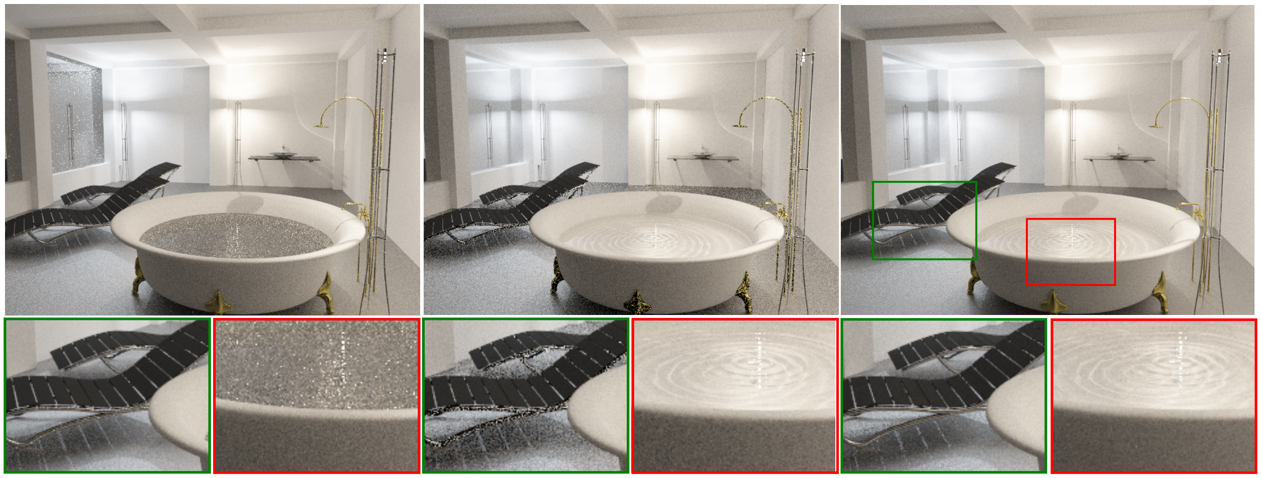
A Path Space Extension for Robust Light Transport Simulation

Toshiya Hachisuka[1,3] Jacopo Pantaleoni[2] Henrik Wann Jensen[3]
[1]Aarhus University [2]NVIDIA Research [3]UC San Diego



Monte Carlo Path Integration
RMS Error: 0.07836

Photon Density Estimation
RMS Error: 0.04638

Our Unified Framework
RMS Error: 0.01246

Figure 1: *Equal-time comparison of rendered images of a bathroom scene with realistic lighting fixtures. The scene includes highly glossy reflections and complex caustics due to lighting fixtures which are both typical in interior design. Existing approaches of light transport simulation, Monte Carlo path integration and photon density estimation, are inefficient to render this type of scenes. Our new framework of light transport simulation automatically combines Monte Carlo path integration and photon density estimation by extending the sampling space of light transport paths, and produces a significantly more accurate solution in the same computation time.*

Abstract

We propose a new sampling space for light transport simulation which allows us to unify two popular algorithms with orthogonal strengths: unbiased Monte Carlo path integration and photon density estimation. Traditionally, unbiased Monte Carlo path integration had been considered the only approach for accurate light transport simulation. However, recent work in photon density estimation has demonstrated that there are several practical scene configurations where photon density estimation is more efficient and accurate. In order to take the best of both worlds without relying on a heuristic choice of the algorithms, we combine both algorithms through a theoretically rigorous application of multiple importance sampling. Our contributions are two-fold: first, we introduce a path space extension that serves as a basis for a unified view of unbiased Monte Carlo path integration and photon density estimation. This extension gives us a mathematical ground for fully robust light transport simulation algorithms; second, we extend the theoretical analysis of provably good multiple importance sampling strategies by considering the presence of a density estimation method. This analysis leads to important conditions for obtaining a nearly optimal combination of unbiased Monte Carlo path integration and photon density estimation. We demonstrate that the resulting algorithm can robustly render many scene configurations that were previously considered intractable.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Raytracing;

1 Introduction

Efficiently solving the light transport problem under various scene configurations has been a core research topic in photorealistic image synthesis for more than 25 years. Since the rendering equation [Kajiya 1986] formulates the light transport problem as a recursive integral equation, solving the light transport problem is essentially equivalent to solving integrals. Although there are many ways to solve integrals, in photorealistic image synthesis, Monte Carlo integration has been a popular approach due to its generality.

The main difficulty is that many scene configurations introduce ill-behaved features in integrals such as discontinuities and singularities. Such features can significantly slow down the convergence of Monte Carlo integration. For example, path tracing [Kajiya 1986] rapidly converges to an accurate solution if a scene consists of diffuse materials and direct diffuse illumination. However, by adding an object with specular reflections and a small light source, path tracing can suddenly converge very slowly, because its sampling techniques do not efficiently capture light transport paths under such a configuration. In other words, path tracing is not *robust* to such changes in a scene configuration.

One notable development on improving robustness of light transport simulation is multiple importance sampling [Veach and Guibas 1995]. The main idea is to combine multiple Monte Carlo integrators such that the combined integrator gives us a more accurate solution than using each integrator alone. Multiple importance sampling led to the development of bidirectional path tracing [Lafortune and Willems 1993; Veach and Guibas 1995], which is still considered one of the most robust light transport simulation algorithms.

Since multiple importance sampling builds upon *unbiased* Monte Carlo integration, bidirectional path tracing considers only modes of light transport that can be sampled with unbiased sampling techniques. This is an inconvenient restriction since it is known that there are certain paths of light that cannot be sampled with any form of unbiased local path sampling [Veach 1998]. In other words, even bidirectional path tracing is not fully robust. Indeed, recent work on photon density estimation [Hachisuka et al. 2008; Hachisuka and Jensen 2009; Knaus and Zwicker 2011], which is a *biased* estimator, has demonstrated that the biased methods can be more efficient than the unbiased ones in several cases. Ideally, we would like to take the best of both in order to obtain a single robust light transport algorithm.

Combining photon density estimation with unbiased Monte Carlo path integration through multiple importance sampling presents two major challenges: the first is a fundamental difference in the dimensionality of the sampling spaces representing paths of a given length. Figure 2 shows all the bidirectional path tracing and photon density estimation techniques that can sample a given path of length two. While all bidirectional path tracing techniques require three vertices to construct this path, photon density estimation requires four. This difference is formalized by the area measure of the respective probability density functions of the same path, which is dA^{-3} in the Monte Carlo integration case and dA^{-4} in the density estimation case. The second challenge is the need to quantify the effect of bias on the optimality of the known multiple importance sampling weighting strategies.

We introduce a novel path space that extends the sampling space of light transport paths by *vertex perturbations*. Using this framework, we can describe Monte Carlo path integration and photon density estimation under the same space. Figure 3 illustrates the basic idea. Our second contribution is an extended theoretical analysis of the original multiple importance sampling framework in the presence of bias. The results of this analysis allow us to combine these two algorithms with a provably good combination strategy.

These two contributions allow us to build a new rendering algorithm, *unified path sampling*, that subsumes Monte Carlo path integration and photon density estimation under a single framework. This new algorithm is considerably more robust than of the two approaches alone. Figure 1 highlights our results under a typical scene configuration in interior design.

2 Background

Multiple importance sampling [Veach and Guibas 1995] is a powerful framework that allows the use of multiple unbiased Monte Carlo integration techniques with different probability density functions to solve an integral. The basic idea is that, if we need to compute an integral

$$I = \int_{\Omega} f(\mathbf{x}) d\mu(\mathbf{x}) \tag{1}$$

over some domain Ω, we can combine M different techniques to generate samples in this same domain Ω. Each i-th technique has a different probability density function $p_i(\mathbf{x})$ and approximates different parts of the integrand f better than the others. We then weight contributions of individual samples to build an estimator for the given integral. To be precise, if the i-th technique is used to generate n_i samples $\{X_{i,j} : i = 1, \ldots, M, j = 1, \ldots, n_i\}$, multiple importance sampling gives us the unbiased estimator of I as

$$I = \mathrm{E}\left[\sum_{i=1}^{M} \frac{1}{n_i} \sum_{j=1}^{n_i} \frac{w_i(X_{i,j}) f(X_{i,j})}{p_i(X_{i,j})}\right], \tag{2}$$

as long as

$$\sum_{i=1}^{M} w_i(\mathbf{x}) = 1 \tag{3}$$

and $w_i(\mathbf{x}) = 0$ whenever $p_i(\mathbf{x}) = 0$.

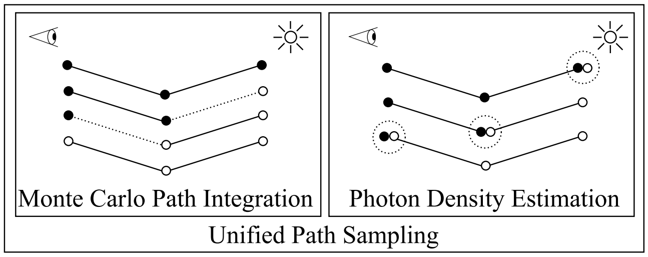


Figure 2: *Key concept of our unified path sampling framework. Monte Carlo path integration considers only connections of subpaths via local path sampling (left). On the other hand, photon density estimation considers only connections of subpaths via range query (right). The concept of our unified path sampling framework is to combine all such previous sampling techniques under a single framework using multiple importance sampling. The white circles are light vertices sampled from the light source, and the black circles are eye vertices sampled from the eye. The dotted lines/circles are path connections using local path sampling and range query, respectively.*

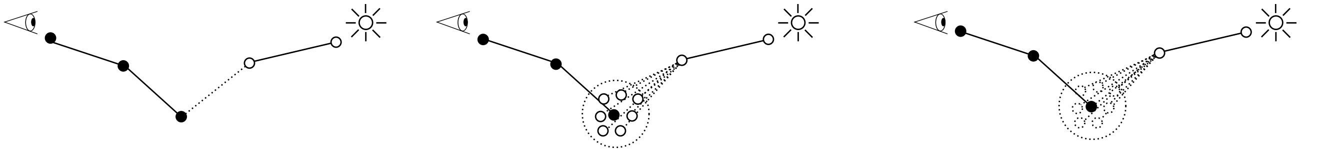
One notable application of multiple importance sampling in light transport simulation is bidirectional path tracing [Lafortune and Willems 1993; Veach and Guibas 1994]. Bidirectional path tracing generates a family of complete light transport paths by connecting two subpaths traced from the eye and a light source. By changing the length of each subpath, we can consider multiple sampling techniques with different probability density functions that sample the exact same path. Veach and Guibas [1995] showed that we can apply multiple importance sampling to combine them by evaluating and combining their probability density functions.

Bidirectional path tracing is robust to many different types of illumination. However, fundamental limitations of unbiased Monte Carlo integration methods do not allow bidirectional path tracing to capture some important light transport paths [Veach 1998]. To be precise, following Heckbert's regular expression of light transport [1990][1], any path that does not contain a substring DD (i.e., two subsequent "diffuse" interactions) cannot be sampled by any unbiased Monte Carlo path sampling technique.

Such transport paths include specular reflections of caustics due to a point light source seeing through a pinhole camera. At the moment, capturing these phenomena (or even variations that are essentially analogous such as using a small area light source instead) is practical only with photon density estimation [Jensen 1996] and its recent progressive extensions [Hachisuka et al. 2008; Knaus and Zwicker 2011]. This situation forces users to choose between unbiased Monte Carlo path integration (e.g., bidirectional path tracing) and photon density estimation, depending on the scene configuration.

Some recent work proposed a combination of these two algorithms to a various degree, and presented a more robust algorithm than using one of the algorithms alone [Bekaert et al. 2003; Hachisuka and Jensen 2009; Vorba and Křivánek 2011], however, none of exiting work attempted to fully unify unbiased Monte Carlo path integration and photon density estimation. Our main contribution over prior work is a novel mathematical framework that achieves the complete unification of these two algorithms for the first time.

[1] Under Heckbert's terminology, "D" is used to represent any *non-singular* scattering phenomena, which indeed includes glossy reflections, whereas "S" is used to represent singular scattering only.



(a) A path in MC path integration (b) Corresponding paths in density estimation (c) Corresponding path in our extended space

Figure 3: *Path space extension for unifying Monte Carlo path integration and photon density estimation. A path sampled in Monte Carlo path integration corresponds to one of the infinitely many possible paths in photon density estimation. The probability density of the path in (a) is not in the same space as the corresponding path in (b). We propose to use an extended path space of Monte Carlo path integration by considering a random perturbation of the last connecting light vertex within the neighborhood of the originally sampled vertex as in (c). With this extension, both Monte Carlo path integration and photon density estimation covers exactly the same path space.*

3 Theory

In the following subsections, we first introduce path space extension via *vertex perturbation*, which defines an extended path space that covers unbiased Monte Carlo path integration and photon density estimation under a unified definition. Figure 3 illustrates this idea. We then describe some important conditions that need to be considered when we apply multiple importance sampling for biased estimators. Table 1 summarizes our notations that are used throughout the paper.

Notation	Description
E_i	ith vertex generated from the eye
L_i	ith vertex generated from the light source
r	radius of range query in photon density estimation
N_L	the number of light vertices
N_E	the number of eye vertices
M	the number of techniques
p_{mc}	probability density in MC integration
p_{de}	probability density in density estimation
p_{emc}	probability density in extended MC integration
p_{ups}	probability density in unified path space
A_{N_E}	contribution from the eye subpath with N_E vertices
A_{N_L}	contribution from the light subpath with N_L vertices

Table 1: *Descriptions of the notations used in the paper.*

3.1 Path Space Extension via Vertex Perturbation

3.1.1 Problem with Existing Path Spaces

Suppose that we generated $N_L = 2$ vertices traced from a light source and $N_E = 3$ vertices traced from the eye. Monte Carlo path integration generates a complete path of length $N_L + N_E - 1$ by connecting the last eye vertex and the last light vertex. The probability density function of a complete path is given by the multiplication of probability density of each sub path [Veach 1998]:

$$\begin{aligned} &p_{\mathrm{mc}}\left(E_1E_2E_3L_2L_1\right) \\ &= p(E_1)p(E_2|E_1)p(E_3|E_2)p(L_2|L_1)p(L_1), \end{aligned} \tag{4}$$

where $p(x|y)$ is probability density of x given y. We assume that the processes of generating light subpaths and eye subpaths are statistically independent. Note that probability density functions do reflect the sampling procedure such as importance sampling.

We now consider photon density estimation to generate the same complete path. The important point is that photon density estimation generates a complete path of length $N_L + N_E - 2$ by considering the last eye vertex and the last light vertex as a single vertex, not $N_L + N_E - 1$ as in Monte Carlo path integration. Therefore, in order to generate a complete path of length four, we need to use six vertices, not five. We can divide the number of vertices in an arbitrary way, however, suppose for now that we use $N_L = 3$ light vertices and $N_E = 3$ eye vertices. The left side of Figure 4 illustrates this case. The probability density of a complete path is

$$\begin{aligned} &p_{\mathrm{de}}\left(E_1E_2E_3L_3L_2L_1\right) \\ &= p(E_1)p(E_2|E_1)p(E_3|E_2)p(L_3|L_2)p(L_2|L_1)p(L_1). \end{aligned} \tag{5}$$

Note that the size of a query region does not change the probability density function, but only the path contributions.

It is rather tempting to consider that the probability density of a complete path in photon density estimation is proportional to the size of a range query πr^2. One can see that this is incorrect by noting that such a wrong definition results in an extra division by πr^2 due to the division by the incorrect probability density *and* the conversion of flux to radiance. Conceptually, we can think of photon density estimation as connection of paths via range query. Similar to the connection procedure via shadow rays in bidirectional path tracing, this connection procedure does not affect the probability density function.

Equation 4 and Equation 5 highlight the problem of combining these two techniques. Suppose that those probability density functions use the area measure dA (i.e., each vertex is a point on surface) [Veach 1998]. Since Equation 4 is a multiplication of *five* probability density functions, the overall probability density function is defined with the measure dA^{-5}. However, since Equation 5 is a multiplication of *six* terms, the overall probability density function is defined with the measure dA^{-6}.

Therefore, these two probability density functions are defined in *different spaces* with different numbers of dimensions. Figure 3 (a) and (b) illustrate this difference. These probability density functions have different measures because a single complete path in Monte Carlo path integration corresponds to a single path out of infinitely many in photon density estimation over dA. In other words, the two probability density functions do no cover the same sampling space. It is thus incorrect to use them in multiple importance sampling which is made to combine samples in the same domain.

3.1.2 Path Probability Densities in Unified Path Space

We resolve this inconsistency by extending the path space used for Monte Carlo path integration via random perturbation of the connection vertex. Figure 4 illustrates this idea. We add a randomly perturbed copy of E_3 at the connection as a new light vertex, L_3, within a query range with the area πr^2 of photon density estimation. With this extension, the two probability density functions have exactly the same measure for the same path. The probability density function of a complete path in this extended space is

$$\begin{aligned} &p_{\mathrm{emc}}\left(E_1E_2E_3L_3L_2L_1\right) \\ &= p_{\mathrm{mc}}\left(E_1E_2E_3L_2L_1\right)p(L_3|E_3L_2) \\ &= p_{\mathrm{mc}}\left(E_1E_2E_3L_2L_1\right)p(L_3|L_2) \\ &= p(E_1)p(E_2|E_1)p(E_3|E_2)\pi^{-1}r^{-2}p(L_2|L_1)p(L_1). \end{aligned} \tag{6}$$

We have $p(L_3|E_3L_2) = \pi^{-1}r^{-2} = p(L_3|E_3)$ since we pick another vertex uniformly within a query range around E_3 with the area $\pi^{-1}r^2$, given the connection E_3L_2. Since generations of E_3 and L_2 are independent Markov processes, we can derive that $p(L_3|E_3L_2) = p(L_3|L_2) = p(L_3|E_3)$.

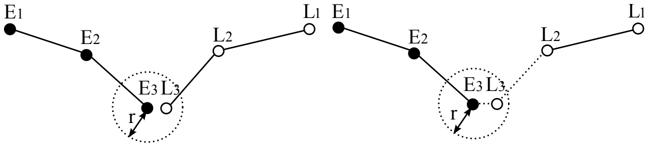
In order for photon density estimation to generate L_3 with the same probability density function, we assume that the probability density is constant within the query range. This is a reasonable assumption that has been used in recent work on photon density estimation techniques as an asymptotic case [Hachisuka et al. 2008; Knaus and Zwicker 2011].

Using this extended path space, the probability density function of a complete path in both Monte Carlo path integration and photon density estimation is uniformly defined as

$$\begin{aligned} &p_{\text{ups}}\left(E_1 \dots E_{N_E} L_{N_L} \dots L_1\right) \\ &= p_{\text{de}}\left(E_1 \dots E_{N_E} L_{N_L} \dots L_1\right) \\ &= p_{\text{emc}}\left(E_1 \dots E_{N_E} L_{N_L} \dots L_1\right) \\ &= \prod_{i=1}^{N_E} p(E_i|E_{i-1}) \prod_{i=1}^{N_L} p(L_i|L_{i-1}). \end{aligned} \tag{7}$$

The difference between Monte Carlo path integration and photon density estimation is handled by the definitions of L_{N_L} and $p(L_{N_L}|L_{N_L-1})$. For photon density estimation, they are the same as before: L_{N_L} is the last light vertex and $p(L_{N_L}|L_{N_L-1})$ is the probability density function of generating L_{N_L} given L_{N_L-1}. For Monte Carlo path integration, L_{N_L} is a random perturbation of E_{N_E} and $p(L_{N_L}|L_{N_L-1}) = p(L_{N_L}|E_{N_E}) = \pi^{-1}r^{-2}$. The rest of the definitions stay the same.

Figure 6 highlights the importance of considering the correct path space. As we mentioned earlier, the number of dimensions of the space of Monte Carlo path integration is lower than the number of dimensions of the space of photon density estimation for the same path. If we ignore this difference, the multiple importance sampling weight becomes scene-scale dependent. In other words, just by scaling the entire scene, which should generate the exact same image, the weights for each method can become arbitrarily larger or smaller. With our extended path space of Monte Carlo path integration, we can achieve scale invariance as expected.



Photon density estimation Extended Monte Carlo sampling

Figure 4: *Subpaths connections with our extended Monte Carlo sampling and photon density estimation.*

3.1.3 Path Contributions in Unified Path Space

Our framework also provides a unified definition of the contribution of a complete path in unbiased Monte Carlo path integration and photon density estimation. Given the contributions from eye subpath and light subpath, we can use this single definition of the contribution of the complete path A_C

$$A_C = A_{N_E} C_{\text{ups}}(E_{N_E}, L_{N_L}) A_{N_L}, \tag{8}$$

where we defined a new connection term $C_{\text{ups}}(E_{N_E}, L_{N_L}) = \pi^{-1}r^{-2}$ which denotes a connection via range query.

In the case of Monte Carlo path integration, we have

$$\begin{aligned} A_C &= A_{N_E} C_{\text{ups}}(E_{N_E}, L_{N_L}) A_{N_L} \\ &= A_{N_E} C_{\text{ups}}(E_{N_E}, L_{N_L}) \frac{C_{\text{mc}}(L_{N_L}, L_{N_L-1}) A_{N_L-1}}{p(L_{N_L}|L_{N_L-1})} \\ &= A_{N_E} \pi^{-1} r^{-2} \frac{C_{\text{mc}}(E_{N_E}, L_{N_L-1}) A_{N_L-1}}{\pi^{-1}r^{-2}} \\ &= A_{N_E} C_{\text{mc}}(E_{N_E}, L_{N_L-1}) A_{N_L-1}, \end{aligned} \tag{9}$$

where C_{mc} is defined as the connection term as in bidirectional path tracing [Veach 1998]. This is exactly the same as the unweighted contribution in bidirectional path tracing.

For photon density estimation, we have

$$\begin{aligned} A_C &= A_{N_E} C_{\text{ups}}(E_{N_E}, L_{N_L}) A_{N_L} \\ &= A_{N_E} \pi^{-1} r^{-2} A_{N_L}, \end{aligned} \tag{10}$$

which is also exactly the contribution of a single photon. In our final algorithm, the contribution will be weighted according to multiple importance sampling as we will describe later.

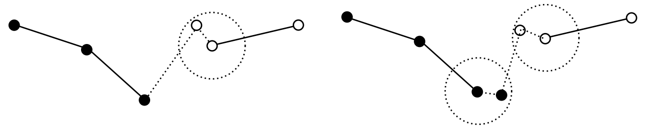
3.1.4 Other Possible Extensions

We have found this unified path space a powerful concept beyond combining Monte Carlo path integration and photon density estimation. Although we used a random perturbation of the last eye vertex as a new last light vertex in order to describe these two approaches, this is not the only possibility.

For example, considering vertex perturbation at the last light vertex (not at the eye vertex) makes this unified path space compatible with virtual spherical light sources [Hašan et al. 2009] (Figure 5 (a)). Such a path space can potentially provide another set of sampling techniques in Monte Carlo path integration via importance sampling according to incoming radiance. Note that local path sampling usually performs importance sampling according to BRDFs, not incoming radiance [Lafortune and Willems 1993; Veach and Guibas 1995].

It is also possible to consider random perturbation at multiple locations (Figure 5 (b)). This set of multiple perturbations corresponds to performing photon density estimation at multiple locations. Such a technique has not been explored as far as we know and it might potentially be efficient for a certain type of light transport. Since there are many possibilities and our focus is the combination of Monte Carlo path integration and photon density estimation, we have not tested such other extensions. However, we believe that our unified path space can lead to many different sampling techniques.

One can also consider that our framework completes the duality between *implicit* and *explicit* connections only partially present in bidirectional path tracing. In forward path tracing with next event estimation, there are two kinds of paths; paths that hit a light source, which are often called *implicit* connections, and paths that are generated by sampling a light source and performing an *explicit* connection to this point. In bidirectional path tracing, however, vertices are connected only by *explicit* connections. In other words, even if an eye vertex and a light vertex land on the exact same location, bidirectional path tracing does not consider this case as an *implicit* connection between the eye subpath and the light subpath. This is because the probability of such a connection happening is zero since each vertex is a point. Our framework completes this duality of implicit-explicit connections by introducing *implicit* subpath connections via density estimation. Although this fact does not affect our implementation, we found it an interesting theoretical observation.



(a) VSLs (b) Multiple density estimation

Figure 5: *Other possible extensions of the path space. (a) If we perturb the last light vertex, our extended path space covers the same path space as virtual spherical light sources [Hašan et al. 2009]. (b) It is also possible to consider multiple perturbations, which suggest unexplored sets of sampling techniques.*

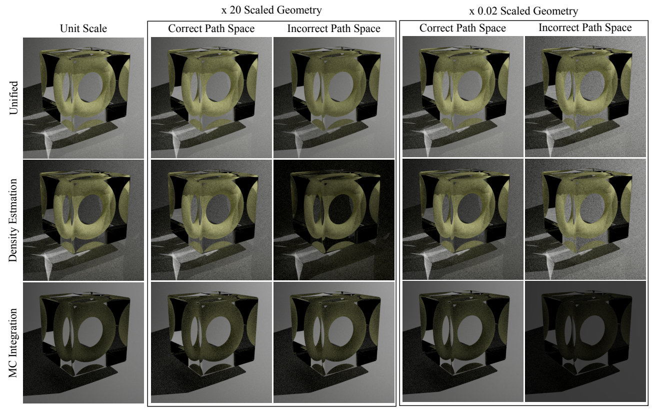


Figure 6: *Importance of considering the correct path space. Simply scaling the scene should result in exactly the same image, thus each contribution from density estimation and Monte Carlo path integration should stay the same. Without considering the correct path space, the contribution from each technique becomes scene-scale dependent. Our correct sampling space is scale independent as expected.*

3.2 Multiple Importance Sampling with Density Estimation

We now consider how multiple importance sampling can be applied to the combination of photon density estimation and Monte Carlo path integration. As we mentioned earlier, we provide a theoretical analysis on how bias in photon density estimation affects this combination. The end result is simple, but surely affects the implementation: we should use progressive photon density estimation [Hachisuka et al. 2008] with its alpha parameter equal to $\frac{2}{3}$. We describe our analysis in the following.

3.2.1 Problem with Biased Estimators

Veach and Guibas [1995] introduced a few weighting strategies which are provably good, in the sense that the resulting error is within constant away from that of the best possible weighting strategy. One such strategy is the *balance heuristic* defined as

$$w_i(\mathbf{x}) = \frac{n_i p_i(\mathbf{x})}{\sum_{k=1}^{n} n_k p_k(\mathbf{x})}. \tag{11}$$

where w_i is the weight for a sample that was generated by the ith technique, p_i is the probability density function of the ith technique, and n_i is the number of samples for the ith technique. Optimality of the resulting estimator $\hat{F}$ is shown as

$$\mathrm{Var}[\hat{F}] - \mathrm{Var}[F] \leq \left(\frac{1}{\min_i n_i} - \frac{1}{\sum_i n_i} \right) \mu^2, \tag{12}$$

where Var is an operator that returns variance, F is any estimator that is possible with multiple importance sampling, and μ is the exact solution. In other words, no other estimator F can further reduce variance from the balance heuristic more than the bound defined by the right hand side of this inequality. The left hand side is also called "variance gap" [Veach 1998]. This inequality also shows that the balance heuristic has a provably small additional error over all other strategies, since error in unbiased techniques are solely characterized by variance.

From this optimality claim and all the detailed derivations by Veach in his dissertation [1998], we can see that the original formulation of multiple importance sampling considers combinations of unbiased Monte Carlo integrators only[2].

This is undesirable since we would like to include density estimation into the multiple importance sampling framework, which is neither an unbiased method nor a pure Monte Carlo integrator. We would also like to have a *provably good* combination of unbiased Monte Carlo integration and biased photon density estimation, not just *any* combination that can be arbitrary worse than the unknown truly optimal combination. Since error in biased estimators is characterized by both bias and variance, having provably small variance does not necessarily mean that a combined estimator also has provably small error. We thus need to extend the theoretical analysis of multiple importance sampling to include photon density estimation.

3.2.2 Bias-Aware Balance Heuristic

In Appendix A, we show how to extend the original derivations of the balance heuristic to the case where one of the techniques is a biased estimator. This can be the case in our setting if we consider photon density estimation only at the vertex at which we actually performed photon density estimation. A provably good weighting strategy in this case is

$$\hat{w}_i(\mathbf{x}) = \frac{n_i p'_i(\mathbf{x})}{\sum_{k=1}^{M} n_k p'_k(\mathbf{x})}, \tag{13}$$

where

$$p'_i(\mathbf{x}) = \begin{cases} p_i(\mathbf{x}) & (i \neq n) \\ p_n(\mathbf{x}) \frac{1}{(1+r_n)^2 + n_n p_n(\mathbf{x}) A r_n^2} & (i = n), \end{cases} \tag{14}$$

A is a constant, and r_n is the relative magnitude of the contribution of the bias to the sampled value. Here the nth technique is biased. Note that if there is no bias $r_n = 0$, we obtain $p'_i = p_i$ and Equation 13 turns into the original balance heuristic in Equation 11. Note also that taking an infinite number of samples $\lim N \to \infty$ turns the weight for a biased method into zero, which makes the combined estimate converge to the correct solution even if $r_n \neq 0$.

Using this bias-aware balance heuristic as a combination strategy, the resulting estimator $\hat{F_B}$ satisfies the following inequality:

$$\mathrm{Error}[\hat{F_B}]^2 - \mathrm{Error}[F]^2 \leq \left(\frac{1}{\min_i n_i} - \frac{1}{\sum_i n_i} \right) \mu^2. \tag{15}$$

Notice the difference from Equation 12. This inequality is defined with the operator Error that returns error which includes both bias and variance. Similar to "variance gap", we call the left hand side as "error gap", which is the difference of errors between the provably good combination and any other combination.

[2] Note that just ensuring consistency of biased estimators is not enough. The problem we are considering in this paper is finding a *nearly optimal* combination of biased and unbiased estimators with *a finite number of samples*, not *any* combination or *infinite number of samples*. Note also that bias we consider in this paper is bias from density estimation, not bias that comes from one estimator not covering the entire sampling domain.

Unfortunately, we cannot use this provably good strategy in practice. In order to use this strategy, one would have to evaluate the magnitude of bias relative to the sampled value, r_i. Even if we had a method to estimate r_i, the provably good weighting strategy would require the additional ability of estimating r_i of samples which were *not even sampled* by a biased technique.

We propose one practical solution to this problem, which is to use the original balance heuristic in combination with progressive photon mapping [Hachisuka et al. 2008]. Since bias in progressive photon mapping is guaranteed to converge to zero [Knaus and Zwicker 2011] as we add more samples, the difference between the original balance heuristic (Equation 11) and the bias-aware balance heuristic (Equation 13) are expected to converge to zero at an infinite number of samples. The challenge however is that we still would like to pursue a provably good combination with any number of samples. In the following subsections, we describe a condition on this approach that keeps the resulting combination provably good.

3.2.3 Error Gap of the Balance Heuristic

In order to analyze the influence of bias, we first look at the consequence of using the original balance heuristic by ignoring bias in biased estimators. In any biased estimator, error is characterized by the following *bias-variance decomposition*:

$$\mathrm{Error}[F]^2 = \mathrm{Var}[F] + \mathrm{Bias}[F]^2. \tag{16}$$

We then look at the error gap (*not* the variance gap) of the original balance heuristic

$$\begin{aligned}&\mathrm{Error}[\hat{F}]^2 - \mathrm{Error}[F]^2\\ &= \mathrm{Var}[\hat{F}] + \mathrm{Bias}[\hat{F}]^2 - \mathrm{Var}[F] + \mathrm{Bias}[F]^2\\ &= \mathrm{Var}[\hat{F}] - \mathrm{Var}[F] + \mathrm{Bias}[\hat{F}]^2 - \mathrm{Bias}[F]^2.\end{aligned} \tag{17}$$

Substituting this equation to Equation 12, we obtain

$$\begin{aligned}&\mathrm{Error}[\hat{F}]^2 - \mathrm{Error}[F]^2\\ &\leq \left(\frac{1}{\min_{\mathrm{i}} n_i} - \frac{1}{\sum_i n_i}\right)\mu^2 + \mathrm{Bias}[\hat{F}]^2 - \mathrm{Bias}[F]^2\\ &\leq \left(\frac{1}{\min_{\mathrm{i}} n_i} - \frac{1}{\sum_i n_i}\right)\mu^2 + \mathrm{Bias}[\hat{F}]^2.\end{aligned} \tag{18}$$

Therefore, the error gap of the original balance heuristic under the presence of biased estimator is bounded by the original bound plus the additional term due to bias. Comparing this inequality and the inequality in Equation 15, using the original balance heuristic can be further away from the truly optimal (unknown) combination than the bias-aware balance heuristic by the additional term $\mathrm{Bias}[\hat{F}]^2$. This result shows that, depending on how bias changes according to the number of samples, the balance heuristic can be arbitrary away from a provably good strategy under the presence of a biased estimator.

3.2.4 Condition for a Provably Good Strategy

As we mentioned at the beginning, we use progressive photon density estimation [Hachisuka et al. 2008] with the hope that the original balance heuristic is still close to the provably good combination of the bias-aware balance heuristic. We show that setting the alpha parameter of progressive photon density estimation to $2/3$ can indeed achieve such a combination with the original balance heuristic in the Veach's sense [1995].

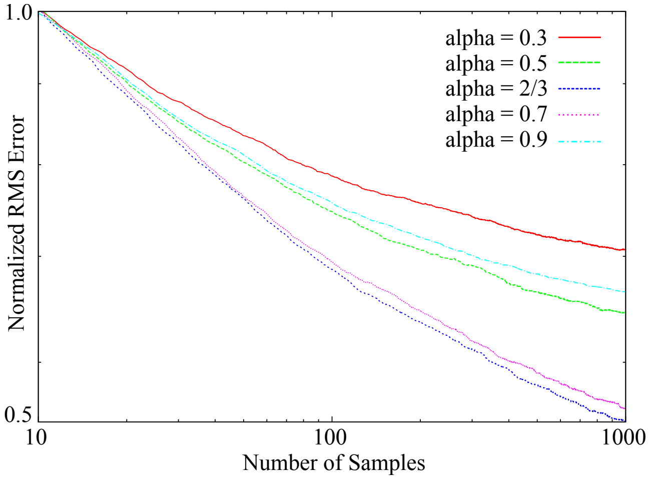


Figure 7: *Error gaps due to different α values. The graph plots RMS errors of the rendered images of the torus scene with different values for the alpha parameter. RMS errors are normalized such that the graph shows the difference in convergence rates. As predicted by our theoretical analysis, $\alpha = 2/3$ gives us the fastest convergence rate.*

Knaus and Zwicker [2011] showed that the asymptotic convergence rates of bias and variance in progressive density estimation are

$$\mathrm{Var} = O\left(\frac{1}{n^{\alpha}}\right) \qquad \mathrm{Bias} = O\left(\frac{1}{n^{1-\alpha}}\right), \tag{19}$$

where α is the parameter that controls the reduction rate of the radius in progressive density estimation. Substituting this result into Equation 18 yields

$$\mathrm{Error}[\hat{F}]^2 - \mathrm{Error}[F]^2 \leq \left(\frac{1}{n_n^{\alpha}} - \frac{1}{\sum_i n_i}\right)\mu^2 + \frac{C}{n_n^{2(1-\alpha)}}, \tag{20}$$

where C is a constant. We used $\min_{\mathrm{i}} n_i = n_n^{\alpha}$ by considering the fact that the variance of progressive photon mapping converges at the rate of $O\left(\frac{1}{n^{\alpha}}\right)$. We take the effect of slower convergence rate into account by replacing n_n by n_n^{α}. Note that this does not affect the derivation of the bias-aware balance heuristic since the derivations do not try to achieve the optimal distribution of the number of samples.

Our goal is to find conditions such that

$$\left(\frac{1}{n_n^{\alpha}} - \frac{1}{\sum_i n_i}\right)\mu^2 + \frac{C}{n_n^{2(1-\alpha)}} \approx \left(\frac{1}{n_n^{\alpha}} - \frac{1}{\sum_i n_i}\right)\mu^2 \tag{21}$$

for large enough $N = \sum_i n_i$. Note that the right hand side also uses the equation $\min_{\mathrm{i}} n_i = n_n^{\alpha}$ since we are now combining progressive photon density estimation and Monte Carlo path integration.

Now, consider the difference between the convergence rates of the bounds of the error gap in the original balance heuristic and the bias-aware balance heuristic:

$$\begin{aligned}\left(\frac{1}{n_n^{\alpha}} - \frac{1}{\sum_i n_i}\right)\mu^2 + \frac{C}{n_n^{2(1-\alpha)}} &\in O\left(\frac{1}{n_n^{\alpha}}\right) + O\left(\frac{1}{n_n^{2(1-\alpha)}}\right)\\ \left(\frac{1}{n_n^{\alpha}} - \frac{1}{\sum_i n_i}\right)\mu^2 &\in O\left(\frac{1}{n_n^{\alpha}}\right).\end{aligned} \tag{22}$$

The difference in convergence rates of the two bounds is minimized at $\alpha = 2/3$, which is the solution for $\alpha = 2(1-\alpha)$. In other words, using $\alpha = 2/3$ makes sure that the bound of the error gap from any other combination strategies reduces with the convergence rate of the bias-aware balance heuristic. The resulting error gap is

$$\begin{aligned}\mathrm{Error}[\hat{F}]^2 - \mathrm{Error}[F]^2 &\le \left(\frac{1}{n_n^{\alpha}} - \frac{1}{\sum_i n_i}\right)\mu^2 + \frac{C}{n_n^{2(1-\alpha)}} \\ &\le (C+1)\left(\frac{1}{n_n^{\alpha}} - \frac{1}{\sum_i n_i}\right)\mu^2 \qquad (23)\end{aligned}$$

Note that any other values of the alpha parameter makes the bound arbitrary away from above with given N. Figure 7 shows the results of a numerical experiment that confirms our theory. We have found that using the alpha value other than $2/3$ results in slower convergence rates. The condition $\alpha = 2/3$ is not only theoretically critical, but also practically important.

3.2.5 Number of Samples in Density Estimation

It has been known that density estimation can be formulated as Monte Carlo integration

$$E\left[L(x)\right] = E\left[\frac{1}{\pi r^2 N}\sum_{i=1}^{N} K(x, x_i)\Phi_i(x_i)\right] \qquad (24)$$

An important point in our context is the number of samples used to evaluate this estimator. Consider a case of bidirectional path tracing where we trace one eye subpath and one light subpath per pixel [Lafortune and Willems 1993; Veach and Guibas 1994]. If we have N pixels, we will trace N light sub-paths over the image and each pixel uses exactly one sample in Monte Carlo path integration. However, for photon density estimation, each estimator uses *all* of the light paths that are traced. In other words, each estimator uses N samples, not a single sample as in Monte Carlo path integration. This needs to be reflected in the computation of weight by setting $n_n = N$. Assuming the equal number of samples for unbiased Monte Carlo method, the end result is simple. We just multiply the probability density of photon density estimation by N, and perform the weight computation as usual.

4 Implementation

Given our unified path space, an implementation of our framework is a relatively straightforward extension of a typical implementation of bidirectional path tracing. Figure 9 shows a pseudocode of our implementation.

4.1 Main Rendering Process

We store eye paths and light paths to buffers, EyePaths and LightPaths, that have the same number of entries as the number of pixels in the image. Our theory does not require us to use the same number of eye subpaths and light subpaths, however, we have found that using this number of subpaths makes the implementation compatible with a typical implementation of bidirectional path tracing.

We generate eye paths and light paths for all pixels via GENEYEPATH() and GENLIGHTPATH() which are exactly the same as bidirectional path tracing. Note that only GENEYEPATH() takes pixel locations. We then build a photon map over all the light vertices by BUILDPM(). In our implementation, we used a spatial grid as an acceleration data structure, but it is possible to use different data structures such as a kD-tree. We reduce the radius for photon density estimation according to the alpha parameter and the number of iterations so far using the probabilistic formulation of progressive photon density estimation [Knaus and Zwicker 2011]. It is also possible to use the original stochastic progressive photon mapping [Hachisuka and Jensen 2009] for this part. Finally, COMBINEPATHS() connects eye subpaths and light subpaths to compute contributions to the image.

```
procedure RENDERING(Scene, Camera, Image, N_itr)
  for all Pixels(i, j)
    do EyePaths(i, j) ← GENEYEPATH(Scene, Camera, $i, j$)
       LightPaths(i, j) ← GENLIGHTPATH(Scene)
  BUILDPM(LightPaths)
  DERadius ← CALCRADIUS($\alpha$, N_itr)
  for all Pixels(i, j)
    do COMBINEPATHS(EyePaths, LightPaths, $i, j$)

procedure COMBINEPATHS(EyePaths, LightPaths, $i, j$)
  $C_{\text{eye}}$ ← CONNECTEYE(EyePaths(i, j), LightPaths(i, j))
  CONNECTLT(LightImage, EyePaths(i, j).V[0], LightPaths(i, j))
  $C_{\text{de}}$ ← CONNECTDE(EyePaths(i, j), LightPaths)
  EyeImage(i, j) ← EyeImage(i, j) + $C_{\text{eye}}$ + $C_{\text{de}}$/NumPixels

procedure BALANCEHEURISTIC(Path, $N_E$, $N_L$)
  PDF_Sum ← 0
  for $s$ ← 0 to Length(Path) + 1
    do $t$ ← Length(Path) + 1 − $s$
       PDF_MC ← PDF_UPS(Path, $s, t$, true )
       PDF_DE ← PDF_UPS(Path, $s, t$, false ) ∗ NumPixels
       if ($s = N_E$ & $t = N_L$) PDF_Path ← PDF_MC
       if ($s = N_E$ & $(t + 1) = N_L$) PDF_Path ← PDF_DE
       PDF_Sum ← PDF_Sum + PDF_MC + PDF_DE
  return (PDF_Path/PDF_Sum)
```

Figure 9: *Pseudocode for our framework.*

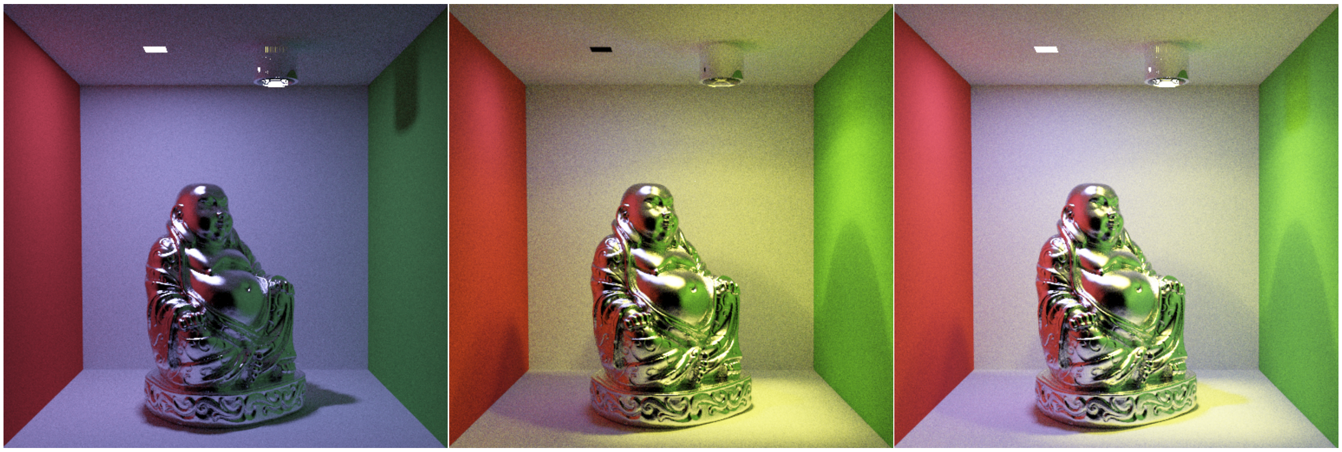
4.2 Subpath Connections

We have three connection procedures that need to be implemented separately. CONNECTEYE() connects the given eye path with at least two vertices and the light path by tracing shadow rays between vertices. This is the same procedure as bidirectional path tracing, except that the weight computation is extended to include photon density estimation. Note that, even though we consider a random perturbation of a vertex in our probability density functions, we do not need to actually perform this random perturbation. The probability density of sampling the exact vertex again is the same, so we do not need to perform such perturbation in practice within the sampling procedure.

CONNECTLT() connects the first eye vertex and all the vertices in the light path via shadow rays. This procedure accumulates contributions to a separate image buffer called LightImage than other two connection procedures. This is also the same as bidirectional path tracing with the extended weight computation.

CONNECTDE() connects the given eye path and *all* the light paths by range queries. A range query should be performed once for each eye vertex, not just only at the end of the eye path. The contribution is divided by NumPixels, which is necessary for correctly taking into account the difference in the number of samples for *path contributions*. We also need to take this factor into account in the weight computation.

The contributions from CONNECTLT() can be accumulated to any pixels in the image, whereas the contributions from CONNECTEYE() and CONNECTDE() are accumulated to a given pixel location (i, j). The final image is simply a sum of two image buffers, EyeImage and LightImage.



Weighted Contribution of MC Path Integration
Weighted Contribution of Density Estimation
Combined Result

Figure 8: *Weighted contribution of Monte Carlo path integration and photon density estimation within our unified path sampling framework. The scene has a Buddha statue with glossy reflections, and two small light sources with different colors (blue and yellow) where the yellow one is enclosed by a metal tube and a lens. The images show computed illumination due to Monte Carlo path integration (left, equivalent to techniques used in bidirectional path tracing), photon density estimation (center, equivalent to techniques used in progressive photon mapping), and the combined result (right). Each approach covers a different component of illumination based on our unified definition of probability density functions of paths. Note that a significant part of illumination can be efficiently covered by photon density estimation as predicted by multiple importance sampling.*

4.3 Weight Computation

Each connection procedure internally calls BALANCEHEURISTIC() in order to properly weight the contribution of *each* sample, not the accumulated contribution such as C_{eye} and C_{de}. Inside this procedure, we just call the evaluation procedure of the probability density function based on our unified path space. The boolean value given to PDF_UPS is true if we are considering the probability density function for Monte Carlo path integration. This boolean value is used for switching how to handle the last light vertex as we described. NumPixels is multiplied to the probability density function of photon density estimation in order to properly account for the difference in the number of samples between Monte Carlo path integration and photon density estimation.

4.4 Compatibilities with Other Rendering Methods

This implementation of our framework subsumes implementations of multiple rendering methods. If one would like to use bidirectional path tracing, we just need to disable the connection by density estimation (CONNECTDE()) and also disable the corresponding probability density evaluation inside BALANCEHEURISTIC(). Likewise, our framework can be converted into (bidirectional) path tracing, light tracing, (progressive) photon mapping, and stochastic progressive photon mapping just by limiting a set of sampling techniques.

Our implementation supports full bidirectional connections and full photon density estimation connections, but some cases are excluded since they are not always useful in practice. In particular, photon density estimation with only one eye vertex or light vertex is excluded. Using only one eye vertex is not possible with a pinhole camera, though this technique can be useful to simulate lens flare. Using only one light vertex in general does not provide benefit since we usually know the exact flux coming from the eye path hit point on a light source. Using photon density estimation in such a case introduces unnecessary bias. We however emphasize that it is easy to add such excluded connections back into our framework.

5 Results

We implemented bidirectional path tracing (BPT) [Veach and Guibas 1995], progressive photon mapping (PPM) [Hachisuka et al. 2008], stochastic progressive photon mapping (SPPM) [Hachisuka and Jensen 2009], and our unified path sampling (UPS) under the same framework. Although our theoretical framework supports different radius per pixel, we chose to use a global radius for all the photon density estimation for simplicity and picked the initial radius by hand. Since our sampling framework subsumes all of these methods, actual implementation of each method is realized by simply turning off certain sampling techniques under the single implementation of our complete framework. The reference solution to Figure 12 was rendered by BPT and others were rendered by SPPM with manual classifications of specular/non-specular materials for glossy reflections.

Note that comparisons with Markov chain Monte Carlo (MCMC) algorithms do not make sense since the contribution of our work is a new combination of existing sampling methods, *not* a new sampling method. The two concepts are completely orthogonal and can be combined naturally. Likewise, vertex perturbations are not related to mutations in MCMC.

We ran all the experiments on an Intel Core i7-2600 at 3.40 GHz with a single thread. The resolution of the images are either 512 × 512 or 640 × 480. We left the images intentionally unconverged to reveal error. Table 2 summarizes the total average number of samples per pixel in our test cases. Overall, we have found that unified path sampling can take more samples than bidirectional path tracing by counting a complete path as one sample. This is because connections via photon density estimation are computationally less costly than connections via local path sampling in our implementation.

Figure 1 highlights the advantage of our method in a realistic illumination setting for interior design. We have modeled a realistic lighting fixtures with emitters and reflectors. The dominant illumination is thus due to caustics, as it is the case in many lighting fixtures of the real world. Bidirectional path tracing, which is la-

Method	BPT	PPM	SPPM	UPS
RMSE	0.05003	0.05934	0.03324	0.02285

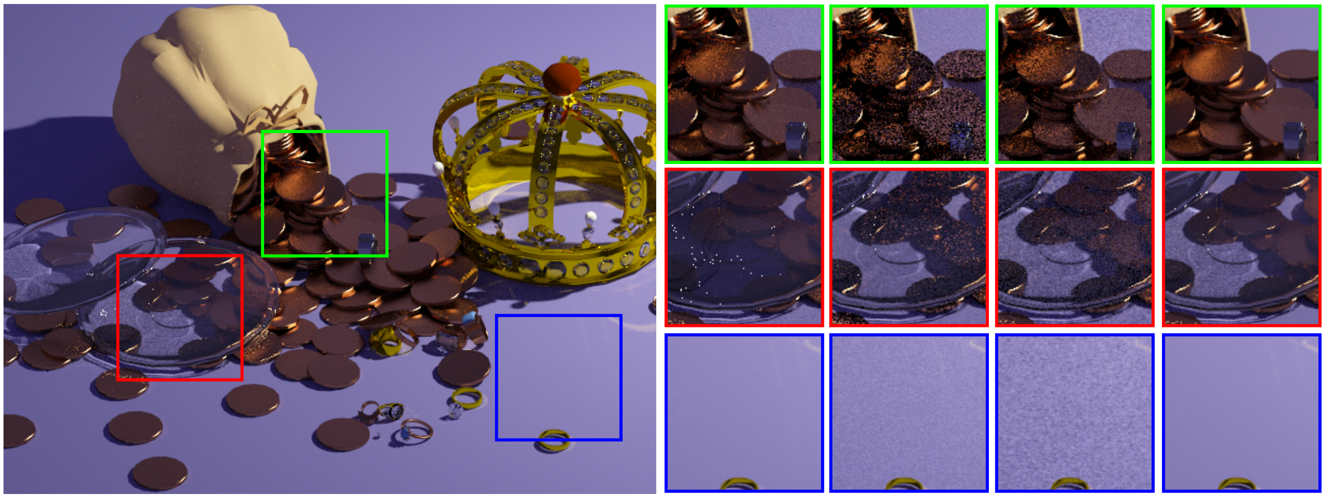


Figure 10: *Scene features high geometric complexity and illumination complexity. The scene has glass plates and coins and a crown with glossy metal illuminated by a small diffuse light source. The image on the left is rendered by our framework (UPS). The close-ups show parts of the images rendered by various methods using the same rendering time (120 min). Bidirectional path tracing (BPT) cannot efficiently render caustics seeing through glass, while glossy reflections are relatively less noisy. Progressive photon mapping (PPM) captures such indirectly visible caustics, but produces noisy results for glossy reflections. Stochastic progressive photon mapping (SPPM) captures all the illumination features reasonably well, but direct illumination is relatively noisy. Our framework (UPS) takes the best of all three approaches and captures all the illumination features efficiently.*

beled as Monte Carlo path integration, is efficient for computing some contributions from direct illumination and glossy reflections, yet indirectly visible caustics exhibit significant amount of noise (e.g., caustics seen through water in the bathtub). Progressive photon mapping, which is labeled as photon density estimation, handles such caustics and reflections of caustics robustly, but a sharp BRDF lobe of the highly glossy material becomes a source of noise. Our unified framework combines the strength of each method under a single framework without any user intervention, and produces a more accurate solution in the same rendering time.

The graph in Figure 11 shows the convergence of the RMS (Root Mean Square) errors of the same scene with different methods. In this graph, we used the equal number of samples as a comparison. This comparison is in favor of bidirectional path tracing in our implementation since Table 2 concludes that bidirectional path tracing is the most computationally costly method per sample. We choose this comparison in order to test if each sample in our framework is fundamentally more efficient than samples in other methods. Even under such a comparison, the graph confirms that our method provides an order of magnitude more accurate solution than both methods for the same number of samples.

Figure 10 compares all of the rendering methods in our tests for another scene using the same rendering time. This scene also features highly glossy reflections, which are difficult to capture efficiently with photon density estimation, and indirectly visible caustics, which are difficult to capture efficiently with Monte Carlo path integration. The comparison includes stochastic progressive photon mapping that already demonstrated efficient rendering of glossy reflections by tracing one bounce ray from a visible point through each pixel [Hachisuka and Jensen 2009]. The issue however is that whether we trace such rays or not is based on a heuristic classification of diffuse/non-diffuse materials. Our unified path sampling framework avoids introducing such a heuristic and combines all the possible techniques with a provably good strategy. Note that diffuse direct illumination is significantly less noisy with our unified path sampling in comparison to stochastic progressive photon mapping. The reason is that Monte Carlo path integration automatically dominates the contribution for such light transport with a provably good combination strategy.

Figure 12 shows another equal-time comparison with bidirectional path tracing for a scene that has only diffuse materials. This scene does not feature any light transport that is particularly challenging for bidirectional path tracing. Even in such a scene configuration, our unified path sampling is still comparable to bidirectional path tracing since our framework subsumes bidirectional path tracing.

Figure 13 shows sequences of rendered images of a simple scene where we have a Cornell box with a glossy box and a glass box with a small diffuse area light source. Despite its relatively simple configuration, bidirectional path tracing and progressive photon mapping already show their inefficiency for capturing certain light transport. Our method shows the advantage over other methods even in this simple scene. Since our method captures all the features equally well, it is also possible to quickly identify overall illumination in the scene only after a few samples.

We emphasize that photon density estimation is important in many real-world scenarios, not just for very special cases. Figure 8 highlights such a case, where we have two light sources; a blue diffuse area light source, and a yellow diffuse area light source enclosed by a metal tube and a lens. The only difference between these two light sources is whether they are modeled after a realistic lighting fixture or a bare emitter. The blue light source directly illuminates the scene, while the yellow light source illuminates the scene via caustics just like many lighting fixtures in the real world. Our unified path sampling algorithm puts higher weight for Monte Carlo path integration techniques for illumination from the blue light source and photon density estimation techniques for illumination from the yellow light source. This is a provably good combination predicted by our theory, and photon density estimation indeed captures a significant portion of illumination due to the yellow light source.

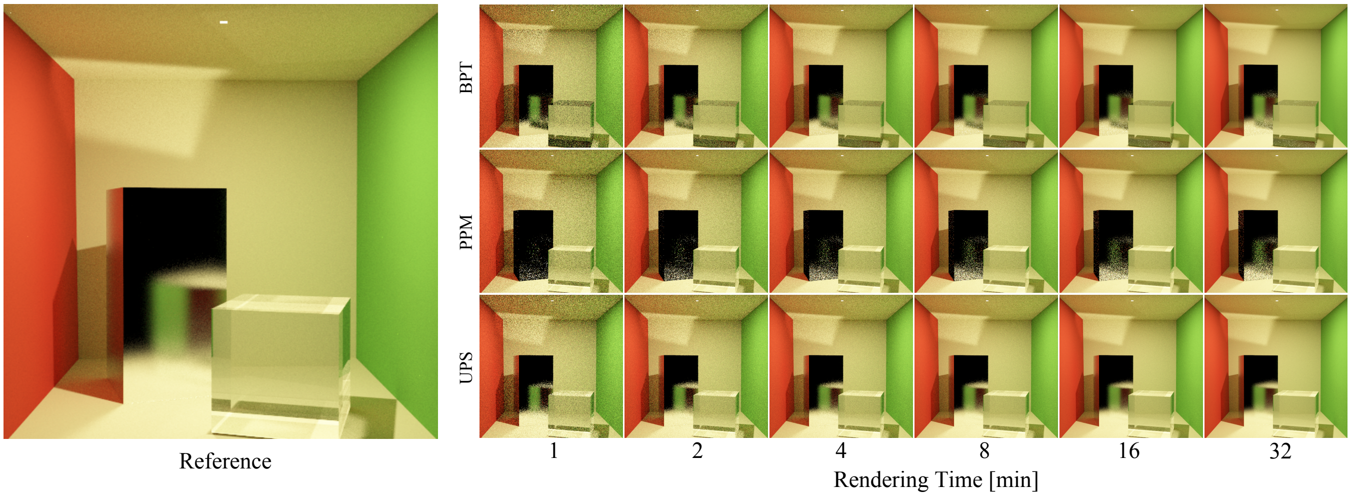


Figure 13: *Sequences of rendered images using different approaches. Even this simple scene reveals issues of using previous approach alone. Bidirectional path tracing (BPT) is inefficient for sampling caustics seen through the glass cube, while progressive photon mapping (PPM) is inefficient for sampling highly glossy reflections. Although these two approaches theoretically guarantee convergence to the correct solution in such cases, the sequences of images show slow convergence in practice. Our framework (UPS) unifies both approaches into a single unified path sampling method and significantly improves convergence speed for such inefficient cases. RMS errors at 32 min are 0.09146 (BPT), 0.08874 (PPM), and 0.01536 (UPS) respectively.*

Scene	BPT	PPM	SPPM	UPS	Time [min]
Bathroom	1396	9494	5313	2085	240
Buddha	38	205	112	72	10
Conference	120	799	475	237	30
Cornell	264	1838	679	484	32
Torus	448	2438	1508	738	60
Treasures	367	1386	1184	618	120
Average	5.33	32.51	17.41	9.71	N/A

Table 2: *Statistics of our experiments. The numbers in the column of each method (BPT: Bidirectional Path Tracing, PPM: Progressive Photon Mapping, SPPM: Stochastic Progressive Photon Mapping, and UPS: Unified Path Sampling) show the average numbers of samples per pixel. The last row shows the average numbers of samples per pixel per minute over our test cases.*

6 Discussion

Although our implementation supports all the sampling techniques that we have introduced in our theory and all of our experiments uses the full combinations, we have found that excluding local path sampling techniques with more than one light vertex can also perform well in practice. This is essentially the combination of path tracing with next event estimations [Kajiya 1986] and photon density estimation. Figure 14 compares rendered images using such a subset of the full combinations and the full combinations within our framework.

Using full combinations is more efficient when we have strong indirect illumination as in our bathroom scene. However, since the cost of taking each sample increases due to the increased number of probability density functions in the weight computation, in the same computation time, using a subset of the combinations can take more samples and be more efficient in many cases.

A few studies also explored applications of multiple importance sampling to regular (not progressive) photon mapping. Bekaert et al. proposed a combination of the regular photon density estimation using multiple importance sampling in the context of their modified photon density estimator [2003]. Due to their connection kernel formulation, their framework cannot not handle caustics from specular materials which our method can efficiently handle. Vorba and Křivánek [2011] described how multiple importance sampling can be used to combine only photon density estimation techniques. Contrarily to their approach, our method does not limit the combinations only to photon density estimation, but provides full combinations of Monte Carlo path integration and photon density estimation.

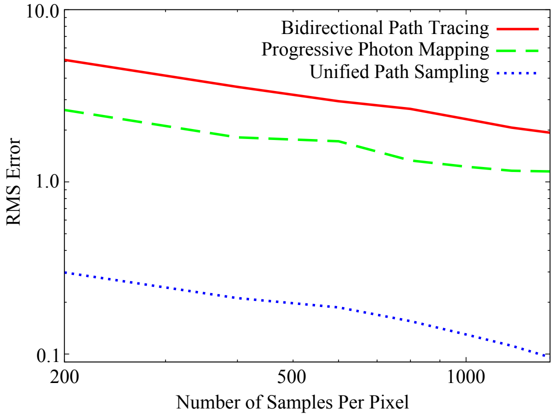
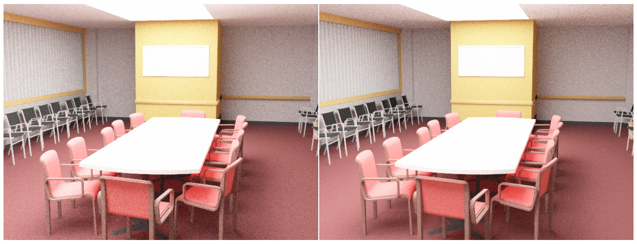


Figure 11: *RMS errors of the bathroom scene using the same average number of samples per pixel. We used equal number of samples as a comparison in this graph, favoring bidirectional path tracing: since each sample of bidirectional path tracing takes more computation time in our implementation, an equal time comparison would make the gap between our method and bidirectional path tracing even larger.*

One concurrent work is vertex merging by Georgiev et al. [2011]. Vertex merging is essentially an application of multiple importance sampling to stochastic progressive photon mapping. In that regard, our work is highly related to their work. Since this is concurrent work and the details of their work has not been fully disclosed, we do not provide side-by-side comparisons in this paper. We however emphasize that there are two critical differences between our work and their work.



Bidirectional path tracing
RMS Error: 0.02152

Unified path sampling
RMS Error: 0.02255

Figure 12: *Conference room with diffuse surfaces and a large diffuse light sources. For this type of scenes, our method (unified path sampling) performs almost as well as bidirectional path tracing since the contribution of Monte Carlo path integration automatically dominates the final image.*

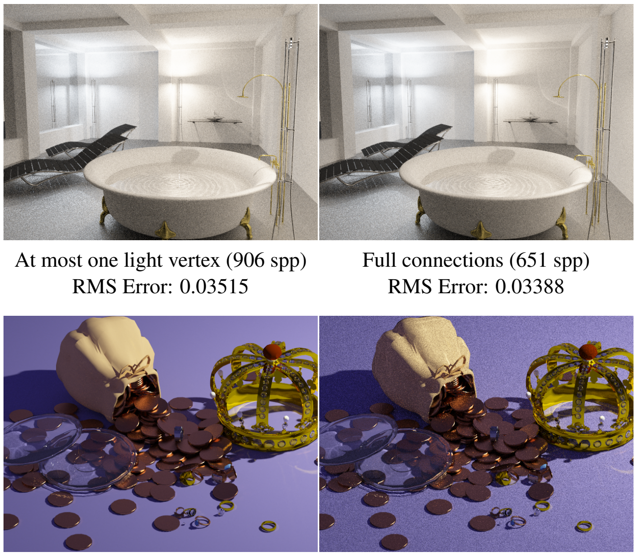
First, we believe that their formulation of the probability density function for photon density estimation is not entirely correct. Their probability density functions are scaled by the area of density estimation kernel, which itself does not give raise to a probability density since the area term is measured by dA, instead of the correct measure dA^{-1}. We have showed that the reciprocal of this area term should be multiplied by the probability density functions of unbiased Monte Carlo path sampling, in order to account for the path space extension required to match the dimensionality of the photon density estimation samples. While we provided an analytic explanation of our formulation, their multiplication of the area term seems not to follow a rigorous mathematical reasoning. Our formulation further provides a clear connection between our framework, photon density estimation, and other techniques including virtual spherical light sources. This is an important difference which opens up the possibility to explore other techniques as future work.

Second, their claim of the overall convergence rate of $O(1/N)$ indicates that their formulation does not take into account the convergence rate of progressive density estimation in multiple importance sampling. As we have described, this is an important factor to consider if we would like to obtain a provably good combination of estimators without considering bias. We have demonstrated also that different alpha values results in different overall convergence rates. The overall convergence rate of $O(1/N)$ cannot simply be achieved if we use progressive density estimation. It is also not clear from their descriptions if their formulation takes into account the difference in the number of samples for density estimation and Monte Carlo path integration in the weight computation. This is another essential factor for obtaining accurate balanced weights.

We would also like to emphasize that none of those previous studies analyzed the effect of bias to a provably good combination in multiple importance sampling, which is one of our theoretical contributions. Our work is also one of the firsts to fully investigate the combination of progressive photon mapping using multiple importance sampling. Our unified path space however is not limited to progressive photon density estimation, but covers general photon density estimation techniques and even more as we have described. The analysis of a provably good combination in the balanced heuristic under the presence of bias (Section 3.2.4) is the only part that is specific to progressive photon density estimation.

6.1 Limitations

We emphasize that the goal of our work is not improving efficiency of each sampling technique, but finding a better combination by introducing a new set of sampling techniques. These two goals are completely orthogonal as demonstrated in some previous work [Kelemen et al. 2002]. For certain light transport that is fundamentally difficult to sample with any of the combined techniques, our method can still be inefficient.



At most one light vertex (906 spp)
RMS Error: 0.03515

Full connections (651 spp)
RMS Error: 0.03388

At most one light vertex (615 spp)
RMS Error: 0.02385

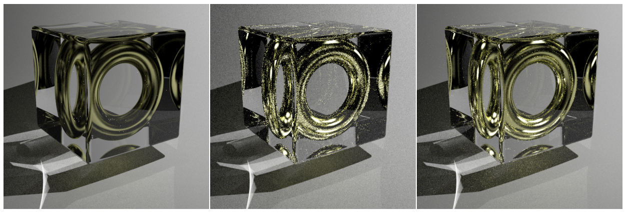
Full connections (224 spp)
RMS Error: 0.05008

Figure 14: *Effect of using full bidirectional connections. The images are rendered in 30 min with/without considering full bidirectional connections between light vertices and eye vertices. Our framework can include all bidirectional connections with an arbitrary number of light vertices, with the presence of photon density estimation. Considering full connections can slightly improve the accuracy of solution in some cases (top two images). However, in other cases, because of its higher computational cost, using full connections can lead to noise due to insufficient sampling rate (bottom two images).*

Figure 15 demonstrates one such example, where a torus with highly glossy material is embedded in a glass cube. The illumination on the torus is due to highly glossy reflections of indirectly visible caustics. In this setting, a light transport path that has a significant contribution to the image can be sampled only by considering multiple highly glossy reflections at once. Neither Monte Carlo path integration nor photon density estimation performs such sampling, since we sample each subpath vertex by local path sampling. Although our sampling framework still improves upon each approach, we simply need many samples in order to render this scene without noise. It is thus interesting as future work to incorporate advanced sampling methods such as Markov chain Monte Carlo sampling [Veach and Guibas 1997; Hachisuka and Jensen 2011] in our framework.

There are a couple of other limitations in its current form. First, while we opted to ignore bias in our method, our derivation shows that bias surely affects a provably good weighting strategy in multiple importance sampling. Indeed, our bias-aware heuristic is a provably good strategy under the presence of a biased estimator. Our further analysis in combination with progressive photon mapping showed that ignoring bias does not significantly affect the optimality of combination when we set $\alpha = 2/3$. We found our decision to be reasonable in practice as we have demonstrated in the paper, but further investigation might lead to a better alternative.

Second, even though our derivation provides a first step in theoretical analysis of the effect of bias in multiple importance sampling, we have made two assumptions that might be restrictive in some cases. The first assumption is that we only have one biased method in a combined solution, and the second assumption is that



MC Integration (RMSE: 0.11989)

Density Estimation (RMSE: 0.11497)

Unified (RMSE: 0.05999)

Figure 15: *Fundamentally difficult case. The scene is a modified version of the torus scene where the surface material of the torus is highly glossy. Most of the illumination on the torus is due to highly glossy reflections of caustics which are indirectly visible through a glass cube. Such paths of light are fundamentally difficult to sample efficiently even with our unified framework. Our algorithm however is still able to produces a more accurate result than the standard approaches in the same rendering time.*

the contribution to bias from each biased sample is constant. We made these assumptions in order to make a theoretical analysis of the weighting function tractable. It would however be desirable if further research could relax some parts of these assumptions by using our derivation as a stepping-stone.

We also have to mention that implementation of the entire framework can be challenging as it subsumes bidirectional path tracing and photon density estimation. This means that engineering effort of implementing our framework is at least equivalent to engineering effort of implementing those two approaches in total. An efficient parallel implementation of our framework will pose several challenges as was the case in bidirectional path tracing [Pajot et al. 2011].

7 Conclusion

We have presented a new sampling framework for a light transport algorithm that combines unbiased Monte Carlo path integration and photon density estimation based on multiple importance sampling. The key idea is to extend the space of Monte Carlo path integration by introducing perturbation of path vertices. This extension provides a unified view of the sampling spaces, and serves as a solid theoretical foundation for the application of multiple importance sampling. We have also provided theoretical analysis on how bias from photon density estimation affects a provably good combination of multiple sampling techniques. In order to avoid the impractical requirement of evaluating bias, we describe how progressive photon density estimation can be used to keep a provably good combination under plausible conditions. We have demonstrated the improved robustness and efficiency of the resulting algorithm in comparison to bidirectional path tracing and progressive photon mapping. We believe that our unified path sampling framework will find many practical applications for photorealistic image synthesis and also lead to further development of robust light transport simulation methods which can efficiently handle all kinds of illumination. As such, our contribution could be summarized as providing a new basis for more robust future physically-based rendering methods.

Acknowledgements

We would like to thank Youichi Kimura (Studio Azurite) for providing us the bathroom model. We also thank Jaakko Lehtinen for his feedback on our initial draft.

References

BEKAERT, P., SLUSSALEK, P., HAVRAN, R. C. V., AND SEIDEL, H. 2003. A custom designed density estimator for light transport. Tech. rep., Max-Planck-Institut für Informatik.

GEORGIEV, I., KŘIVÁNEK, J., AND SLUSALLEK, P. 2011. Bidirectional light transport with vertex merging. In *ACM SIGGRAPH Asia 2011 Sketches*, 27:1–27:2.

HACHISUKA, T., AND JENSEN, H. W. 2009. Stochastic progressive photon mapping. In *ACM SIGGRAPH Asia 2009 Papers*, 141:1–141:8.

HACHISUKA, T., AND JENSEN, H. W. 2011. Robust adaptive photon tracing using photon path visibility. *ACM Transaction on Graphics 30* (October), 114:1–114:11.

HACHISUKA, T., OGAKI, S., AND JENSEN, H. W. 2008. Progressive photon mapping. In *ACM SIGGRAPH Asia 2008 Papers*, 130:1–130:8.

HAŠAN, M., KŘIVÁNEK, J., WALTER, B., AND BALA, K. 2009. Virtual spherical lights for many-light rendering of glossy scenes. In *ACM SIGGRAPH Asia 2009 Papers*, 143:1–143:6.

HECKBERT, P. S. 1990. Adaptive radiosity textures for bidirectional ray tracing. In *ACM SIGGRAPH 1990 Papers*, 145–154.

JENSEN, H. 1996. Global illumination using photon maps. In *Rendering Techniques '96*, 21–30.

KAJIYA, J. T. 1986. The rendering equation. In *ACM SIGGRAPH 1986 Papers*, vol. 20, 143–150.

KELEMEN, C., SZIRMAY-KALOS, L., ANTAL, G., AND CSONKA, F. 2002. A simple and robust mutation strategy for the metropolis light transport algorithm. In *Eurographics 2002*, vol. 21, 531–540.

KNAUS, C., AND ZWICKER, M. 2011. Progressive photon mapping: A probabilistic approach. *ACM Transaction on Graphics 30* (May), 25:1–25:13.

LAFORTUNE, E. P., AND WILLEMS, Y. D. 1993. Bidirectional path tracing. In *Proceedings of Computer Graphics*, 95–104.

PAJOT, A., BARTHE, L., PAULIN, M., AND POULIN, P. 2011. Combinatorial bidirectional path-tracing for efficient hybrid cpugpu rendering. In *Eurographics 2011*, 315–324.

VEACH, E., AND GUIBAS, L. 1994. Bidirectional estimators for light transport. In *Proceedings of the Fifth Eurographics Workshop on Rendering*, 147–162.

VEACH, E., AND GUIBAS, L. J. 1995. Optimally combining sampling techniques for monte carlo rendering. In *ACM SIGGRAPH 1995 papers*, 419–428.

VEACH, E., AND GUIBAS, L. J. 1997. Metropolis light transport. In *ACM SIGGRAPH 1997 papers*, 65–76.

VEACH, E. 1998. *Robust Monte Carlo methods for light transport simulation*. PhD thesis, Stanford University, Stanford, CA, USA. AAI9837162.

VORBA, J., AND KŘIVÁNEK, J. 2011. Bidirectional photon mapping. In *Proceedings of CESCG 2011*, The 15th Central European Seminar on Computer Graphics.

A Bias-Aware Multiple Importance Sampling

A.1 Problem Settings

In order to take bias into account in multiple importance sampling, we consider a biased estimator as an unbiased estimator of a biased solution. This makes it possible to characterize bias as a result of modifications to an original integrand. We denote such modifications by the ith sampling technique as $b_i(x)$. Following the same notation as Veach's, the jth sample from the ith technique has the following contribution:

$$F_{i,j} = \frac{w_i(X_{i,j})(f(X_{i,j}) + b_i(X_{i,j}))}{p_i(X_{i,j})}, \tag{25}$$

where $w_i(X_{i,j})$ is the weight function, $f(X_{i,j})$ is the integrand, $b_i(X_{i,j})$ is the modifications to the original integrand that introduces bias, and $p_i(X_{i,j})$ is the probability density function. The expected (and potentially biased) value from the ith technique is then given by

$$\mu_i = E[F_{i,j}] = \int_\Omega w_i(x)(f(x) + b_i(x))\mathrm{d}\mu(x). \tag{26}$$

Note that $b_i(X_{i,j})$ is not bias itself, but rather the contribution to the bias from the sample $X_{i,j}$. We can describe bias from the ith technique as the difference between the biased solution (i.e. μ_i) and the correct solution;

$$B[F_{i,j}] = E[F_{i,j}] - \int_\Omega w_i(x)f(x)\mathrm{d}\mu(x) = \int_\Omega w_i(x)b_i(x)\mathrm{d}\mu(x). \tag{27}$$

Our goal is to find a weighting strategy which has expected error that is not arbitrarily far away from the truly optimal weighting strategy. This is also what the original balance heuristic achieves. Our contribution is a derivation that shows necessary modifications for the original balance heuristic to keep this optimality in combination with a biased technique.

The goal of the derivation of the original balance heuristic is to minimize variance of the solution. This is because the error of unbiased Monte Carlo techniques is solely characterized by variance. However, in our setting, we also need to take bias into account. We therefore minimize the squared error based on *bias-variance decomposition*:

$$E[(F - \hat{\mu})^2] = V[F] + B[F]^2, \tag{28}$$

where $\hat{\mu}$ is the correct solution, $V[F]$ is variance, and $B[F]$ is bias. Minimizing squared error including bias in general, however, is a very challenging task. This is because bias is a systematic error that happens because of various reasons, and it is often difficult to define general characteristics of bias in order to perform any theoretical analysis on quantities including bias. We therefore make a couple of assumptions that are often reasonable in rendering.

First, we only consider the case where we have one nth biased technique in addition to other $n-1$ unbiased techniques. This can be true in our method if we restrict ourselves to consider only one photon density estimation technique. Second, we assume that the contribution to bias from each sample is constant. This is also reasonable in photon density estimation as we only consider neighboring photons which tend to cause similar error within each radiance estimate. We can thus set $b_i(x) = 0$ for $i \neq n$ and $b_n(x) = b_n$ for $i = n$. Notice that we are overloading the notation of b_n for readability. We can then expand $E[(F - \hat{\mu})^2]$ as follows:

$$\begin{aligned} E[(F - \hat{\mu})^2] &= V\left[\sum_{i=1}^{n} \frac{1}{n_i} \sum_{j=1}^{n_i} F_{i,j}\right] + B\left[\sum_{i=1}^{n} \frac{1}{n_i} \sum_{j=1}^{n_i} F_{i,j}\right]^2 \\ &= \int_\Omega \left(\sum_{i=1}^{n-1} \frac{w_i^2(x)f^2(x)}{n_i p_i(x)} + \frac{w_n^2(x)(f(x)+b_n)^2}{n_n p_n(x)}\right)\mathrm{d}\mu(x) \\ &+ \left(\int_\Omega w_n(x)b_n\mathrm{d}\mu(x)\right)^2 - \sum_{i=1}^{n} \frac{1}{n_i}\mu_i^2, \end{aligned} \tag{29}$$

In the following derivations, we will show how to minimize the sum of the first two terms. The last term $\sum_{i=1}^{n} \frac{1}{n_i}\mu_i^2$ has the same bound as the original derivation by Veach thus the last term is independent from weighting functions.

A.2 Minimizing Squared Error

Even after we made some simplifying assumptions, Equation (29) is still difficult to minimize with respect to w_i, since bias introduced the integral term $\left(\int_\Omega w_n(x)b_n\mathrm{d}\mu(x)\right)^2$. In order to yield the optimal w_i including this term, it seems that we need to solve an integral equation which is often intractable. However, we can show that minimizing a point-wise expression,

$$\sum_{i=1}^{n-1} \frac{w_i^2(x)}{n_i p_i(x)} + \frac{w_n^2(x)(1 + r_n(x))^2}{n_n p_n(x)} + Ar_n^2(x)w_n^2(x) \tag{30}$$

indeed suffices to minimize the full expression of $D[F]$ in Equation (29), where we defined $r_n(x) = \frac{b_n}{f(x)}$ and $A = \int_\Omega \mathrm{d}\mu(x)$ for readability.

We first start by trying to find an alternative expression for $\left(\int_\Omega w_n(x)b_n\mathrm{d}\mu(x)\right)^2$. In order to obtain such an expression, we use the Cauchy-Schwarz inequality and consider the bound of this term;

$$\left(\int_\Omega w_n(x)b_n\mathrm{d}\mu(x)\right)^2 \leq A\int_\Omega w_n^2(x)b_n^2\mathrm{d}\mu(x). \tag{31}$$

Minimizing the bound in general does not minimize the original term since the bound might not have the same extrema as the original function. In our case, however, the bound and the function happen to have extrema at exactly the same points since

$$\frac{\partial}{\partial w_n}\left(\int_\Omega w_n(x)b_n\mathrm{d}\mu(x)\right)^2 = A\frac{\partial}{\partial w_n}\int_\Omega w_n^2(x)b_n^2\mathrm{d}\mu(x). \tag{32}$$

Note that this is possible because of our assumptions on bias. Since the latter is the upper bound of the function, minimizing the bound also minimizes the function given the fact that they have extrema at the same points.

Furthermore, since the sums of two functions $f(x) + g(x)$ and $f(x) + h(x)$ have the same extrema if $\frac{dg}{dx} = \frac{dh}{dx}$, minimizing

$$\int_\Omega \left(\sum_{i=1}^{n-1} \frac{w_i^2(x)f^2(x)}{n_i p_i(x)} + \frac{w_n^2(x)(f(x)+b_n)^2}{n_n p_n(x)} + Aw_n^2(x)b_n^2\right)\mathrm{d}\mu(x) \tag{33}$$

is equivalent to minimizing the corresponding sums in Equation (29). We can yield Equation 30 by dividing the integrand of this equation by $f^2(x)$.

A.3 Bias-Aware Balance Heuristic

Using the method of Lagrange multipliers, the minimum value of Equation 30 is attained when all $n+1$ partial derivatives (n derivatives for w_i and one for λ) of the expression

$$\sum_{i=1}^{n-1} \frac{w_i^2}{n_i p_i} + \frac{w_n^2(1 + r_n)^2}{n_n p_n} + Ar_n^2 w_n^2 + \lambda\left(\sum_{i=1}^{n} w_i - 1\right) \tag{34}$$

are zero. Note that we dropped the notation (x) similar to the Veach's derivation since this is a point-wise minimization of the function. The solution to this equation yields Equation 13.