Eurographics Symposium on Rendering (2006)
Tomas Akenine-Möller and Wolfgang Heidrich (Editors)

Sequential Sampling for Dynamic Environment Map Illumination

Abhijeet Ghosh, Arnaud Doucet, and Wolfgang Heidrich

The University of British Columbia†

Abstract

Sampling complex illumination in the form of environment maps has received a lot of attention in computer graphics. Recent work in this area has demonstrated that drawing samples from the product of light and BRDF produces superior results to other sampling strategies. However, existing methods in this area consider only individual frames, and do not take advantage of coherence in animations. In this paper, we introduce a sequential sampling approach for dynamic environment map illumination. Our algorithm efficiently samples from the product of illumination and BRDF, while exploiting temporal coherence. We demonstrate significant performance benefits over existing methods.

Keywords: Methods and Applications – Monte Carlo Techniques; Rendering – Ray Tracing; Rendering – Global Illumination.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [COMPUTER GRAPHICS]: Three-Dimensional Graphics and Realism, Raytracing.

1. Introduction

The sampling of complex direct illumination in the form of high dynamic range (HDR) environment maps has received significant attention in recent years, with major applications in realistic relighting. The best known techniques for direct illumination sample from the product of the incident illumination and the surface reflectance [BGH05, CJAMJ05, TCE05]. With advancements in HDR acquisition technologies, HDR video environments are becoming increasingly available. This availability has spawned recent work on sampling dynamic illumination from such HDR video sequences [HSK*05, WWL05]. However, these techniques only take the dynamic importance of the illumination into account while proposing samples for the video sequence. Such techniques are problematic in the presence of high frequencies in both the illumination as well as the surface BRDFs. While one can produce noise-free images by using one set of samples for all surface locations, doing so eliminates the impact of dimmer light sources, which should dominate the reflection for certain surface orientations.

In this work, we aim to efficiently sample from the product distribution of the illumination and the BRDF in a video sequence with dynamic illumination using a *Sequential Monte Carlo (SMC)* sampling strategy. The basic idea is to generate samples according to the product distribution in the first frame of the sequence, and thereafter to filter these samples (particles) in the subsequent frames according to the dynamic product distribution. This sequential sampling mechanism is more efficient than independently re-regenerating the samples for every time step (Figure 1), especially for scenes with high frequencies in both the dynamic illumination and the BRDF. At the same time, our method avoids systematic under-estimation of reflections at certain angles, which is common to dynamic importance methods generating point light approximations of the environment.

Our solution to sampling from the dynamic product dis-

† E-mail:{ghosh, arnaud, heidrich}@cs.ubc.ca

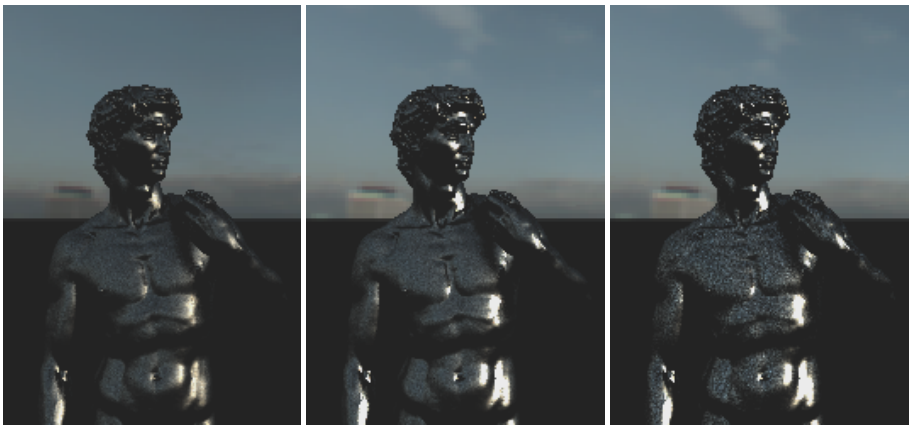

Figure 1: *Quality comparison of our SMC sampling algorithm with bidirectional importance sampling for a sky probe sequence. Left:* 1^{st} *frame rendered using bidirectional importance sampling in* 8 *seconds. Center:* 5^{th} *frame of sequence rendered using SMC sampling in* 4 *seconds. Right: Comparison image for* 5^{th} *frame generated using bidirectional importance sampling in* 4 *seconds.*

tribution is a two-step approach. We assume that we have already obtained a sample set according to the product distribution of the previous frame. For the first frame of the animation, we generate this sample set with bidirectional importance sampling [BGH05]. In the first step of our algorithm, samples distributed according to the product distribution of the previous time step are propagated in time using sequential importance sampling. The product distribution at the new time step is incrementally estimated using the weights of the sequential importance. The second step extends the path of each of these samples using a Markov Chain Monte Carlo (MCMC) kernel whose invariant distribution is the target distribution at the current time step. The MCMC kernel is implemented using the Metropolis-Hastings (MH) algorithm [MRR*53]. No visibility tests are performed during either of the two steps. Visibility is finally tested at the end of the second step in order to obtain a Monte Carlo estimate of the direct illumination. This approach has the following benefits:

- We need to propagate only a small number of samples in time to estimate the direct illumination as these samples are distributed according to the target distribution at each time step. This makes sample propagation very efficient.
- The normalization constant for the product distribution at each time step can be incrementally computed using the sequential importance weights. Thus, the normalization constant, i.e. the un-occluded reflected radiance at each time step can be estimated very efficiently without drawing a large number of samples.
- Sample generation cost at each time step is independent of the cost of sampling from the BRDF representation since the algorithm only requires to evaluate the BRDF but does not require to sample from it. Thus, any complex BRDF representation can be used without impacting sample generation cost.
- The method creates samples on the fly and does not require any expensive precomputation.

2. Related Work

The computation of the direct illumination in a scene is a costly task, especially in the presence of complex real-world light sources such as environment maps and other image-based representations. Much effort has been focused on the development of efficient techniques for completing this task in the presence of static as well as dynamic illumination.

2.1. Sampling from Environment Maps and BRDFs

The easiest way of dealing with this situation is by sampling from either the environment map or the BRDF, but not both. To sample from the energy distribution of the environment, researchers have used point relaxation schemes [CD01, KK03], hierarchical distribution schemes [ODJ04, Deb05], or inversion of the cumulative density function (CDF) [LRR05, SHS02]. Agarwal et al. [ARBJ03] introduced a sampling method for environment maps that takes into account both the energy distribution in the environment map and the solid angle separating the samples to avoid oversampling of bright regions.

All these methods work best for high-frequency illumination, but low-frequency BRDFs. In cases where the BRDF contains high frequencies, but the illumination does not, sampling form the BRDF reduces the variance. Depending on the BRDF representation, different strategies can be used, for example for analytical BRDFs (e.g. [Shi00]), or tabulated BRDFs [MH97, LRR04].

When both the environment and the BRDF contain high frequencies, neither of these methods works very well. Veach and Guibas [VG95] were the first to consider this problem. Their multiple importance sampling combined different sampling distributions such as illumination and BRDF distributions using *balance* heuristics.

However, as shown recently, sampling directly from the product of light and BRDF produces even better results. Several strategies have recently been introduced to this end. Burke et al. [BGH05] present two algorithms for sampling according to the product distribution, one based on rejection sampling and the other based on sampling-importance-resampling (SIR), the latter of which is also used by Talbot et al. [TCE05]. Clarberg et al. [CJAMJ05] present a technique for efficiently sampling the product of the illumination and the BRDF using a hierarchical wavelet representation. Their method is very efficient for tabulated BRDFs but requires significant precomputation for environment maps making it unsuitable for dynamic illumination. Lawrence et al. [LRR05] present an approach for compressing cumulative density functions for efficient inversion and apply it to sampling from many precomputed environment map PDFs for different surface orientations. This can be seen as an approximation of the product distribution.

Our proposed SMC mechanism for sampling dynamic illumination is very general and can be applied in combination with any sampling scheme discussed above for proposing samples in the first time step. In fact, sequential sampling would also help overcome the precomputation requirements of some of these techniques for a dynamic sequence. In our work, we employ Burke et al.'s SIR algorithm for bidirectional importance sampling [BGH05] in the first frame of the sequence.

2.2. Sampling from Dynamic Illumination

Several researchers have considered the problem of sampling from dynamic illumination. For example, Sbert et al. [SSSK04] presented a method for reusing light paths computed in one frame of a light animation sequence in all other frames using multiple importance sampling. The indirect illumination in each frame of the sequence is approximated as weighted contributions from these precomputed virtual point lights. This method works on moving point lights, not complex environments.

Some researchers have looked at path re-usage in the context of global illumination for a sequence of camera animation via reprojection of the primary ray hits on to the image plane. Here, techniques that are possibly biased [HDMS03] as well as unbiased techniques [MFSSK06] using multiple importance sampling have been proposed for the path re-usage. Our proposed SMC sampling mechanism can also be applied to a camera animation sequence for sample propagation in an unbiased manner, while being more efficient than multiple importance sampling.

Recent work on dynamic environment maps, including Wan et al. [WWL05] and Havran et al [HDS03], approximates environments with a set of point lights that are drawn according to the energy distribution in the environment map, and evolve smoothly over time. However, this procedure introduces a systematic error for specular materials if none of the chosen point lights resides inside the specular lobe. In this case, one would expect to see a specular reflection of a dimmer part of the environment, but these methods cannot produce this result.

2.3. Metropolis Sampling for Global Illumination

Our method includes a Metropolis sampling strategy. Veach & Guibas [VG97] first applied Metropolis sampling to the problem of image synthesis and developed a general, robust and unbiased algorithm called Metropolis Light Transport (MLT) that was well suited for hard cases for sampling because of its localized exploration and path re-usage properties. Fan et al. [FCL05] recently applied the Metropolis algorithm for efficiently sampling coherent light paths for photon mapping.

Cline et al. [CTE05] presented an efficient unbiased method to solve correlated integral problems with a hybrid algorithm that uses Metropolis sampling-like mutation strategies in a standard Monte Carlo integration setting, overcoming the startup bias problem of MLT. They apply energy redistribution over the image plane to reduce variance of path tracing for global illumination. Our work is similar in spirit in the sense of using initial Monte Carlo sampling for proposing samples for the subsequent time step, followed by Metropolis sampling in order to explore the new direct illumination target distribution.

3. Sequential Monte Carlo Sampling

As mentioned in the introduction, we propose a sequential Monte Carlo (SMC) sampling algorithm for sampling according to the dynamic product distribution of the illumination and the surface reflectance during an animation sequence. Traditional SMC algorithms in the literature deal with the case where the target distribution of interest at time n, defined on Ω_n, is of a higher dimension than the target distribution at time $n-1$ [DdFG01]. A classical example of this is an SMC algorithm applied to sequential Bayesian inference. In our case, the target distribution at every time step, i.e. the direct illumination integral, is defined on a common space of the hemisphere of directions Ω. Hence, we employ a class of SMC samplers recently developed for a common domain [dMDJ06] to the problem of sampling from the product distribution of incident illumination and BRDF in the presence of dynamic illumination.

Our SMC sampling algorithm is a two-step approach: we start with samples created according to the product distribution in the previous time step, and propagate them in time

using sequential importance sampling followed by a possible resampling step. We then employ an appropriate MCMC transition kernel to redistribute the samples according to the product distribution at the new time step. Thereafter these samples, now distributed according to the new target distribution, are used for visibility testing.

Consider the direct illumination at a point for a given observer direction ω_r:

$$L_r(\omega_r) = \int_\Omega f_r(\omega_i \rightarrow \omega_r) \cos\theta_i L_i(\omega_i) V(\omega_i) d\omega_i, \quad (1)$$

with L_i denoting the incident illumination from an environment map, f_r representing the BRDF, and V being the binary visibility term. Note that we treat L_i and f_r as scalar-valued functions here and throughout the text. In practice, L_i and f_r are color-valued functions, from which we derive the scalar-valued ones by averaging the color channels.

The target distribution of interest for direct illumination is the product distribution p of incident illumination and the BRDF:

$$p(\omega_i) := \frac{f_r(\omega_i \rightarrow \omega_r)\cos\theta_i L_i(\omega_i)}{\int_\Omega f_r(\omega_i \rightarrow \omega_r)\cos\theta_i L_i(\omega_i) d\omega_i} = \frac{\tilde{p}(\omega_i)}{L_{ns}}. \quad (2)$$

Here, $\tilde{p}(\omega_i) = f_r(\omega_i \rightarrow \omega_r)\cos\theta_i L_i(\omega_i)$ is the un-normalized importance function, and $L_{ns} = \int_\Omega f_r(\omega_i \rightarrow \omega_r)\cos\theta_i L_i(\omega_i) d\omega_i$ is the un-occluded reflected radiance in the viewing direction and is the normalization constant of the target distribution. Burke et al. [BGH05] refer to L_{ns} as *"radiance, no-shadows"*.

Our SMC algorithm works as follows: we start with samples $\omega_{i,n-1}^{(j)}$ and weights $W_{n-1}^{(j)}, j = 1,...,N$, such that the weighted samples are proportional to the product distribution of BRDF and illumination in frame $n-1$. These samples represent the incident light directions for one surface point, that is, the point visible at a specific pixel. For the first frame, these samples are obtained by bidirectional importance sampling, and all weights are $1/N$. The two steps of our method are then *sample propagation* followed by *MCMC transition* to adjust the samples to the product distribution in the next frame.

Step 1: Sample Propagation: We propogate the samples $\omega_{i,n-1}^{(j)}$ in time using sequential importance sampling. The un-normalized *incremental* weight $\tilde{w}_n$ of every sample for sequential importance at time n is given by the following ratio:

$$\tilde{w}_n^{(j)} = \frac{\tilde{p}_n(\omega_{i,n-1}^{(j)})}{\tilde{p}_{n-1}(\omega_{i,n-1}^{(j)})}, \quad (3)$$

This weight is just the ratio of the target function evaluated at the sample point at time n to that evaluated at time $n-1$ for the same point, and represents the change in weighting of a sample due to changes in the target distribution. The normalized weights for the N samples at time n are then given by:

$$W_n^{(j)} = \frac{W_{n-1}^{(j)} \cdot \tilde{w}_n^{(j)}}{\sum_{k=1}^N W_{n-1}^{(k)} \cdot \tilde{w}_n^{(k)}}. \quad (4)$$

This tracking of weights to represent evolving target distributions is called *Sequential Monte Carlo Sampling*, or SMC for short [dMDJ06]. Note that we can use the SMC mechanism for sampling from a dynamic sequence even if we use a different proposal distribution q instead of p for the first frame. The only difference here would be that we would have to appropriately weight the samples of the first frame for sequential importance. The un-normalized weights $\tilde{w}_1^{(j)}$ would then need to be computed as:

$$\tilde{w}_1^{(j)} = \frac{\tilde{p}_1(\omega_{i,1}^{(j)})}{q_1(\omega_{i,1}^{(j)})},$$

and then normalized to obtained $W_1^{(j)}$.

Step 1a: Resampling. As the variance between the proposal distribution q_n and the target distribution p_n tends to increase with n, the variance of the un-normalized importance weights tends to increase resulting in a potential degeneracy of the particle approximation. This degeneracy can be measured using the criterion of *effective sample size* (ESS) $(\sum_{j=1}^N (W_n^{(j)})^2)^{-1}$ [LC98]. The ESS takes values between 1 and N. If the ESS is below a pre-specified threshold, say $N/2$, we resample the N samples according to the weights $W_n^{(j)}$ and set the weights of the resampled samples equal to $1/N$. This resampling step discards samples with low weights while copying the samples with high weights multiple times, thus keeping the samples according to the target distribution. Note that the mutations in the next step make sure we do not keep multiple identical samples.

Step 2: MCMC Transitions. After sample propagation and potential resampling, we apply an MCMC kernel $K_n(\omega_i, \omega_i')$ of invariant distribution p_n to every sample $\omega_{i,n-1}^{(j)}$ in order to obtain new samples $\omega_{i,n}^{(j)}$. The new samples $\omega_{i,n}$ are marginally distributed according to

$$p_n(\omega_i') = \int_\Omega p_{n-1}(\omega_i) K_n(\omega_i, \omega_i') d\omega_i. \quad (5)$$

We employ the Metropolis-Hastings algorithm (MH) for these transitions, with a mix of local random walk moves and independent proposal moves. For a detailed description of the MH algorithm, we refer the reader to the chapter

Symbol	Description
$\omega_i^{(j)}$	j^{th} sample (incident direction)
$W_n^{(j)}$	Weight for j^{th} sample in frame n
$\tilde{w}_n^{(j)}$	Unnormalized weight for j^{th} sample in frame n
N	Number of samples per pixel
p_n	Target PDF for frame n
$\tilde{p}_n$	Unnormalized importance (product of illumination and BRDF)
q_n	Proposal distribution at time n
$p_{n-1:n}^{k/P}$	The k^{th} of P intermediate distributions between frame $n-1$ and frame n

Table 1: *Summary of notation used in this paper.*

on Metropolis Sampling by Pharr in the SIGGRAPH 2004 course notes [DJA*04]. When using the MH algorithm, the MCMC kernel K_n of invariant distribution p_n is described in terms of an acceptance probability of a proposed transition:

$$a(\omega_i^{(j)} \rightarrow \omega_i'^{(j)}) = \min\{1, \frac{\tilde{p}_n(\omega_i'^{(j)}) q(\omega_i'^{(j)} \rightarrow \omega_i^{(j)})}{\tilde{p}_n(\omega_i^{(j)}) q(\omega_i^{(j)} \rightarrow \omega_i'^{(j)})}\}, \quad (6)$$

where $\omega_i^{(j)}$ is the current sample, $\omega_i'^{(j)}$ is the proposed sample using the transition kernel q, and a is the acceptance probability of the proposed transition.

We mix local walk moves with independent proposal moves as these independent moves are required to prevent the SMC samples from getting stuck in local possibly narrow modes of the target distribution. For high frequency dynamic lighting, we choose local walk moves with uniform random directional perturbations of up to a few degrees of the samples, while choosing samples from the environment map (EM) for the independent proposal moves. When using local random walk moves, there is an equal probability of transition between $\omega_i^{(j)}$ and $\omega_i'^{(j)}$, i.e. $q(\omega_i^{(j)} \rightarrow \omega_i'^{(j)}) = q(\omega_i'^{(j)} \rightarrow \omega_i^{(j)})$. Thus, the acceptance probability a_{local} of the local walk moves is given by:

$$a_{local}(\omega_i^{(j)} \rightarrow \omega_i'^{(j)}) = \min\{1, \frac{\tilde{p}_n(\omega_i'^{(j)})}{\tilde{p}_n(\omega_i^{(j)})}\}. \quad (7)$$

When using independent proposal moves from the EM, the transition probability of the proposed sample is given by $q(\omega_i^{(j)} \rightarrow \omega_i'^{(j)}) = L_{i,n}(\omega_i'^{(j)}) / \int_\Omega L_{i,n} d\omega_i$. Hence the acceptance probability of the independent move is given by:

$$\begin{aligned} a_{EM}(\omega_i^{(j)} \rightarrow \omega_i'^{(j)}) &= \min\{1, \frac{\tilde{p}_n(\omega_i'^{(j)}) L_{i,n}(\omega_i^{(j)}) / \int_\Omega L_{i,n} d\omega_i}{\tilde{p}_n(\omega_i^{(j)}) L_{i,n}(\omega_i'^{(j)}) / \int_\Omega L_{i,n} d\omega_i}\} \\ &= \min\{1, \frac{f_r(\omega_i'^{(j)} \rightarrow \omega_r) \cos\theta_i'^{(j)}}{f_r(\omega_i^{(j)} \rightarrow \omega_r) \cos\theta_i^{(j)}}\}. \end{aligned} \quad (8)$$

In practice, we propose several MCMC moves per sample for good exploration of the target distribution.

Note the SMC algorithm as described above is unbiased: Step 1 corresponds to an importance sampling step, which produces the correct distribution at time n using the distribution at time $n-1$ as importance. The variance of this distribution is reduced by applying the MCMC algorithm in Step 2. Since MCMC algorithm works on an unbiased distribution, startup-bias is avoided. A formal argument can be found in [dMDJ06].

3.1. Direct Illumination Estimate

With the sample sets and weights derived above, we can now estimate the reflected radiance at a surface location as

$$L_{N,n,smc}(\omega_r) = L_{ns,n} \sum_{j=1}^{N} W_n^{(j)} \cdot V(\omega_{i,n}^{(j)}). \quad (9)$$

Equation 9 can be interpreted as the scaling of the unoccluded reflected radiance $L_{ns,n}$ by the weighted average of N visibility tests performed along the directions contributing most significantly to the radiance at time n. Here, the normalization constant $L_{ns,n}$ at time n can be incrementally estimated as

$$\frac{L_{ns,n}}{L_{ns,n-1}} = \frac{\int_\Omega \tilde{p}_n(\omega_{i,n-1}) d\omega_{i,n-1}}{\int_\Omega \tilde{p}_{n-1}(\omega_{i,n-1}) d\omega_{i,n-1}} \approx \sum_{j=1}^{N} W_{n-1}^{(j)} \cdot \tilde{w}_n^{(j)}. \quad (10)$$

The derivation of this result is given in Appendix A. It is worth pointing out that this incremental estimate of the normalization constant $L_{ns,n}$ at time n according to Equation 10 provides the crucial advantage for the SMC algorithm in terms of computational expense over a pure MC approach such as bidirectional sampling where the proper estimate of L_{ns} requires drawing large number of samples. To estimate $L_{ns,n}/L_{ns,1}$, one can use the product of estimates given by Equation 10 from time $t = 2$ to n. $L_{ns,1}$ is estimated in our case during bidirectional sampling for the first time step.

A summary of the method can be found in Algorithm 1. The complexity of this algorithm is in $O(N)$.

Algorithm 1 SMC Sampling for Dynamic Illumination

1: INITIALIZATION
- Set $n = 1$.
- For $j = 1,...,N$ draw $\omega_{i,1}^{(j)} \sim p_1$ using bidirectional sampling and set $W_1^{(j)} = 1/N$.

Iterate steps 2 and 3:

2: WEIGHTING AND RESAMPLING
- Set $n = n+1$.
- Compute new weights $W_n^{(j)}$ according to Equations 3 and 4.
- If $ESS < Threshold$, resample and set $W_n^{(j)} = 1/N$.

3: SAMPLING
- For $j = 1,...,N$ draw $\omega_{i,n}^{(j)} \sim K_n(\omega_{i,n-1}, \omega_{i,n})$.
- Estimate $L_{ns,n}$ according to Equation 10.
- Estimate reflected radiance according to Equation 9.

4. Variance Reduction with Intermediate Distributions

The aim of the SMC sampling algorithm as discussed in Section 3 is to "smoothly" move samples from the target distribution at time $n-1$ to the target distribution at time n. More formally, we have samples

$$\omega_{i,n-1}^{(j)} \sim p_{n-1}(\omega_i) = \frac{\tilde{p}_{n-1}(\omega_i)}{L_{ns,n-1}},$$

and we want to move towards samples

$$\omega_{i,n}^{(j)} \sim p_n(\omega_i) = \frac{\tilde{p}_n(\omega_i)}{L_{ns,n}}.$$

This transition is smooth under the assumption that $p_{n-1} \approx p_n$. However, this may not be true in practice, especially in the case where the dynamic illumination is in the form of a high frequency HDR video environment. If the discrepancy between the two successive distributions is too high, this will result in high variance in the un-normalized incremental weights $\tilde{w}_n^{(j)}$, and thus indirectly result in high variance in the normalized weights $W_n^{(j)}$. The variance in $W_n^{(j)}$ can be countered with the resampling step after sequential importance sampling, resulting in good estimates of the posterior $\tilde{p}_n$. However, the resampling step does not affect the variance in the un-normalized weights $\tilde{w}_n^{(j)}$, which can lead to high variance in the incremental estimate of the normalization constant $L_{ns,n}$ according to Equation 10.

In this scenario, we introduce a sequence of intermediate distributions [GM98] between the original distribution p_{n-1} and the new one p_n in order to select a smooth transition that the sample can follow. These intermediate distributions are blends of the original distributions:

$$p_{n-1:n}^{\gamma}(\omega_i) \propto \tilde{p}_{n-1:n}^{\gamma}(\omega_i) = [\tilde{p}_{n-1}(\omega_i)]^{1-\gamma}[\tilde{p}_n(\omega_i)]^{\gamma}, \quad (11)$$

such that

$$p_{n-1:n}^{0}(\omega_i) = p_{n-1}(\omega_i),\ p_{n-1:n}^{1}(\omega_i) = p_n(\omega_i).$$

In practice, we introduce P discrete intermediate distributions: $p_{n-1:n}^{k/P}(\omega_i)$, where $k = 1,...,P$. We can use these new distributions to reduce variance with little additional cost. The idea is to reduce the variance in the incremental weights $\tilde{w}_n$ by computing them as a product of P incremental weights $\tilde{w}^k$ of these intermediate distributions. The consecutive intermediate distributions $p_{n-1:n}^{k/P}(\omega_i)$ and $p_{n-1:n}^{(k-1)/P}(\omega_i)$ are closer to each other by construction than $p_{n-1}(\omega_i)$ is to $p_n(\omega_i)$, resulting in flatter weights $\tilde{w}^k$, as compared to $\tilde{w}_n$.

The SMC algorithm with P intermediate distributions requires slight modifications to the algorithm discussed in Section 3. Instead of first computing the un-normalized weights $\tilde{w}_n^{(j)}$ and the normalized weights $W^{(j)}$ and then doing the MCMC transitions, the algorithm for P intermediate distributions computes the un-normalized weights $\tilde{w}_n^{(j)}$ as a product of P intermediate un-normalized weights $\tilde{w}^{k,(j)}$ that each involve an MCMC kernel of invariant distribution $p_{n-1:n}^{k/P}$. Here, the intermediate un-normalized weights $\tilde{w}^{k,(j)}$ are computed as:

$$\tilde{w}^{k,(j)} = \frac{\tilde{p}_{n-1:n}^{k/P}(\omega_i^{k-1,(j)})}{\tilde{p}_{n-1:n}^{(k-1)/P}(\omega_i^{k-1,(j)})}. \quad (12)$$

Assuming we have samples $\{W_{n-1}^{(j)}, \omega_{i,n-1}^{(j)}\}$ approximating p_{n-1}. The algorithm then proceeds as follows:

Algorithm 2 SMC with Intermediate Distributions

1: INITIALIZATION:
- We write $\omega_i^{0,(j)} = \omega_{i,n-1}^{(j)}$ and set $\tilde{w}_n^{(j)} = 1$.

2: ITERATION: for $k = 1,...,P$
- Compute $\tilde{w}^{k,(j)}$ according to Equation 12.
- Set $\tilde{w}_n^{(j)} = \tilde{w}_n^{(j)} \cdot \tilde{w}^{k,(j)}$.
- Sample $\omega_i^{k,(j)} \sim K_k(\omega_i^{k-1,(j)}, \omega_i'^{k-1,(j)})$ of invariant distribution $p_{n-1:n}^{k/P}(\omega_i)$.

At the end of the P iterations of intermediate distributions, the normalized weights $W_n^{(j)}$ are still computed according to Equation 4, and resampled if the ESS is below the prespecified threshold. Finally, the normalization constant $L_{ns,n}$

and the direct illumination estimate are computed according to Equations 10 and 9 respectively.

In general, there is greater benefit in terms of variance reduction in the estimate of the target distribution with the introduction of a sequence of P intermediate distributions involving one MCMC move each than with a single distribution involving P MCMC moves, while being only a slightly more expensive in terms of computation time (Figure 3 in the results section). This benefit is because a sequence of intermediate distributions simultaneously reduces the variance in the un-normalized weights $\tilde{w}_n^{(j)}$ while exploring the target distribution at time n.

5. Unoccluded Illumination Estimate with Path Sampling

In this section, we explore an alternative estimate for the unoccluded radiance L_{ns}, which can be used in place of the method described in section 3.1. We obtain this alternate solution by path sampling [GM98]. In the statistics literature, the path of a sample is defined as the continuous trajectory of a sample over time. It should not be confused with light paths in classical global illumination literature. With this definition of path, *path sampling* refers to smoothly moving samples from one distribution to the next.

Considering a continuous path of distributions

$$p^{\gamma}_{n-1:n}(\omega_i) = \frac{\tilde{p}^{\gamma}_{n-1:n}(\omega_i)}{L^{\gamma}_{ns,n-1:n}},$$

the following path sampling identity holds [GM98]:

$$\log(\frac{L_{ns,n}}{L_{ns,n-1}}) = \int_0^1 \int \frac{d\log(\tilde{p}^{\gamma}_{n-1:n}(\omega_i))}{d\gamma} p^{\gamma}_{n-1:n}(\omega_i) d\omega_i d\gamma. \tag{13}$$

In our case, the logarithm of the target function $\tilde{p}$ is given by

$$\log(\tilde{p}^{\gamma}_{n-1:n}(\omega_i)) = (1-\gamma)\log(\tilde{p}_{n-1}(\omega_i)) + \gamma\log(\tilde{p}_n(\omega_i))$$

according to Equation 11.

Thus, the derivative of the logarithm of the function is

$$\frac{d\log(\tilde{p}^{\gamma}_{n-1:n}(\omega_i))}{d\gamma} = \log(\frac{\tilde{p}_n(\omega_i)}{\tilde{p}_{n-1}(\omega_i)}). \tag{14}$$

When considering a discrete path of P intermediate distribution, we can approximate Equation 13 with

$$\begin{aligned} \log(\tfrac{L_{ns,n}}{L_{ns,n-1}}) &= \tfrac{1}{P}\textstyle\sum_{k=1}^{P} \int \log(\tfrac{\tilde{p}_n(\omega_i)}{\tilde{p}_{n-1}(\omega_i)}) p^{k/P}_{n-1:n}(\omega_i) d\omega_i \\ &\approx \tfrac{1}{P}\textstyle\sum_{k=1}^{P}\sum_{j=1}^{N} W^{k,(j)} \log(\tfrac{\tilde{p}_n(\omega_i^{k,(j)})}{\tilde{p}_{n-1}(\omega_i^{k,(j)})}). \end{aligned} \tag{15}$$

Note that Equation 15 involves computing the normalized weights $W^{k,(j)}$ for every intermediate distribution $p^{k/P}_{n-1:n}$, which is not required when using the standard form of the intermediate distributions. The un-normalized intermediate weights $\tilde{w}^{k,(j)}$ are still computed according to Equation 12.

When using path sampling, we can also obtain an estimate of $\log(L_{ns}/L_{ns,1})$ as follows

$$\log(\frac{L_{ns,n}}{L_{ns,1}}) = \sum_{t=2}^{n} \log(\frac{L_{ns,t}}{L_{ns,t-1}}). \tag{16}$$

Computing the normalization constant using path sampling is a bit more expensive than the standard introduction of the P intermediate distributions as discussed in Section 4. However, the estimate of L_{ns} according to Equation 15 generally has lower variance than using P intermediate distributions with the standard ratio according to Equation 10.

6. Implementation

We have implemented the algorithm described above in a system that offers two rendering modes: a relighting mode, and a mode that allows for free camera movement.

Relighting. In relighting mode, the camera and object are in fixed locations, and only the environment can change. The initial frame is rendered using bidirectional importance sampling. For all subsequent frames, the samples for each pixel are propagated and mutated as described above, and form the sample set for the same pixel in the next frame (Figures 4 and 5).

Free camera movement. Once the camera is allowed to move freely, a surface point will project to different pixels in different frames. We take this into account by tracking the motion of the surface points at which the samples were generated. When we ray-trace a sample for one frame, we also store the corresponding information into the next frame. To this end, we compute which pixel the surface point will project to at the next time step, and store all samples from the current frame into that pixel. The memory requirement for this procedure is about 300 Bytes/pixel. When we want to compute the illumination for the next frame, most pixels will therefore have a sample set from the previous frame associated with them. We propagate and mutate those as discussed. Other pixels might not have a sample set due to disocclusion, or differences in sampling rate. We start bidirectional importance sampling for these pixels only (Figure 6).

General object movements can be dealt with the same way as camera movements: by knowing where object points will be located in the next time step, we can store sample information at the appropriate pixel locations. Currently, our implementation does not support this kind of object motion.

Figure 2: *Quality comparison of single distribution vs sequence of intermediate distributions with David model (Phong BRDF $s = 50$, $k_s = k_d = 0.5$) in Grace Cathedral HDR EM. Left: Single distribution with 10 MCMC moves. Center: 10 intermediate distributions with L_{ns} computed with standard ratio. Right: 10 intermediate distributions with L_{ns} computed with path sampling. Rendering times are identical (5 seconds).*

7. Results

In this section we compare the results of our unbiased SMC sampling algorithm with bidirectional importance sampling for rendering from HDR video environments. Images were generated with a reasonably well-optimized ray tracer using a voxel grid as the acceleration data structure for intersection queries. Our comparisons examine the output quality of the two rendering algorithms for a fixed amount of computing time. We performed these tests on a 3.6 GHz Xeon running Linux SuSE 9.0.

Figure 1 presents a comparison of our SMC algorithm with bidirectional importance sampling for a sequence of the sky probe gallery [SJW*04]. The image on the left is the first frame of the sequence rendered at a high quality using bidirectional sampling ($N = 16$, $M = 800$) in 8 seconds. The image in the center is the 5^{th} frame of the sequence rendered using our SMC algorithm ($N = 16$, $P = 5$) with path sampling in 4 seconds. The BRDF of the David model in these images has a high specular exponent (Phong $s = 50$) and no diffuse component. Under these conditions of high frequency lighting and highly specular BRDF, our SMC algorithm does much better than bidirectional importance sampling for the same computation time of 4 seconds. In this case, bidirectional sampling could only use a smaller number of samples ($M = 200$) to estimate L_{ns} for the same compute time (right).

Figure 2 presents the quality comparison of renderings produced with our SMC algorithm when using a single distribution (left) versus when using a sequence of intermediate distributions with standard ratio (center) and path sampling (right). These images correspond to the first frame rendered by our SMC algorithm after rotating the EM by 1.5° along the radial direction simulating a small change in the HDR illumination of the Grace Cathedral. The sequence of intermediate distributions greatly help in reducing the variance in the incremental computation of L_{ns} compared to a single distribution, while path sampling improves the quality of the estimate a bit more. Here, the BRDF of the David model has a significant diffuse component. Hence, the incremental estimate of the L_{ns} has higher variance compared to Figure 1 as the SMC algorithm uses only a very small number of samples to approximate the L_{ns}.

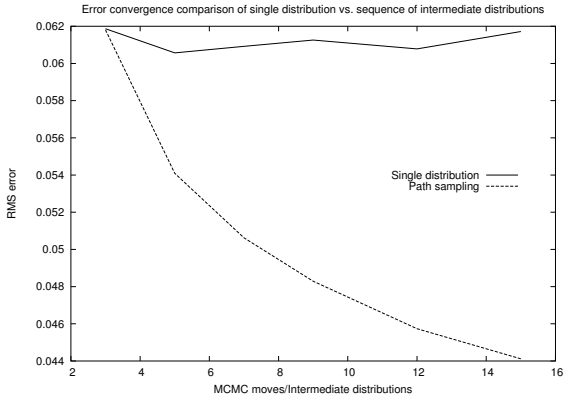

Figure 3: *Convergence plots of RMS errors for single distribution with multiple MCMC moves and sequence of intermediate distributions with one MCMC move each. Note how the RMS error reduces fast when using a sequence of intermediate distributions while the error does not really reduce much with just one distribution.*

In Figure 3, we present a comparison of the convergence in terms of RMS errors for a single distribution with multiple MCMC moves and a sequence of intermediate distributions

with path sampling. The RMS error plot was computed for same frame rendered in Figure 2 with our SMC algorithm. It is clear from the plot that multiple MCMC moves for a single distribution do not help much in reducing the variance in the incremental computation of L_{ns}, though they may help in exploring the target function $\tilde{p}$. A sequence of intermediate distributions with 1 MCMC move per distribution, on the other hand, is effective in reducing the variance in the computation of L_{ns}.

The introduction of a sequence of intermediate distributions also helps in reducing the degeneracy of the samples. We tracked the sample degeneracy, in terms of ESS, for sequence of 100 frames while rendering the David model (Phong BRDF $s = 50$, $k_s = k_d = 0.5$) in the Grace Cathedral HDR EM with rotations to the EM by 1.5° per frame. We observed that, on an average, the samples corresponding to 36% of the pixels required resampling when using a single distribution with 5 MCMC moves per sample. This fraction reduced to 18% when using a sequence of 5 intermediate distributions and 1 MCMC move per sample with path sampling. Hence, the additional cost of computing a sequence of intermediate distributions is offset to an extent by having to resample fewer samples. In practice, we used only up to $5-7$ intermediate distributions as this was enough to reduce the variance of most pixels. However, in order to maintain the quality of the renderings over time, we tracked the incremental estimate of L_{ns} and explicitly computed $L_{ns,n}$ using a large number of samples whenever the ratio $L_{ns,n}/L_{ns,1}$ altered significantly.

Figure 4 presents the quality comparison between bidirectional sampling and our SMC sampling algorithm for a dynamic environment sequence from dawn to dusk from the sky probe gallery [SJW*04]. We used the sample HDR sky probe images from the gallery that have been captured at 10 minute intervals as key frames of our sequence and interpolated to create 3 additional frames between each key frame. The David model in these images has the same highly specular BRDF as in Figure 1, and in this situation, our SMC algorithm performs much better than bidirectional importance sampling for the same computation time of 4 seconds. The SMC samples also have a lot more temporal coherence, greatly reducing flickering in the animation (please refer the video).

Figure 5 presents the quality comparison between bidirectional sampling and our SMC sampling algorithm for the same sky probe sequence from sunrise to sunset, except that the BRDF of the David model now has a significant diffuse component (Phong BRDF $s = 50$, $k_s = k_d = 0.5$). In this case, the difference in the quality of renderings produced by the two algorithms is not much as the SMC algorithm needs to estimate the reflected radiance due to a wider lobe with a small number of samples. However, the images corresponding to the SMC algorithm still have lower variance than bidirectional sampling wherever the specular contribution is high.

In Figure 6, we present example renderings with our SMC sampling algorithm adapted for changing viewpoint as discussed in Section 6. We rendered a sequence with the camera moving from right to left of the David model with 2° rotation of the camera between every frame. The images in the bottom row are false color visualizations of the pixels corresponding to re-projected points that used SMC sampling (green) and new exposed pixels that used bidirectional sampling (red) for the two viewpoints. As shown here, in a sequence involving a slowly moving camera, most pixels can effectively use samples propagated from neighboring pixels in subsequent frames.

Figure 6: *SMC sampling for a camera animation sequence with the David model in the Grace Cathedral EM. Top row: Images rendered with SMC sampling for dynamic viewpoints. Bottom row: False color visualization of bidirectional samples (red) and SMC samples (green).*

The speedup that we get in the case of moving cameras is lower than the speedup for relighting, since we cannot use SMC for all pixels due to occlusion, and since the reprojection of samples into the next frame consumes time. While we find a speedup of about a factor of 2 for relighting, the speedup for moving cameras is only 1.6. Note that a moving camera is in some sense the worst case scenario, since all

Figure 4: *Quality comparison between bidirectional sampling and our SMC sampling algorithm for a specular BRDF in the sky probe gallery sequence. Top row: Bidirectional sampling ($N = 16$, $M = 200$). Bottom row: SMC sampling ($N = 16$, $P = 5$, path sampling). All images took the same compute time of* 4 *seconds.*

points in the scene move. In general scenes with only few objects moving, one would expect a speedup somewhere between the extremes of relighting and the camera movement.

8. Conclusions

In this paper we have introduced the use of Sequential Monte Carlo methods for efficiently computing direct illumination in the presence of both high frequency Illumination and BRDF. By propagating samples over time, the method makes efficient use of coherence across frames. We demonstrate that this approach results in significantly reduced variance for the same compute time compared to other state-of-the-art methods.

Sequential Monte Carlo samplers have been the focus of recent research activities in statistics and machine learning. The sampling strategies used in this paper are at the leading edge of methods developed in those areas. We believe that these methods are promising for solving other sampling problems in computer graphics, for example for global illumination with photon maps.

9. Acknowledgments

We would like to thank David Burke for providing us his raytracer, Paul Debevec for the HDR sky probe gallery and environment maps, Derek Bradley for helping with the video, and the Stanford Digital Michelangelo project for the David model. We also thank our anonymous reviewers for their valuable comments and suggestions. The first author was supported by a UBC University Graduate Fellowship for the year 2005-06.

References

[ARBJ03] AGARWAL S., RAMAMOORTHI R., BELONGIE S., JENSEN H. W.: Structured importance sampling of environment maps. *ACM Transactions on Graphics (Proc. SIGGRAPH) 22*, 3 (July 2003), 605–612. 2

[BGH05] BURKE D., GHOSH A., HEIDRICH W.: Bidirectional importance sampling for direct illumination. In *Proc. of Eurographics Symposium on Rendering* (June 2005), pp. 147–156. 1, 2, 3, 4

Figure 5: *Quality comparison between bidirectional sampling and our SMC sampling algorithm for a BRDF with a diffuse component in the sky probe gallery sequence. Top row: Bidirectional sampling ($N = 16$, $M = 200$). Bottom row: SMC sampling ($N = 16$, $P = 5$, path sampling). All images took the same compute time of 4 seconds.*

[CD01] COHEN J., DEBEVEC P.: Light-gen. HDRshop plugin, 2001. http://www.ict.usc.edu/~jcohen/lightgen/lightgen.html. 2

[CJAMJ05] CLARBERG P., JAROSZ W., AKENINE-MÖLLER T., JENSEN H. W.: Wavelet importance sampling: Efficiently evalutating products of complex functions. *ACM Transactions on Graphics (Proc. SIGGRAPH) 24*, 3 (Aug. 2005), 1166–1175. 1, 3

[CTE05] CLINE D., TALBOT J., EGBERT P.: Energy redistribution path tracing. *ACM Transactions on Graphics (Proc. SIGGRAPH) 24*, 3 (Aug. 2005), 1186–1195. 3

[DdFG01] DOUCET A., DE FREITAS N., GORDON N.: *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, New York, 2001. 3

[Deb05] DEBEVEC P.: A median cut algorithm for light probe sampling, 2005. SIGGRAPH 2005 Poster. 2

[DJA*04] DUTRE P., JENSEN H. W., ARVO J., BALA K., BEKAERT P., MARSCHNER S., PHARR M.: *SIGGRAPH Course Notes: State of the Art in Monte Carlo Global Illumination*. ACM SIGGRAPH, 2004. 5

[dMDJ06] DEL MORAL P., DOUCET A., JASRA A.: Sequential monte carlo samplers. *J. Royal Statist. Soc. B 68*, 3 (2006), 1–26. 3, 4, 5

[FCL05] FAN S., CHENNEY S., LAI Y.: Metropolis photon sampling with optional user guidance. In *Proc. of Eurographics Symposium on Rendering* (June 2005), pp. 127–138. 3

[GM98] GELMAN A., MENG X. L.: Simulating normalization constants: From importance sampling to bridge sampling to path sampling. *Statist. Sci. 13* (1998), 163–185. 6, 7

[HDMS03] HAVRAN V., DAMEZ C., MYSZKOWSKI K., SEIDEL H.: An efficient spatio-temporal architecture for animation rendering. In *Proc. of Eurographics Symposium on Rendering* (June 2003), pp. 106–117. 3

[HDS03] HAVRAN V., DMITRIEV K., SEIDEL H.:

Goenometric diagram mapping of hemisphere. In *Short Presentations (Eurographics 2003)* (2003). 3

[HSK*05] HAVRAN V., SMYK M., KRAWCZYK G., MYSZKOWSKI K., SEIDEL H.: Interactive system for dynamic scene lighting using captured video environement maps. In *Proc. of Eurographics Symposium on Rendering* (June 2005), pp. 43–54. 1

[KK03] KOLLIG T., KELLER A.: Efficient illumination by high dynamic range images. In *Proc. of Eurographics Symposium on Rendering* (June 2003), pp. 45–51. 2

[LC98] LIU J., CHEN R.: Sequential monte carlo for dynamic systems. *J. Amer. Statist. Assoc. 93* (1998), 1032–1044. 4

[LRR04] LAWRENCE J., RUSINKIEWICZ S., RAMAMOORTHI R.: Efficient BRDF importance sampling using a factored representation. *ACM Transactions on Graphics (Proc. SIGGRAPH) 23*, 3 (Aug. 2004), 496–505. 2

[LRR05] LAWRENCE J., RUSINKIEWICZ S., RAMAMOORTHI R.: Adaptive numerical cumulative distribution functions for efficient importance sampling. In *Proc. of Eurographics Symposium on Rendering* (June 2005), pp. 11–20. 2, 3

[MFSSK06] MÉNDEZ-FELIU A., SBERT M., SZIRMAY-KALOS L.: Reusing frames in camera animation. *Journal of WSCG, 14* (2006). 3

[MH97] MCCOOL M. D., HARWOOD P. K.: Probability trees. In *Proc. Graphics Interface* (1997), pp. 37–46. 2

[MRR*53] METROPOLIS N., ROSENBLUTH A. W., ROSENBLUTH M. N., TELLER A. H., TELLER E.: Equation of state calculations by fast computing machines. *Journal of Chemical Physics 21*, 6 (1953), 1087–1092. 2

[ODJ04] OSTROMOUKHOV V., DONOHUE C., JODOIN P.-M.: Fast hierarchical importance sampling with blue noise properties. *ACM Transactions on Graphics (Proc. SIGGRAPH) 23*, 3 (Aug. 2004), 488–495. 2

[Shi00] SHIRLEY P.: *Realistic Ray Tracing*. A K Peters, Natick, 2000. 2

[SHS02] SECORD A., HEIDRICH W., STREIT L.: Fast primitive distribution for illustration. In *Proc. of Eurographics Workshop on Rendering* (June 2002), pp. 215–226. 2

[SJW*04] STUMPFEL J., JONES A., WENGER A., TCHOU C., HAWKINS T., DEBEVEC P.: Direct hdr capture of the sun and sky, 2004. SIGGRAPH 2004 Poster. 8, 9

[SSSK04] SBERT M., SZECSI L., SZIRMAY-KALOS L.: Real-time light animation. *Computer Graphics Forum (Eurographics 04 Proceedings) 23*, 3 (Sept. 2004), 291–299. 3

[TCE05] TALBOT J., CLINE D., EGBERT P.: Importance resampling for global illumination. In *Proc. of Eurographics Symposium on Rendering* (June 2005), pp. 139–146. 1, 3

[VG95] VEACH E., GUIBAS L.: Optimally combining sampling techniques for monte carlo rendering. In *Proc. of ACM SIGGRAPH '95* (Aug. 1995), pp. 419–428. 3

[VG97] VEACH E., GUIBAS L.: Metropolis light transport. In *Proc. of ACM SIGGRAPH '97* (1997), pp. 65–76. 3

[WWL05] WAN L., WONG T., LEUNG C.: Spherical q2-tree for sampling dynamic environment sequences. In *Proc. of Eurographics Symposium on Rendering* (June 2005), pp. 21–30. 1, 3

Appendix A: Normalization Constant

The normalization constant $L_{ns,n}$ at time n can be incrementally estimated according to Equation 10 in Section 3.1. This result can be explained as follows: the weighted empirical distribution $\{W_{n-1}^{(j)}, \omega_{i,n-1:n}^{(j)}\}$ obtained after the MCMC sampling step is an approximation of $p_{n-1}(\omega_{i,n-1})K_n(\omega_{i,n-1},\omega_{i,n})$ according to Equation 5. The expectation $E(\tilde{w}_n)$ of the incremental weights $\tilde{w}_n$ with respect to this joint distribution $p_{n-1}\cdot K_n$ is

$$
\begin{aligned}
&E(\tilde{w}_n)\\
&= \int_\Omega \int_\Omega \tilde{w}_n p_{n-1}(\omega_{i,n-1}) K_n(\omega_{i,n-1},\omega_{i,n})\, d\omega_{i,n}\, d\omega_{i,n-1}\\
&= \int_\Omega \int_\Omega \frac{\tilde{p}_n(\omega_{i,n-1})}{\tilde{p}_{n-1}(\omega_{i,n-1})} p_{n-1}(\omega_{i,n-1}) K_n(\omega_{i,n-1},\omega_{i,n})\\
&\qquad d\omega_{i,n}\, d\omega_{i,n-1}\\
&= \frac{1}{L_{ns,n-1}} \int_\Omega \int_\Omega \tilde{p}_n(\omega_{i,n-1}) K_n(\omega_{i,n-1},\omega_{i,n})\, d\omega_{i,n}\, d\omega_{i,n-1}\\
&= \frac{1}{L_{ns,n-1}} \int_\Omega \tilde{p}_n(\omega_{i,n-1})\, d\omega_{i,n-1}\\
&= \frac{L_{ns,n}}{L_{ns,n-1}}.
\end{aligned}
$$

The Monte Carlo estimate of this expectation is given by

$$E(\tilde{w}_n) \approx \sum_{j=1}^{N} W_{n-1}^{(j)} \cdot \tilde{w}_n^{(j)}. \tag{17}$$