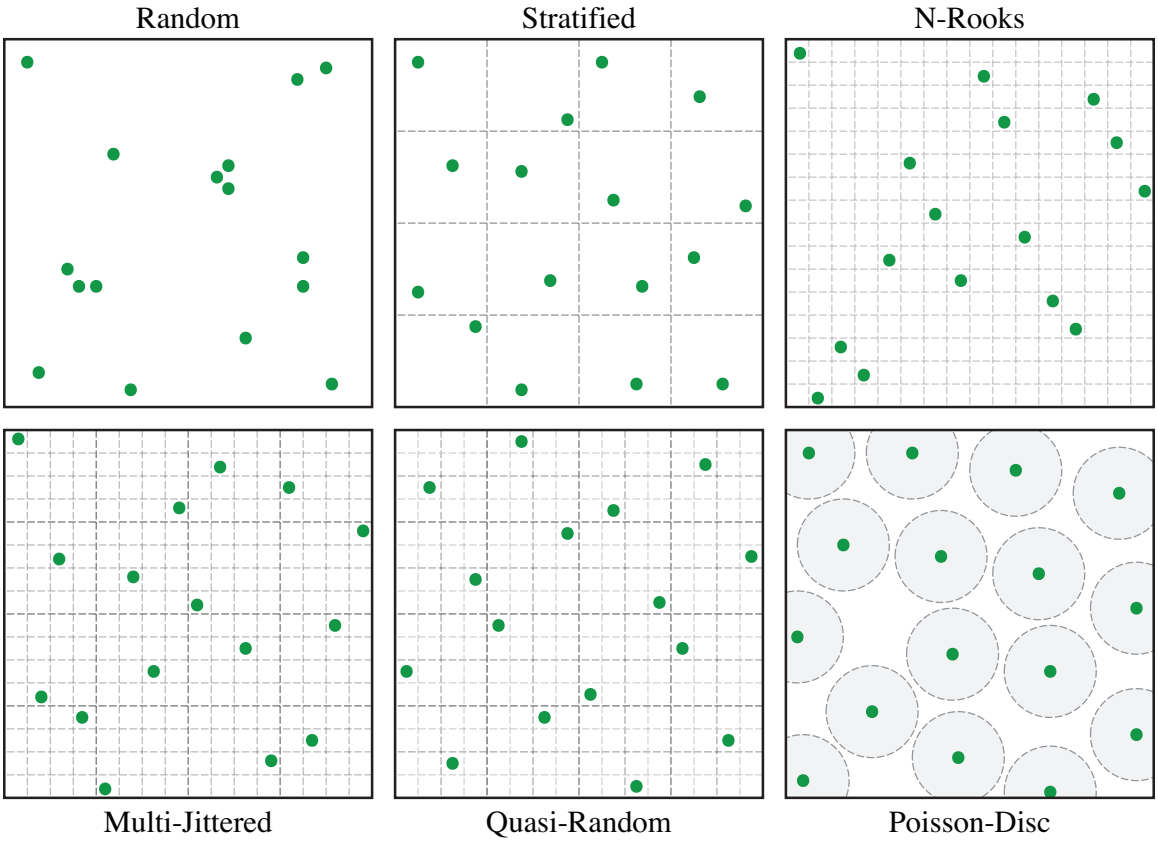

Figure A.4: An illustration comparing several 2D sampling approaches each using 16 samples. Purely random sampling (top left) can suffer from clumping, which increases variance by undersampling other regions of the integrand. All of the other approaches illustrated try to minimize this clumping to reduce variance.

measures how much the density of the points varies from one location to the next. Quasi-Monte Carlo techniques take this a step further and completely replace the use of pseudo-random numbers with judiciously tailored deterministic sequences. These quasi-random numbers are constructed explicitly to minimize clumping by reducing the discrepancy of the point set. Quasi-Monte Carlo sampling is therefore also referred to as low-discrepency sampling. Niederreiter [1992] provides an excellent theoretical introduction to Quasi-Monte Carlo techniques. Quasi-Monte Carlo techniques were introduced to computer graphics by Keller. See his dissertation and recent technical report for a thorough introduction [1998; 2003].

A.3.4 Adaptive Sampling

Like importance sampling, adaptive sampling is another technique which tries to intelligently place more samples where they will provide the most benefit. However, importance sampling requires some *a priori* knowledge of the function being integrated, whereas adaptive sampling relaxes this requirement by inferring this information *on the fly* during the sampling process.

Adaptive sampling tries to reduce variance by making adaptive decisions based on the samples taken so far and placing future samples in areas of high variance. The motivation for this approach is that regions of the integrand which are smooth can be well approximated using only a few samples, whereas regions with rapid variation require more samples to faithfully capture the behavior of the integrand.

A number of adaptive sampling techniques have been developed in computer graphics, most of which deal exclusively with adaptively sampling the image plane. In addition to introducting ray tracing, Whitted [1980] also proposed the use of a hierarchical adaptive sampling technique to increase the sample density around image discontinuities. Several other researchers have proposed stochastic adaptive sampling of the image plane [Dippé and Wold, 1985; Mitchell, 1987; Painter and Sloan, 1989; Mitchell, 1991].

Though adaptive sampling can be a very effective optimization technique, a number of issues complicate its use in practice. Firstly, choosing a robust adaptive refinement criterion can be very difficult and application specific. This can be particularly challenging in the context of Monte Carlo ray tracing since each Monte Carlo sample contains significant variance, making features and noise indistinguishable during adaptive sample refinement. Nevertheless, successful applications of Monte Carlo adaptive sampling in computer graphics do exist. For instance, the irradiance caching algorithm described in Chapter 3 and our volumetric radiance caching scheme developed in Chapter 6 are in essence approaches for adaptively sampling the indirect illumination on surfaces and within media, respectively. Another consideration with adaptive sampling is that, unlike importance sampling, it can introduce unbounded systematic errors unless special precautions are taken [Kirk and Arvo, 1991]. This systematic error, called *bias*, is

discussed in the next section.

A.3.5 Biased Monte Carlo

Estimators that always produce the correct expected value (like the ones we have considered so far) are called *unbiased* estimators. By using unbiased estimators, due to Equation A.15, we can be confident that the expected value of the estimator, using any number of samples, will always be correct. The only error in unbiased estimators is variance.

However, it is also possible to construct estimators which do not satisfy Equation A.15. These estimators are called *biased*, since, in addition to variance, they also contain a systematic bias:

$$\beta\left[\left\langle F^N\right\rangle\right] = E\left[\left\langle F^N\right\rangle\right] - F, \tag{A.34}$$

which is the difference in the expected value of the estimator and the actual value of the integral.

A biased estimator is called *consistent* if it converges to the correct result with more samples. Hence, consistent biased estimators satisfy Equation A.16 but not Equation A.15. A simple example of a consistent but biased estimator for I would be

$$F \approx (b-a)\frac{1}{N+1}\sum_{i=0}^{N-1} f(X_i). \tag{A.35}$$

It is easy to see that this estimator does not have the correct expected value, but as we increase the number of samples it does converge to the solution.

Sometimes it is useful to use biased estimators if they result in significantly reduced variance. All the techniques developed in this dissertation are consistent biased Monte Carlo methods, which introduce bias for exactly this purpose.

Sometimes, a more informative way to think about biased estimators is that they compute *unbiased* estimates of the *wrong* answer! In effect, biased estimators replace the integral I with a different integral $\bar{I}$, which has some desirable properties. For instance, it could be easier to integrate, or it could have lower variance. In a consistent estimator, as we increase the number of samples, the properties of $\bar{I}$ start to resemble the true function I more closely.

This alternative interpretation can be useful because it provides higher level knowledge about the exact form of the bias introduced. For instance, in Chapter 8 we proved that photon mapping actually computes an *unbiased* estimate of the *blurred* radiance. The amount of blur is directly controlled by the size and number of photons in the radiance estimate. This also makes it clear that photon mapping is consistent, since as we decrease the amount of blur to zero, the bias disappears.