

Optimised Path Space Regularisation

Philippe Weier, Marc Droske, Johannes Hanika, Andrea Weidlich, Jiří Vorba



Figure 1: Our method regularises complex paths, such as the depicted refracted caustics, to drastically reduce the variance of path tracing estimators in exchange for a small systematic error. A single hyperparameter β controls the user-defined level of tolerated noise during an offline optimisation process and offers different bias-variance trade-offs during rendering. The left and right image show how the noise and convergence time for path tracing and path guiding, respectively, are improved when applying our method with two distinct sets of learnt parameters. We render the right image to achieve equal noise using an adaptive sampler, whereas the left image has a fixed count of 512 samples per pixels.

Abstract

We present Optimised Path Space Regularisation (OPSR), a novel regularisation technique for forward path tracing algorithms. Our regularisation controls the amount of roughness added to materials depending on the type of sampled paths and trades a small error in the estimator for a drastic reduction of variance in difficult paths, including indirectly visible caustics. We formulate the problem as a joint bias-variance minimisation problem and use differentiable rendering to optimise our model. The learnt parameters generalise to a large variety of scenes irrespective of their geometric complexity. The regularisation added to the underlying light transport algorithm naturally allows us to handle the problem of near-specular and glossy path chains robustly. Our method consistently improves the convergence of path tracing estimators, including state-of-the-art path guiding techniques where it enables finding otherwise hard-to-sample paths and thus, in turn, can significantly speed up the learning of guiding distributions.

CCS Concepts

• Computing methodologies → Rendering;

1. Introduction

Realistically representing and understanding our world is a challenge that various fields have been hard at work on. In computer graphics, particularly for physically-based rendering, this translates to algorithms that can simulate the light transport with such accu-

racy that it can be hard to distinguish between a synthetic image and a photograph. However, very often, there is a trade-off between algorithmic complexity and the range of phenomena we can simulate efficiently. Among the many techniques that try to solve the light transport problem, path tracing is ubiquitous in production;

its success relies mainly on its simplicity and ease of integration in rendering systems. However, physical phenomena such as visible or indirectly visible focused light through specular interfaces, also called caustics, can be extremely challenging to simulate efficiently.

Techniques such as bidirectional path tracing [VG95a], photon-mapping [Jen96] or vertex-connection and merging [GKDS12] can be suitable for rendering some or all of the aforementioned light effects. However, straightforward algorithms such as path tracing are preferred in production systems over more involved techniques which are often fragile and less flexible. Improving the convergence of renders while maintaining visual fidelity has been a challenge since the first path-traced movies arrived on our screens [DH09]. Allowing some error in the final image to reduce variance can be a powerful choice when it enables rendering scenes that, previously, would have been impossible to render in a reasonable amount of time. Therefore, many production systems apply a form of *regularisation* called *roughening* [DH09; FHP*18], which consists in adding roughness to materials to ease the sampling of difficult paths. Kaplanyan et al. [KD13] introduced a similar kind of regularisation based on the use of *mollifiers* [HA98] which allows the sampling of previously *unsamplable paths* via a biased but *consistent* estimation. Unfortunately, adding roughness to materials can introduce a fair amount of bias in the rendered image.

Introducing the right amount of roughness to reduce variance while keeping the error low is central to our work. We leverage differentiable rendering [LADL18; NVZJ19] to model the problem as a joint bias-variance minimisation and selectively regularise vertices along specific paths during light transport. While the estimator resulting from using our roughening model is biased, we demonstrate that it can be made consistent in presence of mollifiers. Furthermore, our regularisation scheme can work on arbitrary scenes without the need to re-run an optimisation process, allowing a straightforward implementation in any forward path-tracer.

More concisely, our contributions are :

- A novel path space regularisation technique that depends on the type of sampled path and improves the sampling of complex caustics and glossy path chains (Section 4.1).
- A lightweight roughening model optimised entirely offline for the desired bias-variance trade-off (Section 4.4).
- The derivation of a consistent estimator based on mollifiers that uses our proposed regularisation strategy. (Section 4.2)

2. Related Work

The first unbiased Monte Carlo integration strategy to solve light transport was proposed by Kajia [Kaj86] using a single integral formulation known as the rendering equation. Veach later showed how different strategies could be optimally combined using multiple importance sampling (MIS) [VG95b]. This allowed the design of robust estimators in the context of forward path tracing by combining both the importance sampling of the incident light at a point as well as the bidirectional scattering function distribution (BSDF). Bidirectional path tracing [VG95a; LW98] is also the result of combining both importance and light tracing into a single algorithm capable of efficiently resolving directly visible caustics while not impeded by the typical restrictions that light tracing imposes.

Unfortunately, sampling indirectly seen caustics or so-called "SDS" paths, is still as hard for bidirectional path tracing as for a unidirectional estimator since the singularities due to specular interfaces now appear in both light transport directions. Photon mapping [Jen96] is a two-pass algorithm; photons landing on diffuse surfaces are stored in a tree-like data structure and later, during a forward path tracing pass, the incident radiance at the camera is computed using kernel-based radiance estimation. Progressive photon mapping [HOJ08] and its stochastic variant [HJ09; KZ11] lead to consistent estimators by reducing the radius of the kernel as the number of samples used increases. Unfortunately, for large scenes with highly detailed geometry, the large number of photons needed can become impractical. Furthermore, in regions of the scene where no caustics are visible, the photon pass might be unnecessary and slower compared to simple path tracing [GPSK18].

Path guiding is a variance reduction technique that aids in difficult light transport conditions. Its exceptional versatility in solving complex light simulations, even using a simple unidirectional path tracer, has made it a strong choice in any renderer [VHH*19]. Path guiding techniques learn an incoming light field distribution at every point in the scene to importance sample the indirect illumination and effectively *guide* the traced rays towards light sources. Vorba et al. [VKS*14] use a mixture of Gaussians to learn directional distributions from both importons and photons in a back and forth online learning strategy. Müller et al. [MGN17] learn from the forward path traced particles only and use an efficient quadtree structure to represent the directional distributions providing a practical application for forward path tracing algorithms. Recent works also focused on guiding the product of the BSDF and the indirect light using mixtures of Gaussians [HEV*16] or deep neural networks [MMR*18].

Techniques such as manifold next event estimation (MNEE) [HDF15] or, more recently, specular manifold sampling (SMS) [ZGJ20] can be used in classical path tracing to deterministically find intermediate vertices that obey Fermat's principle such as to form an admissible path for next event estimation. While MNEE only handles refraction through specular interfaces, SMS generalises the concept to allow the sampling of both reflected and refracted caustics in a unified framework. Unfortunately, the Newton solver used by those techniques can be subject to numerical instabilities and, while it can generalise to glossy interfaces, the overhead of performing a manifold walk in that case often no longer compensates for its variance reduction effect.

The regularisation technique introduced by Kaplanyan et al. [KD13] allows the sampling of previously unsamplable paths due to singularities arising in the integrand. This is achieved by introducing mollifiers of the Dirac delta distributions to regularise light transport. In a similar setting, Jendersie et al. [JG19] apply a set of heuristics to reduce the bias introduced when using mollifiers for the particular case of micro-facet distributions. Regularisation can also be combined with other strategies using MIS [BIOP13] or in more complex light transport algorithms such as Markov Chain Monte Carlo [ŠK16]. Our work is closely related to the path space regularisation technique introduced by Kaplanyan et al. and can be seen as an extension of their work where the initial mollification radius is learnt depending on the sampled type of path in the scene.

3. Preliminaries

In this section, we recall key concepts underlying our proposed regularisation method. This includes the path space formulation of the rendering equation, singularities in path space, an introduction to differentiable rendering and the definition of mollifiers.

3.1. Path Space Formulation

Veach defines the space of paths \bar{x} of length k segments on the scene manifold \mathcal{M} :

$$\mathcal{P}_k = \left\{ \bar{x} = (x_0, x_1, \dots, x_k) \mid (x_0, x_1, \dots, x_k) \in \mathcal{M}^{k+1} \right\}$$

The union of all such space forms the *path space* $\mathcal{P} = \bigcup_{k=1}^{\infty} \mathcal{P}_k$. The *measurement* M , typically a pixel value, in the path space integral formulation [Vea97] can then be written as

$$M = \int_{\mathcal{P}} m(\bar{x}) d\mu(\bar{x}) \quad (1)$$

The *measurement contribution function* $m(\bar{x})$ is defined as

$$m(\bar{x}) = W_e(x_0, x_1) T(\bar{x}) L_e(x_k, x_{k-1}) \quad (2)$$

where W_e is the response of the sensor at x_0 , $T(\bar{x})$ the path's throughput and $L_e(x_k, x_{k-1})$ the emitted radiance at vertex x_k towards x_{k-1} . The throughput corresponds to the product of the reflectance functions ρ along a path:

$$T(\bar{x}) = G(x_0, x_1) \prod_{j=1}^{k-1} \rho(x_{j-1}, x_j, x_{j+1} \mid \alpha_j) G(x_j, x_{j+1})$$

where we assume $\alpha_j \in \mathbb{R}$ to be the roughness extracted from the material description at the j -th vertex, and $G(x_j, x_{j+1})$ is the *geometric term* accounting for the conversion from solid angle to area measure as well as the visibility term.

In the rest of this paper, we will assume, without loss of generality, that the reflectance function can always be considered diffuse above a certain roughness threshold, and zero roughness indicates a perfectly specular BSDF. Moreover, we suppose paths always start at the sensor with vertex x_0 .

3.2. Singularities in path tracing

Veach [Vea97] extended the regular expression notation introduced by Heckbert [Hec90] for light paths with two more symbols describing the directional and spatial properties of sensors and emitters (see Figure 2). He describes a complete path using a *full-path regular expression*:

$$E(S|D)(S|D)(S|D)^*(S|D)(S|D)L$$

where each end-point was assigned a colour for clarity. With the help of this notation, Veach proved in his thesis the necessary condition for a path to be samplable when using only local sampling strategies such as BSDF sampling or next event estimation:

Theorem 1 Let \bar{x} be a path generated by a local sampling algorithm for which the measurement contribution function $m(\bar{x})$ is non-zero. Then \bar{x} necessarily has the form $E(S|D)^*DD(S|D)^*L$, i.e. it must contain the substring DD . Furthermore, it is possible to generate any path of this form using local sampling strategies.

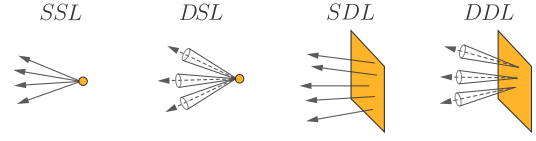


Figure 2: Extended Heckbert notation for emitters. The different regular expressions that can describe an emitter. For example, *SSL* describes a point light with emission over a set of directions with measure zero.

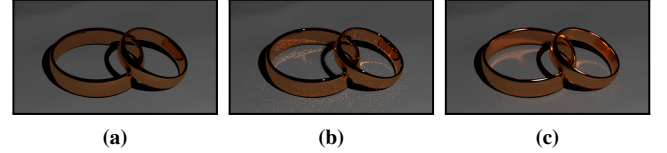


Figure 3: Rendering of a scene with a perfectly specular material and a point light source. Part of the light transport is missing when using classical path tracing with NEE (a). The missing caustics appear if we slightly increase the size of the area light (b) or the roughness of the material (c).

Note that for a pinhole camera (*ESD*) and a point light source (*DSL*), a directly visible caustic has the form $ESDD S^+ DSL$, which contains the substring DD and is thus samplable. Even if path tracing with NEE cannot sample such a path (Figure 3a), a light tracing algorithm could use next event estimation at the sensor to sample this kind of path.

A key observation is that, as seen in Figure 3b and 3c, directly and indirectly visible caustics can both be sampled if we regularise any of the specular interaction after a diffuse vertex in the path [KD13].

3.3. Mollifier

A d -dimensional *mollifier* is a sequence of smooth positive distributions $\{\varphi_r(x)\}$ where r denotes the *mollification bandwidth* [HA98], and for which the support has to vanish as r approaches zero and the sequence converges to a Dirac delta distribution. This is mathematically formulated as follows [KD13]:

$$\left. \begin{aligned} \|\text{supp}(\varphi_r)\| &\propto O(r^d) \\ \int_{\mathbb{R}^d} \varphi_r(x) dx &= 1 \end{aligned} \right\} \Rightarrow \lim_{r \rightarrow 0} \varphi_r(x) = \delta(x) \quad (3)$$

Kaplanyan et al. also show that to achieve a consistent estimator in a Monte Carlo framework, the bandwidth r , which goes to zero in the limit, depends on the index of the sample n used for the integration and must satisfy

$$O(n^{-1/d}) < r_n < O(1)$$

A valid sequence they propose is

$$r_n = r_0 n^{-\lambda} \quad \text{with } \lambda \in (0; 1/d)$$

for a d -dimensional mollifier and an initial bandwidth r_0 .

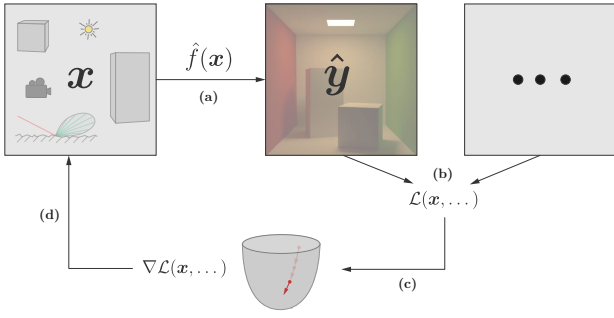


Figure 4: Overview of the pipeline of a differentiable renderer. A model \hat{f} is used to predict some output data \hat{y} (a) (e.g. a rendering algorithm producing an image). We use the output of the model and some additional training data (e.g. a reference image) to compute an objective function (b). Finally, the gradients of the differentiable parameters are computed (c) and used to update the differentiable parameters for the next round (d).

3.4. Differentiable Rendering

Given a single vector \mathbf{x} that represents a desired set of scene parameters (e.g. material parameters, vertex positions, etc.) we are interested in a model \hat{f} , where we use the hat notation to indicate that f is an estimator, that closely approximates the training data \mathbf{y} . This optimisation problem corresponds to a minimisation of the error between the data \mathbf{y} and the model $\hat{f}(\mathbf{x}) = \hat{\mathbf{y}}$:

$$\mathbf{x}_{\text{opt}} = \arg \min_{\mathbf{x}} \mathcal{L}(\hat{f}(\mathbf{x}), \mathbf{y})$$

where \mathcal{L} is called the *objective function* or *loss function* (in the context of data science). The loss usually describes some error between the reference data \mathbf{y} and the model $\hat{f}(\mathbf{x})$ but could in practice represent anything.

Using a system like Redner [LADL18] or the one we use in our work, Mitsuba 2 [NVZJ19], we can propagate derivatives across an entire rendering engine using either manually derived derivatives or automatic differentiation. Together with an appropriate objective function we can formulate a typical gradient descent algorithm; a gradient descent update for the set of optimisable parameters \mathbf{x} at iteration t is defined as

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta_t \cdot \nabla \mathcal{L}(\mathbf{x}_t)$$

where η_t is the *step-size* (or learning rate) used at iteration t . An overview of a typical pipeline is illustrated in Figure 4.

4. Method

In practice, almost specular materials with roughness close to zero introduce significant variance due to so-called *near-singularities*, and can lead to problems similar to the one described in Section 3.2. Therefore, we want to distinguish scattering events on the continuum of roughness to handle those cases appropriately. Concretely, a material with roughness close to zero should be handled similarly to a perfectly specular material, and vertices with high roughness should be considered diffuse above a certain threshold. To better

handle those cases, our regularisation uses the notion of *accumulated roughness* which, intuitively, propagates the roughness present in a path to the last surface interaction, and, in turn, increases the likelihood of a successful connection to a light source. With this approach, the roughening we introduce also depends on the path prefix and automatically handles near-specular and perfectly specular interactions similarly. On the other hand, our approach does not allow the regularisation of path *solely* formed of perfectly specular vertices as no roughness can be accumulated in that case. Handling this specific type of path is orthogonal to our work but could easily be addressed using mollifiers of the Dirac distributions as was done by Kaplanyan et al. [KD13].

We describe the mathematical tools to describe the accumulated roughness in Section 4.1 and analyse the inherent bias-variance trade-off that arises from using our regularisation in Section 4.3. Section 4.4 presents a model to optimise our regularisation strategy and its associated cost function, which exposes a single hyperparameter to control the desired variance threshold. Finally, we discuss practical aspects of our model and demonstrate how the learnt parameters can be used in existing path tracers without further training.

4.1. Accumulated Roughness

Monotonicity For what follows, we define the partial order \leq to be the elementwise inequality:

$$\mathbf{v} \leq \mathbf{w} \text{ iff } v_i \leq w_i \forall i \text{ with } \mathbf{v}, \mathbf{w} \in \mathbb{R}^n$$

where v_i and w_i denote the i -th component of \mathbf{v} and \mathbf{w} , respectively. This allows us to specify a continuous function $g : \mathbb{R}^n \mapsto \mathbb{R}$ such that g is monotone with respect to the partial order \leq , that is

$$\mathbf{v} \leq \mathbf{w} \Rightarrow g(\mathbf{v}) \leq g(\mathbf{w}) \quad (4)$$

Accumulated Roughness Function For any path $\bar{\mathbf{x}} \in \mathcal{P}_k$ of length k we define the *path roughness* as

$$\bar{\alpha}_{\bar{\mathbf{x}}} = \bar{\alpha} = (\alpha_1, \dots, \alpha_{k-1}) \quad (5)$$

where the endpoints have no associated roughness and α_i is the roughness associated with the i -th vertex reflectance function. Now, for the function g described earlier to represent an *accumulated roughness function*, we add the following property:

$$g((0, \dots, 0, v_n)) = v_n \quad (6)$$

which, intuitively, ensures that we only propagate the roughness along $n - 1$ interactions. We can now define the *regularised path roughness* as

$$\bar{\alpha}' = (\alpha_1, \dots, \alpha_{k-2}, \alpha'_{k-1}) \text{ and } \alpha'_{k-1} = g(\bar{\alpha}) \quad (7)$$

which, by construction and the properties of g (Equation 4 and 6), ensures that $\alpha'_{k-1} \geq \alpha_{k-1}$, and the prime notation denotes a regularisation. In the rest of our work, we always refer to g as the *accumulated roughness function* whereas α'_{k-1} represents its output, i.e., the *accumulated roughness* for a path of length k .

Regularised Path Space Formulation We can now integrate our regularisation into to the path space formulation (Equation 1) to get the *regularised measurement*:

$$M' = \int_{\mathcal{P}} m'(\bar{\mathbf{x}}') d\mu(\bar{\mathbf{x}}') \quad (8)$$

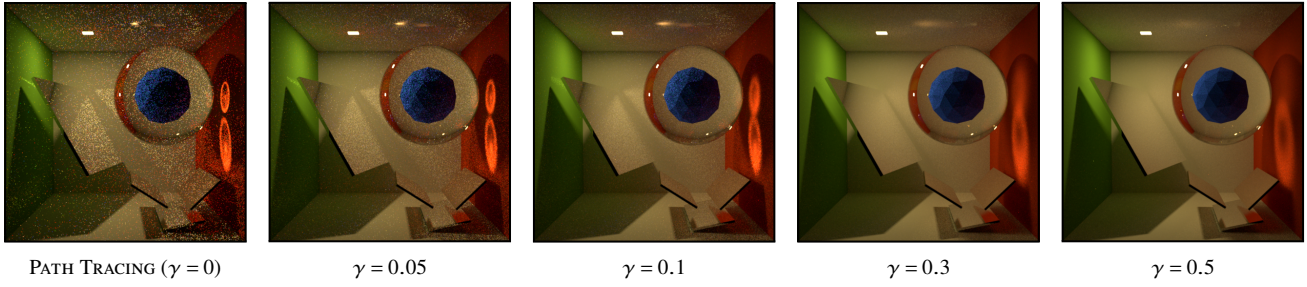


Figure 5: Application of the regularised next event estimation (Algorithm 1) to a Cornell Box featuring complex caustics. We render the scene with varying attenuation factors γ at 512 samples per pixel for different bias-variance trade-offs.

where m' denotes the *regularised measurement contribution function*, and $\bar{\mathbf{x}}'$, with a slight abuse of notation, represents the *regularised path*, that is

$$\bar{\mathbf{x}}' = (\mathbf{x}_0, \dots, \mathbf{x}'_{k-1}, \mathbf{x}_k) \quad (9)$$

where \mathbf{x}'_{k-1} is simply the original vertex \mathbf{x}_{k-1} with accumulated roughness α'_{k-1} . For the measurement contribution function, the regularisation only affects the reflectance function at the $(k-1)$ -th vertex where we use the accumulated roughness α'_{k-1} instead of the original roughness α_{k-1} . Concretely, the *regularised reflectance function* is described as

$$\rho' = \begin{cases} \rho(\mathbf{x}_{j-1}, \mathbf{x}_j, \mathbf{x}_{j+1} \mid \alpha_j), & \text{if } j \neq k-1 \\ \rho(\mathbf{x}_{j-1}, \mathbf{x}_j, \mathbf{x}_{j+1} \mid \alpha'_{k-1}), & \text{otherwise} \end{cases}$$

for all $1 \leq j < k$. Following classic Monte Carlo integration, we can define the *biased estimator* of the regularised measurement \hat{M}' as

$$\hat{M}' = \frac{1}{N} \sum_{i=0}^{N-1} \frac{m'(\bar{\mathbf{x}}'_i)}{p(\bar{\mathbf{x}}'_i)} \quad (10)$$

where the regularised path $\bar{\mathbf{x}}'_i$ has associated probability density function $p(\bar{\mathbf{x}}'_i)$.

Attenuation Factor The bias introduced by the roughening depends on the choice of accumulated roughness function. Therefore, we parameterize the function g with an additional parameter γ , the *attenuation factor*, that controls the amount of bias introduced in the regularised path. We propose to use the following function, which satisfies the imposed continuity and monotonicity property

(Equation 4) for any value of $\gamma \in [0, 1]$:

$$g(\bar{\alpha}, \gamma) = g_k(\bar{\alpha}, \gamma) = 1 - \left((1 - \alpha_{k-1}) \prod_{i=1}^{k-2} (1 - \gamma \cdot \alpha_i) \right) \quad (11)$$

where we assume that the underlying scattering model's roughness can be mapped to the interval $\alpha_i \in [0, 1]$. Intuitively, the attenuation factor controls how much roughness is accumulated along a path and confers g the following additional property:

$$\lim_{\gamma \rightarrow 0} g(\bar{\alpha}, \gamma) = g(\bar{\alpha}, 0) = \alpha_{k-1}$$

In other words, the bias is reduced to 0 as the attenuation factor goes to zero. Note that the monotonicity property, together with the guarantee that $\gamma \in [0, 1]$, implies that the function is also monotonic with respect to γ . Therefore, reducing the attenuation factor will always decrease the bias in the rendered image. The function above is not the only one to fulfil the properties introduced in Equation 4 and 6; in the early stages of our work, we explored other possibilities, such as the maximal roughness along a path or adding roughness to the last scattering event based on the number of previous bounces in the scene or the throughput of the path. However, those strategies did not perform well when used as the underlying model of the optimisation process presented in Section 4.4. Either because the model could not generalise well to arbitrary scenes (also known as overfitting) or because the large number of extra parameters needed to control the model became unpractical. Our proposed accumulated roughness function has shown to be a simple yet robust choice in that context.

Application to Path Tracing As Kaplanyan et al. remarked, regularisation is best applied at next event estimation only to avoid unnecessary bias [KD13] (Section 4.4). Only then is the type of path known and fully formed. Hence, the only change required to apply our regularisation strategy to a path tracer with NEE is to update the BSDF with the accumulated roughness before performing next event estimation and add the contribution of the regularised path (Figure 6 and Algorithm 1). Figure 5 shows the result of applying our regularisation strategy to a classic path tracer using different attenuation factors.

A common basis for roughness The concepts introduced so far are agnostic of the underlying type of roughness used by materials. However, the optimal parameters for different roughness

Algorithm 1 Regularised Next Event Estimation

Input: Attenuation factor γ and sampled path $(\mathbf{x}_0, \dots, \mathbf{x}_{k-1})$ with associated path roughness $\bar{\alpha} = (\alpha_1, \dots, \alpha_{k-1})$

Output: Regularised Monte Carlo estimate of measurement M'

- 1 $\alpha'_{k-1} \leftarrow g(\bar{\alpha}, \gamma)$ ▷ (Equation 11)
 - 2 $\mathbf{x}'_{k-1} \leftarrow$ update \mathbf{x}_{k-1} with accumulated roughness $\bar{\alpha}'_{k-1}$
 - 3 $\mathbf{x}_k \leftarrow$ sample emitter position
 - 4 $m'(\bar{\mathbf{x}}') \leftarrow W_e(\mathbf{x}_0, \mathbf{x}_1) T(\bar{\mathbf{x}}') L_e(\mathbf{x}_k, \mathbf{x}'_{k-1})$ ▷ (Equation 2)
 - 5 $p(\bar{\mathbf{x}}') \leftarrow$ probability of sampling regularised path $\bar{\mathbf{x}}'$
 - 6 **return** $m'(\bar{\mathbf{x}}')/p(\bar{\mathbf{x}}')$ ▷ (Equation 10)
-

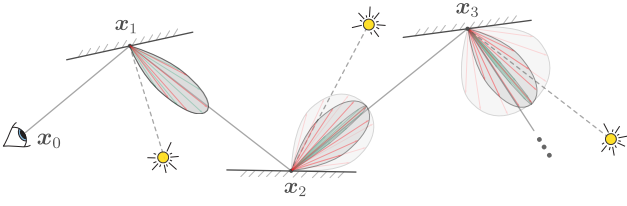


Figure 6: Path tracing with Regularised Next Event Estimation. The original BSDF lobes at every intersection are shown in green, in red and semi-transparent red the updated lobe for NEE using a low and high γ value, respectively. The regularisation increases the probability of performing a successful next event estimation for the indirect illumination.

models can slightly vary and adhering to one type of roughness for the optimisation avoids any discrepancy. In our work, we will optimise for the widely used Beckmann micro-facet distributions on which state-of-the-art multi-layered materials are based [GHZ18] and conversion for measured BSDFs is readily available [DHI*15; HKHD16]. While Beckmann roughness is unbounded in theory, we make the reasonable assumption that, for roughness above one, the material can be considered diffuse during the optimisation. Lambertian surfaces, on the other hand, are simply considered to have a roughness of one to correspond to the ideal case of diffuse (D) vertices in Heckbert notation. When materials are not isotropic, a conservative approach using the minimum roughness across the anisotropy axes works well. Note that our model is not restricted by the type of roughness used, and we could optimise it for different distributions such as the GGX distribution or the mean-cosine used in the Henyey–Greenstein phase function. Further information on how we handle multi-layered materials can be found in our supplemental material.

4.2. Consistent Estimator

While our proposed regularised estimator is biased (Equation 12), we can, if desired, turn it into a consistent one if we suppose that mollifiers (Section 3.3) can represent every type of BSDF in the scene. Under this assumption, we write the unregularised BSDF as a mollifier $\varphi_{\alpha_{k-1}}$ with mollification bandwidth α_{k-1} , for any path $\bar{x} \in \mathcal{P}_k$ of length k with associated path roughness $\bar{\alpha}$. Furthermore, we can write an offset variant of the accumulated roughness g from Equation 11, that is

$$h(\bar{\alpha}, \gamma, \alpha_{k-1}) = g(\bar{\alpha}, \gamma) - \alpha_{k-1}$$

We recall that our proposed accumulated roughness function has the property that $\lim_{\gamma \rightarrow 0} g(\bar{\alpha}, \gamma) = \alpha_{k-1}$ and therefore

$$\lim_{\gamma \rightarrow 0} h(\bar{\alpha}, \gamma, \alpha_{k-1}) = 0$$

Finally, we can simply use the continuity of g , and therefore h , together with Equation 3 to conclude that

$$\lim_{\gamma \rightarrow 0} \varphi_h(x) = \delta(x)$$

where $\varphi_h(x)$ corresponds to the regularised BSDF controlled solely by the attenuation factor γ on which h depends. To ensure a consistent Monte Carlo estimator we can therefore progressively reduce

the attenuation factor using the same sequence as proposed by Kaplanyan et al., that is

$$h_n(\bar{\alpha}, \gamma_n, \alpha_{k-1}) = h_0(\bar{\alpha}, \gamma_0, \alpha_{k-1}) n^{-\lambda} \quad \text{with } \lambda \in (0; 1/d)$$

for a d -dimensional mollifier with initial bandwidth h_0 which depends on the initial attenuation factor γ_0 . Nevertheless, reducing the attenuation factor progressively has non-trivial implications on the estimator's variance [JHD20][Figure 17] and our work focuses on the more general problem of finding good *initial* attenuation factors that minimise the joint bias and variance of the estimator.

4.3. Bias Variance Trade-off

The estimator of a rendered image composed of N pixels can be represented as a single vector $\hat{f} = [\hat{M}_0 \cdots \hat{M}_{N-1}]^T$ where \hat{M}_j is the j -th pixel's measurement estimator. Moreover, if \hat{f} is unbiased $E[\hat{f}] = f$. Similarly we can define the *regularised estimator*

$$\hat{f}' = [\hat{M}'_0 \cdots \hat{M}'_{N-1}]^T \quad (12)$$

where \hat{M}'_j is the regularised measurement estimator of pixel j (Equation 8). To measure the bias and variance of the regularised estimator we can compute the mean squared error (MSE) using the following identity:

$$\text{MSE}(\hat{f}') = E[(\hat{f}' - f)^2] = \text{Var}[\hat{f}'] + \text{Bias}^2[\hat{f}']$$

The results from Figure 5 show that attenuation factors closer to one introduce more error and progressively remove caustic paths from light transport. To get an estimate of how much bias and variance we introduce with our regularisation method, we compute the MSE with respect to a reference render for different sample count and varying attenuation factors (Figure 7). We observe that the target number of samples per pixel directly affects the estimated optimal γ value that minimises the mean squared error. However, attenuation factors close, but not equal, to zero seem to significantly decrease the overall error across different sample count, which indicates that

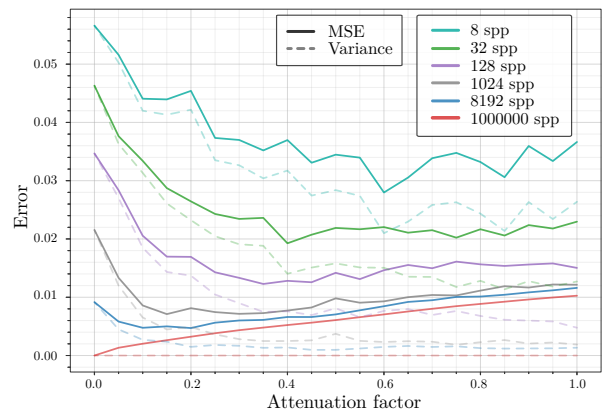


Figure 7: Empirical analysis of the variance-bias trade-off. We compute the MSE and variance for different attenuation factors on the Cornell Box scene from Figure 5. The estimator using one million samples per pixels (spp) has no variance and its MSE therefore solely represents the bias introduced by the regularisation.

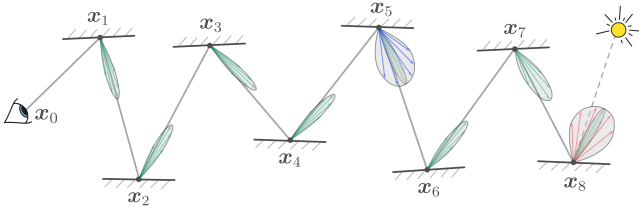


Figure 8: Recursive accumulated roughness. An illustration of our proposed recursive accumulated roughness applied to a path of length nine. The original BSDF lobes at every intersection are shown in green, in blue, the virtual lobe that corresponds to the accumulated roughness of the first five-bounce subpath $(x_1, x_2, x_3, x_4, x_5)$ and in red, the final lobe used to perform next event estimation with recursively evaluated accumulated roughness of the remaining sub-path (x'_5, x_6, x_7, x_8) .

lower attenuation factors are probably the best way to achieve both low variance and low bias at the same time. Note that while those observations hold true in this specific scene, it is hard to predict the attenuation factor's behaviour on arbitrary scenes, and a simple grid search as performed for this scene would not be practical. Furthermore, we have considered only a single attenuation factor for the whole light transport; as we will see in the next section, increasing our model's complexity can further reduce the overall mean squared error achievable.

4.4. Optimised Path Space Regularisation

This section shows how we can further reduce the overall error in rendered images by increasing our roughening model's complexity. We also describe how we use differentiable rendering to tackle the optimisation problem at hand and, finally, introduce the associated objective function that we use during gradient descent to control our regularisation's inherent bias-variance trade-off.

Path Space Dependent Attenuation The roughening scheme proposed so far allows us to trade noise for bias only to a small extent. When we fix the attenuation factor, the whole light transport is affected in the same way, but variance distribution in the render is generally not uniform: some parts of the scene may need more or less roughening to achieve similar variance. We seek to increase our current model's complexity to account for such disparities and treat them appropriately while staying lightweight and generalise as much as possible. Therefore, it is crucial to keep the number of optimisable parameters low and, ideally, scene independent to allow the optimiser to find a suitable set of parameters.

We extend the attenuation factor to depend on the type of sampled path and rewrite it as a function of the path roughness $\gamma(\vec{\alpha})$. This allows us to roughen every type of path independently, and we reformulate Equation 11 as:

$$g(\vec{\alpha}) = g_k(\vec{\alpha}) = 1 - \left((1 - \alpha_k) \prod_{i=1}^{k-1} (1 - \gamma(\vec{\alpha}) \cdot \alpha_i) \right) \quad (13)$$

where, for conciseness, we dropped g 's dependence on γ .

Discrete path space classification To later optimise the attenuation factors that control the accumulated roughness along a path in practice, we need some form of discretisation of the path roughness. We use a logarithmic scale on the roughness to better approximate near-specular lobes: the origin of most of the variance introduced and for which we need fine-tuning. On the other hand, higher roughness values are already well handled using next event estimation, and we discretise them more coarsely. Concretely, the conversion from roughness α to the bin's index i with quantization level Q is defined as follows:

$$i = \min \left(\left\lfloor (2^{\sqrt{\alpha}} - 1) \cdot (Q + 1) \right\rfloor, Q - 1 \right)$$

Using the described discretisation to learn an attenuation factor for every possible path roughness, we would have to optimise a total of $\sum_{i=2}^{k-1} Q^i$ parameters. Clearly, this poses a problem even for reasonable path lengths and quantization level Q . Therefore, in our implementation, we use $Q = 4$ and learn attenuation factors only for paths consisting of five scattering events at most ($k = 6$). Most complex lightning effects, such as directly and indirectly visible caustics, are included in that range.

Recursive Accumulated Roughness To handle higher-order paths, we use a simple recursive heuristic that can be formulated as a new accumulated roughness function $g_k^r(\vec{\alpha})$ which, by construction, still satisfies the monotonicity property:

$$g_k^r(\vec{\alpha}) = \begin{cases} g_k(\vec{\alpha}), & \text{if } k \leq 6 \\ g_{k-4}^r((\alpha'_5, \alpha_6, \dots, \alpha_{k-1})), & \text{otherwise} \end{cases} \quad (14)$$

where $\alpha'_5 = g_6((\alpha_1, \dots, \alpha_5))$ from Equation 7. Intuitively, the recursion aggregates five-bounce subpaths until the remaining path corresponds to a base case ($k \leq 6$). Figure 8 illustrates how the method applies to a path of length nine.

Variance-aware Objective Function Assuming a total of $P = \sum_{i=2}^{k-1} Q^i$ optimisable attenuation factors, we stack them in a single vector $\gamma = [\gamma_0 \dots \gamma_{P-1}]^T$. We also write the regularised estimator's dependence on the attenuation factors explicitly as $f'(\gamma)$ (Equation 12) and compactly formulate the optimisation problem we are trying to solve as follows:

$$\gamma_{\text{opt}} = \arg \min_{\gamma} \mathcal{L}(\hat{f}'(\gamma), f) \quad (15)$$

For the optimisation, we use the state-of-the-art *Adam* optimiser [KB14] with the default parameters and a learning rate $\eta = 5 \cdot 10^{-4}$. Our loss function uses a mean absolute percentage error (MAPE) which reduces the impact of outliers (fireflies) and allows the optimiser to automatically adapt to different brightness levels in the scene more robustly. To let the optimiser focus on the variance reduction we also add an extra term that solely accounts for the estimator's variance. Our loss function can thus be formulated as follows:

$$\mathcal{L}(\hat{f}'(\gamma), f) = \text{MAPE}(\hat{f}'(\gamma), f) + \beta \cdot \text{Var}(\hat{f}'(\gamma)) \quad (16)$$

where the hyperparameter β gives control over the acceptable level

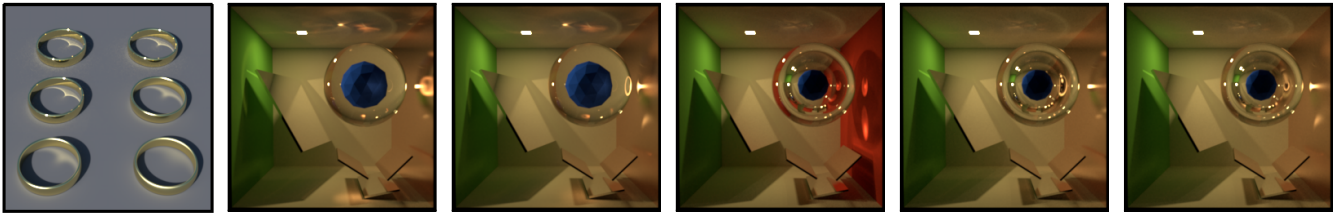


Figure 9: Scenes used during the optimisation process. In addition to the scene from Figure 5, we use six additional scenes featuring complex caustics and varying glossiness to train our model.

of noise and the variance term is estimated using the regularised estimator’s sample variance.

Training Set In addition to the scene in Figure 5, our training set comprises six scenes that aim to represent hard-to-sample paths, such as directly and indirectly visible caustics for different roughnesses and path lengths (Figure 9). The types of paths represented in the training set, for which we want to learn accurate attenuation factors, are typically found even in more complex scenes. As mentioned earlier, we only learn γ for paths of length $k \leq 6$. If we reduce the maximal path length to $k < 6$ we start missing interesting caustic paths, while increasing it beyond $k = 6$ does not show any notable improvements and drastically increases memory requirements. As our model is not geometry-aware, the scenes are not required to be significantly different to learn good parameters during optimisation. However, it is important that the desired hard-to-sample paths are present in the training set. The ring scene, for example, was used to ensure that both visible and indirectly visible caustics are present across different range of glossiness.

Optimisation While the number of scenes we are training on can seem low, our model only cares about the type of sampled path for a given pixel contribution. Every attenuation factor is responsible for some partitioning of the path space, and the learning process effectively *overfits* each optimised γ parameter to a specific type of sampled path roughness. Consequently, during the optimisation process, all the entries in γ are independent and can be updated simultaneously at every gradient step. In the context of deep learning, a similar, but not equivalent, type of model referred to as a *mixture of experts* exists [ME14]. The main idea consists of splitting a problem into smaller homogeneous problems for which the *experts* are responsible for finding optimal solutions. In our model, each overfitted attenuation factor can be seen as an expert for a specific type of path roughness and allows our model to generalise to a large variety of scenes; we train the model on the type of sampled path rather than the lighting or geometric information which can significantly change from one scene to another. To perform a gradient descent optimisation and minimise the loss function described in Equation 16, we need to fix both the number of samples per pixel and the beta parameter used; together, they fix the noise level for which we want to optimise γ (Equation 15). The learning process needs roughly ten thousand steps to converge for any value of β , and takes about twelve hours on a Titan RTX GPU.

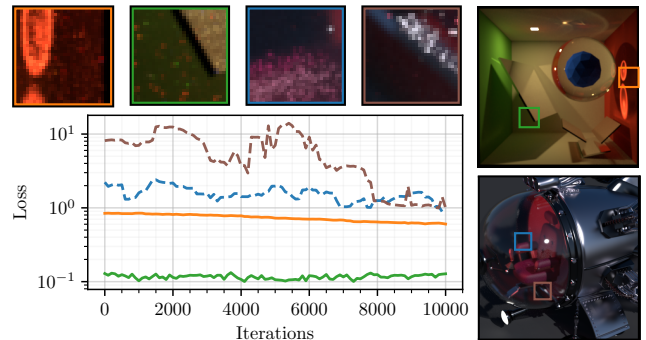


Figure 10: Training and validation using the mean loss over 32 by 32 pixels wide crops. The bottom left plot depicts the error for every iteration of the gradient descent on two crops from the training data set (straight lines) and two crops from the validation data set (dashed lines). In this example, we use 1024 samples per pixels for the optimisation and we fix $\beta = 0.001$ for the loss. The insets correspond to the rendered crops at the last iteration (10’000).

Validation Figure 10 shows how the optimiser, starting with a uniform initial guess of $\gamma = 0.5$, progressively reduces the error until convergence during training and validation. This also shows empirically that our model robustly handles scenes with geometry and materials that greatly differ from our training set. Note that the scenes evaluated in Section 5 use the same set of learnt parameters obtained from the described training set and, therefore, further contribute to validate our model.

5. Implementation and Results

In this section, we present several results of applying our regularisation strategy to path tracing estimators and important considerations regarding the generalisation of our model. We compare all the results for two sets of optimised parameters to demonstrate our method’s inherent tradeoff between variance and bias; we choose $\beta = 0.001$ and $\beta = 0.05$ (Equation 16) to represent a relatively low and moderate tolerance to bias, respectively. To analyse the perceptual error introduced by our regularisation, we use the \mathcal{FLIP} error, which approximates the difference perceived by humans when alternating between a rendered image and a corresponding ground truth [ANA*20]. In addition to the result presented in this section, the supplemental video contains real-time visualisations of our optimi-

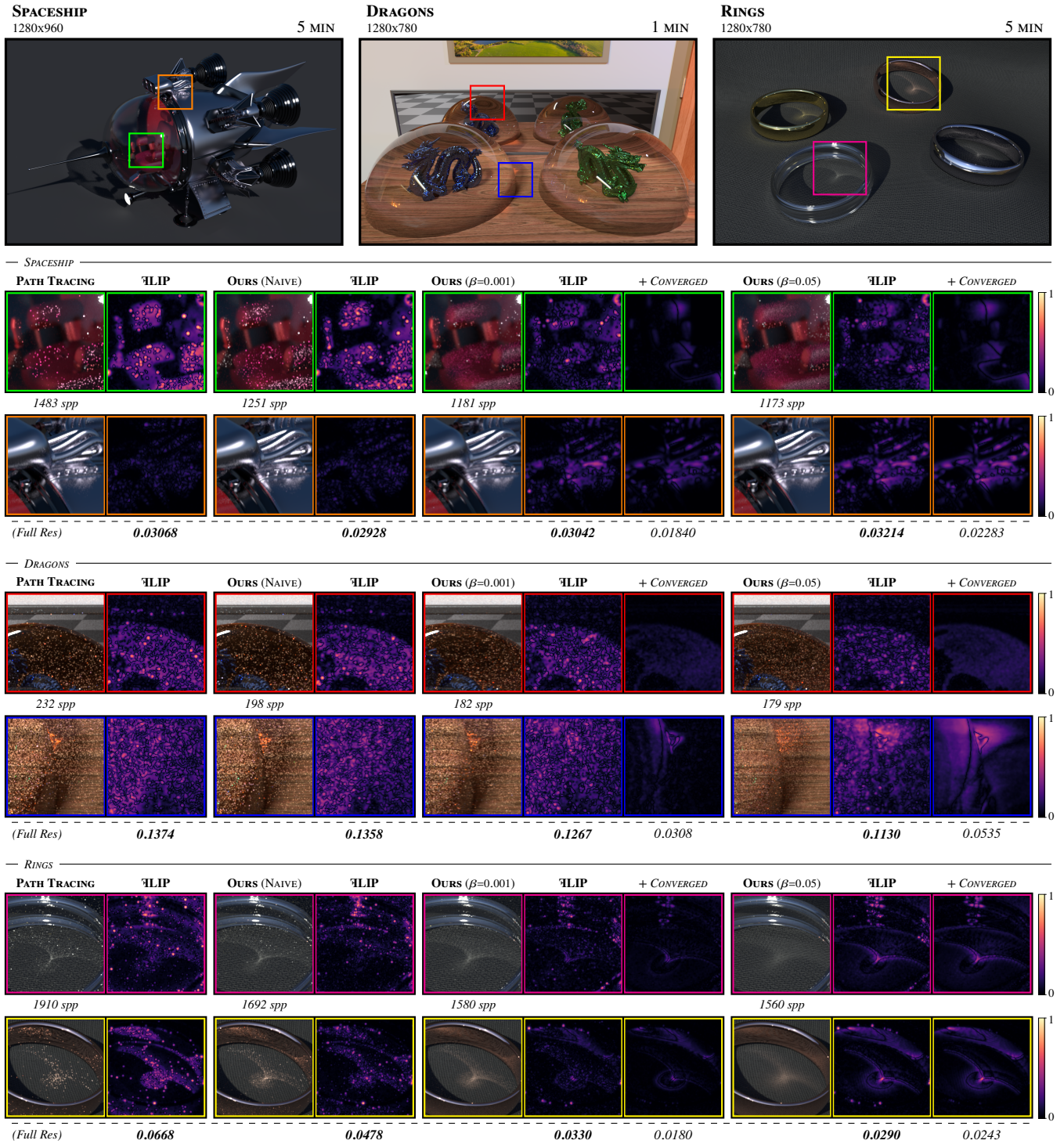


Figure 11: Equal time comparison between regular path tracing and path tracing with regularised NEE for two sets of learnt parameters. We compare three path-traced scenes containing hard-to-sample paths against our extended path tracing algorithm with regularised next event estimation and a naive version that uses a constant non-optimised attenuation factor. For each technique, we indicate the mean FLIP error on the full resolution image. For our method, we also compute the FLIP error on the converged image to visualise the bias introduced. Our method achieves consistently lower variance for a small systematic error compared to regular path tracing. The slightly more aggressive roughening ($\beta = 0.05$) further reduces the variance but introduces more significant visible bias.

sation process and animated versions of the beach scene depicted in Figure 1 that demonstrate the temporal stability of our method.

5.1. Optimised Path Space Regularisation for Path Tracing

Adding our regularisation to a path tracer is straightforward; to implement the regularised next event estimation, we only need to store the materials' original roughness along a path into an adequate data structure and query the accumulated roughness of the subpath to temporarily change the roughness of the vertex where we perform next event estimation (Line 1 and 2 of Algorithm 1).

Memory The memory requirements of our method are quite low thanks to the discrete path space classification and our recursive formulation of the accumulated roughness (Equation 14); with a discretisation level of four and aggregating five-bounce subpaths ($k = 6$), every hyperparameter β we optimise for requires us to store a total of $P = \sum_{i=2}^{k-1} Q^i = 1360$ attenuation factors (5.216 kB).

Multiple Importance Sampling When using our regularised next event estimation with multiple importance sampling (MIS) [VG95b], we must ensure that other techniques are also evaluating the regularised path. While this can seem a minor issue at first, after using the regularised BSDF for the contribution of a complete path of length k , we can no longer use the same regularised BSDF to sample a path of length $k + 1$ without introducing additional error in the estimator, a standard optimisation among path tracers in absence of regularisation. A simple way to avoid any unnecessary bias is to construct the *path prefix* of length $k - 1$ using the unbiased BSDF while only taking into account contributions from the complete regularised path of length k , that is, the MIS weighted contributions of the regularised BSDF and regularised NEE. We use this solution in all the results that use our regularisation. However, if speed is critical, regularising every vertex along a path and reusing the regularised BSDF to sample the path prefix of length $k - 1$ achieves close to equal performance to classical path tracing while only adding a small additional error compared to our proposed solution. For instance, an ESDSL path (in standard Heckbert notation) is regularised identically both when reusing and not reusing the regularised BSDF since we never alter the first bounce, the second bounce already has maximal roughness, and we effectively only regularise the last specular bounce before connecting to a light source.

Results Analysis We compare our regularised path tracer against classical path tracing with next event estimation and a naive solution corresponding to an unoptimised version of our method. The naive solution uses a constant $\gamma \approx 0.01$ for each type of sampled path and corresponds to the attenuation factor that our optimised regularisation with $\beta = 0.001$ uses for a path roughness $\bar{\alpha} = (0.0, 1.0)$, i.e. a caustic arising from a perfectly specular material. The naive strategy is slightly faster as it does not depend on the type of sampled path and can evaluate Equation 11 on the fly. We perform equal time comparisons between three different scenes (Figure 11). The SPACESHIP scene is illuminated by an environment map containing a strong directional sun and a small area light source inside the cockpit. As shown in the green

insets, this results in challenging SDS paths and long specular chains inside the spaceship that are extremely hard to sample for path tracing. Our method samples these challenging paths robustly and starts resolving paths that path tracing has not encountered yet. The converged crop's FLIP error also shows how our method keeps the perceptible error extremely low even in regions where regularisation was applied. In the orange insets, we can observe that the bias introduced can sometimes be hard to notice by looking at two images side by side, and the FLIP error allows us to identify the systematic error better. The DRAGONS scene is a much simpler scenario for simple path tracing; however, our technique, while less effective in general, can still regularise unexpected long specular chains arising on the glossy plastic border of the mirror or help reduce variance on the two reflected glass hemispheres. The RINGS scene shows the benefit of using our regularisation when sampling both visible and indirectly visible caustics. A significant amount of variance is not related to the directly visible caustic but corresponds to longer glossy chains, which our optimised model will automatically roughen more.

5.2. Comparison with Specular Manifold Sampling

Specular Manifold Sampling (SMS), by Zeltner et al. [ZGJ20], addresses the problem of caustic paths due to perfectly specular interactions. While their technique is unbiased, the Newton-based solver used to find an explicit connection to a light source through a specular interface can be subject to numerical instability and rely on suitable surface derivatives to converge. They provide an interesting two-stage manifold walk improvement for normal-mapped surfaces that aids in that regard. However, real geometry can still be challenging and requires careful selection of the solver's parameters to achieve good performance and not run into ill conditions. We compare our regularisation to their unbiased method in the POOL scene shown in Figure 12. We found that the following choice of parameters for their solver achieved the best result for this particular scene: we set the solver's threshold and uniqueness threshold for caustics to $1e-6$, the maximal number of trials to one million and the maximal number of iterations of the solver to 20. We observe that SMS is extremely powerful at finding sharp caustics but can suffer the overhead of costly manifold walks in occluded regions or when the solver does not converge to a solution and falls back to BSDF sampling. Note that our technique is *biased* and trades some error to increase the rendering algorithm's robustness, whereas SMS aims to find the *exact* solution to the caustic problem. On the other hand, SMS's biased variant requires a thoughtful choice of the maximal number of trials not to suffer from excessive energy loss or performance drop. Further comparisons for different water depths and their unbiased approach are available in our supplemental material. While SMS can, in theory, handle rough surfaces, it is generally not advised since classical path tracing, and therefore our regularised path tracer will rapidly outweigh the costly offset manifold walks as the surface's roughness increases.

5.3. Application to Path Guiding

Our method is remarkably unintrusive, and we can combine it with most path guiding techniques without any additional consideration than the one discussed for path tracing. All the results that use path

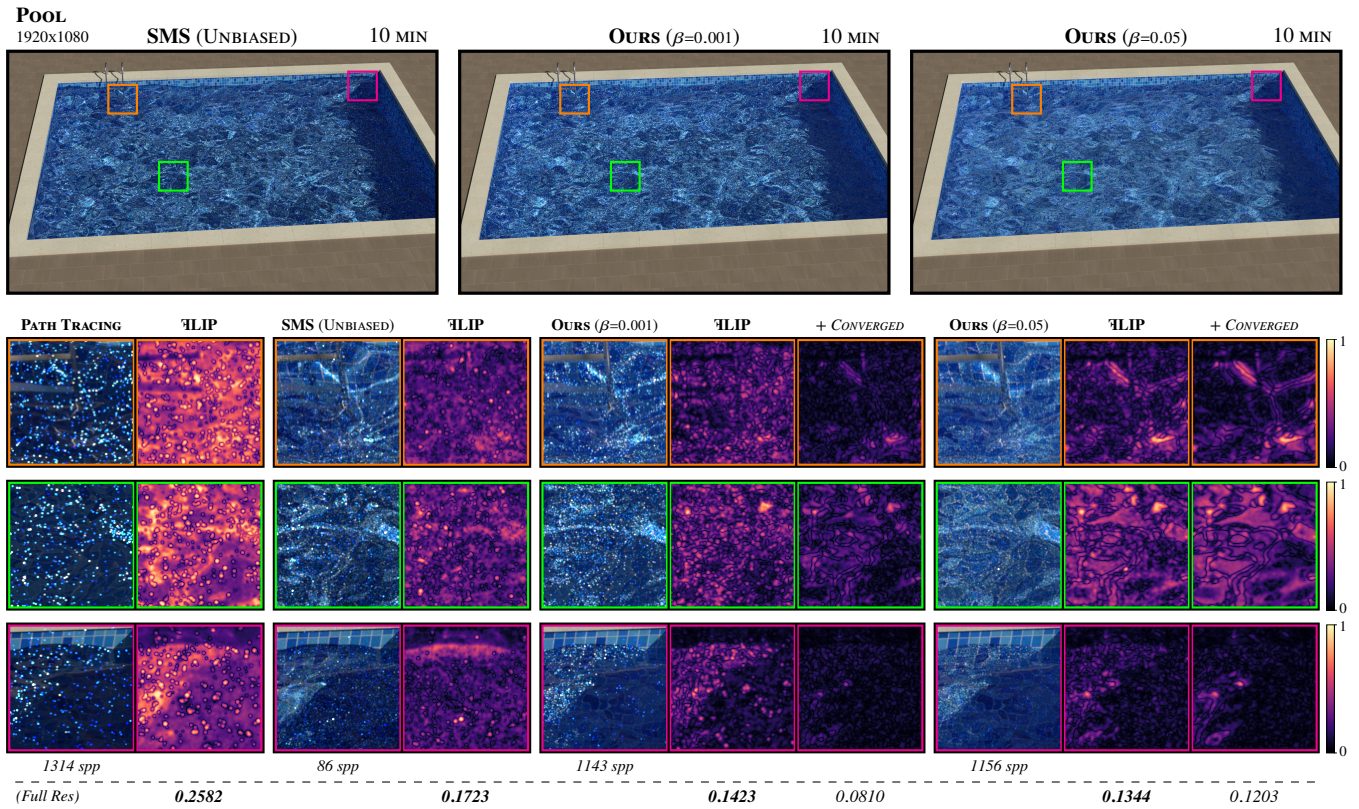


Figure 12: Equal time comparison between regular path tracing, Specular Manifold Sampling (SMS) and path tracing with regularised NEE for two sets of learnt parameters. While Specular Manifold sampling is unbiased, our optimised regularisation with $\beta = 0.001$ closely competes in finding caustic paths and achieves overall lower mean FLIP error for equal render-time. Moreover, our method with $\beta = 0.05$ achieves drastic variance reduction at the cost of a more significant systematic error.

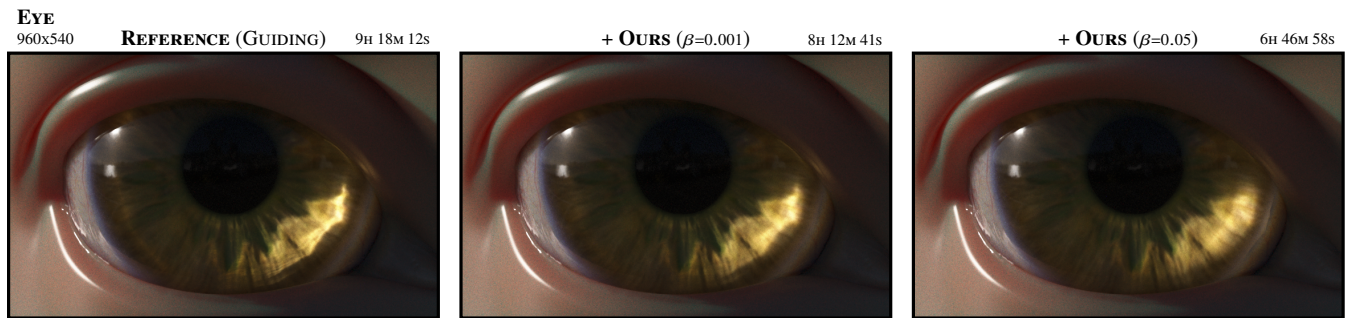


Figure 13: Regularisation in complex scenes. The scene features a complex double refractive caustic through the cornea of an eye, multi-layered materials and volumetric light transport for the skin. Applying our regularisation can still reduce render-time in that context and maintains low visible error.

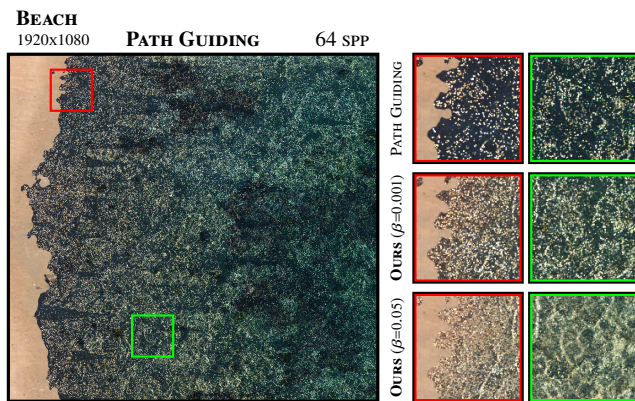


Figure 14: Fixed sample count comparison between path guiding and our regularised path guiding. Adding our regularised next event estimation to path guiding algorithms enables finding otherwise hard-to-sample paths and can significantly speed up the learning of guiding distributions.

guiding are based on the method from Müller et al. [MGN17], which has been widely adopted in the movie industry. We believe that our regularisation can be used with any recent guiding strategy as we only regularise next event estimation, which is usually orthogonal to the problem guiding addresses. Figure 14 demonstrates the benefit of using our regularisation in the context of path guiding. At 64 samples per pixel pure path guiding has not yet been able to learn accurate guiding distributions in the entire scene, especially in the presence of intricate geometry resulting in noticeable artifacts. Our regularisation helps finding contributions more consistently and therefore also improves the learning process. The reduced variance of the estimator can lead to significant performance gain as can be seen in Figure 1.

5.4. Perceived Error Discussion

The error added by our regularisation has the advantage of being imperceptible under smooth illumination. Under high-frequency illumination or at shadow boundaries, the blur introduced by our method is conceptually similar to adaptively increasing the size of light sources depending on the material’s roughness. Therefore, it results in plausible and hardly identifiable bias without directly comparing to the reference, as seen in Figure 13. In concert with the ability to sample complex paths and to only introduce bias when the estimator’s variance would be significantly higher, we believe our regularisation is a robust addition to any path tracer.

5.5. Limitations and Future Work

Our regularisation method’s limitations are directly related to the complexity of our model and its ability to generalise. While the accumulated roughness is independent of the path roughness, we have no information related to the scene’s geometry. We can observe this limitation in Figure 1, where the more aggressive roughening with $\beta = 0.05$ has a low bias in the shallow water, but the blur becomes noticeable as the water becomes deep. Our model has no way of

accommodating the two different cases. We mitigate those effects by ensuring sharp caustics in our training set, guaranteeing that our objective function might penalise even a slight regularisation in the corresponding path. Another interesting aspect for future work would be integrating path guiding frameworks into the optimisation process and experiment with adaptive reduction of the mollifier bandwidth in conjunction with our consistent formulation of the accumulated roughness to further reduce bias without increasing the variance of the estimator dangerously.

6. Conclusion

In this paper, we introduce the concept of *accumulated roughness* along a path and how we can use it as a regularisation strategy for next event estimation. We further extend the accumulated roughness with a path space dependent *attenuation factor* which enables us to control the inherent trade-off between bias and variance. On this basis, we devise a model whose objective function directly controls the desired level of tolerated noise using a single hyperparameter and use differentiable rendering to train our model on scenes featuring complex light transport and surfaces with varying glossiness. Our method is temporally stable, extremely simple to add to any path tracer, and shows great robustness in complex scenes where it consistently helps reducing the variance of estimators while maintaining a low perceptual error. Finally, we show that adding our regularisation to a regular path tracer enables us to compete with state-of-the-art techniques such as Specular Manifold Sampling while achieving similar or better convergence in the presence of arbitrary surface roughness or complex geometry.

Acknowledgements We want to thank Javor Kalojanov for early discussions related to the optimisation process, Jean-Marie Aubry for his mathematical contributions and Shilin Zhu for his extensive feedback on the writing. Thanks to Benedikt Bitterli for providing the models present in the SPACESHIP and DRAGONS scenes, the Stanford University Computer Graphics Laboratory for providing the dragon model and Weta Digital for kindly allowing us to use the BEACH and EYE scene.

References

- [ANA*20] ANDERSSON, PONTUS, NILSSON, JIM, AKENINE-MÖLLER, TOMAS, et al. “FLIP: A Difference Evaluator for Alternating Images”. *Proceedings of the ACM on Computer Graphics and Interactive Techniques* 3.2 (2020), 15:1–15:23 8.
- [BIOP13] BOUCHARD, GUILLAUME, IEHL, JEAN-CLAUDE, OSTROMOUKHOV, VICTOR, and POULIN, PIERRE. “Improving Robustness of Monte-Carlo Global Illumination with Directional Regularization”. *SIGGRAPH Asia 2013 Technical Briefs*. SA ’13. Association for Computing Machinery, 2013. ISBN: 9781450326292. DOI: [10.1145/2542355.2542383](https://doi.org/10.1145/2542355.2542383) 2.
- [DH09] DIMIAN, DANNY and HERBST, KARL. “Practical Uses of a Ray Tracer for Cloudy with a Chance of Meatballs”. *SIGGRAPH 2009: Talks*. SIGGRAPH ’09. New Orleans, Louisiana: Association for Computing Machinery, 2009. ISBN: 9781605588346. DOI: [10.1145/1597990.1598034](https://doi.org/10.1145/1597990.1598034) 2.
- [DHI*15] DUPUY, JONATHAN, HEITZ, ERIC, IEHL, JEAN-CLAUDE, et al. “Extracting microfacet-based brdf parameters from arbitrary materials with power iterations”. *Computer Graphics Forum*. Vol. 34. 4. Wiley Online Library, 2015, 21–30 6.

- [FHP*18] FASCIONE, LUCA, HANIKA, JOHANNES, PIEKÉ, ROB, et al. "Path tracing in production". en. *ACM SIGGRAPH 2018 Courses*. Vancouver British Columbia Canada: ACM, Aug. 2018, 1–79. ISBN: 9781450358095. doi: [10.1145/3214834.3214864](https://doi.org/10.1145/3214834.3214864). (Visited on 06/04/2020) 2.
- [GHZ18] GUO, YU, HASAN, MILOS, and ZHAO, SHUANG. "Position-free monte carlo simulation for arbitrary layered BSDFs". *ACM Trans. Graph.* 37 (2018), 279:1–279:14 6.
- [GKDS12] GEORGIEV, ILIYAN, KŘIVÁNEK, JAROSLAV, DAVIDOVIČ, TOMÁŠ, and SLUSALLEK, PHILIPP. "Light Transport Simulation with Vertex Connection and Merging". *ACM Trans. Graph.* 31.6 (Nov. 2012). ISSN: 0730-0301. doi: [10.1145/2366145.2366211](https://doi.org/10.1145/2366145.2366211) 2.
- [GPSK18] GRITTMANN, PASCAL, PÉRARD-GAYOT, ARSÈNE, SLUSALLEK, PHILIPP, and KŘIVÁNEK, JAROSLAV. "Efficient Caustic Rendering with Lightweight Photon Mapping". *Computer Graphics Forum* 37.4 (2018). EGSR '18. ISSN: 1467-8659 2.
- [HA98] HEGLAND, M. and ANDERSEN, R. S. "A mollification framework for improperly posed problems". *Numerische Mathematik* 78.4 (1998), 549–575. doi: [10.1007/s002110050325_3](https://doi.org/10.1007/s002110050325_3).
- [HDF15] HANIKA, JOHANNES, DROSKE, MARC, and FASCIONE, LUCA. "Manifold Next Event Estimation". *Comput. Graph. Forum* 34.4 (July 2015), 87–97. ISSN: 0167-7055 2.
- [Hec90] HECKBERT, PAUL S. "Adaptive Radiosity Textures for Bidirectional Ray Tracing". *Proceedings of the 17th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '90. Dallas, TX, USA: Association for Computing Machinery, 1990, 145–154. ISBN: 0897913442. doi: [10.1145/97879.978953](https://doi.org/10.1145/97879.978953).
- [HEV*16] HERHOLZ, SEBASTIAN, ELEK, OSKAR, VORBA, JIŘI, et al. "Product Importance Sampling for Light Transport Path Guiding". *Comput. Graph. Forum* 35.4 (July 2016), 67–77. ISSN: 0167-7055 2.
- [HI09] HACHISUKA, TOSHIYA and JENSEN, HENRIK WANN. "Stochastic Progressive Photon Mapping". *ACM SIGGRAPH Asia 2009 Papers*. SIGGRAPH Asia '09. Yokohama, Japan: Association for Computing Machinery, 2009. ISBN: 9781605588582. doi: [10.1145/1661412.1618487](https://doi.org/10.1145/1661412.1618487) 2.
- [HKHD16] HOLZSCHUCH, NICOLAS, KAPLANYAN, ANTON, HANIKA, JOHANNES, and DACHSBACHER, CARSTEN. "Estimating Local Beckmann Roughness for Complex BSDFs". *ACM SIGGRAPH 2016 Talks*. SIGGRAPH '16. Anaheim, California: Association for Computing Machinery, 2016. ISBN: 9781450342827. doi: [10.1145/2897839.2927416](https://doi.org/10.1145/2897839.2927416) 6.
- [HOJ08] HACHISUKA, TOSHIYA, OGAKI, SHINJI, and JENSEN, HENRIK WANN. "Progressive Photon Mapping". *ACM SIGGRAPH Asia 2008 Papers*. SIGGRAPH Asia '08. Singapore: Association for Computing Machinery, 2008. ISBN: 9781450318310. doi: [10.1145/1457515.1409083](https://doi.org/10.1145/1457515.1409083) 2.
- [Jen96] JENSEN, HENRIK WANN. "Global Illumination Using Photon Maps". *Proceedings of the Eurographics Workshop on Rendering Techniques '96*. Porto, Portugal: Springer-Verlag, 1996, 21–30. ISBN: 3211828834 2.
- [JG19] JENDERSIE, JOHANNES and GROSCH, THORSTEN. "Microfacet Model Regularization for Robust Light Transport". *Computer Graphics Forum (Proc. of EGSR)* 38.4 (July 2019), 39–47. doi: [10.1111/cgf.13768](https://doi.org/10.1111/cgf.13768) 2.
- [JHD20] JUNG, ALISA, HANIKA, JOHANNES, and DACHSBACHER, CARSTEN. "Spectral Mollification for Bidirectional Fluorescence". *Computer Graphics Forum (Proceedings of Eurographics)* 39.2 (2020) 6.
- [Kaj86] KAJIYA, JAMES T. "The Rendering Equation". *Proceedings of the 13th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '86. New York, NY, USA: Association for Computing Machinery, 1986, 143–150. ISBN: 0897911962. doi: [10.1145/15922.15902](https://doi.org/10.1145/15922.15902) 2.
- [KB14] KINGMA, DIEDERIK P and BA, JIMMY. "Adam: A method for stochastic optimization". *arXiv preprint arXiv:1412.6980* (2014) 7.
- [KD13] KAPLANYAN, ANTON and DACHSBACHER, CARSTEN. "Path Space Regularization for Holistic and Robust Light Transport". *Computer Graphics Forum* 32 (May 2013). doi: [10.1111/cgf.12026](https://doi.org/10.1111/cgf.12026) 2–5.
- [KZ11] KNAUS, CLAUDE and ZWICKER, MATTHIAS. "Progressive Photon Mapping: A Probabilistic Approach". *ACM Trans. Graph.* 30.3 (May 2011). ISSN: 0730-0301. doi: [10.1145/1966394.1966404](https://doi.org/10.1145/1966394.1966404) 2.
- [LADL18] LI, TZU-MAO, AITTALA, MIKA, DURAND, FRÉDO, and LEHTINEN, JAAKKO. "Differentiable Monte Carlo Ray Tracing through Edge Sampling". *ACM Trans. Graph. (Proc. SIGGRAPH Asia)* 37.6 (2018), 222:1–222:11 2, 4.
- [LW98] LAFORTUNE, ERIC and WILLEMS, YVES. "Bi-Directional Path Tracing". *Proceedings of Third International Conference on Computational Graphics and Visualization Techniques (Compugraphics' 93)* (Jan. 1998) 2.
- [ME14] MASOUDNIA, SAEED and EBRAHIMPUR, REZA. "Mixture of experts: A literature survey". *Artificial Intelligence Review* 42 (Aug. 2014). doi: [10.1007/s10462-012-9338-y](https://doi.org/10.1007/s10462-012-9338-y) 8.
- [MGN17] MÜLLER, THOMAS, GROSS, MARKUS, and NOVÁK, JAN. "Practical Path Guiding for Efficient Light-Transport Simulation". *Computer Graphics Forum* 36.4 (June 2017), 91–100. ISSN: 1467-8659. doi: [10.1111/cgf.13227](https://doi.org/10.1111/cgf.13227) 2, 12.
- [MMR*18] MÜLLER, THOMAS, MCWILLIAMS, BRIAN, ROUSSELLE, FABRICE, et al. "Neural Importance Sampling". *CoRR* abs/1808.03856 (2018). arXiv: [1808.03856](https://arxiv.org/abs/1808.03856). URL: <http://arxiv.org/abs/1808.03856> 2.
- [NVZJ19] NIMIER-DAVID, MERLIN, VICINI, DELIO, ZELTNER, TIZIAN, and JAKOB, WENZEL. "Mitsuba 2: A Retargetable Forward and Inverse Renderer". *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 38.6 (Dec. 2019). doi: [10.1145/3355089.3356498](https://doi.org/10.1145/3355089.3356498) 2, 4.
- [ŠK16] ŠIK, MARTIN and KŘIVÁNEK, JAROSLAV. "Improving Global Exploration of MCMC Light Transport Simulation". *ACM SIGGRAPH 2016 Posters*. SIGGRAPH '16. Association for Computing Machinery, 2016. ISBN: 9781450343718. doi: [10.1145/2945078.2945128](https://doi.org/10.1145/2945078.2945128) 2.
- [Vea97] VEACH, ERIC. "Robust Monte Carlo Methods for Light Transport Simulation". PhD thesis. Stanford University, 1997 3.
- [VG95a] VEACH, ERIC and GUIBAS, LEONIDAS. "Bidirectional Estimators for Light Transport". *Photorealistic Rendering Techniques*. Ed. by SAKAS, GEORGIOS, MÜLLER, STEFAN, and SHIRLEY, PETER. Berlin, Heidelberg: Springer Berlin Heidelberg, 1995, 145–167. ISBN: 978-3-642-87825-1 2.
- [VG95b] VEACH, ERIC and GUIBAS, LEONIDAS J. "Optimally Combining Sampling Techniques for Monte Carlo Rendering". *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '95. New York, NY, USA: Association for Computing Machinery, 1995, 419–428. ISBN: 0897917014. doi: [10.1145/218380.218498](https://doi.org/10.1145/218380.218498) 2, 10.
- [VHH*19] VORBA, JIRI, HANIKA, JOHANNES, HERHOLZ, SEBASTIAN, et al. "Path Guiding in Production". *ACM SIGGRAPH 2019 Courses*. SIGGRAPH '19. Los Angeles, California: Association for Computing Machinery, 2019. ISBN: 9781450363075. doi: [10.1145/3305366.3328091](https://doi.org/10.1145/3305366.3328091) 2.
- [VKS*14] VORBA, JIRI, KARLIK, ONDREJ, ŠIK, MARTIN, et al. "On-Line Learning of Parametric Mixture Models for Light Transport Simulation". *ACM Trans. Graph.* 33.4 (July 2014). ISSN: 0730-0301. doi: [10.1145/2601097.2601203](https://doi.org/10.1145/2601097.2601203) 2.
- [ZGJ20] ZELTNER, TIZIAN, GEORGIEV, ILIYAN, and JAKOB, WENZEL. "Specular Manifold Sampling for Rendering High-Frequency Caustics and Glints". *Transactions on Graphics (Proceedings of SIGGRAPH)* 39.4 (July 2020). doi: [10.1145/3386569.3392408](https://doi.org/10.1145/3386569.3392408) 2, 10.