Eurographics Symposium on Rendering - Experimental Ideas & Implementations (2018)
T. Hachisuka and W. Jakob (Editors)

Matrix Bidirectional Path Tracing

Chakravarty R. Alla Chaitanya[1,2] and Laurent Belcour[3] and Toshiya Hachisuka[4] and Simon Premoze[2] and Jacopo Pantaleoni[5] and Derek Nowrouzezahrai[1]

[1]McGill University, [2]University of Montreal, [3]Unity Technologies, [4]The University of Tokyo, [5]NVIDIA Research

Abstract

Sampled paths in Monte Carlo ray tracing can be arbitrarily close to each other due to its stochastic nature. Such clumped samples in the path space tend to contribute little toward an accurate estimate of each pixel. Bidirectional light transport methods make this issue further complicated since connecting paths of sampled subpaths can be arbitrarily clumped again. We propose a matrix formulation of bidirectional light transport that enables stratification (and low-discrepancy sampling) in this connection space. This stratification allows us to distribute computation evenly across contributing paths in the image, which is not possible with standard bidirectional or Markov chain solutions. Each element in our matrix formulation represents a pair of connected eye- and light-subpaths. By carefully reordering these elements, we build a 2D space where equally contributing paths are distributed coherently. We devise an unbiased rendering algorithm that leverages this coherence to effectively sample path space, consistently achieving a $2-3\times$ speedup in radiometrically complex scenes compared to the state-of-the-art.

CCS Concepts

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

1. Introduction

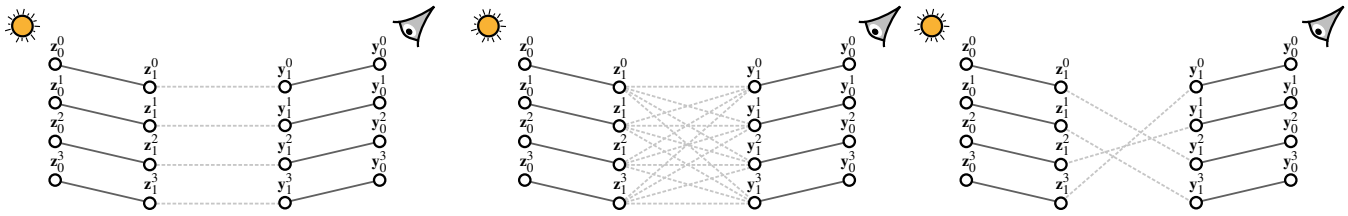
Realistic image synthesis relies on time-consuming physically-accurate light transport simulations. Light transport in scenes where only a small subset of emitters contribute energy to the image sensor, and only after undergoing many scattering events, are particularly challenging to simulate. Bidirectional path tracing (BDPT) [VG94, LW93] aims to address inefficiencies of sampling such complex paths by sampling subpaths between the eye and the light, forming all possible pairs of *complete light transport paths* by connecting vertices along the two subpaths. Each such connection can correspond to a separate *sampling strategy*, and the hope of BDPT is to more thoroughly explore the space of all possible energy-carrying paths (called *path space*) by sampling them in such varied ways. Metropolis Light Transport [VG97, KSKAC02] (MLT) further aims to more thoroughly explore high-energy regions in the path space by mutating "promising" paths (i.e., those that contribute non-negligible energy to the sensor).

BDPT samples eye- and light-subpaths independently for each pixel of the output image and does not exploit the benefits of stratified sampling as the connection between independent subpath vertices do not have any spatial relationship between them. This is even more so true in large, realistic scenes with many lights, often only a small subset of the light sources contribute to the final image. Here, both BDPT and MLT can have poor performance: for BDPT, significant time is wasted constructing subpath pairs that ultimately contribute no energy to the image and it is possible that similar subpaths are sampled again within other pixels with low contributions. While a combinatorial set of connections across an entire image can lead to many more contributing paths, the quadratic cost of computing it prevents its practical application. On the other hand, MLT-based approaches exhibit non-uniform convergence over the image, which is undesirable in practice since some light paths might not be sampled at all even after significant computation times. This makes it difficult for a user to assess the progress of an MLT simulation and, namely, whether relevant transport contributions for an image have all yet to be sampled.

We introduce *Matrix BDPT* (M-BDPT), a new path space sampling strategy that allows us to stratify connected paths in bidirectional approaches, while avoiding the quadratic cost of exhaustively checking combinatorial connections of sampled subpaths. To do so, we reorder potential subpath vertex connections into a set of transport matrices in a manner that allows us to exploit coherence in the path space. By introducing such coherence, we can perform stratified sampling in a set of *2D connection subspaces*, significantly reducing the Monte Carlo variance. M-BDPT achieves a 2-to-3$\times$ speedup for equal quality rendering compared to BDPT, and often even higher speedups with more uniform convergence compared to MLT, in radiometrically challenging scenes where only a few long-chain paths contribute non-negligible energy to the image.

We present the following contributions:

- A matrix formulation of eye- and light-subpath connections that allows us to exploit coherence in light transport,
- A stratification scheme tailored to connection space, based on a spatial reordering of the eye- and light-subpath vertices, and
- A new global sampling technique that forms connections between large families of subpath vertices.



a) BDPT b) Combinatorial BDPT [PBPP11] c) Matrix-BDPT (Ours)

Figure 1: *We illustrate how subpath connections are formed across pixels for an example 4-vertex path and a single sampling technique (i.e., $\{s,t\} = \{1,1\}$) in: (a) standard BDPT, (b) Combinatorial BDPT [PBPP11], and (c) M-BDPT. BDPT combines subpath pairs independently at each pixel, using the same connection across (s,t) strategies. Combinatorial BDPT performs a costly brute-force connection between all pairs of subpath vertices across every light-subpath and every (pixel's) eye-subpath. Our M-BDPT methods uses a specialized stratified sampling to selectively connect eye- and light-subpaths across the image, only forming a number of connections proportional to the number of eye-paths (i.e., the same sampling rate as standard BDPT). Here, we visualize only a single connection of the all-pairs connections that BDPT makes, whereas Figure 2 shows the all-pairs connections for a single eye- and light-subpath for BDPT.*

2. Related Work

Physically-based light transport is a mature research area, and we highlight only the most pertinent related works below: approaches that build atop BDPT to address issues concerning path formation and connection, light-path reuse, and transport clustering. Quasi-Monte Carlo methods [Kel95] generate samples which are not uniformly distributed to avoid sample clumping. These samples can be generated using a low-discrepancy sequence such as the Halton sequence, the Sobol sequence, and the Faure sequence. As the eye- and light-subpaths are sampled independently in BDPT, even Quasi-Monte Carlo sampling for generating the individual subpaths cannot achieve good spatial stratification in the path space.

Connecting paths. Pajot et al. [PBPP11] suggested a combinatorial connection scheme of *all pairs* of eye- and light-subpath vertices *across an entire image*, as opposed to only connecting pairs of vertices through a single pixel (as in traditional BDPT; see Figure 1). By optimizing connection computation on the GPU, they improve upon traditional BDPT, but the complexity of considering all such combinatorial connections can lead to scalability issues in radiometrically complex scenes. We avoid this quadratic complexity by subsampling connections, selecting only a sparse (and carefully reordered) subset of all possible connections using a stratified low-discrepancy scheme.

As with our work, bidirectional lightcuts [WKB12] reduce the number of bidirectional connections by prioritizing connections *at the endpoints* of subpaths using a spatial data structure over path vertices. By computing conservative throughput bounds at terminal vertices, they choose which subpaths to connect. This approach clearly demonstrates the potential benefits of selective bidirectional connection, however the manner in which connections are chosen results in a (controllably) biased estimator. We, instead, reorder subpath connections and leverage the resulting coherence with stratified sampling, resulting in a new unbiased estimator that benefits from the advantages of connection subsampling.

Reusing paths. Bekaert et al. [BSH02] proposed combining paths traced through nearby pixels. Each such path is designed to contribute to surrounding pixels in a local neighborhood in a provably effective manner. This approach, while only originally proposed for unidirectional path tracing techniques, yields an order of magnitude improvement for indirect illumination in scenes of low-to-moderate radiometric complexity. One could imagine extending this local reuse policy to a bidirectional estimator, however here correlation artifacts can become a serious concern.

Popov et al. [PRDD15] construct and cache importance functions in connection space for certain path subsets, which they sample for nearby eye paths. An important drawback they discuss is that the reuse of subpaths can introduce correlation. Inspired by this work, we opt to avoid caching importance and instead build a matrix formulation of subpath connection spaces that allows us to benefit directly from stratified, low-discrepancy sampling. As such, we do not rely on specialized MIS schemes. Most bidirectional techniques that share/reuse subpaths components exhibit some form of correlation, e.g., photon mapping and many light techniques [Kel97, DKH*14].

In independent work, Pantaleoni [Pan12] reformulated BDPT in a matrix form and proposed a low-discrepancy sampling approach based on reordered local throughput contributions. For each light or eye vertex, a connection vertex is selected from their matrix using a k-nearest neighbors search according to throughput. This reordering algorithm is GPU-accelerated and results in a 30% percent reduction in variance. We instead introduce a reordered matrix formulation that promotes path throughput coherence; this allows us to perform stratified sampling in the path space, reducing variance without necessitating an k-nearest neighbor searches.

Clustering paths and virtual light methods. Hašan et al. [HPB07] framed VPL connection subsampling as a matrix completion problem, and they approximate the sum of VPLs contributions to pixels by selectively sampling rows and columns. They are also able to take advantage of hardware-accelerated rasterization to populate the matrix rows and columns and for many scenes, their matrix is amenable to low rank approximation. Ou and Pellacini [OP11] further extended Hašan et al.'s matrix approximation by clustering similar surface samples in order to further reduce the rank of the VPL transfer matrix: by operating at two scales, Ou and Pellacini first capture the coarse structure of the matrix by globally clustering lights and only afterwards do they selectively refine these clusters to choose the most important light vertices to connect to a given view sample. Our matrix formulation of BDPT is inspired

by these aforementioned matrix formulations of the VPL connection problem. Our formulation can be thought as a generalization of the existing formulations since VPL connection is a subset of all the possible BDPT connections. Similar to VPL methods, we reformulated BDPT to connect a single light-subpath with *multiple* eye-subpaths. Instead of clustering VPLs, we use this matrix formulation to achieve efficient stratification of BDPT paths.

3. Background

We review the path integral formulation for (surface-only) light transport (Section 3.1) and the BDPT integral estimator (Section 3.2), before presenting our M-BDPT estimator (Section 4).

3.1. Path Integral Formulation

We adopt the path integral formulation of light transport [Vea97] as the basis for our matrix formulation. The observed radiance at pixel j is expressed as the integral of the throughput f_j of every possible light transport path that connects points on a light source to points on the j^{th} pixel sensor element

$$I_j = \int_\Omega f_j(\bar{\mathbf{x}})\, d\mu(\bar{\mathbf{x}}), \tag{1}$$

where Ω is the set of all paths $\Omega = \cup_{k=1}^{\infty}\Omega_k$ of all lengths k. A path is expressed as vector $\bar{\mathbf{x}} = [\mathbf{x}_0, \ldots, \mathbf{x}_k]$ of 3D surface points in an environment, and μ is the area-product measure over this space of paths $d\mu(\mathbf{x}) = \prod_{i=0}^{k} dA(\mathbf{x}_i) = dA(\mathbf{x}_i)\cdots dA(\mathbf{x}_k)$. The throughput of a path $\bar{\mathbf{x}}$ is simply the geometry-weighted product of the bidirectional scattering reflectance distribution functions (BSDFs) at each interior path vertices $\{\mathbf{x}_1 \cdots \mathbf{x}_{k-1}\}$, the emission at the light source vertex $\mathbf{x}_k$, the sensor importance at $\mathbf{x}_0$:

$$\begin{aligned} f_j(\bar{\mathbf{x}}) =& L_e(\mathbf{x}_k \to \mathbf{x}_{k-1}) G(\mathbf{x}_k \leftrightarrow \mathbf{x}_{k-1}) W_e(\mathbf{x}_1 \to \mathbf{x}_0) \\ & \prod_{i=1}^{k-1} f_s(\mathbf{x}_{i+1} \to \mathbf{x}_i \to \mathbf{x}_{i-1}) G(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i-1}), \end{aligned} \tag{2}$$

where L_e is the emitted radiance (non-zero iff $\mathbf{x}_k$ is on an emitter), G the geometric term, f_s the BSDF, and W_e the sensor importance.

3.2. Bidirectional Path Tracing

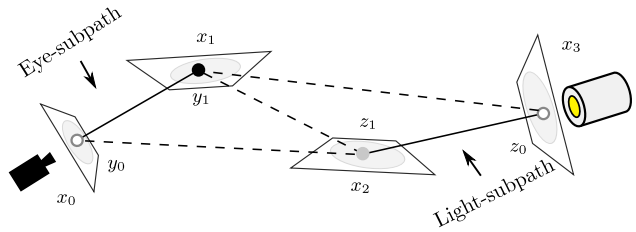


Figure 2: *In BDPT, light paths are formed by connecting (dotted lines) combinations of eye- and light-subpath (solid lines) vertices.*

Bidirectional path tracing is a Monte Carlo integral estimator of Equation 1 that draws path space samples according to a *pdf* that is constructed "bidirectionally": independent (and incomplete) subpaths are traced through the scene from the eye and the light, and these subpaths are constructed by sequentially importance sampling local scattering and/or emission distribution functions. In order to form a full path integration sample, the two independent eye- and light-subpaths are connected together.

The traditional BDPT algorithm averages the contribution of many full light transport path samples, each of which is constructed by tracing a single length-s eye- and length-t light-subpath per pixel. Full paths are formed by connecting *all pairs* of the subpath vertices (as opposed to simply connecting the subpath endpoints; see Figure 2). As such, a single eye- and light-subpath results in *many* full path space samples, and the process is repeated (per pixel) in order to increase the total sampling rate.

An important detail of this BDPT algorithm lies in the manner in which it forms and weighs subpath vertex connections. Specifically, consider a length-$k = (s+t+1)$ full transport path generated from one of the all-pair eye- and light-subpath vertex connections in Figure 2. There exists $s+t+2$ different strategies capable of sampling the length-k path. The traditional BDPT algorithm weighs each contribution using multiple importance sampling (MIS) according to these additional strategies [VG94, LW93], generally reducing the variance of the estimator. The final MIS expression of the traditional BDPT estimator

$$I_j = E\left[\sum_{s\geq 0}\sum_{t\geq 0} w(\bar{\mathbf{x}}_{s,t}) \frac{f_j(\bar{\mathbf{x}}_{s,t})}{p(\bar{\mathbf{x}}_{s,t})}\right], \tag{3}$$

where $\bar{\mathbf{x}}_{s,t} = [\mathbf{y}_0, \ldots, \mathbf{y}_s, \mathbf{z}_t, \ldots, \mathbf{z}_0]$ is a full light path through pixel i, $[\mathbf{y}_0, \ldots, \mathbf{y}_s]$ is the eye-subpath, $[\mathbf{z}_0, \ldots, \mathbf{z}_t]$ is the light-subpath (Figure 2), $p(\bar{\mathbf{x}}_{s,t})$ is the probability density function used to generate the eye-subpath $\bar{\mathbf{y}}$ and the light-subpath $\bar{\mathbf{z}}$, and $w(\bar{\mathbf{x}}_{s,t})$ is the MIS weight. The full BDPT algorithm averages many independent evaluations of Equation 3 to obtain a result with a higher sampling rate.

Problem Statement It is important to reiterate that BDPT only samples and connects eye- and light-subpath vertices *within* a pixel, and not across pixels. Even if we know that a particular set of subpaths within one pixel has zero or low contributions after connection, arbitrarily similar set of subpaths can be sampled again within other pixels and causes low-contributing paths. We thus propose a procedure for selecting candidate eye and light-subpath vertices across the image. This allows us to stratify connections across the image and avoids the same low-contributing paths to happen repeatedly across different pixels. To do so, we present a matrix reformulation of BDPT that introduces coherence in potential throughput connections by efficiently reordering subpath vertices across an image. This coherence, in turn, allows us to perform stratified sampling in the subpath connection space and reduce the variance of the full bidirectional path samples compared to the traditional BDPT (or the combinatorial BDPT) methods.

4. Matrix Bidirectional Path Tracing

Unlike BDPT, which starts by generating an eye- and light-subpath through a single pixel, we first generate N eye- and L light-subpaths over the image. We treat fixed-length eye- and light-subpaths from this set separately (up to the maximum possible length, computed as the sum of the longest eye- and light-subpaths we sampled, $s < s_{\max}$ and $t < t_{\max}$.), modeling their unshadowed throughput connection in matrices $M_{s,t}$.

Matrix rows in $M_{s,t}$ index a single length-s eye-subpath, and matrix columns index a single length-t light-subpath, from the set of subpaths we initially sampled (see Figure 3). Each matrix entry stores the unshadowed full path throughput estimate which is trivial to compute as it does not require any ray-tracing and can use information built up incrementally during subpath construction and storage. The contents of the matrix are never directly used in the rendering. When sampling matrix elements, we pick an eye- and a light-subpath vertex and connect them together. Only after sampling, we compute the full path throughput estimate for this connection. We discuss this *matrix construction process* in Section 4.1.

After populating every matrix $M_{s,t}$, for $0 \leq s \leq s_{\max}$ and $0 \leq t \leq t_{\max}$, we apply an efficient reordering scheme to introduce coherence in these throughput matrices (see Section 4.2) which, in turn, allows us to sample full transport paths using stratified (or low-discrepancy) samples (see Section 4.3).

4.1. Matrix Construction

Our M-BDPT algorithm treats eye- and light-subpaths, sampled over the entire image, separately according to their lengths (or, equivalently, their connection strategy (s,t)). Here, the contribution over the entire image from eye-subpaths of length-s and light-subpaths of length-t can be expressed as a Monte Carlo estimate with MIS as

$$I_j^{s,t} = E\left[\sum_{(p,q)\in\mathcal{S}} w\left([\bar{\mathbf{y}}_s^p, \bar{\mathbf{z}}_t^q]\right) \frac{f_j\left([\bar{\mathbf{y}}_s^p, \bar{\mathbf{z}}_t^q]\right)}{p\left([\bar{\mathbf{y}}_s^p, \bar{\mathbf{z}}_t^q]\right)}\right], \tag{4}$$

where p and q index the length-s eye-subpaths $\bar{\mathbf{y}}_s^p = [y_0^p, \cdots, y_s^p]$ and length-t light-subpaths $\bar{\mathbf{z}}_t^q = [\mathbf{z}_t^q, \cdots, \mathbf{z}_0^q]$, $\mathcal{S}$ is the set of all such subpaths, and $p\left([\bar{\mathbf{y}}_s^p, \bar{\mathbf{z}}_t^q]\right)$ is the sampling probability used to generate the full connected transport path.

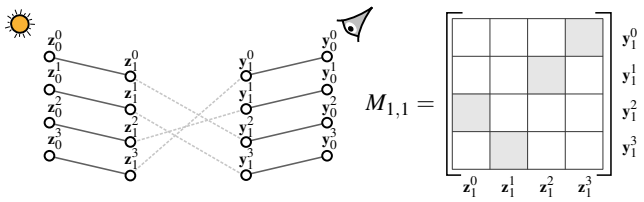


Figure 3: *Let $M_{s,t}$ denote the connection matrix for a given (s,t) configuration (e.g., here $(s,t) = (1,1)$). Entries $M_{s,t}^{i,j}$ correspond to the connection throughput divided by its sampling probability.*

It is important to distinguish between the quantities in Equations 3 and 4. In the traditional BDPT, we generate an eye- and a light-subpath and consider all-pairs connections between them, and compute their contribution towards an image pixel. Equation 3 evaluates the path integral of all possible path lengths (i.e., full global illumination) using a single pair of an eye- and a light-subpaths. In M-BDPT, we fix the path length and estimate the contribution from different eye- and light-subpaths towards the entire image. Equation 4 estimates the path integral for a *single fixed path length* (e.g., just 1-bounce of indirect illumination) using a collection of pairs of an eye- and a light-subpath.

Using this formulation, instead of evaluating all the (s,t) pairs for a couple of eye and light paths, we can quasi-randomly select pairs (p,q) for each (s,t) pair. Hence, if the contribution of a light path $\bar{\mathbf{z}}^q$ is zero, that is $f_j([\bar{\mathbf{y}}_s^p, \bar{\mathbf{z}}_t^q]) = 0\ \forall s,t,q$, then this null contribution is not evaluated $(s+1)(t+1)$ times.

This formulation is equivalent to formulating the matrix of all eye- and light-subpath pairs for a given (s,t) couple (see Figure 3). We denote this matrix $M_{s,t}$ as

$$M_{s,t}^{p,q} = w\left([\bar{\mathbf{y}}_s^p, \bar{\mathbf{z}}_t^q]\right) \frac{f_j\left([\bar{\mathbf{y}}_s^p, \bar{\mathbf{z}}_t^q]\right)}{p\left([\bar{\mathbf{y}}_s^p, \bar{\mathbf{z}}_t^q]\right)}. \tag{5}$$

For example, performing a combinatorial evaluation of BDPT, as in [PBPP11], corresponds to

$$I_j^{s,t} = \mathbf{1}_N^T M_{s,t}\, \mathbf{1}_M \big/ (N \times M) \tag{6}$$

where $\mathbf{1_N}$ is a size-N vector filled with ones. The traditional BDPT algorithm can also be expressed as summing the diagonal elements of this matrix;

$$I_j^{s,t} = \sum \mathrm{diag}(M_{s,t}) \big/ N. \tag{7}$$

Here, we require a square matrix since there are an equal number of eye- and light-subpaths.

Storage We only organize subpath connections into matrices as an expositional tool; in practice, we never explicitly form nor operate on any matrices. The matrices provide a concrete two-dimensional representation of the connection between eye- and light-subpaths. We store a set of eye- and light-subpaths and the corresponding indices for reordering the matrix. While these matrices are dense, we only sparsely subsample them and do not evaluate an entire matrix.

4.2. Matrix Subpath Reordering

In order to benefit from the relative stratification of a low-discrepancy sequence, we need to reorder our matrix to ensure local coherency. The use of a low-discrepancy sequence is beneficial when the integrand is continuous and slowly varying. However, without reordering, the connection matrix has purely random indexing (see Figure 4). To provide a better continuity in the matrix,

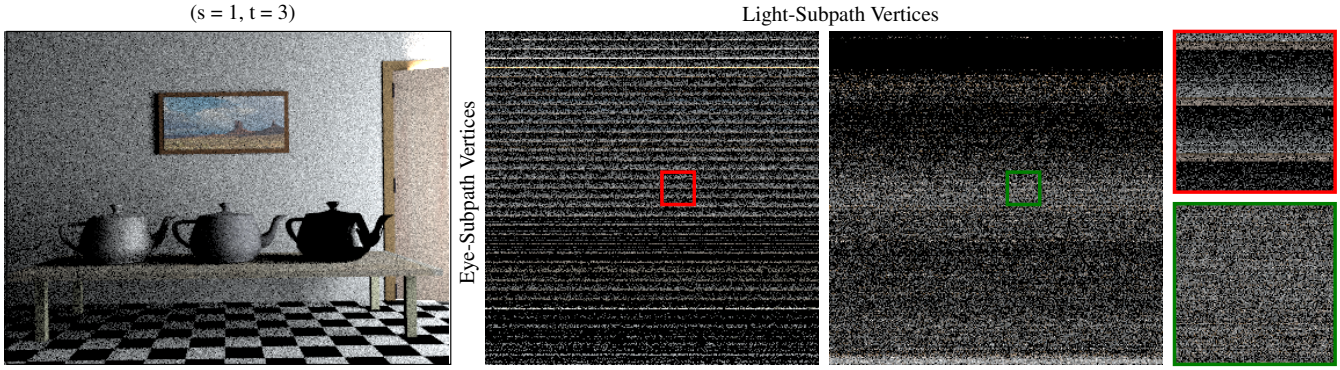


Figure 4: *Veach door scene rendered using technique $(s,t) = (1,3)$. We visualize the matrix structure that arises through unordered and ordered approaches. Ordered approach introduces coherence by accounting for spatial proximity between eye- and light-subpaths (inset).*

we reorder the rows and columns using a spatial reordering based on the distance between vertices. We compute the $L2$ norm for each eye- and light-subpath vertices within each submatrix (s,t). For an eye-subpath $\bar{\mathbf{y}}$, the distance metric $d(\bar{\mathbf{y}})$ is

$$d(\bar{\mathbf{y}}) = \sum_{k=0}^{s-1} ||\mathbf{y}_k - \mathbf{y}_{k+1}||^2 \text{ and } d(\bar{\mathbf{z}}) = \sum_{k=0}^{t-1} ||\mathbf{z}_k - \mathbf{z}_{k+1}||^2,$$

where $d(\bar{\mathbf{z}})$ is the distance metric for a light-subpath $\bar{\mathbf{z}}$. Note that this reordering does not ensure that vertices (from different subpaths) close to one another will correspond to close indices in the matrix. It provides however, a distinction between lengthy paths and short paths. The connection throughput then should be relatively coherent between indices. We sort the eye- and light-subpath vertices independently within each submatrix (s,t) based on the above metric.

We experienced alternative metrics such as an Hilbert curve coding distance (evaluating the position of the last vertex of a path on the 1D Hilbert curve). We found no major difference in variance reduction using this metric. However the cost of evaluation was much higher and thus we opted for the faster metric. Note that while our matrix formulation is similar to the one of Pantaleoni [Pan12], our metric is much easier to implement and faster to evaluate. Pantaleoni used the throughput of close vertices to reorder the matrix. This technique requires to evaluate a k-nearest search for each vertex.

Paths with similar distances, according to our metric, tend to have similar throughputs, given their geometric configuration. The main idea is to place similar paths in neighboring matrix elements and to cluster them together with a similar geometric throughput (i.e., the product of all subpath vertex G terms); we only consider spatial distances, and not orientation distances. This approximates the geometric throughput. Our distance metric is effective in higher dimensional spaces, as we consider distances from the first subpath vertex to the current subpath vertex (endpoint). As a result, we achieve good spatial stratification in the connection space. For Morton or Hilbert codes, the subpath vertex endpoints are considered alone when representing them in this higher dimensional space.

4.3. Sampling the Matrix Entries

The next step is to build the indices (p,q) from a set of eye- and light-subpaths and select the list of pairs $\mathcal{S}$. We select entries in the matrix using a two dimensional low-discrepancy sampler. Given that we want to evaluate K entries in the matrix, we generate K two-dimensional points $\varepsilon \in [0,1]^2$, scale them using the matrix size and use the integer part as the index $(p,q) = (\lfloor N\varepsilon_1 \rfloor, \lfloor L\varepsilon_2 \rfloor)$.

Matrix Structure Figure 4 visualizes the matrix structure of two different approaches: unordered and ordered. We render the door scene using the sampling strategy $(s,t) = (1,3)$. Unordered approach connects each eye-subpath vertex to all light-subpath vertices and computes their contribution for the corresponding pixel of the output image as shown in Figure 4. In this example, there are $1,728$ eye- and $1,728$ light-subpath vertices for the given strategy technique. Ordered approach reorders eye- and light-subpath vertices and connects them, and computes their contribution similar to the unordered approach. We can observe that the matrix image using the ordered version is a lot smoother when compared to the unordered approach. This could be attributed to the fact that we reorder subpaths; thus, achieving spatial coherence of the resulting transport matrix.

In case of the traditional BDPT algorithm, we select diagonal entries of the unordered matrix and connect the corresponding eye- and light-subpath vertex using Equation 7. This results in a very sparse matrix since the diagonal entries alone are considered. It is computationally wasteful since our visualization clearly shows that there are many contributing paths outside the diagonal entries. In contrast, the low-discrepancy sampling in M-BDPT helps us pick samples that are spatially coherent since the eye- and light-subpath vertices are ordered in the matrix; thereby, exploiting the coherence of the light transport field.

Our matrix formulation decouples vertex connections from the usual subpath pairs of classical BDPT formulation in a much more general form. This allows us to evaluate m connections per vertex, each with a connection to a different light subpath. Our formulation can be thought of as a stochastic sum (including additional visibility connections) over the entries of the M matrix, or as an integral over path connection space. Stratified sampling can lead to sig-

nificant variance reduction in these integral estimates. Contrarily, BDPT evaluates the path integral by independently connecting generated eye- and light-subpaths. Since the eye- and light-subpaths are sampled independently, the endpoints of the two subpaths do not have any spatial relationship and cannot achieve good spatial stratification in the connection space. Stratified or low-discrepancy sampling achieves superior variance reduction when the integrand is smooth with respect to its parameterization. The unordered connection matrix is not a proper space for such stratified sampling techniques as it lacks coherence or regularity. The manner in which we reorder the matrix rows and columns induces a new parameterization of the connection space that is more suitable for the application of stratified or low-discrepancy sampling.

5. Implementation

We implement our M-BDPT algorithm atop the PBRT *v3* framework [PJH16] and we set N and L equal to the number of image pixels. We also rendered our scenes using PBRT's BDPT and MLT implementations, the latter of which is based on multiplexed MLT [HKD14]. Ground truth images are rendered using BDPT, usually with multi-day rendering times.

We parallelize M-BDPT using a different, less efficient strategy compared to standard BDPT parallelization strategies. BDPT samples eye- and light-subpaths independently for each pixel of the output image. The output image is divided into tiles and each one of these tiles is processed by a separate core. M-BDPT generates multiple eye- and multiple light-subpaths for each iteration per pixel, so the tiling mechanism would not work in this case. There are three distinct stages in our algorithm: construction, reordering, and sampling.

We use Intel's OpenMP runtime libraries to parallelize each module separately. We first generate subpaths as described above. We next reorder the subpaths and then sample the matrix entries using a Latin hypercube sampling technique to compute the pixel contribution. We use the Standard Template Library (STL) sorting algorithm to sort the eye- and light-subpath vertices based on our distance metric. We constitute matrices depending on the sampling strategy (s,t) and each matrix computes radiance estimate independently. Thus, each matrix is handled by a separate core similar to the tiling mechanism used in the BDPT algorithm. In this section, we detail our M-BDPT algorithm. We highlight its behavior, in pseudo code, in Algorithm 1.

We use all eye-subpaths in the matrix, so that we consider the contribution towards each pixel for each corresponding (s, t)-connection. We do use reordered eye-subpaths in our algorithm: the motivation for this is that similar eye-subpaths will lead to similar importance transport on the light-subpaths. Without eye-subpath reordering, connected light-subpaths will not be stratified across different eye-subpaths, which can lead to additional correlation.

5.1. Selective Reordering

We implemented a brute-force sampling using our M-BDPT algorithm. In this approach, we connect each eye-subpath vertex to all light-subpath vertices within the submatrix (s,t). This is a combinatorial BDPT since it performs every connection for a given eye-subpath vertex. We use this as a baseline to illustrate the potential gains of our approach. For implicit path sampling techniques (path tracing without next-event estimation and light tracing), reordering does not provide any benefit as shown in Figure 5 (path tracing has been omitted from the figure for brevity). In such cases, we use the traditional BDPT algorithm's approach by picking the diagonal entries of the matrix using Equation 7. We choose either BDPT or M-BDPT approach depending on the sampling strategy technique.

Algorithm 1 Matrix Bidirectional Path Tracing

```
// Matrix formulation
Generate eye- and light-subpaths
Arrange the subpaths in terms of matrices

// Reorder the subpaths
Compute L2 norm for each eye- and light-subpath vertex
for t = 1 to k do
    for s = 0 to k do
        // For each submatrix (s, t)
        Reorder the subpaths based on their L2 norm

// Sample the matrix entries
for t = 1 to k do
    for s = 0 to k do
        // For each submatrix (s, t)
        for p = 0 to N do
            // For each eye-subpath
            Latin hypercube sample m light-subpaths
            Compute the pixel contribution by connecting the
            eye-subpath to m light-subpath vertices
```

6. Results

Figure 6 shows the split screen results for our modified Veach's door scene, where we add adjacent rooms with non-contributing (w.r.t. our view) emitters. We compare M-BDPT to BDPT for the same pixel sampling rate. BDPT results are noisy, as it cannot efficiently find contributing paths in this scene. In BDPT, we connect all vertices of eye- and light-subpaths even if they are not successful. This example uses a modified door scene with unreachable lights. Since we do not try to connect every vertex of a path in M-BDPT, the probability of connecting a reachable source is higher for a set of all eye-subpath vertices.

Figure 7 compares root mean square error (RMSE) images of M-BDPT and MLT in the door scene. Here, we compare BDPT, M-BDPT, and MLT at equal rendering time. MLT is slow to converge as its mutation strategies cannot capture light transport paths that contribute to the image, and it often spends significant time exploring low-contribution regions of the path-space integrand. On the other hand, M-BDPT is able to find more contributing paths as its connections vary across (s,t) strategies. We highlight a small distinction here in terms of comparisons in Figure 7: all the images used are for equal time comparisons, except the left most M-BDPT image. We reemphasize that M-BDPT converges faster

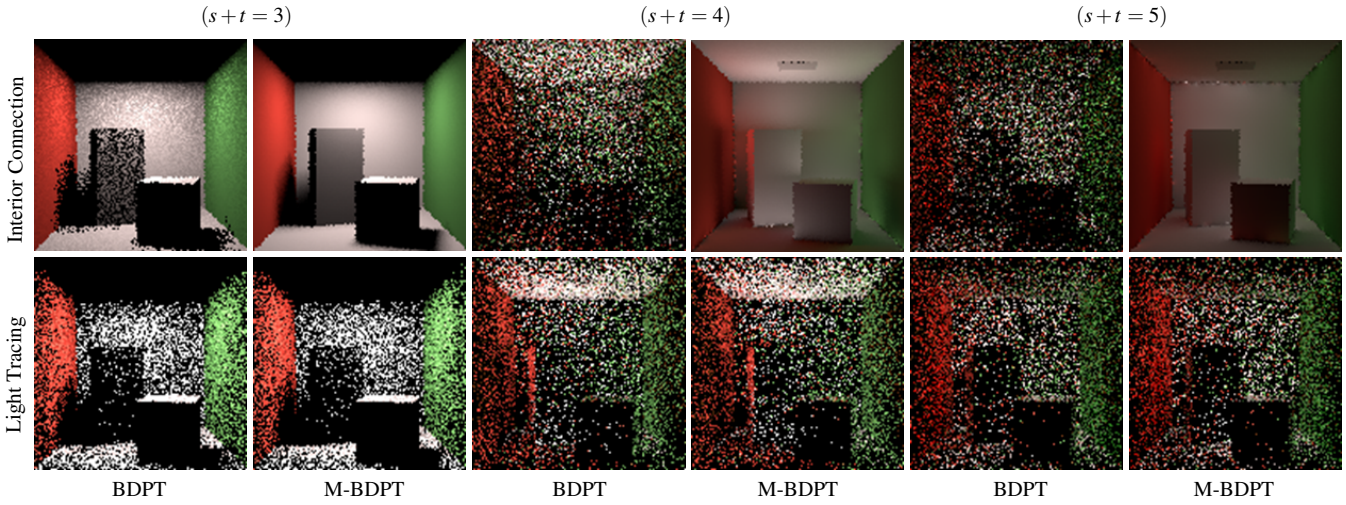


Figure 5: *Cornell box scene rendered with BDPT and M-BDPT using different path sampling techniques. Top row: contributions from interior connections* $(s \neq 0, t \neq 1)$. *Bottow row: contributions from light tracing* $(t = 1)$. *M-BDPT does not perform better than BDPT for light tracing but it significantly outperforms BDPT otherwise.*

than MLT from this time point onwards. We observe a similar performance/accuracy relationship for other scenes, discussed below. Figures 7 and 10 show two datapoints for M-BDPT in order to highlight the progression of convergence rate in comparison with MLT.

We modeled a stacked house scene (Figure 8) with eleven buildings, each of which has three emitters. While such a configuration is typical, it is also troublesome in practice since only a few light sources contribute to the image. In order to reduce the inefficiencies of existing rendering algorithms, artists often carefully remove invisible light sources using a manual, trial-and-error methodology. Without any such manual pre-process, we rendered a view into one of the rooms using BDPT and M-BDPT, as depicted in Figure 9. This room is a part of the building shown in Figure 8. Figure 10 also shows equal-time comparisons for the living room scene. Our algorithm works well in scenes with both diffuse and glossy surfaces.

7. Discussions and Limitations

Our matrix formulation separately handles subpaths with different lengths, with submatrices for each length. This separation does not exist in the single matrix formulation such as lightcuts [HPB07, OP11]. While our method can alternatively utilize a single matrix for all path lengths, we empirically found no gain in doing so. Instead, using submatrices, we can allocate the same number of samples per submatrix, which removes a part of the noise caused by stochastic sampling of subpath length. This stratification over subpath lengths is beneficial when indirect illumination dominates and subpaths of "sufficient" length are hard to generate stochastically.

Figure 11 shows the results for Veach's egg scene. We compare M-BDPT to BDPT and MLT using equal sampling rates *and* equal time comparison renderings. In this case, MLT does not outperform BDPT, which could be attributed to the additional overhead of finding suitable connections for this scene. M-BDPT is on par with BDPT, even though connections between subpaths succeed often with BDPT in this scene.

We note that our implementation of M-BDPT is not fully optimized as a part of our implementation is not suited for efficient multithreading. Since we generate subpaths once for each iteration per pixel, we need to reuse a set of light-subpaths while connecting them with eye-subpaths. This reuse prevents us from using a tile-based acceleration, since light-subpaths are not associated with a subset of pixels.

Figure 12 shows the equal time comparison results for the bathroom scene rendered with BDPT and M-BDPT. In this experiment, M-BDPT reuses light-subpaths by connecting one eye-subpath vertex to m light-subpath vertices where $m = 2, 4, 8$. This results in correlation artefacts as seen from the increasing RMSE errors in Figure 12. We do not perform a full connection for the m light-subpaths to the eye-subpath but only pick some of them. In such a case, it can be detrimental if some paths have multiple vertices with a high contribution. We have m long light-subpaths with equal contribution from all vertices but the contribution varies a lot between two different light-subpaths. If we use M-BDPT with $m > 1$ as shown in Figure 12, we will increase variance compared to BDPT since we are not reusing all the vertices of a given light-subpath.

8. Conclusions

We introduced a novel bidirectional light transport sampling method that reorders eye- and light-subpaths using a new matrix formulation. Our formulation introduces and exploits coherence in the transport connection matrix. By reordering subpaths, we can use stratified sampling in connection space, leveraging spatial coherence between subpath vertices. We integrated our sampling method into a new unbiased rendering algorithm, matrix bidirectional path tracing, which achieves significant variance reduction compared to bidirectional path tracing and metropolis light transport algorithms.

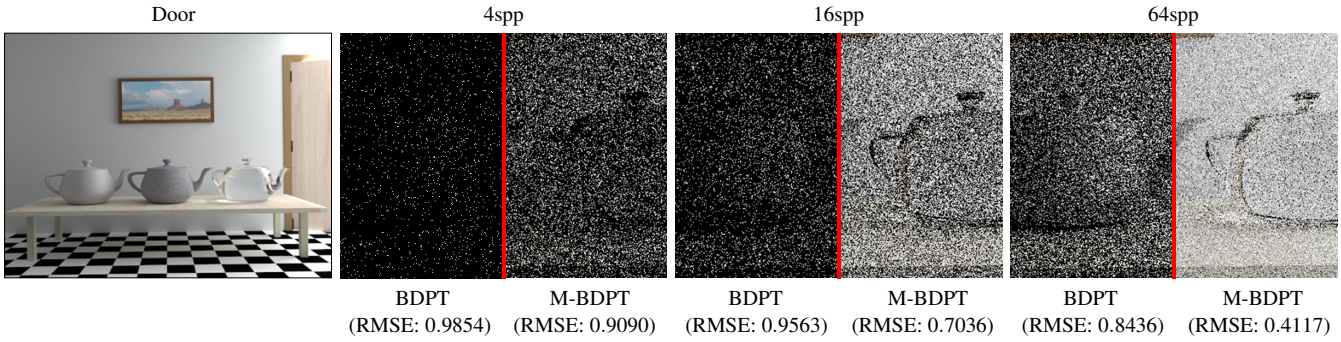


Figure 6: *Door scene rendered using BDPT and M-BDPT. There is an emitter in the adjacent room and ten other non-contributing emitters. M-BDPT is more efficient than BDPT at forming meaningful connections between eye- and light-subpaths, at the same pixel sampling rate.*

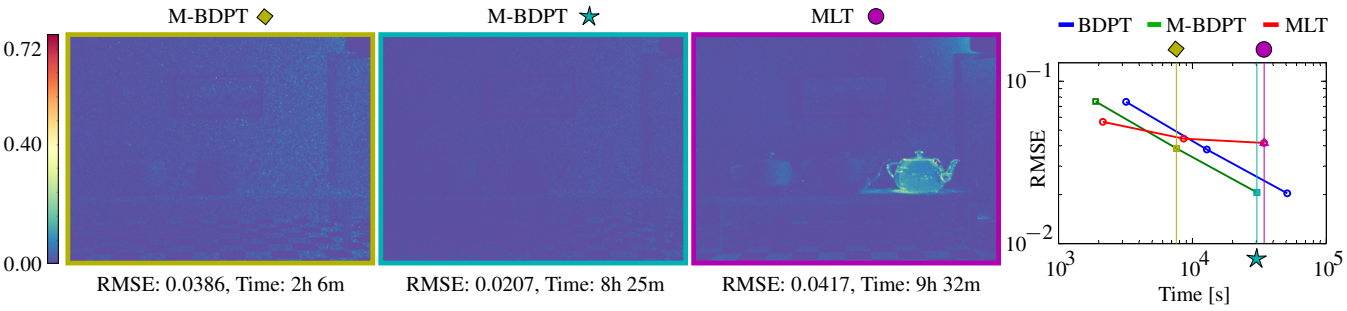


Figure 7: *RMSE errors are shown in false color for the door scene. Convergence plot on the right shows RMSE versus rendering time for BDPT, M-BDPT, and MLT. The three false color images for M-BDPT and MLT are represented by the vertical lines on the convergence plot. M-BDPT converges faster than MLT as we increase the number of samples.*

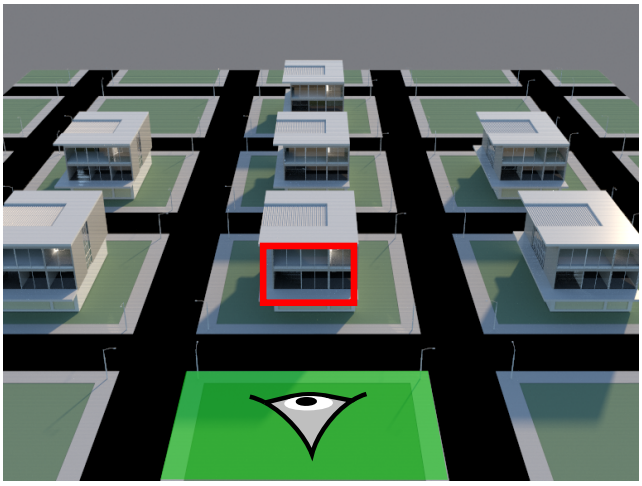


Figure 8: *Front and top views of the stacked house scene. The camera overlooks the house as indicated in the house.*

There are several interesting research avenues that follow our approach. One could imagine using our matrix formulation to seed MCMC sampling approaches. MCMC techniques could also benefit from spatial coherence in the resulting connection transport matrix, proposing mutations that could seek out new sets of eye- and light-subpath connections that are otherwise not explored by existing mutations.

References

[BSH02] BEKAERT P., SBERT M., HALTON J.: Accelerating path tracing by re-using paths. In *Proceedings of the 13th Eurographics Workshop on Rendering* (2002), pp. 125–134. 2

[DKH*14] DACHSBACHER C., KŘIVÁNEK J., HAŠAN M., ARBREE A., WALTER B., NOVÁK J.: Scalable realistic rendering with many-light methods. *Comput. Graph. Forum 33*, 1 (Feb. 2014), 88–104. doi:10.1111/cgf.12256. 2

[HKD14] HACHISUKA T., KAPLANYAN A. S., DACHSBACHER C.: Multiplexed metropolis light transport. *ACM TOG 33*, 4 (July 2014), 100:1–100:10. doi:10.1145/2601097.2601138. 6

[HPB07] HAŠAN M., PELLACINI F., BALA K.: Matrix row-column sampling for the many-light problem. *ACM TOG 26*, 3 (July 2007). doi:10.1145/1276377.1276410. 2, 7

[Kel95] KELLER A.: A quasi-monte carlo algorithm for the global illumination problem in the radiosity setting. In *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing* (1995), pp. 239–251. 2

[Kel97] KELLER A.: Instant radiosity. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques* (1997), ACM Press/Addison-Wesley Publishing Co., pp. 49–56. doi:10.1145/258734.258769. 2

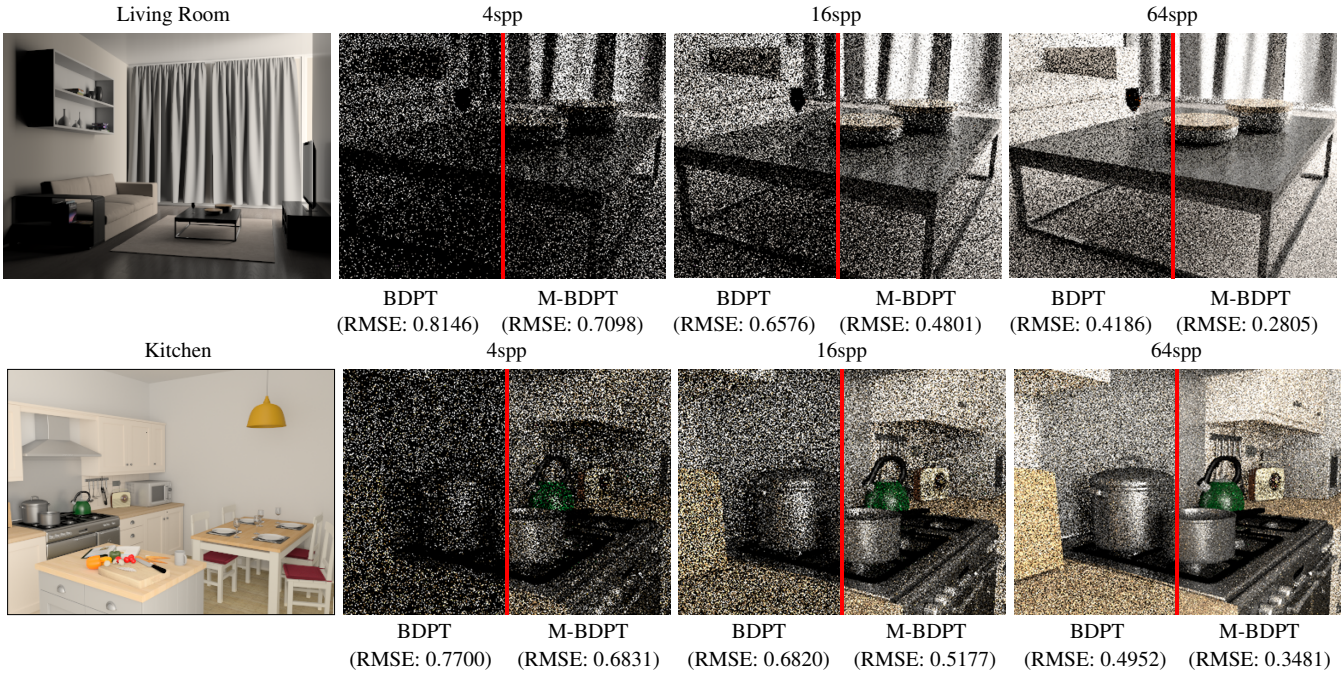


Figure 9: *Comparisons of BDPT and M-BDPT. Both rooms are parts of the building of the stacked house scene.*

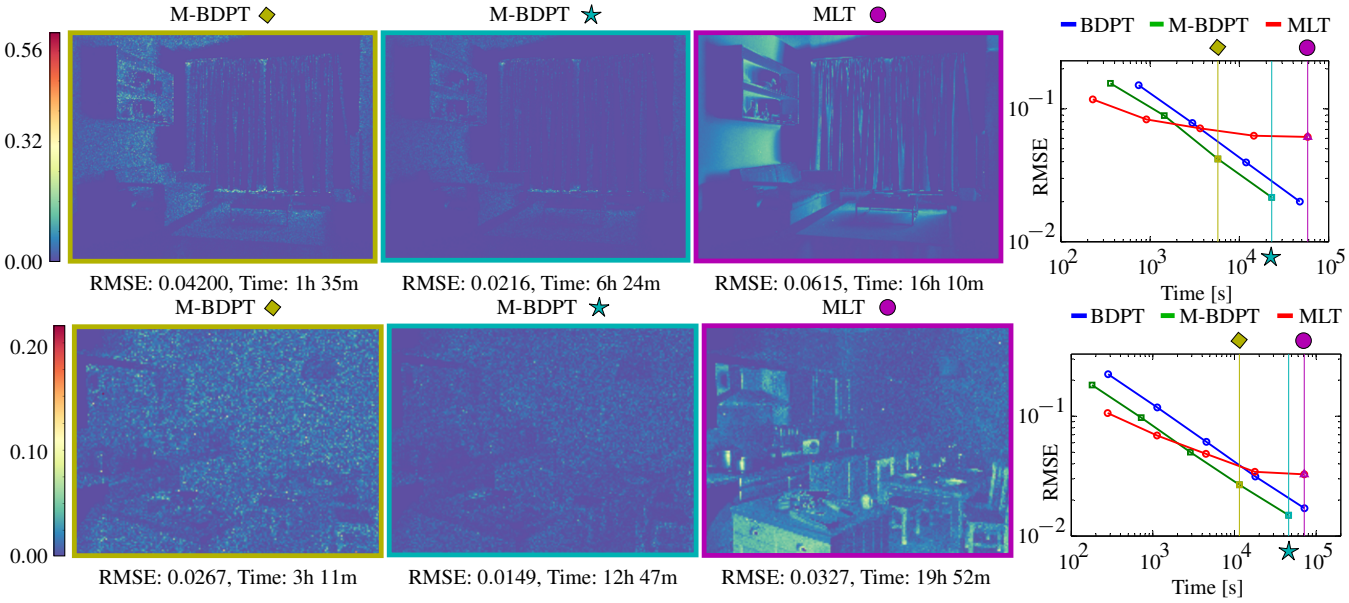


Figure 10: *RMSE errors shown in false color. The M-BDPT and MLT images can be seen on the convergence plot (right) with the corresponding vertical lines. M-BDPT gives better results when compared to MLT and BDPT for a combination of glossy and diffuse surfaces. M-BDPT converges faster than MLT as well.*

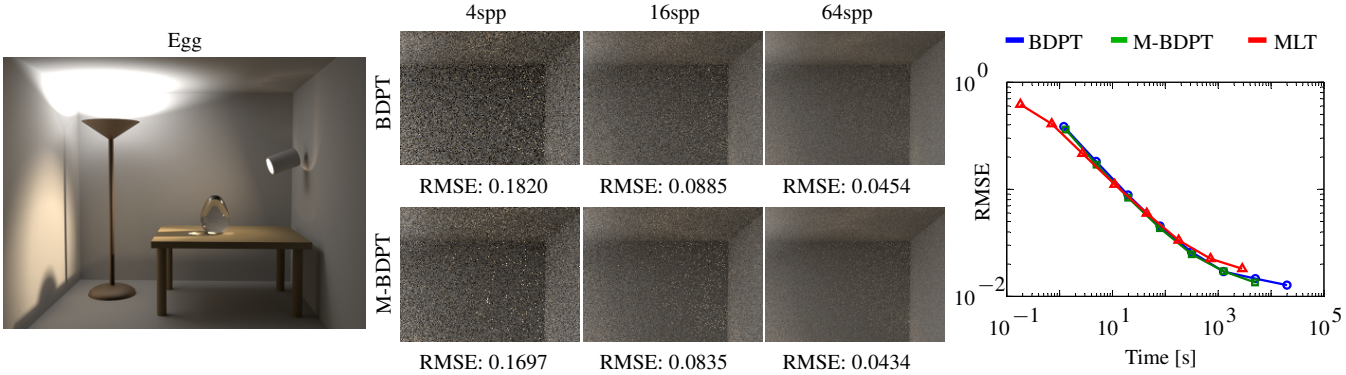


Figure 11: *Veach egg scene rendered using BDPT, M-BDPT, and MLT. The convergence plot for all of them is shown on the right. MLT does not perform better than BDPT for this simple scene. M-BDPT achieves a comparable result to the others, despite its overhead of reordering subpath vertices.*

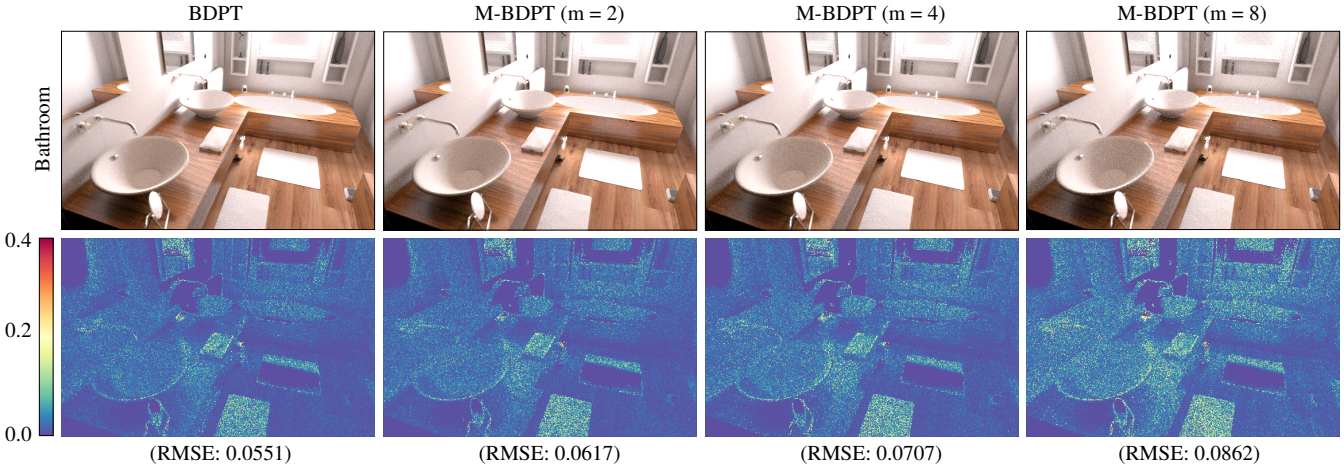


Figure 12: *Bathroom scene rendered in 3 minutes with BDPT and M-BDPT. M-BDPT connects one eye-subpath vertex to m light-subpath vertices. As m increases, M-BDPT reuses more and more light-subpath vertices resulting in increased sample correlation.*

[KSKAC02] KELEMEN C., SZIRMAY-KALOS L., ANTAL G., CSONKA F.: A simple and robust mutation strategy for the metropolis light transport algorithm. *Computer Graphics Forum 21*, 3 (2002), 531–540. doi:10.1111/1467-8659.t01-1-00703. 1

[LW93] LAFORTUNE E. P., WILLEMS Y. D.: Bi-directional path tracing. In *Proceedings of the 3rd International Conference on Computer Graphics and Visualization Techniques* (1993), pp. 145–153. 1, 3

[OP11] OU J., PELLACINI F.: Lightslice: Matrix slice sampling for the many-lights problem. *ACM TOG 30*, 6 (Dec. 2011), 179:1–179:8. doi:10.1145/2070781.2024213. 2, 7

[Pan12] PANTALEONI J.: System and method for estimating light transport using bidirectional path tracing, Oct. 2012. US Patent 8,279,220. URL: https://www.google.com/patents/US8279220. 2, 5

[PBPP11] PAJOT A., BARTHE L., PAULIN M., POULIN P.: Combinatorial bidirectional path-tracing for efficient hybrid cpu/gpu rendering. *Computer Graphics Forum 30*, 2 (2011), 315–324. doi:10.1111/j.1467-8659.2011.01863.x. 2, 4

[PJH16] PHARR M., JAKOB W., HUMPHREYS G.: *Physically Based Rendering: From Theory to Implementation*, 3rd ed. Morgan Kaufmann Publishers Inc., 2016. 6

[PRDD15] POPOV S., RAMAMOORTHI R., DURAND F., DRETTAKIS G.: Probabilistic connections for bidirectional path tracing. In *Proceedings of the 26th Eurographics Symposium on Rendering* (2015), pp. 75–86. doi:10.1111/cgf.12680. 2

[Vea97] VEACH E.: *Robust Monte Carlo methods for light transport simulation*. PhD thesis, Stanford University, 1997. 3

[VG94] VEACH E., GUIBAS L. J.: Bidirectional estimators for light transport. In *Proceedings of Eurographics Rendering Workshop* (1994), pp. 147–162. 1, 3

[VG97] VEACH E., GUIBAS L. J.: Metropolis light transport. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques* (1997), ACM Press/Addison-Wesley Publishing Co., pp. 65–76. doi:10.1145/258734.258775. 1

[WKB12] WALTER B., KHUNGURN P., BALA K.: Bidirectional lightcuts. *ACM TOG 31*, 4 (July 2012), 59:1–59:11. doi:10.1145/2185520.2185555. 2