



HAL
open science

A Practical Path Guiding Method for Participating Media

Hong Deng, Beibei Wang, Rui Wang, Nicolas Holzschuch

► **To cite this version:**

Hong Deng, Beibei Wang, Rui Wang, Nicolas Holzschuch. A Practical Path Guiding Method for Participating Media. Computational Visual Media, 2020, 6, pp.37-51. 10.1007/s41095-020-0160-1 . hal-02486014

HAL Id: hal-02486014

<https://inria.hal.science/hal-02486014>

Submitted on 20 Feb 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Practical Path Guiding Method for Participating Media

Hong Deng¹, Beibei Wang¹ ✉, Rui Wang², Nicolas Holzschuch³

© The Author(s) 2015. This article is published with open access at Springerlink.com

Abstract Rendering translucent materials is costly: light transport algorithms need to simulate a large number of scattering events inside the material before reaching convergence. The cost is especially high for materials with a large albedo or a small mean-free-path, where higher-order scattering effects dominate. In simple terms, the paths get lost in the medium. Path guiding has been proposed for surface rendering to make the convergence faster by guiding the sampling process. In this paper, we introduce a path guiding solution to translucent materials. We learn an adaptive approximate representation of the radiance distribution in the volume and use this representation to sample the scattering direction, combining with the phase function sampling by resampled importance sampling. The proposed method significantly improves the performance of light transport simulation in participating media, especially for small lights and media with refractive boundaries. Our method can handle any homogeneous participating media, from highly scattering to low scattering, from highly absorption to low absorption, from isotropic media to highly anisotropic media.

Keywords Monte Carlo, ray tracing, path guiding, volume light transport, participating media rendering.

1 Introduction

Computing illumination simulation in scenes with participating media is still a costly process, even with algorithms specifically designed to handle them. To render

high-albedo materials, such as milk or skin, we need to simulate a large number of scattering events before convergence. Path tracing, as a simple and powerful method, has been used to simulate these materials in the industry recently. However, it takes a very long time to converge. It's even worse for small light sources. The main reason is that sampling the out-scatter direction only depends on the importance sampling of the phase function, and has no knowledge of the place of the light. This results in a lot of low contribution path in the end.

Path guiding has been recently introduced for surface rendering [6, 23, 28, 31]. The common goal is to find an “optimal” distribution that can approximate the actual path integral and make convergence faster. In these prior works, the incoming radiance distribution of some samples is learned and further used by combining with Bidirectional Reflection Distribution Function with multiple importance sampling [23] or product importance sampling [6]. Besides path guiding methods in path space, several works have focused on the primary sample space [5, 24, 32]. Some of these works used machine learning for path guiding. Path guiding makes convergence faster compared to the original path tracing.

Zero-variance methods [20, 22] have been proposed for participating media, but only for isotropic homogeneous media. Recently, it is further improved in a concurrent work [7], which used path guiding for arbitrary homogeneous media based on zero-variance random walk theory.

In this paper, we present a different path guiding method for arbitrary homogeneous participating media. Similar to Müller et al. [23], we use a SD-Tree to represent the incoming radiance distribution in the scene. However, the distribution is sampled inside the volumes rather than on the surfaces, and also learned in a bidirectional manner, considering difficult paths due to the participating media boundaries. We propose a resampling method, combining this incoming radiance distribution with the phase function to better sample the product of two high frequency functions. Our resampling method only requires a couple of samples to approximate the outgoing radiance

1 Nanjing University of Science and Technology, Nanjing, 210094, China. E-mail: hong.deng.bot@njust.edu.cn, beibei.wang@njust.edu.cn.

2 State Key Lab of CAD&CG, Zhejiang University, Hangzhou, 310058, China. E-mail: rwang@cad.zju.edu.cn.

3 Univ. Grenoble-Alpes, Inria, CNRS, Grenoble INP, LJK, 38000 Grenoble, France. E-mail: Nicolas.Holzschuch@inria.fr.

Manuscript received: 2020-02-06; accepted: 2020-02-20.

distribution. For more efficiency, we introduce a selective sampling strategy, choosing the sampling method based on the phase function. We use path guiding to select direction in the medium. Our method results in faster convergence than original path tracing, greatly reducing noise in the pictures without impact on accuracy. On top of our method, denoising methods can further remove the noise. Our method produces a better input for denoising compared to classical path tracing. To summarize, we make the following technical contributions:

- a modified learning method for SD-Tree to represent incoming radiance distribution in the volume,
- a resampled importance sampling method to sample the product of incoming radiance distribution function and phase function,
- a selective sampling strategy to choose the sampling method based on the phase function anisotropy of the media.

In the next Section, we review some of the previous work on rendering participating media. In Section 4, we present path guiding in participating media. We present our results, compare with previous work and analyze performances in Section 5, and conclude in Section 6.

2 Previous Work

2.1 Participating Media Rendering

Path tracing: Simulating light transport in participating media requires simulating multiple scattering events in the medium. Early approaches used Monte-Carlo integration to compute illumination [21, 26, 27].

In recent works, Georgiev et al. [4] used the product of geometry and scattering terms for importance sampling in rendering participating media. Compared to ours, this method is limited to low-order scattering, while we handle both high-order and low-order scattering.

Photon Mapping: Chandrasekhar [3] introduced the *radiative transfer equation*, describing radiation transport in participating media. Jensen and Christensen [16] apply this equation to light transport, and presented an algorithm based on Photon Mapping. Jarosz et al. [12–14] extend this algorithm using beams instead of photons for faster computations, with less noise. Krivánek et al. [19] automatically selects between beams, points and paths in participating media using multiple importance sampling. Bitterli and Jarosz [2] further extend the idea by tracing photon planes and volume.

Photon-mapping-based methods can provide high quality simulations of light transport in participating media, but they are usually biased. In contrast, our method provides

unbiased result.

Radiance Caching: Jarosz et al. [11] introduced radiance caching in participating media and decreased the rendering cost, by storing radiance at sample places in an octree. Marco et al. [17] improved this work by considering the second-order occlusion. Compared with these methods, our method also improves the performance, but in a different way. Furthermore, our method is unbiased, while theirs are biased.

2.2 Path Guiding

Several path guiding methods have been proposed to drive the importance sampling of outgoing ray directions based on learned information. Most of them focus on surface-to-surface interactions and one work presented a path tracing to participating media.

Path Space: Jensen [15] proposed to place regular bins to collect photons traced from light and reconstructs histograms from these bins to guide sampling directions when tracing paths from camera. Vorba et al. [31] learned the incoming radiance distribution of sampled points in the scene by shooting photons and use the trained distribution to guide direction sampling in path tracing. The learned incoming radiance distribution is represented with Gaussian Mixture Model. Müller et al. [23] use an adaptive binary tree in the spatial domain and a quad tree in the angular domain to store the incoming radiance distribution. They use multiple importance sampling of the Bidirectional Reflection Distribution Function (BRDF) and the learned distribution. Herholz et al. [6] used the product of the trained incoming light distribution and BRDF to sample the outgoing direction, resulting in higher sampling quality, at the cost of the product operator.

Simon et al. [28] proposed to use path guiding only for paths with high variance or difficult paths, and use regular path tracing for other paths, as path guiding is more expensive compared to the regular path tracing.

Deep learning has also been used in path guiding. Müller et al. [24] proposed a deep neural network model to present the probability density function of samples. The learned model was leveraged for sampling the ray outgoing direction. The advantage of this work is independence from scenes and local point parameters (e.g. including textured BRDF). However, this method is biased and the sampling is very expensive.

Primary Sample Space: Other researchers have worked on path guiding in Primary Sample Space (PSS) instead of path space. PSS is the space of random numbers that are used to

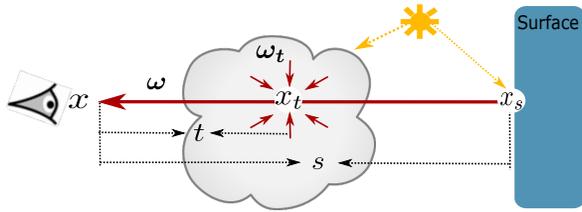


Fig. 1 Light transport in participating media: symbols used in equation 1.

generate paths. PSS has been widely used in Metropolis Light transport algorithms [30]. Guo et al. [5] proposed a path guiding method in PSS. The random numbers are sampled from an trained structure, kd tree, to generate the paths. Müller et al. [24] and Zheng and Zwicker[32] provided PSS based methods for path sampling using deep learning.

One issue of the PSS based methods is that they can not handle long path very well, as the random number count depends on the path length. We did not use them for participating media as these materials result in very long paths, due to many scattering events.

2.3 Zero-Variance Based Methods

Contrary to path space and primary sample space path guiding methods who use data-driven models or neural networks to guide sampling, zero-variance random walks rely on analytical models for sampling.

Křivánek and d’Eon [20] used zero-variance random walk in dense, isotropic and highly scattering media, by fitting a slab to the bounding geometry, which helps to guide the random walks back to the surface. Later, Meng et al. [22] improved it, considering thin geometry or back-lit cases.

Herholz et al. [7] used zero-variance path sampling theory and exploited the vMF model to represent the involved radiance distributions to guide all the sampling strategy in volumes including distance, direction, russian roulette and splitting. Compared to their work, we use different representation and only focus on direction sampling.

3 Background

3.1 Radiative Transfer Equation

We consider a scene containing objects with translucent material. Each of these is assumed to be made of an homogeneous material, with index of refraction η , scattering coefficient σ_s , absorption coefficient σ_a and phase function $p(\vec{\omega}, \vec{\omega}_t)$. We note ℓ the mean-free path inside the material (mfp), with $1/\ell = \sigma_t = \sigma_s + \sigma_a$.

Radiance leaving point x in the direction $\vec{\omega}$ is the sum of exitant radiance from the nearest surface along this direction

and in-scattered radiance from the medium among the whole length of the ray [3] (see Figure 1):

$$L(x, \vec{\omega}) = T_r(x \leftrightarrow x_s)L(x_s, \vec{\omega}) + \int_0^s T_r(x \leftrightarrow x_t)\sigma_s(x_t)L_i(x_t, \vec{\omega})dt, \tag{1}$$

where T_r is the transmittance, defined as:

$$T_r(x \leftrightarrow x_t) = e^{-\sigma_t \|x-x_t\|}, \tag{2}$$

s is the distance through the medium to the nearest surface at $x_s = x - s\vec{\omega}$, and $x_t = x - t\vec{\omega}$ with $t \in (0, s)$. $L(x_s, \vec{\omega})$ is the exit radiance from the nearest surface, which is governed by the *rendering equation* [18]. $L_i(x_t, \vec{\omega})$ is the in-scattering radiance at x_t from all direction $\vec{\omega}_t$ over the sphere of directions $\Omega_{4\pi}$ using the phase function p , defined as:

$$L_i(x_t, \vec{\omega}) = \int_{\Omega_{4\pi}} p(\vec{\omega}, \vec{\omega}_t)L(x_t, \vec{\omega}_t)d\vec{\omega}_t. \tag{3}$$

3.2 SD-Tree Path Guiding

Our proposed method is based on the practical path guiding method proposed by Müller et al. [23], thus we give a detailed review. They used a binary tree to represent the spatial distribution and quad trees to represent the angular distribution of incoming radiance, so we call it *SD-Tree Path Guiding*.

They learned an approximate representation of the scene’s spatio-directional radiance field in an unbiased and iterative manner. To that end, they propose an adaptive spatio-directional hybrid data structure, referred to as SD-Tree, for storing and sampling incident radiance. The SD-tree consists of an upper part — a binary tree that partitions the 3D spatial domain of the light field and a lower part—a quadtree that partitions the 2D directional domain.

They use an iterative scheme similar to the one proposed by Vorba et al. [31]. They trained the incoming radiance sequence L_1, L_2, \dots, L_M , where L_1 was estimated with just BSDF sampling, and for all $k > 1$, L_k was estimated by combining samples of L_{k-1} and the BSDF via multiple importance sampling. Two SD Trees are kept at the same time, while one is used to represent L_{k-1} and the other is used for updating the incoming radiance distribution in the k_{th} iteration.

For a given shading point, two steps are performed to sample the outgoing direction of reflected ray: first, they descend spatially through the binary tree to find the leaf node containing the shading point position; next, they sample the outgoing direction from the quadtree contained in the spatial leaf node via hierarchical sample warping.

The SD-Tree method [23] can not be used for participating media directly, for two reasons. First, the SD-Tree is learned from the camera side, this leads to very expensive learning costs for participating media with boundaries. In figure 5, we use the unmodified SD-Tree

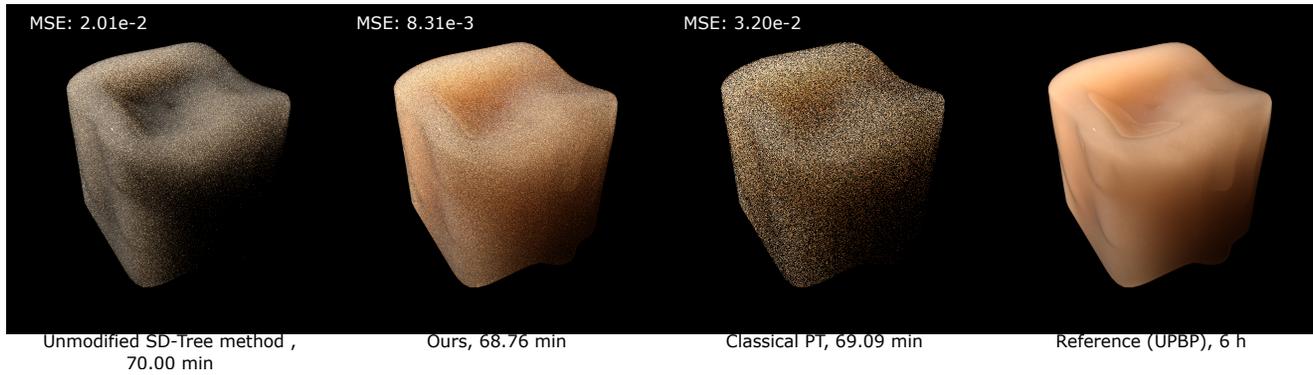


Fig. 2 Comparison between SD-Tree path guiding [23] and our method, with equal time, on the Candle scene. The SD-Tree method has difficulty finding paths connecting to the light source and produces a darker image.

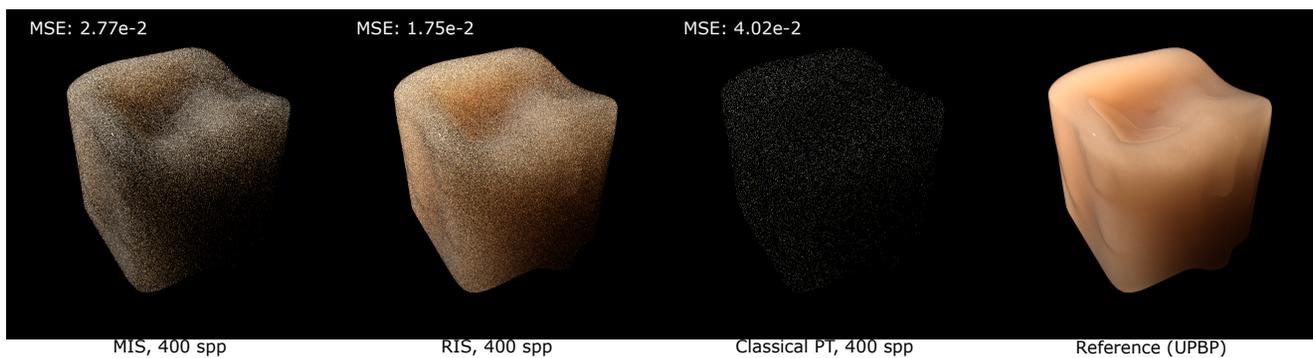


Fig. 3 Comparison between multiple importance sampling and our RIS with equal SPP (400) on the Candle Scene.

method [23] on the Candle scene. It produces a significantly darker result than the reference, due to the difficulty in finding the path to the light source from the camera side. Second, multiple importance sampling is not efficient to sample the product of two high-frequency functions (the incoming light distribution and the phase function). Figure 3 illustrates this issue: multiple importance sampling produces a darker image than resampling. This issue also appears without participating media, although it is less visible. Figure 4 shows a comparison between multiple importance sampling and resampling with high-frequency BRDFs and illumination: resampling reduces the noise in the picture compared with multiple importance sampling.

In this paper, we extend this method to participating

media and solve these issues.

3.3 Resampled Importance Sampling

Monte Carlo integration of a function $f(\vec{\omega})$ is defined:

$$\hat{I} = \frac{1}{M} \sum_{j=1}^M \frac{f(\vec{\omega}_j)}{q(\vec{\omega}_j)}$$

, where M is the sample count and q is the probability density function.

Ideally, the optimal sampling method is sampling q with the same distribution as f , which results in zero variance and error. However, this is impractical. $q(\vec{\omega})$ may not be sampled. Talbot et al [29] introduced resampled importance sampling (RIS) to approximately sample $q(\vec{\omega})$ by firstly

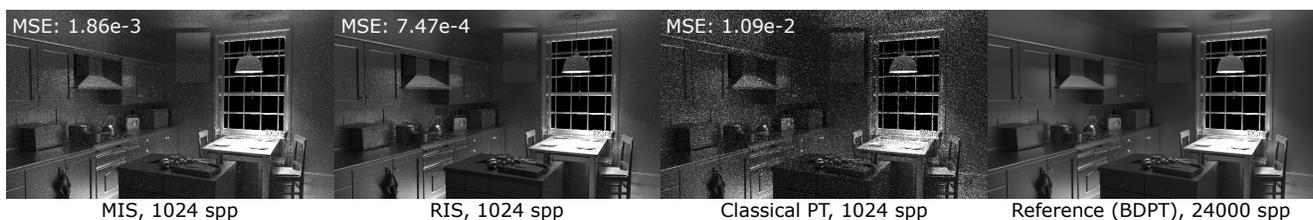


Fig. 4 Comparison between multiple importance sampling and resampling on the Kitchen scene, with highly glossy surfaces. Resampling reduces the noise even without participating media.

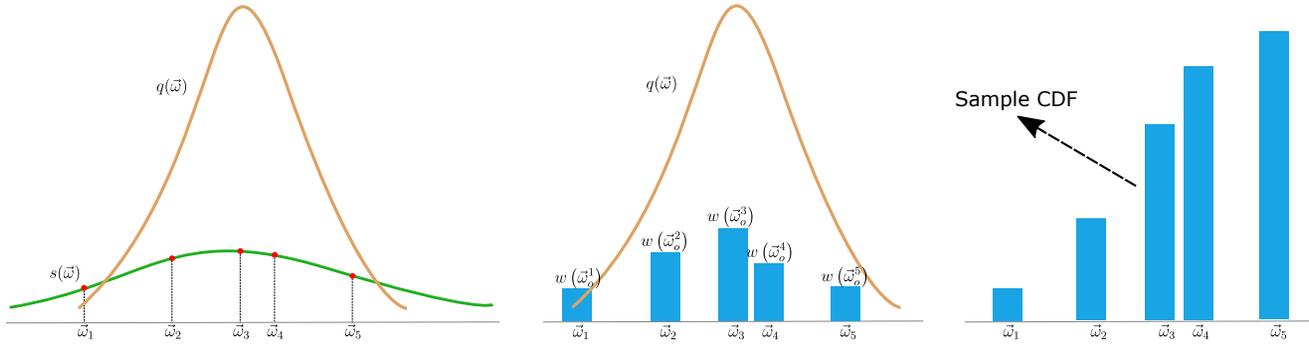


Fig. 5 To sample target function (orange curve), we first choose a function as the source function (green), and then sample the source function and get 5 outgoing direction candidates. In the second step, we evaluate the weight of each candidate which is target function divided by source function. Finally, we compute the Cumulative Distribution Function (CDF) of the weights of the candidates and use it to get the final sample.

sampling another function $s(\vec{\omega})$ which can be sampled. RIS is a robust variance reduction technique to sample a function.

We define some notations at first. $q(\vec{\omega})$ is called target function, and $s(\vec{\omega})$ is called source function, which can be arbitrary function which can be sampled.

And the entire RIS algorithm runs as follows (see Figure 5):

1. Sample N candidates ($N \geq 1$) from $s(\vec{\omega}_i)$.
2. Calculate weight for each candidate: $w(\vec{\omega}_o^i) = \frac{q(\vec{\omega}_i)}{s(\vec{\omega}_i)}$.
3. Choose the final sample from the candidates with probability proportional to the candidate weights $w(\vec{\omega}_o^i)$.

A weighting term W is added to the standard Monte Carlo estimator for unbiasedness:

$$W = \frac{1}{N} \sum_{i=1}^N w(\vec{\omega}_o^i).$$

Combining the weighting term W , we get the RIS Monte Carlo estimator:

$$\hat{I}_{ris} = \frac{1}{M} \sum_{j=1}^M \frac{f(\vec{\omega}_j)}{q(\vec{\omega}_j)} \cdot \frac{1}{N} \sum_{i=1}^N w(\vec{\omega}_o^i).$$

In Figure 6, we compare importance resampling and multiple importance sampling to sample two function product. Given the analytic formula of two functions, we know the target function. By comparison, we can see RIS provides closer curve to the target function, thus RIS produces higher sampling quality than MIS.

4 Our Algorithm

We presents a path guiding method for participating media. We use a SD-tree structure to represent the incident light field (Section 4.1). Our method has two steps: in the training step, the distribution is learned by shooting light rays; in the rendering step, we sample the out-scattered

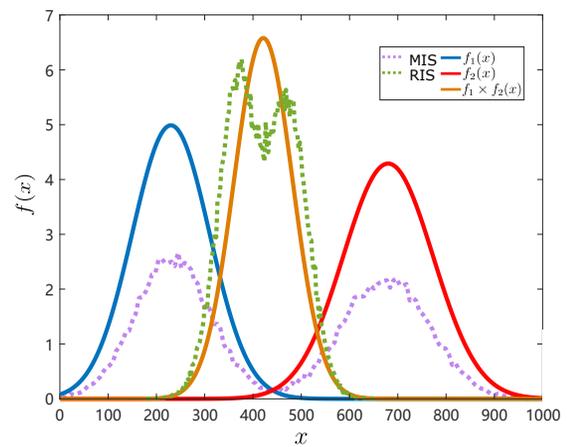


Fig. 6 Comparison between multiple importance sampling and RIS of two function f_1 and f_2 product. The RIS produces result much closer to the target function (product of two functions) than MIS.

direction (Section 4.2) using importance resampling of the trained distribution and the phase function (Section 4.3). To further improve the efficiency of our method, we propose a method to select the optimal sampling approach between multiple importance sampling and importance resampling, depending on the anisotropy of the phase function (Section 4.4).

4.1 Volume Representation for Radiance Distribution

We propose a representation for incoming radiance distribution in participating media with SD trees: a spatial tree for discrete samples in the volume and a directional tree in each leaf node for angular distribution.

For each media, the spatial tree is built starting form the axis aligned bounding box (AABB) of the scene. Nodes are divided into two child nodes when the contained sample count is larger than a threshold. The division dimension is chosen iteratively. The threshold depends on the iteration

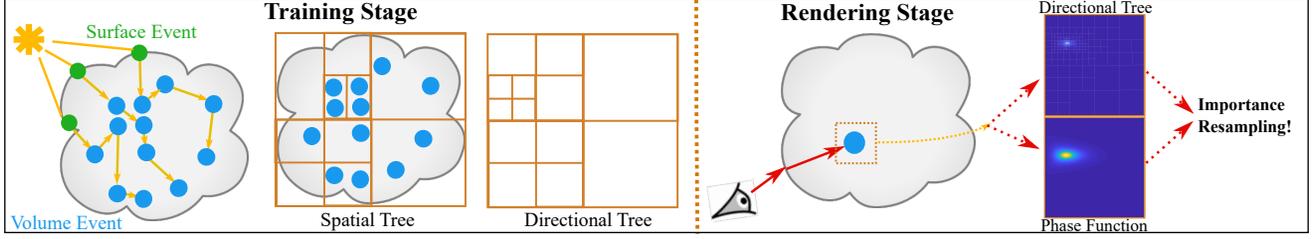


Fig. 7 Our algorithm runs in two steps: training step and rendering step. In the training step, we train the incoming radiance distribution iteratively. In each iteration, the light rays are shot from light sources, scattered in the participating media, and stored in a spatial-directional hierarchy. In the rendering step, the camera rays are scattered in the media, while the outgoing direction sampling method are chosen by considering the anisotropy of the phase function. For high anisotropic media, the outgoing directions are sampled via the resampling product of the learned incoming radiance and the phase function. For low anisotropic media, the outgoing directions are sampled via multiple importance sampling of the learned incoming radiance and the phase function.

(see Section 4.2). The total count of STree node is bounded by the memory budget, or precisely the total STree node count (N).

When arriving the leaf node, a directional node is built starting from a sphere, and subdivided into four child nodes. Each node is subdivided when its contained energy is larger than 1% of the entire energy in the entire directional tree.

4.2 Learning and Rendering

Our method runs in two steps: a learning step and rendering step. Figure 7 shows an overview of our method.

In the learning step, we learn the incoming radiance distribution of sampled position in the volume from the light side. Participating media usually has refractive boundaries. If the path starts from the camera, it's difficult to reach the light sources, especially when the light sources are small. On the other side, it's easier in a bidirectional manner. Further more, the learned distribution from the light side is view independent, so it can be reused when only camera is modified.

We train the incoming radiance distribution iteratively, use $c \times 2048$ photon count in the first iteration and increase photon count as a factor of 2. In each iteration, we shoot rays from the light sources by sampling the light sources. The rays are traced in the scene until getting refracted into the media and then start the random walk in the media. For each scattering event, the outgoing direction is got by importance sampling the phase function. For the STree, the leaf node is subdivided, when the leaf node contains more than $c \times \sqrt{2^k}$. c is a constant value related to the resolution of the DTree, and set as 12000, similar to [23]. k represents the iteration count. The DTree is also updated according to the rule described in Section 4.1. The maximum iteration count is set as 3-6 for all our test scenes.

In the rendering step, following the classical path tracing, we shoot rays from the camera, and trace them in the scene. When the rays arrive at the volume, for each scattering event, we first traverse the spatial tree to find the leaf

node that contains the scattering event. Then we use the directional tree in the leaf node for direction sampling. The outgoing direction is computed by a combination of importance resampling and multiple importance sampling of the learned incoming direction and the phase function. We will show the details about our importance resampling method in Section 4.3.

4.3 Resampling of Incoming Radiance Distribution and Phase Function

In the rendering step, we have two known functions: the phase function $p(\vec{\omega}_i, \vec{\omega}_o)$ and the learned incoming radiance distribution function $\hat{L}_i(\mathbf{x}, \vec{\omega}_o)$ (we use $\vec{\omega}_o$ from the point view of camera) at position \mathbf{x} . We use $\hat{L}_i(\vec{\omega}_o)$ for short. Our goal is to importance sample the direction $\vec{\omega}_o$ considering the product of the two functions $p(\vec{\omega}_i, \vec{\omega}_o) \times \hat{L}_i(\mathbf{x}, \vec{\omega}_o)$, called the target function.

We use importance resampling approach [29] to approximate the product sampling of the two functions (see Figure 6):

1. use weighted addition of phase function and incoming radiance distribution function as the source function, sample of the source function and get n outgoing direction candidates, $\vec{\omega}_o^k$
2. evaluate the weight of each candidate (Equation 4),
3. compute the Cumulative Distribution Function (CDF) of the weights of the candidate out-scattering direction and use it to importance sample the final out-scattered direction.
4. evaluate the scattering event for the chosen direction (Equation 5).

The weight of each sample is defined:

$$w(\vec{\omega}_o^k) = \frac{q_{\hat{L}_i}(\vec{\omega}_o^k) \times q_p(\vec{\omega}_i, \vec{\omega}_o^k)}{\lambda \times q_p(\vec{\omega}_i, \vec{\omega}_o^k) + (1 - \lambda) \times q_{\hat{L}_i}(\vec{\omega}_o^k)}, (1 \leq k \leq n), \quad (4)$$

where k represents the k_{th} candidate, $q_{\hat{L}_i}$ represents the pdf of incoming radiance distribution function, q_p represents the pdf of phase function and λ is the weight of the phase function (set as 0.9 in all of our test scenes).

In our implementation, with only 6 candidates, our method can provide convincing results.

After choosing the out-scattering direction $\vec{\omega}_o$, we evaluate the scatter event:

$$f(\vec{\omega}_i, \vec{\omega}_o) = \frac{p(\vec{\omega}_i, \vec{\omega}_o) \times t}{q_{\hat{L}_i}(\vec{\omega}_o) \times q_p(\vec{\omega}_i, \vec{\omega}_o)} \quad (5)$$

where t is the weight computed as:

$$t = \frac{\sum_{k=0}^n w(\vec{\omega}_o^k)}{n}. \quad (6)$$

4.4 Selective Importance Sampling

On top of the importance resampling method, we propose a selective importance sampling method, combining multiple importance sampling (MIS) and resampled importance sampling (RIS), based on the anisotropy of the phase function. We use MIS for low anisotropic media (g smaller than 0.5) and RIS for high anisotropic media (g larger than 0.5).

For MIS, we sample the phase function with probability μ and the incoming radiance distribution function with probability $1 - \mu$, where μ is set as 0.5 for all the test scenes. The details of RIS can be found in Section 4.3.

The main reason for the selective importance sampling is that MIS method can achieve faster convergence than RIS for isotropic media or very low anisotropic media but suffers from high variance for high anisotropic media.

5 Results and Discussion

We have implemented our algorithm inside the Mitsuba Renderer [10]. We compared our algorithm against original path tracing and UPBP (Unified Points, Beams and Paths) [19] for quality validation. We use UPBP as the reference in most cases and Path tracing for the Bathroom Scene and Candle Scene with rough boundaries. All timings in this section are measured on a 2.20GHz Intel i7 (20 cores, 40 threads) with 32 GB of main memory.

All materials in our scenes are homogeneous materials, with Henyey-Greenstein phase functions and refractive boundaries. Material properties are from Křivánek *et al.* [19], Narasimhan *et al.* [25] and Holzschuch [8] (see Table 1).

We measure the convergence rate with Mean Square Error (MSE).

5.1 Qualitative Validation

We first compare our method with the reference solution and also classical Path Tracing with equal time in Figure 8,

Name	α			ℓ			g
	R	G	B	R	G	B	
Oil	0.004	0.454	0.100	9.71	11.63	2.74	0.9
Wax	0.980	0.962	0.750	0.65	0.63	0.59	0.8
B. Sphere	0.955	0.677	0.457	4.55	3.23	2.17	0.8
Marble	0.999	0.998	0.997	0.46	0.38	0.33	0.0
Fog	0.973	0.973	0.973	12.16	12.16	12.16	0.3

Tab. 1 Parameters for the materials used in this paper.

9, 11, 12, 13 and 14. To valid the ability to handle high frequency effects, we also provide a comparison between our method and path tracing for Bumpysphere Scene with single scattering only in Figure 10. In all these test scenes, our method provides better quality (smaller MSE) than path tracing with equal time, thanks to our path guiding.

We also compare the denoised results (using Intel Open Image Denoise [9]) with our method and equal time path tracing as the inputs. By both visual and quantitative comparison, our method provides a better input for denoising. The cost of denoising is negligible.

Among all these scenes, we provide both high anisotropic media, such Wax in Figure 9 and Bumpysphere media in Figure 11 and also isotropic media, such as Marble in Figure 12 and Fog in Figure 14.

5.2 Performance and Timings

For all test scenes, we report the computation time of our method, Path tracing with equal time and the reference in Table 2. In our method, we also provide the cost of training step.

Comparing our method and classical path tracing with equal time, our method has less errors (MSE), which confirms the higher quality of our method. With equal time, our method has less sample count than path tracing, due to the extra cost for the resampled importance sampling in the path guiding.

Our method has an extra training cost to learn the incoming radiance distribution. The training costs depend on the scene complexity, media types and the learning iterations. For the equal time comparison, the training time is included in the total time.

5.3 Selective Sampling Method Validation

In Figure 15, we show the results of two sampling methods: multiple importance sampling (MIS) and importance resampling (RIS). For isotropic media ($g = 0.0$), MIS provides higher quality than RIS with equal rendering time. The cost of MIS is less expensive than RIS, so it can have more sample count with same time. For high anisotropic media ($g = 0.8$), RIS has higher quality

scene	Our Method						Path Tracing (Equal Time)			Reference	
	#iter.	time (train)	memory	spp (render)	time (render)	time (total)	MSE	#sample	time	MSE	time
		m	MB		m	m			m		h
BumpS.	4	5	0.17	3397	9.98	14.98	6.04e-3	7989	15.20	6.48e-2	6
Candle	6	10	0.67	344	10.08	20.08	2.73e-2	1395	20.05	9.29e-2	6
Oil	5	5	0.07	2531	9.96	14.96	5.41e-4	3848	14.93	6.44e-4	6
Athena	6	11	7.25	1876	18.02	29.02	2.13e-2	3629	29.36	4.80e-2	6
Stilllife	6	14	4.76	6998	59.80	73.80	3.29e-3	10586	75.00	5.93e-3	6
Bathroom	3	10	12.30	3954	20.22	30.22	3.73e-2	6081	30.37	7.74e-2	223.96

Tab. 2 Computation time, memory costs and error for our test scenes. #iter. represents the training iterations.

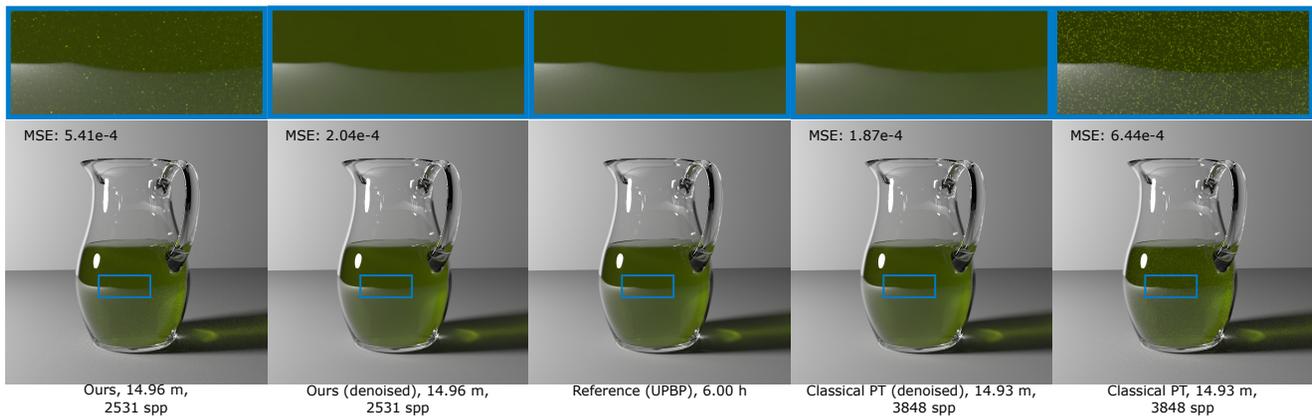


Fig. 8 Comparison between our method and path tracing with equal time on the Oil Scene.

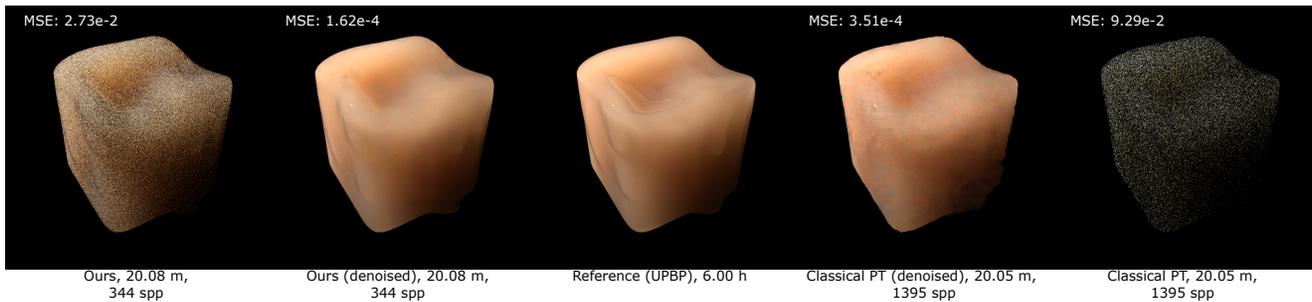


Fig. 9 Comparison between our method and path tracing with equal time on the Candle Scene.

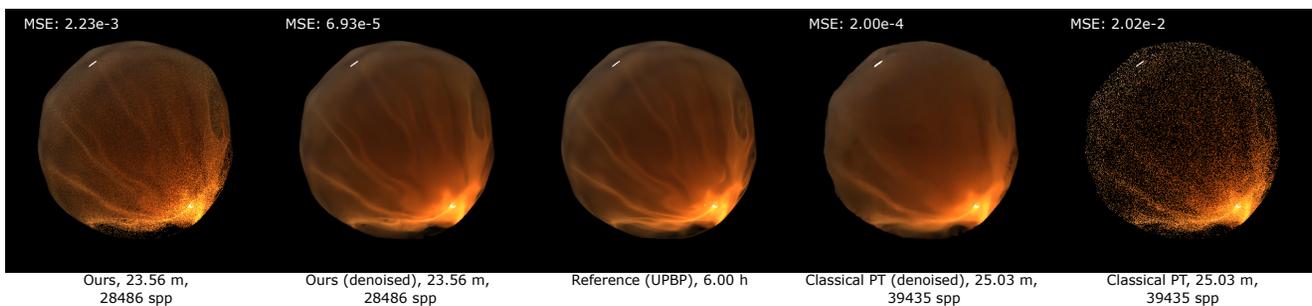


Fig. 10 Comparison between our method and path tracing with equal time on the Bumpy Sphere Scene (Single Scattering Only).

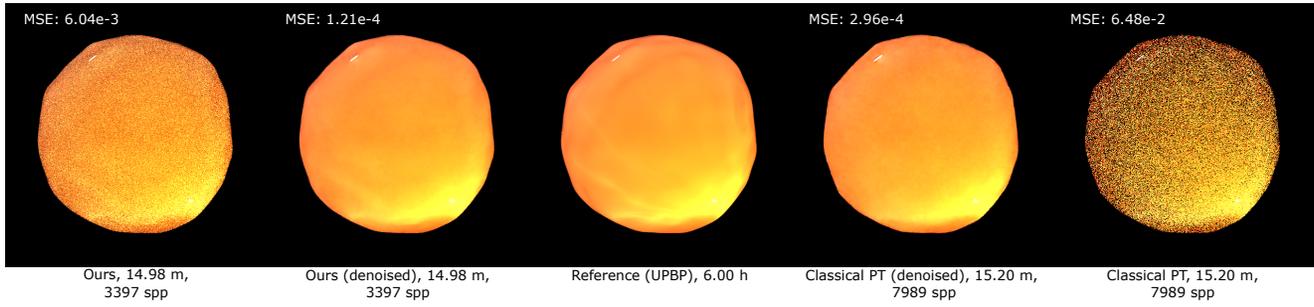


Fig. 11 Comparison between our method and path tracing with equal time on the Bumpy Sphere Scene (Full Solution).

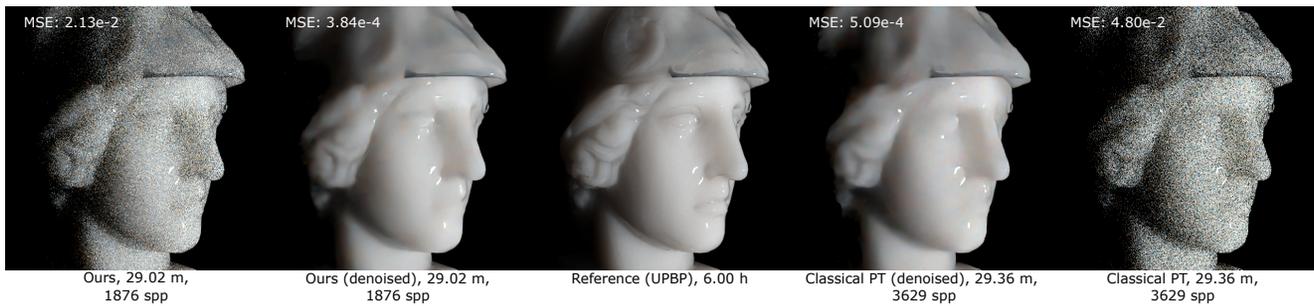


Fig. 12 Comparison between our method and path tracing with equal time on the Athena Scene.

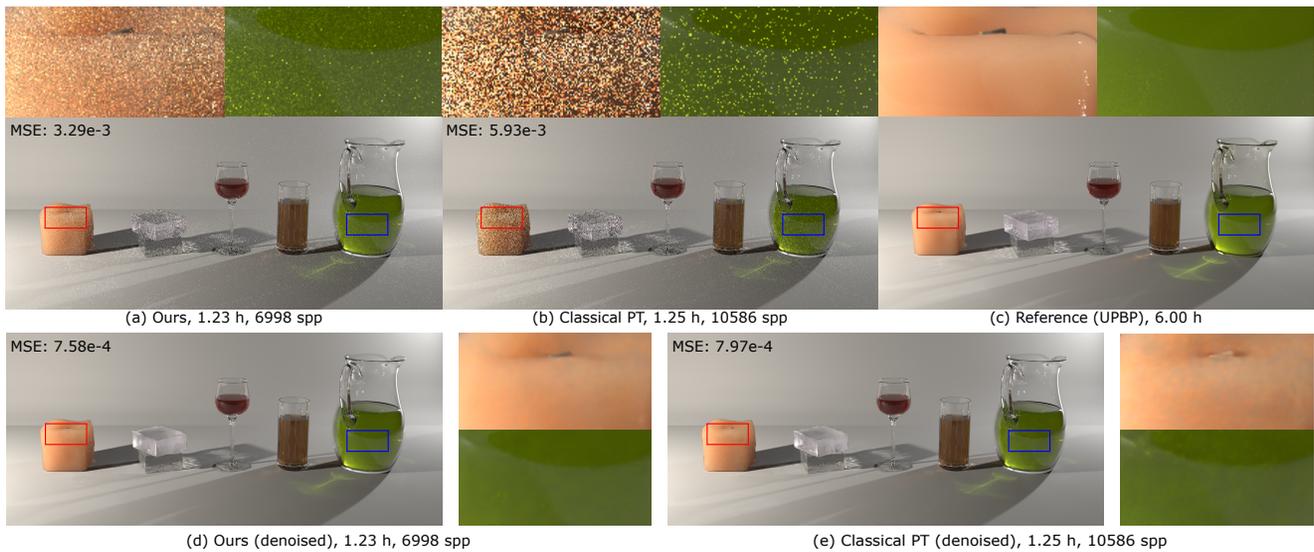


Fig. 13 Comparison between our method and path tracing with equal time on the Stilllife Scene.

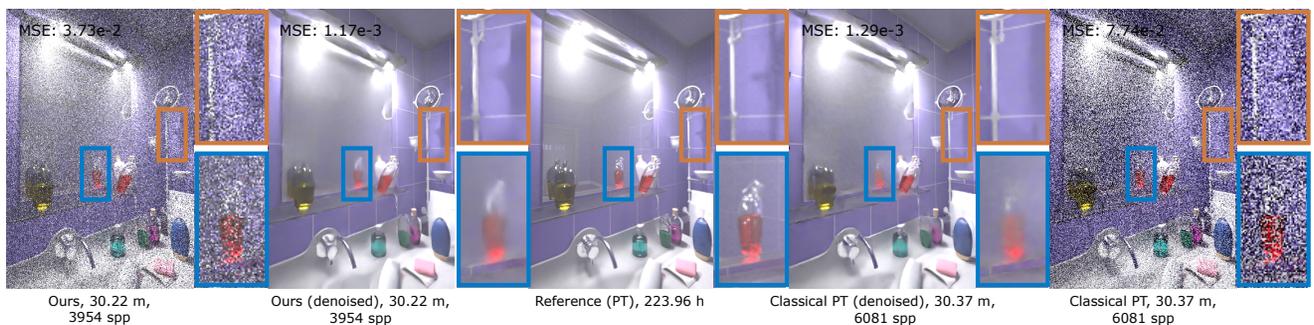


Fig. 14 Comparison between our method and path tracing with equal time on the Bathroom Scene.

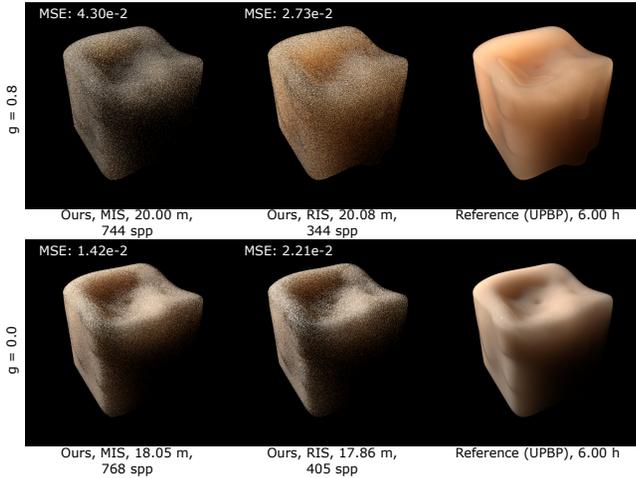


Fig. 15 Comparison between our method with multiple importance sampling (MIS) and our method with importance resampling (RIS) on the Candle Scene. Material: $g = 0.0$ and $g = 0.8$.

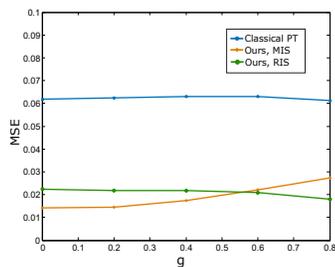


Fig. 16 Error (MSE) for our method with multiple importance sample, our method with resampled importance sampling and path tracing with equal time (about 20 minutes), as a function of media anisotropy, for the Candle Scene.

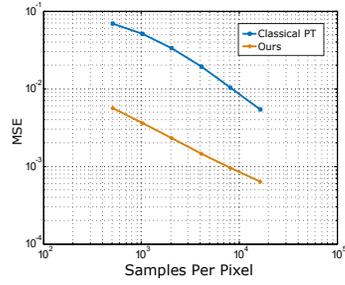


Fig. 17 Error (MSE) for our method and path tracing, as a function of the sample count per pixel, for the Candle Scene.

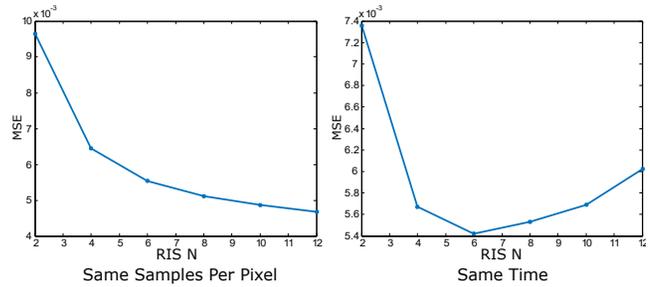


Fig. 18 Error (MSE) as a function of resampling sample count with equal samples per pixel (spp: 1029) (left) and equal time (30 minutes) (right) for the Candle Scene.

than MIS. Using high anisotropic media and a small area light source, both the incoming light distribution and the phase function have high frequency. Multiple importance sampling between these two functions might result in very low probability density function in one of the functions. This is shown in Figure 3. The same problem also exists in surface rendering, but it becomes even more obvious in participating media due to the high dimension of paths. On the other hand, resampling tries to sample the product of the two functions, though only uses 6 candidates, and still provides higher sampling quality.

Figure 16 displays the impact of the media anisotropy on path tracing and our two different sampling methods: MIS and RIS. With low anisotropy (from 0.0 to about 0.5), MIS provides better quality than RIS. After increasing the anisotropy further, RIS provides better quality than MIS, thanks to the higher sampling quality. Thus, in our implementation, we use 0.5 as a threshold to choose the sampling methods.

5.4 Parameter Analysis

Figure 17 shows convergence (error) of our method and path tracing with varying sample count. Increasing the sampling count, both methods have decreasing error. Our method has less error than path tracing consistently.

Figure 18 and 19 show the impact of several parameters on our algorithm for the Candle Scene (Figure 9):

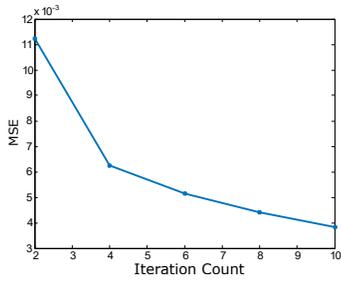


Fig. 19 Error (MSE) as a function of training iterations with different training time and same rendering time (30 minutes) for the Candle Scene.

- Figure 18 displays error (MSE) as a function of resampling sample count with same spp and same rendering time respectively. With the same sample per pixel, the difference between our result and the reference decreases, as the resampling sample count increases. With same rendering time, the difference between our result and the reference decreases at first and then increases after 6. More sample candidates for resampling produces higher quality sample, but the cost becomes very expensive with a lot of candidates. With equal time, less sample per pixel can be performed, and that’s the reason why the rendering quality decrease when more resampling candidates are added after 6 candidates. In practice, we use 6 resampling candidates for all the test scenes.
- Figure 19 displays the difference (MSE) between our method and reference as a function of the training iterations. Increasing the training iterations decreases the errors. The decreasing slope becomes slower after 4 iterations. With more iterations, we can get higher quality, but the training time increases as a factor of 2. So we use 3-6 iterations for all of our test scenes.

Figure 20 and 21 show the impact of surface and media parameters on our algorithm for the Candle Scene (Figure 9):

- Figure 20 shows the impact of surface roughness on our algorithm. We compare our method and path tracing with varying surface roughness (0.01, 0.1 and 0.5). With low roughness (0.01 or 0.1), our method has higher quality than path tracing with equal time. With high roughness (0.5), path tracing has higher quality than ours. When the surface is rough, the rays for the camera have higher chance to reach the light source, thus our method loses the benefit of guiding the camera rays to the light sources. With the extra cost for guiding, our method results in slower convergence for roughness surfaces.
- Figure 21 displays the impact of media anisotropy and mean free path on our algorithm. From isotropic media

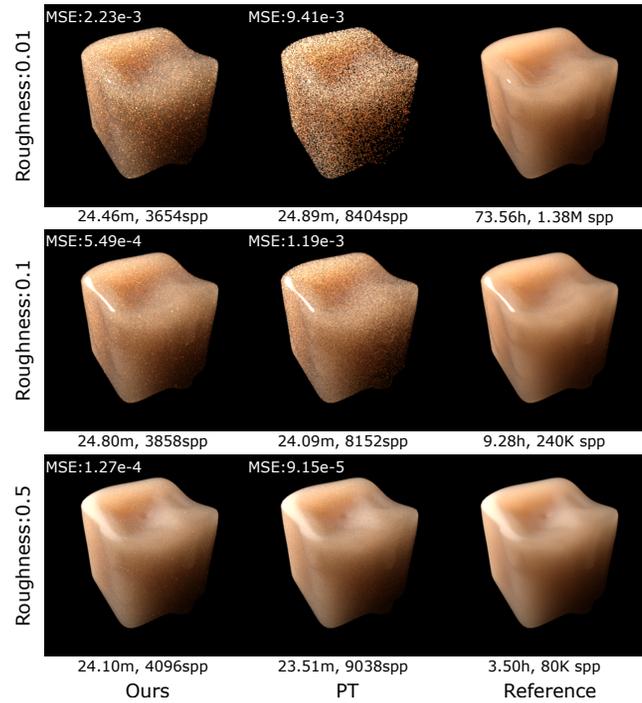


Fig. 20 Comparison between our method and path tracing with equal time on the Candle Scene with varying surface roughness.

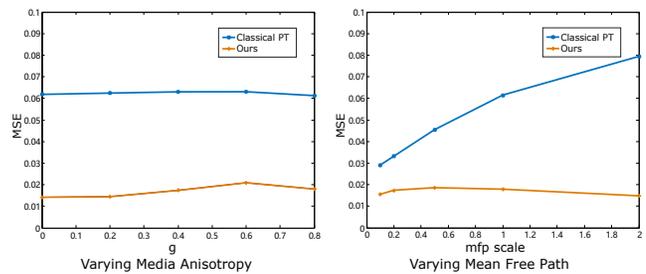


Fig. 21 Error (MSE) for our method and path tracing with equal time (about 20 minutes), as a function of media anisotropy and the media mean free path, for the Candle Scene.

to high anisotropic media or from small mean free path to large mean free path, our method produces better quality results than the equal-time path tracing consistently. The rendered images can be found in the supplemental material.

5.5 Discussion and limitation

In Figure 2, we compare our method with Müller et al. [23] on the Candle Scene. Müller et al. produces darker result, as their method has difficulties to find paths to light from the camera side due to the refractive boundaries and the multiple importance sampling used in their method leads the path to exit the participating media too early. Thus, we believe Müller et al. [23] can not be used for participating media.

Our method shows advantage for media with refractive boundaries. As shown in Figure 20, when the surface roughness increases, our method loses its benefits due to the expensive sampling method. An automatic switching between the proposed sampling methods and the simple phase sampling will be helpful.

We use path tracing as our comparison method, as our method is based on path tracing. We would like to keep the comparison fair.

Our method does not rely on denoising method. We provided the denoised results to show that our method can produce better inputs than classical path tracing.

We only consider the out scattering direction sampling in our paper, but distance sampling is also very important for media simulation. Including distance sampling will further improve the convergence speed.

6 Conclusion

We have presented a practical path guiding method for participating media. We use a SD-tree to represent the incoming radiance distribution in the volume. We propose a novel product importance sampling method, combining the incoming radiance distribution and the phase function, with only a couple of samples. The proposed method provides faster convergence compared to classical path tracing.

We also did a full analysis of the light and phase function distributions, and provided a selective importance sampling strategy, to avoid expensive product importance sampling where it does not required. In the end, we got even faster convergence.

Our algorithm is suitable for any media, from low frequency effect where multiple scattering dominant to high frequency effects where single scattering dominant.

In the future, we will consider introduce our method to heterogeneous participation media and we also want to combine the volume path guiding representation and

the surface path guiding representation. As for non-exponential correlated media [1], changing exponential distance sampling to non-exponential one will stimulate deeper thoughts in volumetric path guiding domain.

Acknowledgements

We thank the reviewer for the valuable comments. This work has been partially supported by the National Key R&D Program of China under grant No. 2017YFB0203000, the National Natural Science Foundation of China under grant No. 61802187 and 61872223, the Natural Science Foundation of Jiangsu under grant No. BK20170857, the fundamental research funds for the central universities No. 30918011320 and ANR project ANR-15-CE38-0005 "Materials".

Open Access This article is distributed under the terms of the Creative Commons Attribution License which permits any use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.

References

- [1] J. Adrian, A. Carlos, and G. Diego. A radiative transfer framework for spatially-correlated materials. *ACM Transactions on Graphics*, 37(4):1–13.
- [2] B. Bitterli and W. Jarosz. Beyond points and beams: Higher-dimensional photon samples for volumetric light transport. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)*, 36(4), July 2017.
- [3] S. Chandrasekhar. *Radiative transfer*. Dover publications, New York, 1960.
- [4] I. Georgiev, J. Krivánek, T. Hachisuka, D. Nowrouzezahrai, and W. Jarosz. Joint importance sampling of low-order volumetric scattering. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2013)*, 32(6), nov 2013.
- [5] J. Guo, P. Bauszat, J. Bikker, and E. Eisemann. Primary sample space path guiding. In W. Jakob and T. Hachisuka, editors, *Eurographics Symposium on Rendering - EI & I*, pages 73–82, July 2018.
- [6] S. Herholz, O. Elek, J. Vorba, H. Lensch, and J. Krivanek. Product importance sampling for light transport path guiding. In *Eurographics Symposium on Rendering*, 2016.
- [7] S. Herholz, Y. Zhao, O. Elek, D. Nowrouzezahrai, H. P. A. Lensch, and J. Krivanek. Volume Path Guiding Based on Zero-Variance Random Walk Theory. *ACM Transactions on Graphics*, Mar. 2019.
- [8] N. Holzschuch. Accurate computation of single scattering in participating media with refractive boundaries. *Comput. Graph. Forum*, 34(6):48–59, 2015.
- [9] Intel. Intel Open Image Denoise. <https://openimagedenoise.github.io/index.html>.
- [10] W. Jakob. Mitsuba renderer. <http://www.mitsuba-renderer.org/>, 2010.

- [11] W. Jarosz, C. Donner, M. Zwicker, and H. W. Jensen. Radiance caching for participating media. *Acm Transactions on Graphics*, 27(1):1–11, 2008.
- [12] W. Jarosz, D. Nowrouzezahrai, I. Sadeghi, and H. W. Jensen. A comprehensive theory of volumetric radiance estimation using photon points and beams. *ACM Trans. Graph.*, 30(1):5:1–5:19, Jan. 2011.
- [13] W. Jarosz, D. Nowrouzezahrai, R. Thomas, P.-P. Sloan, and M. Zwicker. Progressive photon beams. *ACM Trans. Graph. (proc. SIGGRAPH Asia)*, 30(6), 2011.
- [14] W. Jarosz, M. Zwicker, and H. W. Jensen. The beam radiance estimate for volumetric photon mapping. *Comput. Graph. Forum (proc. Eurographics)*, 27(2):557–566, Apr. 2008.
- [15] H. W. Jensen. Importance driven path tracing using the photon map. In *Rendering Techniques '95*, pages 326–335, 1995.
- [16] H. W. Jensen and P. H. Christensen. Efficient simulation of light transport in scenes with participating media using photon maps. In *SIGGRAPH*, pages 311–320, 1998.
- [17] M. Julio, J. Adrian, J. Wojciech, and G. Diego. Second-order occlusion-aware volumetric radiance caching. *ACM Transactions on Graphics*, 2018.
- [18] J. T. Kajiya. The rendering equation. *SIGGRAPH Comput. Graph.*, 20(4):143–150, 1986.
- [19] J. Křivánek, I. Georgiev, T. Hachisuka, P. Vévoda, M. Šik, D. Nowrouzezahrai, and W. Jarosz. Unifying points, beams, and paths in volumetric light transport simulation. *ACM Trans. Graph. (proc. SIGGRAPH)*, 33(4):1–13, Aug. 2014.
- [20] J. Křivánek and E. d'Eon. A zero-variance-based sampling scheme for monte carlo subsurface scattering. In *ACM SIGGRAPH 2014 Talks*, SIGGRAPH '14, pages 66:1–66:1, 2014.
- [21] E. P. Lafortune and Y. D. Willems. Rendering participating media with bidirectional path tracing. In *Proceedings of the Eurographics Workshop on Rendering Techniques '96*, pages 91–100, 1996.
- [22] J. Meng, J. Hanika, and C. Dachsbacher. Improving the dwivedi sampling scheme. *Computer Graphics Forum*, 35(4):37–44, 2016.
- [23] T. Müller, M. Gross, and J. Novák. Practical path guiding for efficient light-transport simulation. *Computer Graphics Forum (Proceedings of EGSR)*, 36(4):91–100, June 2017.
- [24] T. Müller, B. McWilliams, F. Rousselle, M. Gross, and J. Novák. Neural importance sampling. *arXiv preprint arXiv:1808.03856*, 2018.
- [25] S. G. Narasimhan, M. Gupta, C. Donner, R. Ramamoorthi, S. K. Nayar, and H. W. Jensen. Acquiring scattering properties of participating media by dilution. *ACM Trans. Graph. (proc. Siggraph)*, 25(3):1003–1012, July 2006.
- [26] S. N. Pattanaik and S. P. Mudur. Computation of global illumination in a participating medium by monte carlo simulation. *The Journal of Visualization and Computer Animation*, 4(3):133–152, 1993.
- [27] M. Pauly, T. Kollig, and A. Keller. Metropolis light transport for participating media. In *Proceedings of the Eurographics Workshop on Rendering Techniques 2000*, pages 11–22, 2000.
- [28] F. Simon, A. Jung, J. Hanika, and C. Dachsbacher. Selective guided sampling with complete light transport paths. *Transactions on Graphics (Proceedings of SIGGRAPH Asia)*, 37(6), 2018.
- [29] J. F. Talbot, D. Cline, and P. Egbert. Importance resampling for global illumination. In *Proceedings of the Sixteenth Eurographics Conference on Rendering Techniques*, EGSR '05, pages 139–146, 2005.
- [30] E. Veach and L. J. Guibas. Metropolis light transport. In *Computer Graphics (ACM SIGGRAPH '97 Proceedings)*, volume 31, pages 65–76, 1997.
- [31] J. Vorba, O. Karlík, M. Šik, T. Ritschel, and J. Křivánek. On-line learning of parametric mixture models for light transport simulation. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2014)*, 33(4), aug 2014.
- [32] Q. Zheng and M. Zwicker. Learning to importance sample in primary sample space. *Computer Graphics Forum (Proc. Eurographics)*, 38(2), 2019.



Hong Deng is currently finishing as a Master of CS (Second year) at Nanjing University of Science and Technology. His main area of interest are in physically-based rendering, game development and real-time graphics.



Beibei Wang is an Associate Professor at Nanjing University of Science and Technology. She received her PhD from Shandong University in 2014 and visited Telecom ParisTech from 2012 to 2014. She worked as a Postdoc in Inria from 2015 to 2017. She joined NJUST in March 2017. Her research interests include rendering.



Rui Wang received the bachelor degree in computer science and the PhD degree in mathematics from Zhejiang University, China. Now, He is a professor with the State Key Laboratory of CAD&CG and the College of Computer Science and Technology, Zhejiang University. He is working in the area of computer graphics and has interests including real-time rendering, realistic rendering, GPU-based computation, and 3D display techniques.



Nicolas Holzschuch is a Senior Researcher at INRIA Grenoble Rhône-Alpes, and the scientific leader of the MAVERICK research team. He received his PhD from Grenoble University in 1996 and his Habilitation in 2007. He joined INRIA in 1997. His research interests include photorealistic rendering and real-time rendering, with an emphasis on material models and participating media.