

CALCULATION OF MIS WEIGHTS FOR BIDIRECTIONAL PATH TRACING WITH PHOTON MAPS IN PRESENCE OF DIRECT ILLUMINATION

S. V. ERSHOV, A. G. VOLOBOY*

Keldysh Institute of Applied Mathematics of RAS,
Miusskaya Sq. 4, Moscow, Russia, 125047

*Corresponding author. E-mail: voloboy@gin.keldysh.ru

DOI: 10.20948/mathmontis-2020-48-8

Summary. Multiple importance sampling (MIS) is a well-known method for noise reduction in Monte-Carlo ray tracing. It weights contributions from merging camera and light paths in different vertices. Since noise strongly depends on these weights, the problem of the optimal choice of weight to reach the minimal noise is very important. For bi-directional Monte-Carlo ray tracing with photon maps (BDPM), different join paths are not statistically independent because several light paths are checked against the same camera path and vice versa. As a result, the optimal weights which minimize the noise functional in the classic Monte Carlo ray tracing and in BDPM are different. In this paper we calculate weights for the simple reduced case of just two strategies, i.e. merging at just two vertices of camera ray. We show that these weights obey an integral equation which is qualitatively different from the well-known MIS formulae for uncorrelated samples. The integral equation is solved analytically in a closed form and one can see that the MIS weights for BDPM algorithm depend on the number of rays and scene geometry. In this paper we also correctly take into account the direct illumination to pixel luminance.

1 INTRODUCTION

Currently simulation of light propagation is widely used not only in optical engineering but also in design of new materials. It is intensively applied in architectural, automotive and aircraft design tasks. If wave effects can be neglected, various kinds of stochastic ray tracing are a good choice. This realm mainly includes Metropolis light propagation and Monte-Carlo ray tracing (MCRT). When an image should be calculated, the classic forward ray tracing from light source is inefficient and various bi-directional modifications [1] are applied. The weak side of all stochastic methods is that their results are noisy. The amplitude of noise depends on the method of ray generation and scattering.

Therefore the problem of the optimal probability distribution of ray scattering has been addressed since long. An important foundation which many modern approaches originated from is the Veach's work [2]. Its theorems about Multiple Importance Sampling (MIS) in Monte-Carlo based methods are still a base for current research. The idea is

2010 Mathematics Subject Classification: 78-04, 65C05, 65C20.

Key words and Phrases: Monte-Carlo ray tracing, photon maps, reduction of noise, multiple importance sampling, weights..

to generate several random numbers, one for each “strategy” (i.e. probability density which admits an efficient generation of samples), and then sum their contributions to the accumulated average with weight (which usually depends on the phase space point). With the optimal choice of weights for each strategy, the noise can be substantially reduced, and [2] suggested the famous “balanced heuristic” and “power heuristic” methods of calculation of weights. A proof had been produced that albeit these weights are only sub-optimal, the resulting noise does not much exceed the absolute minimum. Recently this latter point had been attacked in e.g. [3].

Veach results are based on the theorem which assumes independent samples, but then were applied to the more advanced methods like bi-directional Monte Carlo path tracing (BDPT), bi-directional Monte-Carlo ray tracing with photon maps (BDPM) [4, 5] or their combination [6]. Meanwhile now the “samples” i.e. light paths connecting the source and the camera, happen to be not independent because e.g. in BDPM or “vertex merging” the same light path is merged with many camera paths and vice versa. So the resulting joined full trajectories have common part and thus they are not independent.

In this work we investigate the weights optimal to BDPM and demonstrate the equations for them are qualitatively different from the famous Veach heuristics. This is because for the bidirectional Monte Carlo path tracing method or for BMCRT with photon maps the noise does not follow the rule like in the classic MCRT [7]. Therefore the optimal weights which minimize the noise functional in the classic MCRT and in BDPM are different. Since the BDPM noise is a quadratic functional over the ray contribution as it is shown in [7], it is also quadratic in weights. So while calculation of weights that minimize that noise functional looks mathematically trivial it is not so in practice.

There is an infinite set of weights in bidirectional MCRT with own set of weights for each join path length. The weights from different sets are defined in different functional spaces (they have different number of arguments, i.e. vertices and so on), while all they are “coupled” in the noise functional. There are also problems with ray absorption etc. As a result the optimal weights obey an infinite system of linear integral equations which are extremely complex. And their kernels must be calculated from solving yet other integral equations similar to the “rendering equation” [8] and so on.

We derive the formulae for the optimal MIS weights in case of photon maps, i.e. when each camera ray is merged with several light rays. Unlike the algebraic equations of the Veach’s “balance (or power) heuristic”, these are integral equations which happily admit solution in closed form (i.e. it is an analytic formula which includes several integrals of the input functions). One can also see that they depend not only on the BDFs and light emission distribution, like in the Veach case, but also they depend on the number of rays and scene geometry.

2 BACKGROUND

The issue of optimal MIS weights choosing is very important. It influences on quality of resultant realistic image or, what is the same, on a convergence speed of algorithm. This is why lot of studies is devoted to the issue.

The MIS formulae and heuristic methods for optimal weights calculation were proposed in [2, 9]. The formulae derived there depend on the scattering characteristics (BDFs) and

distribution of light source emission only. Further this approach was applied in many papers of which we should recommend [10].

Some extension was suggested in [11] which operates probability density of the join paths (i.e. rays from the source to the camera obtained by “merging vertices” of the forward and backward MCRT). The number of vertices in the join path and two halves which constitute it is different, because two close ones merge into one. Thus the phase spaces are different, but the author unify them and calculate the density in the space of join paths. Naturally, it is proportional to the importance of the join path, and also contains scale factor proportional to the squared merging radius. What is important here is that this radius can be different in different parts of the scene. As a result, after substitution the density of the join path into Veach-like formulae the authors can investigate dependence on the merging radius. The remaining problem is that this is still a one-sample density which is insufficient to treat “correlation terms” in BDPM [7], so accounting for the integration radius does not help much. Notice that our approach does not use the density of join paths, but only those of the light and camera paths, because the noise is an integral with narrow kernels due to “vertex merging”, and these weakly converge to the delta function so that not causing much problems.

The general idea of the papers [12, 13] is that the forward MCRT ray hits within the integrating sphere around the point of the camera ray. This method is named photon mapping or “vertex merging”. In the same time it is possible to connect the segment point of the camera ray with the forward MCRT point (“vertex connection”). Of course here it is necessary check the shading, etc. In general these are two different bidirectional ray tracing methods. And the authors united those using weights to produce the promising “vertex connection and merging” (VCM) method. But the weights are all the same fractionally rational as those of Veach. Non-locality and non-linearity (outside the Veach’s “power heuristic”) is absent there.

S. Popov and others in [14] proposed original algorithm for generation of full paths connecting camera and light source. They stochastically generated several paths with different weights. After that they chose weights to minimize stochastic noise in resultant image. Their algorithm differs from ours because they neglect statistical dependence between “camera rays of different length”. This sounds reasonable, but even if they are really independent, they are merged with the same light path. The authors claim they have a full derivation which does account for correlations and leads to integral equations, but it is absent in the paper. The initial equations they start from are correct but too complex to treat and the authors simplified the problem by replacing the actual noise by its upper bound.

Naturally, this does not guarantee the weights found do lead to a smaller noise: although the upper bound decreases, the noise itself may even increase.

Several works [15, 16, 17] are devoted to the issue of efficient computation of the MIS weights.

In [18] the problem of calculations of the optimal weights had been considered for a *limited* MIS when we mix contributions from just *two* vertices, M -th and $(M + 1)$ -th of the camera path, when M is fixed. This consideration assumed that there is no direct illumination at the camera vertices up to the last (i.e. the $(M + 1)$ -th), see Section 7 of this paper. While possible in reality, this is a severe limitation, especially when M is large

enough. Current paper eliminates this restriction.

3 LIMITED MIS IN BDPM

The general idea is described in [18]: we trace camera ray until it underwent $M + 1$ *diffuse* events, it never can go further. While the direct and caustic illumination is collected in all of them, *diffuse* one is collected only in the M -th and $M + 1$ -th hits, with weights w_0 and $w_1 = 1 - w_0$ correspondingly. These weights are deterministic functions of $M + 1$ vertices of the *join* trajectory (merged camera and light paths), counting from camera. These weights affect the noise while the limiting accumulated image luminance is the same, and so can be chosen so as to reduce the noise. Details and reasoning produced in [18] up to Section 7 remain still valid. We also assume that $M > 0$ so that the density of camera hits starting from the M -th is *diffuse*. The difference is in Eq. (3) of [18], where the contribution to pixel luminance from merging the camera and light paths neglected the direct illumination while here it will be taken into account.

4 RAY CONTRIBUTION

Here and below all calculations are for **one** pixel. The total flux of all scene lights is assumed 1 to not bother about scaling between the density of photons and irradiance.

Retaining the direct illumination, the contribution¹ to pixel luminance from merging the camera path $\{x_0, x_1, \dots\}$ (counting from the first camera hit x_0) and light path $\{x_0^{(F)}, x_1^{(F)}, \dots\}$ (counting vertices from the light source) is

$$\begin{aligned}
 C &= \sum_{m=0}^M E(x_0, \dots, x_m) L_0(x_{m-1}, x_m) \\
 &+ \sum_{n \geq 2} K(x_M - x_n^{(F)}) w_0(\boldsymbol{\mathcal{X}}, x_{n-1}^{(F)}) E(\boldsymbol{\mathcal{X}}) f(x_{n-1}^{(F)} \rightarrow x_n^{(F)}, x_M \rightarrow x_{M-1}, x_M) \\
 &+ \sum_{n \geq 2} K(x_{M+1} - x_n^{(F)}) w_1(\boldsymbol{\mathcal{X}}, x_{M+1}) E(\boldsymbol{\mathcal{X}}, x_{M+1}) f(x_{n-1}^{(F)} \rightarrow x_n^{(F)}, x_{M+1} \rightarrow x_M, x_{M+1}) \\
 &+ w_1(\boldsymbol{\mathcal{X}}, x_{M+1}) E(\boldsymbol{\mathcal{X}}, x_{M+1}) L_0(x_M, x_{M+1})
 \end{aligned}$$

where

$$\boldsymbol{\mathcal{X}}_m \equiv (x_0, \dots, x_m)$$

is the initial part of the camera path,

$$x \rightarrow y \equiv \frac{y - x}{|y - x|}$$

denotes the *unit* vector from x to y , K is the integration kernel, w_0 and $w_1 = 1 - w_0$ are the weights, $L_0(x, y)$ is the surface luminance in point y under *direct* illumination coming

¹That is, accumulated pixel value increases by C .

from the point x , $f(v, u, x)$ is *diffuse* BDF in luminance factor units at surface point x for illumination direction (towards the surface) v and direction of scattering (away from the surface) u , $E(x_0, \dots, x_m)$ is the energy (or transmission factor in [10] terms) of the camera ray *before* hitting x_m , while for the light ray the energy is always 1. Notice that since the direct component is taken explicitly (via the terms L_0), it is not taken from photon maps to avoid double counting, i.e. the light segments $[x_0^{(F)}, x_1^{(F)}]$ which represent the *direct* illumination do not interact with integration spheres and thus the sums start from $n = 2$.

The “energy” or camera ray, which is also named *transmission factor*, is defined as usual: it is 1 just after leaving the camera, i.e. $E(x_0) = 1$ and then

$$E(x_0, \dots, x_{m+1}) = \mu(x_{m-1}, x_m)E(x_0, \dots, x_m) \quad (1)$$

where

$$\mu(y, x) \equiv \int f(v, x \rightarrow y, x) |(v \cdot n(x))| d^2v, \quad (2)$$

n being the local normal.

Below we shall denote

$$\mathcal{L}_0(x_0, \dots, x_M) \equiv \sum_{m=0}^M E(x_0, \dots, x_m) L_0(x_{m-1}, x_m)$$

which is what the pure path tracing (without using FMCRT) would accumulate. Ray contribution then becomes

$$\begin{aligned} C &= \mathcal{L}_0(\boldsymbol{\mathcal{X}}_M) + w_1(\boldsymbol{\mathcal{X}}_{M+1})E(\boldsymbol{\mathcal{X}}_{M+1})L_0(x_M, x_{M+1}) \\ &+ \sum_{n \geq 2} K(x_M - x_n^{(F)})w_0(\boldsymbol{\mathcal{X}}_M, x_{n-1}^{(F)})E(\boldsymbol{\mathcal{X}}_M)f(x_{n-1}^{(F)} \rightarrow x_n^{(F)}, x_M \rightarrow x_{M-1}, x_M) \quad (3) \\ &+ \sum_{n \geq 2} K(x_{M+1} - x_n^{(F)})w_1(\boldsymbol{\mathcal{X}}_M, x_{M+1})E(\boldsymbol{\mathcal{X}}_{M+1})f(x_{n-1}^{(F)} \rightarrow x_n^{(F)}, x_{M+1} \rightarrow x_M, x_{M+1}) \end{aligned}$$

5 NOISE

In BDPM (with or without weights), the variance of the pixel luminance calculated from N_F forward rays and N_B backward rays (started from the same pixel) obeys the general law [7], see also [18]:

$$\begin{aligned} V &= \frac{1}{N_F N_B} (\langle\langle C^2 \rangle\rangle - \langle\langle C \rangle\rangle^2) + \frac{1 - N_F^{-1}}{N_B} (\langle\langle C^2 \rangle_F \rangle_B - \langle\langle C \rangle\rangle^2) \\ &+ \frac{1 - N_B^{-1}}{N_F} (\langle\langle C^2 \rangle_B \rangle_F - \langle\langle C \rangle\rangle^2) \end{aligned}$$

Here $\langle \cdot \rangle_B$ is the averaging over the BMCRT ensemble for the fixed FMCRT ray and $\langle \cdot \rangle_F$ is the averaging over the FMCRT ensemble for the fixed camera ray. Notice the linear

term $\langle\langle C \rangle\rangle$ is independent from the order of averaging so we drop subscripts here. It is also independent from weights, while $\langle\langle C^2 \rangle\rangle$ and $\langle\langle C \rangle_F^2 \rangle_B$ depend on them.

Averaging over the ensemble of light paths resp. camera paths is

$$\langle \cdot \rangle_F = \int (\cdot) p_F(x_0^{(F)}, x_1^{(F)}, \dots, x_n^{(F)}, \dots) dx_0^{(F)} \dots dx_n^{(F)} \dots \quad (4)$$

$$\langle \cdot \rangle_B = \int (\cdot) p_B(x_0, x_1, \dots, x_{M+1}) dx_0 dx_1 \dots dx_{M+1} \quad (5)$$

where p_F and p_B are the probability densities of the light and camera paths. Since we assume that FMCRT uses Russian roulette to kill rays while keeps ray energy, p_F is *not* normalized. Below we shall sometimes use the spatial and sometimes angular probability densities keeping in mind the obvious relation between differentials

$$d^2(x \rightarrow y) = s(x, y) d^2 y \quad (6)$$

$$s(y, x) \equiv |(x \rightarrow y) \cdot n(x)| \quad (7)$$

where n is the local normal.

While neither explicit form nor the properties of $p_F(\dots)$ are used, for the camera path density we need the recurrence relation. Namely, in BMCRT where direction of the next segment of the camera ray is chosen proportionally to BDF,

$$p_B(x_0, \dots, x_{M+1}) = \tilde{f}(v, u, x_M) |(v \cdot n(x_M))| s(x_M, x_{M+1}) \times p_B(x_0, \dots, x_M), \quad (8)$$

where

$$\begin{aligned} \tilde{f}(v, u, x_M) &\equiv \frac{f(v, u, x_M)}{\mu(x_M, x_{M+1})} \\ v &\equiv x_{M+1} \rightarrow x_M \\ u &\equiv x_M \rightarrow x_{M-1} \end{aligned} \quad (9)$$

is the “normalized” BDF for BMCRT, and the total backward scattering μ was defined in (2).

Notice that when $N_F \rightarrow \infty$ while N_B is fixed the noise does not vanish. This remaining noise $N_B^{-1} (\langle\langle C \rangle_F^2 \rangle_B - \langle\langle C \rangle\rangle^2)$ can be naturally termed the “BMCRT noise”, and $\langle\langle C \rangle_F^2 \rangle_B$ named “BMCRT term”. Similarly, when $N_B \rightarrow \infty$ while N_F is fixed, the remaining noise $N_F^{-1} (\langle\langle C \rangle_B^2 \rangle_F - \langle\langle C \rangle\rangle^2)$ is termed “FMCRT noise” and $\langle\langle C \rangle_B^2 \rangle_F$ is named the “FMCRT term”. The last quadratic average $\langle\langle C^2 \rangle\rangle$ will be named the “cross term”.

Again, like in [18], we neglect the “FMCRT noise”, because for most practical cases it is much smaller than the other two terms. Assuming $N_F \gg 1$, we arrive at the approximate noise law:

$$N_B V \approx N_F^{-1} (\langle\langle C^2 \rangle\rangle - \langle\langle C \rangle\rangle^2) + \langle\langle C \rangle_F^2 \rangle_B - \langle\langle C \rangle\rangle^2 \quad (10)$$

Now let us calculate the weight-dependent quadratic averages $\langle\langle C^2 \rangle\rangle$ and $\langle\langle C \rangle_F^2 \rangle_B$.

5.1 Cross term $\langle\langle C^2 \rangle\rangle$

Averages over the ensemble of light paths (4) of the terms proportional to the 0th or 1st power of kernel are all limited for $S \rightarrow 0$. So in $\langle\langle C^2 \rangle\rangle$ dominating are terms quadratic in kernel. Let us assume that the integration kernel K is uniform within the integration sphere, so that $K^2 = S^{-1}K$ where S is its area. Then

$$\begin{aligned}
 C^2 &= \frac{1}{S} \sum_{n \geq 2} K(x_M - x_n^{(F)}) w_0^2(\boldsymbol{x}_M, x_{n-1}^{(F)}) g_M^2(\boldsymbol{x}_M, x_{n-1}^{(F)}, x_n^{(F)}) \\
 &+ \frac{1}{S} \sum_{n \geq 2} K(x_{M+1} - x_n^{(F)}) w_1^2(\boldsymbol{x}_{M+1}) g_{M+1}^2(\boldsymbol{x}_{M+1}, x_{n-1}^{(F)}, x_n^{(F)}) \\
 &+ \sum_{n \neq n'} K(x_M - x_n^{(F)}) K(x_{M+1} - x_{n'}^{(F)}) \times (\dots) \\
 &+ \sum_{n \geq 2} K(x_M - x_n^{(F)}) K(x_{M+1} - x_n^{(F)}) \times (\dots) \\
 &+ O(1), \\
 g_m(\boldsymbol{x}_m, x_{n-1}^{(F)}, x_n^{(F)}) &\equiv E(\boldsymbol{x}_m) f(x_{n-1}^{(F)} \rightarrow x_n^{(F)}, x_m \rightarrow x_{m-1}, x_m)
 \end{aligned}$$

where we dropped the terms proportional to the product $K(x_M - x_n^{(F)})K(x_M - x_{n'}^{(F)})$ which for $n \neq n'$ and $S \rightarrow 0$ vanishes because in a “normal” scene the light path segment can not have zero length. Therefore

For *different* light path vertices $n \neq n'$ the average of $K(x_M - x_n^{(F)})K(x_{M+1} - x_{n'}^{(F)})$ times a limited function over the light paths is limited for $S \rightarrow 0$. The product $K(x_M - x_n^{(F)})K(x_{M+1} - x_n^{(F)})$ of kernels with different centres is 0 for a.e. camera path, thus the average of a limited function times it over the combined camera and light paths ensemble thus goes to 0 as $S \rightarrow 0$. Therefore, the two first lines strongly dominate for $S \rightarrow 0$.

Now let us applying to them averaging (4) then (5) and taking into account that by virtue of FMCRT

$$p_F(x_{n-1}^{(F)}, x) d^2 x_{n-1}^{(F)} = |(v \cdot n(x))| I_n(v, x) d^2 v = |(v \cdot n(x))| I_n(v, x) s(x, x_{n-1}^{(F)}) d^2 x_{n-1}^{(F)} \quad (11)$$

where n is the local normal, I_n is *angular density* of illumination of the n -th order (i.e. after $n - 1$ scattering events, $n = 1$ is for direct) in x from direction $v \equiv x_{n-1}^{(F)} \rightarrow x$. We then obtain

$$\begin{aligned}
 \langle\langle C^2 \rangle\rangle &\approx S^{-1} \int w_0^2(\boldsymbol{x}_{M+1}) E^2(\boldsymbol{x}_M) f^2(x_{M+1} \rightarrow x_M, x_M \rightarrow x_{M-1}, x_M) \\
 &\quad \times J(x_{M+1} \rightarrow x_M, x_M) p_B(\boldsymbol{x}_M) s(x_M, x_{M+1}) d^2 \boldsymbol{x}_{M+1} \\
 &+ S^{-1} \int w_1^2(\boldsymbol{x}_{M+1}) b(x_M, x_{M+1}) E^2(\boldsymbol{x}_{M+1}) p_B(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \quad (12)
 \end{aligned}$$

where

$$I \equiv \sum_{n \geq 2} I_n \quad (13)$$

is the angular density of the *full* diffuse illumination,

$$J(v, x) \equiv |(v \cdot n(x))| I(v, x) \quad (14)$$

is *diffuse* irradiance and

$$b(x_M, x_{M+1}) \equiv \int f^2(v', x_{M+1} \rightarrow x_M, x_{M+1}) J(v', x_{M+1}) d^2 v' \quad (15)$$

Notice this integral is over the *diffuse* illumination only.

Introducing

$$Q(\boldsymbol{x}_{M+1}) \equiv E(\boldsymbol{x}_M) E(\boldsymbol{x}_{M+1}) p_B(\boldsymbol{x}_{M+1}) \quad (16)$$

and applying to it the recurrence relation (1) one obtains two useful identities:

$$E(\boldsymbol{x}_{M+1}) p_B(\boldsymbol{x}_{M+1}) = E^{-1}(\boldsymbol{x}_M) Q(\boldsymbol{x}_{M+1}) \quad (17)$$

$$E^2(\boldsymbol{x}_{M+1}) p_B(\boldsymbol{x}_{M+1}) = \mu(x_{M-1}, x_M) Q(\boldsymbol{x}_{M+1}) \quad (18)$$

Substituting (18) into (12) we arrive at

$$\begin{aligned} \langle\langle C^2 \rangle\rangle &\approx S^{-1} \int w_0^2(\boldsymbol{x}_{M+1}) f(x_{M+1} \rightarrow x_M, x_M \rightarrow x_{M-1}, x_M) \\ &\quad \times I(x_{M+1} \rightarrow x_M, x_M) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \\ &\quad + S^{-1} \int w_1^2(\boldsymbol{x}_{M+1}) b(x_M, x_{M+1}) \mu(x_{M-1}, x_M) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \end{aligned} \quad (19)$$

Then, since $w_0 = 1 - w_1$,

$$\begin{aligned} \langle\langle C^2 \rangle\rangle &\approx S^{-1} \int f(v, u, x_M) I(v, x_M) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \\ &\quad + S^{-1} \int w_1^2(\boldsymbol{x}_{M+1}) \beta(x_{M-1}, x_M, x_{M+1}) I(v, x_M) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \\ &\quad - 2S^{-1} \int w_1(\boldsymbol{x}_{M+1}) f(v, u, x_M) I(v, x_M) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \end{aligned} \quad (20)$$

where

$$\beta(x_{M-1}, x_M, x_{M+1}) \equiv f(x_{M+1} \rightarrow x_M, x_M \rightarrow x_{M-1}, x_M) + \mu(x_{M-1}, x_M) \frac{b(x_M, x_{M+1})}{I(v, x_M)} \quad (21)$$

5.2 BMCRT term $\langle\langle C \rangle_F^2\rangle_B$

Averaging (3) over the light paths ensemble and using (11), (13), (6) and (14), we have for $S \rightarrow 0$ when the kernel is nearly a delta-function,

$$\langle C \rangle_F \approx \mathcal{L}_0(\boldsymbol{x}_M) + E(\boldsymbol{x}_M)G_0(\boldsymbol{x}_M) + E(\boldsymbol{x}_{M+1})w_1(\boldsymbol{x}_{M+1})L(x_M, x_{M+1}) \quad (22)$$

where:

$$L_d(x_M, x_{M+1}) \equiv \int f(v, x_{M+1} \rightarrow x_M, x_{M+1})J(v, x_{M+1})d^2v \quad (23)$$

is luminance under *diffuse* illumination, $L = L_0 + L_d$ is the *full* luminance and

$$\begin{aligned} G_k(\boldsymbol{x}_M) \equiv & \int w_k(\boldsymbol{x}_{M+1})f(x_{M+1} \rightarrow x_M, x_M \rightarrow x_{M-1}, x_M) \\ & \times J(x_{M+1} \rightarrow x_M, x_M)s(x_M, x_{M+1})d^2x_{M+1} \end{aligned} \quad (24)$$

Squaring (22) and averaging over the BMCRT paths gives

$$\begin{aligned} \langle\langle C \rangle_F^2\rangle_B = & \int \mathcal{L}_0^2(\boldsymbol{x}_M)p_B(\boldsymbol{x}_M)d^2\boldsymbol{x}_M \\ & + \int G_0^2(\boldsymbol{x}_M)\rho(\boldsymbol{x}_M)d^2\boldsymbol{x}_M \\ & + \int w_1^2(\boldsymbol{x}_{M+1})L^2(x_M, x_{M+1})E^2(\boldsymbol{x}_{M+1})p_B(\boldsymbol{x}_{M+1})d^2\boldsymbol{x}_{M+1} \\ & + 2 \int \mathcal{L}_0(\boldsymbol{x}_M)G_0(\boldsymbol{x}_M)E(\boldsymbol{x}_M)p_B(\boldsymbol{x}_M)d^2\boldsymbol{x}_M \\ & + 2 \int w_1(\boldsymbol{x}_{M+1})\mathcal{L}_0(\boldsymbol{x}_M)L(x_M, x_{M+1})E(\boldsymbol{x}_{M+1})p_B(\boldsymbol{x}_{M+1})d^2\boldsymbol{x}_{M+1} \\ & + 2 \int w_1(\boldsymbol{x}_{M+1})G_0(\boldsymbol{x}_M)L(x_M, x_{M+1})\mu^{-1}(x_{M-1}, x_M) \\ & \quad \times E^2(\boldsymbol{x}_{M+1})p_B(\boldsymbol{x}_{M+1})d^2\boldsymbol{x}_{M+1} \end{aligned}$$

where

$$\rho(\boldsymbol{x}_M) \equiv E^2(\boldsymbol{x}_M)p_B(\boldsymbol{x}_M) \quad (25)$$

Applying (17) and (18) this becomes

$$\begin{aligned}
 \langle\langle C \rangle_F^2 \rangle_B &= \int \mathcal{L}_0^2(\boldsymbol{x}_M) p_B(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M \\
 &+ \int G_0^2(\boldsymbol{x}_M) \rho(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M \\
 &+ \int w_1^2(\boldsymbol{x}_{M+1}) L^2(x_M, x_{M+1}) \mu(x_{M-1}, x_M) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \\
 &+ 2 \int G_0(\boldsymbol{x}_M) E^{-1}(\boldsymbol{x}_M) \mathcal{L}_0(\boldsymbol{x}_M) \rho(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M \\
 &+ 2 \int w_1(\boldsymbol{x}_{M+1}) E^{-1}(\boldsymbol{x}_M) \mathcal{L}_0(\boldsymbol{x}_M) L(x_M, x_{M+1}) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \\
 &+ 2 \int w_1(\boldsymbol{x}_{M+1}) G_0(\boldsymbol{x}_M) L(x_M, x_{M+1}) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1}
 \end{aligned}$$

Then,

$$\begin{aligned}
 &\int G_0(\boldsymbol{x}_M) E^{-1}(\boldsymbol{x}_M) \mathcal{L}_0(\boldsymbol{x}_M) \rho(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M \\
 &\quad + \int w_1(\boldsymbol{x}_{M+1}) E^{-1}(\boldsymbol{x}_M) \mathcal{L}_0(\boldsymbol{x}_M) L(x_M, x_{M+1}) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \\
 &= \int E(\boldsymbol{x}_M) \mathcal{L}_0(\boldsymbol{x}_M) L_d(u, x_M) p_B(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M
 \end{aligned}$$

(here we replaced L with I using (30) and recalled normalization $w_0 + w_1 = 1$) we finally arrive at

$$\begin{aligned}
 \langle\langle C \rangle_F^2 \rangle_B &= \int \mathcal{L}_0(\boldsymbol{x}_M) (\mathcal{L}_0(\boldsymbol{x}_M) + 2E(\boldsymbol{x}_M) L_d(u, x_M)) p_B(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M \\
 &+ \int G_0^2(\boldsymbol{x}_M) \rho(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M \\
 &+ \int w_1^2(\boldsymbol{x}_{M+1}) \mu(x_{M-1}, x_M) I^2(v, x_M) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \\
 &+ 2 \int G_0(\boldsymbol{x}_M) \left(\int w_1(\boldsymbol{x}_{M+1}) I(v, x_M) Q(\boldsymbol{x}_{M+1}) d^2 x_{M+1} \right) d^2 \boldsymbol{x}_M
 \end{aligned}$$

Applying the definition (24) to the inner integral in the 4th line we have

$$\begin{aligned}
 \langle\langle C \rangle_F^2 \rangle_B &= \int \mathcal{L}_0(\boldsymbol{x}_M) (\mathcal{L}_0(\boldsymbol{x}_M) + 2E(\boldsymbol{x}_M) L_d(u, x_M)) p_B(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M \\
 &+ \int (G_0(\boldsymbol{x}_M) + G_1(\boldsymbol{x}_M))^2 \rho(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M \\
 &+ \int w_1^2(\boldsymbol{x}_{M+1}) \mu(x_{M-1}, x_M) I^2(v, x_M) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \\
 &- \int G_1^2(\boldsymbol{x}_M) \rho(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M
 \end{aligned}$$

From (24), and normalization $w_0 + w_1 = 1$ it follows that $G_0 + G_1 = L_d(x_M \rightarrow x_{M-1}, x_M)$ thus

$$\begin{aligned}
 \langle\langle C \rangle_F^2 \rangle_B &= \int \mathcal{L}_0(\boldsymbol{x}_M) (\mathcal{L}_0(\boldsymbol{x}_M) + 2E(\boldsymbol{x}_M)L_d(u, x_M)) p_B(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M \\
 &+ \int L_d^2(u, x_M) \rho(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M \\
 &+ \int w_1^2(\boldsymbol{x}_{M+1}) \mu(x_{M-1}, x_M) I^2(v, x_M) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \\
 &- \int G_1^2(\boldsymbol{x}_M) \rho(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M
 \end{aligned} \tag{26}$$

or

$$\begin{aligned}
 \langle\langle C \rangle_F^2 \rangle_B &= \int (\mathcal{L}_0(\boldsymbol{x}_M) + E(\boldsymbol{x}_M)L_d(u, x_M))^2 p_B(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M \\
 &+ \int w_1^2(\boldsymbol{x}_{M+1}) \mu(x_{M-1}, x_M) I^2(v, x_M) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \\
 &- \int G_1^2(\boldsymbol{x}_M) \rho(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M
 \end{aligned} \tag{27}$$

5.3 Resulting noise

Substituting (20) and (26) into (10) we have

$$\begin{aligned}
 N_{BV} &\approx n_F^{-1} \int f(v, u, x_M) I(v, x_M) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \\
 &+ \int (\mathcal{L}_0(\boldsymbol{x}_M) + E(\boldsymbol{x}_M)L_d(u, x_M))^2 p_B(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M \\
 &- (n_F^{-1} + 1) \langle\langle C \rangle \rangle^2 \\
 &+ \int w_1^2(\boldsymbol{x}_{M+1}) a(x_{M-1}, x_M, x_{M+1}) \mu(x_{M-1}, x_M) I(v, x_M) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \\
 &- \int G_1^2(\boldsymbol{x}_M) \rho(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M \\
 &- 2n_F^{-1} \int w_1(\boldsymbol{x}_{M+1}) f(v, u, x_M) I(v, x_M) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1}
 \end{aligned} \tag{28}$$

where

$$\begin{aligned}
 n_F &\equiv SN_F \\
 a(x_{M-1}, x_M, x_{M+1}) &\equiv n_F^{-1} \tilde{f}(v, u, x_M) + n_F^{-1} \frac{b(x_M, x_{M+1})}{L(x_M, x_{M+1})} + L(x_M, x_{M+1})
 \end{aligned} \tag{29}$$

and \tilde{f} was defined in (9). We also used the obvious relation that in absence of specular objects and volumetric absorption angular density of *secondary (diffuse)* illumination equals the *full* radiance of the surface point seen from that direction:

$$I(x_{M+1} \rightarrow x_M, x_M) = L(x_M, x_{M+1}) \quad (30)$$

6 OPTIMAL WEIGHTS

By definition, this are the weights which minimize the noise functional. In our case there is only one independent weight, let it be w_1 . Variation of noise (28) with respect to it is

$$\begin{aligned} \frac{N_B}{2} \delta V &\approx \int \delta w_1 w_1(\boldsymbol{x}_{M+1}) a(x_{M-1}, x_M, x_{M+1}) \mu(x_{M-1}, x_M) \\ &\quad \times I(v, x_M) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \\ &\quad - \int \delta G_1(\boldsymbol{x}_M) G_1(\boldsymbol{x}_M) \rho(\boldsymbol{x}_M) d^2 \boldsymbol{x}_M \\ &\quad - n_F^{-1} \int \delta w_1(\boldsymbol{x}_{M+1}) f(v, u, x_M) I(v, x_M) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \end{aligned}$$

where here and below

$$\begin{aligned} u &\equiv x_M \rightarrow x_{M-1} \\ v &\equiv x_{M+1} \rightarrow x_M \end{aligned}$$

and

$$\delta G_1(\boldsymbol{x}_M) \equiv \int \delta w_1(\boldsymbol{x}_{M+1}) f(v, u, x_M) J(v, x_M) s(x_M, x_{M+1}) d^2 x_{M+1}$$

Expanding,

$$\begin{aligned} \frac{N_B}{2} \delta V &\approx \int \delta w_1(\boldsymbol{x}_{M+1}) (w_1(\boldsymbol{x}_{M+1}) a(x_{M-1}, x_M, x_{M+1}) \mu(x_{M-1}, x_M) \\ &\quad | -n_F^{-1} f(v, u, x_M) - G_1(\boldsymbol{x}_M) \\ &\quad \times I(v, x_M) Q(\boldsymbol{x}_{M+1}) d^2 \boldsymbol{x}_{M+1} \end{aligned}$$

The optimal weight is the one for which the functional V reaches its minimum, i.e. $\delta V = 0$ for an arbitrary δw_1 . This happens if and only if

$$w_1(\boldsymbol{x}_{M+1}) (n_F^{-1} \beta(x_{M-1}, x_M, x_{M+1}) + \mu(x_{M-1}, x_M) I(v, x_M)) = n_F^{-1} f(v, u, x_M) + G_1(\boldsymbol{x}_M)$$

or

$$w_1(\boldsymbol{x}_{M+1}) \left(n_F^{-1} \tilde{f}(v, u, x_M) + n_F^{-1} \frac{b(x_M, x_{M+1})}{I(v, x_M)} + I(v, x_M) \right) = n_F^{-1} \tilde{f}(v, u, x_M) + \frac{G_1(\boldsymbol{x}_M)}{\mu(x_{M-1}, x_M)}$$

Let

$$\begin{aligned} a(x_{M-1}, x_M, x_{M+1}) &\equiv n_F^{-1} \tilde{f}(v, u, x_M) + n_F^{-1} \frac{b(x_M, x_{M+1})}{L(x_M, x_{M+1})} + L(x_M, x_{M+1}) \\ W_1(v, u, x_M) &\equiv \frac{n_F^{-1} \tilde{f}(v, u, x_M)}{a(x_{M-1}, x_M, x_{M+1})} \end{aligned} \quad (31)$$

then

$$w_1(x_0, \dots, x_{M+1}) = W_1(v, u, x_M) + \frac{\tilde{G}_1(x_0, \dots, x_M)}{a(x_{M-1}, x_M, x_{M+1})}$$

where

$$\begin{aligned} \tilde{G}_1(x_0, \dots, x_M) &\equiv \int w_1(x_0, \dots, x_{M+1}) \tilde{f}(v, u, x_M) J(v, x_M) s(x_M, x_{M+1}) d^2 x_{M+1} \\ &= \int w_1(x_0, \dots, x_{M+1}) \tilde{f}(v, u, x_M) J(v, x_M) d^2 v \\ v &\equiv x_{M+1} \rightarrow x_M \end{aligned}$$

Substituting this w_1 into the above definition of $\tilde{G}_1(x_0, \dots, x_M)$ we have

$$\begin{aligned} \tilde{G}_1(x_0, \dots, x_M) &= \int W_1(v, u, x_M) \tilde{f}(v, u, x_M) J(v, x_M) d^2 v \\ &\quad + n_F \tilde{G}_1(x_0, \dots, x_M) \int W_1(v, u, x_M) J(v, x_M) d^2 v \end{aligned}$$

from what it follows that

$$\tilde{G}_1(x_0, \dots, x_M) = \frac{A(u, x_M)}{1 - n_F B(u, x_M)}$$

where

$$A(u, x_M) \equiv \int W_1(v, u, x_M) \tilde{f}(v, u, x_M) J(v, x_M) d^2 v \quad (32)$$

$$B(u, x_M) \equiv \int W_1(v, u, x_M) J(v, x_M) d^2 v \quad (33)$$

Finally

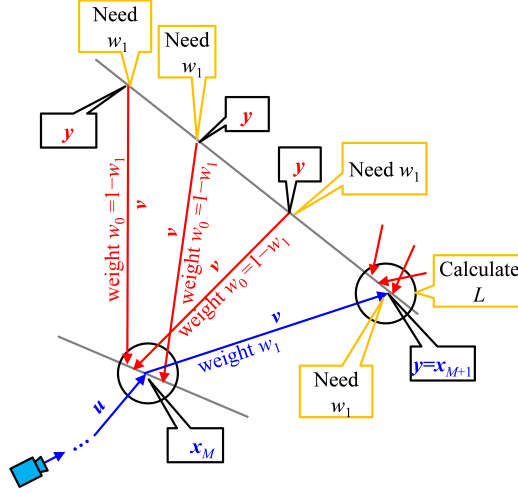


Figure 1: Use of weights in the limited MIS. Blue color relates to camera ray, red color relates to diffuse light ray.

$$w_1(x_{M-1}, x_M, x_{M+1}) = W_1(v, u, x_M) + \frac{A(u, x_M)}{a(x_{M-1}, x_M, x_{M+1})(1 - n_F B(u, x_M))} \quad (34)$$

and it depends only on three vertices (x_{M-1}, x_M, x_{M+1}) of the join path.

7 CALCULATION OF WEIGHT IN RAY TRACING

As seen from (3), for kernel which is S^{-1} inside integration sphere the camera path (x_0, \dots, x_{M+1}) after interaction with *all* light photons increases the luminance of pixel by

$$\begin{aligned} \Delta L &= \sum_{m=0}^M E(x_0, \dots, x_m) L_0(x_{m-1}, x_m) \\ &+ E(x_0, \dots, x_M) \sum_p w_0(x_{M-1}, x_M, x_p) f(v_p, x_M \rightarrow x_{M-1}, x_M) \\ &+ w_1(x_{M-1}, x_M, x_{M+1}) E(x_0, \dots, x_{M+1}) L(x_M, x_{M+1}) \end{aligned} \quad (35)$$

where \sum_p is over the *diffuse* FMCRT \mathbf{p} photons (x_p, v_p) inside the integration sphere around x_M and $L(x_M, x_{M+1})$ is estimation of the *full* radiance at x_{M+1} towards x_M (diffuse component is from photon maps, the direct one being calculated deterministically), calculated as it would be without weights. This use of weights is schematically shown in Figure 1.

We therefore need weight $w_1(x_{M-1}, x_M, y)$ for the the following y :

1. Hit point of the *camera* ray after scattering at x_M
2. *Previous* hitpoint of all *diffuse* FMCRT photons which hit the integration sphere about x_M

For each y the weight is given by (34) which includes *integrals* $b(x_M, y)$ and $L(x_M, y)$. As seen from (15) and (23), they can be estimated with Monte-Carlo method from photon maps:²

$$b(x_M, y) \approx \frac{1}{SN_F} \sum_p f^2(v_p, y \rightarrow x_M; y) \quad (36)$$

$$L(x_M, y) \approx L_0(x_M, y) + \frac{1}{SN_F} \sum_p f(v_p, y \rightarrow x_M; y) \quad (37)$$

where the sums are over the *diffuse* FMCRT \mathbf{p} photons (x_p, v_p) inside the integration sphere around y . These calculations are very much similar to the usual estimation of surface radiance from photon maps. Then (29), (31) give us the base component of weight $W_1(v, u, x_M)$ for this set of directions $v = y \rightarrow x_M$.

Regrettably their result is inevitably *noisy* while (as said in the very beginning) the weights must be *deterministic* functions of the join path so that the weight calculated for given (x_{M-1}, x_M, y) must be *the same* during all the MCRT process. Meanwhile in BDPM the photon maps change from iteration to iteration, thus had that same trajectory (x_{M-1}, x_M, y) been encountered latter, the weight calculated for it would be different.

A simple remedy is to *freeze* the photon map used for calculation of integrals in the weight formula so that it is the same for all iterations. For example, we can always use photon map from the 1st iteration. In this case $L(x_M, x_{M+1})$ is calculated differently: for the 3rd line of (35) it is calculated from the “main” photon maps of current iteration while for the weight it is calculated from the *frozen* map.

The scheme of calculations is shown in Figure 2 (left).

Besides, we need A and B which are independent from y . The integrals (32) and (33) with respect to the measure $J(v, x_M)d^2v$ proportional to the number of the incident photons per surface area also can be estimated with the Monte-Carlo method as:

$$A \approx \frac{1}{SN_F} \sum_p W_1(v_p, u, x_M) \tilde{f}(v_p, u, x_M) \quad (38)$$

$$B \approx \frac{1}{SN_F} \sum_p W_1(v_p, u, x_M) \quad (39)$$

where the sums are again over the FMCRT photons (x_p, v_p) which hit the integration sphere around x_M .

Again to make the result deterministic we use photons from the frozen map. Notice we need W_1 another set of directions than those in (36), (37) because there v_p was from the “main” photon map while now it is from the frozen photon map.

We therefore need to calculate W_1 there, too. This is by the same (36), (37), only for y which is now the previous hit from not the main but the frozen map.

The scheme of calculations is shown in Figure 2 (right).

²The full radiance L , can be calculated either completely from photon maps (using both *direct* and *diffuse* photons), or its *direct* component L_0 can be calculated separately while and only the *diffuse* component (23) is taken from the photon maps

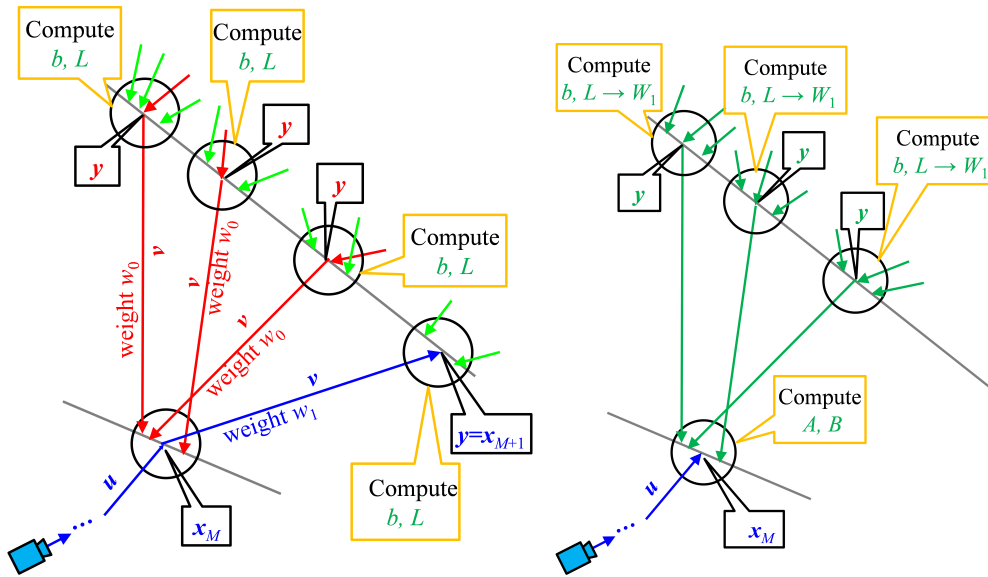


Figure 2: **Left:** Calculation of weights assuming A and B had been already calculated. Blue color relates to camera ray, red color relates to light ray from current photon map and green color relates to light ray from frozen photon map. In each integration sphere we compute b, L from the *frozen* photon map and then calculate w_1 from them. **Right:** Calculation of A and B from the frozen photon map. In each integration sphere we compute b, L from the *frozen* photon map, then calculate W_1 from them and eventually average into A and B .

8 CONCLUSION

The “full” MIS requires that all vertices in the join path be used for intersection of light and camera rays. It happens that even for very simple scenes the problem is very sophisticated. We therefore proposed a compromise sub-optimal approach when only two weights can be not 0. These strategies are: either terminate camera ray at the given vertex or continue it by yet one segment. Then, we treat the weights of these two strategies as functions of the camera path and the last light path vertex but not of the “early” part of light path. In this case the optimal weights obey a linear integral equation that admits solution in close form, i.e. the solution is an analytic formula which though depends on some integrals that must be calculated numerically. We demonstrate how they can be calculated using FMCRT in presence of the direct illumination to pixel luminance. The algorithm does not require too sophisticated calculations and is applicable in practice.

Unlike the widely used Veach heuristic [2], [10] the optimal MIS weights for BDPM algorithm are not local and depend on the scene properties in points outside the current path. Besides, they depend not only on BDFs and distribution of light source emission but also on the number of forward (light) paths traced in one BDPM iteration and on the area of the integration sphere.

REFERENCES

- [1] M. Pharr and G. Humphreys, *Physically Based Rendering: From Theory To Implementation*, 2nd ed., Morgan Kaufmann Publishers Inc., CA, USA (2010).

- [2] E. Veach, *Robust Monte-Carlo methods for light transport simulation*, Dissertation, Stanford Univ. (1997).
- [3] M. Sbert, V. Havran, and L. Szirmay-Kalos, “Multiple importance sampling revisited: breaking the bounds”, *EURASIP Journal on Advances in Signal Processing*, 15(1–15) (2018).
- [4] H. W. Jensen, “Global illumination using photon maps”, *Proceedings of the Eurographics Workshop on Rendering Techniques '96, London, UK*, 21–30 (1996).
- [5] H. W. Jensen and P. Christensen, “High quality rendering using ray tracing and photon mapping”, *ACM SIGGRAPH 2007 Courses, New York, NY, USA*, Course 8 (1–130) (2007).
- [6] N. Dodik, “Implementing probabilistic connections for bidirectional path tracing in the Mitsuba Renderer”, <https://www.cg.tuwien.ac.at/research/publications/2017/dodik-2017-pcbpt/> (Accessed July 17, 2020).
- [7] S. V. Ershov, D. D. Zhdanov, and A. G. Voloboy, “Estimation of noise in calculation of scattering medium luminance by MCRT”, *Math. Montisnigri*, **45**, 60–73 (2019).
- [8] J. T. Kajiya, “The rendering equation”, *Proceedings of the 13th Annual Conference on Computer Graphics and Interactive Techniques SIGGRAPH '86*, 143–150 (1986).
- [9] E. Veach and L. J. Guibas, “Optimally combining sampling techniques for Monte Carlo rendering”, *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques SIGGRAPH '95*, 419–428 (1995).
- [10] J. Vorba, “Bidirectional photon mapping”, *Proceedings of CESC 2011: The 15th Central European Seminar on Computer Graphics, Prague*, 25–32 (2011).
- [11] T. Hachisuka, J. Pantaleoni, and H. W. Jensen, “A path space extension for robust light transport simulation”, *ACM Trans. Graph.*, **31**(6), 191(1-10) (2012). DOI: 10.1145/2366145.2366210.
- [12] I. Georgiev, *Implementing vertex connection and merging*, Tech. rep., Saarland Univ. (2012).
- [13] I. Georgiev, J. Křivánek, T. Davidovič, and P. Slusallek, “Light transport simulation with vertex connection and merging”, *ACM Trans. Graph.*, **31**(6), 192(1–10) (2012).
- [14] S. Popov, R. Ramamoorthi, F. Durand, and G. Drettakis, “Probabilistic Connections for Bidirectional Path Tracing”, *Computer Graphics Forum*, **34**(4), 75–86 (2015).
- [15] D. van Antwerpen, *Recursive MIS computation for streaming bdpt on the GPU*, Tech. rep., Delft Univ. of Technology (2012).
- [16] Q. Liu and Y. Zhang, “Light transport simulation via generalized multiple importance sampling”, arXiv:1803.04305v2 [cs.GR] (2018).
- [17] V. Frolov, A. Kharlamov, V. Galaktionov, and K. Vostryakov, “Multiple reference octrees for a GPU photon mapping and irradiance caching”, *Programming and Computer Software*, **40**(4), 208–214 (2014).
- [18] S. V. Ershov, D. D. Zhdanov, and A. G. Voloboy, “Calculation of MIS weights for bidirectional path tracing with photon maps”, *Proceeding of conference on Computing for Physics and Technology, Moscow*, 20(1–5) (2020).

Received June 20, 2020