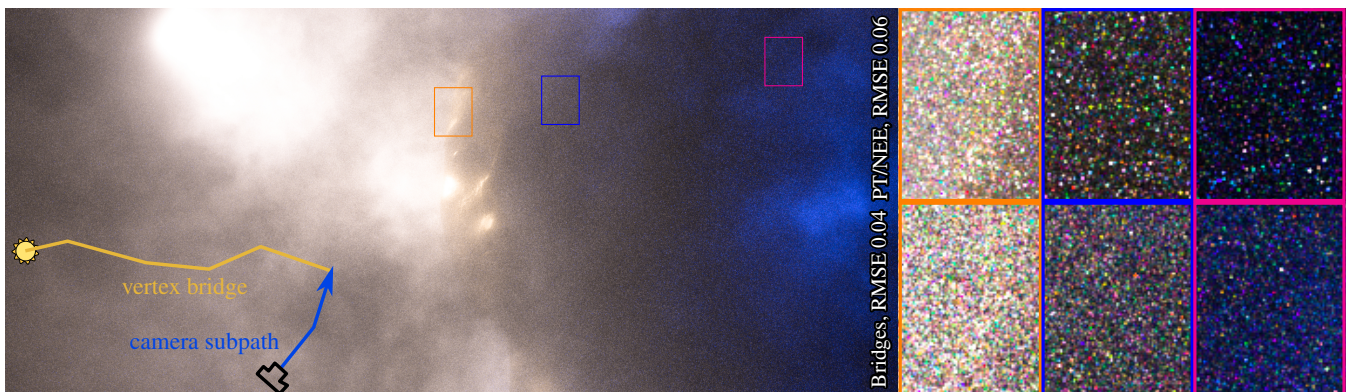


# Bridge Sampling for Connections via Multiple Scattering Events

Vincent Schüßler , Johannes Hanika  and Carsten Dachsbacher 

Karlsruhe Institute of Technology, Germany



**Figure 1:** A golden statue submerged in a heterogeneous cloud volume (scattering albedo 100%, 404M 4D voxels with motion blur; the phase function approximates Mie scattering by an equal mixture of two Henyey-Greenstein lobes of mean cosines  $g = 0.98$  and  $g = -0.4$ ). The image is rendered up to a maximum path length of 10 vertices. We importance sample the multiple scattering contribution by bridges which take into account the phase functions and the geometry terms of the measurement contribution. In an approximate equal time comparison, this leads to much improved rendition of the multiple scattering (blue/magenta insets where the statue shadows the indirect illumination in the volume). PT/NEE resolves the direct illumination better (orange inset near the white light), though neglecting the multiple scattering.

## Abstract

Explicit sampling of and connecting to light sources is often essential for reducing variance in Monte Carlo rendering. In dense, forward-scattering participating media, its benefit declines, as significant transport happens over longer multiple-scattering paths around the straight connection to the light. Sampling these paths is challenging, as their contribution is shaped by the product of reciprocal squared distance terms and the phase functions. Previous work demonstrates that sampling several of these terms jointly is crucial. However, these methods are tied to low-order scattering or struggle with highly-peaked phase functions.

We present a method for sampling a bridge: a subpath of arbitrary vertex count connecting two vertices. Its probability density is proportional to all phase functions at inner vertices and reciprocal squared distance terms. To achieve this, we importance sample the phase functions first, and subsequently all distances at once. For the latter, we sample an independent, preliminary distance for each edge of the bridge, and afterwards scale the bridge such that it matches the connection distance. The scale factor can be marginalized out analytically to obtain the probability density of the bridge. This approach leads to a simple algorithm and can construct bridges of any vertex count. For the case of one or two inserted vertices, we also show an alternative without scaling or marginalization.

For practical path sampling, we present a method to sample the number of bridge vertices whose distribution depends on the connection distance, the phase function, and the collision coefficient. While our importance sampling treats media as homogeneous we demonstrate its effectiveness on heterogeneous media.

## CCS Concepts

• **Computing methodologies** → **Rendering**; Ray tracing;

## 1. Introduction

The fundamental path sampling technique in Monte Carlo rendering is to simulate the random walk of a photon. This *analog path tracing* is usually enhanced by explicit *connection* of subpaths to form more capable techniques, e.g. path tracing with next-event estimation (NEE) or full bidirectional path tracing. These are significant especially for importance sampling the light emission, which analog path tracing from the sensor is incapable of. It has to rely on randomly reaching a light source, which often severely limits its sampling quality, even though it handles all remaining terms well.

In participating media, sampling straight line connections between two vertices is less effective. Common media, e.g. fog or water, exhibit high albedo and highly-peaked forward scattering. Thus a major share of the light transport happens via sequences of multiple scattering events. These form paths around the straight connection, as shown in the example in Fig. 2. It also demonstrates how path tracing with next-event estimation struggles to sample these paths. Like all methods based on analog path extensions, it tends to get lost in less important regions of the medium.

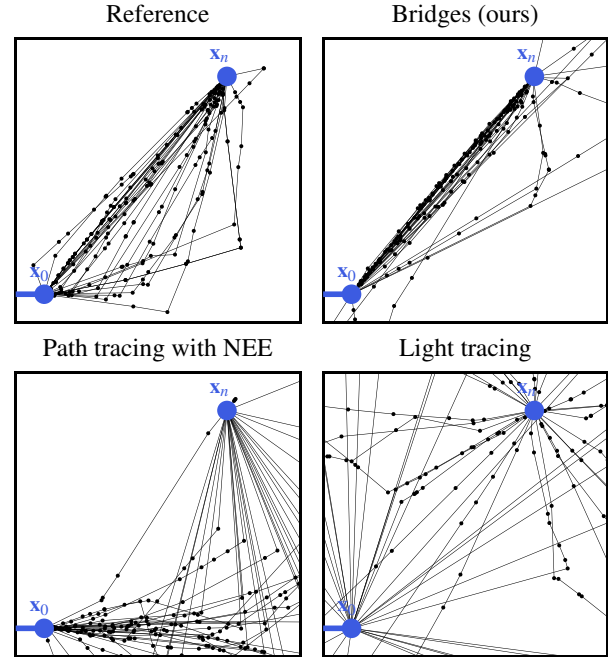
Our goal is to explicitly sample this type of connecting subpath between two vertices, which we refer to as a *bridge*. This is challenging, since we need to handle two aspects of the geometric configuration along the bridge: the forward-scattering peak in the phase functions leads to near-specular transport, and the geometric singularity in the distance between vertices can cause infinite variance. Therefore we need to importance sample the product of these terms for robust sampling.

Fortunately, we can turn to our prior observation about analog path tracing: it is mostly very good at sampling paths, with the critical flaw of terminating at a random point instead of our target. Our main idea is to fix this *after* sampling a preliminary path: we rotate and uniformly scale it, such that its endpoint will match the target. By preserving angles, this transformation maintains the important geometric properties of the initial sampling. The probability density (PDF) of this sampling technique involves marginalization, i.e. for a given bridge we need to integrate over all ways in which we could have sampled it. This turns out to be surprisingly simple for our purposes and we derive an analytical result.

Sampling this way, we forego importance sampling of the directions at endpoints, though other vertex connection techniques share this property. This is usually handled well by combination through multiple importance sampling (MIS). We also cannot control the total track length and thus transmittance, but typically this does not cause a lot of variance in the case of forward scattering. All other terms of the bridge are importance sampled by our algorithm.

In summary, we make the following contributions:

- a general formulation of the bridge sampling algorithm and its PDF derivation (Section 3.1),
- our recommended instance of this algorithm, which samples distances on bridges of arbitrary vertex count based on scaling the bridge (Section 3.2),
- a second instance without scaling or marginalization, which works with one or two inserted vertices (Section 3.3),
- an algorithm for sampling the number of vertices on a bridge, based on connection distance and medium properties (Section 4).



**Figure 2:** Path distributions produced by different sampling algorithms for connecting a subpath ending at  $\mathbf{x}_0$  to a light source at  $\mathbf{x}_n$  over 8 additional vertices in an infinite medium with Henyey-Greenstein phase function and mean cosine  $g = 0.95$ . The random walk of both path and light tracing is uninformed about the goal vertex and struggles to focus on the important regions of path space. Bridge sampling is conditional on both endpoints and successfully samples paths near the straight line connection. The reference is the distribution of paths that would be optimal for importance sampling, i.e. proportional to their measurement contribution. It also shows paths that first follow the forward-scattering phase function to the right, before turning towards the light source, which in a complete renderer would be handled by MIS combination of our sampling and phase function sampling. We compute the reference by resampling 30 paths from 6000000 candidates generated by path tracing with NEE.

## 2. Background and related work

Realistic rendering is commonly formalized by expressing the measurement  $I_j$  of pixel  $j$  as the integral of the *measurement contribution function*  $f_j$  over all paths  $\bar{\mathbf{x}} \in \mathcal{P}$  between sensors and light sources:

$$I_j = \int_{\mathcal{P}} f_j(\bar{\mathbf{x}}) d\bar{\mathbf{x}}. \quad (1)$$

The measurement contribution of a path  $\bar{\mathbf{x}} = (\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_k)$  is

$$f_j(\bar{\mathbf{x}}) = W_j(\mathbf{x}_0, \mathbf{x}_1) L_e(\mathbf{x}_k, \mathbf{x}_{k-1}) G(\mathbf{x}_0, \mathbf{x}_1) T(\mathbf{x}_0, \mathbf{x}_1) \cdot \prod_{i=1}^{k-1} f_s(\mathbf{x}_{i-1}, \mathbf{x}_i, \mathbf{x}_{i+1}) G(\mathbf{x}_i, \mathbf{x}_{i+1}) T(\mathbf{x}_i, \mathbf{x}_{i+1}), \quad (2)$$

with pixel response  $W_j$ , emitted radiance  $L_e$ , BSDF or phase function  $f_s$ , transmittance  $T(a, b) = \exp\left(-\int_0^a \mu_t(a-t) \omega_{a \rightarrow b} dt\right)$

with extinction coefficient  $\mu_t$  and geometry term  $G(x,y) = C(x,y)C(y,x)\|x-y\|_2^{-2}$ , where  $C(x,y)$  is the cosine term if  $x$  is on a surface. Note that  $f_s$  and  $L_e$  include the scattering and absorption coefficients  $\mu_s$  or  $\mu_a$  respectively if they describe a volumetric interaction. For a more thorough definition and explanation of all terms and for an overview of volumetric rendering, we refer to the survey by Novak et al. [NGHJ18].

**Extended next-event estimation in volumes.** Rendering participating media is hard, especially in the presence of high frequency content in the phase functions (most commonly: very peaked forward scattering, but Mie theory also predicts backward peaks) since this poses angular constraints on paths with high throughput. Even without this, the geometry terms  $G$  contained in the measurement contribution are a challenge. The inverse square distance term is unbounded if path vertices come closer together. If light sources are contained in a volume this term can cause infinite variance in the estimator. To counter this, equiangular sampling includes the offending  $G$  term in the importance sampling [KF12]. This method works best for isotropic phase functions, since these are omitted from the PDF. To address this, Georgiev et al. [GKH\*13] propose joint importance sampling of distances and phase functions, but resort to tabulation and thus still result in suboptimal sampling for generic and spiky phase functions. *Once More Scattered NEE* (OMNEE) solves the special case of a single additional scattering vertex between a vertex and the light source for homogeneous media and arbitrary forward scattering phase functions [HWD22]. Our method works for full-sphere phase functions and can insert an arbitrary number of extra vertices, and we show an equivalent formulation for the special case of a single vertex. *Multiple vertex next event estimation* (MVNEE) [WHD17] adds multiple vertices on the way to the light, too. As pointed out by the authors and confirmed by our results, the resulting PDF is not ideal, especially for longer paths.

**Glossy light transport.** Solving for a path with multiple vertices which fulfills angular constraints dictated by the phase functions is similar in spirit to highly glossy/specular light transport on surfaces. This has been explored for Markov chain rendering [Jak13] as well as for extending Monte Carlo next event estimation [HDF15]. Our formulation is much simpler and works without an iterative predictor/corrector scheme since path vertices in the medium are less constrained as compared to surfaces.

**Product Sampling.** Sampling the product of material properties and incident light [HEV\*16] has been explored for volumetric scattering [HZE\*19], by casting both incident light and phase function to a Gaussian mixture model and subsequently computing the product of all mixed-term Gaussians in closed form. As shown previously, the product of incident light and a peaky phase function can become bi-modal [HWD22]. This means a product importance sampling scheme can not simply rely on the product of two Gaussian approximations, which would remain unimodal. Thus, such an approach will require many lobes in a Gaussian mixture to obtain good accuracy.

**Caching light transport.** Employing memory to store light paths to compute volumetric light transport has been explored in great detail [KGH\*14]. This incurs an implementation, memory, and

**Table 1:** Summary of our notation.

Term	Description
$\mathbf{x} \in X, \mathbf{x}' \in X'$	(Preliminary) path vertex position
$\omega \in \Omega, \omega' \in \Omega'$	(Preliminary) path edge direction
$d \in D, d' \in D'$	(Preliminary) path vertex distance
$\omega_{a \rightarrow b}, \omega'_{a \rightarrow b}$	direction from $\mathbf{x}_a$ to $\mathbf{x}_b$ ( $\mathbf{x}'_a$ to $\mathbf{x}'_b$ )
$\omega_i, \omega'_i$	direction from $\mathbf{x}_{i-1}$ to $\mathbf{x}_i$ ( $\mathbf{x}'_{i-1}$ to $\mathbf{x}'_i$ )
$d_i, d'_i$	distance between $\mathbf{x}_{i-1}$ and $\mathbf{x}_i$ ( $\mathbf{x}'_{i-1}$ and $\mathbf{x}'_i$ )
$\bar{\mathbf{x}}_{i,j} \in \bar{X}$	the sequence $(\mathbf{x}_i, \mathbf{x}_{i+1}, \dots, \mathbf{x}_j)$
$\bar{\omega}_{i,j} \in \bar{\Omega}$	the sequence $(\omega_i, \omega_{i+1}, \dots, \omega_j)$
$\bar{d}_{i,j} \in \bar{D}$	the sequence $(d_i, d_{i+1}, \dots, d_j)$
$\bar{\mathbf{x}}, \bar{\omega}, \bar{d}$	shorthand for $\bar{\mathbf{x}}_{0,n}, \bar{\omega}_{0,n}, \bar{d}_{0,n}$
$\bar{\mathbf{x}}', \bar{\omega}', \bar{d}'$	analogous to above

runtime overhead to evaluate the PDF of all active estimators to compute the multiple importance sampling weights. Path guiding [VHH\*19] employs memory to explicitly store sampling densities to guide random walks towards high contribution paths. This approach also handles visibility, while our importance sampling scheme does not. A specialized class of algorithms is concerned with subsurface scattering, and guides transport paths back to the surface [Kd14].

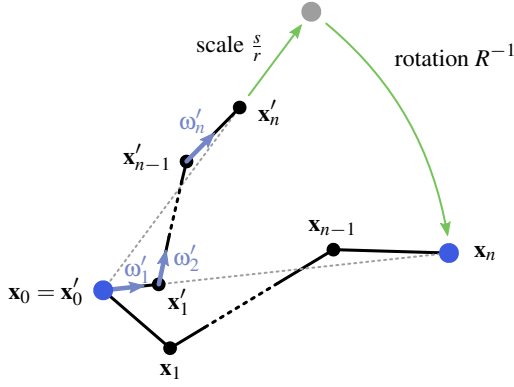
**Approximations for high-order scattering.** Analytic expressions for multiple scattering are hard to obtain and represent a well-researched area [dEo16]. These formulations are especially useful for extremely long paths and when the visibility between transport vertices is known to be unobstructed or can be ignored. With this paper, we address a scattering regime up to the lower two-digit vertex counts.

**Time-of-flight imaging.** Connections via additional scattering vertices in media [JMM\*14] or on surfaces [PVG19] can also be used to control the total track lengths of light paths. This is important when considering the time-of-flight of light, while we focus on the more common steady-state simulation. Our sampling aims to fulfill angular constraints on multiple additional scattering vertices in media, while track length is only of secondary concern.

### 3. Sampling vertex bridges

Our goal is to connect two vertices  $\mathbf{x}_0$  and  $\mathbf{x}_n$  via a chain of  $n$  edges, while importance sampling all inner phase functions and reciprocal squared distance terms. To accomplish this, we first sample a preliminary subpath  $\bar{\mathbf{x}}'_{0,n}$  starting from  $\mathbf{x}'_0 = \mathbf{x}_0$ , which we call a *bridge*. Next, we rotate and scale the bridge around  $\mathbf{x}_0$ , such that its endpoint  $\mathbf{x}'_n$  matches the target  $\mathbf{x}_n$ , as shown in Fig. 3. This affine transformation preserves angles between edges and therefore the phase function PDF under mild assumptions (see Section 3.1). Therefore, we can easily make use of stock phase function sampling while constructing the bridge. Since we stay in solid angle domain, all reciprocal squared distance terms are sampled naturally as well. Please see Table 1 for a description of our notation.

The remaining challenge is to sample suitable vertex distances and compute the PDF of the resulting vertices. In this section, we



**Figure 3:** Our goal is to connect the endpoints  $\mathbf{x}_0$  and  $\mathbf{x}_n$  via a subpath with  $n-1$  additional vertices and importance-sampled directions  $\bar{\omega}_{2,n} = (\omega'_2, \dots, \omega'_n)$ . We sample a preliminary path  $\bar{\mathbf{x}}'_{0,n}$  beginning from  $\mathbf{x}'_0 = \mathbf{x}_0$  and  $\omega'_1 = \omega'_{0 \rightarrow n}$ , and then proceed via free-flight and phase function sampling. To connect  $\mathbf{x}_0$  and  $\mathbf{x}_n$ , we rotate and scale this path around  $\mathbf{x}_0$  such that  $\mathbf{x}'_n$  ends up at  $\mathbf{x}_n$ . Notably, this affine transformation preserves angles and thus phase function importance sampling weights.

present two alternatives for distance sampling. The first one samples preliminary independent distances using the exponential distribution and realizes the correct bridge length through scaling (Section 3.2). It is simple to implement for any number of vertices, but incurs a non-constant Jacobian for its projection step. The second alternative can get rid of this factor by directly sampling solutions to the constrained problem (Section 3.3), i.e. it always samples a unit scale factor, but is limited to inserting a maximum of two vertices. Beforehand, we will derive the PDF of our sampling scheme, independently from any distance sampling.

### 3.1. PDF derivation

To implement our sampling scheme into the Monte Carlo rendering framework, we need to compute the PDF of sampling a bridge. This is not a simple change of variables, since the dimensionalities do not match: We sample  $n$  vertices, but we need a PDF for only the inner  $n-1$  vertices  $\bar{\mathbf{x}}_{1,n-1}$ , since  $\mathbf{x}_n$  is predetermined.

We approach this in two steps. Firstly, we match the dimensionality by lifting  $\bar{\mathbf{x}}_{1,n-1}$  to an auxiliary space. To do so, we pad  $\bar{\mathbf{x}}_{1,n-1}$  with the auxiliary vertex  $\mathbf{x}'_n$ . This allows us to define a bijective map to the preliminary vertices  $\bar{\mathbf{x}}'_{1,n}$ , since their scale is determined by  $\mathbf{x}'_n$ . In the second step, we marginalize out  $\mathbf{x}'_n$  and obtain the PDF on  $\bar{\mathbf{x}}_{1,n-1}$ .

**Mapping to the auxiliary space.** We define the bijective map from auxiliary space  $\bar{X} \times X'_n$  to preliminary vertices  $\bar{X}'$  by

$$m(\bar{\mathbf{x}}_{1,n-1}, \mathbf{x}'_n) = \mathbf{x}_0 + \frac{r}{s}(R(\mathbf{x}_1 - \mathbf{x}_0), \dots, R(\mathbf{x}_n - \mathbf{x}_0)) = \bar{\mathbf{x}}'_{1,n}, \quad (3)$$

where  $R$  is the shortest rotation from  $\omega_{0 \rightarrow n}$  to  $\omega'_{0 \rightarrow n}$  and  $r$  and  $s$  are the preliminary and the target connection distance, respectively:

$$r := \|\mathbf{x}'_n - \mathbf{x}'_0\|_2, \quad s := \|\mathbf{x}_n - \mathbf{x}_0\|_2. \quad (4)$$

Since  $R$  is orthogonal and  $\mathbf{x}'_n$  maps to itself, the Jacobian of  $m$  is:

$$\det J_m(\bar{\mathbf{x}}_{1,n-1}, \mathbf{x}'_n) = \left| \begin{array}{c|c} \text{diag}\left(\frac{r}{s}, \dots, \frac{r}{s}\right) & * \\ \hline \mathbf{0} & \mathbf{I}_3 \end{array} \right| = \left(\frac{r}{s}\right)^{3(n-1)}. \quad (5)$$

With the product PDFs of phase function sampling  $p_{\bar{\Omega}'}(\bar{\omega}'_{2,n})$  and distance sampling  $p_{\bar{D}'}(\bar{d}'_{1,n})$ , we can now derive the PDF in the auxiliary space  $\bar{X} \times X'_n$ , using  $d'_i = \frac{r}{s}d_i$ :

$$\begin{aligned} p_{\bar{X} \times X'_n}(\bar{\mathbf{x}}_{1,n-1}, \mathbf{x}'_n) &= p_{\bar{X}'}(m(\bar{\mathbf{x}}_{1,n-1}, \mathbf{x}'_n)) |\det J_m(\bar{\mathbf{x}}_{1,n-1}, \mathbf{x}'_n)| \\ &= p_{\bar{X}'}(\bar{\mathbf{x}}'_{1,n}) \left(\frac{r}{s}\right)^{3(n-1)} \\ &= \left(\prod_{i=1}^n \frac{1}{\left(\frac{r}{s}d_i\right)^2}\right) p_{\bar{\Omega}'}(\bar{\omega}'_{2,n}) p_{\bar{D}'}\left(\frac{r}{s}\bar{d}'_{1,n}\right) \left(\frac{r}{s}\right)^{3(n-1)} \\ &= \left(\prod_{i=1}^n \frac{1}{d_i^2}\right) p_{\bar{\Omega}'}(\bar{\omega}'_{2,n}) p_{\bar{D}'}\left(\frac{r}{s}\bar{d}'_{1,n}\right) \left(\frac{r}{s}\right)^{n-3}. \end{aligned} \quad (6)$$

**Marginalization.** The remaining step is to marginalize out the auxiliary vertex  $\mathbf{x}'_n$ . We illustrate the configuration in Fig. 4. Since we chose  $\omega'_1 = \omega_{0 \rightarrow n}$  and  $\omega_1$  is fixed by  $\mathbf{x}_1$ ,  $R$  is already determined by  $\omega'_1 = R\omega$ . Note that this works only because of the symmetry of the construction: the shortest rotation between  $\omega'_1$  and  $\omega_1$  is the same as between  $\omega_{0 \rightarrow n}$  and  $\omega'_{0 \rightarrow n}$ . With  $R$  fixed,  $\mathbf{x}'_n$  is on the line  $\mathbf{x}'_n = \mathbf{x}_0 + rR\omega_{0 \rightarrow n}$  and we must only integrate over  $r$ . We multiply by  $r^2$  for the change to the spherical domain:

$$\begin{aligned} p_{\bar{X}}(\bar{\mathbf{x}}_{1,n-1}) &= \int_{\Omega'} \int_0^\infty p_{\bar{X} \times X'_n}(\bar{\mathbf{x}}_{1,n-1}, \mathbf{x}'_n) r^2 dr d\delta_{\omega_{0 \rightarrow n}}(\omega'_{0 \rightarrow n}) \\ &= \left(\prod_{i=1}^n \frac{1}{d_i^2}\right) p_{\bar{\Omega}}(\bar{\omega}'_{2,n}) \int_0^\infty p_{\bar{D}'}\left(\frac{r}{s}\bar{d}'_{1,n}\right) \frac{r^{n-1}}{s^{n-3}} dr. \end{aligned} \quad (7)$$

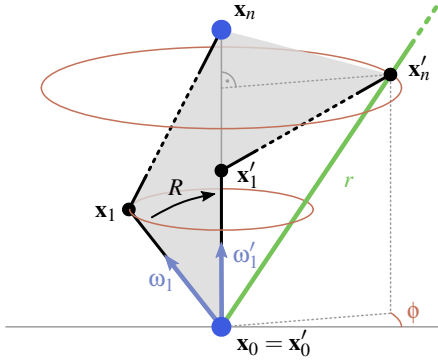
### 3.2. Distance sampling based on scaling

Next, we need to choose a specific distance sampling distribution. It should allow to compute the value of the marginal integral and result in a favorable PDF. Motivated by free-flight sampling, we pick the exponential distribution with scale parameter  $\lambda$

$$p_{\bar{D}'}(\bar{d}'_{1,n}) = \prod_{i=1}^n \lambda \exp(-\lambda d'_i) = \lambda^n \exp(-\lambda \|\bar{d}'_{1,n}\|_1), \quad (8)$$

and insert it into the marginal integral from Eq. (7):

$$\begin{aligned} &\int_0^\infty p_{\bar{D}'}\left(\frac{r}{s}\bar{d}'_{1,n}\right) \frac{r^{n-1}}{s^{n-3}} dr \\ &= \frac{\lambda^n}{s^{n-3}} \int_0^\infty \exp\left(-\frac{\lambda \|\bar{d}'_{1,n}\|_1}{s} r\right) r^{n-1} dr \\ &= \frac{\lambda^n}{s^{n-3}} (n-1)! \left(\frac{\lambda \|\bar{d}'_{1,n}\|_1}{s}\right)^{-n} = \frac{s^3 (n-1)!}{\|\bar{d}'_{1,n}\|_1^n}. \end{aligned} \quad (9)$$



**Figure 4:** For marginalization, we need to integrate over all  $\mathbf{x}'_n$  which we could have sampled to construct the bridge  $\bar{\mathbf{x}}_{0,n}$ . Since the rotation  $R$  is already determined by our fixed choice of  $\omega'_1$ ,  $\mathbf{x}'_n$  is on a line (green) and the integration is just over radius  $r$ . The depicted  $\mathbf{x}'_n$  is at  $r = s$ . Also note the rotational symmetry of the construction: for each value of  $\phi$ , the shortest rotation between  $\omega'_1$  and  $\omega_1$  is the same as between  $\omega_{0 \rightarrow n}$  and  $\omega'_{0 \rightarrow n}$ .

This leads to Algorithm 1 and the PDF in product vertex area measure:

$$p_{\bar{X}}(\bar{\mathbf{x}}_{1,n-1}) = \left( \prod_{i=1}^n \frac{1}{d_i^2} \right) p_{\bar{\Omega}}(\bar{\omega}'_{2,n}) \frac{s^3 (n-1)!}{\|\bar{d}_{1,n}\|_1^n}. \quad (10)$$

#### Algorithm 1 Sample bridge from $\mathbf{x}_0$ to $\mathbf{x}_n$

```

n ← sample vertex count with probability P_n ▷ see Section 4
x'_0 ← x_0
p ← P_n
omega'_1 ← omega_{0→n}
for i ← 1, ..., n do
  if i > 1 then
    omega'_i ← sample phase function given omega'_{i-1}
    p ← p p_Omega(omega'_i) ▷ phase function PDF
    d'_i ← -ln(1 - rand()) ▷ rand() ∈ [0, 1)
    x'_i ← x'_{i-1} + d'_i omega'_i
  compute rotation R such that x'_n - x_0 = (||x'_n - x_0||_2 / ||x_n - x_0||_2) R(x_n - x_0)
  for i ← 1, ..., n do
    x_i ← x_0 (||x_n - x_0||_2 / ||x'_n - x_0||_2) R^T(x'_i - x_0)
    omega ← R^T omega'_i
    d_i ← (||x_n - x_0||_2 / ||x'_n - x_0||_2) d'_i
  log p ← lgamma(n) - n ln ∏_{i=1}^n d_i ▷ log space to handle large n
  p ← p ||x_n - x_0||_2^3 exp(log p) ▷ product solid angle PDF

```

**Discussion.** The scale parameter  $\lambda$  does not appear in the result, which means that sampling of preliminary distances is scale-invariant in this case. This indicates that the result is not related to free flights (see Appendix A for an alternative view) and we cannot use it to sample transmittance. Still, as we will show, the resulting

PDF is suitable for our purposes. We canceled part of the  $s^{-(n-3)}$  term, with only a constant  $s^3$  remaining. The total travelled distance  $\|\bar{d}_{1,n}\|_1$  has low variance with strong forward scattering, and the factorial varies only with bridge length.

### 3.3. Constraint-based distance sampling for $n = 2$ and $n = 3$

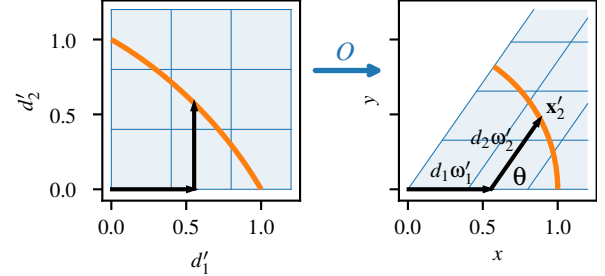
With scaling-based sampling we still incur a non-constant Jacobian, which is an additional source of variance. The Jacobian occurs, since we take infeasible paths, i.e. with unfit  $\|\mathbf{x}'_n - \mathbf{x}'_0\|_2$ , and project them to feasible paths that achieve the fixed connection distance  $\|\mathbf{x}_n - \mathbf{x}_0\|_2$ . In this section, our goal is to uniformly sample the space of feasible paths to get rid of the Jacobian term. We achieve this for the case of one or two inserted vertices, i.e.  $n = 2$  or  $n = 3$ , and show how generalization to higher dimensions remains difficult.

**Space of feasible paths.** To describe the space of feasible paths, we define the linear map  $O$  from preliminary vertex distances  $\bar{d}'$  to the vector  $\mathbf{x}'_n - \mathbf{x}'_0$ , i.e.  $\mathbf{x}'_n - \mathbf{x}'_0 = O\bar{d}'$ :

$$O = \begin{pmatrix} | & | & & | \\ \omega'_1 & \omega'_2 & \dots & \omega'_n \\ | & | & & | \end{pmatrix} \in \mathbb{R}^{3 \times n}. \quad (11)$$

This uses the edge directions  $\omega'_i$  that are already determined by phase function sampling.  $\bar{d}'$  produces a feasible path, if it satisfies the constraints on

1. connection distance  $\|O\bar{d}'\|_2 = \|\mathbf{x}_n - \mathbf{x}_0\|_2 = s$ ,
2. distance positivity  $\forall i \in \{1, \dots, n\}: d'_i > 0$ .



**Figure 5:** Constraint-based sampling for  $n = 2$  with  $\omega'_1, \omega'_2$  in the  $xy$ -plane and  $\omega'_1 \cdot \omega'_2 = \cos \theta$ .  $O$  maps a pair of distances  $d'_1, d'_2$  to  $x'_2$ . The feasible region (orange), i.e. where  $d'_1, d'_2 > 0$  and in this example  $\|O\bar{d}'\|_2 = 1$ , is an arc of length  $\theta$ . We can sample valid distances uniformly by first sampling a point on this arc uniformly and then solving for distances using  $O$ .

**Algorithm idea.** The first constraint tells us that  $O$  maps all feasible distances to a region on the sphere (or circle if  $n = 2$ ) with radius  $s = \|\mathbf{x}_n - \mathbf{x}_0\|_2$ , which we call the feasible region. The distance positivity constraint restricts the feasible region to an intersection of  $n$  halfspaces, which results in an arc for  $n = 2$  (see Fig. 5) and a spherical triangle for  $n = 3$ . If  $O$  is invertible, we can pick a random point  $\mathbf{y}$  in the feasible region defined by  $O^{-1}\mathbf{y} > 0$  and solve for a distance vector  $\bar{d}' = O^{-1}\mathbf{y}$ . Unfortunately, this is in general not the case, as  $\text{rank}(O) < 3$ , but we can generalize our idea using the singular value decomposition (SVD).

**Sampling algorithm.** We describe our algorithm for the general case, even though we will later show that it achieves the desired PDF only for  $n = 2$  and  $n = 3$ . First, we compute a SVD of  $O = U\Sigma V^T$ , i.e.  $U \in \mathbb{R}^{3 \times 3}$  and  $V \in \mathbb{R}^{n \times n}$  are orthogonal matrices,  $k = \text{rank}(O)$  and

$$\Sigma = \begin{pmatrix} \tilde{\Sigma} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \in \mathbb{R}^{3 \times n}, \quad \tilde{\Sigma} = \text{diag}(\sigma_1, \dots, \sigma_k). \quad (12)$$

We refer to the column vectors of  $U$  and  $V$  as  $u_i$  and  $v_i$ . The image of  $O$  is  $\text{im}(O) = \langle u_1, \dots, u_k \rangle$  and its null space is  $\text{ker}(O) = \langle v_{k+1}, \dots, v_n \rangle$ .

Our idea of choosing a random point  $\mathbf{y}$  in the feasible region of the image  $\text{im}(O)$  followed by inversion is still applicable, only that the inversion is generally not unambiguous. We need to additionally sample a point in the feasible region of the null space  $\text{ker}(O)$  if  $k < n$ . To sample  $\bar{d}$ , we follow these steps:

1. Sample  $\mathbf{y}' \in \mathbb{R}^k$  uniformly in the feasible region of the image  $\text{im}(O)$ .
2. Set  $\mathbf{y}_{1,\dots,k} := \tilde{\Sigma}^{-1} \mathbf{y}'$ .
3. Sample  $\mathbf{y}_{k+1,\dots,n}$  uniformly in the feasible region of the null space  $\text{ker}(O)$ .
4. Set  $\bar{d} = V\bar{c}$ .

We can define both feasible regions again using inequalities on the sampled distances, analogous to the invertible case.

**PDF derivation.** We can use the result from Eq. (7) and insert the value of the marginal integral. Since in this case  $r = s$ , the integral vanishes completely, and we simplify using  $\frac{r}{s} = 1$ :

$$\begin{aligned} & \int_0^\infty p_{\bar{D}'} \left( \frac{r}{s} \bar{d}_{1,n} \right) \frac{r^{n-1}}{s^{n-3}} d\delta_s(r) = p_{\bar{D}'}(\bar{d}_{1,n}) s^2 \\ & = p_{Y_{1,\dots,k}}(\mathbf{y}_{1,\dots,k}) p_{Y_{k+1,\dots,n}}(\mathbf{y}_{k+1,\dots,n} | \mathbf{y}_{1,\dots,k}) s^2 \\ & = p_{Y'_{1,\dots,k}}(\mathbf{y}_{1,\dots,k}) \left( \prod_{i=1}^k \sigma_k \right) p_{Y_{k+1,\dots,n}}(\mathbf{y}_{k+1,\dots,n} | \mathbf{y}_{1,\dots,k}) s^2 \quad (13) \\ & = \frac{1}{V_{\text{img}}} \left( \prod_{i=1}^k \sigma_k \right) \frac{1}{V_{\text{null}}(\mathbf{y}_{1,\dots,k})} s^2, \end{aligned}$$

where we define  $V_{\text{img}}$  and  $V_{\text{null}}$  as the volumes of the feasible regions in the image and null space, respectively. If  $n = 2$ , this PDF is equivalent to OMNEE [HWD22] (see Appendix B).

**Discussion** Our sampling achieves the goal of keeping the part of the PDF in Eq. (13) constant if  $k = n$ . Otherwise sampling the null space introduces a non-constant term, since the volume of its feasible region depends on  $\mathbf{y}_{1,\dots,k}$ . Therefore we would have to sample  $\mathbf{y}_{1,\dots,k}$  non-uniformly in order to cancel this term. In addition, just computing the volume of a feasible region in null space quickly turns intractable: the general case of computing the volume of a halfspace intersection in  $n$  dimensions is NP-hard [DF88]. In summary, this approach does not seem promising for scaling to many vertices.

#### 4. Sampling the number of vertices

One key advantage of bridge sampling is that it provides the freedom to choose the vertex count for a connecting subpath. So far, we

only addressed sampling bridges of predetermined vertex counts, leaving aside the question of choosing a suitable vertex count.

The reference for vertex count is the analog scattering process in the medium: We would like to match the distribution of the number of bridge vertices to the collision count of paths in the medium weighted by their contribution. Several medium properties influence this distribution. High density in relation to connection distance leads to more collision events on average. Losses of contribution due to these collisions are reduced by a high scattering albedo (reduced absorption) and a strong forward scattering phase function (reduced out-scattering).

The sampling decision for bridge vertex counts needs to consider these effects subject to medium properties. This is not a simple task, even when assuming a homogeneous medium. For example in transmittance estimation, collision along a connection in the medium is described by a Poisson process [GMH\*19]. This tells us the distribution of collision counts for the given track length, i.e. the connection distance. In our case, the track length is not known in advance, since it is itself a result of our bridge sampling.

Therefore we resort to precomputing the expected contribution in a homogeneous medium. For each vertex count and phase function parameter, we store the precomputed contribution as a 1D function of the product  $s\mu_t$ . During sampling, we recover a distribution for sampling by evaluating this function for each possible vertex count and normalizing to the sum (see Algorithm 2). In the following, we explain our precomputation.

#### 4.1. Defining the bridge estimator

We define the  $n$ -bridge estimator that computes the transport between two vertices  $\mathbf{a} = \mathbf{x}_0$  and  $\mathbf{b} = \mathbf{x}_n$  with  $n - 1$  scattering events in between as:

$$B_n = \frac{f_{\text{bridge}}(\bar{\mathbf{x}}_{1,n-1})}{p_{\bar{X}}(\bar{\mathbf{x}}_{1,n-1})}, \quad B_1 = \frac{T(\mathbf{a}, \mathbf{b})}{s^2} \quad (14)$$

where  $f_{\text{bridge}}$  is the measurement contribution for the bridge subpath without phase functions and cosines at  $\mathbf{a}$  and  $\mathbf{b}$ :

$$f_{\text{bridge}}(\bar{\mathbf{x}}_{1,n-1}) = \left( \prod_{i=1}^n \frac{T(\mathbf{x}_{i-1}, \mathbf{x}_i)}{d_i^2} \right) \left( \prod_{i=1}^{n-1} \mu_s f_s(\dots) \right). \quad (15)$$

$B_1$  introduces no additional vertices and represents the deterministic connection done by conventional next-event estimation. For the complete transport, we sum over all estimators:  $\mathcal{B}^\infty = \sum_{n=1}^\infty B_n$ . As the contribution of many  $B_n$  will be quite low, depending on the configuration, our goal is to stochastically evaluate only a single term with probability  $P_n$ :

$$P_n = \frac{B_n}{\mathcal{B}^\infty}. \quad (16)$$

This is a good estimator of the sum, as each  $B_n$  has low variance individually. To handle the residual variance of  $B_n$ , we target the square root of its second moment [VKK18, Appendix B]:

$$P_n \sim \sqrt{\mathbb{E}[B_n^2]} \quad (17)$$

by precomputing it for homogeneous media.

## 4.2. Precomputing $\sqrt{\mathbb{E}[B_n^2]}$ for homogeneous media

With the assumption of a homogeneous medium, we can precompute an approximation to  $\sqrt{\mathbb{E}[B_n^2]}$  for use as  $P_n$ , which is the probability for constructing  $n - 1$  inner bridge vertices. We assume an infinite medium with constant  $\mu_s, \mu_t$ , and phase function, as well as full visibility and perfect importance sampling of the phase function. Note that our assumptions do not have to strictly hold in practice, since we use them for importance sampling only. Together with the PDF from Eq. (10),  $B_n$  reduces to:

$$B_n = \frac{\mu_s^{n-1} \|\bar{d}_{1,n}\|_1^n \exp(-\mu_t \|\bar{d}_{1,n}\|_1)}{s^3 (n-1)!}. \quad (18)$$

The only remaining random variable in the estimator is the total track length of the bridge  $\|\bar{d}_{1,n}\|_1 =: D_\Sigma$ . By linearity, we can thus reduce  $\sqrt{\mathbb{E}[B_n^2]}$  to the expectation of a function  $b_n$  of  $\mu_t D_\Sigma$ :

$$\begin{aligned} \sqrt{\mathbb{E}[B_n^2]} &= \sqrt{\mathbb{E} \left[ \left( \frac{\mu_s^{n-1} D_\Sigma^n \exp(-\mu_t D_\Sigma)}{s^3 (n-1)!} \right)^2 \right]} \\ &= \frac{\mu_s^{n-1} \sqrt{\mathbb{E}[b_n(\mu_t D_\Sigma)^2]}}{\mu_t^n s^3 (n-1)!}, \end{aligned} \quad (19)$$

with  $b_n(x) := x^n e^{-x}$ . The distribution of  $D_\Sigma$  depends on the phase function, number of vertices  $n$ , and the connection distance  $s$ . For each combination of  $2 \leq n \leq n_{\max}$  and phase function parameters, we precompute and store an approximation to the 1D function

$$\tilde{b}_n(s\mu_t) = \frac{\sqrt{\mathbb{E}[b_n(s\mu_t D_\Sigma^t)^2]}}{(n-1)!}, \quad (20)$$

where  $D_\Sigma^t := s^{-1} D_\Sigma$  is the normalized track length. In our computed approximations, we always use the Henyey-Greenstein phase function. The resulting functions tend to look very smooth (see Fig. 6) and we approximate each function using a single cubic spline. We use Monte Carlo integration with our bridge sampling to evaluate Eq. (20) during precomputation.

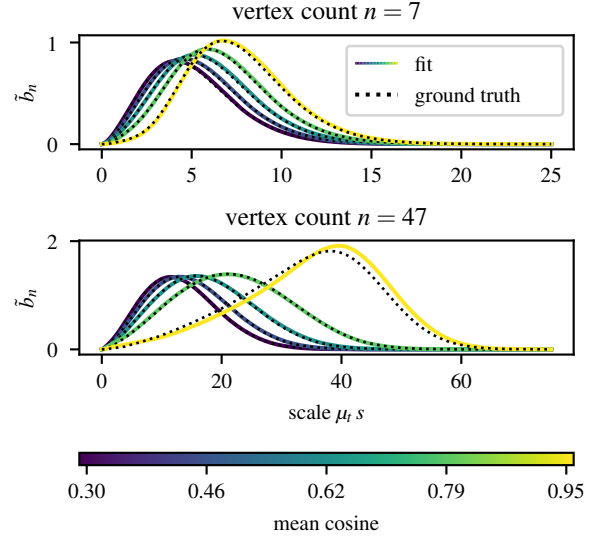
## 4.3. Importance sampling $n$

To recover a probability distribution for sampling  $n$ , we follow Algorithm 2. We evaluate Eq. (19) using the spline approximation and medium parameters for each  $n$  up to  $n_{\max}$  and normalize. We show vertex count distributions computed by our algorithm in Fig. 7. We can choose between bridge sampling and next-event estimation by including  $n = 1$  with  $\sqrt{\mathbb{E}[B_1^2]} = B_1$  (see Eq. (14)).

## 5. Results

### 5.1. Parametric unit test case

We evaluate our method primarily on a parametric unit test case, for which we provide source code [SHD24]. Our test is a point light source in an infinite homogeneous medium. We use the Henyey-Greenstein phase function with mean cosine  $g$ . The scattering albedo is set to 0.99 in our test cases to emphasize multiple scattering. We parameterize the medium extinction by specifying  $s\mu_t$ ,



**Figure 6:**  $\tilde{b}_n$  and our stored approximation for two different vertex counts and different mean cosines. With increasing mean cosine, its peak shifts towards  $s\mu_t = n$ . The function is smooth and mostly approximated well by a spline.

### Algorithm 2 Sample path length

---

```

 $\tilde{P}_1 \leftarrow \frac{T(\mathbf{a}, \mathbf{b})}{s^2} \quad \triangleright \text{see Eq. (14)}$ 
norm  $\leftarrow \tilde{P}_1$ 
for  $n = 2, \dots, n_{\max}$  do
  spline  $\leftarrow$  lookup( $n, g$ )
   $\tilde{P}_n \leftarrow \frac{\mu_s^{n-1}}{\mu_t^n s^3}$  spline( $s\mu_t$ )  $\triangleright$  see Eq. (19)
  norm  $\leftarrow$  norm +  $\tilde{P}_n$ 
select  $n$  with probability  $P_n = \frac{\tilde{P}_n}{\text{norm}}$ 

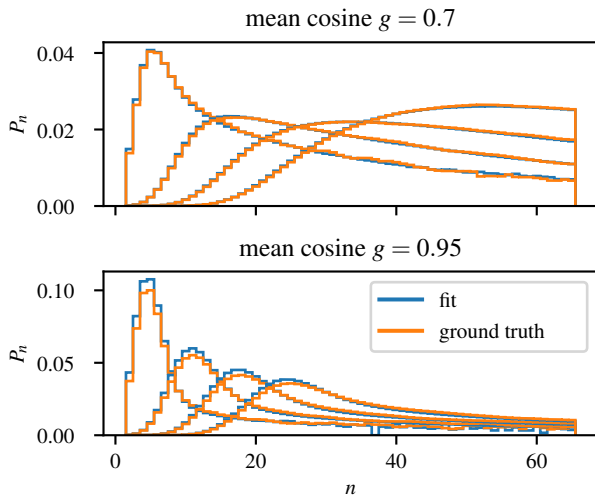
```

---

which is the expected number of collisions along the distance  $s$  between camera and light source. By default, we use our scaling-based distance sampling for bridges. In our results, we use  $k$  to refer to the path length as number of vertices, which includes camera and light source. We abbreviate the number of samples per pixel as spp.

**Normalization and errors.** For comparison to reference images, we use a relative root mean squared error (rRMSE), which is the RMSE normalized by mean image brightness of the reference. We also expose all renderings such that the reference achieves a fixed mean brightness. This is to make noise patterns better comparable between different sets of parameters. The numbers next to inset labels in our result figures are always rRMSE errors.

**Comparison to related work.** We compare our bridge sampling to other methods for extended next-event estimation in volumes: equiangular sampling, MVNEE and OMNEE. In Fig. 8, we show the behavior of these methods in a strongly forward scattering medium ( $g = 0.95$ ) with different combinations of medium extinctions and path lengths. In Fig. 9, we show a similar comparison with fixed extinction, but different combinations of mean cosines and path length. These comparisons include only the forward-scattering



**Figure 7:** Our approximated distribution for importance sampling  $n$  and its reference distribution. We show the distribution evaluated with albedo 0.99 for different scale parameters  $s_{\mu} = 4, 10, 16, 22$ , where the peak shifts to the right with increasing scale. The less peaked  $g = 0.7$  leads to longer track length on average and therefore selects higher vertex counts. A lower albedo would reduce the contribution of these paths and thus the tail of the distribution. Note that we omit the next-event estimation  $n = 1$  case in this plot.

part of the phase function, as OMNEE and MVNEE do not handle backscattering. Our bridge sampling performs well consistently in all these configurations. While OMNEE and MVNEE also show good results for single scattering, they cannot sample longer paths well. For OMNEE and equiangular, this is due to the reliance on path tracing for  $k > 3$ . MVNEE explicitly constructs longer connecting subpaths, similar to our method. However, since it perturbs vertices independently, its sampling does not reflect important vertex correlations caused by phase functions and geometry terms. Our bridge sampling shares the slight degradation of performance towards lower mean cosine. Since it targets the case of pronounced forward scattering, this is expected and generally not concerning. We list render times in Table 2. The runtime performance of our method is comparable to previous methods, with an overhead over path tracing with equiangular sampling of up to  $2\times$ . Note that this test case might not reflect performance in a full renderer, due to the absence of visibility tests and our implementation in Python, which is JIT-compiled and executed on the GPU through JAX.

**Table 2:** Render times for computing the first row of Fig. 8.

time (ms)	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$
Equiangular	5.89	6.55	7.43	8.59	9.80
MVNEE	8.89	11.33	13.65	15.59	16.49
OMNEE	8.90	7.13	7.22	8.74	9.07
Bridges	12.22	11.87	12.79	16.38	18.26

**Scaling-based and constraint-based sampling.** In Fig. 10, we compare our scaling-based sampling (see Section 3.2) with our theoretically superior constraint-based sampling (see Section 3.3). We demonstrate lower standard deviation of the constraint-based estimator when not considering transmittance. However, we do not observe this difference in practice, even when rendering a thin medium. This can be attributed to the difference in the distribution of sampled track lengths as shown in Fig. 11. Depending on the angular configuration, constraint-based sampling can lead to much longer track lengths on average, increasing transmittance variance. Therefore it tends to lose its theoretical benefit if transmittance is significant. An intuitive explanation for this behavior is that in comparison to constraint-based sampling, our scaling-based approach is closer to sampling the transmittance term.

**Forward-only and full sphere phase function.** We examine the variance increase that comes with enabling backwards scattering in our method in Fig. 12. This usually has only a very moderate effect, especially for higher values of  $g$ . Even for these, bridge sampling can sometimes construct paths with large track length when backward scattering is enabled. These would produce outlier samples because of their low PDF, but this effect is usually counterbalanced by their low transmittance. In extremely thin media this is not the case and bridge sampling produces outlier samples.

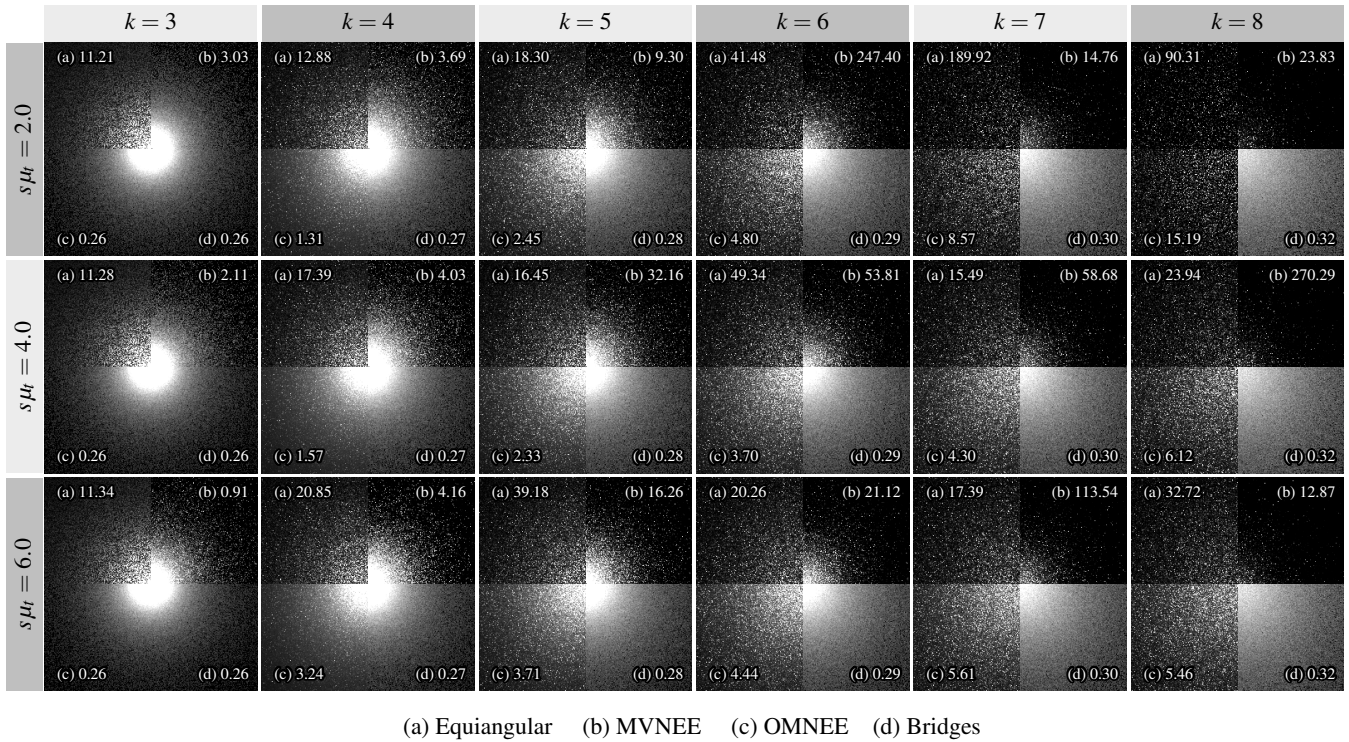
**Vertex count sampling.** We evaluate our method for sampling the vertex count of a bridge in Fig. 13. Besides our presented second moment-based method, we compare against using the first moment in its place, and the Poisson distribution, which is the method used by MVNEE. Since Poisson tends to have a light tail, we further compare against a version of Poisson with a scaled parameter. Sampling the vertex count well seems to matter only starting at longer paths, i.e. above 10 vertices. In these cases, our sampling outperforms Poisson, which starts to show outliers. Our second moment-based sampling shows similar results to our first moment-based sampling, although it can be more robust in thinner media.

**Scaling to long paths.** We examine the scalability of our approach to long paths up to  $k = 100$  vertices in Fig. 14. While there is noticeable variance increase with more vertices, our sampling is still usable, while path tracing with equiangular connections hardly finds any contributing paths. Nevertheless, even with  $g = 0.98$  the distribution of radiance looks very diffuse after so many scattering events, casting in doubt the importance of sampling the exact visibility in these cases.

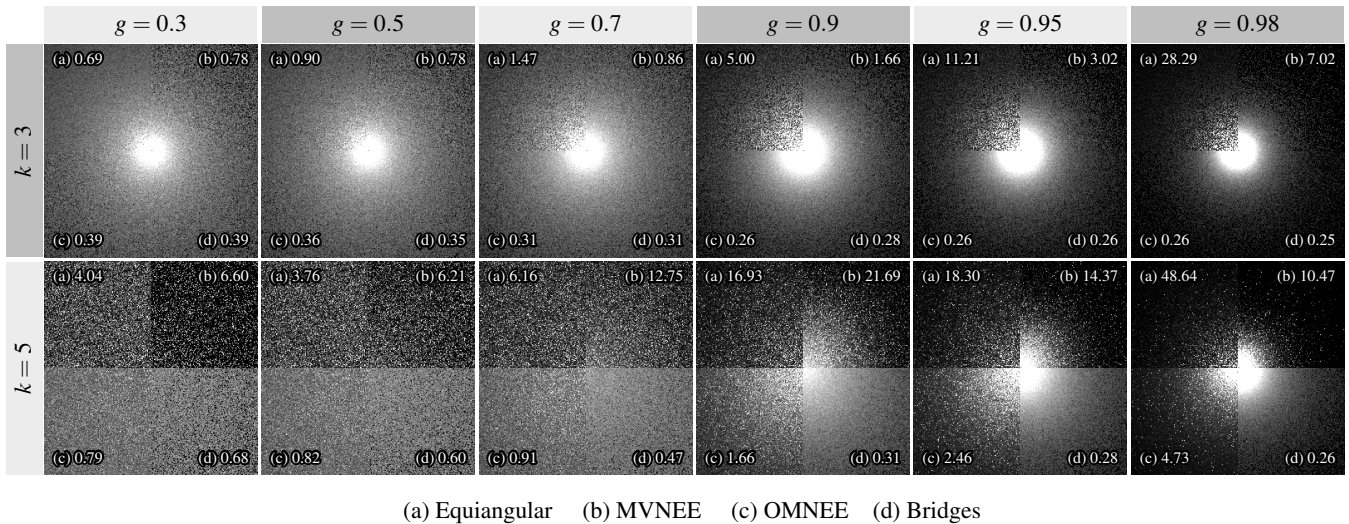
## 5.2. Renderer integration and heterogeneous media

We also implemented our bridge sampling in a spectral renderer with support for heterogeneous media. To construct a bridge, we read  $\mu_t$  at the endpoint of the current path and use this to run Algorithm 2. We continue to construct a tentative path following Algorithm 1, assuming a homogeneous medium with unit extinction. That is, in this step we do not evaluate the heterogeneous medium at path edges or vertices. After path construction is finished, we evaluate the transmittances along the path to compute the measurement contribution. This includes testing each edge for visibility. Unlike in Section 5.1, we do not directly connect camera and light

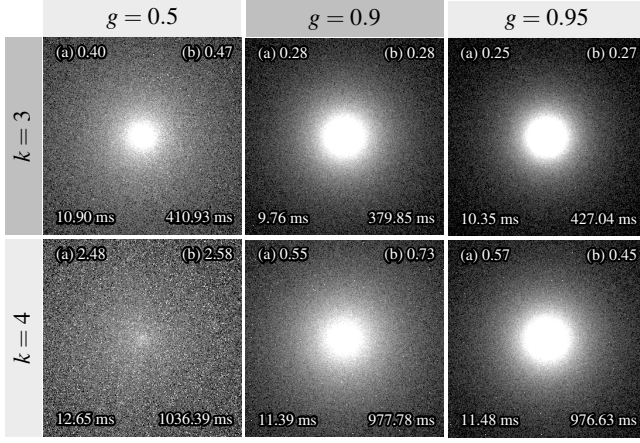




**Figure 8:** Comparison of our sampling with different extinctions  $\mu_t$  and path lengths  $k$ ,  $g = 0.95$ , and 16 spp. Equiangular sampling struggles with the peaked phase function, while all other methods handle the single scattering case  $k = 3$  well. For  $k > 3$ , equiangular and OMNEE have to rely on path tracing a prefix, worsening their results with longer paths. MVNEE also struggles to construct longer paths, since its independent perturbations neglect vertex correlation. Our bridge sampling performs well consistently, with the error deteriorating only slightly towards longer paths.

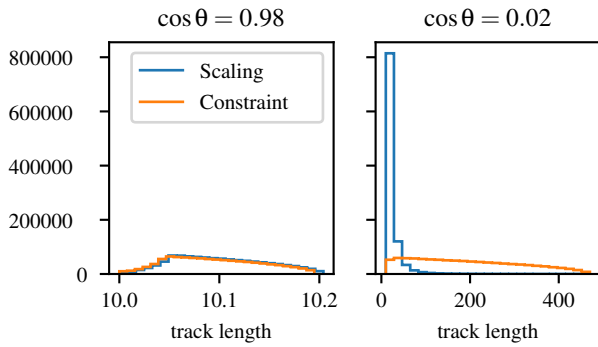
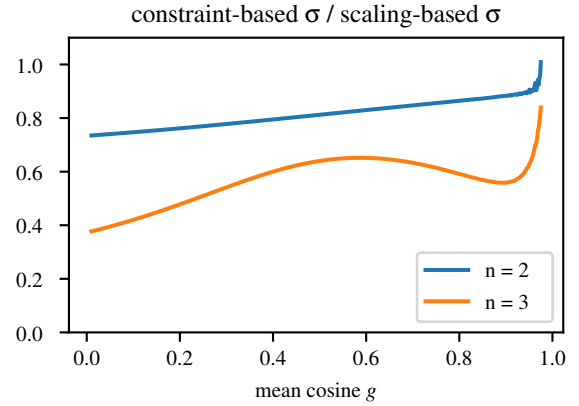


**Figure 9:** Comparison of our sampling with different mean cosines  $g$  and path lengths  $k$ ,  $s\mu_t = 2$  and 16 spp. Equiangular sampling handles moderately forward scattering well, but cannot deal with highly peaked phase functions. Similar to OMNEE, the variance of our bridge sampling increases with less pronounced forward scattering, due to a increased variance in the sampled track lengths. This also lets variance increase faster with path length  $k$ . While this is generally not a problem, as our method is targeted at strong forward scattering, we still achieve relatively low errors consistently.



(a) Scaling-based (b) Constraint-based

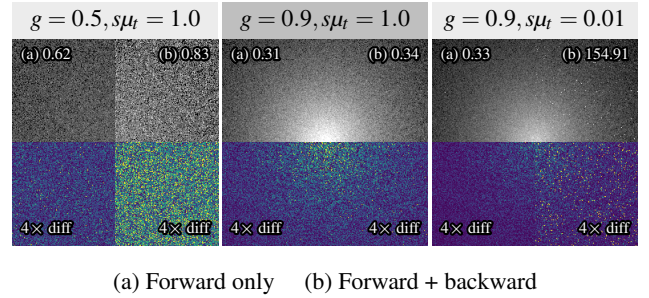
**Figure 10:** Comparison of our scaling-based bridge sampling against our constraint-based sampling with  $n = 2$  ( $k = 3$ ) and  $n = 3$  ( $k = 4$ ). Left: Renderings with 16 spp in a thin medium with  $s\mu_t = 0.1$ . Although constraint-based sampling sometimes achieves slightly lower error, differences are negligible. Right: We compare the ratio of standard deviations  $\sigma$  of both estimators without transmittance as a function of mean cosine  $g$ . In this theoretical comparison, constraint-based sampling achieves distinctly lower standard deviation especially for low  $g$ , but the ratio approaches 1 towards  $g = 1$ .



**Figure 11:** Histograms of sampled total track lengths for two different angular configurations of bridges with 3 edges in a plane and  $s = \|\mathbf{x}_n - \mathbf{x}_0\|_2 = 10$ . With near-parallel directions (left,  $\cos \theta = 0.98$ ), the track length is close to  $s$  for both sampling methods. Constraint-based sampling leads to a much wider distribution of track lengths with near-orthogonal directions (right,  $\cos \theta = 0.02$ ).

sources via bridges in this implementation. Therefore the single-scattering contribution does not benefit from our method in the following comparisons.

In Figure 1, we show a medium with density varying in space and time (for motion blur). We approximate Mie scattering with a 50% mixture of two Henyey-Greenstein lobes with strong forward scattering mean cosine of  $g = 0.98$  and a backward lobe of  $g = -0.4$  to show performance of the algorithm also in the presence of backward scattering. Our sampling technique provides a clear benefit over standard volumetric path tracing with next event estimation. Note that this setup is intentionally challenging for our algorithm since it contains spot emitters with non-diffuse angular character-

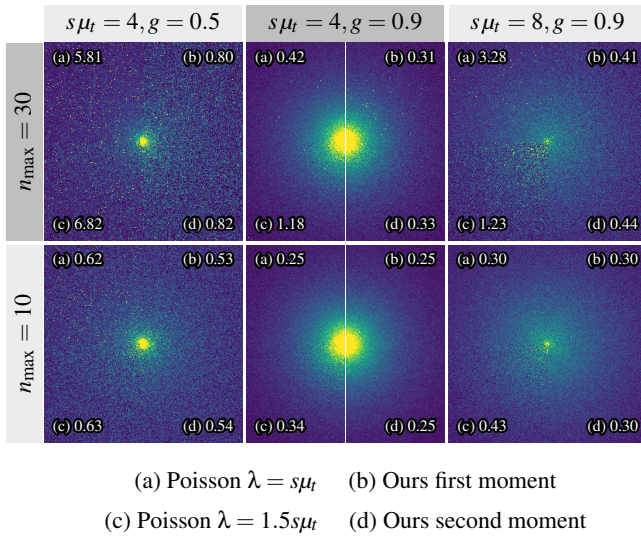


(a) Forward only (b) Forward + backward

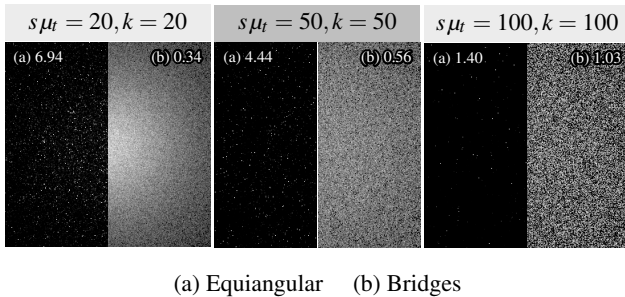
**Figure 12:** Enabling backward scattering leads to a slight increase in variance with  $g = 0.5$ . With stronger forward scattering ( $g = 0.9$ ), the backwards hemisphere has only negligible extra contribution. In a very thin medium with  $s\mu_t = 0.01$ , long paths caused by backwards scattering produce outlier samples, best visible in the difference image. For this comparison, we use a path length of  $k = 5$ , 16 spp, and the same exposure for (a) and (b).

istics, heterogeneous volume properties, backward scattering, and occlusion, violating all assumptions of our technique.

Figures 15 and 16 shows a scene with an occluder casting light shafts into a medium with varying parameters. We compare our method (Bridges) against path tracing with next-event estimation (PT/NEE) and bidirectional path tracing (BDPT). BDPT handles moderate forward-scattering ( $g = 0.5$ ) well, but struggles with strongly-peaked phase functions ( $g = 0.95$ ). The sampling quality of our method deteriorates with increasing medium heterogeneity. One issue here might be our simplistic path length sampling, which is based on the end vertex density. This leads to sampling longer paths than necessary in dense regions of the medium.



**Figure 13:** Comparison of different strategies for selecting vertex counts for bridge sampling with 32 spp. We show this comparison in false-color to highlight outlier samples. Significant differences emerge only for many vertices ( $n_{\max} > 10$ ). Poisson sampling undersamples these bridges with many vertices, even with a scaled parameter, causing outliers. Using our first moment-based sampling largely mitigates this problem, although only our second moment-based sampling is robust in the case with  $s\mu_t = 4, g = 0.9$  (top row, center).



**Figure 14:** Our bridge sampling can scale to very long paths, as shown in this comparison for paths with up to  $k = 100$  vertices,  $g = 0.98$  and 16 spp. Variance eventually increases with  $k$ , but remains at a moderate level. Note that the error for equiangular is unreliable here, as it produces only few samples with significant contribution.

**Overhead** Our method comes with some run-time overhead as compared to simple next-event estimation. First, multiple edges will have to be tested for visibility, incurring an extra cost in the ray tracing when initially traversing the BVH from the root. Secondly, computing the transmittance in a heterogeneous medium has a cost for each voxel that is accessed. Since next-event estimation steps through voxels along the direct connection, this is going to be cheaper than our longer track lengths.

To match the runtime of simple volumetric path tracing in Fig. 1, we use a Russian roulette decision whether to construct a bridge at an end vertex or not, with a survival probability of 30%. That is, for this maximum path length of 10, sampling bridges is about  $3\times$  as expensive as standard next event estimation. The probability of 30% turned out to result in about equal rendering time for a fixed sample count in this scene, in turn increasing variance. Practical applications should rather base this decision on a more global argument whether or not a connection is expected to yield a good contribution [VK16].

In Fig. 17, we show our method on a scene with more interesting geometry and visibility. We render with a maximum path length of 14 vertices. Even though the medium is homogeneous in this scene, the overhead of our method is higher than in Figs. 15 and 16 due to higher cost of visibility tests. Nevertheless, our method achieves lower error in an equal-time comparison against PT/NEE and BDPT.

## 6. Conclusion and future work

We presented *bridge sampling*, a technique to connect light transport path end points to light sources via multiple scattering and an arbitrary number of intermediate vertices. This technique can sample all the reciprocal squared distance and internal phase function terms, for arbitrary phase functions (including both forward and backward scattering). We showed how this works well for a variety of parameters in homogeneous volumes, up to 100 extra vertices and compared against current volume sampling techniques specializing in peaked forward scattering. The technique is simple to implement and, while based on consideration of homogeneous volumes, has clear benefit when applied to heterogeneous volumes.

The resulting PDF does not consider the phase function or emission distribution function at either end point of the bridge. The phase function can easily be included by the same strategy as standard next event estimation: in addition to sampling a bridge, also extend the path by sampling the phase function. Including the emitter term may be an interesting future problem to consider, especially for strongly directional spot lights. One approach could be to generalize our method by using it to form connections in bidirectional path tracing. This could also enable further applications of our method where the light source is outside of the medium, e.g. to subsurface scattering or rendering of clouds.

Since the marginalization over  $\mathbf{x}_n$  worked out analytically, in the future it might be possible to use this approach to derive a closed form formulation of the transport in homogeneous media with complex phase functions.

## References

- [dEo16] D’EON, EUGENE. *A Hitchhiker’s Guide to Multiple Scattering*. 2016. URL: <http://www.eugenedeon.com/hitchhikers3>.
- [Dev86] DEVROYE, LUC. *Non-Uniform Random Variate Generation*. 1986. DOI: [10.1007/978-1-4613-8643-8\\_12](https://doi.org/10.1007/978-1-4613-8643-8_12).
- [DF88] DYER, MARTIN E. and FRIEZE, ALAN M. “On the complexity of computing the volume of a polyhedron”. *SIAM Journal on Computing* 17.5 (1988), 967–974 6.

- [GKH\*13] GEORGIEV, ILIYAN, KŘIVÁNEK, JAROSLAV, HACHISUKA, TOSHIYA, et al. “Joint Importance Sampling of Low-Order Volumetric Scattering”. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 32.6 (Nov. 2013), 1–14. DOI: [10/gbd5qs3](https://doi.org/10/gbd5qs3).
- [GMH\*19] GEORGIEV, ILIYAN, MISSO, ZACKARY, HACHISUKA, TOSHIYA, et al. “Integral Formulations of Volumetric Transmittance”. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 38.6 (Nov. 8, 2019), 154:1–154:17. DOI: [10/dfn6](https://doi.org/10/dfn6).
- [HDF15] HANIKA, JOHANNES, DROSKE, MARC, and FASCIONE, LUCA. “Manifold Next Event Estimation”. *Computer Graphics Forum (Proceedings of Eurographics Symposium on Rendering)* 34.4 (June 2015), 87–97. DOI: [10.1111/cgf.126813](https://doi.org/10.1111/cgf.126813).
- [HEV\*16] HERHOLZ, SEBASTIAN, ELEK, OSKAR, VORBA, JIŘÍ, et al. “Product Importance Sampling for Light Transport Path Guiding”. *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering)* 35.4 (2016), 25:1–25:19. DOI: [10.1111/cgf.129503](https://doi.org/10.1111/cgf.129503).
- [HWD22] HANIKA, JOHANNES, WEIDLICH, ANDREA, and DROSKE, MARC. “Once-More Scattered Next Event Estimation for Volume Rendering”. *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering)* 41.4 (2022). DOI: [10.1111/cgf.145833](https://doi.org/10.1111/cgf.145833), 6, 12.
- [HZE\*19] HERHOLZ, SEBASTIAN, ZHAO, YANGYANG, ELEK, OSKAR, et al. “Volume Path Guiding based on Zero-Variance Random Walk Theory”. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 38.3 (June 2019). DOI: [10.1145/32306353](https://doi.org/10.1145/32306353).
- [Jak13] JAKOB, WENZEL. “Light Transport on Path-Space Manifolds”. PhD thesis. Cornell University, Aug. 2013 3.
- [JMM\*14] JARABO, ADRIAN, MARCO, JULIO, MUNOZ, ADOLFO, et al. “A Framework for Transient Rendering”. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 33.6 (Nov. 2014), 177:1–177:10. DOI: [10/gfznb83](https://doi.org/10/gfznb83).
- [Kd14] KŘIVÁNEK, JAROSLAV and D’EON, EUGENE. “A Zero-variance-based Sampling Scheme for Monte Carlo Subsurface Scattering”. *SIGGRAPH Talks*. 2014, 66:1–66:1. DOI: [10/gfzq7n3](https://doi.org/10/gfzq7n3).
- [KF12] KULLA, CHRISTOPHER and FAJARDO, MARCOS. “Importance Sampling Techniques for Path Tracing in Participating Media”. *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering)* 31.4 (June 2012), 1519–1528. DOI: [10/f35f4k3](https://doi.org/10/f35f4k3).
- [KGH\*14] KŘIVÁNEK, JAROSLAV, GEORGIEV, ILIYAN, HACHISUKA, TOSHIYA, et al. “Unifying points, beams, and paths in volumetric light transport simulation”. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 33.4 (July 2014). DOI: [10.1145/2601097.26012193](https://doi.org/10.1145/2601097.26012193).
- [NGHJ18] NOVÁK, JAN, GEORGIEV, ILIYAN, HANIKA, JOHANNES, and JAROSZ, WOJCIECH. “Monte Carlo Methods for Volumetric Light Transport Simulation”. *Computer Graphics Forum (Proceedings of Eurographics State of the Art Reports)* 37.2 (May 1, 2018), 551–576. DOI: [10/gd2jqj3](https://doi.org/10/gd2jqj3).
- [PVG19] PEDIREDLA, ADITHYA, VEERARAGHAVAN, ASHOK, and GKIOULEKAS, IOANNIS. “Ellipsoidal Path Connections for Time-Gated Rendering”. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 38.4 (July 2019). DOI: [10/gf5jbm3](https://doi.org/10/gf5jbm3).
- [SHD24] SCHÜSSLER, VINCENT, HANIKA, JOHANNES, and DACHSBACHER, CARSTEN. *Implementation of Bridge Sampling for Connections via Multiple Scattering Events*. June 2024. DOI: [10.5281/zenodo.120685297](https://doi.org/10.5281/zenodo.120685297).
- [VHH\*19] VORBA, JIŘÍ, HANIKA, JOHANNES, HERHOLZ, SEBASTIAN, et al. “Path Guiding in Production”. *ACM SIGGRAPH Courses*. 2019, 18:1–18:77. DOI: [10.1145/3305366.33280913](https://doi.org/10.1145/3305366.33280913).
- [VK16] VORBA, JIŘÍ and KŘIVÁNEK, JAROSLAV. “Adjoint-Driven Russian Roulette and Splitting in Light Transport Simulation”. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 35.4 (July 2016), 42:1–42:11. DOI: [10/f89mcz11](https://doi.org/10/f89mcz11).
- [VKK18] VÉVODA, PETR, KONDAPANENI, IVO, and KŘIVÁNEK, JAROSLAV. “Bayesian Online Regression for Adaptive Direct Illumination Sampling”. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 37.4 (July 2018), 125:1–125:12. DOI: [10/gd52ss6](https://doi.org/10/gd52ss6).
- [WHD17] WEBER, PASCAL, HANIKA, JOHANNES, and DACHSBACHER, CARSTEN. “Multiple Vertex Next Event Estimation for Lighting in Dense, Forward-Scattering Media”. *Computer Graphics Forum (Proceedings of Eurographics)* 36.2 (Apr. 2017), 21–30. DOI: [10/gbm2gf3](https://doi.org/10/gbm2gf3).

## Appendix A: Alternative view on scaling-based sampling

Sampling  $n$  independent exponentially distributed random numbers and scaling them to unit  $L^1$  norm is a method to generate uniform random points on the unit  $n$ -simplex with surface area  $\frac{1}{(n-1)!}$ . This is a special case of sampling the Dirichlet distribution with  $\alpha = 1$  [Dev86, p. 594]. An alternative view on our sampling algorithm is that we first sample a point on the unit simplex and then project it accordingly. Since the scaling transformation is bijective in this case, this view does not require explicit marginalization.

## Appendix B: Equivalence to OMNEE

We show that our constraint-based sampling (Section 3.3) has the same PDF as OMNEE [HWD22] for the case  $n = 2$  and forward-only scattering. Without loss of generality, we can express  $O$  using a rotation  $\tilde{R}$  and the angle  $\theta$  between  $\omega'_1$  and  $\omega'_2$ :

$$O = \tilde{R} \begin{pmatrix} 0 & \sin \theta \\ 0 & 0 \\ 1 & \cos \theta \end{pmatrix}. \quad (21)$$

With  $a = \sqrt{1 + \cos \theta}$  and  $b = \sqrt{1 - \cos \theta}$  its SVD is

$$O = \tilde{R} \begin{pmatrix} b\sqrt{\frac{1}{2}} & a\sqrt{\frac{1}{2}} & 0 \\ 0 & 0 & 1 \\ a\sqrt{\frac{1}{2}} & -b\sqrt{\frac{1}{2}} & 0 \end{pmatrix} \begin{pmatrix} a & 0 \\ 0 & b \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \sqrt{\frac{1}{2}} & \sqrt{\frac{1}{2}} \\ -\sqrt{\frac{1}{2}} & \sqrt{\frac{1}{2}} \end{pmatrix}. \quad (22)$$

The arc length of the feasible region is found as

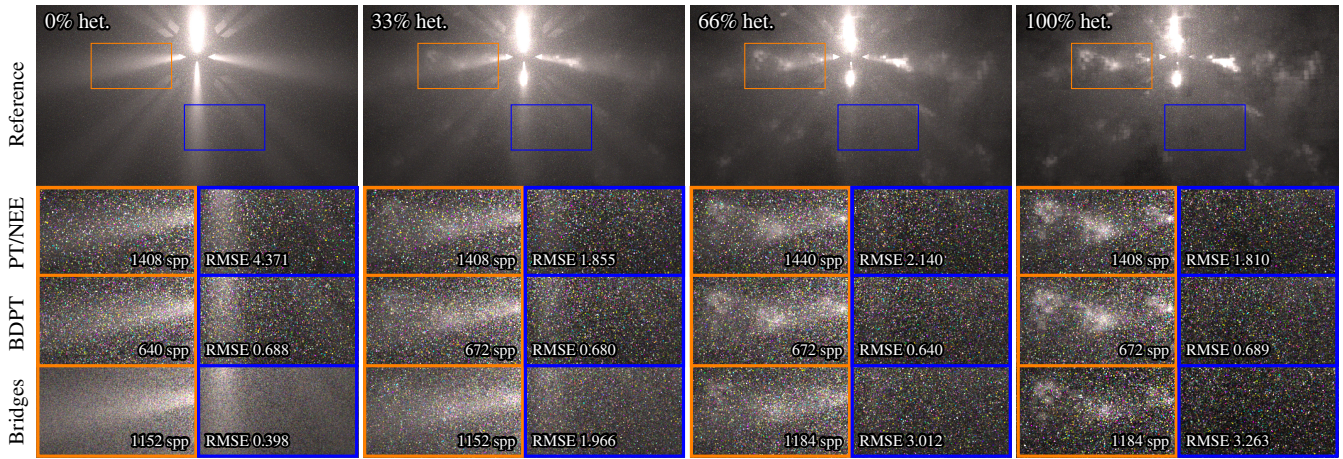
$$s \arccos(n_1 \cdot n_2) = s \arccos(\cos \theta) = s \theta \quad (23)$$

using the normal vectors that define its intersected halfspaces

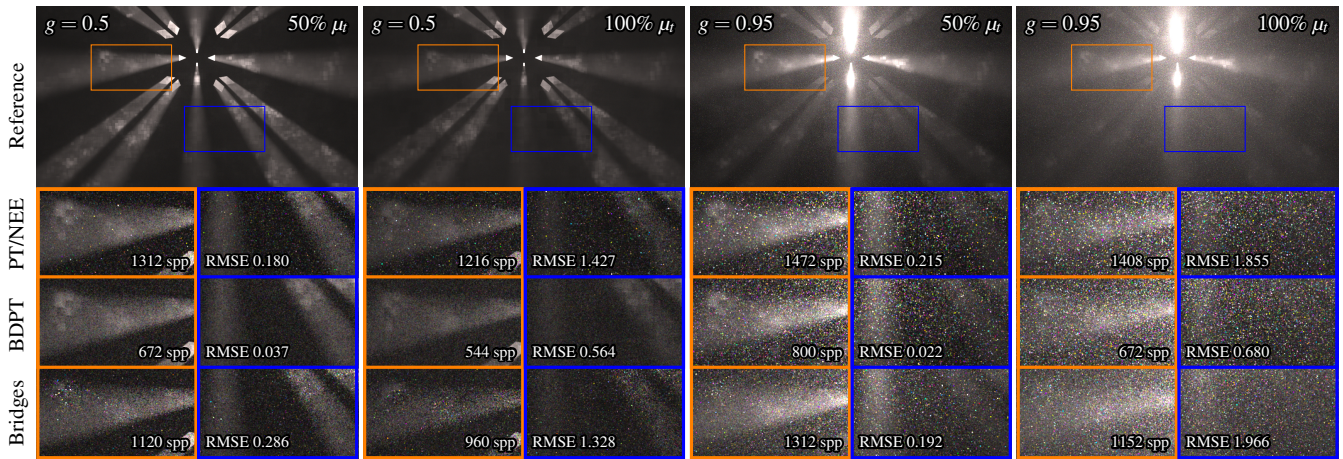
$$n_1 = \frac{1}{z} \begin{pmatrix} a^{-1}\sqrt{\frac{1}{2}} \\ b^{-1}\sqrt{\frac{1}{2}} \end{pmatrix}, \quad n_2 = \frac{1}{z} \begin{pmatrix} -a^{-1}\sqrt{\frac{1}{2}} \\ b^{-1}\sqrt{\frac{1}{2}} \end{pmatrix}, \quad (24)$$

where  $z = \sqrt{\frac{1}{2a^2} + \frac{1}{2b^2}}$ . Inserting into Eq. (13), we obtain the result from OMNEE

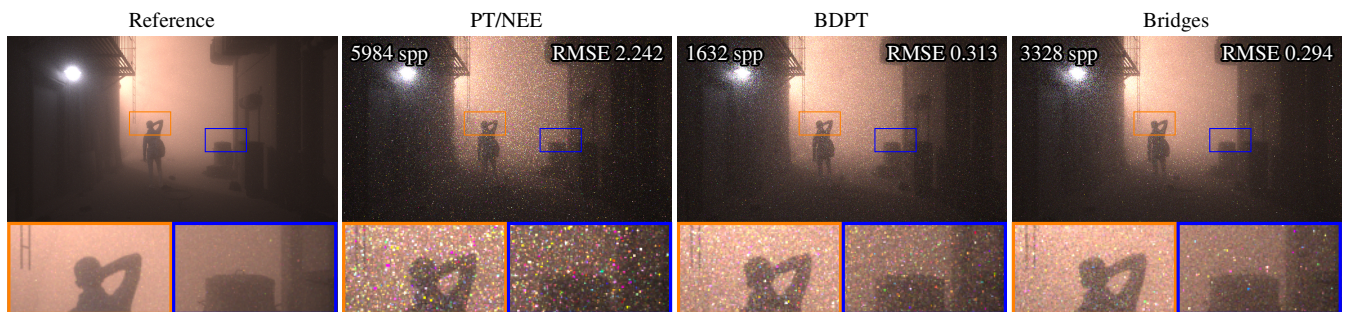
$$\frac{1}{s\theta} abs^2 = \frac{s\sqrt{1 - \cos^2 \theta}}{\theta} = \frac{s \sin \theta}{\theta}. \quad (25)$$



**Figure 15:** A partially shadowed light source in a dense, forward-scattering ( $g = 0.95$ ) medium of varying heterogeneity (het.), rendered in equal time (180s). Our bridge sampling handles the multiple-scattering contribution well in homogeneous media, but deteriorates with increasing heterogeneity.



**Figure 16:** Equal-time comparison (180s) for different parameters of the scene from Fig. 15 with 33% heterogeneity. While BDPT does not handle strongly-peaked phase functions ( $g = 0.95$ ) well, it outperforms our sampling with less pronounced forward-scattering ( $g = 0.5$ ). Likewise, in thinner media ( $50\% \mu_t$ ), multiple-scattering is not as distinct and our method shows less improvement.



**Figure 17:** Equal-time comparison (600s) with more complex geometry. The alley is mainly lit by a street lamp behind the corner and filled with dense, homogeneous fog (HG mixture of  $g = 0.98$  and  $g = -0.4$ ), leading to long paths with significant contribution. Our method is slowed down by its increased number of visibility tests, but still outperforms PT/NEE and BDPT due to improved sampling quality.