Generalized Resampled Importance Sampling: Foundations of ReSTIR

DAQI LIN*, University of Utah, USA
MARKUS KETTUNEN*, NVIDIA, Finland
BENEDIKT BITTERLI, NVIDIA, USA
JACOPO PANTALEONI, NVIDIA, Germany
CEM YUKSEL, University of Utah, USA
CHRIS WYMAN, NVIDIA, USA

Fig. 1. *Our new generalized resampled importance sampling (GRIS) theory extends resampled importance sampling [Talbot 2005] to guarantee convergence even when applied to correlated samples arising from spatiotemporal reuse (i.e., Bitterli et al. [2020]). GRIS allows applying ReSTIR to reuse arbitrary paths, shown with paths of length 10 in the* Carousel *and* Paris Opera House*. Main images compare naive path tracing and our new ReSTIR PT in equal time (80 ms at* 1920×1080*). Insets show equal-time path tracing, ReSTIR GI [Ouyang et al. 2021], our ReSTIR PT, plus a converged reference. We significantly improve quality for glossy interreflection, reflections, refractions, and other high-frequency lighting. For* Carousel*, MAPE errors: path tracing (1.63), ReSTIR GI (0.45), and ReSTIR PT (0.39). Corresponding errors in* Opera House*: 1.28, 0.39, and 0.33. (*Carousel *©carousel_world;* Paris Opera House *courtesy ©GoldSmooth from TurboSquid.)*

As scenes become ever more complex and real-time applications embrace ray tracing, path sampling algorithms that maximize quality at low sample counts become vital. Recent *resampling* algorithms building on Talbot et al.'s [2005] resampled importance sampling (RIS) reuse paths spatiotemporally to render surprisingly complex light transport with a few samples per pixel. These reservoir-based spatiotemporal importance resamplers (ReSTIR) and their underlying RIS theory make various assumptions, including sample independence. But sample reuse *introduces correlation*, so ReSTIR-style iterative reuse loses most convergence guarantees that RIS theoretically provides.

We introduce generalized resampled importance sampling (GRIS) to extend the theory, allowing RIS on correlated samples, with unknown PDFs and taken from varied domains. This solidifies the theoretical foundation, allowing us to derive variance bounds and convergence conditions in ReSTIR-based samplers. It also guides practical algorithm design and enables advanced path reuse between pixels via complex shift mappings.

We show a path-traced resampler (ReSTIR PT) running interactively on complex scenes, capturing many-bounce diffuse and specular lighting while shading just one path per pixel. With our new theoretical foundation, we can also modify the algorithm to guarantee convergence for offline renderers.

CCS Concepts: • **Computing methodologies** → **Rendering**.

ACM Reference Format:
Daqi Lin, Markus Kettunen, Benedikt Bitterli, Jacopo Pantaleoni, Cem Yuksel, and Chris Wyman. 2022. Generalized Resampled Importance Sampling: Foundations of ReSTIR. *ACM Trans. Graph.* 41, 4, Article 75 (July 2022), 23 pages. https://doi.org/10.1145/3528223.3530158

1 INTRODUCTION

Monte Carlo algorithms form the core of modern rendering. While originally only feasible in offline renderers, ray-tracing hardware [Kilgariff et al. 2018] has made such algorithms practical in real-time systems as well. However, strict real-time constraints in games limit feasible per-frame ray counts [Halen et al. 2021], giving many modern real-time path tracers budgets of *at most* one path per pixel.

Importance sampling reduces variance at low sample counts by improving sample distributions. But this becomes challenging for

*Joint first authors; equal contribution.

Authors' addresses: Daqi Lin, University of Utah, USA, daqi@cs.utah.edu; Markus Kettunen, NVIDIA, Finland, mkettunen@nvidia.com; Benedikt Bitterli, NVIDIA, USA, bbitterli@nvidia.com; Jacopo Pantaleoni, NVIDIA, Germany, jpantaleoni@nvidia.com; Cem Yuksel, University of Utah, USA, cem@cemyuksel.com; Chris Wyman, NVIDIA, USA, chris.wyman@acm.org.

0730-0301/2022/7-ART75 $15.00
https://doi.org/10.1145/3528223.3530158

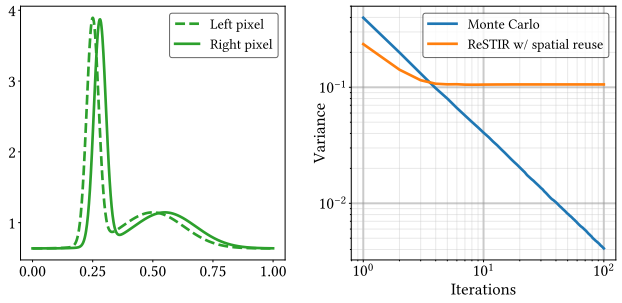


Fig. 2. *Imagine a two-pixel image, with ReSTIR [Bitterli et al. 2020] separately integrating two 1D functions (left). ReSTIR promises exponential growth in "effective" sample count at linear cost, but each ReSTIR iteration only adds two new independent samples. Other reused samples are duplicates, causing convergence to the wrong result (right). Our GRIS theory explains when such cases occur and how to guarantee proper convergence. For a less abstract, rendered example of inaccurate convergence due to correlations, see Figure 10b.*

complex global lighting, e.g., Figure 1, where sampling from optimal distributions is impossible. Path guiding aims to learn complex distributions online, but requires updating complex data structures [Vorba et al. 2019] or neural models [Müller et al. 2019].

A new family of algorithms based on resampled importance sampling (RIS) [Talbot 2005] instead continually evolves a population of samples towards their optimal distribution via sample reuse within and across frames. Ideally, each sample converges to its "perfect" importance distribution given sufficient reuse. Such reservoir-based spatiotemporal importance resampling (ReSTIR) algorithms work for direct lighting [Bitterli et al. 2020], global illumination [Ouyang et al. 2021], and volume scattering [Lin et al. 2021]. ReSTIR leverages GPU parallelism via a streaming algorithm, reducing error up to 100× compared to equal-time renderings without reuse.

However, convergence of these randomized distributions is poorly studied. Nabata et al. [2020] approximate convergence for Talbot RIS with an upper bound, but only without sample reuse between pixels. Bitterli et al. [2020] show these distributions are *unbiased*, but do not prove they converge in all circumstances.

In fact, in Figure 2 we show a trivial example where sample reuse, despite being unbiased, converges to a wrong result.

Ultimately, ReSTIR ignores a key issue: RIS assumes *independent and identically distributed* (i.i.d.) samples, often from a single source distribution. Reuse violates this independence, and ignoring the assumption slows convergence or causes divergence. Prior work empirically suggests sufficiently small correlation does not impede convergence [Bitterli et al. 2020; Lin et al. 2021; Ouyang et al. 2021; Wyman and Panteleev 2021]. But it remains unclear if and when their correlation minimization efforts (e.g., randomizing reused spatial neighbors) guarantee convergence. When resampling for more complex lighting, maintaining sufficient decorrelation may be impossible without a deeper theoretical understanding.

We introduce *generalized resampled importance sampling* (GRIS), a new theoretical framework that lifts the i.i.d. assumption and helps understand, design, and discuss convergence for complex samplers, like ReSTIR. With GRIS, we can apply resampling to combine correlated candidate samples, drawn from potentially different domains and mapped to estimate a single integral (see Section 4).

Many derivations in Talbot [2005] and Bitterli et al. [2020] are special cases of our theory; we generalize prior work while proving conditions under which ReSTIR is unbiased and consistent.

Our main contributions include that we:

- Derive RIS with paths from other pixels by shift mappings,
- Give conditions for unbiasedness and convergence,
- Derive MIS weights satisfying the convergence constraints and help minimize variance, (Section 4.4),
- Explain how some prior ReSTIR design decisions, e.g., M-capping, are *vital* for ensuring convergence (Section 6.4),
- Show proper shift mappings help control noise when spatiotemporally reusing paths (Section 7),
- Design shift mappings with improved performance and quality by BSDF lobe specific connections (Sections 7.5 and 7.6),
- Apply GRIS theory to derive our ReSTIR PT that can reuse paths e.g., through glass.

Specifically, to guarantee convergence (see Section 5) when integrating a function f, one must:

- Use correct MIS weights during sample reuse,
- Select the target function $\hat{p}$ so $f/\hat{p}$ is not arbitrarily large,
- Control samples' resampling weights w_i so Var $[\sum w_i] \to 0$,
- Ensure sufficient sample count across f's domain, specifically to have enough "canonical" samples (see Section 5.5), and
- When temporally resampling, use a reasonable M-cap to limit correlations between frames.

With our new theory and shift maps, we more efficiently reuse samples, obtaining a robust, unbiased light transport algorithm that can handle even very complex lighting scenarios while remaining fully amenable to efficient GPU parallelization and real-time use (see Figure 1). We also show that, without temporal resampling, ReSTIR can further be used to largely accelerate offline renderers. Our paper includes source code, allowing readers to experiment.[1]

While many proofs and derivations reside in our supplemental material, Sections 4 and 5 remain mathematically dense. We have boxed key results throughout, and starred sections (⋆) skippable by readers less interested in theory. For engineers, we suggest reading through Section 4.1 and then skipping to Section 7.

1.1 Paper Roadmap

In Section 2 we cover key background GRIS builds on, including an overview of closely related resampling and sample reuse algorithms.

In Section 3 we briefly review the state-of-the-art in resampled importance sampling theory and motivate the need for extending it.

In Section 4 we present our new generalization of RIS to resample from multiple input domains Ω_i into a target domain Ω, using shift maps $T_i : \Omega_i \to \Omega$ analogous to those in gradient-domain rendering. We establish conditions under which GRIS unbiasedly integrates any function f defined over Ω, and conditions ensuring output sample distributions converge to the specified target resampling PDF $\bar{p}$.

[1] https://github.com/DQLin/ReSTIR_PT

In Section 5 we show integration error directly relates to the variance of a RIS normalization factor. When this variance disappears, GRIS becomes a zero-variance integrator. We can achieve this by taking additional samples from the current, *canonical* pixel (not just from distant neighbors), and using robust resampling MIS weights.

By configuring ReSTIR to obey the GRIS convergence constraints, in Section 6, we observe it becomes a non-Markovian chain, forever exploring path space with one sample per pixel. In a still scene, averaging frames converges, and real-time usage gives a single state of the chain each frame. Cross-frame correlations hinder convergence for offline rendering, but spatial reuse remains beneficial.

Section 7 designs shift mappings for cross-pixel path reuse, and presents several new shift modifications to improve efficiency. Section 8 discusses our implementation of ReSTIR PT, and Section 9 presents results and experimental validation.

Our supplemental document Section S contains mathematical proofs and derivations, additional analysis, and more details about our implementation. We also provide a results viewer and video.

x, y	A general input to a function
$\bar{\mathbf{x}}$, $\mathbf{x}_i$	A path and a vertex i on the path
Ω_i	Domain from which samples are drawn
Ω	Domain of integration of our function f
X_i	Input sample for RIS, often a sequence $(X_i)_{i=1}^M$
Y	Sample Y selected via RIS
	($Y = X_s$ in the simple case or $Y = T_s(X_s)$ in general)
M, N	Number of input and output samples for RIS
$p_X(\cdot)$	Probability density of random variable X at a location
$p(\cdot)$	Shorthand for the above when the random variable is clear
$\hat{p}(\cdot)$	Unnormalized *target* distribution (we *aim* to select $Y \propto \hat{p}$)
$\bar{p}(\cdot)$	Normalized target PDF (i.e., $\bar{p} = \hat{p}/\|\hat{p}\|_1$)
$f(\cdot)$	Function to integrate (e.g., the path contribution function)
$g_i(\cdot)$	A contribution function for $X_i \in \Omega_i$ to integrate f in Ω
W_i	*Unbiased contribution weights*; estimate reciprocal PDFs
w_i	*Resampling weights*; RIS selects one X_i based on $w_i/\sum w_j$
c_i	*Contribution MIS weights*; prior works' MIS weights
m_i	Our new *resampling MIS weights*
$T_i(\cdot)$	A shift mapping; maps samples from domain Ω_i to Ω
$\left\|\frac{\partial T_i}{\partial x}\right\|$	Jacobian of shift mapping T_i
$\hat{p}_{\leftarrow i}(\cdot)$	"$\hat{p}$ from i." Generalizes $\hat{p}$ to include shift maps from Ω_i
C	Various constants, as bounds in convergence proofs
R, $\|R\|$	Canonical samples and their number

Table 1. *Summary of paper notation*

2 BACKGROUND

Sampling, importance sampling, and sample reuse are key to modern renderers and form a substantial body of research in graphics. By generalizing RIS and ReSTIR, we closely relate to resampling and sample reuse methods; we focus discussion on those techniques after briefly summarizing the larger context of sampling.

Our work fits into a rich history of path reuse [Bauszat et al. 2017; Bekaert et al. 2002], path guiding [Müller et al. 2017; Vorba et al. 2014], path perturbations and mutations [Kelemen et al. 2002; Veach and Guibas 1997], next-event estimation [Donikian et al. 2006; Vévoda et al. 2018], ratio estimators [Heitz et al. 2018; Stachowiak 2015], bidirectional path connections [Chaitanya et al. 2018; Lafortune and Willems 1993; Popov et al. 2015; Tokuyoshi and Harada 2019], light importance sampling [Peters 2021; Shirley et al. 1996], structures to accelerate sample lookups [Jensen 2001; Moreau et al. 2019; Walter et al. 2005], low-discrepancy sampling [Heitz and Belcour 2019], and online learning based sampling [Müller et al. 2019; Pantaleoni 2020; Vorba et al. 2014; Zhu et al. 2021].

When targeting real-time, even with hardware ray tracing, improving quality at iso-performance is insufficient; results with low sample budgets need denoising [Chaitanya et al. 2017; Hasselgren et al. 2020]. Widely used denoisers [Schied et al. 2018] work as post-processes, discarding intermediate rendering data. Some recent denoisers directly process samples from the renderer [Gharbi et al. 2019], and resampling [Talbot 2005] or sequential Monte Carlo [Ghosh et al. 2006] methods can be seen as filtering probability distributions, improving sample quality in filtered regions. Gradient-domain rendering [Hua et al. 2019; Kettunen et al. 2015; Lehtinen et al. 2013] can also be viewed as using correlated samples to cancel noise in path space [Kettunen 2020]. Path space filtering [Binder et al. 2019; Pantaleoni 2020] averages path contribution over spatiotemporal neighborhoods, trading variance for bias.

2.1 Resampling Algorithms

Our work generalizes recent resampling methods, which build on *sampling importance resampling* (SIR) [Rubin 1987]. SIR obtains better-distributed samples $(Y_i)_{i=1}^N = (Y_1, \ldots, Y_N)$ by subsampling a set of i.i.d. samples $(X_i)_{i=1}^M$ proportional to resampling weights $w_i = \hat{p}(X_i)/p(X_i)$, where $\hat{p}(x)$ represents a desired (potentially unnormalized) target distribution. As M grows, the distribution of samples Y_i converges to $\bar{p} = \hat{p}(x)/\|\hat{p}\|$.[2] See Guetz [2012] for an in-depth overview of SIR. A related method is Population Monte Carlo (PMC) [Cappé et al. 2004; Lai et al. 2007], which combines resampling, mutation, and regeneration to evolve a sample population towards a target distribution over multiple iterations.

Resampled Importance Sampling. Talbot [2005] introduces RIS, which provides proper normalization for SIR-selected samples when used in Monte Carlo integration. RIS extends SIR to allow sourcing $(X_i)_{i=1}^M$ from different probability distributions in a single domain, and provides multiple importance sampling (MIS) weights that ensure convergence to the target distribution in such cases.

Reservoir Sampling. Chao [1982] introduces a reservoir sampling algorithm that picks a random sample from input set $(X_i)_{i=1}^M$ in a single-pass streaming manner. A *reservoir* stores the selected sample, current stream length M, and sum of weights w_i ($i \le M$); each new stream element X_i replaces the selected sample with probability $w_i/\sum_{j\le i} w_j$. Reservoir sampling pairs naturally with resampling; combined they perform RIS in a streaming manner with constant memory footprint.

Reservoir-based spatiotemporal RIS. ReSTIR [Bitterli et al. 2020] uses chained reservoir resampling to share samples across pixels and frames. It alternately generates new independent samples for each reservoir (e.g., per pixel) and reuses samples between similar reservoirs (i.e., domains). Sharing well-distributed samples across integration domains improves sample distribution and amortizes costs of generating initial samples. ReSTIR was first applied to direct lighting with reuse in screen space, where the integration domains

[2] For functions, we use $\|\cdot\|$ without a subscript for the 1-norm $\|\cdot\|_1$ (e.g., $\|\hat{p}\|$).

The RIS Algorithm	Talbot et al. [2005] *Identically distributed samples*	Talbot [2005] *Differently distributed samples*	GRIS [Ours] *Correlations & different source domains*
(1) Generate M initial candidate samples: $(X_1, \ldots, X_M)$	Samples from same domain: $X_i \in \Omega$ with same PDF p	Samples from same domain: $X_i \in \Omega$ with different PDF p_i	Samples from arbitrary domains: $X_i \in \Omega_i$; intractable p_i are OK
(2) Evaluate their *unbiased contribution weights*: W_i	$W_i = 1/p(X_i)$	$W_i = 1/p_i(X_i)$	W_i must unbiasedly estimate $1/p_i(X_i)$
(3) Evaluate their *resampling weights*: w_i	$w_i = \frac{1}{M}\hat{p}(X_i)W_i$	$w_i = m_i(X_i)\,\hat{p}(X_i)W_i$	$w_i = m_i(T_i(X_i))\,\hat{p}(T_i(X_i))\,W_i\,\lvert\partial T_i/\partial X_i\rvert$
(4) Select s proportionally to w_i and output Y in Ω	Simply output: $Y = X_s$	Simply output: $Y = X_s$	Output sample mapped from Ω_i to Ω: $Y = T_s(X_s)$

Fig. 3. *We generalize Talbot's [2005] resampled importance sampling in various ways. (Red) Basic RIS assumes i.i.d. samples X_i, all drawn with one PDF p from the domain Ω of integrand f. (Blue) More advanced forms allow candidates with different PDFs $p_i(X)$, adding MIS terms m_i to remain unbiased. Sample reuse, as in ReSTIR [Bitterli et al. 2020], adds correlations between candidate samples X_i and requires using unbiased estimates of $1/p_i(X_i)$ for W_i. But current theory fails to guarantee convergence in these cases (e.g., Figure 2). (Green) Our new theory corrects this, providing convergence guarantees even with correlated candidate samples X_i from arbitrary domains Ω_i and differing, intractable PDFs p_i. The unbiased estimate for the integral of f is $f(Y)W_Y$ in all cases, with W_Y defined in Equation 2.*

are fixed across pixels (i.e., the surface of all lights). Recent work extends ReSTIR to world-space sample reuse [Boissé 2021; Boksansky et al. 2021] and longer paths [Lin et al. 2021; Ouyang et al. 2021] for global illumination, where integration domains and reasoning about correctness become more complex.

Shift mappings. Path reuse algorithms evaluate pixel color by reusing path samples between pixels. As in ReSTIR and RIS, the formulation of Bekaert et al. [2002] requires path samples to come from a shared domain, while more recent work [Bauszat et al. 2017] allows different integration domains and explicitly defines *shift mappings* to map paths between them. This better reuses complex light transport paths including specularity from glass and mirrors. By extending RIS and ReSTIR to utilize general shift mappings, we handle these complexities better than Ouyang et al. [2021].

Shift mappings originally arose in gradient-domain rendering [Kettunen et al. 2015; Lehtinen et al. 2013], where the image is reconstructed with discrete image gradients, evaluated by subtracting a path's contribution from its copies *shifted* into adjacent pixels.

Many shift mappings have been proposed: reconnecting to the first rough vertex [Lehtinen et al. 2013], manifold exploration shifts [Lehtinen et al. 2013] and half-vector copying [Kettunen et al. 2015] for specular transport, random number replay [Hua et al. 2019; Kettunen et al. 2015; Manzi et al. 2016], and numerous extensions to e.g., bidirectional path tracing [Manzi et al. 2015], photon mapping [Gruson et al. 2018; Hua et al. 2017], participating media [Gruson et al. 2018], vertex connection and merging [Sun et al. 2017], and spectral rendering [Petitjean et al. 2018]. Hua et al.'s 2019 recent survey provides a deeper view of shift mappings and gradient-domain rendering. Path perturbations and shift mappings also occur in Metropolis Light Transport [Van de Woestijne et al. 2017; Veach and Guibas 1997] and local QMC exploration [Tessari et al. 2017].

3 RESAMPLED IMPORTANCE SAMPLING REVIEW

Before introducing GRIS in Section 4, we first review resampled importance sampling (RIS) using the notation and terminology of our generalized theory. Figure 3 highlights differences between existing theory, e.g., Talbot [2005], and our new generalization.

3.1 Identically Distributed Samples

Basic RIS takes as input a sequence of independent and identically distributed (i.i.d) random samples $(X_i)_{i=1}^M$ in some domain Ω, distributed with known PDF p. The goal is to randomly pick Y from the sequence so that its PDF, p_Y, constitutes a better importance sampler for integrating function f over Ω.

More precisely, we define a non-negative target function $\hat{p}$ and choose Y randomly such that, as the input sample count M grows, the realized PDF p_Y better and better approximates a normalized $\hat{p}$ (i.e., p_Y approximates $\bar{p}(x) = \hat{p}(x)/\|\hat{p}\|$).

Algorithmically, from inputs X_i we select one, $Y = X_s$, with probability $\Pr[s=i] = w_i/\sum_{j=1}^M w_j$, using resampling weights w_i. Prior work defines w_i as $\hat{p}(X_i)/p(X_i)$ (e.g., Bitterli et al. [2020], Eq. 5). As weights are relative, selection probability is invariant to multiplicative constants, and we define

$$w_i = \frac{1}{M}\hat{p}(X_i)W_i \qquad \text{and} \qquad W_i = \frac{1}{p(X_i)} \tag{1}$$

for notational consistency. The PDF of the selected sample Y is intractable, but its *unbiased contribution weight*

$$W_Y = \frac{1}{\hat{p}(Y)}\sum_{i=1}^{M} w_i \tag{2}$$

can be used in place of $1/p_Y(Y)$ (e.g., Bitterli et al. [2020], Eq. 12). Assuming $p_Y > 0$ where $f > 0$, i.e., $\operatorname{supp} f \subset \operatorname{supp} Y$, we have

$$\int_\Omega f(x)\,\mathrm{d}y = \mathbb{E}\left[f(Y)W_Y\right]\ . \tag{3}$$

Given appropriate constraints, p_Y converges to $\bar{p}$ and the Monte Carlo variance $\mathrm{Var}\,[f(Y)W_Y]$ asymptotically approaches the variance expected if Y had PDF *exactly* $\bar{p}$. Choosing $\hat{p}$ proportional to f guarantees the estimate $f(Y)W_Y$ is, asymptotically, zero-variance.

3.2 Differently Distributed Samples

If the samples X_i have different PDFs p_i, the situation becomes more complex. This requires what we call *resampling MIS*, a partition of unity with weights m_i, for $m_i \geq 0$ and

$$\sum_{i=1}^{M} m_i(x) = 1 \tag{4}$$

for all x in $\hat{p}$'s support. Talbot [2005] proposes weights analogous to Veach's [1998] balance heuristic,

$$m_i(x) = \frac{p_i(x)}{\sum_{j=1}^{M} p_j(x)}. \tag{5}$$

The key algorithmic change is then replacing the $1/M$ term in w_i (Equation 1) with these MIS weights (see Figure 3, blue column), i.e.,

$$w_i = m_i(X_i)\,\hat{p}(X_i)\,W_i \qquad \text{and} \qquad W_i = \frac{1}{p_i(X_i)}\,. \tag{6}$$

Assuming at least one PDF p_i covers each $x \in \mathrm{supp}\,\hat{p}$, Equation 3 holds with W_Y from Equation 2 using these updated w_i. Convergence requires more assumptions than in Section 3.1, but is achievable (e.g., Section 5.7).

3.3 Why Generalize Resampling?

Early applications of RIS, e.g., for BSDF importance sampling, aim to choose a $\hat{p}$ that cheaply approximates f so that resampling from multiple cheaply generated candidates speeds convergence.

ReSTIR, however, *reuses* samples across pixels to amortize costs for simultaneous estimation of multiple integrals. With this goal, $\hat{p}$ need not be simpler than f, if reusing prior samples is cheaper than generating a new one. ReSTIR also gains efficiency if a reused sample's PDF better approximates the target integrand. Due to iterative use of RIS, in such cases, using $\hat{p} = f$ may be reasonable, especially for complex paths (e.g., Lin et al. [2021]).

Talbot's RIS theory assumes independent samples X_i lying in a shared domain Ω. ReSTIR stretches these assumptions, so it may not retain *any* theoretical convergence guarantees. In fact, with seemingly innocuous algorithmic modifications, correlated reuse can cause convergence to a wrong result.

4 GENERALIZED RIS

Our generalized resampled importance sampling (GRIS) allows mapping samples between domains and identifies the constraints for which this is unbiased and converges.

Unlike traditional RIS, which selects from independent samples in one domain, we allow potentially correlated inputs $(X_i)_{i=1}^{M}$ from different domains Ω_i. Generalized RIS randomly selects sample X_s and maps it to f's domain Ω via a *shift mapping*, $Y = T_s(X_s)$, so that the PDF of Y approaches target $\bar{p}$ (i.e., a normalized $\hat{p}$).

4.1 Overview

Before delving into theoretical details for our generalization, we briefly overview our approach and relate it to traditional RIS.

We assume input samples X_i, perhaps from varying domains Ω_i, need not be independent and are paired with unbiased contribution weights $W_i \in \mathbb{R}$ that can replace $1/p_i(X_i)$ for integration. This explicitly allows prior resampled inputs; while a resampled input X_i has an intractable PDF p_i, its weight W_i *is* tractable (i.e., Equation 2). We formalize unbiased contribution weights in Section 4.2.

To reuse samples to integrate f over Ω, we must map our random samples $X_s \in \Omega_s$ into Ω with a shift mapping $T_s : \Omega_s \to \Omega$. This shift modifies the PDF via the PDF transformation laws,[3⋆] requiring the shift map's Jacobian determinant, $|\partial T_i/\partial x|$. We formalize shift mappings in Section 4.3.

Algorithmically, this changes various aspects of RIS (see Figure 3, green column). We must transform samples to a common domain Ω, so resampling weights include shift maps T_i and their determinants:

$$w_i = m_i(T_i(X_i))\,\hat{p}(T_i(X_i))\,W_i \cdot |\partial T_i/\partial X_i|\ . \tag{7}$$

We do not require tractable p_i; we may use $W_i = 1/p_i(X_i)$, but we may also use numerical contribution weights W_i from e.g., a prior RIS pass (Equation 2). Before using the selected sample for integration (or further resampling), we must shift it to the appropriate domain, i.e., our output sample is $Y = T_s(X_s)$.

Unbiased contribution weights W_Y for output Y are again given by Equation 2. With the constraints we derive below, p_Y converges to $\bar{p}$ such that $\mathrm{Var}[f(Y)W_Y]$ is guaranteed to approach $\mathrm{Var}[f(Y)/\bar{p}(Y)]$. This achieves asymptotic zero-variance integration with a single Y if $\hat{p} \propto f$.

4.2 Unbiased Integration with Generalized RIS

Again, we assume potentially correlated input samples $(X_i \in \Omega_i)_{i=1}^{M}$ with arbitrary source domains Ω_i. Furthermore, samples X_i must be paired with unbiased contribution weights W_i, acting as replacements for potentially intractable reciprocal PDFs $1/p_i(X_i)$.

We first derive an unbiased integrator for function f over domain Ω, assuming almost arbitrary resampling weights w_i. In Section 4.4, we replace these arbitrary w_i with weights that lead to asymptotic convergence to the target PDF $\bar{p}$.

We formally define unbiased contribution weights W_i as follows:

Definition 4.1. An *unbiased contribution weight* $W \in \mathbb{R}$ for a random variable $X \in \Omega$ is any real-valued random variable W for which

$$\mathbb{E}\,[f(X)W] = \int_{\mathrm{supp}(X)} f(x)\,\mathrm{d}x \tag{8}$$

for any integrable function $f : \Omega \to \mathbb{R}$.

The expression $f(X)W$ generalizes the ratio $f(X)/p(X)$ in Monte Carlo integration: if p is tractable, we can use $W = 1/p(X)$. If not, as when picking X with RIS, these weights still allow unbiased integration. The integral is naturally limited to where $p > 0$, i.e., $\mathrm{supp}(X)$. Similar definitions have been used outside computer graphics, e.g., Liu and Liu [2001] and Liang and Cheon [2009]. Apart from the

[3⋆] With $y = T(x)$, we have $P_Y(y) = \left|\frac{\partial P}{\partial y}\right| = \left|\frac{\partial P}{\partial x}\right|\left|\frac{\partial x}{\partial y}\right| = P_X(x)\left|\frac{\partial T}{\partial x}\right|^{-1}$.

reciprocal PDF estimates in RIS, the conditional PDFs in the continuous multiple importance sampling framework [West et al. 2020] are also examples of unbiased contribution weights.

Unbiased contribution weights naturally replace the reciprocal of the marginal PDF; in fact, they unbiasedly estimate it,

$$\mathbb{E}\left[W \mid X\right] = \frac{1}{p_X(X)}. \tag{9}$$

This is not coincidence, but equivalence. Any unbiased estimator for the reciprocal marginal PDF (Equation 9) is an unbiased contribution weight (Equation 8) and vice versa (Theorem A.1).

In RIS, we resample X_i proportionally to w_i. We need to express the contribution of chosen sample X_s that gives an unbiased estimate for the integral of f. To do this, we start by assigning each sample X_i a corresponding *contribution function* $g_i : \Omega_i \to \mathbb{R}$ that gets evaluated if selecting index $s = i$.

We then look at the expectation of $g_s(X_s)W_s$ divided by the RIS selection probability of index s. The PMF of the selection index is $p_s(i) = w_i / \sum_{j=1}^{M} w_j$, and with some caution,[4★] we get

$$\begin{aligned}\mathbb{E}\left[\frac{g_s(X_s)W_s}{p_s(s)}\right] &= \mathbb{E}\left[\sum_{i=1}^{M} g_i(X_i)\frac{\cancel{p_s(i)}}{\cancel{p_s(i)}}W_i\right] = \\ &= \sum_{i=1}^{M}\int_{\mathrm{supp}(X_i)} g_i(x_i)\,\mathrm{d}x_i,\end{aligned} \tag{10}$$

The first step expands the expectation as a sum over the possible cases, and the second step utilizes the definition of unbiased contribution weights to transform the sum of expectations into a sum of integrals. RIS naturally skips sampling areas where the random variables have zero PDF, limiting integration to the supports of X_i. Equipped with this result, we can now proceed to transform the remaining sum of integrals into the desired integral of f by carefully choosing unknowns g_i.

Choosing g_i so the right-hand-side becomes the integral of f yields an unbiased contribution for the selected sample $Y = X_s$. In the special case that X_i are all from the same domain Ω and support S, and all w_i are positive in S, we can recover basic RIS by choosing $g_i = \frac{1}{M} f$ for all i, giving:

$$\mathbb{E}\left[\frac{1}{M}f(Y)\frac{\sum_{j=1}^{M} w_j}{w_s}W_s\right] = \int_{\mathrm{supp}(Y)} f(x)\,\mathrm{d}x. \tag{11}$$

Comparing to Equation 8, we observe that in this restricted case the expectation is of form $\mathbb{E}[f(Y)W_Y]$ with

$$W_Y = \frac{1}{M}\frac{\sum_{j=1}^{M} w_j}{w_s}W_s, \tag{12}$$

making W_Y an unbiased contribution weight for Y, i.e., $\mathbb{E}[f(Y)W_Y]$ integrates any function f over the support of Y. In Section 4.3, we extend our result to samples X_i coming from multiple domains Ω_i.

Degenerate case. If all w_i are 0, no sample is selected and the contribution is zero. Intuitively, one may think of returning a zero-contribution null-sample $Y_\emptyset$ outside the sampling and integration domains (i.e., $\hat{p}(Y_\emptyset) = f(Y_\emptyset) = 0$). The value of $W_{Y_\emptyset}$ is then irrelevant, and can be set to zero.

4.3 Shift Mapping

In GRIS, samples X_i may originate from arbitrary domains Ω_i. To integrate $f : \Omega \to \mathbb{R}$ with samples $X_i \in \Omega_i$, we must transform the right-hand side of Equation 10 into the integral of f. To do this, we choose g_i that map X_i from Ω_i to Ω and evaluate f at the result.

Since a map from Ω_i to Ω changes the variables of integration, it must be bijective. Mappings between complicated domains can be non-trivial to construct, so we settle for a bijection from a subset of Ω_i to its image in Ω. As in prior work (e.g., Manzi et al. [2014]), we call such bijections *shift mappings*, T_i, and associate one with each domain Ω_i.

Definition 4.2. A *shift mapping* T_i from Ω_i to Ω is a bijective function from a subset $\mathcal{D}(T_i) \subset \Omega_i$ to its image $\mathcal{I}(T_i) \subset \Omega$.

Intuitively, we should choose contribution functions

$$g_i(x) = c_i(y_i)\, f(y_i) \left|\frac{\partial T_i}{\partial x}\right|, \tag{13}$$

where y_i is shorthand for $T_i(x)$, *contribution MIS* weights $c_i : \Omega \to \mathbb{R}$ are an arbitrary partition of unity $\sum_{i=1}^{M} c_i(y) = 1$ for $y \in \Omega$, and $\left|\frac{\partial T_i}{\partial x}\right|$ is the Jacobian determinant of $x \mapsto y_i$. In principle, this implements

$$\sum_{i=1}^{M}\int_{\Omega_i} g_i(x)\,\mathrm{d}x = \int_{\Omega} f(x)\,\mathrm{d}x, \tag{14}$$

but care is required in the details; e.g., Equation 13 is not defined for $x \notin \mathcal{D}(T_i)$. We fix this by defining $g_i(x) = 0$ for $x \notin \mathcal{D}(T_i)$ and updating the contribution MIS weights c_i to compensate.

We assume weights w_i are arbitrary non-negative random variables related to target function $\hat{p}$ as follows: $w_i > 0$ iff $X_i \in \mathcal{D}(T_i)$ and $\hat{p}(Y_i) > 0$. Essentially, $w_i > 0$ when $Y_i = T_i(X_i)$ exists and is in the support of $\hat{p}$, otherwise $w_i = 0$ to avoid choosing X_i. Later, we slightly relax this constraint.

Under these assumptions, each possible Y must be in $\mathrm{supp}\,\hat{p}$ and be samplable as $Y = T_i(X_i)$ by one or more X_i that has positive PDF (i.e., $X_i \in \mathrm{supp}\,X_i$), and vice versa. Mathematically,[5★]

$$\mathrm{supp}\,Y = \mathrm{supp}\,\hat{p} \cap \bigcup_{i=1}^{M} T_i(\mathrm{supp}\,X_i). \tag{15}$$

This implies $\mathrm{supp}\,Y \subset \mathrm{supp}\,\hat{p}$. Later, we assume $\mathrm{supp}\,\hat{p} \subset \mathrm{supp}\,Y$, which also implies $\mathrm{supp}\,Y = \mathrm{supp}\,\hat{p}$.

Carefully substituting the above g_i, with $g_i(x) = 0$ if $x \notin \mathcal{D}(T_i)$, into the left-hand side of Equation 10, this gives the equality

$$\mathbb{E}\left[c_s(Y)f(Y)\left|\frac{\partial T_s}{\partial X_s}\right|\frac{\sum_{j=1}^{M} w_j}{w_s}W_s\right] = \int_{\mathrm{supp}(Y)} f(x)\,\mathrm{d}x, \tag{16}$$

where W_s is the unbiased contribution weight of X_s.

[4★] Technically, this equation requires that $w_i > 0$ whenever $g_i(X_i) \neq 0$, but we will later introduce a partition of unity that lifts this requirement.

[5★] If $\mathrm{supp}\,X_i$ is larger than the domain of T_i, we set $T_i(\mathrm{supp}\,X_i) = T_i(\mathrm{supp}\,X_i \cap \mathcal{D}(T_i))$.

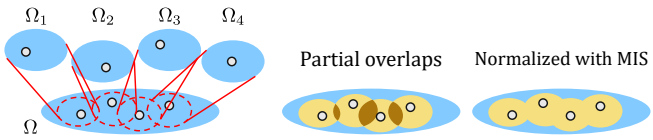


Fig. 4. *Unbiased integration with candidate samples from multiple domains. The domain would be covered unevenly without contribution MIS weights c_i. Similar weighting is used for sample selection probabilities with resampling MIS weights m_i in Section 4.4.*

The constraints that contribution MIS weights c_i must fulfill for this to hold are that for all $y \in \operatorname{supp} Y$,

$$\sum_{\substack{i=1 \\ y \in T_i(\operatorname{supp} X_i)}}^{M} c_i(y) = 1. \tag{17}$$

Interpret this as: every realizable y, possibly from multiple Ω_i, must be covered exactly once in total (see Figure 4 for an illustration). The summation only accounts for domains Ω_i from which y can be realized as $y = T_i(x_i)$ with non-zero PDF. In principle, negative values of c_i work, but later we find that only $c_i \geq 0$ allow chaining multiple passes of GRIS.

Again, the expectation in Equation 16 is of the form $\mathbb{E}[f(Y)W_Y]$ for arbitrary integrable function f in Ω, and the right-hand side integrates f over the support of Y, per the definition of unbiased contribution weights in Equation 8. This means that with

$$W_Y = c_s(Y)\left(W_s \left|\frac{\partial T_s}{\partial X_s}\right|\right)\left[\frac{\sum_{j=1}^{M} w_j}{w_s}\right], \tag{18}$$

$f(Y)W_Y$ unbiasedly estimates the integral of f over the support of Y, and that W_Y is an unbiased estimate for $1/p_Y(Y)$.

This specifies when generalized RIS can integrate an arbitrary function f: when the supports of random variates X_i (mapped to Ω via T_i) together cover the support of f.

This is automatically fulfilled if we choose one sampling domain, say Ω_1, as f's domain, use the identity shift $T_1(x) = x$ on Ω_1, and generate X_1 with an importance sampler designed for the integrand f, i.e., so that $p(x_1) > 0$ whenever $f(x_1) > 0$; we will later call such samples *canonical*. Since p_{X_1} is known, we can use the unbiased contribution weight $W_1 = 1/p_{X_1}(X_1)$.

We later show earlier ReSTIR samplers can, a posteriori, be built on these observations, but first Section 4.4 covers how to realize convergence to target density $\bar{p}$ by setting w_i with Equation 19. The corresponding W_Y is then given by Equation 22.

Relaxing constraints ⋆. The condition that $w_i > 0$ when $\hat{p}(Y_i) > 0$, for $Y_i = T_i(X_i)$, can be relaxed by also allowing $w_i = 0$ when $c_i(Y_i) = 0$ or $W_i = 0$, i.e., when the expectation does not change. The validity of Equation 17 must be explicitly guaranteed in $\operatorname{supp} \hat{p} \cap \bigcup_i T_i(\operatorname{supp} X_i)$ to make Equation 15 hold. We derive these constraints and the unbiasedness of the estimator in Section S.7.1. Using the w_i from the next section removes the need for these constraints.

4.4 Asymptotically Perfect Importance Sampling ⋆

Above, we generalized RIS to multiple domains for unbiased integration with near-arbitrary weights. Like Talbot's RIS, the goal of GRIS is producing samples following a desired distribution; we want the marginal probability density p_Y of output sample Y to converge to $\bar{p}$ as the input sample count approaches infinity.

We show this occurs with the following resampling weights:

$$w_i = \begin{cases} m_i(T_i(X_i))\,\hat{p}(T_i(X_i))\,W_i \cdot \left|\frac{\partial T_i}{\partial X_i}\right|, & \text{if } X_i \in \mathcal{D}(T_i) \\ 0, & \text{otherwise} \end{cases}, \tag{19}$$

given resampling MIS weights m_i and unbiased contribution weights W_i. As normalizing w_i gives resampling probabilities, w_i must be non-negative. It follows that m_i and W_i must also be non-negative (proofs in Section S.7.2), which we assume hereafter.

Weight w_i will be zero outside of $\operatorname{supp} \hat{p}$, as $\hat{p} = 0$. Requirements for m_i are similar to those for c_i. For all y in $\operatorname{supp} Y$,

$$\sum_{\substack{i=1 \\ y \in T_i(\operatorname{supp} X_i)}}^{M} m_i(y) = 1, \tag{20}$$

but we also require $m_i \geq 0$. The sum only includes indices that can generate y with a positive PDF. Unbiased integration also requires $m_i(y) > 0$ whenever $c_i(y) \neq 0$ so the m_i do not invalidate the partition of unity formed by c_i (proof in Section S.7.3).

Directly substituting w_i from Equation 19 into Equation 18 yields the unbiased contribution weight for new sample Y,

$$W_Y = \left[\frac{c_s(Y)}{m_s(Y)}\right]\frac{1}{\hat{p}(Y)}\sum_{j=1}^{M} w_j. \tag{21}$$

The condition that $m_i(y) > 0$ when $c_i(y) \neq 0$ now naturally avoids division by zero. Next, we show that choosing $m_i = c_i$ is ideal, naturally fulfilling this requirement.

W_Y in Equation 21 has multiple sources of variance. The sum $\sum_{j=1}^{M} w_j$ varies with inputs X_j and the ratio $c_s(Y)/m_s(Y)$ varies with index s. Asymptotically approaching the desired sample density $\bar{p}$ requires the sum variance $\operatorname{Var}[\sum_{j=1}^{M} w_j]$ to approach zero. Even if fixing this sum as a constant, ratio $c_s(Y)/m_s(Y)$ can add significant variance. This disappears by selecting $c_i(y) = m_i(y)$.

Our improved and simpler unbiased contribution weight for GRIS becomes

$$W_Y = \frac{1}{\hat{p}(Y)}\sum_{j=1}^{M} w_j, \tag{22}$$

which we used to reformulate traditional RIS (Equation 2) to prepare for this generalization. We use this expression for W_Y hereafter.[6]

This allows deriving Theorem A.2 in Appendix A, which guarantees asymptotic convergence of p_Y, the PDF of resampled Y, to $\bar{p}$: consider the behavior of a sequence of resampling results Y_M (with $\operatorname{supp} \hat{p} \subset \operatorname{supp} Y_M$) as M increases. If variance of the summed

[6] The unbiased contribution weights in Equation 22 skip the division by M often seen in RIS and ReSTIR formulas, as our weights w_j already include this factor in the resampling MIS weights m_i (Equation 19); selecting $m_i = 1/M$ gives the prior formulations.

resampling weights goes to zero,

$$\mathrm{Var}\left[\sum_{i=1}^{M} w_{M,i}\right] \xrightarrow{M\to\infty} 0, \tag{23}$$

then $\bar{p}(Y)/p_Y(Y)$ converges to 1 in the mean-square sense.[7⋆] This sum approximates the integral of $\hat{p}$:

$$\mathbb{E}\left[\sum_{i=1}^{M} w_i\right] = \mathbb{E}\left[\hat{p}(Y)W_Y\right] = \int_{\mathrm{supp}\,Y} \hat{p}(y)\,\mathrm{d}y = \|\hat{p}\|. \tag{24}$$

Increasing input sample count M adds more terms to the sum; this tends to make each w_i smaller. Convergence of p_Y to $\bar{p}$ is subject to $\sum_{i=1}^{M} w_i$ approaching $\|\hat{p}\|$ as $\mathrm{Var}\left[\sum_{i=1}^{M} w_i\right] \to 0$, i.e.,

$$W_Y = \frac{1}{\hat{p}(Y)}\sum_{j=1}^{M} w_j \approx \frac{\|\hat{p}\|}{\hat{p}(Y)} = \frac{1}{\bar{p}(Y)} \quad \text{for large } M\,. \tag{25}$$

The guarantee from Equation 23 is quite strong. While convergence of p_Y to $\bar{p}$ may not be pointwise (new samples may introduce temporary fluctuations), the probability of errors of any given size approaches zero, and each subset of Ω will, asymptotically, receive the correct ratio of samples.

In addition, as shown in Section 5.2, integration variance also goes to zero (if $\hat{p} \propto f$), i.e., in the limit we get the variance expected if Y were *exactly* distributed with target PDF $\bar{p}$.

5 CONVERGENCE AND VARIANCE ANALYSIS ⋆

Above we presented a new GRIS theory and conditions for asymptotic convergence to a target distribution, but we have yet to discuss its asymptotic behavior as an integral estimator, particularly for Monte Carlo sampling. As infinite sample counts are impractical, we also want to analyze variance when using finitely many samples.

5.1 Reasonable Distributions ⋆

Before studying variance, we start by formally defining a *reasonable* importance sampling distribution:

Definition 5.1 (Reasonable distribution)**.** We say a PDF p is a reasonable importance sampling distribution for a non-negative function f (or p is reasonable for [integrating] f) if a bound C_f exists such that

$$f(x) \le C_f\, p(x) \quad \text{for all } x. \tag{26}$$

We also say a random variate X with unbiased contribution weight W_X is reasonably distributed for f, if there exists a bound C_f

$$f(X)W_X \le C_f \quad \text{with probability 1.} \tag{27}$$

Essentially, a reasonable distribution guarantees bounded Monte Carlo contributions. In standard Monte Carlo, $f(X)/p(X) \le C_f$, and for unbiased contribution weights, $f(X)W_X \le C_f$.

[7⋆] Mathematically, this means $\mathbb{E}\left[\left|\frac{\bar{p}(Y_M)}{p_Y(Y_M)} - 1\right|^2\right] \xrightarrow{M\to\infty} 0$.

5.2 Asymptotic Variance of Integral Estimation ⋆

Asymptotic convergence of distribution p_Y naturally gets reflected in integration variance. Assuming $\bar{p}$ is a reasonable distribution for function f, then the unbiased integral estimate $f(Y)W_Y$ asymptotically has variance due only to any mismatch of $\bar{p}$ and f. If $\hat{p}$ is chosen proportional to f, then $\bar{p} \propto f$ and estimate $f(Y)W_Y$ will be asymptotically zero-variance.

We formalize this in Theorem A.3. Convergence of p_Y to $\bar{p}$ is provided by Theorem A.2, and if $0 \le f \le C_f\,\hat{p}$ for some $C_f > 0$, then

$$\mathrm{Var}\left[f(Y)W_Y\right] \xrightarrow{M\to\infty} \mathrm{Var}\left[\frac{f(X)}{\bar{p}(X)}\right], \tag{28}$$

where X has density $\bar{p}$. The zero-variance result follows naturally if $f(x)/\bar{p}(x)$ is constant.

A key takeaway is that if $\hat{p}$ is not proportional to f, increasing the input sample count eventually leads to diminishing returns; further variance reduction requires choosing a $\hat{p}$ better matching f.

5.3 Variance in the Finite Case ⋆

Above, we studied the asymptotic behavior of GRIS as sample count increases without bound. In practice, we are limited to finite M, so we aim to minimize variance in some computation budget. Fortunately, we may give explicit variance bounds for our integral estimate:

Theorem 1. *With the assumptions of Theorem A.3,*

$$\mathrm{Var}\left[f(Y)W_Y\right] \le \mathrm{Var}\left[\frac{f(X)}{\bar{p}(X)}\right] + b, \tag{29}$$

where X is distributed with density $\bar{p}$ and

$$b = C_f^2\sqrt{\mathrm{Var}\left[\sum_{i=1}^{M} w_i\right]}\left(\|\hat{p}\| + 2\sqrt{\mathrm{Var}\left[\sum_{i=1}^{M} w_i\right]}\right). \tag{30}$$

Proof. Section S.5.4. □

Here, C_f is the bounding constant for a reasonable distribution $\hat{p}$ for f. Theorem 1 says resampling converges to $\bar{p}$ by decreasing $\mathrm{Var}[\sum_{i=1}^{M} w_i]$, which acts as a concrete proxy for the current convergence state. As $\mathrm{Var}[\sum_{i=1}^{M} w_i]$ approaches zero, remaining variance stems from potential mismatches between $\hat{p}$ and f.

The law of total variance provides another decomposition of variance; applied to $f(Y)W_Y$, we get

$$\mathrm{Var}[f(Y)W_Y] = \mathrm{Var}\left[\frac{f(Y)}{p_Y(Y)}\right] + \mathbb{E}\left[f(Y)^2\,\mathrm{Var}\left[W_Y \mid Y\right]\right], \tag{31}$$

which says variance has two sources: mismatches between f and marginal density p_Y, and $\mathrm{Var}\left[W_Y \mid Y\right]$, the mean squared deviation of W_Y from its conditional expectation $1/\bar{p}(Y)$. Intuitively, if W_Y approaches $1/\bar{p}(Y)$ such that $\mathrm{Var}[W_Y \mid Y]$ goes to zero, naturally $\mathrm{Var}[f(Y)W_Y]$ also approaches $\mathrm{Var}[f(Y)/\bar{p}(Y)]$.

For the special case of equality $f(x) = C_f\hat{p}(x)$, we can derive the exact variance (using Equation 22),

$$\mathrm{Var}\left[f(Y)W_Y\right] = \mathrm{Var}\left[\frac{f(Y)}{\hat{p}(Y)}\sum_{i=1}^{M} w_i\right] = C_f^2\,\mathrm{Var}\left[\sum_{i=1}^{M} w_i\right]. \tag{32}$$

Independent of $\hat{p}$'s proportionality to f, our analysis shows the importance of reducing $\mathrm{Var}[\sum_i w_i]$ given finite samples. In practice, we can minimize variance by making w_i more uniform. In particular, preventing singularities in w_i avoids unbounded variance.

5.4 Avoiding Singularities ⋆

Generalized RIS does not automatically remove singularities from Monte Carlo integration. As usual, avoiding large outliers requires additional guarantees. More concretely, if $\sum_{i=1}^{M} w_i$ is unbounded, the contribution

$$f(Y)W_Y = \frac{f(Y)}{\hat{p}(Y)} \sum_{i=1}^{M} w_i \tag{33}$$

can also be unbounded.

Clearly, very large w_i are detrimental to our goal of bringing $\mathrm{Var}[\sum_{i=1}^{M} w_i]$ to zero; we should aim to make the weight sum,

$$\sum_{i=1}^{M} w_i = \sum_{\substack{i=1 \\ X_i \in \mathcal{D}(T_i)}}^{M} m_i(T_i(X_i)) \cdot \hat{p}(T_i(X_i))W_i \cdot \left|\frac{\partial T_i}{\partial X_i}\right|, \tag{34}$$

as uniform as possible. Going through the terms, we identify potential challenges to maintaining uniformity:

(1) Some samples y may be reachable via only finitely many T_i even in the limit
(2) Resampling MIS weights m_i can greatly exceed $1/M$
(3) The product $\hat{p}(T_i(X_i))W_i$ can be unbounded
(4) Jacobians can be unbounded

These can be tackled one-by-one, e.g., (1) adding samples from all source domains when increasing M, and (4) modifying shift map domains to cut off extreme Jacobians (while maintaining bijectivity).

Instead, we simultaneously solve all four by designing a suitable sampling scheme and two robust MIS weight families. This guarantees bounded contributions, asymptotic convergence to $\hat{p}$, and realizes asymptotic zero-variance integration with samples from multiple domains.

5.5 Canonical Samples ⋆

Designing MIS weights for samples arising from multiple strategies normally requires knowing PDFs for samples with all strategies. Resampling gives up access to such PDFs. Instead, we design robust MIS weights by assuming samples X_i are associated with non-negative unnormalized target distributions $\hat{p}_i$, much like $\hat{p}$, that act as proxies for p_{X_i}.

RIS, and our generalization, conceptually need an infinite stream of random samples to asymptotically converge to desired distribution $\bar{p}$. If a subset of $\operatorname{supp}\hat{p}$ is covered by this stream only finitely many times, p_Y can not generally converge to $\bar{p}$ in this subset.

Sometimes, as in light transport, covering the support of $\hat{p}$ with samples X_i from *other* domains Ω_i may be challenging. By taking samples from an importance sampler that directly targets $\hat{p}$, we can cover $\operatorname{supp}\hat{p}$ as many times as needed for convergence. We define samples X_i that directly target $\hat{p}$ in Ω with the identity shift map and $\hat{p}_i = \hat{p}$ to be *canonical*, motivated by Bitterli [2021], and present the following mathematical definition:

Definition 5.2 (Canonical Sample)**.** An input sample $X_i \in \Omega_i$ is *canonical* if its domain is Ω, it uses the identity shift map $T_i(x) = x$, uses $\hat{p}_i = \hat{p}$, and covers $\operatorname{supp}\hat{p}$ (i.e., $\operatorname{supp}\hat{p} \subset \operatorname{supp}X_i$).

We denote the set of indices of canonical samples in $1, \ldots, M$ by R and their number by $|R|$. Later, we find that if canonical sample count increases sufficiently as the total input count increases, the MIS weights in Section 5.6 guarantee asymptotic convergence of p_Y to $\bar{p}$ when resampling from multiple domains.

5.6 Designing Robust MIS Weights ⋆

As motivated in Section 5.4, we design resampling MIS weights to guarantee bounded contribution for the chosen sample Y, assuming input samples X_i are reasonably distributed for target functions $\hat{p}_i$. To simplify the derivation, we define a new symbol, "$\hat{p}$ from i",

$$\hat{p}_{\leftarrow i}(y) = \begin{cases} \hat{p}_i\left(T_i^{-1}(y)\right)\left|\partial T_i^{-1}/\partial Y_i\right|, & \text{if } y \in T_i(\operatorname{supp}X_i) \\ 0 & \text{otherwise} \end{cases}, \tag{35}$$

i.e. for the sample y, evaluate its proxy PDF $\hat{p}_i$ at the sample location x in the original domain Ω_i, multiplied by the Jacobian determinant of the shift.

We aim to bound the resampling weights w_i and construct two families of MIS weights that guarantee this. We then derive their upper bounds, which decrease with additional canonical samples. Later, we utilize these bounds to guarantee convergence of p_Y to $\bar{p}$.

Generalized Talbot MIS. The first family, which we derive in Section S.1, generalizes the weights of Talbot [2005] into the following:

$$m_i(y) = \frac{\hat{p}_{\leftarrow i}(y)}{\sum_{j=1}^{M} \hat{p}_{\leftarrow j}(y)}. \tag{36}$$

Talbot's [2005] form is obtained by assuming independent samples over one domain ($\Omega_i = \Omega$, $T_i(x) = x$) and using exact PDFs p_k in place of $\hat{p}_{\leftarrow k}$. This MIS family is analogous to the balance heuristic [Veach 1998] between possible sources of sample Y.

Generalized pairwise MIS. The second family of MIS weights, derived in Section S.1, generalizes Bitterli's [2021] pairwise MIS, originally given for a single canonical sample and domain ($|R| = 1$, $\Omega_i = \Omega, T_i(x) = x$) for the *defensive* variant below. The key benefit of pairwise MIS is a significant cost reduction from $O(M^2)$ to $O(M\,|R|)$. This comes from restricting application of MIS to individual pairs of target functions, each involving only the target $\hat{p}$ and the source $\hat{p}_{\leftarrow i}$, if i is not a canonical sample, and an average of MIS between pairs $(\hat{p}, \hat{p}_{\leftarrow j})$ otherwise. We discuss the generalized pairwise MIS family more in Section S.1, but present here the *uniform* variant, which gives all inputs equal weight if they have the same $\hat{p}_{(\leftarrow)}$ values,

$$m_i(y) = \begin{cases} \frac{1}{M-|R|} \sum_{j \notin R} \frac{\hat{p}(y)}{|R|\hat{p}(y)+(M-|R|)\hat{p}_{\leftarrow j}(y)}, & \text{if } i \in R \\ \frac{\hat{p}_{\leftarrow i}(y)}{|R|\hat{p}(y)+(M-|R|)\hat{p}_{\leftarrow i}(y)}, & \text{if } i \notin R \end{cases}, \tag{37}$$

and a slightly less efficient but often more robust *defensive* variant, which always gives canonical samples higher MIS weights than

non-canonical samples,

$$m_i(y) = \begin{cases} \frac{1}{M} + \frac{1}{M} \sum_{j \notin R} \frac{\hat{p}(y)}{|R|\hat{p}(y)+(M-|R|)\hat{p}_{\leftarrow j}(y)}, & \text{if } i \in R \\ \frac{M-|R|}{M} \frac{\hat{p}_{\leftarrow i}(y)}{|R|\hat{p}(y)+(M-|R|)\hat{p}_{\leftarrow i}(y)}, & \text{if } i \notin R \end{cases}. \quad (38)$$

Resampling weight bounds. With these definitions, we can guarantee resampling weights w_i stay bounded (Theorem A.4): If the m_i are given by Equation 36, 37 or 38, and a sample X_i is reasonably distributed for integrating $\hat{p}_i$, i.e., $\hat{p}_i(X_i)W_i \le C_i$ for some C_i, then the resampling weight of X_i is bounded as

$$w_i \le \frac{C_i}{|R|}. \quad (39)$$

The condition that $\hat{p}_i(X_i)W_i$ is bounded is equivalent to a bounded relative error of W_i from its ideal value $1/\bar{p}_i(X_i)$. Starting from independent samples, we can guarantee this inductively:

Bounded variance. If we independently sample X_i with a reasonable importance sampling strategy for target function $\hat{p}_i$ (i.e., $\hat{p}_i(X_i) \le C_i\, p_i(X_i)$), then $\hat{p}_i(X_i)W_i = \hat{p}_i(X_i)/p_i(X_i) \le C_i$, and Equation 39 applies to X_i.

If all input samples X_i fulfill $\hat{p}_i(X_i)W_i \le C_i$ for constants C_i, then

$$\hat{p}(Y)W_Y = \sum_{i=1}^{M} w_i \le \sum_{i=1}^{M} \frac{C_i}{|R|}, \quad (40)$$

and Equation 39 applies to Y, since $\hat{p}(Y)W_Y$ is bounded.

Inductively, chaining GRIS by starting from independent samples retains a finite worst-case resampling weight sum and a finite worst-case contribution $f(Y)W_Y$, assuming $f/\hat{p}$ is bounded.

A bounded random variable like $f(Y)W_Y$ features finite variance. Further, averaging such variables converges to the expectation. This important property is not automatically guaranteed by the resampling MIS weights earlier ReSTIR work implicitly used, as $m_i = 1/M$ often fails to properly partition unity, and additionally may not account for the singularities in Equation 33.

Constant resampling weights. Constant resampling MIS weights $m_i(y) = 1/M$ are very cheap to evaluate. But such MIS weights only sometimes fulfill the constraints of m_i (Equation 20), i.e., when any realizable sample X_i could have been sampled from all other domains Ω_j with positive PDF.[8] This is generally not true, but an important exception is when all the input samples are canonical. This can be achieved e.g., by producing the input samples with GRIS that uses at least one canonical sample each time, and one of the resampling MIS schemes above. If constant weights are still used in the general case, convergence to $\bar{p}$ is lost along with our convergence and variance results. Bias can still be removed by using proper contribution MIS, as proposed by Bitterli et al. [2020].

Tractable PDFs. If all input samples have known, tractable PDFs (e.g., X_i come from importance samplers with known PDFs p_i), the generalized pairwise and Talbot MIS weights can be modified to use p_i instead of $\hat{p}_i$, with instances of $\hat{p}$ replaced with PDF p_c of a fixed canonical sample X_c. The canonical samples must have a PDF reasonable for integrating $\hat{p}$. See Section S.1.4 for more information.

[8] This alone is not generally enough to guarantee finite variance.

5.7 Guaranteeing Convergence ⋆

So far, we showed GRIS achieves asymptotic convergence of p_Y to $\bar{p}$ simply by requiring $\mathrm{Var}\left[\sum_{i=1}^{M} w_i\right] \to 0$. In this section we show how to guarantee convergence in a direct application of generalized RIS theory. Section 6 extends this analysis to multi-pass algorithms that guarantee convergence in a streaming manner, requiring only finite memory and amortizing computation between multiple integrals.

Independent samples. We assume the case of multiple domains with robust resampling MIS weights (Section 5.6), applying Theorem A.4 to obtain a bound $w_i \le C_i/|R|$ on the resampling weights. If we also assume that pairs (X_i, W_i) are independent, then the w_i are independent, and $\mathrm{Var}\left[\sum_{i=1}^{M} w_i\right] = \sum_{i=1}^{M} \mathrm{Var}\left[w_i\right]$. We bound the variances by Popoviciu's inequality as

$$\sum_{i=1}^{M} \mathrm{Var}\left[w_i\right] \le \sum_{i=1}^{M} \frac{1}{4} \frac{C_i^2}{|R|^2}, \quad (41)$$

which converges to zero if $|R|$ grows fast enough compared to M and C_i. A practical constraint asserts the importance sampling quality of additional samples does not grow worse without bound, i.e., there exists an upper bound C such that $C_i \le C$ for all i. Then we get $\mathrm{Var}\left[\sum_{i=1}^{M} w_i\right] \le C^2 \frac{M}{4|R|^2}$, guaranteeing convergence to zero if $|R|$ grows faster than $\sqrt{M}$ such that $|R|/\sqrt{M} \to \infty$. For example, $|R| \approx c\, M^{0.5001}$ for some $c > 0$ converges (slowly) in the limit. But more practically, we may ensure the ratio of canonical samples, $|R|/M$, never falls below some constant $\gamma > 0$ for large enough M; this guarantees a worst-case convergence rate of $O(1/M)$ in terms of variance.

Dependent samples. Our GRIS theory does not assume sample independence; convergence and variance results only assume that $\mathrm{Var}\left[\sum_{i=1}^{M} w_i\right] \to 0$ is true. For independent samples, this constraint is easy to prove. For dependent samples, this constraint may not be true. An easy counter-example uses duplicate samples X_i; no variance reduction can occur with increased sample count.

We still get convergence if sample correlation is weak enough. Assume the case of $|R|/M \ge \gamma$ and $w_i \le C/|R|$ for all i (e.g., a single-domain with $m_i = 1/M$,[9⋆] or multi-domains with our novel m_i). While this may not converge generally, we can guarantee convergence by assuming correlations between resampling weights w_i and w_{i+k} tend to zero as $k \to \infty$. More precisely, we assume there exists a non-negative sequence b_k such that regardless of i, the correlation $\rho_{i,i+k} \le b_k$, and $b_k \to 0$. Then, we can manipulate

$$\mathrm{Var}\left[\sum_{i=1}^{M} w_i\right] = \sum_{i=1}^{M} \mathrm{Var}\left[w_i\right] + 2 \sum_{i=1}^{M} \sum_{k=1}^{M-i} \mathrm{Cov}(w_i, w_{i+k}), \quad (42)$$

where the first term converges to zero by the argument following Equation 41, and for the second term we derive in Section S.2

$$\sum_{i=1}^{M} \sum_{k=1}^{M-i} \mathrm{Cov}(w_i, w_{i+k}) \le \frac{C^2}{4\gamma^2} \left(\frac{1}{M} \sum_{k=1}^{M} b_k\right) \xrightarrow{M \to \infty} 0, \quad (43)$$

[9⋆] If $\hat{p}(X_i)W_i \le C$ for a single-domain with constant MIS, then $w_i = \hat{p}(X_i)W_i/M \le C/M = C/|R|$ if all samples are canonical.

where γ is the minimum ratio of $|R|/M$ and C is an upper bound for all the C_i. The mean of b_k converges to zero since b_k converges to zero, and we get convergence with dependent samples.

Section S.2 also contains a generalized result allowing the ratio $|R|/M$ to decrease as M grows if the maximum correlation b_k falls quickly enough to compensate. For example, if $\sum_{k=1}^{\infty} b_k < \infty$, we can guarantee convergence with $|R| \geq c \cdot M^{0.5001}$ for some $c > 0$.

6 AMORTIZATION OVER AN IMAGE WITH RESTIR

Here, we reformulate two ReSTIR variants on top of generalized RIS: a novel progressive offline renderer and a reformulation of Bitterli et al. [2020] using GRIS. We also reinterpret ReSTIR as an unbiased explorative non-Markovian chain and explain when averaging the images produced with it converges to the ground truth.

In light transport, we aim to produce an image where each pixel's color I_i is determined by integrals

$$I_i = \int_{\Omega} h_i(\bar{\mathbf{x}}) f(\bar{\mathbf{x}}) \, \mathrm{d}\bar{\mathbf{x}}, \tag{44}$$

for pixel index i, paths Ω from sensor to a light, image filter h_i, and path contribution function f. A path $\bar{\mathbf{x}}$ generally contributes to few pixels due to filter h_i.

Common filtering methods sample paths for each pixel and *splat* contributions $f(X)/p_X(X)$ onto the image with kernel h_j. This allows, without loss of generality, integrating pixel i only over paths Ω_i *directly* contributing to it. This corresponds to using a box filter for h_i, but generalizing to more complex filters is straightforward.

Integrating I_i for each pixel only over its domain Ω_i, gives

$$I_i = \int_{\Omega_i} f(\bar{\mathbf{x}}) \, \mathrm{d}\bar{\mathbf{x}}, \tag{45}$$

and sharing paths between domains Ω_i and Ω_j is impossible without path modification. We aim to more efficiently share paths between integrals by incorporating shift mappings into our resampling.

6.1 Formulation

We associate pixels i with path space domains Ω_i, integrands f_i (i.e., f restricted[10⋆] to Ω_i), and target functions $\hat{p}_i$, which could be e.g., grayscale path contribution functions $|f_i|$ or still cheaper approximations with bounded relative error.

We assume that each pixel i is equipped with a sampler for canonical paths X_i that are reasonable for integrating $\hat{p}_i$. The samples could e.g., be directly importance sampled for $\hat{p}_i$, or resampled with RIS from multiple reasonably importance sampled initial candidates.

6.2 Reservoirs and Weighted GRIS

We slightly extend the discussion on reservoirs in Section 2 to generalize reservoir merging to multiple input domains. A reservoir r stores a path X_r, its weight W_r, and a sample count M_r, as per the traditional use of reservoirs for sampling X_r from a stream of inputs. In that context, M_r is the number of samples the current X_r is resampled from, as X_r is randomly retained or replaced with the right probability at the encounter of each new input sample, and M_r is increased by one. A *reservoir merge* of reservoirs r_1 and r_2 builds a new reservoir r_m, with X_{r_m} resampled from X_{r_1} and X_{r_2} as if it were resampled from the concatenation of the input samples of r_1 and r_2, and M_{r_m} is simply $M_{r_1} + M_{r_2}$.

The interpretation of M_r as a sample count is too strict for ReSTIR: a reservoir merge simply resamples X_{r_m} with RIS from canonical samples X_{r_1} and X_{r_2}, with resampling MIS weights $m_{r_i}(y) = M_{r_i}/(M_{r_1} + M_{r_2})$. The meaning of M_r in this context is relative weight for the corresponding sample. Since we use the X_r for estimating an integral and the M_r define the relative weights of these samples, we refer to the M_r as *confidence weights*. In fact, ReSTIR even caps M_r to a constant M_c, limiting confidence on old samples, invalidating the old interpretation as a sample count.

Reservoir merging generalizes to weighted GRIS, with proper MIS weights (Section 5.6), simply by multiplying the $\hat{p}$ and $\hat{p}_{\leftarrow}$ in the MIS formulas by the corresponding reservoir's M_r; the resampling result is stored in X_{r_m}, and $M_{r_m} = \min(M_c, \sum_j M_{r_j})$. This generalized form of reservoir merging is used in the next section.

6.3 ReSTIR as Chained GRIS

We rewrite the key aspects of the ReSTIR algorithm, i.e., Bitterli et al. [2020, Algorithm 5], as a sequence of GRIS resampling steps; we will refer to the stages of this algorithm later:

Let Y_i^{t-1} be a resampled (or sampled) path for pixel i on frame t–1, stored for later reuse along with its unbiased contribution weight $W_{Y_i^{t-1}}$. For each frame t, in ReSTIR, we

(1) *(Initial candidates)* Generate an independent sample X_i^t for each pixel i and evaluate its contribution weight $W_{X_i^t}$.
(2) *(Temporal reuse)* Use GRIS to select Z_i by resampling between last frame's sample Y_i^{t-1} and new sample X_i^t. Pixel correspondences may be identified via motion vectors.
(3) *(Spatial reuse)* Each pixel selects numerous random spatial neighbors j, and selects Y_i^t by resampling between Z_i and neighbor samples Z_j via GRIS. This step may be executed multiple times with the assignment $Z_i := Y_i^t$.
(4) Estimate the pixel integral, $I_i^t \approx f_i(Y_i^t) W_{Y_i^t}$.

ReSTIR typically stores a reservoir for each pixel i. The new samples X_i^t are treated as reservoirs with $M_r = 1$, and are merged with the reservoir storing Y_i^{t-1}, accounting for the confidence weights. Spatial resampling works akin to a stochastic convolution, merging in reservoirs from random nearby pixels. The last sample Y_i^t is stored in that pixel's reservoir, and its confidence weight from the spatial reuse passes is used in the next frame's temporal resampling step.

Appendix B discusses additional details related to performance and correctness.

6.4 Path Space Exploration via M-capping

Capping M_r to constant M_c (Section 6.2) is critical. Without limiting M_r, the relative weights of new samples exponentially approach zero, causing convergence to the wrong result as in Figure 2.

With M-capping, the relative weight of the temporally reused sample is approximately limited to at most $M_c/(M_c + 1)$, which should intuitively fulfill the convergence constraints in Section 5.7.

[10⋆] Domain restriction: $\mathcal{D}(f_i) = \Omega_i \subset \Omega$ and $f_i(x) = f(x)$ in Ω_i.

[11] The abovementioned constraint is a requirement for *input samples*; even if the constraint is fulfilled, the ReSTIR result itself still will not converge since the number of spatial input samples is not increased ($M \not\to \infty$). Instead, ReSTIR will explore the path space in such a way that its average over frames will now converge in a still scene: Assume that we hypothetically resample, with $\hat{p} = f_i$, for pixel i a path from one of past frames, i.e., $Y = Y_i^s$ where s is random. Its PDF now approaches $f_i/\|f_i\|$, the variance of its contribution converges to zero, and the said contribution is the mean of the ReSTIR frames, $f_i(Y)W_Y = \frac{f_i(Y)}{\hat{p}(Y)} \sum_{t=1}^{T} \frac{1}{T} \cdot f_i(Y_i^t) W_{Y_i^t}$.

This gives us an intriguing interpretation of ReSTIR: with *state* defined as one path X_i for each pixel, ReSTIR produces an unbiased, explorative non-Markovian chain whose PDF approximates f better with more input samples. Averaging images of this chain converges in a still scene. In real-time, we display single states of the unbiased chain, updated in time by sampling, shifting and resampling paths.

6.5 ReSTIR for Offline Rendering

Temporal path reuse reduces current frame variance but correlates samples temporally. This slows convergence if temporally accumulating in a progressive renderer (Figure 9b). Instead, we propose rendering independent frames with spatial-only GRIS. This speeds convergence and should allow easier integration to existing systems.

This offline algorithm is a simple two-pass method. In a single iteration, the first pass performs one or more rounds of cross-pixel reuse with GRIS to resample canonical samples from other pixels. The second pass simply averages the produced images.

Proving convergence of the mean is now easy, despite the correlations in the spatial reuse passes: GRIS with proper MIS weights remains unbiased despite the correlations, and by Equation 40, the contributions of GRIS with a fixed number of spatial reuse passes remain bounded. Hence, averaging independently sampled frames converges.

Strictly speaking, the above convergence is true only for scalar functions f_i. In practice, f_i is vector-valued, and we use e.g., a grayscale $\hat{p}_i = |f_i|$, which guarantees convergence of a path $Y = Y_i^s$ sampled from a random frame s to the brightness $|f_i|$. Literally evaluating the unbiased contribution yields,

$$f_i(Y)W_Y = \frac{f_i(Y)}{|f_i(Y)|} \cdot \sum_{t=1}^{T} \frac{1}{T} \left|f_i(Y_i^t)\right| W_{Y_i^t}, \tag{46}$$

which may include color noise. However, in the offline context we assume budget for multiple samples, and thus recommend the explicit mean formula

$$\tilde{I}_i = \frac{1}{T} \sum_{t=1}^{T} f_i(Y_i^t) W_{Y_i^t} \tag{47}$$

as it removes color noise.

The convergence of $Y = Y_i^s$ in brightness opens potentially interesting future work: a random subset of the Y_i^t could be resampled and stored for each pixel to e.g., bootstrap the rendering of the next animation frame, or for re-rendering after material changes.

[11] This would lead to $b_k = (M_c/(M_c+1))^k$ with $b_k \to 0$, but an exact mathematical proof is hard due to correlations and complicated MIS weights.

7 DESIGNING SHIFT MAPPINGS

Previous light transport techniques that manipulate and reuse paths (e.g., gradient-domain rendering) introduce various shift mappings to map paths between pixels. Generally, no single shift map is optimal and the best one depends on both scene properties as well as their computational efficiency on different hardware. In this section, we describe key properties of effective shift maps and introduce common building blocks for practical shift mappings.

We also describe a novel design principle for effective GPU-based shift mappings, how to choose the shifting strategy based on the sampled BSDF lobe, and new heuristics for avoiding noise.

7.1 Shift Mapping

A shift map T takes a path $\bar{\mathbf{x}}$ from pixel k and maps it to another path $\bar{\mathbf{y}} = T(\bar{\mathbf{x}})$ in pixel j. We call the original path $\bar{\mathbf{x}}$ the *base path*, and the shifted $\bar{\mathbf{y}}$ the *offset path*. Using Veach's [1998] vertex parametrization, we define a generic shift map T from Ω_k to Ω_j as

$$T([\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, ...]) = [\mathbf{y}_0, \mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, ...] \ . \tag{48}$$

Vertex $\mathbf{y}_0$ is normally specified on the sensor and $\mathbf{y}_1$ comes from tracing through pixel j, accounting for depth-of-field parameters.

When designing shift maps, the main freedom (and challenge) is designing a heuristic for vertices $\mathbf{y}_2$ and beyond so the shift approximately retains the path contribution, $f_k(T_k(\bar{\mathbf{x}})) \approx f_j(\bar{\mathbf{x}})$ and $|\partial T_k/\partial \bar{\mathbf{x}}| \approx 1$. Maximizing similarity of path contributions roughly equates to reusing (nearly) the same paths for nearby pixels, a common design heuristic. Figure 5 shows a hybrid shift mapping of random replay and reconnection as an example.

Local decisions. A common strategy to find offset paths $\bar{\mathbf{y}}$ builds them sequentially, vertex-by-vertex, starting from $\mathbf{y}_1$ and analyzing local base and offset path geometry. For each i, the next offset vertex $\mathbf{y}_{i+1}$ is decided based on base path vertices $\mathbf{x}_{i-1}$, $\mathbf{x}_i$, and $\mathbf{x}_{i+1}$ plus offset path vertices $\mathbf{y}_{i-1}$ and $\mathbf{y}_i$. For example, if vertices $\mathbf{x}_i$, $\mathbf{x}_{i+1}$ and $\mathbf{y}_i$ have rough materials, a common strategy connects the base and offset paths by choosing $\mathbf{y}_{i+1} = \mathbf{x}_{i+1}$. Previous vertices $\mathbf{x}_{i-1}$ and $\mathbf{y}_{i-1}$ can also be used to perform half-vector copy [Kettunen et al. 2015].

Ensuring bijectivity. Sequential construction of offset paths sometimes halts abruptly: e.g., in half-vector copy, local decisions can map a refraction into total internal reflection, but the reverse never happens, breaking bijectivity. Bijectivity is not always achievable throughout the path space, but that poses no big problem: not all paths need belong to the shift mapping domain. The shift may, after trying to shift a path, simply return "undefined." This marks the path as not belonging in the shift's domain.

Any successful shift must be invertible: if $\bar{\mathbf{x}}$ shifts to $\bar{\mathbf{y}}$, an inverse shift must exist to map $\bar{\mathbf{y}}$ back to $\bar{\mathbf{x}}$. Often slightly more is guaranteed by designing symmetric shift mappings where if $T_{k\to j}(\bar{\mathbf{x}}) = \bar{\mathbf{y}}$, then $T_{j\to k}(\bar{\mathbf{y}}) = \bar{\mathbf{x}}$. Removing paths from a map's domain may cause noise and waste computation, but neglecting bijectivity introduces significant bias.

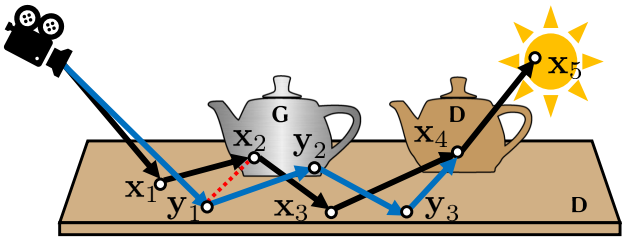


Fig. 5. *A hybrid shift mapping. The* **base path** *selects* $\mathbf{x}_4$ *for reconnection, since both* $\mathbf{x}_3$ *and* $\mathbf{x}_4$ *are rough. The* **offset path** *copies the random numbers of the base at* $\mathbf{x}_1$ *and* $\mathbf{x}_2$ *to construct similar scatter directions for* $\mathbf{y}_1$ *and* $\mathbf{y}_2$ *and reconnects* $\mathbf{y}_3$ *to* $\mathbf{x}_4$*. This is the earliest reconnection giving two consecutive rough/diffuse vertices. Without connectability conditions the* **offset path** *would connect* $\mathbf{y}_1$ *to* $\mathbf{x}_2$ *(a glossy vertex), potentially giving a path of near-zero contribution as* $\mathbf{y}_1 \leftrightarrow \mathbf{x}_2 \leftrightarrow \mathbf{x}_3$ *is far from an ideal reflection.*

7.2 Common Building Blocks

Local decisions for building offset paths often stem from fixing some property of the base path; such invariants share information between the paths, making shift inversion possible. Here we briefly review some common strategies for shift mapping:

Vertex copy (reconnection). Reconnecting offset and base paths as soon as possible is common, as vertex sharing is cheap and often keeps path contributions similar. If $\mathbf{x}_i$, $\mathbf{x}_{i+1}$, and $\mathbf{y}_i$ all lie on rough materials, Lehtinen et al. [2013] reconnect the offset path to the base path by setting $\mathbf{y}_{i+1} = \mathbf{x}_{i+1}$. Subsequent vertices of $\bar{\mathbf{x}}$ are normally copied too. This strategy is good for diffuse and rough materials.

Half-vector copy. Reconnection breaks path similarity for near-specular vertices. Kettunen et al. [2015] transform the base path's half-vector into local tangent space, copy it to the offset path, and re-trace the vertex $\mathbf{y}_{i+1}$ in the reflection (or refraction) direction.

Direction copy. Direction copy takes the exitant direction from a base path vertex and copies it to the offset path, in global coordinates, and re-traces to find the next vertex $\mathbf{y}_{i+1}$. Direction copy is often used with environment mapping.

Random replay. Random replay copies the base path's random numbers to re-trace $\mathbf{y}_{i+1}$ with the method used by the base path. Random replay often makes decisions roughly similar to copying the half-vector or direction, or reconnecting to an area light in the case of next-event-estimation.

Manifold exploration. If reconnection is impossible due to specularities, Lehtinen et al. [2013] find the next connectable vertex, copy it, and apply manifold exploration [Jakob and Marschner 2012] for intermediate (near-)specular vertices. This strategy iteratively constructs high-quality offset paths, albeit at relatively high cost.

While reconnecting quickly is often a good strategy, there are pitfalls. Some challenges for path shifts include: not closely approximating ideal reflections on high-gloss surfaces, trying to reconnect through occlusions, drastically changing the reconnection segment length, shifting between different objects or materials, or simply diverging too far (e.g., due to reflection or refraction).

7.3 A Full Shift Mapping

Full shift mappings combine these building blocks, often based on simple heuristics. For example, Kettunen et al. [2015] sequentially analyze base and offset paths to find suitable reconnections using a simple condition: vertices $\mathbf{x}_i$, $\mathbf{x}_{i+1}$, and $\mathbf{y}_i$ must all be considered "sufficiently" rough. If this test passes, the base and offset paths are reconnected, otherwise a half-vector copy is used and the test is repeated for the next vertex. Hua et al. [2019] show equivalent results by replacing the half-vector copy by random replay. We found Hua et al.'s approach more efficient on the GPU and slightly more general, so we adopt it with several improvements.

7.4 Shift Mappings Optimized for Real-Time Rendering

We study two different shift mappings, and modify them as needed to make them suitable for a GPU implementation that can target real-time rendering:

- The reconnection shift [Lehtinen et al. 2013] sets $\mathbf{y}_2 = \mathbf{x}_2$ to always connect at the first indirect vertex. This works well for mostly diffuse scenes. ReSTIR GI [Ouyang et al. 2021] implicitly uses this choice but trades correctness for performance.
- A hybrid of random replay and reconnection [Hua et al. 2019] that postpones the reconnection by using random replay if certain connectability conditions are not fulfilled. We present an improved variant of this shift mapping.

The reconnection shift is easy to implement efficiently: it only requires storing the reconnection vertex and re-evaluating the path contribution. Implementing the hybrid shift efficiently is non-trivial: random replay is not efficient without reconnections, but due to potential reconnection postponing, all base path vertices need to be stored as candidates for reconnection. This is not ideal since GPU ray tracing is often memory-bound.

We minimize memory use by letting the base path select a single potential reconnection vertex; we precompute the first base path vertex $\mathbf{x}_i$ that satisfies the connectability condition for $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$. Reconnection must happen at this vertex, or it does not happen. This only requires storing vertex $\mathbf{x}_{i+1}$ instead of the full path.

This constraint is reasonable. For useful path reuse, the base and offset paths should be relatively similar; if similar enough, they should also agree on the reconnection index. Further, our bijectivity requirement *forces* this guarantee: when building $\mathbf{y}$, if we find it disagrees on the earliest possible reconnection vertex, the shift must return "undefined" as it would not be invertible.

7.5 Connectability Conditions

We propose two novel improvements for the connectability conditions compared to previous work, and we find these to often result in a significant noise and artifact reduction with our method.

Distance condition. Area formulations of the rendering equation include geometry terms that become singular for short path segments, e.g., in corners. In unidirectional path tracing, this singularity stems from next-event-estimation but is eliminated by standard MIS.

Similar singularities appear when reconnecting nearby vertices, causing increased noise near geometry edges.

We propose reducing this problem by skipping reconnections that introduce short segments. This is similar to Manzi et al.'s [2014] distances test, but instead of spawning a manifold walk, we postpone reconnection by performing random replay. More concretely, we only allow reconnection to $\mathbf{x}_{i+1}$ if $\|\mathbf{x}_{i+1}-\mathbf{x}_i\| \geq d_{\text{max}}$. By symmetry, the offset path must fulfill $\|\mathbf{x}_{i+1}-\mathbf{y}_i\| \geq d_{\text{max}}$ for $\mathbf{y}_{i+1}$ to become $\mathbf{x}_{i+1}$, as described in Section 7.4.

Lobe-specific connectability. Kettunen et al. [2015] test reconnection feasibility by ensuring roughness values of $\mathbf{x}_i$, $\mathbf{x}_{i+1}$, and $\mathbf{y}_i$ all exceed a given threshold. But since BSDFs often sum multiple separate lobes, such tests are ambiguous, e.g., on a material with a bottom diffuse layer and a top clear coat. Choosing good shift maps for such materials is a long-standing problem [Kettunen 2020]: how to classify roughness of composite materials with just one parameter? Any single strategy likely mistreats at least one layer.

Unidirectional path tracers often optimize sampling by decomposing and evaluating only one BSDFs lobe per vertex [Szécsi et al. 2003]. We propose the same for shift mappings: we examine the roughness of just the selected lobe, and otherwise proceed as described in Section 7.4. We detail the path extension with lobe indices that is required for lobe-specific connectability in Section 7.6.

7.6 Extending Paths with Lobe Indices

For a path tracer generating N paths by different techniques for each path length d, the standard path integral can be written

$$I = \sum_{d=1}^{\infty} \sum_{n=1}^{N} \int_{\Omega_d} \omega_n(\bar{\mathbf{x}}) f(\bar{\mathbf{x}}) \, \mathrm{d}\bar{\mathbf{x}} \,, \tag{49}$$

where ω_n is the MIS weight for path strategy n. For simplicity, if we assume $N = 2$, then $n = 1$ uses next-event estimation to sample the last vertex and $n = 2$ uses BSDF sampling. Balance heuristic MIS weights give $\omega_n(\bar{\mathbf{x}}) = \frac{p_n(\bar{\mathbf{x}})}{p_1(\bar{\mathbf{x}})+p_2(\bar{\mathbf{x}})}$, where p_1 and p_2 are the NEE and BSDF sampling PDFs of the path, and sum over all BSDF lobes.

Our improved shift strategies from Section 7.4 require splitting BSDFs into lobes. We transform Equation 49 to use a lobe-extended path space, allowing us to implement ReSTIR by simple substitution, without any need for heuristic argumentation.

We combine paths $\bar{\mathbf{x}} = (\mathbf{x}_0, \ldots, \mathbf{x}_d)$ with sequences of lobe indices $\bar{\ell} = (\ell_1, \ldots, \ell_{d-1})$ into an extended path space of length-d paths $\tilde{\Omega}_d$ represented by pairs $(\bar{\mathbf{x}}, \bar{\ell})$. Each ℓ_j is a positive integer $1 \leq \ell_j \leq N_{\text{lobe}}$ (the BRDF model's lobe count) or special symbol $\ell_{d-1} = \mathcal{N}$ if our path ends with next-event estimation.

With f denoting the usual path contribution function, we define a partial contribution function $f_{\bar{\ell}}$ as follows: for fully BSDF-sampled paths, it evaluates only lobe ℓ_j at each vertex $\mathbf{x}_j$. If the last vertex $\mathbf{x}_d$ is NEE-sampled, i.e., $\ell_{d-1} = \mathcal{N}$, the BSDF at $\mathbf{x}_{d-1}$ is evaluated with all lobes.

Denoting the set of lobe index sequences by L_d, we can rewrite Equation 49 as

$$I = \sum_{d=1}^{\infty} \sum_{\bar{\ell} \in L_d} \int_{\Omega_d} \omega_{n(\bar{\ell})}(\bar{\mathbf{x}}) f_{\bar{\ell}}(\bar{\mathbf{x}}) \, \mathrm{d}\bar{\mathbf{x}}, \tag{50}$$

for $n(\bar{\ell}) \in \{1, 2\}$ based on whether the last vertex is NEE- or BSDF-sampled. Finally, we combine the sums into an integral similar to Veach [1998], which integrates over our extended path space $\tilde{\mathbf{x}} \in \tilde{\Omega}$ for pairs $\tilde{\mathbf{x}} = (\bar{\mathbf{x}}, \bar{\ell})$ of all lengths d:

$$I = \int_{\tilde{\Omega}} \omega_{n(\bar{\ell})}(\bar{\mathbf{x}}) f_{\bar{\ell}}(\bar{\mathbf{x}}) \, \mathrm{d}\tilde{\mathbf{x}}. \tag{51}$$

This formulation allows use of shift mappings that reason about the BSDF lobes, which is not possible in vertex-based path spaces.

To shift a path, we test if vertex $\mathbf{x}_j$ is sufficiently rough by examining the roughness of the lobe ℓ_j chosen to sample vertex $\mathbf{x}_{j+1}$. We treat NEE-sampled vertices as rough if at least *one* of their BRDF lobes is sufficiently rough. All light vertices are treated as rough. If all three reconnection vertices ($\mathbf{x}_j$, $\mathbf{x}_{j+1}$ and $\mathbf{y}_j$) pass the roughness and distance conditions (Section 7.5), we execute the reconnection; otherwise we sample $\mathbf{y}_{j+1}$ via random replay.

When using random replay, base and offset paths typically select the same BRDF lobes, as reusing random numbers over nearby paths gives similar per-vertex choices. A reconnection shift copies the lobe index from the base path vertex. In both cases, path contribution is likely preserved. Separating BRDF lobes frequently increases efficiency of our hybrid shift significantly (see Figure 12).

8 IMPLEMENTATION

We apply our GRIS theory in a proof-of-concept path tracing algorithm we call ReSTIR path tracing (ReSTIR PT). We build on the Falcor GPU rendering framework [Kallweit et al. 2021], and implement ReSTIR PT as chained GRIS passes, per Section 6.3.

ReSTIR PT can use any shift map to reuse paths between pixels, but we implement the two from the previous section: a *hybrid shift* combining random replay and reconnection with our lobe-specific improvements, and a simpler *reconnection shift* that always reconnects to the first indirect vertex.

Like many path tracers, ours only evaluates the sampled BSDF lobe for BSDF-sampled vertices and evaluates all lobes for NEE-sampled vertices. We treat lobe selections as additional path parameters, as described in Section 7.6, using the sampled lobe roughness to choose between reconnection and random replay.

Our ReSTIR PT implementation handles full surface-to-surface light transport. Volumetric media requires a volumetric shift map; Lin et al. [2021] implicitly defines one possibility and Gruson et al. [2018] propose another, though finding fast volumetric shifts for resampling remains interesting future work.

We have two prototypes, targeting unbiased real-time and offline light transport, though neither is performance optimized. This contrasts with Ouyang et al. [2021], a biased[12] but optimized precursor. Building on our generalized theory, ReSTIR PT is an unbiased global illumination method that better handles specular light transport, thanks to supporting arbitrary shift maps.

While we expect benefits to direct illumination from our GRIS theory, our implementations primarily address indirect light. We use ReSTIR DI [Bitterli et al. 2020] for direct lighting.

Below, we discuss our design choices and implementation details.

[12] Ouyang et al. [2021] explain some key sources of bias such as assuming Lambertian scattering at reconnection vertices; following our GRIS theory removes all bias.

8.1 Jacobian Determinants

We first give Jacobian determinants for the reconnection and random replay shifts. We assume base and offset paths up to vertex i are fixed; probability densities below are to be understood as conditional to earlier path state.

We denote by ω_i^x the unit vector from $\mathbf{x}_i$ to $\mathbf{x}_{i+1}$, and the corresponding random numbers leading from vertex $\mathbf{x}_i$ to $\mathbf{x}_{i+1}$ by $\bar{\mathbf{u}}_i^x$. The offset path features similar notation with y.

Solid angle. When using the common solid angle parametrization, the Jacobian for the reconnection shift is (e.g., Kettunen et al. [2015])

$$\left|\frac{\partial \omega_i^y}{\partial \omega_i^x}\right| = \left|\frac{\cos\theta_2^y}{\cos\theta_2^x}\right| \frac{\|\mathbf{x}_{i+1}-\mathbf{x}_i\|^2}{\|\mathbf{x}_{i+1}-\mathbf{y}_i\|^2}, \tag{52}$$

for $\theta_2^\bullet$ the angle between $\omega_i^\bullet$ and the geometric surface normal at $\mathbf{x}_{i+1} = \mathbf{y}_{i+1}$. The Jacobian for deciding y_{i+1} by random replay is

$$\left|\frac{\partial \omega_i^y}{\partial \omega_i^x}\right| = \left|\frac{\partial \omega_i^y}{\partial \bar{\mathbf{u}}_i^y}\right| \left|\frac{\partial \bar{\mathbf{u}}_i^y}{\partial \bar{\mathbf{u}}_i^x}\right| \left|\frac{\partial \bar{\mathbf{u}}_i^x}{\partial \omega_i^x}\right| = \frac{p_{\omega_i^x}(\mathbf{x}_{i+1})}{p_{\omega_i^y}(\mathbf{y}_{i+1})}, \tag{53}$$

i.e., the ratio of the solid angle sampling probabilities of the next vertices, given the paths up to vertex i.

As we use local shift decisions, the Jacobian of the full path shift is the product of Jacobians from each vertex.

Primary-sample space (PSS). Path tracers build paths $\bar{\mathbf{x}}$ based on random number sequences $\bar{\mathbf{u}} = (u_1, u_2, \ldots)$, defining a primary-sample space $\mathcal{U}$. In practice, integrals of form $\int_\Omega f(\bar{\mathbf{x}})\,\mathrm{d}\bar{\mathbf{x}}$ are evaluated $\int_{\mathcal{U}} \frac{f(x(\bar{\mathbf{u}}))}{p_X(x(\bar{\mathbf{u}}))}\,\mathrm{d}\bar{\mathbf{u}}$, where x builds paths using random numbers.

Our prototypes use primary-sample space for easier implementation. Results are identical, but interpretations change: the integrand is $f(x(\bar{\mathbf{u}}))/p_X(x(\bar{\mathbf{u}}))$ over domain $\mathcal{U}$ and the PDF of the primary samples is $p_U = 1$. Section S.3 briefly introduces the PSS formulation of the path integral.

The Jacobian for random replay in the PSS parametrization is always 1, and solid-angle Jacobians can be converted into primary-sample space by dividing by the right-hand-side of Equation 53. Due to the vertex-by-vertex construction of the paths, we have

$$\left|\frac{\partial \bar{\mathbf{u}}_i^y}{\partial \bar{\mathbf{u}}_i^x}\right| = \left|\frac{\partial \bar{\mathbf{u}}_i^y}{\partial \omega_i^y}\right| \left|\frac{\partial \omega_i^y}{\partial \omega_i^x}\right| \left|\frac{\partial \omega_i^x}{\partial \bar{\mathbf{u}}_x}\right| = \frac{p_{\omega_i^y}(\mathbf{y}_{i+1})}{p_{\omega_i^x}(\mathbf{x}_{i+1})} \left|\frac{\partial \omega_i^y}{\partial \omega_i^x}\right| . \tag{54}$$

Mixing PSS and path space shifts ⋆. Mixing random replay with path space shifts poses a conceptual challenge. Path samplers often consume more random numbers than the path dimensionality; this dimensionality mismatch means Jacobians between path space and primary-sample space do not exist. Bitterli et al. [2017] bijectively map between paths and their random numbers by padding paths with extra, unused dimensions. Assuming this theoretical bijection exists often allows mixing path space shifts and random replay.

8.2 Reservoir Storage

Our reservoirs (Section 6.3) consume 88 bytes per path while supporting our hybrid shift. Beyond storing contribution weights W_r and confidence weights M_r (Section 6), we store information needed for our shift map: the path's chosen reconnection vertex and a seed for random replay. See Section S.4 for details.

8.3 Parameters

In the following, we refer to steps of the ReSTIR algorithm as described in Section 6.3.

Offline. For offline rendering, we sample 32 initial path trees for each pixel via path tracing, and resample one path with RIS (as in Lin et al. [2021]). The selected path gets reused over three iterations of spatial reuse. All reuse passes resample from the current pixel and six neighbors selected from a 10-pixel radius via a low-discrepancy sequence. We found this a near-optimal configuration for offline rendering. Section S.6 contains a parameter ablation.

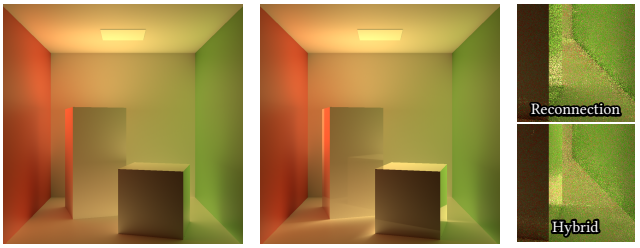
Real-time. For real-time rendering, we resample the path from only one path tree sampled with path tracing. A single spatial reuse pass selects three random neighbors in a 20-pixel radius; this keeps spatial reuse costs low and relies more on temporal reuse to improve distributions. We cap M with $M_c = 20$, i.e., the prior frame confidence is at most 20× that of new samples.

Connectability thresholds. Good values for distance and roughness thresholds are important, but relatively consistent. A per-lobe GGX-roughness threshold of 0.2 generally works well; scene scale affects the distance threshold, we used 1% - 5% scene size in different test scenes. The camera distance to the region of interest and material glossiness affects the optimal parameter. We leave automatically setting ideal parameters to future work.

Resampling MIS. We use the defensive variant of pairwise MIS (Equation 38) for spatial reuse. The real-time variant additionally uses generalized Talbot MIS (Equation 36) for temporal reuse. For both, $|R| = 1$, and convergence is realized in the offline case by rendering multiple independent frames.

9 RESULTS AND DISCUSSION

Below we first validate that convergence of ReSTIR PT matches our guarantees from Section 5. In Section 9.2 we quantify the quality of our shift maps from Section 7, and in Section 9.3, compare ReSTIR PT to recent global illumination algorithms. We present results separately for our real-time and offline variants. All performance numbers were measured on a GeForce RTX 3090 at 1920×1080.



(a) *Default Cornell Box* **(b)** *With glossy boxes* **(c)** *Two shifts*

Fig. 6. *A 3-bounce reference for two variants of the Cornell Box. (a) Default version, roughness 0.5. (b) Two glossy boxes with roughness 0.15. (c) A quality comparison of 1 spp ReSTIR PT with reconnection shift (5 ms) and hybrid shift (12 ms) in the glossy variant.*

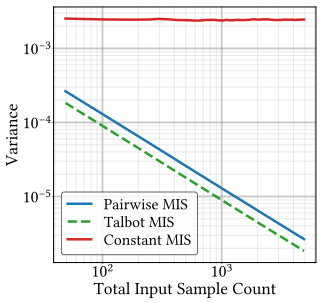


Fig. 7. *Single-sample integration error with GRIS, with increasing input samples from each pixel within a 7×7 reuse window. Asymptotic zero-variance integration is realized with generalized Talbot and pairwise MIS weights but not with constant MIS weights (red).*

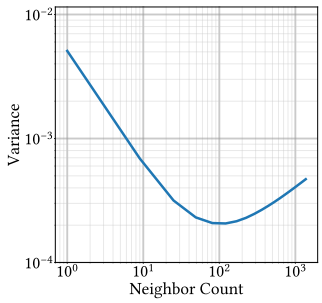


Fig. 8. *Breaking $|R| > O(\sqrt{M})$ loses convergence guarantees. Here, we reuse from a central (canonical) sample ($|R| = 1$) and an increasing window of M pixels around it. Even with proper MIS weights, without new canonical samples faraway pixels are increasingly worse matches. Eventually, this offsets any benefit from reusing more paths.*

9.1 Convergence Results

We study convergence in the Cornell Box scene, with modified materials mixing Lambertian and GGX microfacet BSDFs [Walter et al. 2007] with roughness 0.5 (Figure 6a).

We evaluate convergence behavior for three-bounce indirect illumination, and use the reconnection shift for simplicity. We compute error in grayscale images to map the convergence results to GRIS.

Asymptotic convergence with fixed reuse window. Sections 5.6 and 5.7 give robust Talbot and pairwise resampling MIS weights that asymptotically realize single-sample zero-variance integration, under certain correlation and importance sampling criteria. In Figure 7, we fix the ratio $|R|/M$ as we increase the independent input sample count over a fixed window of spatial-only reuse.

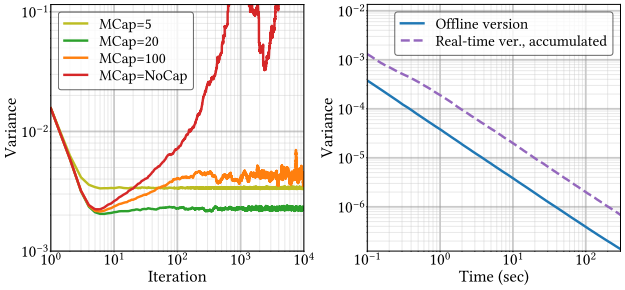
Both our generalized pairwise and Talbot MIS weights realize a linear curve showing asymptotic zero-variance integration, but constant MIS weights do not. While Talbot weights have lower per-sample error, the lower algorithmic complexity of pairwise MIS can achieve similar variance 6-7× cheaper.

Non-convergence with increasing reuse window. We only guarantee convergence if $|R| > O(\sqrt{M})$. One case that breaks this criterion reuses samples from each pixel over an increasing window, i.e., $|R| = 1$ but M grows. This may not converge if part of path space is only covered by the canonical (central) sample. Adding more non-canonical samples decreases the chance to select the central one, so poorly-covered areas slowly become sampled worse. Figure 8 shows reuse over larger windows initially lowers variance, but variance increases beyond some point.

Temporal history and M-cap. In Section 6.3 we discuss capping the temporal confidence weight M_r in ReSTIR reservoirs. As the cap M_c increases, b_k approaches 1 in Section 5.7. Using $M_c = \infty$ corresponds to $b_k = 1$, where we lose all guarantees. Figure 2 shows a simple example of this failure, converging to a static and wrong result.

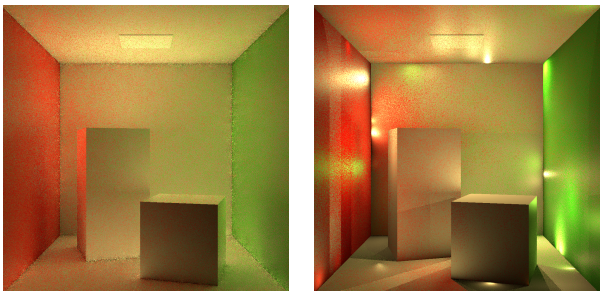
In Figure 9a we show ReSTIR PT's integration error with temporal reuse and increasing frame counts. Colors correspond to different M-cap values; the scene is static to avoid errors from animation.

Pixels compute a new independent sample on each iteration, which is resampled with the temporal result from the prior frame.



(a) *Effect of M-cap* **(b)** *Real-time (M=20) vs. offline*

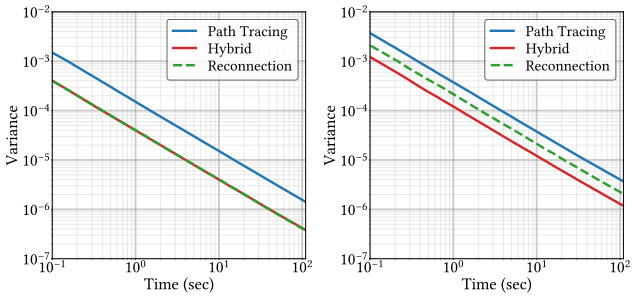
Fig. 9. *(a) Error of ReSTIR PT with temporal reuse, with increasing frame counts and different M-cap values. A large M-cap eventually increases noise, while low values do not minimize error. Good M-caps (green) give consistently low errors. (b) Our offline method (blue) turns off temporal reuse, which converges faster when averaging frames; it avoids the frame-to-frame correlation introduced by temporal reuse.*



(a) *M-cap: $M_c = 20$* **(b)** *No M cap*

Fig. 10. *Image of Figure 9a after 10^4 iterations, with and without M-cap. Like Figure 2, (b) shows convergence to a wrong result due to temporal correlations.*

The old sample's relative weight is $M_r/(M_r+1)$, which drastically favors this sample. This is akin to an exponential moving average.



(a) *Default Cornell Box* **(b)** *With glossy boxes*

Fig. 11. *Error comparison of our ReSTIR PT with the reconnection shift (blue) and the hybrid shift (green). Both shift mappings are good for rough scenes (left), but the hybrid shift (green) is more suited for scenes with glossy surfaces (right).*

Reusing prior frame samples initially improves Monte Carlo variance. But until M_r reaches the cap, the relative weight of new samples diminishes, increasing frame-to-frame correlation. This is analogous to *sample impoverishment* common in SIR: fewer and fewer unique samples remain, unless this correlation buildup is halted.

Eventually, increased correlation overshadows the benefits of reuse (red, orange) leading to higher variance until M-cap is reached. Capping M to maximize the benefits of temporal reuse (green) is key to real-time rendering with ReSTIR. Figure 10 shows the visual impact of capping M.

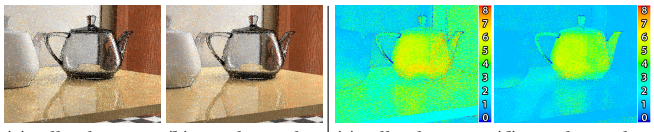
In Figure 9b we show the convergence behavior when averaging consecutive frames of our real-time ReSTIR PT with a finite M-cap (purple). We empirically find convergence, as the M-cap decorrelates temporally distant frames, i.e., it forgets temporal correlations.

Offline rendering. The goal of offline rendering is slightly different: instead of maximizing the individual quality of each frame, we want to produce the best image over a longer rendering time. We find that for this purpose, the correlations from temporal reuse hurt more than they help, and we propose turning off temporal reuse for offline rendering. The additional rendering time allows us to use slightly different rendering parameters (Section 8.3), and due to both improvements, the resulting algorithm often converges significantly faster (Figure 9b, blue).

9.2 Shift Mapping Results

We study ReSTIR PT with the reconnection and hybrid shifts in the Cornell Box scene. We analyze the results in the rough variant (GGX roughness 0.5, Figure 6a), and variant with glossy boxes (roughness 0.15, Figure 6b).

Reconnection is good for rough. We plot convergence in Figure 11 with a path tracing baseline. As known from gradient-domain rendering (e.g. Kettunen et al. [2015]), the reconnection shift is efficient for rough surfaces (Figure 11a, green) allowing cheap path reuse from indirect light. Because the reconnection shift (green) always reconnects after a primary hit, regardless of BSDF, it less efficiently reuses paths involving glossy interactions (Figure 11b, green).



(a) *All Lobes* *MAPE: 0.316* *Time: 51.7 ms*
(b) *Random Lobe* *MAPE: 0.297* *Time: 38.7 ms*
(c) *All Lobes* *Avg. path length pre-connection: 2.3*
(d) *Random Lobe* *Avg. path length pre-connection: 1.3*

Fig. 12. *Evaluating just one random BSDF lobe enables lobe-specific shift maps providing more efficient reuse, less noise, and shorter render time. The heatmaps show lobe-specific shifts decrease the average path length on multi-lobe materials. See full image in Figure 15.*

Hybrid is more robust. Random replay better handles glossy surfaces; our hybrid shift inherits this property (Figure 11b, red) while also remaining effective on rough surfaces (Figure 11a, red).

Visual comparison. Figure 6 shows shift map behavior on varied material types. For the glossy Cornell Box we render one path per pixel with ReSTIR PT (Figure 6c) with spatiotemporal reuse. On rough surfaces our hybrid shift (Figure 6c, bottom) behaves similar to a reconnection shift (Figure 6c, top), but our hybrid's distance criteria helps decrease noise at box edges.

Our hybrid shift postpones reconnections on glossy materials by inserting a random replay. This often improves later reconnections, reducing noise on glossy surfaces (Figure 6c, side of glossy box).

The lower reconnection shift quality is somewhat offset by its lower cost; equal-time comparisons allow averaging multiple independent iterations of the algorithm (e.g., as in Figure 15).

Separate handling of lobes. Many renderers only evaluate one random lobe per BSDF evaluation. Varying the selected shift map per lobe significantly reduces noise and improves performance (see Figure 12). Reconnecting works well for rough BSDFs, and random replay is effective on glossy BSDFs. Neither shift is ideal for multi-lobe materials, but shifting lobes separately fixes the issue.

Caustics. Caustics paths (i.e., {LS+DE} per Heckbert [1990]) are important in highly specular scenes. Interestingly, we find our two shift maps work well for different types of caustics with ReSTIR PT.

Our hybrid shift effectively reuses paths for contact caustics, i.e., light concentrating on a nearby surface (see Figure 13, top). When tracing an offset path through the bunny, random replay produces paths similar to the base path; if this path hits the same light source, random replay gets good path reuse. Reconnection shifts, however, often fail to reconnect on near-delta BRDFs.

Caustics from distant highlights, e.g., the lamp reflection in Figure 13, bottom, perform poorly with the hybrid shift. Offset paths generated by random replay easily diverge enough to miss the small highlight, increasing noise. Conversely, reconnection only changes the incident direction slightly when reconnecting to the distant window, minimizing path divergence and increasing path contributions.

We expect a manifold exploration shift [Lehtinen et al. 2013] would improve both cases; while expensive if applying to all paths, ReSTIR PT works with single path resampled from the path tree, making such a shift feasible. This is interesting future work.

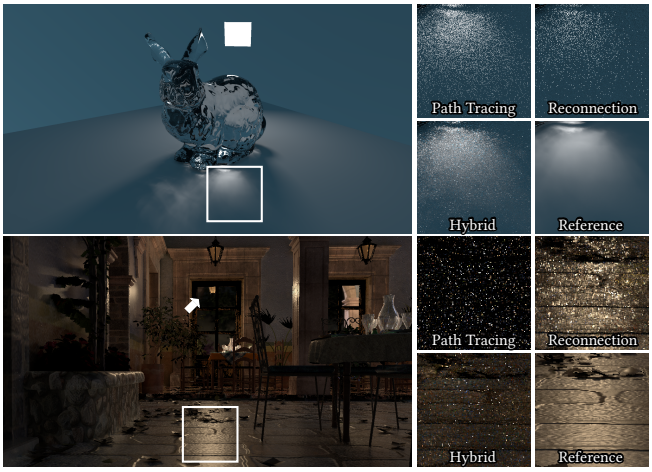
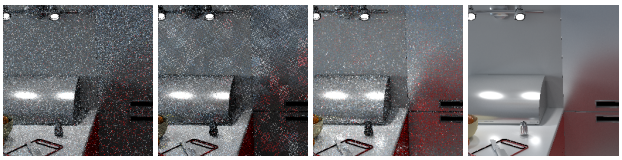


Fig. 13. *(Top) Equal time (25 ms) comparison of path tracing and ReSTIR PT with different shifts for our glass bunny. (Bottom) For San Miguel (60 ms), we show only indirect light to emphasize caustic contributions. Note the two shifts excel on different kinds of caustics.*



(a) *Path Tracing* *MAPE: 0.958* **(b)** *BPR* *MAPE: 0.898* **(c)** *Ours (reconn.)* *MAPE: 0.325* **(d)** *Reference*

Fig. 14. *(b) BPR [Bekaert et al. 2002] often reduces error versus (a) path tracing, but causes distracting structural artifacts at low sample counts. (c) Our ReSTIR PT gives less error without structural artifacts, despite also reusing spatially. Equal-time comparison in Kitchen (33 ms for uncropped images). All methods use ReSTIR DI for direct light.*

9.3 Rendering Results

In this section we compare our results in three contexts: explicitly versus Bekaert et al.'s [2002] path reuse, general quality comparisons to other real-time methods, and comparisons in an offline context.

Average light levels vary widely across our scenes, so we tone map for visual presentation. But we report errors with MAPE[13] (mean absolute percentage error) on HDR results. This L1 metric is more resistant to occasional fireflies in sample-reuse algorithms.

For comparisons, we used Falcor's [Kallweit et al. 2021] built-in unidirectional path tracer, and implemented ReSTIR PT, Bekaert-style path reuse (BPR) [Bekaert et al. 2002], ReSTIR GI [Ouyang et al. 2021], and ReSTIR DI [Bitterli et al. 2020] using this framework.

Versus Path Reuse. In Figure 14, we compare ReSTIR PT with a reconnection shift, path tracing and BPR. We reimplemented BPR per Bekaert et al. [2002], by sampling a path tree for each pixel, dividing the image into N-rooks tiles, and connecting pixels to all paths within a tile using a reconnection shift. We use 16-pixel tiles.

But path reuse over tiles, especially at low sample counts, introduces obvious tile boundaries. ReSTIR-style reuse does not introduce artificial edges, as reuse radii are selected separately per-pixel. Due to tile artifacts, we skip further real-time comparisons with Bekaert.

Real-Time Rendering. For real-time comparisons, *all* algorithms compute direct lighting via ReSTIR DI [Bitterli et al. 2020], unless otherwise mentioned. Image differences thus depend on how various algorithms compute indirect illumination. The second and third columns in Figure 15 toggle ReSTIR DI. It greatly reduces variance in direct light, but noise in indirect light requires other methods.

For equal-time comparisons, we render with ReSTIR PT using the hybrid shift and increase sample counts in other methods to reach (approximately) equal time. For ReSTIR PT with the reconnection shift, we run multiple independent ReSTIR chains simultaneously to match this cost. Figures 12 to 15 were captured during camera motion, to prevent ReSTIR PT from overly relying on temporal reuse for visual quality. We use no antialiasing or denoising.

In Figure 15, our ReSTIR PT with either our hybrid or reconnection shifts achieves the lowest error. Our hybrid shift significantly improves quality for glossy and refractive surfaces, e.g., the refractive wine glasses, the mirror in SanMiguel, the metal in VeachAjar, and the multi-layer surfaces in Zeroday and VeachAjar scene.

Our hybrid shift also reduces noise near geometric edges (e.g., the first Kitchen inset). But the reconnection shift outperforms when more, cheaper samples are better, e.g., on sufficiently rough surfaces (the second Kitchen inset) or to help reduce color noise (the second Zeroday inset).

Numerically, ReSTIR PT achieves 24% - 75% lower MAPE compared to path tracing (both with ReSTIR DI), but we find subjective visual improvement much larger. The supplemental material contains a result viewer for visual inspection.

Figure 1 compares ReSTIR PT on two complex scenes, with MAPE 16% lower than ReSTIR GI. While ReSTIR GI handles diffuse surfaces well, glossy surfaces and refractions and handled poorly, introducing bias. All methods in Figure 15 are unbiased.

Offline Rendering. For offline comparisons, we show 5 second equal-time renderings in representative scenes (see Figure 16). In Figure 17 we show convergence plots (up to 640 seconds) for all scenes. These figures show results and measure error *only* for indirect light.

Our ReSTIR PT with either hybrid or reconnection shifts outperforms path tracing and BPR in both short and long rendering times. BPR outperforms path tracing except in Zeroday, where BPR's reconnection shifts interact poorly with the shiny multi-layer surfaces. ReSTIR PT with reconnection shifts benefits from the amortization of sample costs when using GRIS, converging much faster than BPR.

Across our scenes, BPR reaches the same MAPE up to 2.8× faster than path tracing, while our ReSTIR PT with the reconnection shift converges up to 14.4× faster, and ReSTIR PT with the hybrid shift converges up to 10× faster. ReSTIR GI gives good quality sampling on diffuse surfaces, but adds bias; its glossy reflections can be low-quality even after accumulating many frames.

Interestingly, our hybrid shift's advantage in real-time rendering does not fully transfer to offline, though both shifts provide a large improvement over alternate methods. The only exception is

[13] We use $\mathrm{MAPE}(I, I_{\mathrm{gt}}) = \mathrm{mean}\left(\frac{|I - I_{\mathrm{gt}}|}{0.01 \cdot \mathrm{mean}(\tilde{I}_{\mathrm{gt}}) + \tilde{I}_{\mathrm{gt}}}\right)$, for $\tilde{I}_{\mathrm{gt}}$ a grayscale ground-truth.

VeachAjar, where the glossy teapots significantly benefit from the hybrid shift mapping in equal-time comparisons.

We postulate the following explanation. While better shift maps improve spatial reuse, they also improve temporal reuse (which improves *future* spatial reuse). Better temporal reuse is thus a very beneficial investment. For offline rendering we disable temporal reuse, losing this extra advantage from improved temporal shift maps (like our hybrid shift).

Equal-time comparisons are impacted by GPU thread divergence, as threads often trace path of different lengths. Comparing convergence at equal sample counts (Figure 18) rather than equal time, our hybrid shift is generally better than the reconnection shift, except for the caustic paths in Figure 13 (bottom). This suggests run-time cost is overcoming our hybrid shift benefits. Divergence can be improved by better workload balancing and other optimizations, which may make our hybrid shift more generally appealing.

10 CONCLUSION

We introduce a generalized RIS theory, extending resampled importance sampling [Talbot 2005] to enable reusing correlated paths, taken from multiple domains (pixels) using context-aware shift mappings. Resampling gives asymptotically perfect importance sampling according to a user-specified target function $\hat{p}$; choosing $\hat{p} = f$ yields asymptotically zero-variance integration. See Figure 3 for a summary of key algorithmic differences versus RIS.

Building on this theory, we reformulate ReSTIR's spatiotemporal reuse to remain consistent and unbiased even for long paths or complex specular transport. As in Bitterli et al. [2020] and Ouyang et al. [2021], our streaming algorithms amortizes path generation over many pixels, leveraging GPU parallelism and temporal path histories to dramatically reduce variance at interactive framerates.

Beyond better robustness from a consistent and unbiased algorithm, we further improve on ReSTIR GI [Ouyang et al. 2021] with a new hybrid shift map that reduces noise. This extends the toolbox of commonly-used building blocks for shift maps by accounting for the sampled BSDF lobe and preselecting connection vertices to limit memory traffic. Somewhat surprisingly, ReSTIR PT can even reuse simple caustic paths in many cases.

Our theory gives conditions under which we can guarantee these improvements. Users must account for domain changes with appropriate MIS weights; avoid unbounded $f/\hat{p}$ ratios; control resampling weight variance Var $[\sum w_i]$; ensure a sufficient number of canonical paths to avoid undersampling parts of the domain; and, if accumulating frames, use a reasonable M-cap to limit temporal correlation.

Based on these constraints, we introduce an interactive ReSTIR PT and show how to properly reuse paths for consistent offline rendering. These redesigned algorithms avoid unpleasant surprises, e.g., cases like Figure 2 where pixels converge slowly, if at all.

10.1 Future work

We believe our GRIS theory will help drive a variety of future research on resampling and path reuse algorithms.

Shift mappings and gradient-domain rendering. Our extended lobe-path space could benefit shift mapping in gradient-domain problems, including for more efficient manifold exploration [Lehtinen et al. 2013]. Manifold exploration could also help where local shift decisions prove limiting to our hybrid shift (e.g., Figure 13, bottom).

Color noise. ReSTIR resamples paths using grayscale target functions $|f|$, which importance samples pixel brightness but not chroma $f/|f|$. Modifications could perhaps resample between wavelengths or use hero wavelengths to help with color noise.

MIS between shift mappings. GRIS can naturally MIS between multiple shift maps, as it supports correlated samples. E.g., we could resample a single sample with multiple shift maps applied, automatically giving more effective shifts higher selection probabilities.

Undersampling. GRIS, ReSTIR, and sample reuse can still suffer from undersampling. In screen space implementations, like ours, reuse may be limited by undersampling of tiny geometry or very high frequency light, e.g., sharp caustics. This can cause noise, streaks, or splotches and requires further investigation.

ACKNOWLEDGMENTS

We want to thank Aaron Lefohn for discussions on and support of this research, including funding via an NVIDIA Professor Partnership to the University of Utah. Additional thanks to Yaobin Ouyang for discussions on ReSTIR GI, and Matt Pharr, Marco Salvi, and Pete Shirley for valuable feedback on early paper drafts.

REFERENCES

Pablo Bauszat, Victor Petitjean, and Elmar Eisemann. 2017. Gradient-Domain Path Reusing. *ACM Trans. Graph.* 36, 6, Article 229 (nov 2017), 9 pages. https://doi.org/10.1145/3130800.3130886

Philippe Bekaert, Mateu Sbert, and John H Halton. 2002. Accelerating Path Tracing by Re-Using Paths.. In *Rendering Techniques*. 125–134.

Nikolaus Binder, Sascha Fricke, and Alexander Keller. 2019. Massively parallel path space filtering. *arXiv preprint arXiv:1902.05942* (2019).

Benedikt Bitterli. 2021. *Correlations and reuse for fast and accurate physically based light transport.* Ph.D. Dissertation. Dartmouth College. http://benedikt-bitterli.me/Data/dissertation.pdf

Benedikt Bitterli, Wenzel Jakob, Jan Novák, and Wojciech Jarosz. 2017. Reversible jump Metropolis light transport using inverse mappings. *ACM Transactions on Graphics (TOG)* 37, 1 (2017), 1–12.

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 (TOG)* 39, 4 (2020), 148–1.

Guillaume Boissé. 2021. World-Space Spatiotemporal Reservoir Reuse for Ray-Traced Global Illumination. In *SIGGRAPH Asia 2021 Technical Communications* (Tokyo, Japan) *(SA '21 Technical Communications)*. Association for Computing Machinery, New York, NY, USA, Article 22, 4 pages. https://doi.org/10.1145/3478512.3488613

Jakub Boksansky, Paula Jukarainen, and Chris Wyman. 2021. Rendering Many Lights with Grid-Based Reservoirs. In *Ray Tracing Gems II*. Springer, 351–365.

Olivier Cappé, Arnaud Guillin, Jean-Michel Marin, and Christian P Robert. 2004. Population monte carlo. *Journal of Computational and Graphical Statistics* 13, 4 (2004), 907–929.

Chakravarty R. Alla Chaitanya, Laurent Belcour, Toshiya Hachisuka, Simon Premoze, Jacopo Pantaleoni, and Derek Nowrouzezahrai. 2018. Matrix Bidirectional Path Tracing. In *Proceedings of the Eurographics Symposium on Rendering: Experimental Ideas & Implementations* (Karlsruhe, Germany) *(SR '18)*. Eurographics Association, Goslar, DEU, 23–32. https://doi.org/10.2312/sre.20181169

Chakravarty R. Alla Chaitanya, Anton S. Kaplanyan, Christoph Schied, Marco Salvi, Aaron Lefohn, Derek Nowrouzezahrai, and Timo Aila. 2017. Interactive Reconstruction of Monte Carlo Image Sequences Using a Recurrent Denoising Autoencoder. *ACM Trans. Graph.* 36, 4, Article 98 (jul 2017), 12 pages. https://doi.org/10.1145/3072959.3073601

Min-Te Chao. 1982. A general purpose unequal probability sampling plan. *Biometrika* 69, 3 (1982), 653–656. https://doi.org/10.2307/2336002

Michael Donikian, Bruce Walter, Kavita Bala, Sebastian Fernandez, and Donald P. Greenberg. 2006. Accurate Direct Illumination Using Iterative Adaptive Sampling. *IEEE Transactions on Visualization and Computer Graphics* 12, 3 (may 2006), 353–364. https://doi.org/10.1109/TVCG.2006.41

Fig. 15. *Comparing path tracing and two variants of our ReSTIR PT (reconnection and hybrid shift) for real-time rendering (all methods use ReSTIR DI for direct light). Images are captured during an animation sequence. MAPE are computed using HDR images.*

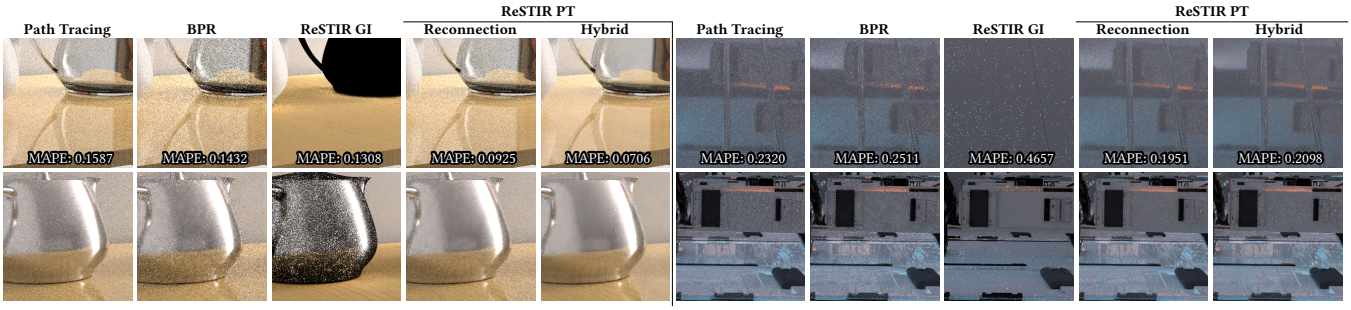


Fig. 16. *Visual comparison in the VeachAjar and ZeroDay scenes using our offline ReSTIR variant, all rendered in 5 seconds.*

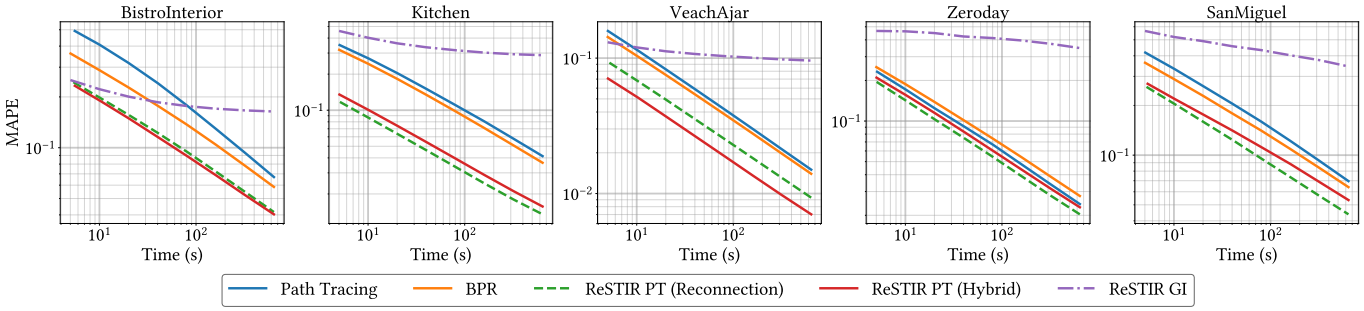


Fig. 17. *Offline rendering error comparison in terms of time spent on rendering (5-640 second range).*

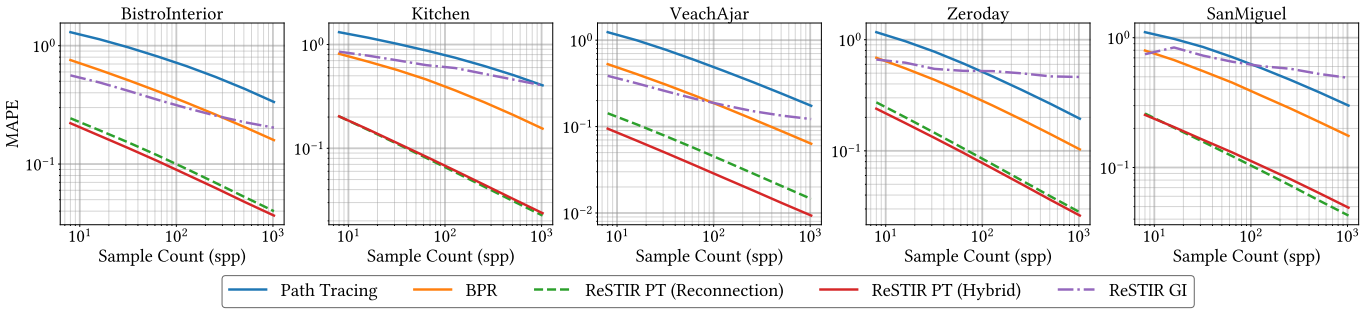


Fig. 18. *Offline rendering error comparison in terms of sample count.*

Michaël Gharbi, Tzu-Mao Li, Miika Aittala, Jaakko Lehtinen, and Frédo Durand. 2019. Sample-Based Monte Carlo Denoising Using a Kernel-Splatting Network. *ACM Trans. Graph.* 38, 4, Article 125 (jul 2019), 12 pages. https://doi.org/10.1145/3306346.3322954

Abhijeet Ghosh, Arnaud Doucet, and Wolfgang Heidrich. 2006. Sequential Sampling for Dynamic Environment Map Illumination. In *Symposium on Rendering*, Tomas Akenine-Moeller and Wolfgang Heidrich (Eds.). The Eurographics Association. https://doi.org/10.2312/EGWR/EGSR06/115-126

Adrien Gruson, Binh-Son Hua, Nicolas Vibert, Derek Nowrouzezahrai, and Toshiya Hachisuka. 2018. Gradient-domain volumetric photon density estimation. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–13.

Adam Guetz. 2012. *Monte Carlo Methods for Structured Data.* Stanford University.

Henrik Halen, Andreas Brinck, Kyle Hayward, and Xiangshun Bei. 2021. Global Illumination Based on Surfels. In *SIGGRAPH Courses; Advances in Real-Time Rendering.*

Jon Hasselgren, Jacob Munkberg, Marco Salvi, Anjul Patney, and Aaron Lefohn. 2020. Neural Temporal Adaptive Sampling and Denoising. *Computer Graphics Forum* (2020). https://doi.org/10.1111/cgf.13919

Paul S Heckbert. 1990. Adaptive radiosity textures for bidirectional ray tracing. In *Proceedings of the 17th annual conference on Computer graphics and interactive techniques.* 145–154.

E. Heitz and L. Belcour. 2019. Distributing Monte Carlo Errors as a Blue Noise in Screen Space by Permuting Pixel Seeds Between Frames. *Computer Graphics Forum* 38, 4 (2019), 149–158. https://doi.org/10.1111/cgf.13778

Eric Heitz, Stephen Hill, and Morgan McGuire. 2018. Combining Analytic Direct Illumination and Stochastic Shadows. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games.* Association for Computing Machinery, New York, NY, USA, Article 2, 11 pages. https://doi.org/10.1145/3190834.3190852

Binh-Son Hua, Adrien Gruson, Derek Nowrouzezahrai, and Toshiya Hachisuka. 2017. Gradient-domain photon density estimation. In *Computer Graphics Forum*, Vol. 36. Wiley Online Library, 31–38.

Binh-Son Hua, Adrien Gruson, Victor Petitjean, Matthias Zwicker, Derek Nowrouzezahrai, Elmar Eisemann, and Toshiya Hachisuka. 2019. A Survey on Gradient-Domain Rendering. In *Computer Graphics Forum*, Vol. 38. Wiley Online Library, 455–472.

Wenzel Jakob and Steve Marschner. 2012. Manifold exploration: A markov chain monte carlo technique for rendering scenes with difficult specular transport. *ACM Transactions on Graphics (TOG)* 31, 4 (2012), 1–13.

Henrik Wann Jensen. 2001. *Realistic Image Synthesis Using Photon Mapping.* A. K. Peters, Ltd., USA.

Simon Kallweit, Petrik Clarberg, Craig Kolb, Kai-Hwa Yao, Theresa Foley, Yong He, Lifan Wu, Lucy Chen, Tomas Akenine-Moller, Chris Wyman, Cyril Crassin, and Nir Benty. 2021. The Falcor Rendering Framework. https://github.com/NVIDIAGameWorks/Falcor https://github.com/NVIDIAGameWorks/Falcor.

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
Markus Kettunen. 2020. *Gradient-Domain Methods for Realistic Image Synthesis.* Ph.D. Dissertation. Aalto University.
Markus Kettunen, Marco Manzi, Miika Aittala, Jaakko Lehtinen, Frédo Durand, and Matthias Zwicker. 2015. Gradient-domain path tracing. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 1–13.
Emmett Kilgariff, Henry Moreton, Nick Stam, and Brandon Bell. 2018. NVIDIA Turing Architecture In-Depth. https://developer.nvidia.com/blog/nvidia-turing-architecture-in-depth/. [Online; accessed 9-December-2021].
Eric P. Lafortune and Yves D. Willems. 1993. Bi-Directional Path Tracing. In *Proc. the International Conference on Computational Graphics and Visualization Techniques*, Vol. 93. 145–153.
Yu-Chi Lai, Shao Hua Fan, Stephen Chenney, and Charcle Dyer. 2007. Photorealistic image rendering with population monte carlo energy redistribution. In *Proceedings of the 18th Eurographics conference on Rendering Techniques.* 287–295.
Jaakko Lehtinen, Tero Karras, Samuli Laine, Miika Aittala, Frédo Durand, and Timo Aila. 2013. Gradient-domain metropolis light transport. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 1–12.
Faming Liang and Sooyoung Cheon. 2009. Monte Carlo dynamically weighted importance sampling for spatial models with intractable normalizing constants. 197 (dec 2009), 012004. https://doi.org/10.1088/1742-6596/197/1/012004
Daqi Lin, Chris Wyman, and Cem Yuksel. 2021. Fast Volume Rendering with Spatiotemporal Reservoir Resampling. *ACM Transactions on Graphics (TOG)* 40, 6 (2021), to appear.
Jun S Liu and Jun S Liu. 2001. *Monte Carlo strategies in scientific computing*. Vol. 10. Springer, 36–37.
Marco Manzi, Markus Kettunen, Miika Aittala, Jaakko Lehtinen, Frédo Durand, and Matthias Zwicker. 2015. Gradient-domain bidirectional path tracing. (2015).
Marco Manzi, Markus Kettunen, Frédo Durand, Matthias Zwicker, and Jaakko Lehtinen. 2016. Temporal gradient-domain path tracing. *ACM Transactions on Graphics (TOG)* 35, 6 (2016), 1–9.
Marco Manzi, Fabrice Rousselle, Markus Kettunen, Jaakko Lehtinen, and Matthias Zwicker. 2014. Improved sampling for gradient-domain metropolis light transport. *ACM Transactions on Graphics (TOG)* 33, 6 (2014), 1–12.
Pierre Moreau, Matt Pharr, and Petrik Clarberg. 2019. Dynamic Many-Light Sampling for Real-Time Ray Tracing. In *High-Performance Graphics - Short Papers*, Markus Steinberger and T. Foley (Eds.). The Eurographics Association. https://doi.org/10.2312/hpg.20191191
Thomas Müller, Brian Mcwilliams, Fabrice Rousselle, Markus Gross, and Jan Novák. 2019. Neural Importance Sampling. *ACM Trans. Graph.* 38, 5, Article 145 (oct 2019), 19 pages. https://doi.org/10.1145/3341156
Thomas Müller, Markus Gross, and Jan Novák. 2017. Practical Path Guiding for Efficient Light-Transport Simulation. *Computer Graphics Forum* 36, 4 (2017), 91–100. https://doi.org/10.1111/cgf.13227
Kosuke Nabata, Kei Iwasaki, and Yoshinori Dobashi. 2020. Resampling-Aware Weighting Functions for Bidirectional Path Tracing Using Multiple Light Sub-Paths. *ACM Trans. Graph.* 39, 2, Article 15 (mar 2020), 11 pages. https://doi.org/10.1145/3338994
Yaobin Ouyang, Shiqiu Liu, Markus Kettunen, Matt Pharr, and Jacopo Pantaleoni. 2021. ReSTIR GI: Path Resampling for Real-Time Path Tracing. *Computer Graphics Forum* 40, 8 (2021), 17–29. https://doi.org/10.1111/cgf.14378
Jacopo Pantaleoni. 2020. Online Path Sampling Control with Progressive Spatiotemporal Filtering. *SN Computer Science* 279 (aug 2020).
Christoph Peters. 2021. BRDF Importance Sampling for Polygonal Lights. *ACM Trans. Graph.* 40, 4, Article 140 (jul 2021), 14 pages. https://doi.org/10.1145/3450626.3459672
Victor Petitjean, Pablo Bauszat, and Elmar Eisemann. 2018. Spectral Gradient Sampling for Path Tracing. In *Computer Graphics Forum*, Vol. 37. Wiley Online Library, 45–53.
Stefan Popov, Ravi Ramamoorthi, Fredo Durand, and George Drettakis. 2015. Probabilistic Connections for Bidirectional Path Tracing. *Comput. Graph. Forum* 34, 4 (jul 2015), 75–86.
DB Rubin. 1987. A Noniterative Sampling/Importance resampling alternative to data augmentation for creating a few imputations when fractions of missing information are modest: The SIR algorithm. *J. Amer. Statist. Assoc.* 82 (1987), 544–546.
Christoph Schied, Christoph Peters, and Carsten Dachsbacher. 2018. Gradient Estimation for Real-Time Adaptive Temporal Filtering. *Proc. ACM Comput. Graph. Interact. Tech.* 1, 2, Article 24 (2018). https://doi.org/10.1145/3233301
Peter Shirley, Changyaw Wang, and Kurt Zimmerman. 1996. Monte Carlo Techniques for Direct Lighting Calculations. *ACM Trans. Graph.* 15, 1 (jan 1996), 1–36. https://doi.org/10.1145/226150.226151
Tomasz Stachowiak. 2015. Stochastic Screen-Space Reflections. In *Advances in Real Time Rendering, (ACM SIGGRAPH Courses).* https://doi.org/10.1145/2776880.2787701
Weilun Sun, Xin Sun, Nathan A Carr, Derek Nowrouzezahrai, and Ravi Ramamoorthi. 2017. Gradient-Domain Vertex Connection and Merging.. In *EGSR (EI&I).* 83–92.
László Szécsi, László Szirmay-Kalos, and Csaba Kelemen. 2003. Variance reduction for Russian-roulette. (2003).
Justin Talbot, David Cline, and Parris Egbert. 2005. Importance Resampling for Global Illumination. In *Eurographics Symposium on Rendering (2005)*, Kavita Bala and Philip Dutre (Eds.). The Eurographics Association. https://doi.org/10.2312/EGWR/EGSR05/139-146
Justin F Talbot. 2005. *Importance resampling for global illumination.* Brigham Young University.
Lorenzo Tessari, Johannes Hanika, and Carsten Dachsbacher. 2017. Local quasi-monte carlo exploration. In *Proceedings of the Eurographics Symposium on Rendering: Experimental Ideas & Implementations.* 71–81.
Yusuke Tokuyoshi and Takahiro Harada. 2019. Hierarchical Russian Roulette for Vertex Connections. *ACM Trans. Graph.* 38, 4, Article 36 (jul 2019), 12 pages. https://doi.org/10.1145/3306346.3323018
Joran Van de Woestijne, Roald Frederickx, Niels Billen, and Philip Dutré. 2017. Temporal coherence for metropolis light transport. In *Eurographics Symposium on Rendering-Experimental Ideas & Implementations.* Eurographics Association, 55–63.
Eric Veach. 1998. *Robust Monte Carlo methods for light transport simulation.* Stanford University.
Eric Veach and Leonidas J. Guibas. 1997. Metropolis Light Transport. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques.* USA, 65–76. https://doi.org/10.1145/258734.258775
Petr Vévoda, Ivo Kondapaneni, and Jaroslav Křivánek. 2018. Bayesian Online Regression for Adaptive Direct Illumination Sampling. *ACM Trans. Graph.* 37, 4, Article 125 (jul 2018), 12 pages. https://doi.org/10.1145/3197517.3201340
Jiří Vorba, Johannes Hanika, Sebastian Herholz, Thomas Müller, Jaroslav Křivánek, and Alexander Keller. 2019. Path Guiding in Production. In *ACM SIGGRAPH 2019 Courses* (Los Angeles, California) *(SIGGRAPH '19).* ACM, New York, NY, USA, Article 18, 77 pages. https://doi.org/10.1145/3305366.3328091
Jiří Vorba, Ondřej Karlík, Martin Šik, Tobias Ritschel, and Jaroslav Křivánek. 2014. On-line learning of parametric mixture models for light transport simulation. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–11.
Bruce Walter, Sebastian Fernandez, Adam Arbree, Kavita Bala, Michael Donikian, and Donald P. Greenberg. 2005. Lightcuts: A Scalable Approach to Illumination. 24, 3 (jul 2005), 1098–1107. https://doi.org/10.1145/1073204.1073318
Bruce Walter, Stephen R Marschner, Hongsong Li, and Kenneth E Torrance. 2007. Microfacet Models for Refraction through Rough Surfaces. *Rendering techniques* 2007 (2007), 18th.
Rex West, Iliyan Georgiev, Adrien Gruson, and Toshiya Hachisuka. 2020. Continuous multiple importance sampling. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 136–1.
Chris Wyman and Alexey Panteleev. 2021. Rearchitecting Spatiotemporal Resampling for Production. In *ACM/EG Symposium on High Perfrormance Graphics.* 23–41. https://doi.org/10.2312/hpg.20211281
Junqiu Zhu, Yaoyi Bai, Zilin Xu, Steve Bako, Edgar Velázquez-Armendáriz, Lu Wang, Pradeep Sen, Miloš Hašan, and Ling-Qi Yan. 2021. Neural Complex Luminaires: Representation and Rendering. *ACM Trans. Graph.* 40, 4, Article 57 (jul 2021), 12 pages. https://doi.org/10.1145/3450626.3459798

A THEOREMS

In this appendix we present the most important mathematical theorems for generalized RIS. The proofs are found in the supplemental document, Section S.5. Convergence with dependent samples is proved in Section S.2.

A.1 Unbiased Contribution Weights, Section 4.2

THEOREM A.1. *Let X and real-valued W be random variables in Ω. The following are equivalent:*

(1) For all integrable $f : \Omega \to \mathbb{R}$,

$$\mathbb{E}\left[f(X)W\right] = \int_{\mathrm{supp}(X)} f(x)\,\mathrm{d}x, \tag{55}$$

(2) W is an unbiased estimator for X's reciprocal marginal density,

$$\mathbb{E}\left[W \mid X\right] = \frac{1}{p_X(X)}. \tag{56}$$

PROOF. Section S.5.1. □

A.2 Asymptotic Sample Distribution, Section 4.4

Theorem A.2 (Asymptotic Sample Distribution). *Assume, for each M (separately, starting from some M_0) a sequence of samples $(X_i \in \Omega_i)_{i=1}^{M}$ (we omit the index M for brevity), and that we resample Y_M $(=T_{s_M}(X_{s_M}))$ proportionally to weights $w_{M,i}$ given by Equation 19. Assume also that the generated samples cover the support of $\hat{p}$,*

$$\operatorname{supp}\hat{p} \subset \operatorname{supp} Y_M \qquad \text{when } M \geq M_0. \tag{57}$$

If the variance of the weight sums tends to zero, i.e.,

$$\operatorname{Var}\left[\sum_{i=1}^{M} w_{M,i}\right] \xrightarrow{M\to\infty} 0, \tag{58}$$

Then,

(1) p_Y converges to $\bar{p}$ in probability, i.e., for any $\varepsilon > 0$

$$\Pr[|p_Y(Y) - \bar{p}(Y)| > \varepsilon] \xrightarrow{M\to\infty} 0. \tag{59}$$

(2) the density ratio $\bar{p}(Y)/p_Y(Y)$ approaches 1 in mean square, i.e.,

$$\mathbb{E}\left[\left|\frac{\bar{p}(Y)}{p_Y(Y)} - 1\right|^2\right] \xrightarrow{M\to\infty} 0. \tag{60}$$

(3) the integral of the absolute error of p_Y from $\bar{p}$ approaches 0, i.e.,

$$\int_\Omega |p_Y(y) - \bar{p}(y)|\,\mathrm{d}y \xrightarrow{M\to\infty} 0. \tag{61}$$

(4) in the set in which $p_Y(y)$ converges, it converges to $\bar{p}(y)$ (except for a possible set of zero measure),

(5) each subset of Ω will asymptotically get the correct ratio of samples.

Proof. Section S.5.2. □

A.3 Asymptotic Variance, Section 5.2

Theorem A.3 (Asymptotic Variance). *In addition to the assumptions of Theorem A.2, assume that $f \geq 0$ and*

$$f \leq C_f \hat{p} \qquad \text{for some } C_f > 0. \tag{62}$$

Then, the generated samples Y cover the support of $\hat{p}$ and f, and

(1) $f(Y)W_Y$ approaches $f(Y)/\bar{p}(Y)$ in mean, mean square and probability, i.e.,

$$\mathbb{E}\left[\left|f(Y)W_Y - \frac{f(Y)}{\bar{p}(Y)}\right|^p\right] \xrightarrow{M\to\infty} 0 \quad \text{for } p = 1, 2, \quad \text{and} \tag{63}$$

$$\Pr\left[\left|f(Y)W_Y - \frac{f(Y)}{\bar{p}(Y)}\right| > \varepsilon\right] \xrightarrow{M\to\infty} 0 \quad \text{for all } \varepsilon > 0. \tag{64}$$

(2) $f(Y)W_Y$ has asymptotically the variance of $f(X)/\bar{p}(X)$ where X has density $\bar{p}(X)$,

$$\operatorname{Var}[f(Y)W_Y] \xrightarrow{M\to\infty} \operatorname{Var}\left[\frac{f(X)}{\bar{p}(X)}\right]. \tag{65}$$

(3) If $\hat{p}(x) \propto f(x)$, then also

$$\operatorname{Var}[f(Y)W_Y] \xrightarrow{M\to\infty} 0. \tag{66}$$

Proof. Section S.5.3. □

A.4 Resampling Weight Bounds, Section 5.6

Theorem A.4 (Resampling Weight Bounds). *Let the resampling weights w_i of input samples X_i be given by Equation 19, and associate all source domains Ω_i with target distributions $\hat{p}_i$. Let R be the indices of the canonical samples, and assume $|R| \geq 1$.*

If m_i are given by one of the MIS weight schemes defined in Equations 36, 37 or 38, and sample X_i is reasonably distributed for integrating $\hat{p}_i$ (i.e., $\hat{p}_i(X_i)W_i \leq C_i$ for some C_i), then the resampling weight of X_i is bounded as

$$w_i \leq \frac{C_i}{|R|}. \tag{67}$$

Proof. Section S.1.3. □

B CORRECTNESS NOTES

This section discusses some aspects of ReSTIR that may affect performance and correctness.

On Visibility. Relating to the discussion in Section 6.1, ReSTIR DI [Bitterli et al. 2020] used a target $\hat{p}_i$ without the visibility term $V(\mathbf{x}_1 \leftrightarrow \mathbf{x}_2)$. Our ReSTIR PT always considers visibility between vertices. In fact, neglecting visibility makes maintaining convergence guarantees tricky, as it creates paths with positive $\hat{p}_i$ that never get sampled as X_i due to occlusion. This implies $\operatorname{supp}\hat{p}_i \not\subset \operatorname{supp} X_i$, making X_i non-canonical.

Without extra guarantees, Y resampled from X_i no longer covers $\operatorname{supp}\hat{p}_i$, breaking the $\operatorname{supp}\hat{p} \subset \operatorname{supp} Y_M$ assumption of Equation 23 and preventing convergence of p_Y to $\bar{p}$.

ReSTIR DI circumvents this by closely tracking reuse throughout each algorithmic phase. While its $\hat{p}_i$ for temporal reuse checks for visibility, its spatial reuse uses an unoccluded target $\hat{p}_i^{-V}$ to reduce cost. Without full coverage of $\hat{p}_i^{-V}$'s domain, intermediate distributions never converge. The design still ensures coverage of f_i's domain, allowing *final* estimators to remain unbiased. For direct lighting, earlier convergence to $\hat{p}_i$ may only be of theoretic interest, with Bitterli et al.'s [2020] choices working around a bottleneck from visibility costs. In path tracing, ignoring visibility requires much more engineering, as typical path samplers never select occluded paths with $V(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i+1}) = 0$.

Temporal Reuse. Temporal sample reuse is unbiased with proper MIS weights, e.g., our generalized Talbot or pairwise MIS, but a temporal shift mapping is needed, as evaluating MIS weights for GRIS requires bijectively shifting paths between the prior and current frames. In some cases, e.g., conflicting motion vectors, careful map definitions may be needed to retain bijectivity.

This constraint means fully unbiased temporal reuse must evaluate paths in both the current and prior frames, which is tricky in dynamic environments. Biased approximations to temporal MIS can be used, e.g., neglecting visibility [Ouyang et al. 2021], which gives desirable performance improvements for often imperceptible bias. Lin et al. [2021] explicitly account for temporal changes, reporting it reduces response time to dynamic lighting.