Combining Resampled Importance and Projected Solid Angle Samplings for Many Area Light Rendering

Ishaan Shah*
CVIT, IIIT-H
India
ishaan.shah@research.iiit.ac.in

Aakash KT*
CVIT, IIIT-H
India
aakash.kt@research.iiit.ac.in

P. J. Narayanan
CVIT, IIIT-H
India
pjn@iiit.ac.in

Figure 1: We propose to combine Resampled Importance Sampling (RIS) from Bitterli et al. [2020] and Projected Solid Angle Sampling [Peters 2021] for rendering scenes with a large number of area light sources. This figure shows an equal-time comparison of our method with RIS on the Zero-Day (left, 10K lights) and Bistro Exterior (right, 30K lights) scenes. Our method traces a similar number of samples as RIS and achieves a lower Mean Absolute Percentage Error (MAPE) w.r.t a 1M spp reference.

ABSTRACT

Direct lighting from many area light sources is challenging due to variance from both choosing an important light and then a point on it. Resampled Importance Sampling (RIS) achieves low variance in such situations. However, it is limited to simple sampling strategies for its candidates. Specifically for area lights, we can improve the convergence of RIS by incorporating a better sampling strategy: Projected Solid Angle Sampling (ProjLTC). Naively combining RIS and ProjLTC improves equal sample convergence. However, it achieves little to no gain in equal time. We identify the core issue for the high run times and reformulate RIS for better integration with ProjLTC. Our method achieves better convergence and results in both equal sample and equal time. We evaluate our method on challenging scenes with varying numbers of area light sources and compare it to uniform sampling, RIS, and ProjLTC. In all cases, our method seldom performs worse than RIS and often performs better.

CCS CONCEPTS

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

*Both authors contributed equally to the paper

SA Technical Communications '23, December 12–15, 2023, Sydney, NSW, Australia

ACM ISBN 979-8-4007-0314-0/23/12…$15.00
https://doi.org/10.1145/3610543.3626165

KEYWORDS

Direct Lighting, Linearly Transformed Cosines (LTC), ReSTIR

ACM Reference Format:
Ishaan Shah, Aakash KT, and P. J. Narayanan. 2023. Combining Resampled Importance and Projected Solid Angle Samplings for Many Area Light Rendering. In *SIGGRAPH Asia 2023 Technical Communications (SA Technical Communications '23), December 12–15, 2023, Sydney, NSW, Australia.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3610543.3626165

1 INTRODUCTION

Direct lighting from environment & area light sources is at the core of Monte Carlo path tracing. Monte Carlo (MC) methods repeatedly sample light sources to evaluate the direct lighting integral. In the case of environment lights, sampling is in accordance with the environment map's probability distribution function (PDF). However, in the case of area lights, sampling involves first choosing one from a list of all area lights and then choosing a point on it, introducing two sources of variance. Thus, direct lighting especially in the presence of many area light sources is an active research area.

Given a shading point x viewed from direction ω_o and a set of area lights $A = \{A_1, A_2, ...A_j\}$, the radiance L at x is given as:

$$L(x, \omega_o) = \sum_{A_j \in A} \int_{A_j} F(x, \omega_o, y; A_j) dy, \tag{1}$$

where $F(x, \omega_o, y) = f_r(y \rightarrow x, \omega_o)V(y \leftrightarrow x)L_i(y \rightarrow x)G(y \leftrightarrow x)$ is the direct lighting integrand with the Bi-directional Reflectance Distribution Function (BRDF) f_r. y is a point on the light A_j with incoming radiance L_i, V is the binary visibility and G the geometry term. MC is used to estimate Eq. 1 with a discrete probability

$p(A_j|x)$ to sample a light and a continuous probability density (PDF) $p(y_i|A_j)$ to sample a point y_i on this light as:

$$L(x, \omega_o) \approx \langle L(x, \omega_o) \rangle = \frac{1}{N} \sum_{i=1}^{N} \frac{F(x, \omega_o, y_i; A_j)}{p(A_j, y_i)}, \tag{2}$$

where $p(A_j, y_i) = p(A_j|x)p(y_i|A_j)$. The variance of this estimator is low if this joint PDF and by extension the individual conditional PDFs closely approximate F.

Methods for sampling a single light from a list of lights on the GPU, either with uniform sampling [Pharr et al. 2023] or importance sampling with data structures [Moreau et al. 2019] have been previously proposed. These methods aim for low sampling variance from $p(A_j|x)$.

Similarly, methods that sample y_i according to the area of the chosen light [Pharr et al. 2023] or it's solid angle [Arvo 1995; Gamito 2016; Ureña et al. 2013] have also been proposed. These methods aim towards low sampling variance from $p(y_i|A_j)$. A recent method [Peters 2021] achieves a stable implementation with near zero variance using projected solid angle sampling and Linearly Transformed Cosines (LTC). This method achieves the state-of-the-art in sampling points on area lights, albeit at the cost of more computation. We refer to this method as *ProjLTC*.

All of these methods strive to get $p(A_j|x)$ & $p(y_i|A_j)$ to as close an approximation to the PDF of F as possible. Instead of relying on data structures for light sampling and attempting to approximate the PDF of F, Spatiotemporal Reservoir Sampling (ReSTIR) [Bitterli et al. 2020] proposes to use simple and efficient strategies for both $p(A_j|x)$ & $p(y_i|A_j)$. The resulting PDF $p(A_j, y_i)$ is then matched to the true target PDF of F using Resampled Importance Sampling (RIS) [Talbot et al. 2005]. RIS achieves state-of-the-art results on scenes with a large number of light sources.

In this paper, we explore a clear gap: using state-of-the-art direct light framework vis RIS in combination with state-of-the-art point sampling of area lights vis ProjLTC, for rendering scenes with a large number of area lights. A naive combination is to directly use ProjLTC for $p(y_i|A_j)$ in RIS. We show that this results in poor run-time as RIS needs to sample this underlying conditional PDF multiple times which is expensive to do with ProjLTC. We propose to improve the run-time by reformulating RIS to operate on $p(A_j|x)$ using LTCs and then applying ProjLTC once to sample from the selected light. We evaluate our and previous methods on three scenes with varying numbers & sizes of light sources. Compared to RIS-ProjLTC (the naive version), uniform sampling, RIS, and ProjLTC, our method achieves lower error in equal time.

Fig. 1 shows equal time comparisons of our method with RIS on two scenes. In both scenes, our method achieves a lower quantitative & perceptual error.

2 PRELIMINARIES

We start with a review of Resampled Importance Sampling (RIS), which is at the core of ReSTIR [Bitterli et al. 2020]. We then review Linearly Transformed Cosines (LTC) & the ProjLTC method [Peters 2021] which uses LTCs with projected solid angle sampling to sample points on area lights.

2.1 Resampled Importance Sampling

Let $p(x)$ be a distribution that is easy to sample from and $\hat{p}(x)$ be the target distribution which is hard to sample. Given M candidate samples $x = \{x_1, ...x_M\}$ drawn from $p(x)$, RIS chooses an index $z \in \{1, ...M\}$ with probability:

$$p(z|x) = \frac{w(x_z)}{\sum_{i=1}^{M} w(x_i)} \quad \text{where} \quad w(x) = \frac{\hat{p}(x)}{p(x)}. \tag{3}$$

The chosen sample from RIS, a correction factor dependent on $w(x)$ and the target PDF $\hat{p}(x)$ can then be used for unbiased MC estimation.

We refer to and use the version of RIS as described by ReSTIR. In the context of direct lighting with area lights, RIS sets $\hat{p}(x) = F(x, \omega_o, y_i; A_j)$ and $p(x) = p(A_j|x)p(y_i|A_j) = p(A_j|x) \cdot 1/|A_j|$. $|A_j|$ denotes the area of the light source j and the number of candidates is set to $M = 32$. The choice of $p(A_j|x)$ is orthogonal to our contribution & method: we focus on using the best sampling strategy for $p(y_i|A_j)$ in RIS.

2.2 Linearly Transformed Cosines

LTC [Heitz et al. 2016] is a method to compute a plausible analytic approximation to the integral in Eq. 1. This is achieved with a matrix that transforms the light and BRDF from the true BRDF to a cosine distribution where analytic integration is possible. Denoting $E(.)$ as the analytic solution:

$$\int_{A_j} F(x, \omega_o, y; A_j) dy \approx E(A_j). \tag{4}$$

LTCs are ultimately an approximation and hence biased if used for final shading.

2.3 Projected Solid Angle Sampling

Peters [2021] described a method that can sample the projected solid angle of an area light with PDF: $p(y_i|A_j) = \frac{G(y_i \leftrightarrow x)}{S(A_j)}$, where $S(.)$ gives the projected solid angle. Given a selected light source, their method applies LTC, transforming the BRDF to the cosine space where projected solid angle sampling can be used. Their method achieves close to zero variance for diffuse BRDFs and uses an MIS strategy for low variance in the case of glossy BRDFs.

3 METHOD

We aim to improve the equal time and equal sample MC convergence for direct lighting with a large number of area light sources. To that end, we start with a simple combination of RIS and ProjLTC (Sect. 3.1). We observe that this leads to excessive run-time and propose to improve it by reformulating RIS for light sampling only and using ProjLTC for sampling a point on the chosen light and final shading (Sect. 3.2).

3.1 Combining RIS and ProjLTC

In RIS for area light shading, the candidate PDF $p(y_i|A_j)$ is uniform over the area of the chosen light. A naive way to incorporate projected solid angle sampling is to use the sampling and PDF of ProjLTC instead. The problem with this approach can be understood from Sect. 2.1: sampling has to be done M times, and ultimately only one sample is chosen. The majority of the sampling effort

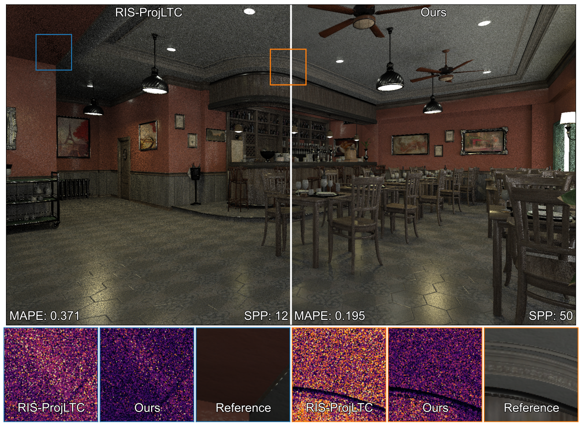

Figure 2: This figure shows equal time (200 ms) comparison of our method (Sect. 3.2) with RIS-ProjLTC (Sect. 3.1) on the Bistro Interior scene with 2K light sources. The insets show difference images with a 1M spp ray-traced reference.

is thus wasted and does not affect the final answer. Furthermore, given that sampling using ProjLTC is more expensive, the efficiency decreases. This is precisely why the original RIS uses other simple & efficient strategies like uniform area sampling for $p(y_i|A_j)$.

3.2 Reformulating RIS

The core issue is that RIS works on *point samples* on area lights. This implies that the candidate PDF $p(x)$ in Eq. 3 should be over all points on all area lights. Therefore, to combine it with ProjLTC, we have no choice but to use ProjLTC for each candidate sample since ProjLTC itself is a point sampling strategy.

We can however reformulate RIS to sample a single light instead of a point on it by setting the target & candidate PDFs to:

$$\hat{p}(x) = E(A_j) \quad \text{and} \quad p(x) = p(A_j|x), \tag{5}$$

where $E(A_j)$ is the analytic solution of LTCs (Eq. 4). Our reformulated version of RIS thus chooses a point on an area light as follows: (1) sample lights from $p(x)$, (2) evaluate $\hat{p}(x)$ on the candidate lights with LTCs, (3) assign weights and sample a single light according to Eq. 3 and (4) apply ProjLTC to this sampled light. Since we use LTCs strictly for sampling and not final shading, our method is unbiased.

Fig. 2 shows an equal time (200 ms) comparison of the naive combination of RIS and ProjLTC (RIS-ProjLTC) with our method combining reformulated RIS with ProjLTC. Our method achieves lower error and traces more samples than RIS-ProjLTC. The difference image insets also show a lower perceptual error for our method.

4 IMPLEMENTATION

We implement our reformulated RIS, projected solid angle sampling, and LTCs in an open-source Vulkan renderer [Peters 2021]. We use the Frostbite BRDF [Lagarde and de Rousiers 2014] as our material model and use LTCs that are optimized for it. Our RIS implementation uses Weighted Reservoir Sampling (WRS) [Wyman 2021] to stream samples, similar to ReSTIR.

ALGORITHM 1: RIS with Projected Solid Angle Sampling

```
1  Def proj_ltc(x, A_j):
     /* Projected Solid Angle Sampling [Peters 2021]                 */
2    L = projected_solid_angle_sample(x, A_j)   // L = f_r · V · L_i · G
3    p̃ = projected_solid_angle_pdf(x, A_j)
4    return L, p̃
5  Def ris_sample_light(x, M):
     /* Refer to [Bitterli et al. 2020] for the definition of
        Reservoir                                                     */
6    Reservoir r
7    r.M = M
8    for i ← 1 to M do
9      generate A_j ~ p(A_j|x)
10     p̂ = E(A_j)                                   // Eq. 5
11     r.update(A_j, p̂ / p(A_j|x))          // Calculate weight Eq. 3
12   r.W = 1/E(r.y) (1/r.M · r.w_sum)      // E(.) is the target PDF p̂
13   return r
14 Def direct_lighting(x):
15   Reservoir r = ris_sample_light(x, 32)          // M = 32
16   L, p̃ = proj_ltc(x, r.y)     // r.y stores the chosen light A_j
17   return L/p̃ · r.W
```

Alg. 1 describes the pseudocode of our method to compute direct lighting given a shading point x (line 14). It starts with sampling of an area light with our reformulated RIS with $M = 32$ (line 15). The next step is to apply projected solid angle sampling (ProjLTC) to the chosen light (line 16) and then finally compute the radiance estimate (line 17). Note that L here is a multiplication of the BRDF, incoming light, visibility, and geometry terms (line 2).

The `ris_sample_light` function (line 5) describes our reformulated RIS that samples a light using LTCs. We use the definition of a `Reservoir` from ReSTIR [Bitterli et al. 2020] (lines 6, 7). Each of the M light candidates are sampled using $p(A_j|x)$ and the target PDF $\hat{p}$ is set to the respective candidate's LTC evaluation (Eq. 5, lines 9, 10). We use the LTC approximation of Hill and Heitz [2016] for efficiency. Finally, line 11 computes the weights (Eq. 3) and updates the reservoir.

The `proj_ltc` method is straightforward: it takes shading point x and the chosen light A_j and samples it using projected solid angle sampling, potentially using MIS [Peters 2021] (line 2). This function also returns the corresponding sampling PDF (line 3).

5 RESULTS & ANALYSIS

We qualitatively & quantitatively compare our method against the following: (1) **Uniform**: Uniform light & uniform area sampling, (2) **ProjLTC**: Uniform light sampling & ProjLTC, (3) **RIS**: RIS with uniform light & uniform area sampling and (4) **RIS-ProjLTC**: Naive combination of RIS with ProjLTC (Sect. 3.1).

We use three scenes with varying numbers and sizes of area lights for the evaluation: (1) Zero-Day, 10K lights, (2) Bistro Interior, 2K lights and (3) Bistro Exterior, 30K lights. All renderings are done at a resolution of 1920 × 1080 on an NVIDIA RTX 3090. We use the Mean Absolute Percentage Error (MAPE) to quantify the error with respect to a ray-traced 1M spp reference.

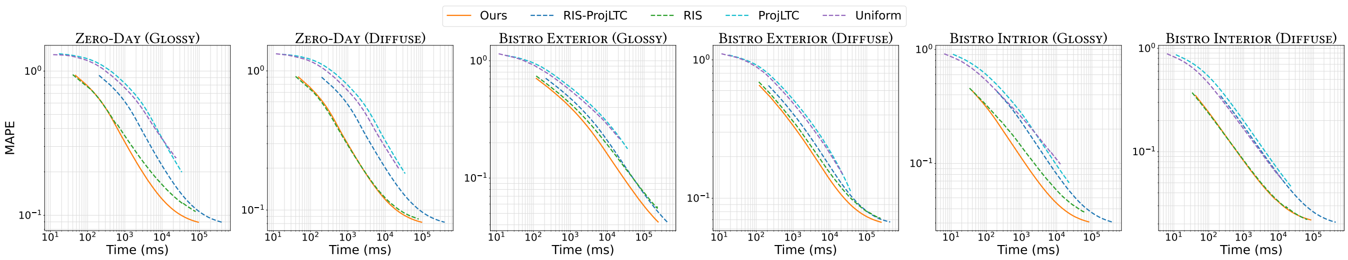

Figure 3: We plot equal time convergence graphs of our method (orange, Sect. 3.2), RIS-ProjLTC (blue, Sect. 3.1), RIS (green), ProjLTC (cyan) and uniform sampling (purple) for both diffuse & glossy versions of three test scenes. Our method achieves lower Mean Absolute Percentage Error (MAPE) in the same amount of time.

Table 1: We show time to render one frame in milliseconds (ms) and equal sample MAPE for RIS and our method on three scenes.

Scene	RIS (ms)	Ours (ms)	MAPE 100 spp RIS	MAPE 100 spp Ours	MAPE 2K spp RIS	MAPE 2K spp Ours
Zero-Day (10K)	3.98	4.70	0.499	**0.458**	0.173	**0.131**
Bistro Int. (2K)	3.33	4.06	0.214	**0.168**	0.066	**0.047**
Bistro Ext. (30K)	11.85	12.03	0.411	**0.378**	0.135	**0.107**

Fig. 1 and Fig. 2 show rendered results with difference images for our and previous methods. Our method traces a nearly equivalent number of samples as RIS while achieving lower error. Fig. 3 shows equal time convergence graphs for all methods. These graphs are plotted for diffuse & glossy versions of the three scenes mentioned above. These graphs show that our method converges to a lower error in equal time and that our method is most beneficial for glossy scenes. Note that it never performs worse than RIS in the case of diffuse scenes. Also note that our method's convergence graph is essentially a translation of RIS-ProjLTC, clearly demonstrating that our method improves its efficiency.

Finally, we show the time taken to render one frame in milliseconds (ms) and equal sample MAPE of our method in comparison to RIS in Table 1. On average, our method takes about 1ms more time to render but achieves a lower error. We refer the reader to our supplementary materials for more results and comparisons.

6 DISCUSSION

Our method makes a contribution to improving the convergence of RIS. ReSTIR uses RIS at its core and thus ReSTIR itself and any improvement on it will benefit from our contribution. For example, the spatio-temporal reuse of samples as already done by ReSTIR can use our method instead to further improve the convergence.

It should be noted that our reformulated RIS will always be faster than the naive combination of RIS and ProjLTC, independent of the hardware and the implementation. To see why, consider that ProjLTC has to evaluate LTCs and then do a few more operations on top for sampling. Since our method stops after evaluating LTCs, it will always be faster.

Our method, like RIS (and ReSTIR) naturally supports textured area lights. Note however that in this case, the target PDF of RIS will be further away from the true target which may affect convergence.

One of the limitations of our method is that it is limited to single-lobed BRDFs as LTCs cannot represent multi-lobed BRDFs.

7 CONCLUSION & FUTURE WORK

We presented a naive combination of Resampled Importance Sampling (RIS) and Projected Solid Angle Sampling (ProjLTC) for rendering scenes with many area lights. We identified issues that make this method inefficient. We then proposed a reformulation of RIS that improved the efficiency of this combination, achieving a similar run-time as RIS with a lower error. We analyzed our method's convergence and compared with previous methods demonstrating superior quantitative and qualitative performance.

For future work, we would like to investigate the performance of using LTCs instead of ProjLTC with our RIS reformulation. We would also like to investigate the performance in presence of textured area lights. Finally, using LTCs for sampling instead of final evaluation, to achieve unbiased rendering is a recent trend which we would like to further explore.

REFERENCES

James Arvo. 1995. Stratified Sampling of Spherical Triangles *(SIGGRAPH '95)*. Association for Computing Machinery, New York, NY, USA, 437–438.

Benedikt Bitterli, Chris Wyman, Matt Pharr, Peter Shirley, Aaron Lefohn, and Wojciech Jarosz. 2020. Spatiotemporal reservoir resampling for real-time ray tracing with dynamic direct lighting. *ACM Transactions on Graphics* 39, 4 (2020).

Manuel N. Gamito. 2016. Solid Angle Sampling of Disk and Cylinder Lights. *Comput. Graph. Forum* 35, 4 (jul 2016), 25–36.

Eric Heitz, Jonathan Dupuy, Stephen Hill, and David Neubelt. 2016. Real-Time Polygonal-Light Shading with Linearly Transformed Cosines. *ACM Trans. Graph.* (2016).

Stephen Hill and Eric Heitz. 2016. Advances in real-time rendering - Real-time area lighting: a journey from research to production. In *ACM SIGGRAPH 2016 Courses*.

Sébastien Lagarde and Charles de Rousiers. 2014. Physically Based Shading in Theory and Practice: Moving Frostbite to PBR. In *ACM SIGGRAPH 2014 Courses*.

P. Moreau, M. Pharr, and P. Clarberg. 2019. Dynamic Many-Light Sampling for Real-Time Ray Tracing *(HPG '19)*. Eurographics Association, Goslar, DEU.

Christoph Peters. 2021. BRDF Importance Sampling for Polygonal Lights. *ACM Trans. Graph.* 40, 4, Article 140 (jul 2021), 14 pages.

Matt Pharr, Wenzel Jakob, and Greg Humphreys. 2023. *Physically based rendering: From theory to implementation.* MIT Press.

Justin F. Talbot, David Cline, and Parris Egbert. 2005. Importance Resampling for Global Illumination. In *EGSR*.

Carlos Ureña, Marcos Fajardo, and Alan King. 2013. An Area-Preserving Parametrization for Spherical Rectangles *(EGSR '13)*. Eurographics Association.

Chris Wyman. 2021. *Weighted Reservoir Sampling: Randomly Sampling Streams.* Apress, Berkeley, CA, 345–349. https://doi.org/10.1007/978-1-4842-7185-8_22