Reversible Jump Metropolis Light Transport using Inverse Mappings

BENEDIKT BITTERLI, Dartmouth College, ETH Zürich, and Disney Research
WOJCIECH JAKOB, École Polytechnique Fédérale de Lausanne (EPFL)
JAN NOVÁK, Disney Research
WOJCIECH JAROSZ, Dartmouth College and Disney Research

We study Markov Chain Monte Carlo (MCMC) methods operating in primary sample space and their interactions with multiple sampling techniques. We observe that incorporating the sampling technique into the state of the Markov Chain, as done in Multiplexed Metropolis Light Transport (MMLT), impedes the ability of the chain to properly explore the path space, as transitions between sampling techniques lead to disruptive alterations of path samples. To address this issue, we reformulate Multiplexed MLT in the Reversible Jump MCMC framework (RJMCMC) and introduce *inverse* sampling techniques that turn light paths into the random numbers that would produce them. This allows us to formulate a novel perturbation that can locally transition between sampling techniques without changing the geometry of the path, and we derive the correct acceptance probability using RJMCMC. We investigate how to generalize this concept to non-invertible sampling techniques commonly found in practice, and introduce *probabilistic* inverses that extend our perturbation to cover most sampling methods found in light transport simulations. Our theory reconciles the inverses with RJMCMC yielding an unbiased algorithm, which we call *Reversible Jump MLT* (RJMLT). We verify the correctness of our implementation in canonical and practical scenarios and demonstrate improved temporal coherence, decrease in structured artifacts, and faster convergence on a wide variety of scenes.

CCS Concepts: • **Computing methodologies → Ray tracing**;

Additional Key Words and Phrases: Ray tracing, photorealistic rendering

ACM Reference format:
Benedikt Bitterli, Wenzel Jakob, Jan Novák, and Wojciech Jarosz. 2017. Reversible Jump Metropolis Light Transport using Inverse Mappings. *ACM Trans. Graph.* 1, 1, Article 1 (January 2017), 12 pages.
DOI: 10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

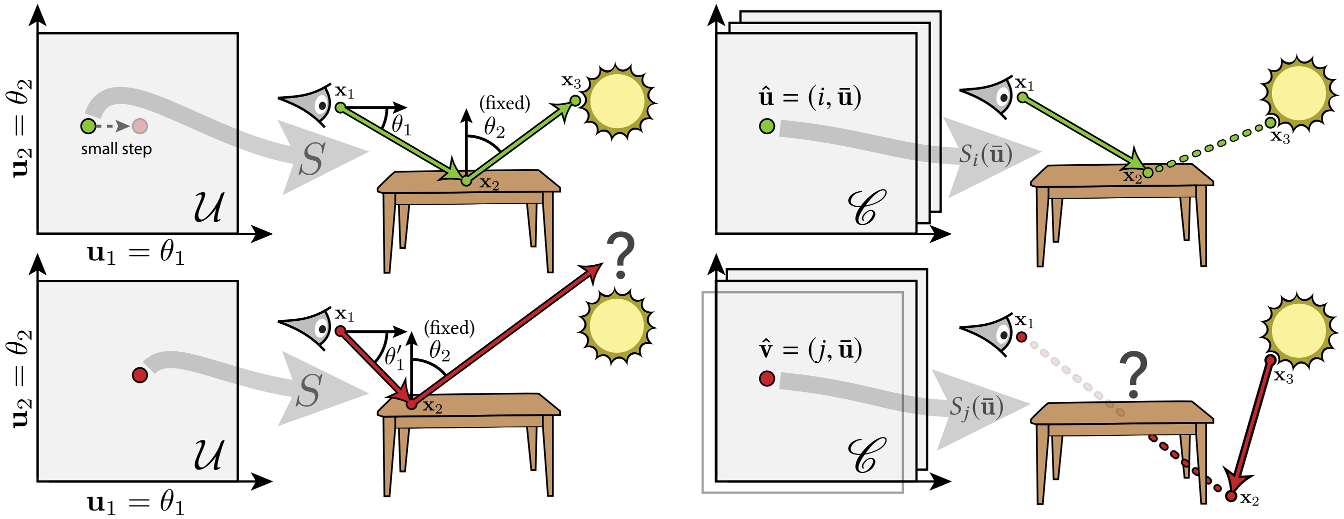
Monte Carlo rendering algorithms simulate the propagation of light by sampling random paths connecting the light source and a virtual sensor. In scenes with complex materials, geometry, or lighting, the space of light paths is large and high-dimensional, but the subset of paths that contribute significantly to the image occupy only a narrow subspace. This makes rendering a notoriously difficult sampling problem even for state-of-the-art unbiased Monte Carlo techniques.

Markov Chain Monte Carlo (MCMC) rendering methods, such as *Metropolis Light Transport* (MLT) [Veach and Guibas 1997] generate a statistically dependent sequence of samples. This dependence augments the sampling process with a form of short-term memory that amortizes the search for important light paths by locally exploring newly discovered regions, which typically allows MCMC methods to handle such problematic scenarios more effectively. In its original path space formulation, MLT allows great flexibility in exploiting knowledge of the integration problem at hand, but requires crafting specialized perturbation strategies for different types of light transport (e.g. caustics, subsurface scattering). This increases its implementation complexity and reduces its generality in modern scenes that contain a combination of many effects.

Kelemen et al. [2002] later proposed a much simpler MCMC rendering approach—*Primary Sample Space MLT* (PSSMLT)—which retrofits Metropolis sampling to existing Monte Carlo methods by treating them as abstract path samplers, and manipulating the random numbers they consume. This view brings many benefits, including ease of implementation and the ability to leverage existing sophisticated importance sampling strategies. *Multiplexed MLT* (MMLT) [Hachisuka et al. 2014] further improves efficiency by allowing the Markov Chain to adaptively select the bidirectional sampling techniques that have high contribution. Unfortunately, operating in the space of random numbers can also make these methods significantly less successful at locally exploring important regions of the state space. For instance, since most sampling schemes construct paths incrementally, a small change in the random numbers used by one vertex on the path generally leads to a ripple change that affects all subsequent vertices, as illustrated in Figure 1(a). In addition, changing path sampling strategies (e.g. starting the path at the light vs. the camera) means that the random numbers for constructing a path are reinterpreted as input to a different sampling strategy, producing an entirely different path, as illustrated in Figure 1(b). Both scenarios turn small perturbations of the random numbers into large, disruptive changes to the path. This inhibits the ability of the Markov Chain to explore the state space locally and increases the likelihood of the chain getting "stuck" in small parts of the state space, exacerbating structured artifacts and temporal instability.

In this paper, we propose *Reversible Jump MLT* (RJMLT), which partially bridges the gap between the flexibility of path-space MLT and the simplicity of PSSMLT-type methods. To accomplish this, we first recast (Section 3) MMLT in the framework of Reversible Jump MCMC [Green 1995], which provides us a mathematical foundation for reasoning about the aforementioned problems. We then enrich the set of mappings available to PSSMLT methods with *inverse* sampling techniques that turn light paths into the random numbers that would produce them. This allows us to formulate a new perturbation that can locally transition between sampling techniques for the same path (Section 4). We consider practical implications of such inverses (Sections 5–6) and reconcile them with the RJMCMC framework. We evaluate the correctness and efficiency of the resulting algorithm (Section 7), showing that we can achieve significantly higher acceptance rates than MMLT and reduce noise and the erratic convergence behavior of PSSMLT methods. Finally, we discuss other ways that reversible jumps and inverse mappings can be used to improve PSSMLT methods in the future (Section 8).

2017. 0730-0301/2017/1-ART1 $15.00
DOI: 10.1145/nnnnnnn.nnnnnnn



(a) Ripple effect of changes to later vertices

(b) Disruptive changes when switching strategies

Fig. 1. Fundamental issues of path sampling using primary sample space: **(a)** Perturbations in PSSMLT cause a ripple change that propagates to later vertices: here, a perturbation of the outgoing direction at the camera causes a large-scale change of the vertex on the light source $\mathbf{x}_3$. In such cases, it can be advantageous to switch to a different sampling strategy, for instance one that explicitly samples a position on a light source rather than intersecting it by chance. **(b)** Such strategy changes are possible using a multiplexed primary sample space such as that of MMLT. However, changing strategies generally leads to a large-scale change to the path geometry that causes the proposed path to be rejected with high probability. The RJMLT technique proposed in this paper introduces efficient strategy perturbations that leave the path geometry intact.

2 RELATED WORK & MATHEMATICAL BACKGROUND

We now discuss the most relevant prior work and establish a common mathematical notation to more firmly relate prior approaches to our contributions. We aggregate the most important terms of our notation in Table 1 for reference.

Table 1. Table of notation

Symbol	Explanation
$\mathcal{P}$	Path Space
$\mathcal{U}$	Primary Sample Space
$\mathcal{P}^k$, $\mathcal{U}^k$	Spaces of paths of length k
o_k	Size of $\mathcal{U}^k$
$\bar{\mathbf{x}} = \mathbf{x}_1 \ldots \mathbf{x}_n$, $\bar{\mathbf{y}} = \mathbf{y}_1 \ldots \mathbf{y}_n$	Light paths
$\bar{\mathbf{u}} = \mathbf{u}_1 \ldots \mathbf{u}_n$, $\bar{\mathbf{v}} = \mathbf{v}_1 \ldots \mathbf{v}_n$	Random number vectors
$\bar{\gamma} = \gamma_1 \ldots \gamma_n$	Auxiliary variables for path inversion
$f(\bar{\mathbf{x}})$	Path contribution function
$p(\bar{\mathbf{x}})$	PDF of sampling $\bar{\mathbf{x}}$
$C(\bar{\mathbf{x}}) = f(\bar{\mathbf{x}})/p(\bar{\mathbf{x}})$	PSSMLT importance function
$S_i(\bar{\mathbf{u}})$	i-th sampling technique of BDPT
$w_i(\bar{\mathbf{u}})$	MIS weight of i-th technique
$p_i(\bar{\mathbf{u}})$	PDF of i-th technique
$f_i(\bar{\mathbf{u}})$	$f(S_i(\bar{\mathbf{u}}))$
$C_i(\bar{\mathbf{u}}) = w_i(\bar{\mathbf{u}}) f_i(\bar{\mathbf{u}})/p_i(\bar{\mathbf{u}})$	MMLT importance function
$\hat{\mathbf{u}} = (i, \bar{\mathbf{u}})$	Multiplexed state with explicit technique
$T(\bar{\mathbf{u}} \to \bar{\mathbf{v}})$	Proposal density of $\bar{\mathbf{v}}$ from state $\bar{\mathbf{u}}$
$r(\bar{\mathbf{u}} \to \bar{\mathbf{v}})$	Probability of accepting proposal $\bar{\mathbf{v}}$
$J[A]$	Jacobian Matrix of function A

Current physically based rendering algorithms can all be expressed as approximations to a *measurement equation*, which considers integrals of the form:

$$I_j = \int_{\mathcal{P}} h_j(\bar{\mathbf{x}}) f(\bar{\mathbf{x}})\,\mathrm{d}\mu(\bar{\mathbf{x}})\,. \tag{1}$$

This equation computes the value of a measurement I_j (usually, a pixel) in terms of an integral over all possible *light paths*. The integrand is composed of the pixel reconstruction filter $h_j(\bar{\mathbf{x}})$ and the contribution $f(\bar{\mathbf{x}})$ of light path $\bar{\mathbf{x}}$ (note the bar notation, which indicates quantities organized as multiple vertices). The domain of all such light paths is *path space* $\mathcal{P}$. We can decompose $\mathcal{P}$ further into subdomains $\mathcal{P}^k$ that contain only light paths of a fixed length k, which together form $\mathcal{P} = \bigcup_{k=2}^{\infty} \mathcal{P}^k$.

Monte Carlo Integration. Traditional Monte Carlo rendering—first pioneered with Kajiya's [1986] rendering equation and the path tracing algorithm—approximate Equation (1) using the estimator:

$$I_j \approx \frac{1}{N} \sum_{i=1}^{N} \frac{h_j(\bar{\mathbf{x}}^{(i)}) f(\bar{\mathbf{x}}^{(i)})}{p(\bar{\mathbf{x}}^{(i)})}, \tag{2}$$

which averages N independently sampled light paths $\bar{\mathbf{x}}^{(i)}$, where $p(\bar{\mathbf{x}}^{(i)})$ is the probability density of the i-th path sample. These probability densities directly determine the variance of the estimator, and it is thus vital that $p(\bar{\mathbf{x}}^{(i)})$ is approximately proportional to the integrand to obtain an efficient integrator. Bidirectional variants of path tracing (BDPT) [Lafortune and Willems 1993, 1996; Veach and Guibas 1994] build on the reciprocal nature of light transport to construct partial light paths from both camera and light sources and connect them to yield large families of sampling strategies. For a length-k path (i.e. with k segments), BDPT considers $k + 2$ different sampling techniques where each strategy importance samples different parts of the integrand. The resulting estimators are typically combined into a joint estimator using *multiple importance sampling* (MIS) [Veach and Guibas 1995].

Markov Chain Monte Carlo. MCMC Rendering algorithms such as MLT [Veach and Guibas 1997] mark a significant departure from classical Monte Carlo rendering. By using local exploration, these methods can remarkably generate a correlated sequence of samples (light paths) with a density that is exactly proportional to an arbitrary non-negative function $C(\mathbf{x})$. Usually this is accomplished using Metropolis-Hastings [Hastings 1970; Veach and Guibas 1997] steps: given current state $\mathbf{x}$, a *proposal* state $\mathbf{y}$ is drawn from a proposal distribution $T(\mathbf{x} \longrightarrow \mathbf{y})$ which is accepted and becomes the next state of the Markov Chain with probability

$$r(\mathbf{x} \longrightarrow \mathbf{y}) = \min\left\{1, \frac{C(\mathbf{y})T(\mathbf{y} \longrightarrow \mathbf{x})}{C(\mathbf{x})T(\mathbf{x} \longrightarrow \mathbf{y})}\right\};$$

otherwise, it is rejected and the process repeats anew. Under relatively weak conditions on $T(\mathbf{x} \longrightarrow \mathbf{y})$, the distribution of Markov Chain states will converge to $C(\mathbf{x})$.

Path-Space MLT. In the original MLT algorithm by Veach and Guibas [1997], the Markov Chain operates on path space $\mathcal{P}$ and the target density C is the path contribution $f(\bar{\mathbf{x}})$. This method relies on a set of two mutation strategies (which change a path significantly) and three perturbation strategies (which change it only slightly). While mutations ensure that all states are reachable, perturbations are the main tool by which the Markov Chain locally explores the state space. Subsequent generalizations of these strategies extend the perturbations to arbitrary chains of specular interactions [Jakob 2013] and rough materials [Hanika et al. 2015; Kaplanyan et al. 2014]. Other variations of MLT include Energy Redistribution Path Tracing [Cline et al. 2005], which independently simulates a large number of Markov Chains that are stratified in image space, and Gradient-Domain MLT [Lehtinen et al. 2013], which estimates gradients and uses them to reconstruct the rendered image. A severe problem with MLT and variations thereof is that admissible light paths lie on a lower-dimensional subset of path space with non-Euclidean structure, which tremendously complicates the design of mutations and perturbations. Existing strategies only target specific physical effects, but modern scenes contain many interactions types that may even occur simultaneously on a single light path.

Primary Sample Space MLT. PSSMLT [Kelemen et al. 2002] takes a different approach that avoids these problems: it is based on the observation that each iteration of a classical Monte Carlo rendering algorithm (e.g. BDPT) consumes a limited number of random variates and uses them to sample one or more light paths. We can express this sampling process more formally in terms of an abstract sampling scheme $S(\bar{\mathbf{u}})$, which takes a vector $\bar{\mathbf{u}}$ of random numbers drawn uniformly, and generates a light path. The domain of these vectors forms *primary sample space* $\mathcal{U}$. Similar to path space, we can further decompose this space over path lengths, such that $\mathcal{U} = \bigcup_{k=1}^{\infty} \mathcal{U}^k$ and $\mathcal{U}^k = [0,1]^{o_k}$. The dimensionality o_k is chosen to be large enough such that all light paths of length k can be sampled by S. This allows us to rewrite Equation (2) explicitly in terms of the sampling scheme, using the relation $\mathbf{x}_i = S(\mathbf{u}_i)$.

PSSMLT operates the Markov Chain in primary sample space $\mathcal{U}$, instead of path space $\mathcal{P}$, and relies on the underlying rendering algorithm to convert such samples into light paths using the sampling scheme(s) at its disposal. The target function $C(\bar{\mathbf{u}})$ then becomes the composition of the path contribution and sampling scheme,

$$C(\bar{\mathbf{u}}) = (f \circ S)(\bar{\mathbf{u}})/p(\bar{\mathbf{u}}). \tag{3}$$

Among its many advantages, PSSMLT admits simple symmetric perturbation strategies due to the Euclidean structure of primary sample space. Additionally, any reflectance model supported by the underlying rendering algorithm is automatically handled by the resulting MCMC algorithm. The generality of this approach has led to numerous applications, e.g. in the context of photon tracing with MCMC sampling [Hachisuka and Jensen 2011], vertex connection merging/unified path sampling [Šik et al. 2016], tempering techniques such as Replica Exchange [Kitaoka et al. 2009], methods that control the Markov Chain by simulating Hamiltonian dynamics [Li et al. 2015], and rendering algorithms steered by arbitrary importance functions [Gruson et al. 2017; Hoberock and Hart 2010]. PSSMLT is most often combined with BDPT [Kelemen et al. 2002], which utilizes $k+2$ different sampling schemes $S_0(\bar{\mathbf{u}}), \ldots, S_{k+1}(\bar{\mathbf{u}})$ to construct paths of length k from shorter eye/camera subpaths. PSSMLT evaluates Equation (3) by generating two subpaths and determining the MIS-weighted contribution for all pairs of vertices. Unfortunately, this sequence of steps is fairly expensive, and many of the vertex pairs will generally contribute little to no energy.

Multiplexed MLT. MMLT [Hachisuka et al. 2014] addresses this problem by informing the Markov Chain about the availability of multiple sampling strategies. It does so by running separate Markov Chains for each path length k, in which the first dimension of primary sample space is used to select one of the $k+2$ techniques of BDPT. A perturbation in MMLT may therefore change both the vector of random numbers, as well as which sampling technique is used to map that vector into a light path. Each Markov Chain uses the modified target function

$$C_j(\bar{\mathbf{u}}) = w_j(\bar{\mathbf{u}})(f \circ S_j)(\bar{\mathbf{u}})/p_j(\bar{\mathbf{u}}), \tag{4}$$

which incorporates the MIS weight $w_j(\bar{\mathbf{u}})$ of the technique. Thus, instead of evaluating all strategies all the time, this choice is adaptively made by the underlying Metropolis-Hastings iteration, allowing the method to focus computation on successful sampling techniques that carry significant energy from the light sources to the camera.

Despite these improvements, MMLT (like PSSMLT) still suffers from structured artifacts and temporal instability since strategy changes and small steps in primary sample space $\mathcal{U}$ lead to large, disruptive changes to the resulting light path (recall Figure 1), inhibiting the Markov Chain's ability to locally explore state space.

Design space of MLT algorithms. At a high level, we consider PSSMLT and MLT to be extremes in a space involving a trade-off between the "opaqueness" of the underlying representation and the design complexity of the resulting algorithm: the random number streams of PSSMLT have no clear physical interpretation, but this also allows for PSSMLT's concise and general perturbation strategy. In contrast, MLT operates on path space, which is a direct encoding of the physical scattering process, but this comes at a cost of significantly increased complexity. MMLT resembles PSSMLT, though the special role of the first dimension of the random number stream makes the representation slightly less opaque. However, incorporating the sampling technique into the state of the Markov Chain inadvertently ties the random number vector to the particular technique for constructing the path and leads to a loss in flexibility, as switching to a different sampling technique effectively amounts to a large scale *mutation* of the path (Figure 1b). Our proposed RJMLT technique lies squarely in between: by allowing an existing path to be turned back into the random numbers that produce it, the distinction between primary sample space and path space ceases to be significant. On the other hand, our method requires the availability of inverse functions for many components of the rendering system.

Reversible Jumps. Our technique builds upon the Reversible Jump MCMC algorithm (RJMCMC) [Green 1995], which was originally developed in the area of Bayesian Statistics. Bayesian applications of MCMC techniques involve sampling from posterior distributions that take some form of evidence (e.g. empirical observations) into account. RJMCMC addresses the case where *multiple* candidate models could describe a given set of observations, and RJMCMC then extends the Metropolis-Hastings algorithm with mutations that transition between these different models. In our context, these models correspond to different bidirectional sampling strategies that can construct the same path using a different number of steps on the camera and light paths.

Concurrent Work. Concurrent to our work, Pantaleoni [2017] bridge path space and primary sample space with a parametric family of mappings and their inverses, and design novel Metropolis-based rendering algorithms based on bidirectional path tracing and density estimation. The underlying idea behind both his work and ours is the same, but is derived using different mathematical frameworks. Notably, our use of RJMCMC allows us to precisely reason about acceptance probabilities in terms of Jacobians. In contrast to Pantaleoni, this enables us to derive the correct acceptance probabilities for layered materials and allows us to design a perturbation with optimal acceptance rates.

Similarly to our work, Otsu et al. [2017] employ invertible mappings to transform samples between (multiplexed) primary-sample space and path space. This allows them to combine perturbation strategies from both spaces, yielding a more robust algorithm that is never much worse than the better of Path-Space MLT and MMLT. However, since these invertible mappings are not used for technique perturbations, switching techniques still results in low acceptance rates as in MMLT. In contrast, we employ inverses to *unconditionally* accept technique perturbations in MMLT. Our method and the concurrent work of Otsu et al. are therefore complementary and their combination would be worth investigating in future work.

3 REVERSIBLE MULTIPLEXED JUMPS

The efficiency gain of MMLT comes from its ability to spend less time on ineffective sampling strategies, but its choice of perturbation strategies inhibit the Markov Chain from transitioning between different sampling techniques. To better analyze this problem and potential solutions, we introduce a novel reformulation of Multiplexed MLT in the Reversible Jump MCMC [Green 1995] framework (RJMCMC). This allows us to reason about perturbations that change sampling techniques in terms of deterministic mappings between spaces. This same framework will later allow us to derive a novel perturbation that leverages inverses to reliably transition between sampling techniques.

We begin by explicitly separating the choice of sampling technique from the rest of primary sample space. The Markov Chain then operates in the product of a discrete and continuous space, $\mathscr{C}^k = \{0, \ldots, k+1\} \times \mathcal{U}^k$ referred to as *multiplexed primary sample space*. In the RJMCMC view, this space is decomposed into subspaces $\bigcup_{i=0}^{k+1} \mathscr{C}_i^k$, where $\mathscr{C}_i^k = \{i\} \times [0,1]^{o_k}$ is the subspace of the i-th sampling technique for paths of length k.

The set of perturbations proposed in Multiplexed MLT will always perturb the position in primary sample space, and may additionally change the technique index. To simplify analysis, we will separate the two concerns and focus on a hypothetical TechniquePerturbation strategy, which keeps the position in primary sample space fixed and only attempts to transition to a different sampling technique. That is, given the current state $\hat{\mathbf{u}} = (i, \bar{\mathbf{u}})$ in $\mathscr{C}_i^k$, the perturbation samples a technique index j from the distribution $T(i \to j)$ and generates the proposal $\hat{\mathbf{v}} = (j, \bar{\mathbf{u}})$ in $\mathscr{C}_j^k$. The distribution $T(i \to j)$ is left unspecified, but for simplicity we assume it to be symmetric, i.e. $T(i \to j) = T(j \to i)$.

In the RJMCMC view, we can model such a perturbation as a *jump* between subspaces. The proposal is generated from the current state by a *deterministic* mapping $h_{ij} : \mathcal{U}^k \to \mathcal{U}^k$ that relates points from one space to the other; that is, $\hat{\mathbf{v}} = (j, h_{ij}(\bar{\mathbf{u}}))$. The RJMCMC acceptance probability for such a proposal is

$$r(\hat{\mathbf{u}} \to \hat{\mathbf{v}}) = \frac{C_j(h_{ij}(\bar{\mathbf{u}}))T(j \to i)}{C_i(\bar{\mathbf{u}})T(i \to j)} \left| J\big[h_{ij}\big](\bar{\mathbf{u}}) \right|, \tag{5}$$

where $J\big[h_{ij}\big]$ is the Jacobian matrix of h_{ij} to account for the density of the mapping. For TechniquePerturbation, the mapping is trivial with $h_{ij}(\bar{\mathbf{u}}) = \bar{\mathbf{u}}$ and an identity Jacobian.

Since only the sampling technique is changed and not the random numbers, we would hope in the MMLT view that the acceptance probability will only depend on the ratio of MIS weights, i.e. on how well the proposed technique samples the current path compared to the current sampling technique. However, expanding and simplifying the terms in Equation (5) yields

$$r(\hat{\mathbf{u}} \to \hat{\mathbf{v}}) = \frac{C_j(\bar{\mathbf{u}})}{C_i(\bar{\mathbf{u}})} \left| \mathbb{I} \right| \tag{6}$$

$$= \frac{w_j(\bar{\mathbf{u}})\, C(S_j(\bar{\mathbf{u}}))}{w_i(\bar{\mathbf{u}})\, C(S_i(\bar{\mathbf{u}}))}, \tag{7}$$

where $\mathbb{I}$ is the identity matrix.

Equation (7) exposes the main problem of this naïve perturbation: Even though the random number vector was not changed, the proposed state uses a *different mapping* to transform that vector into a light path. In general, $S_i(\bar{\mathbf{u}}) \neq S_j(\bar{\mathbf{u}})$, and it is likely that the proposed path (and therefore $C(S_j(\bar{\mathbf{u}}))$) differs from the current path by a significant amount (Figure 1). Such large changes are unlikely to be accepted, which impedes the Markov Chain's ability to transition between different sampling strategies. Multiplexed MLT additionally couples technique changes with perturbations of $\bar{\mathbf{u}}$, but this does not address the problem observed in the naïve TechniquePerturbation: the algorithm suffers from low acceptance rates whenever a proposal changes strategies.

4 INVERTIBLE SAMPLING TECHNIQUES

Ideally, a technique perturbation would leave the current path unchanged while switching techniques. To do so, it must find a new point $\bar{\mathbf{v}}$ in primary sample space such that $S_j(\bar{\mathbf{v}}) = S_i(\bar{\mathbf{u}})$, and jump to $\bar{\mathbf{v}}$ as part of the perturbation.

Assume for now that sampling techniques S_i are well-behaved in the sense that they are *diffeomorphisms*, i.e. all S_i are smooth and possess a smooth inverse. Then, the desired perturbation is easily accomplished using the mapping $h_{ij}(\bar{\mathbf{u}}) = S_j^{-1}(S_i(\bar{\mathbf{u}}))$. Here, we used $S_j^{-1}(\bar{\mathbf{x}})$, which is an *inverse sampling technique* that transforms a light path into the random numbers that would produce it. The Jacobian of this mapping is

$$\begin{aligned}\left|J\left[h_{ij}\right](\bar{\mathbf{u}})\right| &= \left|J\left[S_j^{-1}\circ S_i\right](\bar{\mathbf{u}})\right| &(8)\\ &= \left|J\left[S_j^{-1}\right](S_i(\bar{\mathbf{u}}))\right|\cdot\left|J\left[S_i\right](\bar{\mathbf{u}})\right| &(9)\\ &= \left|J\left[S_j\right](\bar{\mathbf{v}})\right|^{-1}\cdot\left|J\left[S_i\right](\bar{\mathbf{u}})\right|, &(10)\end{aligned}$$

where the last step follows from the inverse function theorem. This allows us to write the Jacobian of the mapping in terms of the Jacobians of the sampling techniques[1].

The Jacobian of a sampling technique is closely related to its PDF. Indeed, if $S_i(\bar{\mathbf{u}})$ is a diffeomorphism, then $\left|J\left[S_i\right](\bar{\mathbf{u}})\right| = p_i(\bar{\mathbf{u}})^{-1}$ [Kelemen et al. 2002]. The acceptance probability of such a proposal is

$$\begin{aligned}r(\hat{\mathbf{u}}\to\hat{\mathbf{v}}) &= \frac{T(j\to i)\,C_j(\bar{\mathbf{v}})}{T(i\to j)\,C_i(\bar{\mathbf{u}})}\frac{p_j(\bar{\mathbf{v}})}{p_i(\bar{\mathbf{u}})} &(11)\\ &= \frac{T(j\to i)\,w_j(\bar{\mathbf{v}})\,f_j(\bar{\mathbf{v}})\,p_j(\bar{\mathbf{v}})^{-1}}{T(i\to j)\,w_i(\bar{\mathbf{u}})\,f_i(\bar{\mathbf{u}})\,p_i(\bar{\mathbf{u}})^{-1}}\frac{p_j(\bar{\mathbf{v}})}{p_i(\bar{\mathbf{u}})} &(12)\\ &= \frac{T(j\to i)\,w_j(\bar{\mathbf{v}})\,f(S_j(\bar{\mathbf{v}}))}{T(i\to j)\,w_i(\bar{\mathbf{u}})\,f(S_i(\bar{\mathbf{u}}))} &(13)\\ &= \frac{T(j\to i)\,w_j(\bar{\mathbf{v}})}{T(i\to j)\,w_i(\bar{\mathbf{u}})}, &(14)\end{aligned}$$

where the last cancellation was obtained with $S_i(\bar{\mathbf{u}}) = S_j(\bar{\mathbf{v}})$, which holds by construction.

Discussion. Equation (14) has several interesting properties. As desired, this acceptance ratio only depends on the relative MIS weight of the proposed and current technique. This allows the Markov Chain to easily transition to the optimal sampling techniques as it explores path space. Additionally, the path contribution function does not appear in the acceptance ratio, and expensive retracing of paths is not required. Notably, the remaining terms in the acceptance ratio can be derived from the current state *before generating the proposal.* This allows us to select an *optimal* proposal distribution with $T(i \to j) = w_j(\bar{\mathbf{u}})$. This will cancel all remaining terms and achieve an acceptance ratio of 1.

We will refer to this perturbation as a *reversible jump*, and the rendering method derived from it as *Reversible Jump MLT.* The method proposed here can be summarized as:

1. Choose a proposal technique j with probability $w_j(S_i(\bar{\mathbf{u}}))$
2. Jump to proposal state $\hat{\mathbf{v}} = (j, S_j^{-1}(S_i(\bar{\mathbf{u}})))$
3. Always accept $\hat{\mathbf{v}}$

5 NON-INVERTIBLE SAMPLING TECHNIQUES

The perturbation introduced in the previous section assumed that sampling techniques are smooth and invertible, but this does not always hold in practice. Sampling methods are not required to be injective and might therefore map several random number vectors to the same path, which introduces ambiguities when attempting to compute inverses. We focus on two cases that are relevant to light transport simulations:

1) Ambiguous Intervals. Primary sample space methods generally apportion a constant number of dimensions to each vertex of a path, of which only a subset may be used at each bounce. Every unused dimension collapses an entire $[0, 1]$ interval in primary sample space onto the same path. Similarly, if a single variate u_i is used to sample a discrete property (e.g. the index of the light source), an interval $u_i \in [a, b]$ may map to the same path. However, when turning an existing path into the random numbers that produce it, we must commit to a specific value of u_i.

2) Mixtures of Sampling Techniques. The second problematic case involves sampling strategies with overlapping support—for instance, consider a simple diffuse-specular BRDF that samples either the diffuse or specular lobe according to a probability α_{diffuse}:

$$S_{\text{phong}}(\mathbf{u}) = \begin{cases} S_{\text{diffuse}}(u_2, \ldots), & \text{if } u_1 < \alpha_{\text{diffuse}},\\ S_{\text{specular}}(u_2, \ldots), & \text{otherwise.}\end{cases} \qquad (15)$$

Following an interaction with the material, most scattered directions can be sampled using *two* distinct vectors of uniform variates corresponding to interactions with the diffuse and specular lobes, respectively. Also, note how the probability density of the sampling scheme is simply the average of the diffuse- and specular PDFs, $p_{\text{phong}}(\mathbf{u}) = (p_{\text{diffuse}}(\mathbf{u}) + p_{\text{specular}}(\mathbf{u}))\,/\,2$, while the Jacobian becomes discontinuous at $u_1 = \alpha_{\text{diffuse}}$:

$$J\left[S_{\text{phong}}\right](\mathbf{u}) = \begin{cases} J\left[S_{\text{diffuse}}\right](u_2, \ldots), & \text{if } u_1 < \alpha_{\text{diffuse}},\\ J\left[S_{\text{specular}}\right](u_2, \ldots), & \text{otherwise.}\end{cases} \qquad (16)$$

Mixtures of sampling techniques lead to non-diffeomorphic mappings, which break the relation between the Jacobian determinant and the probability density $\left|J\left[S\right](\mathbf{u})\right| = p(\mathbf{u})^{-1}$ that we used in the derivation of the acceptance probability for the invertible case in Equation (14).

Extended Path Space. To address these issues, we will first introduce a conceptual construction that extends path space with auxiliary dimensions such that no information about the random numbers is lost during sampling, allowing paths to be inverted exactly. Afterwards, we will introduce probabilistic inverses as a practical solution to non-invertibility and reason about them within the RJMCMC framework.

Suppose that the set of all random number vectors that map to a given path $\bar{\mathbf{x}}$ is given by $\mathcal{M}_{\bar{\mathbf{x}}} = \{\bar{\mathbf{u}} \mid S_j(\bar{\mathbf{u}}) = \bar{\mathbf{x}}\}$. In principle,

[1] To side-step the subtlety that vertices of a light path $\bar{\mathbf{x}} = \mathbf{x}_1 \ldots \mathbf{x}_n \in \mathcal{P}$ reside on 2D subspaces of 3D, we assume that each vertex $\mathbf{x}_i$ can be *locally* parametrized using an orthonormal tangent frame. The concatenation of these local parameterizations then yields a local parameterization of the neighborhood of a light path, which facilitates reasoning about the densities of sampling strategies. For instance, $\left|J\left[S_i\right](\bar{\mathbf{u}})\right|$ refers to the Jacobian determinant of the i-th sampling strategy, which captures the change in volume when a small volume element in $\mathcal{U}$ is mapped to $\mathcal{P}$. This is purely a theoretical concern so these local parameterizations are not required during implementation.

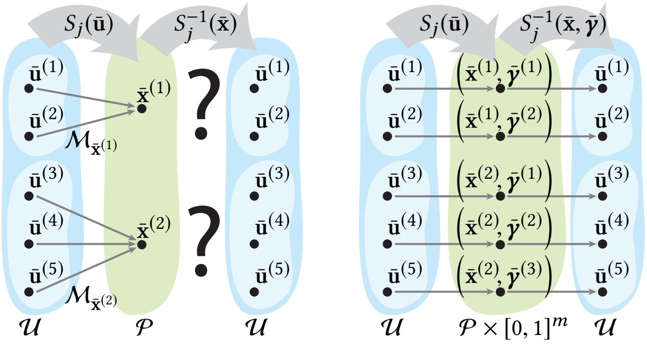

$S_j(\bar{\mathbf{u}})$
$S_j^{-1}(\bar{\mathbf{x}})$
$\bar{\mathbf{u}}^{(1)}$
$\bar{\mathbf{u}}^{(2)}$
$\bar{\mathbf{u}}^{(3)}$
$\bar{\mathbf{u}}^{(4)}$
$\bar{\mathbf{u}}^{(5)}$
$\bar{\mathbf{x}}^{(1)}$
$\bar{\mathbf{x}}^{(2)}$
$\mathcal{M}_{\bar{\mathbf{x}}^{(1)}}$
$\mathcal{M}_{\bar{\mathbf{x}}^{(2)}}$
?
?
$\mathcal{U}$
$\mathcal{P}$
$\mathcal{U}$
$S_j(\bar{\mathbf{u}})$
$S_j^{-1}(\bar{\mathbf{x}}, \bar{\boldsymbol{\gamma}})$
$(\bar{\mathbf{x}}^{(1)}, \bar{\boldsymbol{\gamma}}^{(1)})$
$(\bar{\mathbf{x}}^{(1)}, \bar{\boldsymbol{\gamma}}^{(2)})$
$(\bar{\mathbf{x}}^{(2)}, \bar{\boldsymbol{\gamma}}^{(1)})$
$(\bar{\mathbf{x}}^{(2)}, \bar{\boldsymbol{\gamma}}^{(2)})$
$(\bar{\mathbf{x}}^{(2)}, \bar{\boldsymbol{\gamma}}^{(3)})$
$\mathcal{U}$
$\mathcal{P} \times [0,1]^m$
$\mathcal{U}$

Fig. 2. Non-injective mappings $S_j(\bar{\mathbf{u}})$ (left) lead to ambiguities when transforming from path space to primary sample space. We sidestep this issue by ensuring all mappings are bijective (right); we extend the path space parameterizing its points by an additional parameter $\bar{\boldsymbol{\gamma}}$ produced by $S_j(\bar{\mathbf{u}})$.

we should be able to augment the inverse S_j^{-1} with an additional parameter $\bar{\boldsymbol{\gamma}} \in [0,1]^m$, that disambiguates the inverse so each pair of inputs $S_j^{-1}(\bar{\mathbf{x}}, \bar{\boldsymbol{\gamma}})$ maps to a single entry of $\mathcal{M}_{\bar{\mathbf{x}}}$. The role of $\bar{\boldsymbol{\gamma}}$ will be to encode the “extra” dimensions in $\bar{\mathbf{u}}$ that do not directly sample the path. If such a map can be constructed (assuming a sufficiently large value of m), then we can perform reversible jumps with the following modifications: sampling techniques generate points on an extended path space $\mathcal{P} \times [0,1]^m$, and $S_i(\bar{\mathbf{u}})$ computes the pair $(\bar{\mathbf{x}}, \bar{\boldsymbol{\gamma}})$, where $\bar{\mathbf{x}}$ is the sampled path, and $\bar{\boldsymbol{\gamma}}$ is the auxiliary input that would produce $\bar{\mathbf{u}}$ when used in the inverse $S_i^{-1}(\bar{\mathbf{x}}, \bar{\boldsymbol{\gamma}})$; see Figure 2 for an illustration.

Using the extended path space, no information is lost when transitioning from or to primary sample space, enabling the use of RJMCMC to derive the acceptance probability

$$r(\hat{\mathbf{u}} \to \hat{\mathbf{v}}) = \frac{C_j(\bar{\mathbf{v}})\, T(j \to i)\, \left|J\left[S_i\right](\bar{\mathbf{u}})\right|}{C_i(\bar{\mathbf{u}})\, T(i \to j)\, \left|J\left[S_j\right](\bar{\mathbf{v}})\right|} \tag{17}$$

$$= \frac{C_j(\bar{\mathbf{v}})\, T(j \to i)\, \left|J\left[S_i^{-1}\right](\bar{\mathbf{x}}, \bar{\boldsymbol{\gamma}})\right|^{-1}}{C_i(\bar{\mathbf{u}})\, T(i \to j)\, \left|J\left[S_j^{-1}\right](\bar{\mathbf{y}}, \bar{\boldsymbol{\gamma}})\right|^{-1}} \tag{18}$$

$$= \frac{C_j(\bar{\mathbf{v}})\, T(j \to i)\, \left|J\left[S_j^{-1}\right](\bar{\mathbf{y}}, \bar{\boldsymbol{\gamma}})\right|}{C_i(\bar{\mathbf{u}})\, T(i \to j)\, \left|J\left[S_i^{-1}\right](\bar{\mathbf{x}}, \bar{\boldsymbol{\gamma}})\right|}, \tag{19}$$

where the step from Equation (17) to (18) follows from the inverse function theorem. This construction provides a viable way of supporting non-invertible mappings, but it complicates the implementation as the auxiliary dimensions must be stored and propagated through the entire rendering system.

Probabilistic inverses. We rely on a much lighter-weight solution to resolve ambiguities during path sampling: whenever multiple inverses are available, we simply randomly select one. This is realized by combining strategy change perturbations with an additional step that samples $\bar{\boldsymbol{\gamma}} \in [0,1]^m$ from a uniform distribution. This modification to the proposal results in extra unit factors in the acceptance probability that cancel, hence the acceptance probability of a probabilistic inverse is still given by Equation (19). Since the auxiliary vector $\bar{\boldsymbol{\gamma}}$ is re-sampled as part of every strategy perturbation, it is no longer part of the Markov Chain's state.

6 INVERSES IN PRACTICE

In the previous section, we derived a general framework for handling path sampling methods that are not strictly invertible, and reconciled these ideas with the RJMCMC framework. In this section, we will now discuss inverses in more concrete terms and describe a few simple building blocks that allow correct inversion of many sampling methods used in light transport simulations. We will also discuss how to compute the required Jacobians in more detail.

In practice, light paths are almost always sampled incrementally in a random walk. This corresponds to chaining a series of q low-dimensional sampling techniques, and we can write

$$S_i(\bar{\mathbf{u}}) = \left(g_1\left(\mathbf{u}_1\right), g_2\left(\mathbf{x}_1, \mathbf{u}_2\right), \ldots, g_q\left(\mathbf{x}_{q-1}, \mathbf{u}_q\right)\right), \tag{20}$$

where $\bar{\mathbf{x}}_l = \mathbf{x}_1 \ldots \mathbf{x}_l$ is the path up to vertex $\mathbf{x}_l$, and $g_l(\mathbf{x}_{l-1}, \mathbf{u}_l)$ is the l-th sampling technique along the path, using a subset of the random numbers in $\bar{\mathbf{u}}$. Inverting S_i then reduces to inverting the individual techniques g_l, in an analogous “inverse random walk”. The Jacobian determinant required by Equation (19) turns into a product of determinants for each step, i.e.,

$$\left|J\left[S_i^{-1}\right](\bar{\mathbf{x}}, \bar{\boldsymbol{\gamma}})\right| = \prod_{l=1}^{q} \left|J\left[g_l^{-1}\right]\left(\mathbf{x}_l, \boldsymbol{\gamma}_l\right)\right|.$$

In order to maintain an optimal acceptance ratio and for Equation (14) to hold, we wish for the individual Jacobians $|J[g_l^{-1}](\mathbf{x}_l, \boldsymbol{\gamma}_l)|$ to be equal to the PDF of the corresponding mapping g_l. In the following, we will focus on a single sampling technique g, and drop the subscripts for ease of notation.

Inversion method. Mappings based on the *inversion method* form the basic building blocks of many sampling techniques and are easily handled. Such techniques draw samples from a probability distribution p by mapping uniform variates through the inverse CDF P^{-1}. These mappings are invertible by construction, and to support them in our system we simply require an additional implementation of the CDF P, which serves as a (non-probabilistic) inverse. The Jacobian determinant $\left|J\left[g^{-1}\right]\left(\mathbf{x}, \boldsymbol{\gamma}\right)\right|$ of this inverse is equal to the PDF p of the sampling technique.

In the remainder of this section, we turn to the two sources of non-invertibility discussed in Section 5.

6.1 Ambiguous Intervals

Ambiguous interval arose whenever a dimension u_i in primary sample space could take on any value on an interval $[a,b]$ without changing the generated path. Constructing a probabilistic inverse for this case is fortunately easy: we generate a uniform variate $\gamma \in [0,1]$ and set $u_i = g^{-1}(\gamma)$, where

$$g^{-1}(\gamma) = a + \gamma \cdot (b-a) \quad \text{and} \quad J\left[g^{-1}\right](\gamma) = b - a\,. \tag{21}$$

For entirely unused dimensions of primary sample space this reduces to $g^{-1}(\gamma) = \gamma$ with a unit Jacobian determinant; that is, unused dimensions are simply uniformly re-sampled during inversion and do not influence the acceptance probability.

6.2 Mixtures of Sampling Techniques

Suppose that g consists of a combination of sampling techniques $g_1, \ldots, g_n$ selected at random, where technique g_i is chosen with

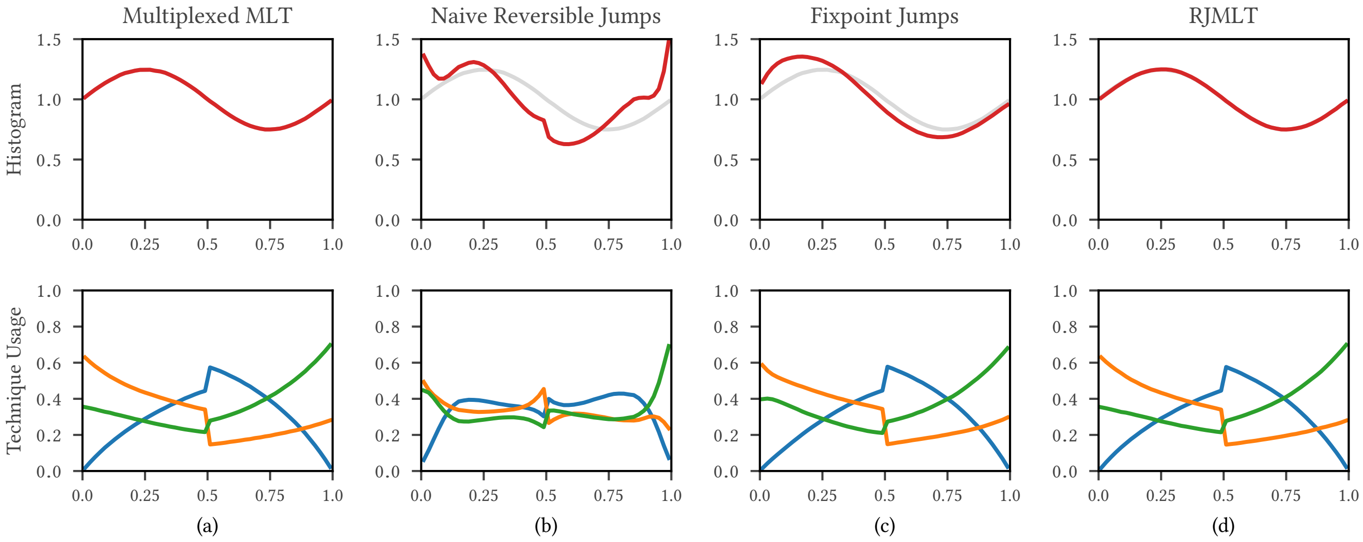


Fig. 3. In a simplified 1D scenario, we compute the stationary distribution (top row) and average use of three available sampling techniques (bottom row) for four different Markov Chain integrators. We demonstrate the correctness of our full approach (d) compared to the Multiplexed MLT ground truth (a). Neglecting the necessary Jacobians (b) or improperly parametrizing inverses (c) leads to skewed distributions, demonstrating the importance of our full theory.

probability α_i. We assume that the technique index t is chosen by the primary sample u_1 such that $\alpha_1 + \ldots + \alpha_{t-1} \leq u_1 < \alpha_1 + \ldots + \alpha_t$.

We now propose a probabilistic inverse that resembles this sampling procedure. First, we randomly select a technique index t from a (yet undisclosed) discrete distribution $T(t)$. We then invert the sample assuming that it was generated by the t-th technique. This disambiguates both which interval the variate u_1 falls into, as well as which mapping g_t^{-1} should be used. For a fixed t, the resulting inverse and Jacobian determinant are then

$$g^{-1}(\mathbf{x}, \boldsymbol{\gamma}) = \left(\alpha_1 + \ldots + \alpha_{t-1} + \gamma_1 \cdot \alpha_t,\ \ g_t^{-1}(\mathbf{x}, \gamma_2, \ldots)\right), \tag{22}$$

$$\left|J\left[g^{-1}\right](\bar{\mathbf{x}}, \bar{\boldsymbol{\gamma}})\right| = \alpha_t \cdot \left|J\left[g_t^{-1}\right](\bar{\mathbf{x}}, \boldsymbol{\gamma}_2 \ldots)\right|. \tag{23}$$

This presence of a technique index t extends the proposal state generated by RJMLT: In addition to selecting the j-th technique of BDPT as MMLT does, the strategy perturbation also selects which of the n sampling techniques should be used to invert g, which yields a slightly modified acceptance probability

$$r((\hat{\mathbf{u}}, t_{\bar{\mathbf{u}}}) \to (\hat{\mathbf{v}}, t_{\bar{\mathbf{v}}})) = r(\hat{\mathbf{u}} \to \hat{\mathbf{v}}) \frac{T(t_{\bar{\mathbf{u}}})}{T(t_{\bar{\mathbf{v}}})}. \tag{24}$$

The last step is to pick a concrete distribution T. Any distribution that samples strategy t with nonzero probability if it could potentially have produced $\mathbf{x}$ is in principle admissible.

We use

$$T(t) = \frac{\alpha_t \cdot \left|J\left[g_t^{-1}\right](\bar{\mathbf{x}}, \bar{\boldsymbol{\gamma}})\right|}{\sum_{s=1}^{n} \alpha_s \cdot \left|J\left[g_s^{-1}\right](\bar{\mathbf{x}}, \bar{\boldsymbol{\gamma}})\right|}, \tag{25}$$

which cancels out the Jacobian (23) in the acceptance ratio, and results in an acceptance rate of 1.[2]

Final algorithm For completeness, we now state the outline of our complete strategy perturbation:

(1) Choose a proposal technique j with probability $w_j(S_i(\bar{\mathbf{u}}))$
(2) Jump to proposal state $\hat{\mathbf{v}} = (j, S_j^{-1}(S_i(\bar{\mathbf{u}})))$
  - If there are ambiguous intervals for elements of $\bar{\mathbf{v}}$, sample them uniformly (Equation 21)
  - If S_j uses a mixture of sampling techniques, select one randomly according to T (Equation 25)

(3) Always accept $\hat{\mathbf{v}}$.

There is one small caveat to step (3): in rare cases, it might be impossible to invert a path due to numerical error. We detect such cases by checking if the path cannot be sampled by the method that we wish to invert (for example, if a direction lies in the wrong hemisphere), and reject proposals in this case.

7 RESULTS

We validate our theory with two implementations of our proposed method. Our first implementation performs MCMC integration in a simplified 1D scenario. The purpose of this simulation is not to compare performance across different techniques, but to demonstrate the correctness of our approach. We find that the high dimensionality of the light transport problem and the use of large steps can often mask subtle biasing issues of Markov Chain rendering methods, whereas these are immediately apparent in the 1D case.

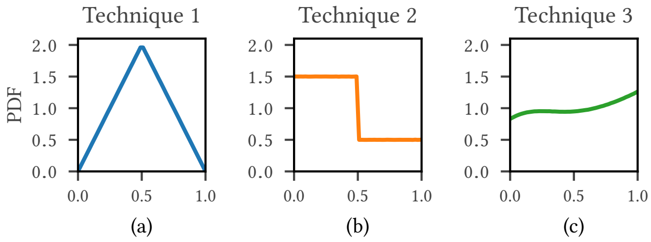
Our second implementation adds reversible jumps to a general ray tracing renderer as an auxiliary perturbation on top of Multiplexed MLT. We evaluate the performance of this implementation in detail in Section 7.2.

7.1 1D Markov Chain Integrator

Our simulation performs Markov Chain integration on the $[0, 1]$ interval with a sinusoid target distribution (Figure 3 (a), top). Each integration method has three sampling techniques at its disposal with PDFs corresponding to: a triangular distribution (a), a step

[2] T has an intuitive interpretation: If g_t is itself not a nested mixture of techniques, then $T(t)$ is simply the discrete probability that $\bar{\mathbf{x}}$ was generated by the t-th technique.

function distribution (b), and a mixture distribution (c) of a sinusoid and a linear function, as illustrated below.



The primary sample space in this scenario has three dimensions: One used for selecting the currently used technique, one for sampling a position and one for selecting a subtechnique in mixture distributions (only utilized in the third sampling technique). We demonstrate the results for this setup in Figure 3.

The histogram of the Markov Chain states (top row) should converge to the sinusoid target distribution if the algorithm is correct. We also record a secondary histogram that keeps track of which sampling techniques were used by the Markov Chain on average at particular points in space (bottom row). For the integration methods considered here, this histogram should converge to the MIS weights of the sampling techniques.

We implemented the equivalent of Multiplexed MLT in this 1D framework as a baseline integrator (Figure 3 (a)), which represents the ground truth for both histograms. In order to amplify potential issues in the perturbations, we disable large steps for all integrators, and only allow the Markov Chain to explore the 1D space using small steps and reversible jumps.

To demonstrate the importance of correctly performing reversible jumps, we implement two versions of our algorithm that use only parts of our theory. The first integrator performs reversible jumps without taking into the account the Jacobian when transitioning between subspaces, and omits the right-most determinant term in Equation (19). Unlike traditional perturbations, reversible jumps are fully deterministic aside from selecting one of the available inverses, and one may be tempted to treat them as a discrete transition with trivial acceptance probability. However, neglecting to incorporate the Jacobian of the transition leads to a heavily distorted stationary distribution (Figure 3 (b)) and a skewed use of the different sampling techniques.

The second integrator includes Jacobians in the acceptance probability (Equation (19)) and inverts sampling mixtures nearly correctly (Equation (22)), but does not fully parametrize $\mathcal{M}_{\bar{\mathbf{x}}}$. Because the third dimension of primary sample space is used to select a subtechnique, an ambiguous interval arises when inverting the mixture distribution. Rather than parametrizing these intervals as described in Section 6.1, this integrator always returns a fixed point inside the interval. Although this approach may not immediately appear incorrect, computing inverses in this manner leads to a biased distribution (Figure 3 (c)).

Finally, we implemented our full approach, using both Jacobians and parametrized inverses. Incorporating our full theory leads to a correct stationary distribution, as shown in Figure 3 (d).

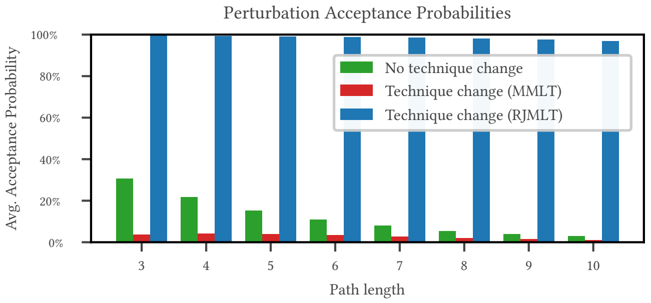


Fig. 4. We visualize the average acceptance probability of perturbations in the Living Room scene, broken down over path length. To demonstrate the benefit of our method, we differentiate between proposals that change the sampling technique, and proposals that leave it unchanged. For Multiplexed MLT, proposals that attempt to change the sampling technique have a significantly lower acceptance probability compared to proposals that do not, and it is difficult for the Markov Chain to transition between sampling techniques. Conversely, technique changes in Reversible Jump MLT are nearly always accepted.

7.2 General Ray Tracing Renderer

Our most general implementation adds RJMLT to an existing rendering system. We implement reversible jumps as an additional perturbation on top of Multiplexed MLT. At each step, the Markov Chain selects either a large step, small step or a technique change with a fixed probability distribution (10%, 85% and 5%, respectively). Unlike Multiplexed MLT, we explicitly separate the technique index from the rest of primary sample space, such that small step perturbations may not change the sampling technique.

We tested our implementation on an array of 8 different test scenes, and gather a series of metrics to evaluate our approach. The scenes feature a wide variety of sampling methods, including layered materials, microfacet models, environment- and area lighting. Because MLT-based rendering methods can only compute a correct solution up to a global scaling factor, we run a path tracer on each scene for several hours to compute noise-free estimates of these factors and supply them to both MMLT and RJMLT. This allows us to exclude differences in scaling factors from the comparison.

The use of correlated samples can make a fair comparison between MCMC-based rendering methods challenging. Because Markov Chains generate most samples with local exploration, integration error can manifest in different forms, such as “splotches”, streaks, wrongly positioned caustics or entirely missing transport modes. The presence of many such artifacts is symptomatic of the Markov Chain “getting stuck” in local maxima, and suggests that inappropriate perturbations are employed by the integrator. These artifacts have implications for both the visual appearance and quantitative convergence, and we provide a series of comparisons to highlight the differences between RJMLT and MMLT:

Convergence Plots. Inappropriate perturbations lead to a slower convergence of the Markov Chain to its target distribution. To perform a quantitative analysis of this fact, we track the MSE (compared to a path traced reference image) of both RJMLT and MMLT as a function of render time. Because of the use of correlated samples, a single run of an MCMC integrator may not be representative, and we run 50 instances of each integrator with different random seeds to get a measure of its average behavior. In addition,

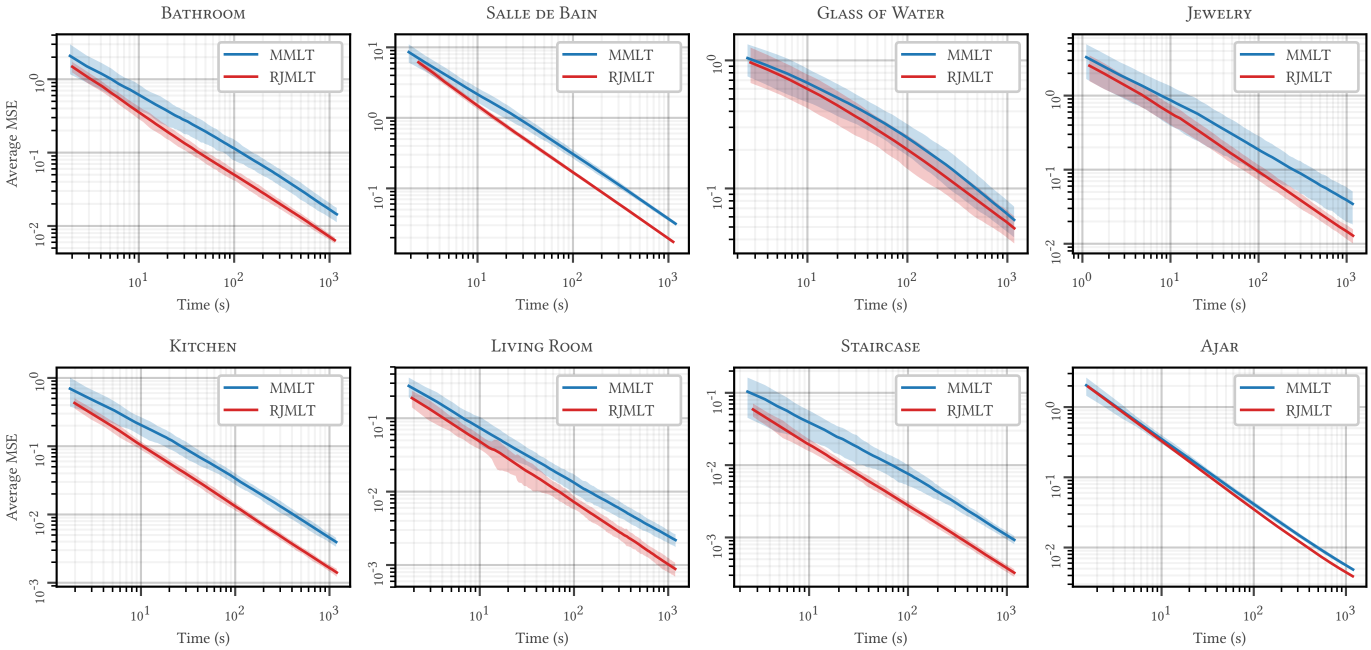


Fig. 5. To compare the convergence of our method to previous work, we measure the MSE over time of 50 independent runs of RJMLT and MMLT in all 8 of our test scenes. We visualize the average MSE curve over time (thin curve), as well as the standard deviation of the MSE across all 50 individual runs (shaded areas). At equal time, RJMLT both has a lower MSE on average, as well as less variation across different runs.

the *variation* of the MSE between runs is an indication of the stability of the integrator. We plot graphs of both of these statistics in Figure 5. The thin line represents the average MSE over time, whereas the shaded area visualizes the standard deviation of the MSE across *all* 50 runs. In all test scenes, RJMLT has a lower average MSE than MMLT at equal render time, sometimes significantly. Additionally, RJMLT exhibits less variation between runs than MMLT, which suggests that our proposed perturbation is more effective at exploring potential light paths than MMLT small steps.

Acceptance Probabilities. We track the average proposal acceptance rates in the Living Room scene, and compare the results in Figure 4, broken down over path length. We differentiate between three different perturbations: MMLT small steps that leave the sampling technique unchanged, MMLT small steps that change the technique, and our proposed RJMLT perturbations. RJMLT small steps and MMLT small steps without technique change have identical behavior, and we only show the MMLT results. As the path length increases, the acceptance rates of small step proposals decreases rapidly. Additionally, MMLT proposals that change the technique index have a significantly lower acceptance probability, dropping as low as 1% at path length 10. In contrast, our proposed perturbations achieve average acceptance probabilities close to 100% across all path lengths. The slight drop for longer paths stems from perturbations that could not perfectly invert a path due to floating point inaccuracy, and were rejected by our algorithm.

Temporal Stability. A drawback of correlated sampling is that noise is spatially coherent. This may not be immediately apparent in still images, but can cause visually disturbing flickering artifacts in rendered animations. To provide a visual comparison of the temporal behavior of MMLT and RJMLT, we render 50 images for each scene and integrator with different random seeds. We compose the renderings into a video, and provide a web-based comparison tool in our supplemental material to compare the temporal videos side-by-side. In all scenes, RJMLT exhibits more stable and visually pleasing noise behavior across runs than MMLT, and in some scenes (e.g. Staircase, Jewelry, Bathroom, Kitchen) significantly so.

Equal-time renderings. Finally, we also provide renderings of RJMLT and MMLT after 5 minutes of render time in Figure 6. We show a subset of our scenes, and provide two insets for each scene highlighting interesting features. Because of the structured nature of MLT noise, it is difficult to judge renderings of the methods by their noise level alone. We have added arrows to the MMLT insets pointing out areas of significant error. These include e.g. streaks on the pots and cutting board in the Kitchen scene; streaky reflections and hard-edge shadows in Living Room; wrongly positioned or missing caustics in Bathroom; and significant splotches in Glass of Water. We include full-size renderings of all our methods in the supplemental material, along with a web-based image comparison tool. We also include a heatmap image of the MSE in each scene, averaged over 50 runs of the integrators.

8 CONCLUSION

We introduced Reversible Jump MCMC to the field of light transport, and presented a reformulation of Multiplexed MLT in this framework. Our analysis showed that technique changes in MMLT

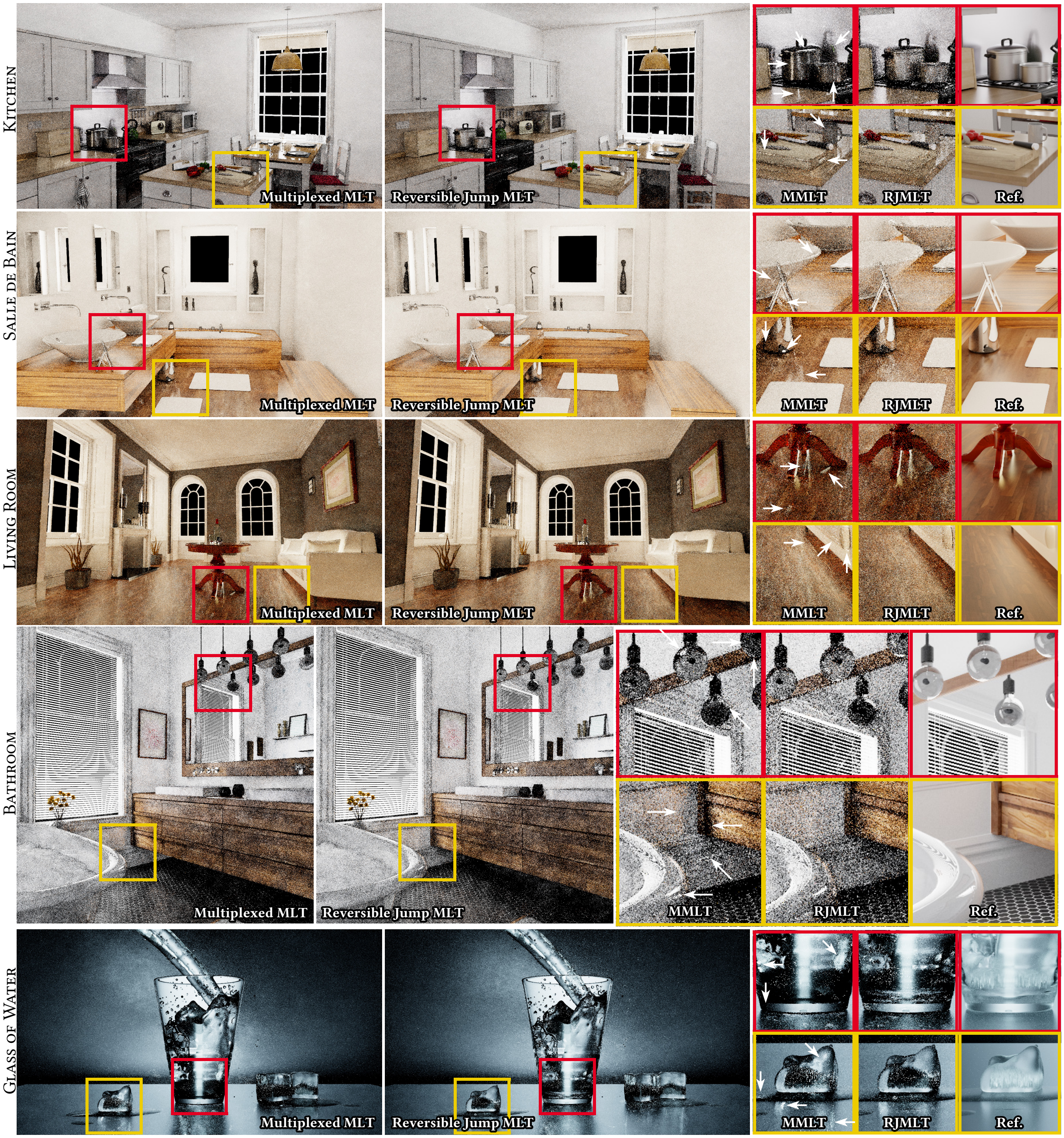


Fig. 6. We compare Multiplexed MLT (left) to Reversible Jump MLT (middle) across 5 different scenes at equal render time. The scenes feature complex illumination and occlusion, glossy caustics and long light path lengths. Because of its ability to transition between sampling techniques easily, RJMLT has fewer artifacts (streaks, splotches, wrongly positioned caustics) than Multiplexed MLT at equal render time. Such artifacts are symptomatic of the Markov Chain "getting stuck" and exploring a small part of path space for too long. Scenes were generously provided by the following blendswap.com artists: Jay-Artist (Kitchen), Mareck (Salle de Bain), Wig42 (Living Room), Nacimus (Bathroom) and Axel (Glass of Water).

amount to disruptive changes to the light path, and predicted difficulties in transitioning between sampling techniques. We introduced the concept of inverse sample functions and used them to constructed a novel perturbation, which we call *Reversible Jump MLT*, that allows the Markov Chain to easily transition between sampling techniques. We perform an in-depth mathematical analysis of this perturbation, and derive the correct acceptance probability using the RJMCMC framework. We describe how to construct probabilistic inverses, and reconcile them with the RJMCMC framework. Two implementations confirm the correctness of our method, and detailed metrics show improved temporal coherence, faster convergence and decrease in artifacts of our method compared to MMLT.

8.1 Limitations

We described general mathematical tools for handling non-invertible sampling methods and how to express these ideas in the RJMCMC framework. We found the techniques we described sufficient to handle nearly every sampling scheme we encountered in practice. However, a few notable exceptions may cause our perturbation to fail for some paths. While this does not affect the correctness of our method, it may slow convergence if such paths dominate transport.

Rejection sampling is an alternative recipe for deriving sampling algorithms for general target distributions. Sampling methods derived in this manner, such as Woodcock tracking [1965], consume a potentially unbounded set of random numbers and are virtually impossible to invert. However, the gains obtained from applying PSSMLT based methods on top of such a sampling scheme are questionable, and we believe this not to be a practical issue.

Certain specialized sampling schemes (e.g. the Box-Muller transform [1958]) produce more than one sample for one set of inputs. Surplus samples are usually cached and reused in future sample queries, introducing a dependence between paths. Given only a single sample, it is not clear how to invert such a scheme and how the correct acceptance ratio should be computed. Our current implementation will simply reject perturbations involving such schemes.

Finally, implementing path inverses requires additional engineering effort for every sampling method used in the underlying rendering algorithm. Although individual inverses are easily derived, the number of sampling techniques present in practical rendering systems can pose a challenge. This is an inevitability of moving beyond an exclusive primary sample space view.

8.2 Future Work

There are a number of interesting directions in which our work could be extended. A new class of perturbations is enabled by the ability to transition between path space and primary sample space, of which RJMLT is only one possibility. For example, the existence of inverses could allow path space methods to temporarily transition to primary sample space, perform a perturbation and transition back to path space. This reintroduces some of the benefits of PSSMLT, such as a better local parametrization of paths, to path space methods. Conversely, inverses could enable crafting path-space-style perturbations and mutations in a PSSMLT framework.

In addition to the theory presented here, Reversible Jump MCMC also provides a framework to reason about *dimension jumps*, i.e. transitions between subspaces of different dimensions akin to the bidirectional mutation of MLT by Veach and Guibas [1997]. There are a number of different ways in which this concept could be applied to light transport, e.g. a new perturbation that allows MMLT to transition between light paths of different length.

Finally, the detailed discussion of sample Jacobians presented in this paper could have applications outside of RJMLT acceptance probabilities. In a sense, the Jacobian determinant is a concise description of how sensitive the sampled path is to perturbations of the random numbers that sample it. This could lead to a novel adaptive step size control for primary sample space methods, which could *locally* estimate the correct step size to reduce rippling effects and achieve optimal mixing rates.

ACKNOWLEDGEMENTS

We thank the following blendswap.com artists for providing the scenes used in this paper: Jay-Artist (Kitchen), Nacimus (Bathroom), Wig42 (Living Room), Mareck (Salle de Bain), Axel (Glass of Water). This work was partially supported by a generous gift from Activision, as well as the National Science Foundation (Grant CNS-1205521).

REFERENCES

George E. P. Box and Mervin E. Muller. 1958. A Note on the Generation of Random Normal Deviates. *The Annals of Mathematical Statistics* 29, 2 (06 1958), 610–611. https://doi.org/10.1214/aoms/1177706645

David Cline, Justin Talbot, and Parris Egbert. 2005. Energy Redistribution Path Tracing. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 24, 3 (July 2005), 1186–1195. https://doi.org/10.1145/1073204.1073330

Peter J. Green. 1995. Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination. *Biometrika* 82 (1995), 711–732.

Adrien Gruson, Mickaël Ribardière, Martin Šik, Jiří Vorba, Rémi Cozot, Kadi Bouatouch, and Jaroslav Křivánek. 2017. A Spatial Target Function for Metropolis Photon Tracing. *ACM Trans. Graph.* 36, 1, Article 4 (Feb. 2017), 13 pages. https://doi.org/10.1145/2963097

Toshiya Hachisuka and Henrik Wann Jensen. 2011. Robust Adaptive Photon Tracing using Photon Path Visibility. *ACM Trans. Graph.* 30, 5, Article 114 (Oct. 2011), 11 pages. https://doi.org/10.1145/2019627.2019633

Toshiya Hachisuka, Anton S. Kaplanyan, and Carsten Dachsbacher. 2014. Multiplexed Metropolis Light Transport. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 33, 4, Article 100 (July 2014), 10 pages. https://doi.org/10.1145/2601097.2601138

Johannes Hanika, Anton Kaplanyan, and Carsten Dachsbacher. 2015. Improved Half Vector Space Light Transport. *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering)* 34, 4 (2015), 65–74. https://doi.org/10.1111/cgf.12679

Wilfred K. Hastings. 1970. Monte Carlo Sampling Methods using Markov Chains and their Applications. *Biometrika* 57, 1 (1970), 97–109. https://doi.org/10.1093/biomet/57.1.97

Jared Hoberock and John C. Hart. 2010. Arbitrary Importance Functions for Metropolis Light Transport.. In *Computer Graphics Forum.* 1993–2003. https://doi.org/10.1111/j.1467-8659.2010.01713.x

Wenzel Jakob. 2013. *Light Transport on Path-Space Manifolds.* Ph.D. Dissertation. Cornell University.

James T. Kajiya. 1986. The Rendering Equation. In *Computer Graphics (Proceedings of SIGGRAPH)*, Vol. 20. ACM, 143–150. https://doi.org/10.1145/15922.15902

Anton S. Kaplanyan, Johannes Hanika, and Carsten Dachsbacher. 2014. The Natural-Constraint Representation of the Path Space for Efficient Light Transport Simulation. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 33, 4, Article 102 (July 2014), 13 pages. https://doi.org/10.1145/2601097.2601108

Csaba Kelemen, László Szirmay-Kalos, György Antal, and Ferenc Csonka. 2002. A Simple and Robust Mutation Strategy for the Metropolis Light Transport Algorithm. *Computer Graphics Forum* 21, 3 (2002), 531–540. https://doi.org/10.1111/1467-8659.t01-1-00703

Shinya Kitaoka, Yoshifumi Kitamura, and Fumio Kishino. 2009. Replica Exchange Light Transport. *Computer Graphics Forum* 28, 8 (12 2009), 2330–2342. https://doi.org/10.1111/j.1467-8659.2009.01540.x

Eric P. Lafortune and Yves D. Willems. 1993. Bi-Directional Path Tracing. In *Proceedings of 3rd International Conference on Computational Graphics and Visualization Techniques (COMPUGRAPHICS '93.* 145–153.

Eric P. Lafortune and Yves D. Willems. 1996. Rendering Participating Media with Bidirectional Path Tracing. In *Rendering Techniques (Proceedings of the Eurographics Workshop on Rendering)*. Springer-Verlag, 91–100.

Jaakko Lehtinen, Tero Karras, Samuli Laine, Miika Aittala, Frédo Durand, and Timo Aila. 2013. Gradient-Domain Metropolis Light Transport. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 32, 4, Article 95 (July 2013), 12 pages. https://doi.org/10.1145/2461912.2461943

Tzu-Mao Li, Jaakko Lehtinen, Ravi Ramamoorthi, Wenzel Jakob, and Frédo Durand. 2015. Anisotropic Gaussian Mutations for Metropolis Light Transport Through Hessian-Hamiltonian Dynamics. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 34, 6, Article 209 (Oct. 2015), 13 pages. https://doi.org/10.1145/2816795.2818084

Hisanari Otsu, Anton S. Kaplanyan, Johannes Hanika, Carsten Dachsbacher, and Toshiya Hachisuka. 2017. Fusing State Spaces for Markov Chain Monte Carlo Rendering. *ACM Transactions on Graphics (Proc. of SIGGRAPH)* 36, 4, Article 74 (2017), 74:1–74:10 pages.

Jacopo Pantaleoni. 2017. Charted Metropolis Light Transport. *ACM Transactions on Graphics (Proc. of SIGGRAPH)* 36, 4, Article 75 (2017), 75:1–75:15 pages.

Martin Šik, Hisanari Otsu, Toshiya Hachisuka, and Jaroslav Křivánek. 2016. Robust Light Transport Simulation via Metropolised Bidirectional Estimators. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 35, 6, Article 245 (Nov. 2016), 12 pages. https://doi.org/10.1145/2980179.2982411

Eric Veach and Leonidas Guibas. 1994. Bidirectional Estimators for Light Transport. In *Photorealistic Rendering Techniques (Proceedings of the Eurographics Workshop on Rendering*. 147–162. https://doi.org/10.1007/978-3-642-87825-1_11

Eric Veach and Leonidas J. Guibas. 1995. Optimally Combining Sampling Techniques for Monte Carlo Rendering. In *Annual Conference Series on Computer Graphics (Proceedings of SIGGRAPH)*. ACM, New York, NY, USA, 419–428. https://doi.org/10.1145/218380.218498

Eric Veach and Leonidas J. Guibas. 1997. Metropolis Light Transport. In *Annual Conference Series on Computer Graphics (Proceedings of SIGGRAPH)*. ACM Press/Addison-Wesley Publishing Co., 65–76. https://doi.org/10.1145/258734.258775

E. Woodcock, T. Murphy, P. Hemmings, and T. Longworth. 1965. Techniques used in the GEM Code for Monte Carlo Neutronics Calculations in Reactors and other Systems of Complex Geometry. In *Proceedings of the Conference on Applications of Computing Methods to Reactor Problems*. 557–579.

Received April 2017; final version July 2017; accepted July 2017