A Variance Analysis of the Metropolis Light Transport Algorithm

Michael Ashikhmin, Simon Premože, Peter Shirley, and Brian Smits

School of Computing, University of Utah, 50 S. Central Campus Dr., Rm. 3190, Salt Lake City, UT 84112–9205, USA

Abstract

The Metropolis Light Transport algorithm is a variant of the classic Metropolis method used in statistical physics. A variance analysis of the Metropolis Light Transport algorithm is presented that bounds its variance in terms of the number of paths used and the intrinsic correlation between samples. It is shown that the variance of a pixel is $\Theta(1/N)$ where N is the number of samples for the entire image. The analysis uses basic probability, Bayes' law, and the principle of stationary distributions. This result implies that the presence of correlation in the algorithm does not make its asymptotic time complexity worse than uncorrelated Monte Carlo methods such as path tracing.

1 Introduction

The error analysis of rendering algorithms is beneficial for understanding their behavior [1,4]. Veach and Guibas [10] introduced a rendering algorithm based on the classic *Metropolis sampling algorithm* [7]. The *Metropolis Light Transport (MLT) algorithm* appears to work well in practice, especially for scenes with significant specular transport. However Veach and Guibas only showed the algorithm's performance on a small number of scenes, and no formal analysis of its behavior has been done. There is an abundance of literature on traditional Metropolis methods [11], however MLT has enough unique characteristics that it merits its own formal analysis. This paper attempts to prove the convergence properties of the MLT algorithm. This does not by itself yield immediate practical implications, but some of the tools used in our proofs may eventually prove to be useful for error estimation and reduction.

We examine the formal error behavior of MLT and show that the variance of a pixel estimate decreases as $\Theta(1/N)$ with respect to the number of paths N used for the solution [1] . This confirms our expectation that MLT's asymptotic behavior is similar to that of the traditional Metropolis methods but the result is expressed through the quantities directly related to properties of the images produces by MLT. Section 2 gives a brief overview of Metropolis sampling and MLT and points out the significant differences between the two methods. We then start with a much simplified analysis and progressively add in more components of MLT. The analysis relies on basic probability, Bayes' law, and the principle of stationary distributions [2]. This approach should help in the understanding of the algorithm and be much more obvious than a single, monolithic derivation of the final result. We close with a discussion of the results and a comparison with a standard Monte Carlo path tracing algorithm.

2 Comparison between Metropolis sampling and MLT

The Metropolis sampling algorithm was first described in a paper by Metropolis et al. [7]. The method generates a set of samples with distribution proportional to a function f, and does so with only the ability to evaluate f at points, as opposed to possessing any more general knowledge about f. Despite its power and simplicity, the Metropolis sampling has two major drawbacks. First, the generated samples are only guaranteed to asymptotically approach a distribution proportional to f. Therefore, the first K samples must be discarded so that the remaining samples are "near" enough to this asymptotic state that significant bias is not introduced. K is based on the particular distribution f and the mutation strategy used to generate successive samples. It is difficult to estimate K in advance and substantial trial and error is used to estimate an appropriate value [6]. Second, successive samples are correlated. So, for a fixed number of steps in a random walk, the variance is increased because of correlation between sample locations.

Veach and Guibas noted that the global illumination problem has a complicated function f defined over the set of paths in a scene. They also noted that the set of paths could be pared down to those originating at a light, passing through a lens, and finally hitting a sensor (e.g., a film plane). Each of these paths s has an associated energy density $f(s)$, and given a set of paths S with a distribution proportional to f, the relative pixel intensities can be determined by counting the paths in S that pass through each pixel (we assume a box pixel filter). In addition, Veach and Guibas introduced a method to eliminate the need to discard K initial samples used

[1] The $\Theta - notation$ asymptotically bounds a function from above and below [3]. For a given function $g(n)$, we denote by $\Theta(g(n))$ the set of functions

$$\Theta(g(n)) = \{f(n) : \exists\, c_1, c_2, n_0 > 0 \text{ s.t. } 0 \le c_1 g(n) \le f(n) \le c_2 g(n)\ \forall\, n \ge n_0\}$$

in traditional Metropolis algorithm. So MLT as defined by Veach and Guibas differs from traditional Metropolis sampling algorithm in two ways. It estimates pixel intensities by counting the number of paths in each pixel, effectively performing many integrations at once. Compare this with "normal" use of Metropolis sampling when the desired distribution is reached first and then observed quantities of a system are computed one by one, usually by drawing a new set of samples to compute each new one. Also, MLT uses a preprocessing step to allow every sample to be used, eliminating the need to discard the first K samples which might, in principle, affect the convergence rate. These changes mean that previous work on classic Metropolis sampling does not make it obvious what the error characteristics of MLT are.

Even more important is the fact that error analysis of traditional Metropolis algorithms usually uses quantities such as correlation time and it is not clear what these quantities should be for MLT. Since the fundamentals of MLT are the same as those of versions of Metropolis algorithm used in, for example, statistical physics, it would probably be possible to adapt known results from this area (see, for example, [8]). Doing so would, however, require making loose analogies and might not be straightforward. It also would provide little insight into some questions specific to the application of the Metropolis algorithm to global illumination, for example, how the convergence rate depend on the brightness of a particular pixel. For these reasons we instead chose to analyze error characteristics of the MLT algorithm from scratch using quantities which are, at least in principle, computable directly for a given image and mutation strategy. Examining these error characteristics is the subject of the rest of the paper.

3 Uncorrelated Analysis

In MLT the mutated paths eventually create a distribution proportional to the screen intensity. In this section we derive the variance of a pixel's intensity assuming we can draw uncorrelated samples in screen space whose density is proportional to screen luminance. In reality, the mutations create correlation between one path and the next. The correlation complicates the analysis and will be dealt with in the next section.

To simplify the analysis we assume that a box filter is used for each pixel value. For uncorrelated samples (no mutations), the MLT algorithm seeds screen space with samples having density proportional to the luminance on the film plane. The luminance estimate $\hat{L}$ for the ith pixel is thus:

$$L \approx \hat{L} = \frac{\overline{L}QX}{N},$$

where $\overline{L}$ is the average film-plane luminance, Q is the number of pixels, X is the

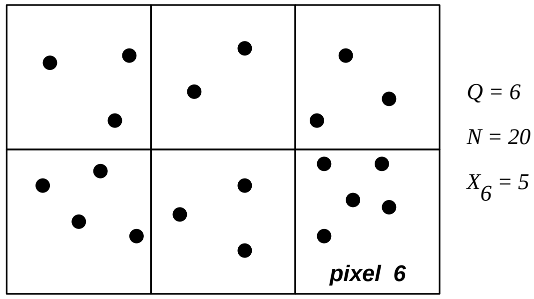

Fig. 1. *Some terms for analysis: Q is the number of pixels, N is the number of samples for the whole screen, and X is the number of samples in the ith pixel.*

number of samples in the pixel, and N is the total number of samples (Figure 1). This simply states that $\hat{L}$ is determined by counting the number of samples X in the ith pixel. All expressions refer to a chosen single pixel, so pixel index i is often not written. Note that the assumptions above are idealized and will not occur in practice; if we could generate samples proportional to screen luminance, we would not need MLT. However, this analysis is something we can build on later.

For a given pixel, we can view the contribution of each sample as a random variable X_j with Bernoulli distribution that is independent of and identically distributed to all other samples. It will contribute either 0 or 1 to the value of X:

$$X = \sum_{j=0}^{N-1} X_j \qquad X_j = \begin{cases} 1 & \text{with } P(\pi), \\ 0 & \text{with } P(1-\pi). \end{cases}$$

Where $\pi = \frac{L}{\overline{L}Q}$ is the fraction of the energy the pixel contains. From the properties of the Bernoulli distribution the variance of an individual sample's contribution to the count is:

$$V(X_j) = \pi(1-\pi) \leq \pi.$$

Because the X_j's are independent and identically distributed, the variance of the actual count X is:

$$V(X) = NV(X_0) = N\pi(1-\pi) \leq N\pi = N\frac{L}{\overline{L}Q}.$$

The variance of the actual estimate is thus:

$$V(\hat{L}) = \frac{\overline{L}^2 Q^2 V(X)}{N^2} \leq \frac{\overline{L}^2 Q^2 \pi}{N} = \frac{\overline{L}QL}{N}.$$

Therefore the variance for a pixel is $\Theta(1/N)$. However, the real MLT algorithm uses correlated samples, which is certain to raise variance. The question is how much, as will be examined in the next section.

4 Correlated Analysis

If we assume that we have a sample X_0 with the underlying distribution from the last section, and subsequent samples are generated via mutation, then samples are correlated. Thus $V(X)$ from the last section grows. This section examines how much it grows. We show that the variance for the correlated case is:

$$V(\hat{L}) \leq \frac{\overline{L}QL}{N}\{1 + \frac{2q}{1-q}\},$$

where q is related to the correlation between adjacent samples in a mutation chain. Note that for $q = 0$ we have the same answer as the uncorrelated analysis as we would expect. For a valid MLT implementation $0 \leq q < 1$ so correlated MLT is still $\Theta(1/N)$. The proof is an involved algebraic manipulation, and readers not interested in the details can skip to the next section.

Proof of variance bound for correlated samples

The variance of the sum of two random variables X_1 and X_2 is expressed as

$$V(X_1 + X_2) = V(X_1) + 2Cov(X_1, X_2) + V(X_2)$$

Definition 1 *A stochastic process X_t for $t \in T$, is said to be* strictly stationary *if the joint distribution functions of the families of random variables*

$$(X_{t_0+h}, X_{t_1+h}, \ldots, X_{t_{n-1}+h}) = (X_{t_0}, X_{t_1}, \ldots, X_{t_{n-1}})$$

are the same for all $h > 0$ and arbitrary selections t_0, t_1, ... , t_{n-1} of T.

This definition asserts that in essence the process is in probabilistic equilibrium and that the particular times at which we examine the process are of no relevance. In particular, the distribution of X_t is the same for each t.

Definition 2 *A stochastic process X_t for $t \in T$ is said to be* wide sense stationary *or* covariance stationary *if it possesses finite second order moments and if $Cov(X_t, X_{t+h}) = E(X_t X_{t+h}) - E(X_t)E(X_{t+h})$ depends only on h for all $t \in T$:*

$$Cov(X_t, X_{t+h}) = Cov(X_0, X_{0+h})$$

Theorem 1 *Let $X = X_0 + X_1 + \ldots + X_{N-1}$ where X_i are identically distributed but not necessarily independent random variables. If a process is wide sense sta-*

tionary and $R_k = Cov(X_i, X_{i+k}) \geq 0 \; \forall k$ then

$$V(X) = V(\sum_{i=0}^{N-1} X_i) \leq NV(X_0) + 2N \sum_{k=0}^{N-1} R_k. \tag{1}$$

Proof 1 *Let $\mu = E(X_0)$. Then*

$$E(\sum_{i=0}^{N-1} X_i) = N\mu$$

and

$$\begin{aligned} V(\sum_{i=0}^{N-1} X_i) &= E((\sum_{i=0}^{N-1} X_i - N\mu)^2) \\ &= E((\sum_{i=0}^{N-1} (X_i - \mu))^2) \\ &= E(\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (X_i - \mu)(X_j - \mu)) \\ &= \sum_{i=0}^{N-1} E((X_i - \mu)^2) + E(\sum_{i=0}^{N-1} \sum_{j=0, j\neq i}^{N-1} (X_i - \mu)(X_j - \mu)) \\ &= E((X_i - \mu)^2) = V(X_i) = V(X_0) \end{aligned}$$

For $j \neq i$, this reduces to:

$$\begin{aligned} E((X_i - \mu)(X_j - \mu)) &= E(X_i X_j - X_i \mu - X_j \mu + \mu^2) \\ &= E(X_i X_j) - 2E(X_0)\mu + \mu^2 = E(X_i X_j) - \mu^2 \\ &= Cov(X_i, X_j) \\ &\equiv R_{j-i} \end{aligned}$$

From the symmetry of the covariance $R_k = R_{-k}$. Rearranging the terms so than only positive $j - i$ are included, we get

$$V(X) = NV(X_0) + 2 \sum_{i=0}^{N-1} \sum_{j=i+1}^{N-1} R_{j-i}$$

In the last double sum, we have one R_{N-1} term, two R_{N-2} terms, three R_{N-3} terms, etc. Changing the index of summation to $k = j - i$ we finally get

$$V(X) = NV(X_0) + 2 \sum_{k=1}^{N-1} (N - k) R_k.$$

Inequality 1 immediately follows if the covariances R_k are non-negative. □

The task now is to give some bound for the sum of covariances when X is the sum of individual MLT sample contributions X_j to the luminance estimate L for the ith pixel. The first step is to express R_k for arbitrary k in terms of R_1, the correlation of adjacent MLT samples. The value of R_1 clearly depends on a particular mutation strategy chosen and will be considered to be an outside parameter for the analysis. We are interested in the adverse case of $R_1 \geq 0$. No strategy inducing negative correlation among MLT samples is currently known (if there were such a strategy, the sum in equation 1 would have been negative and variance $V(X)$ would have been less than in case of independent samples and would have grown at most as fast as $\Theta(N)$). We also will only consider the case of "converged" MLT, therefore we assume that the start-up bias has been completely eliminated (either by discarding a sufficient number of initial samples or by some other means) and sampling is proceeding according to distribution which is close to the stationary one. Reaching this state is necessary for any Metropolis algorithm to give valid results. It is generally difficult to tell how long this convergence would take for a given scene and mutation strategy. A more formal discussion of this problem can be found in [9].

As before, let $\pi = P(X_j = 1)$ be the (unconditional) probability for an MLT sample to contribute 1 to the value of X for ith pixel. Due to stationarity, this π depends only on the pixel index i but not on the sample index j within the sampling sequence.

To compute R_k we need $E(X_0 X_k)$. X_j has a Bernoulli distribution, which allows us to write for a given pixel:

$$E(X_0 X_k) = P(X_0 X_k = 1) = P(X_0 = 1) Pr(X_k = 1 | X_0 = 1) = \pi p_k \tag{2}$$

and

$$R_k = E(X_0 X_k) - E(X_0)E(X_k) = \pi p_k - \pi^2 = \pi(p_k - \pi) \tag{3}$$

where $p_k = P(X_k = 1 | X_0 = 1)$ (conditional probability that X_k will contribute 1 to the value of X given that X_0 has already contributed 1). Due to stationarity p_k depends only on the difference of the arguments, ($P(X_{i+k} = 1 | X_i = 1) = P(X_k = 1 | X_0 = 1)$), and that $p_k = p_{-k}$ due to time reversibility in the stationary state. Both of these facts will be used below.

Consider a sequence of MLT samples $X_0 X_1 \ldots X_m \ldots X_{k-1} X_k$ where X_m is any sample among $X_1 \ldots X_{k-1}$. Then

$$\begin{aligned} P(X_k = 1 | X_0 = 1) = & P(X_k = 1 | X_m = 1) P(X_m = 1 | X_0 = 1) + \\ & P(X_k = 1 | X_m = 0) P(X_m = 0 | X_0 = 1) \end{aligned}$$

or, using shorter notation and the fact that $P(X_m = 0 | X_0 = 1) = 1 - P(X_m = 1 | X_0 = 1) = 1 - p_m$:

$$p_k = p_{k-m} p_m + (1 - p_m) P(X_k = 1 | X_m = 0) \tag{4}$$

We compute $P(X_k = 1|X_m = 0)$ using Bayes' formula:

$$\begin{aligned} P(X_k = 1|X_m = 0) &= \frac{P(X_k = 1 \cap X_m = 0)}{P(X_m = 0)} \\ &= \frac{P(X_m = 0|X_k = 1)P(X_k = 1)}{1 - P(X_m = 1)} \\ &= \frac{(1 - p_{m-k})\pi}{1 - \pi} \end{aligned}$$

Substituting this into equation 4 and using $p_{m-k} = p_{k-m}$ we obtain a recursive formula for p_k through p_m and p_{k-m} where $m < k$:

$$\begin{aligned} p_k &= p_{k-m}p_m + (1 - p_m)(1 - p_{k-m})\frac{\pi}{1 - \pi} \\ &= \frac{p_{k-m}p_m + \pi(1 - p_{k-m} - p_m)}{1 - \pi} \end{aligned}$$

Let us consider even k's and let $m = k/2$. Then

$$\begin{aligned} p_k &= \frac{p_{k/2}^2 - 2\pi p_{k/2} + \pi^2 - \pi^2 + \pi}{1 - \pi} \\ &= \frac{(p_{k/2} - \pi)^2}{1 - \pi} + \pi \end{aligned}$$

We now substitute this result into the expression for R_k:

$$\begin{aligned} R_k &= E(X_0X_k) - E(X_0)E(X_k) \\ &= \pi(\frac{(p_{k/2} - \pi)^2}{1 - \pi} + \pi) - \pi^2 \\ &= \pi\frac{(p_{k/2} - \pi)^2}{1 - \pi} \end{aligned} \tag{5}$$

From equation 3 we have $p_{k/2} - \pi = R_{k/2}/\pi$ and we can further rewrite the expression for R_k as

$$R_k = \frac{R_{k/2}^2}{\pi(1 - \pi)} \tag{6}$$

This recursion allows R_k where k is an exact power of 2 to be expressed through R_1 as follows:

$$R_k = (\frac{1}{\pi(1 - \pi)})^{k-1}R_1^k = \pi(1 - \pi)q^k \tag{7}$$

where $q = R_1/(\pi(1-\pi))$. We have shown that R_k behaves as a geometric progression.

There are some consequences of equations 5 - 7 worth mentioning. First, to have $\lim_{k\to\infty} R_k = 0$ as required by general theory, we need $q < 1$ which at the first glance might be considered as a constraint on maximum acceptable value of R_1 in an MLT algorithm and, therefore, a limitation of MLT approach. This is not true, however: $q = 1$ corresponds to $R_k = \pi(1-\pi)$ for all k and, as follows from equation 5, requires $p_{k/2} = 1$ or $p_{k/2} = 2\pi - 1$. The second solution, namely $p_{k/2} = 2\pi - 1$, is greater than one, therefore, invalid. Since $p_{k/2}$ is a conditional probability, $p_{k/2} = 1$ means that once we have a contribution made to a pixel, all following samples are also contributing to the same pixel with probability 1. Clearly, any reasonable mutation strategy will be enough to avoid this pathological condition and, in fact, we should expect q to be substantially less than 1.

Second, from the equation 5 we can see that all covariances with $k' \geq 2k$ would go to zero if we could at some point during sampling create a sample in such a way that $p_k = \pi$. In particular, to have $R_k = 0$ for all k it is sufficient to have $R_1 = 0$. This corresponds to $P(X_k = 1|X_0 = 1) = Pr(X_k = 1)$, namely, independent sampling. The fact that our expressions show covariances go to zero only for independent samples is certainly something one would expect and provides an independent check for the formulae.

Finally, from the general theory of reversible Markov chains we have $R_k > R_{k+1}$ if $R_1 \geq 0$, i.e. correlation of the samples decreases monotonically. This means that although equation 7 is valid only for all k that are exact powers of 2 we can bound arbitrary R_k by the closest $R_{k'}$ where k' is an exact power of 2 and $k' \leq k$. For each $k' = 2^l$ there will be $2^{l+1} - 2^l = 2^l$ such k's. By doing that we obtain for the sum of covariances:

$$\sum_{k=1}^{N-1} R_k \leq \sum_{l=0}^{\lfloor \log_2(N-1) \rfloor} 2^l R_{2^l} = \sum_{\substack{m=1, \\ m \text{ powers of 2}}}^{N-1} mR_m < \sum_{m=1}^{\infty} mR_m = \frac{\pi(1-\pi)q}{(1-q)^2} \quad (8)$$

Inserting this result into equation 1 we finally obtain the bound written here in several different forms:

$$\begin{aligned} V(X) &\leq N\{V(X_0) + \frac{2\pi(1-\pi)q}{(1-q)^2}\} \\ &= NV(X_0)\{1 + \frac{2q}{(1-q)^2}\} \\ &= NV(X_0)\{1 + \frac{2R_1\pi(1-\pi)}{(\pi(1-\pi) - R_1)^2}\} \end{aligned} \quad (9)$$

Here we used the fact that $V(X_0) = \pi(1-\pi)$. Note that inequalities in 8 are rather loose. Intuitively, we would expect that for all k, and not just for exact powers of two, R_k behave close to a geometric progression (i.e. equation 7 is falid for all k). In this case the estimate improves to

$$V(X) \leq NV(X_0) + 2\pi(1-\pi)\{\sum_{k=0}^{N-1} q^k - 1\} = NV(X_0)\{1 + \frac{2q}{1-q}\} \quad (10)$$

Thus the variance of the actual count X is:

$$V(X) = N\pi(1-\pi)(1 + \frac{2R_1}{\pi(1-\pi) - R_1})$$

Recall that

$$L \approx \hat{L} = \frac{\overline{L}QX}{N},$$

So

$$V(\hat{L}) = \frac{\overline{L}^2 Q^2}{N^2} V(X) \leq \frac{\overline{L}QL}{N}\{1 + \frac{2q}{1-q}\},$$

It is comforting to see that this equation shows the same asymptotic convergence rate for MLT as that for traditional Metropolis algorithm [8], which is what we would expect given the fundamental similarities between the two. Note, however, that this result is expressed through quantities which can be computed for a given pixel using the details of the particular mutation strategy used.

5 Start-up weighting

In the actual MLT algorithm, Veach and Guibas choose an initial sample with probability density $p(x)$. The resulting luminance estimate for a pixel is then:

$$L \approx \hat{L} = \frac{L(X_0)}{p(X_0)} \frac{QX}{N},$$

In the case where $p \propto L$ this is the same as in previous sections. If we generate more samples via mutation (with only one seed X_0), then we will converge to an answer whose bias is determined by how close $p(x)$ is in shape to L. Since both of the fractions above are uncorrelated random variables, we need to use a relation for the product of uncorrelated random variables:

$$V(AB) = V(A)V(B) + V(A)E(B)^2 + V(B)E(A)^2.$$

So to apply to the above with A and B being the terms:

$$A = \frac{L(X_0)}{p(X_0)} \; E(A) = \overline{L} \tag{11}$$

$$B = \frac{QX}{N} \; E(B) = \frac{L}{\overline{L}} \tag{12}$$

A and B are uncorrelated because the start-up weighting is independent from consequent main MLT phase of the algorithm. We can conclude that $V(A)$ is a constant determined by the shape of $p(X_0)$, $E(A)$ is the constant $\overline{L}$, $V(B)$ is $\Theta(1/N)$ as shown in the last section, and $E(B)$ is $\frac{L}{\overline{L}}$. This implies clearly that the variance of $\hat{L}$ is $\Theta(1/N)$. In other words, the convergence rate of the overall algorithm (which includes both start-up weighting *and* the main MLT part), as expected, is not affected by the initialization phase.

6 Discussion

Our basic analysis has shown that MLT does not suffer an asymptotic disadvantage when compared to uncorrelated Monte Carlo methods such as path tracing. There are several other things to note:

Comparison with Monte Carlo path tracing

MLT has a rate of convergence of variance that is the same as MCPT: $V(X) = \Theta(1/N)$ where X is the pixel value and N is the number of paths. However, it should be noted that nowhere in our analysis of MLT did we have to consider parameters such as light source size. MCPT, on the other hand, has its time-constant heavily influenced by light source size. It is possible that in practice MLT also has this property but it is not indicated because our analysis is too conservative and somehow encodes the worst-case for light source size. However, this is not the case. Consider the example in Figure 2. There is a light source shining through a perfectly transmitting table, so the ground plane will have a smooth irradiance pattern except for the "shadow" caused by the glass plate edges.

For MCPT, if we halve the projected area of the light, we will make it half as likely to be hit by a diffuse reflection ray, and we will quadruple the variance. Thus as we make the MCPT variance arbitrarily large by making the light source smaller and smaller.

For MLT, however, our analysis indicates that if the image intensities are not significantly changed, and the correlations between samples is not significantly changed, then the variance cannot increase arbitrarily for small lights. Thus for some light source size, MLT will outperform MCPT. To be sure the correlations between samples is not significantly changed, we note that these two paths would be related to a "caustic mutation" in Veach and Guibas's implementation [10] and this mutation will not be highly sensitive to the light source size. In practice Veach and Guibas have shown that MLT can produce good images of caustics in some circumstances, but this analysis indicates that it is likely to perform more stably than MCPT whenever light sources are allowed to become small in the presence of specular surfaces. However, if a specular path is added that is difficult for a single mutation to handle,

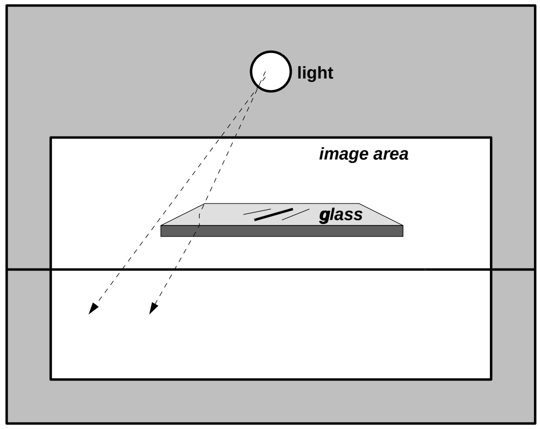

Fig. 2. Two nearby paths for a light shining through a plate of perfectly transmitting glass

such as a reflection of a caustic, the acceptance probability goes down and correlation goes up. So the sometimes unintuitive nature of mutations can mask the causes of increased variance. Future work on MLT should focus on mutation strategies that deal with adversarial cases.

Error estimation

For the uncorrelated case the variance of a pixel luminance can be written:

$$\sigma^2 \approx \frac{\overline{L}QL}{N},$$

where again Q is the number of pixels. and σ is the standard deviation (square root of variance). As the result of our correlated analysis shows, this approximation holds provided no single pixel has a majority of the energy for the whole image. Since standard deviation is usually a reasonable estimate of absolute error we see that there is some built-in stability in the error; it goes down as L goes down. So for dim pixels where our sensitivity to noise improves, the noise is of smaller amplitude. If we express the pixel luminance L as a scale C times $\overline{L}$, and we manipulate the equation so that it is in terms of the samples per pixel N/Q we have:

$$\frac{N}{Q} \approx \frac{C\overline{L}^2}{\sigma^2}$$

If we apply “tone mapping” to make this image displayable on a monitor where the maximum displayable intensity is 1.0, then the average luminance $\overline{L}$ will typically be mapped to some value between 0.1 and 0.5. A standard error of 0.02 would be considered a very good image (the images shown by Veach and Guibas are almost certainly noisier). Finally, we only need to worry about the error in pixels that didn’t exceed 1.0 in value because those especially bright pixels would “burn out” and a small error in their values will not change the displayed image at all. Assuming the average luminance maps to 0.2, we have:

$$\frac{N}{Q} \approx \frac{C(0.2)^2}{(0.02)^2} = 100C.$$

Since the brightest pixels attain value 1.0, the biggest value of C is 5, indicating that we would need 500 samples per pixel.

What one should note is that 500 samples per pixel is an impressively small number for a “difficult” scene (for traditional raytracing one would need close to this number of samples even for a simplest scene, such as Cornell box), and that our numbers above are probably too conservative. For example, assuming a standard error of 0.04 and mapping the average to 0.15 yields about 90 samples per pixel. Due to correlation between samples the actual MLT algorithm will be worse than that. However, even for a large correlation between samples, say $q = 0.9$, we would need at most 20 times as many samples. So it is perhaps not surprising that Veach and Guibas reported such low sampling rates.

Start-up issues

As discussed earlier, if we use a single initial sample to mutate from in MLT we will converge to a possibly incorrect answer. If we want to converge to the correct answer (i.e. have an unbiased algorithm) we need to use a number of seeds that increases as the total number of samples increases. In other words, one run of the MLT algorithm with one initial sample will converge to a variance-free image that is off by some multiplier whose expected value is one but whose actual value will be different from one. So many of these images averaged together would converge to the correct image. If the total number of samples is $N = SM$ where S is the number of seeds and M is the number of paths mutated from each seed, the averaging process will yield variance be $\Theta(1/(SM))$. Because SM is just the total number of samples, the entire unbiased process that uses an increasing number of initial seeds is $\Theta(1/N)$.

Correlation between pixels

So far we have only considered MLT behavior at a single pixel. Veach and Guibas describe three different classes of mutations, bidirectional mutations, lens mutations, and perturbations. The bidirectional mutations either contribute to the same pixel (lens edge not mutated, or mutation not accepted) or contribute to a random pixel if accepted. The lens mutations cause the next path to be through a random pixel. However, the perturbations, especially the lens perturbations, tend to move the image point to a nearby point. This is due to the relatively small changes that are made to the paths. Given this, the mutation strategy has a greater probability of creating a new path passing through a nearby pixel than a distant pixel. This means that there is some positive correlation between nearby pixels. This correlation should have some effect on the noise characteristics of unconverged solutions. The exact effect of the correlation will be determined by the mutation strategy and the parameters of that strategy, as well as by the specific model being rendered. In the images of Veach and Guibas, this correlation seems to cause noise of a lower frequency than tradition Monte Carlo rendering algorithms. Perhaps it is possible to find a mutation strategy that pushes the frequency of the noise into a range where it is masked by features of the image [5].

Acknowledgements

Thanks to Eric Veach for helpful discussions about his work and access to uncompressed images generated by his algorithm. This work was supported by NSF award 97-20192.

References

[1] ARVO, J., TORRANCE, K., AND SMITS, B. A framework for the analysis of error in global illumination algorithms. In *Proceedings of SIGGRAPH '94 (Orlando, Florida, July 24–29, 1994)* (July 1994), A. Glassner, Ed., Computer Graphics Proceedings, Annual Conference Series, ACM SIGGRAPH, ACM Press, pp. 75–84. ISBN 0-89791-667-0.

[2] BILLINGSLEY, P. *Probability and Measure*. John Wiley and Sons, 1995.

[3] CORMEN, T., LEISERSON, C., AND RIVEST, R. *Introduction to Algorithms*. MIT Press, Cambridge, 1990.

[4] DUTRÉ, P. *Mathematical Frameworks for Global Illumination*. PhD thesis, Katholieke Universiteit Leuven, 1996.

[5] FERWERDA, J. A., PATTANAIK, S. N., SHIRLEY, P., AND GREENBERG, D. P. A model of visual masking for computer graphics. In *SIGGRAPH 97 Conference Proceedings* (Aug. 1997), T. Whitted, Ed., Annual Conference Series, ACM SIGGRAPH, Addison Wesley, pp. 143–152. ISBN 0-89791-896-7.

[6] KALOS, M. H., AND WHITLOCK, P. A. *Monte Carlo Methods*. John Wiley and Sons, 1986.

[7] METROPOLIS N., ROSENBLUTH A. W., ROSENBLUTH M. N., TELLER A. H., TELLER E. Equations of state calculations by fast computing machines. *J. Chem. Phys. 21* (1953), 1087.

[8] NEWMAN, M., AND BARKEMA, G. *Monte Carlo Methods in Statistical Physics*. Clarendon Press, 1999.

[9] SZIRMAY-KALOS, L., DORNBACH, P., AND PURGATHOFER, W. On the start-up bias problem of metropolis sampling. In *WSCG'99 - The 7-th International Conference in Central Europe on Computer Graphics* (Feb. 1999), V. Skala, Ed., Univ.of West Bohemia Press. ISBN 80-7082-490-5.

[10] VEACH, E., AND GUIBAS, L. J. Metropolis light transport. In *SIGGRAPH 97 Conference Proceedings* (Aug. 1997), T. Whitted, Ed., Annual Conference Series, ACM SIGGRAPH, Addison Wesley, pp. 65–76. ISBN 0-89791-896-7.

[11] WOOD W. W. AND ERPENBECK J. J. Molecular dynamics and monte carlo calculations in statistical mechanics. *Ann. Rev. Phys. Chem. 27* (1976), 319.