

# Calculation of MIS weights for bidirectional path tracing with photon maps

S.V. Ershov<sup>1</sup>, A.G. Voloboy<sup>1</sup>, D.D. Zhdanov<sup>2</sup>, A.D. Zhdanov<sup>2</sup>  
voloboy@gin.keldysh.ru

<sup>1</sup>Keldysh Institute for Applied Mathematics, Moscow, Russia

<sup>2</sup>National Research University of Information Technologies, Mechanics, and Optics, St. Petersburg, Russia

*A widely used method for noise reduction in Monte-Carlo ray tracing is combing different means of sampling, known as multiple importance sampling (MIS). For bi-directional Monte-Carlo ray tracing with photon maps (BDPM), the join paths are obtained by merging camera and light sub-paths, and since several light paths are checked again the same camera path, and vice versa, the join paths obtained are not statistically independent. Thus the noise in this method obeys laws different from those in simple classic Monte-Carlo with independent samples so the weights that minimize that noise must also be calculated differently. This paper drives that weights for the simplest case when we mix contribution from only two vertices of camera ray. It shows that the weights obey an integral equation which is qualitatively different from the well-known MIS formulae for uncorrelated samples. Besides that, even if forget the integral operator, the weights depend on the integration sphere radius and the number of light rays used. The integral equation is solved analytically in a closed form and it is demonstrated how to perform the necessary calculations in BDPM.*

**Keywords:** Monte-Carlo ray tracing, bi-directional ray tracing, photon maps, reduction of noise, multiple importance sampling, weights.

## 1. Introduction

Lighting simulation and calculation of global illumination is widely applied nowadays in architecture, design of new devices and new materials. Mostly successive technique is Monte Carlo ray tracing (MCRT) [1]. But its main problem is stochastic noise, and it strongly depends on the method of ray generation. Therefore a lot of studies are devoted to choice of the optimal probability distribution of ray scattering [2–5].

Multiple Importance Sampling (MIS) is a famous and powerful approach here. Its base idea is 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 point) [2, 3]. E. Veach produced and published the mathematics of this approach [2]. He elaborated several efficient heuristics of weight calculation and demonstrated that with them the resulting noise is close to its minimal value. But his study is applied to the classic MCRT method when successive random points are absolutely independent.

Many lighting simulation systems use more advanced methods like, for example, bi-directional Monte-Carlo path tracing (BDPT), bi-directional Monte-Carlo ray tracing with photon maps (BDPM) [6] or their combination [7, 8]. In these methods the successive trajectories are not independent. For example, in the photon map technique (the BDPM method) the same forward path is merged with all the backward paths and vice versa, so the resulting joined full trajectories have common "tail" and thus they are not independent. In [5] it was demonstrated that the noise in these methods follows other rules than the rule in the simple or classic MCRT. Therefore the optimal weights that minimize the resulting noise are different from the optimal weights which minimize the noise functional in the classic MCRT.

Since the BDPM noise is a quadratic functional over the ray contribution [5], it is also quadratic in weights.

Thus, basically, calculation of weights that minimize that noise functional is mathematically trivial. But only "basically" because in *any* bi-directional MCRT there is an *infinite* set of weights [2], 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 less important problems with ray absorption etc. As a result, the optimal weights obey an infinite system of linear integral equations, which are *extremely* tedious. And their kernels must be calculated from solving yet other integral equations similar to the "rendering equation" and so on.

## 2. BDPM with fixed BDD, adaptive BDD and MIS

Meanwhile in practice the BDPM method can be used even without MIS, but using a *single* strategy when camera and light sub-paths merge at a pre-defined camera vertices. This approach can be called "fixed BDD". BDD means "backward diffuse depth" i.e. the maximally allowed number of diffuse scattering events for camera ray. Here,

- Camera ray terminates after BDD diffuse events, so we have only BDD+1 vertices for merging
- The integration sphere around the last camera vertex "catches" all light rays
- The integration sphere around the rest vertices "catches" only direct and caustic light rays, i.e. those which did not have any diffuse scattering before.

Here it is important that BDD is not adaptive, i.e. it is chosen at the camera ray beginning and does not depend on the further ray history.

This rigid approach is not optimal. It seems better to choose BDD adaptively, depending on the ray history. Roughly speaking, it may not stop at a surface which optical properties are presented by too sharp bi-directional distribution function (BDF). Because otherwise we have to collect diffuse illumination here

and in case of sharp BDF this results in high noise [9]. In other words, it would be good to decide whether to terminate or continue camera path, i.e. to choose between BDD and BDD+1 at each vertex. Then, since both strategies are correct, i.e. give the same mean luminance, we can use both to increase the number of samples and thus reduce noise.

In this paper we shall consider just this simplified problem: “weighting” results from the cases of BDD and BDD+1. Instead of termination at BDD we just set zero weight for BDD+1. This is an extreme case and normally the weight is not exactly 0, thus diffuse illumination is collected at both last (as for BDD+1) and last but one (as for BDD) vertices. The weights now are just two functions. They are calculated so that the variance of calculated image luminance should be minimal. They depend on the camera sub-path.

We derive the integral equation for the weights under these assumptions and obtain its solution in a closed form. One can see from it that even in this simplest case the optimal weights depend on many factors which are absent in the “balance” heuristic [2] for classic MCRT where they depend on BDFs and light source emission only. Then we show how it can be calculated from photon maps.

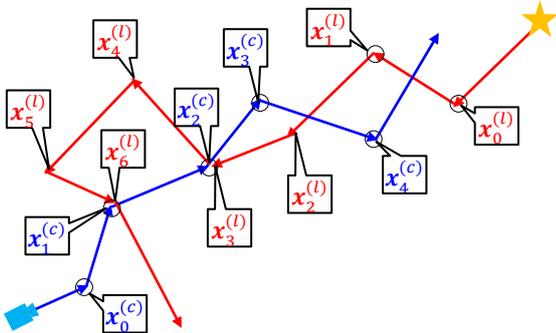
### 3. Weighting the contribution in bi-directional ray tracing

**General note.** Calculations below assume that the total flux of all scene lights is unity. For not unit flux,  $I(\dots)$  and  $L(\dots)$  are irradiance and radiance divided by the total flux.

Regardless of the particular sort of bi-directional MCRT, the application of MIS to it is the same. The contribution to pixel luminance from a light and camera rays is [3]

$$C = \sum_m \sum_n W_{m+n,m} E^{(c)}(\mathbf{x}_0^{(c)}, \dots, \mathbf{x}_m^{(c)}) E^{(l)}(\mathbf{x}_0^{(l)}, \dots, \mathbf{x}_n^{(l)}) \times K(\mathbf{x}_m^{(c)} - \mathbf{x}_n^{(l)}) f(\dots; \mathbf{x}_m^{(c)})$$

where  $m$  cycles over all camera path vertices and  $n$  cycles over all light path vertices,  $K$  is the integration kernel and  $f$  is BDF in luminance units at the point  $\mathbf{x}_m^{(c)}$ .  $E$  is the energy of ray (fig. 1).



**Fig. 1.** Intersection of camera path (blue) and light path (red). Vertices are on diffuse surfaces; the circles are integration spheres. The paths join if a light vertex is with an integration sphere around a camera vertex

Like in [3]  $W_{k,m}$  is the weight for junction at the  $m$ -th camera vertex when the join path of  $k$  vertices (i.e.

the light part of the join path has  $k - m$  vertices). It must be a function of that full path such that

$$\sum_{m=0}^{k-1} W_{k,m} = 1 \quad (1)$$

The weights *may* depend on all  $k$  vertices of the join path, though can be also independent from some of them so we have a very high-dimensional configuration space that it is not good for numerical calculations. Happily it is possible to constrain it. This much simplifies the problem of finding the optimal weights.

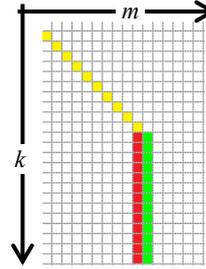
### 4. Weights for fixed BDD (single strategy)

Let us consider the case of fixed BDD= $M$ : direct and caustic rays are taken in vertices up to  $M$ -th; diffuse rays in the  $M$ -th vertex only; further vertices do not collect rays. For the sake of simplicity we assume that there is no specular scattering (caustic rays) in the scene. Then, since a *direct* (light) ray has *single* segment before junction, the weights are:

$$W_{m+1,m} = 1, \quad m = 1, \dots, M-1 \\ W_{k,M} = 1, \quad k > M$$

the rest being 0.

Superimposed tables of weights matrix for BDD= $M$  and BDD= $M+1$  are shown in fig. 2.



**Fig. 2.** Table of weights:  $m$  increases rightwards,  $k$  increases downwards. If  $w_{k,m} = 0$  for both strategies, the cell is *white*; if  $w_{k,m} = 1$  for BDD= $M$ , the cell is *red*; if  $w_{k,m} = 1$  for BDD= $M+1$ , the cell is *green* and if  $w_{k,m} = 1$  for both cases, the cell is *yellow* (green+red)

Obviously normalization (1) holds for both strategies.

### 5. Mixing the two strategies: BDD and BDD+1

We can now *mix* the two above strategies using weights

$$W_{k,m} = \begin{cases} w_0, & m = M \\ w_1 \equiv 1 - w_0, & m = M + 1, \quad k \geq M + 2, \dots \end{cases}$$

where  $w_0$  is the same for all  $k$  and is an *arbitrary* deterministic function of the join path. Then the average pixel luminance calculated with that mixed strategy will be correct which can be shown like in [2] or [3]. We shall also show this simple derivation below.

Since an *arbitrary* function of the full path suits, we can use one which depends only on three vertices  $\mathbf{x}_{M-1}$ ,  $\mathbf{x}_M$  and  $\mathbf{x}_{M+1}$ . With such weights the contribution of camera + light trajectories is

$$C = \sum_{m=0}^M E(\mathbf{x}_0^{(c)}, \dots, \mathbf{x}_m^{(c)}) L_0(\mathbf{x}_m \rightarrow \mathbf{x}_{m+1}; \mathbf{x}_m) + w_0(\mathbf{x}_0^{(c)}, \dots, \mathbf{x}_{M+1}^{(c)}) E(\mathbf{x}_0^{(c)}, \dots, \mathbf{x}_M^{(c)}) \quad (2)$$

$$\begin{aligned}
& \times \sum_{n>0} K(\mathbf{x}_M^{(c)} - \mathbf{x}_n^{(l)}) f(\mathbf{x}_n^{(l)} \rightarrow \mathbf{x}_M^{(c)}, \mathbf{x}_M^{(c)} \\
& \quad \rightarrow \mathbf{x}_{M-1}^{(c)}; \mathbf{x}_M^{(c)}) \\
& + w_1(\mathbf{x}_0^{(c)}, \dots, \mathbf{x}_{M+1}^{(c)}) E(\mathbf{x}_0^{(c)}, \dots, \mathbf{x}_{M+1}^{(c)}) \\
& \times \sum_{n>0} K(\mathbf{x}_{M+1}^{(c)} - \mathbf{x}_n^{(l)}) f(\mathbf{x}_n^{(l)} \rightarrow \mathbf{x}_{M+1}^{(c)}, \mathbf{x}_{M+1}^{(c)} \\
& \quad \rightarrow \mathbf{x}_M^{(c)}; \mathbf{x}_{M+1}^{(c)})
\end{aligned}$$

where  $L_0(\mathbf{u}; \mathbf{x})$  is luminance of a surface point  $\mathbf{x}$  in direction  $\mathbf{u}$  under *direct* illumination and  $\mathbf{x}_{M+1} \rightarrow \mathbf{x}_M$  etc. is unit vector connecting the two vertices.

For the sake of simplicity we assume that FMCRT works so that the light ray energy is always 1 and that  $L_0(\mathbf{u}; \mathbf{x})$  which we assume is calculated exactly (w/o noise!), as it is when there is only a finite number of parallel or point light sources.

## 6. How all this works in BDPM

How is (2) applied in practice? The meaning of this formula is very simple.

First, all works like for the case with fixed BDD+1, i.e. we *always* trace the camera ray to the maximal depth, because so far we do not know whether this vertex will be used or not (because  $w_1=0$ ). The two last vertices are  $\mathbf{x}_M^{(c)}$  and  $\mathbf{x}_{M+1}^{(c)}$ .

Then, we take *all* sources of luminance brought to this camera ray. These are

- photons that hit near  $\mathbf{x}_M^{(c)}$
- photons near  $\mathbf{x}_{M+1}^{(c)}$  (if there is direct then it also at this point)

Then, we process photons near  $\mathbf{x}_M^{(c)}$  like we would do in case of BDD= $M$ , but now the contribution of each photon is scaled by  $w_0(\mathbf{x}_0^{(c)}, \dots, \mathbf{x}_M^{(c)}, \mathbf{y}_p)$  taking the last argument of the weight function as the *previous* hit point  $\mathbf{y}_p$  of that photon. In other words  $\mathbf{y}_p$  is the start of the FMCRT ray segment which ended near  $\mathbf{x}_M^{(c)}$ . Direct and caustic photons, if there are those, are taken with unit weight.

Then, we process photons near  $\mathbf{x}_{M+1}^{(c)}$  like we would do in case of BDD= $M+1$ , but now the contribution of each photon is scaled by  $w_1(\mathbf{x}_0^{(c)}, \dots, \mathbf{x}_{M+1}^{(c)})$ , same for all photons. Same scaling is applied to direct and caustic photons, if there are ones.

And this is all. We therefore need weight function (e.g.  $w_1$ , because  $w_0$  is calculated from it) for the following arguments:

$$(\mathbf{x}_0^{(c)}, \dots, \mathbf{x}_{M+1}^{(c)})$$

and

$$(\mathbf{x}_0^{(c)}, \dots, \mathbf{x}_M^{(c)}, \mathbf{y}_p)$$

where  $\mathbf{y}_p$  are the origins of all diffuse FMCRT ray segments that ended near  $\mathbf{x}_M^{(c)}$ .

## 7. Calculation of noise

Now let us for the sake of simplicity suppose that the direct illumination is negligible. For small  $M$  it is really so in rather many scenes. Then contribution is

$$\begin{aligned}
C &= w_0(\dots) E(\mathbf{x}_0^{(c)}, \dots, \mathbf{x}_M^{(c)}) \sum_{n>0} K(\mathbf{x}_M^{(c)} - \mathbf{x}_n^{(l)}) \\
& \times f(\mathbf{x}_n^{(l)} \rightarrow \mathbf{x}_M^{(c)}, \mathbf{x}_M^{(c)} \rightarrow \mathbf{x}_{M-1}^{(c)}; \mathbf{x}_M^{(c)}) \\
& + w_1(\dots) E(\mathbf{x}_0^{(c)}, \dots, \mathbf{x}_{M+1}^{(c)}) \sum_{n>0} K(\mathbf{x}_{M+1}^{(c)} - \mathbf{x}_n^{(l)}) \\
& \times f(\mathbf{x}_n^{(l)} \rightarrow \mathbf{x}_{M+1}^{(c)}, \mathbf{x}_{M+1}^{(c)} \rightarrow \mathbf{x}_M^{(c)}; \mathbf{x}_{M+1}^{(c)})
\end{aligned} \quad (3)$$

where  $E(\mathbf{x}_0^{(c)}, \dots, \mathbf{x}_M^{(c)})$  is the energy of camera ray before it hits  $\mathbf{x}_M^{(c)}$ . In BMCRT rays starts with  $E(\mathbf{x}_0^{(c)}) = 1$  and then

$$E(\mathbf{x}_0^{(c)}, \dots, \mathbf{x}_{M+1}^{(c)}) = E(\mathbf{x}_0^{(c)}, \dots, \mathbf{x}_M^{(c)}) \mu(\mathbf{x}_{M+1}^{(c)} \rightarrow \mathbf{x}_M^{(c)}, \mathbf{x}_M^{(c)})$$

where

$$\mu(\mathbf{u}, \mathbf{x}) \equiv \int f(\mathbf{v}, \mathbf{u}; \mathbf{x}) |(\mathbf{v} \cdot \mathbf{n}(\mathbf{x}))| d^2\mathbf{v}$$

is the total *backward* scattering.

The noise in (2) i.e. the variance of pixel luminance calculated with it from  $N_F$  forward rays and  $N_B$  backward rays (started from the same pixel) obeys the general law [5]:

$$\begin{aligned}
V &= \frac{1}{N_F N_B} (\langle\langle C^2 \rangle\rangle_B - \langle\langle C \rangle\rangle^2) \\
& + \frac{1 - N_F^{-1}}{N_B} (\langle\langle C^2 \rangle\rangle_B - \langle\langle C \rangle\rangle^2) \\
& + \frac{1 - N_B^{-1}}{N_F} (\langle\langle C^2 \rangle\rangle_B - \langle\langle C \rangle\rangle^2)
\end{aligned} \quad (4)$$

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  $\langle\langle C \rangle\rangle$  is independent from weights. This linear term is independent from the order of averaging so we drop subscripts here.

### Simplified expression of noise

By experience, in the absolute majority of practical cases the first two terms heavily dominate so we can drop the 3<sup>rd</sup>. Then,  $N_F \gg 1$  and we can write the approximate simplified noise law as

$$V = \frac{1}{N_B} \langle N_F^{-1} (\langle C^2 \rangle_F - \langle\langle C \rangle\rangle^2) + (\langle C \rangle_F^2 - \langle\langle C \rangle\rangle^2) \rangle_B \quad (5)$$

Averaging over the illuminating FMCRT rays is simple:

$$\langle (\cdot) \rangle_F = \int (\cdot) |(\mathbf{v} \cdot \mathbf{n}(\mathbf{z}))| I(\mathbf{v}, \mathbf{z}) d^2\mathbf{v}$$

where  $\mathbf{z}$  is the space point where we calculate it and  $I$  is irradiance. Applying it to the ray contribution (3) gives

$$\begin{aligned}
\langle C \rangle_F &= E(\mathbf{X}) \int w_0(\mathbf{X}, \mathbf{x}_{M+1}) f(\mathbf{v}, \mathbf{u}; \mathbf{x}_M) J(\mathbf{v}, \mathbf{x}_M) d^2\mathbf{v} \\
& + E(\mathbf{X}) \mu(\mathbf{u}, \mathbf{x}_M) w_1(\mathbf{X}, \mathbf{x}_{M+1}) L(\mathbf{v}, \mathbf{x}_{M+1})
\end{aligned} \quad (6)$$

where here and below  $\mathbf{x}_{M+1}$  is the *vertex of the join path* after  $\mathbf{x}_M$  so it is light vertex in the first line and camera vertex in the second line;

$$\mathbf{X} \equiv (\mathbf{x}_0, \dots, \mathbf{x}_M)$$

denotes the camera path common for BDD and BDD+1, and we dropped the superscript <sup>(c)</sup> because there are no light ray vertices.

Also here and below  $\mathbf{u}$  is the direction of camera ray

before and  $\mathbf{x}_M$  and  $\mathbf{v}$  is the direction of the *join* path segment after  $\mathbf{x}_M$  (see Figure 3),

$$\begin{aligned}\mathbf{u} &\equiv \mathbf{x}_M \rightarrow \mathbf{x}_{M-1} \\ \mathbf{v} &\equiv \mathbf{x}_{M+1} \rightarrow \mathbf{x}_M\end{aligned}$$

$$J(\mathbf{v}, \mathbf{x}) \equiv |(\mathbf{v} \cdot \mathbf{n}(\mathbf{x}))| I(\mathbf{v}, \mathbf{x})$$

$I(\mathbf{v}, \mathbf{x})$  is irradiance of  $\mathbf{x}$  in direction  $\mathbf{v}$  and  $L(\mathbf{v}, \mathbf{x})$  is radiance from  $\mathbf{x}$  in direction  $\mathbf{v}$ .

Then, assuming that the kernel  $K$  is  $S^{-1}$  within the integration sphere and 0 outside so  $K^2 = S^{-1}K$  we have in the limit  $S \rightarrow 0$

$$\begin{aligned}\langle C^2 \rangle_F &= \frac{E^2(\mathbf{X})}{S} \\ &\times \int w_0^2(\mathbf{X}, \mathbf{x}_{M+1}) f^2(\mathbf{v}, \mathbf{u}; \mathbf{x}_M) J(\mathbf{v}, \mathbf{x}_M) d^2 \mathbf{x}_{M+1} \quad (7) \\ &+ \frac{E^2(\mathbf{X})}{S} \mu^2(\mathbf{u}, \mathbf{x}_M) w_1^2(\mathbf{X}, \mathbf{x}_{M+1}) b(\mathbf{x}_0, \dots, \mathbf{x}_{M+1})\end{aligned}$$

while the cross term is  $o(S^{-1})$ . Here

$$\begin{aligned}b(\mathbf{x}_M, \mathbf{x}_{M+1}) &\equiv \int f^2(\boldsymbol{\omega}, \mathbf{x}_{M+1} \\ &\rightarrow \mathbf{x}_M; \mathbf{x}_{M+1}) J(\boldsymbol{\omega}, \mathbf{x}_{M+1}) d^2 \boldsymbol{\omega}\end{aligned}$$

Now we must average over the ensemble of camera rays. Let us denote the probability density of  $M$  first vertices of camera ray as

$$p_B^{(M)}(\mathbf{x}_0, \dots, \mathbf{x}_M)$$

In a usual BMCRT the distribution of the scattered direction is proportional to BDF in the hit point, so:

$$\begin{aligned}p_B^{(M+1)}(\mathbf{X}, \mathbf{x}_{M+1}) &= p_B^{(M)}(\mathbf{X}) \frac{f(\mathbf{v}, \mathbf{u}; \mathbf{x}_M) |(\mathbf{v} \cdot \mathbf{n}(\mathbf{x}_M))|}{\mu(\mathbf{u}, \mathbf{x}_M)} \\ &\times s(\mathbf{x}_M, \mathbf{x}_{M+1})\end{aligned}$$

where  $s(\dots)$  transforms angular density into spatial, see [2]:

$$\begin{aligned}s(\mathbf{x}_M, \mathbf{x}_{M+1}) &\equiv \frac{|(\mathbf{v} \cdot \mathbf{n}(\mathbf{x}_M))|}{|\mathbf{x}_M - \mathbf{x}_{M+1}|^2} \\ &= \frac{|((\mathbf{x}_M - \mathbf{x}_{M+1}) \cdot \mathbf{n}(\mathbf{x}_M))|}{|\mathbf{x}_M - \mathbf{x}_{M+1}|^3}\end{aligned}$$

Substituting (6) and (7) into (5) and integrating over  $p_B^{(M+1)}(\mathbf{x}_0, \dots, \mathbf{x}_{M+1})$  we arrive at

$$\begin{aligned}N_B V &= n_F^{-1} \int w_0^2(\mathbf{X}, \mathbf{x}_{M+1}) f^2(\mathbf{v}, \mathbf{u}; \mathbf{x}_M) J(\mathbf{v}, \mathbf{x}_M) \\ &\times \rho(\mathbf{X}) d^2 \mathbf{X} d^2 \mathbf{x}_{M+1} \\ &+ n_F^{-1} \int w_1^2(\mathbf{X}, \mathbf{x}_{M+1}) \mu(\mathbf{u}, \mathbf{x}_M) b(\mathbf{x}_M, \mathbf{x}_{M+1}) \\ &\times q(\mathbf{X}, \mathbf{x}_{M+1}) d^2 \mathbf{X} d^2 \mathbf{x}_{M+1} \\ &+ \int G_0^2(\mathbf{X}) \rho(\mathbf{X}) d^2 \mathbf{X} \\ &+ \int w_1^2(\mathbf{X}, \mathbf{x}_{M+1}) \mu(\mathbf{u}, \mathbf{x}_M) L^2(\mathbf{v}, \mathbf{x}_{M+1}) \\ &\times q(\mathbf{X}, \mathbf{x}_{M+1}) d^2 \mathbf{X} d^2 \mathbf{x}_{M+1} \\ &+ 2 \int w_1(\mathbf{X}, \mathbf{x}_{M+1}) L(\mathbf{v}, \mathbf{x}_{M+1}) G_0(\mathbf{X}) \\ &\times q(\mathbf{X}, \mathbf{x}_{M+1}) d^2 \mathbf{X} d^2 \mathbf{x}_{M+1} + \text{const}\end{aligned}$$

where the ‘‘const’’ is independent from weights,

$$n_F \equiv S N_F,$$

$$G_m(\mathbf{X})$$

$$\equiv \int w_m(\mathbf{X}, \mathbf{x}_{M+1}) f(\mathbf{v}, \mathbf{u}; \mathbf{x}_M) J(\mathbf{v}, \mathbf{x}_M) s(\mathbf{x}_M, \mathbf{x}_{M+1}) d^2 \mathbf{x}_{M+1}$$

and

$$\begin{aligned}\rho(\mathbf{x}_0, \dots, \mathbf{x}_M) \\ \equiv E^2(\mathbf{x}_0, \dots, \mathbf{x}_M) p_B^{(M)}(\mathbf{x}_0, \dots, \mathbf{x}_M) s(\mathbf{x}_M, \mathbf{x}_{M+1})\end{aligned}$$

$$\begin{aligned}q(\mathbf{x}_0, \dots, \mathbf{x}_{M+1}) &\equiv f(\mathbf{v}, \mathbf{u}; \mathbf{x}_M) |(\mathbf{v} \\ &\cdot \mathbf{n}(\mathbf{x}_M))| \rho(\mathbf{x}_0, \dots, \mathbf{x}_M)\end{aligned}$$

## 8. Optimal weights

By definition, the optimal weights are those which produce extreme value of noise. Recalling that  $w_0 = 1 - w_1$  we calculate the *variation* of this quadratic expression in response to the change  $w_1 \mapsto w_1 + \delta w_1$ :

$$\begin{aligned}\delta V \\ = \frac{2}{N_B} \int \delta w_1(\mathbf{X}, \mathbf{x}_{M+1}) \Phi(\mathbf{X}, \mathbf{x}_{M+1}) q(\mathbf{X}, \mathbf{x}_{M+1}) d^2 \mathbf{X} d^2 \mathbf{x}_{M+1}\end{aligned}$$

where

$$\begin{aligned}\Phi(\mathbf{X}, \mathbf{x}_{M+1}) &= -n_F^{-1} w_0(\mathbf{X}, \mathbf{x}_{M+1}) f(\mathbf{v}, \mathbf{u}; \mathbf{x}_M) I(\mathbf{v}, \mathbf{x}_M) \\ &+ n_F^{-1} w_1(\mathbf{X}, \mathbf{x}_{M+1}) \mu(\mathbf{u}, \mathbf{x}_M) b(\mathbf{x}_M, \mathbf{x}_{M+1}) \\ &+ w_1(\mathbf{X}, \mathbf{x}_{M+1}) \mu(\mathbf{u}, \mathbf{x}_M) L^2(\mathbf{v}, \mathbf{x}_{M+1}) \\ &+ G_0(\mathbf{X}) (L(\mathbf{v}, \mathbf{x}_{M+1}) - I(\mathbf{v}, \mathbf{x}_M)) \\ &- I(\mathbf{v}, \mathbf{x}_M) \int w_1(\mathbf{X}, \mathbf{x}'_{M+1}) L(\mathbf{v}', \mathbf{x}'_{M+1}) \\ &\times f(\mathbf{v}', \mathbf{u}; \mathbf{x}_M) |(\mathbf{v}' \cdot \mathbf{n}(\mathbf{x}_M))| d^2 \mathbf{x}'_{M+1}\end{aligned}$$

Obviously the extremum condition:  $\delta V = 0$  for an arbitrary  $\delta w_1$  is satisfied if and only if  $\Phi(\mathbf{X}, \mathbf{x}_{M+1}) = 0$ .

Obviously *diffuse* irradiance of a point equals radiance of the surface seen from this point at that direction

$$I(\mathbf{v}, \mathbf{x}_M) = L(\mathbf{v}, \mathbf{x}_{M+1})$$

so  $\Phi(\mathbf{X}, \mathbf{x}_{M+1}) = 0$  implies

$$\begin{aligned}a(\mathbf{v}, \mathbf{u}; \mathbf{x}_M) w_1(\mathbf{x}_0, \dots, \mathbf{x}_{M+1}) - \frac{G_1(\mathbf{x}_0, \dots, \mathbf{x}_M)}{\mu(\mathbf{u}, \mathbf{x}_M)} \\ = n_F^{-1} \tilde{f}(\mathbf{v}, \mathbf{u}; \mathbf{x}_M)\end{aligned}$$

where

$$a(\mathbf{v}, \mathbf{u}; \mathbf{x}) \equiv n_F^{-1} \tilde{f}(\mathbf{v}, \mathbf{u}; \mathbf{x}) + n_F^{-1} \frac{b(\mathbf{x}, \mathbf{y})}{L(\mathbf{v}, \mathbf{y})} + L(\mathbf{v}, \mathbf{y})$$

$\mathbf{y}$  being the hit point of ray fired from  $\mathbf{x}$  in direction  $-\mathbf{v}$  and

$$\tilde{f}(\mathbf{v}, \mathbf{u}; \mathbf{x}) \equiv \frac{f(\mathbf{v}, \mathbf{u}; \mathbf{x})}{\int f(\mathbf{v}, \mathbf{u}; \mathbf{x}) |(\mathbf{v} \cdot \mathbf{n}(\mathbf{x}))| d^2 \mathbf{v}}$$

is ‘‘backward normalized’’ BDF.

This equation admits solution which depends only on  $\mathbf{x}_M$  and camera ray directions before and after it:

$$w_1 = w_1(\mathbf{v}, \mathbf{u}, \mathbf{x}_M)$$

because then

$$\begin{aligned}G_1(\mathbf{u}, \mathbf{x}_M) \\ \equiv \int w_1(\mathbf{v}, \mathbf{u}, \mathbf{x}_M) f(\mathbf{v}, \mathbf{u}; \mathbf{x}_M) J(\mathbf{v}, \mathbf{x}_M) d^2 \mathbf{v}\end{aligned} \quad (8)$$

and  $\mathbf{x}_{M+1}$  can be calculated from  $\mathbf{x}_M$  and  $\mathbf{v}$ . So

$$\begin{aligned}w_1(\mathbf{v}, \mathbf{u}, \mathbf{x}_M) &= \frac{n_F^{-1} \tilde{f}(\mathbf{v}, \mathbf{u}; \mathbf{x}_M)}{a(\mathbf{v}, \mathbf{u}; \mathbf{x}_M)} \\ &+ \frac{\int w_1(\mathbf{v}, \mathbf{u}, \mathbf{x}_M) \tilde{f}(\mathbf{v}, \mathbf{u}; \mathbf{x}_M) J(\mathbf{v}, \mathbf{x}_M) d^2 \mathbf{v}}{a(\mathbf{v}, \mathbf{u}; \mathbf{x}_M)}\end{aligned} \quad (9)$$

This is an integral equation, unlike the ‘‘balanced heuristic’’ from [2]. In practical cases its solution is very expensive although it is mathematically trivial. Substituting  $w_1$  from (9) into (8) we obtain

$$G_1(\mathbf{u}, \mathbf{x}_M) = \frac{\int W_1(\mathbf{v}, \mathbf{u}, \mathbf{x}_M) f(\mathbf{v}, \mathbf{u}; \mathbf{x}_M) J(\mathbf{v}, \mathbf{x}_M) d^2 \mathbf{v}}{1 - n_F \int W_1(\mathbf{v}, \mathbf{u}, \mathbf{x}_M) J(\mathbf{v}, \mathbf{x}_M) d^2 \mathbf{v}}$$

where

$$W_1(\mathbf{v}, \mathbf{u}, \mathbf{x}_M) \equiv \frac{n_F^{-1} \tilde{f}(\mathbf{v}, \mathbf{u}; \mathbf{x}_M)}{a(\mathbf{v}, \mathbf{u}; \mathbf{x}_M)}$$

is the weight ‘‘calculated neglecting  $G_1$ ’’.

Finally,

$$w_1(\mathbf{v}, \mathbf{u}, \mathbf{x}_M) = W_1(\mathbf{v}, \mathbf{u}, \mathbf{x}_M) + \frac{A}{a(\mathbf{v}, \mathbf{u}; \mathbf{x}_M)} \quad (10)$$

where

$$A \equiv \int W_1(\mathbf{v}, \mathbf{u}, \mathbf{x}_M) \tilde{f}(\mathbf{v}, \mathbf{u}; \mathbf{x}_M) J(\mathbf{v}, \mathbf{x}_M) d^2\mathbf{v}$$

$$B \equiv \int W_1(\mathbf{v}, \mathbf{u}, \mathbf{x}_M) J(\mathbf{v}, \mathbf{x}_M) d^2\mathbf{v}$$

Recall that as usual  $\mathbf{u}$  is the direction of camera ray before and  $\mathbf{x}_M$  and  $\mathbf{v}$  is the direction of the *join* path segment after  $\mathbf{x}_M$ .

## 9. Calculation of weight in ray tracing

The general idea is that all the integrals that enter the weight formula are calculated with Monte-Carlo method from photon maps.

In ray tracing  $L(\mathbf{v}, \mathbf{x}_{M+1})$  and  $b(\mathbf{x}_M, \mathbf{x}_{M+1})$  (which is rather similar to  $L(\mathbf{v}, \mathbf{x}_{M+1})$ , just BDF is squared) are calculated as sums over photon hits in the integration sphere.

$$b(\mathbf{v}, \mathbf{x}_{M+1}) \approx \frac{1}{SN_F} \sum_p f^2(\mathbf{v}_p, \mathbf{v}; \mathbf{x}_{M+1}) \quad (11)$$

$$L(\mathbf{v}, \mathbf{x}_{M+1}) \approx \frac{1}{SN_F} \sum_p f(\mathbf{v}_p, \mathbf{v}; \mathbf{x}_{M+1})$$

where the sums are over the FMCRT hits  $(\mathbf{x}_p, \mathbf{v}_p)$  inside the integration sphere around  $\mathbf{x}_{M+1}$ .

Notice we need weight for directions  $\mathbf{v}$  of:

1. All FMCRT photons in the integration sphere about  $\mathbf{x}_M$
2. Camera ray after scattering

Thus we calculate  $L(\mathbf{v}, \mathbf{x}_{M+1})$  and  $b(\mathbf{x}_M, \mathbf{x}_{M+1})$  (which is in fact  $b(\mathbf{v}, \mathbf{x}_{M+1})$ ) for all that directions (or all that  $\mathbf{x}_{M+1}$ , because  $\mathbf{v} = \mathbf{x}_{M+1} \rightarrow \mathbf{x}_M$ ), see Figure 3). Then we have  $W_1(\mathbf{v}, \mathbf{u}, \mathbf{x}_M)$  for all that directions.

Approximate calculation of the integrals A and B is rather simple. We estimate them as

$$A \approx \frac{1}{SN_F} \sum_p W_1(\mathbf{v}_p, \mathbf{u}, \mathbf{x}_M) \tilde{f}(\mathbf{v}_p, \mathbf{u}; \mathbf{x}_M) \quad (12)$$

$$B \approx \frac{1}{SN_F} \sum_p W_1(\mathbf{v}_p, \mathbf{u}, \mathbf{x}_M)$$

where the sums are over the FMCRT hits  $(\mathbf{x}_p, \mathbf{v}_p)$  inside the integration sphere around  $\mathbf{x}_M$ .  $W_1$  for those directions have been all the same calculated above. Notice that direction of the scattered camera ray is not included in that sum although we know  $W_1$  for it, because it has another angular distribution than one needed to estimate this integral.

The scheme of calculation is shown in Figure 3.

Knowing  $A$  and  $B$  and  $W_1$  for all direction of the join path past  $\mathbf{x}_M$ , we calculate  $w_1$  for them and thus weight contributions from continuation of camera ray past  $\mathbf{x}_M$  (with weight  $w_1$ ) and photons near  $\mathbf{x}_M$  (with weight  $1-w_1$ ).

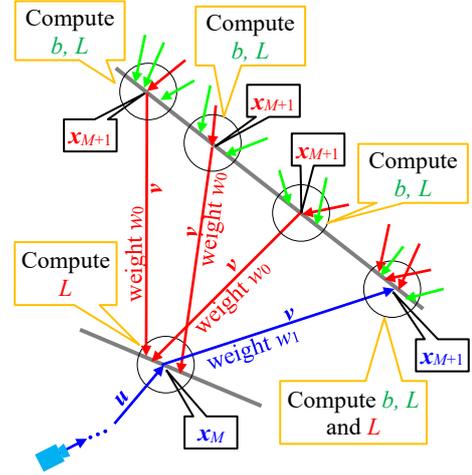
As said in the very beginning weights must be *deterministic* functions of the join path. In other words for given  $(\mathbf{v}, \mathbf{u}, \mathbf{x}_M)$  the weight must be calculated the same during all the MCRT process.

But if we calculate the integrals from photon map

the result will differ from iteration to iteration because photon maps are changed. The 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 1<sup>st</sup> iteration.

Notice that the calculation of luminance brought by the camera ray, described in Section 6, works as usual, i.e. it uses “the main” photon map (from current iteration). The set of arguments  $(\mathbf{v}, \mathbf{u}, \mathbf{x}_M)$  for which we need weights is therefore also taken from this photon map. It is only the sums (11) and (12) which are over the FMCRT hits  $(\mathbf{x}_p, \mathbf{v}_p)$  from the frozen photon map. Meanwhile positions of the integration spheres (that collect those hits) are from the current, iteration-dependent photon map. This is natural because they determine only the ray segment we calculate the weight function for. And if in some next iteration we face the same ray, the weight calculated for it will be exactly the same, thus satisfying the conditions of unbiased estimation.

Fig. 3 illustrates how the calculations are performed.



**Fig. 3.** Calculation of weights for FMCRT rays that brought luminance to given camera ray. 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.  $\mathbf{u}$  is direction of the *join* path segment before  $\mathbf{x}_M$ , and  $\mathbf{v}$  or  $\mathbf{v}$  is direction after it;  $\mathbf{x}_{M+1}$  or  $\mathbf{x}_{M+1}$  is the next (counting from camera) vertex of the *join* path. In each integration sphere we compute  $L$  from the current photon map (used for luminance calculation) and  $b, L$  from the frozen photon map (used for weight calculation)

## 10. Conclusion

Classic MIS requires that all vertices in the join path be used for intersection of light and camera rays, taking contribution from intersection with different weights. The optimal weights minimize the noise functional which in case of BDPM differs from that in the “classic” MCRT [5], so the formulae for the optimal weights are different from those in [2,3]. But they are very sophisticated and it is difficult to apply them in practice.

This paper is an attempt to find a compromise: use

sub-optimal weights to simplify their calculation. Our idea is to weight just two strategies: terminate camera ray at this vertex or continue it by yet one segment. In this case there are only two weights and they are independent from the “early part” of light path. They are “sub-optimal” because the weights in all vertices but two are fixed and independent of the “early part” of light path. So the minimum of noise achieved with them can be reduced further (we hope not much).

We show that in this case the optimal weights obey a linear integral equation that admits solution in close form, i.e. solution is calculated analytically from some integrals that must be calculated numerically. We explain how they can be calculated using FMCRT.

From (10) we see that the optimal weights for this situation

- Are not local i.e. depend on scene properties in points other than current path (via integrals  $b, A, B$ );
  - Depend on BDFs and irradiance;
  - Depend on the number of forward (light) paths traced in iteration;
  - Depend on the area of integration sphere,
- while the “balanced/power heuristic” i.e. the formulae for weights in classic bi-directional ray tracing [2, 3] depend only on BDFs and distribution of light source emission.

In simple extreme cases the weights give “intuitively obvious” result.

- If BDF at  $\mathbf{x}_M$  is *very* sharp,  $w_1 = 1$ , i.e. we go to  $\mathbf{x}_{M+1}$  to collect diffuse illumination.
- If BDF at  $\mathbf{x}_M$  is *smooth* while at  $\mathbf{x}_{M+1}$  either BDF is sharp or illumination is highly nonhomogeneous in space,  $w_1 = 0$ , i.e. we stop at  $\mathbf{x}_M$  and collect all illumination there.
- If the number of light paths  $N_F$  (or integration area  $S$ ) is very large then also  $w_1 = 0$ , i.e. we stop at  $\mathbf{x}_M$ .
- If integration area is very small,  $w_1 = 1$ , i.e. we go to  $\mathbf{x}_{M+1}$  and collect diffuse illumination there.

But not contrast cases, e.g. for BDF at  $\mathbf{x}_M$  and nonhomogeneous illumination at  $\mathbf{x}_{M+1}$  the weight is neither 0 nor 1 and can’t be predicted that simply and (11) is needed.

## Acknowledgments

The study was carried out within the framework of the RFBR grants 18-01-00569 and 20-01-00547.

## References:

- [1] M. Pharr and G. Humphreys. 2010. Physically Based Rendering, Second Edition: From Theory to Implementation (2nd ed.). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- [2] E. Veach. A dissertation: Robust Monte-Carlo methods for light transport simulation, 1997.
- [3] J. Vorba. Bidirectional photon mapping. In Proceedings of CESC 2011: The 15th Central European Seminar on Computer Graphics, Prague, 2011.
- [4] I. Georgiev, J. Křivánek, T. Davidovič, and Ph. Slusallek. 2012. Light transport simulation with vertex connection and merging. ACM Trans.

- Graph. 31, 6, Article 192 (November 2012)
- [5] S. Ershov, D. Zhdanov, and A. Voloboy. Estimation of noise in calculation of scattering medium luminance by MCRT. *Mathematica Montisnigri*, XLV: 60–73, 2019.
- [6] H. W. Jensen, Global illumination using photon maps, in Proceedings of the Eurographics Workshop on Rendering Techniques '96, (London, UK, UK), pp. 21–30, Springer-Verlag, 1996.
- [7] S. Popov, R. Ramamoorthi, F. Durand, and G. Drettakis, Probabilistic Connections for Bidirectional Path Tracing, *Computer Graphics Forum*, 2015.
- [8] N. Dodik, Implementing probabilistic connections for bidirectional path tracing in the Mitsuba Renderer, Sept. 2017.
- [9] S.V. Ershov, D.D. Zhdanov, A.G. Voloboy. Treating diffuse elements as quasi-specular to reduce noise in bidirectional ray tracing // Proceedings of 28<sup>th</sup> International conference on computer graphics and vision GraphiCon-2018, Tomsk, pp. 20-25. URL: <https://www.graphicon.ru/html/2018/papers/20-25.pdf>

## About the authors

Sergey V. Ershov, PhD, senior researcher in the Keldysh Institute of Applied Math of RAS. E-mail: [measure@spp.keldysh.ru](mailto:measure@spp.keldysh.ru)

Alexey G. Voloboy, Doctor of Science in physics and mathematics, leading researcher in the Keldysh Institute of Applied Math of RAS. E-mail: [voloboy@gin.keldysh.ru](mailto:voloboy@gin.keldysh.ru)

Dmitry D. Zhdanov, PhD, associate professor in the Information Technologies, Mechanics and Optics (ITMO) University. E-mail: [ddzhdanov@mail.ru](mailto:ddzhdanov@mail.ru)

Andrey D. Zhdanov, PhD student in the Information Technologies, Mechanics and Optics (ITMO) University. E-mail: [adzhdanov@itmo.ru](mailto:adzhdanov@itmo.ru)