To appear in ACM TOG 0(0).

HairIS: Importance Sampling for Hair Scattering

Jiawei Ou[1,2] Feng Xie[1] Parashar Krishnamachari[1] Fabio Pellacini[2]
[1]PDI/DreamWorks Animation [2]Dartmouth College

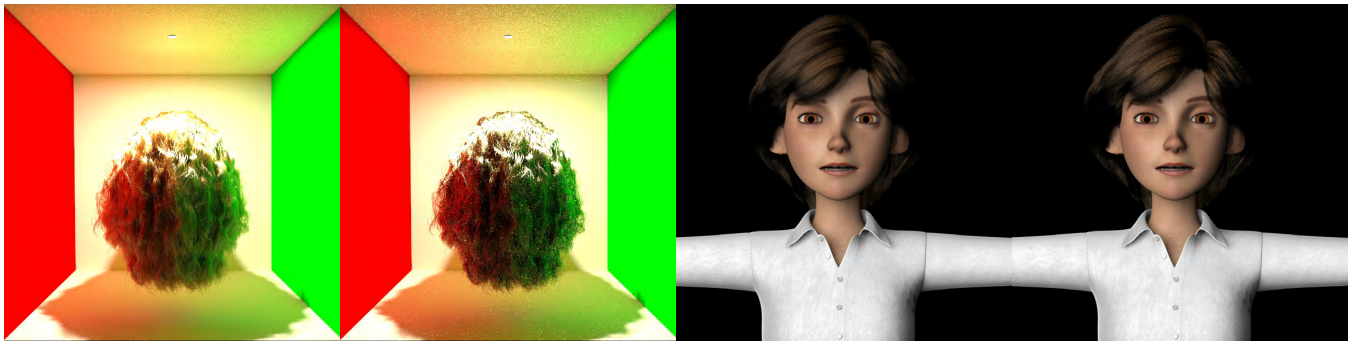

Figure 1: *Comparison between stratified uniform sampling and our importance sampling for hair* bsdf.

Abstract

In this paper we present a practical importance sampling method for the bidirectional scattering distribution function (*bsdf*) of hair material. Our sampling method is derived from a simplified approximation of the Marschener hair model, also known as the artist friendly hair shading model. Specifically, we show that by approximating the *gaussian* components in the hair *bsdf* lobes using *cauchy* distributions, we are able to derive an analytic *pdf*s and an efficient sampling function for each hair *bsdf* lobe. Our method is robust, efficient and accurate. It can be seamlessly integrated with multiple importance sampling to improve the performance of environment lighting, area lighting and indirect lighting for hair. Unlike importance sampling methods based on factorized *bsdf* representations, e.g. wavelet importance sampling and spherical harmonic importance sampling, our method provides a closed-form solution that does not require any precomputation and has very low memory footprint. Compare to uniform sampling, our method significantly reduce the number of samples for rendering hair material.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Color, shading, shadowing, and texture;

Keywords: hair rendering, importance sampling

1 Introduction

Hair is one of the most common element on human and animal characters. High-quality hair rendering is essential to provide believable appearance in digitally-created content. In this paper, we are interested in rendering hair lit by area and environment lighting, without precomputation to support dynamic scenes. This situation is common in nowadays feature film production, where area lighting and HDRI lighting are becoming more and more prevalent.

Marschner et al. [2003] introduced a physically-based scattering model for hair, that captures all the nuances of its appearance. However, this model is computationally expensive, requiring the solution of a cubic equation derived by internal path analysis. Moreover, it is cumbersome for artists to control the appearance of hair by changing the model's parameters directly. To address these problems, Sadeghi et al. [2010] proposed an artist-friendly shading model for hair that approximate Marschner's model using only elementary functions that are easier for artists to control in production environments. In this paper, we are specially interested in this latter model.

Both these hair shading models have narrow peaks in the specular lobe, especially for shiny hair. This causes severe noise when rendered using *Monte Carlo* methods, especially when combined with large area lights and environment maps. Importance sampling is a widely used variance reduction technique for *Monte Carlo* numerical integration. In the context of rendering, importance sampling offers a mean to reduce the variance by taking more samples in regions with significant contribution to the illumination integral. To the best of our knowledge, how to efficiently importance sample the hair scattering functions referenced above is not known.

In this paper, we present an efficient importance sampling method for hair *bsdf*. Our method is capable of significantly improving the quality of the rendered image, as seen in Figure 1, with negligible overhead. We reduce noise by drawing samples from a distribution that approximate well [Sadeghi et al. 2010]'s scattering function. We do so efficiently since drawing samples only requires the evaluation of a few analytic functions, with no precomputation or significant memory footprint. We found our method easy to implement both in a prototype path tracer and in a large production system based on micropolygon rendering. In both cases, results can be further improved by using multiple importance sampling that takes lighting into account as well.

We believe that the main contribution of our work is to provide the first practical importance sampling algorithm for hair *bsdf* that is very effective at reducing noise, while remaining simple to implement and efficient to evaluate. In the remainder of the paper, we start with a brief overview of related work, follow that with our our algorithm and results presentation, and end with discussion and conclusion.

2 Related Work

Photorealistic Hair Rendering There is a large body of work regarding hair modeling and shading. Here we review only the

ones most closely related to this paper, referring the reader to [Ward et al. 2007] for a detailed review. The first prominent work in the field of hair rendering was proposed by Kajiya and Kay [1989]. It modeled hair *bsdf* by computing light scattering from a thin cylinder. Marschner et al. [2003] improve upon this model by incorporating internal path analysis of hair strands. Marschner's work was the first complete physically-based hair shading model, capable of capturing the complex scattering behavior of hair. By approximating Marschner's model, Sadeghi et al. [2010] derived a practical hair shading model that was more efficient and easier for artist to control. d'Eon et al. [2011] proposed an energy conserving hair reflectance model, which included several modifications to Marschner's model to ensure energy conservative during scattering. These models focused on providing accurate scattering *bsdfs* for hair, but none provided an efficient method to importance sample the scattering functions. This is the focus of our work. While our method will also speed up multiple scattering using Monte Carlo methods, it can also be integrated with more efficient solution that focus on this aspect such as [Moon and Marschner 2006; Moon et al. 2008; Zinke et al. 2008].

Importance Sampling Surface Materials High-quality Monte Carlo rendering requires the ability to importance sample realistic *bsdf* models. There has been extensive research on importance sampling for surface *bsdf*, as summarized in [Pharr and Humphreys 2010]. Analytic methods exist only for the simple *bsdf* such as Phong [Phong 1975], Lafortune [Lafortune et al. 1997] and Ward [Larson 1992]. For more complex *bsdf* and measured material, approximations of varying degree of quality are applied. A more general solution is to derive importance sampling functions by using factorized representations or basis projections of *bsdf* [Lawrence et al. 2004; Clarberg et al. 2005; Jarosz et al. 2009]. However, these methods require precomputation and have high memory footprint, making them impractical for spatially-varying materials. Our method uses an accurate analytic approximation that does not suffer from these constrains.

Importance Sampling Hair There is little published work on importance sampling for hair. Neulander et al. [2010] derived a practical importance sampling algorithm based on a cone-shell hair *bsdf* model, which is a variant of Kajiya Kay model. However, this method does not support hair models that have multiple specular lobes with different width and offset, such as Marschner's hair model or its variants; for which we are not aware of any previous works regarding importance sampling.

Hair Rendering under Environment Lighting Hair rendering under environment lighting will benefit from an efficient *bsdf* importance sampling algorithm. This case is so common that algorithms have been developed specifically for it [Ren et al. 2010; Xu et al. 2011]. While these methods work well in their problem domain, they are limited to environment lighting and require a certain amount of precomputation, which makes them impractical for production rendering. Moreover, they are all derived by directly computing the illumination integral with approximation. The lack of *Monte Carlo* sampling process makes them unsuitable for scenes with dynamic occluders.

3 Hair Importance Sampling

We start the presentation of our importance sampling method with a summary of the hair shading model it supports, follow by the derivation of sampling functions for each lobe of the hair *bsdf* and the complete scheme for importance sampling. In this paper, we follow the notation summarized in Table 1 and Figure 2.

symbol	description
$S(\theta_i, \phi_i, \theta_r, \phi_r)$	hair *bsdf*
M_{R}, M_{TT}, M_{TRT}	longitudinal scattering functions
N_{R}, N_{TT}, $N_{\mathrm{TRT\text{-}g}}$, N_{g}	azimuthal scattering functions
ω_i	incoming direction
ω_r	reflected direction
$\mathbf{u}$	hair direction, pointing from the root to the tip
$\mathbf{v}$, $\mathbf{w}$	axes of the normal plane, orthogonal to $\mathbf{u}$
θ_i, θ_r	inclination of ω_i and ω_r w.r.t the normal plane where 0° is perpendicular to $\mathbf{u}$, 90° is $\mathbf{u}$, and -90° is $-\mathbf{u}$
ϕ_i, ϕ_r	azimuthal angles of ω_i and ω_r in the normal plane where $\mathbf{v}$ is 0° and $\mathbf{w}$ is 90°
ϕ	relative azimuthal angle, $\phi = \phi_r - \phi_i$
θ_d	longitudinal difference angle $\theta_d = (\theta_r - \theta_i)/2$
θ_h	longitudinal half angle $\theta_h = (\theta_r + \theta_i)/2$

Table 1: *Summary of notation.*

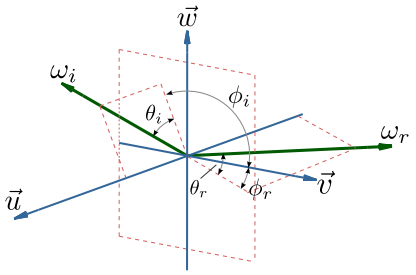

Figure 2: *Local hair coordinate system.*

3.1 Hair Shading Function

Sadeghi et al. [Sadeghi et al. 2010] proposed an artist-friendly hair shading model. In this model, the scattering function $S(\theta_i, \phi_i, \theta_r, \phi_r)$ of hair fibers is decomposed into four individual components: reflection (R), refractive transmission (TT), secondary reflection without glint (TRT-g) and glint (g). Each component is represented as a separate lobe and further factored as the product of a longitudinal term M and a azimuthal term N. The full scattering model is shown as equation (1). Table 2 shows the definition of each individual longitudinal term and azimuthal term used in equation (1).

$$\begin{aligned} S(\theta_i, \phi_i, \theta_r, \phi_r) =& I_{\mathrm{R}} M_{\mathrm{R}}(\theta_h) N_{\mathrm{R}}(\phi) / \cos^2\theta_d + \\ & I_{\mathrm{TT}} M_{\mathrm{TT}}(\theta_h) N_{\mathrm{TT}}(\phi) / \cos^2\theta_d + \\ & I_{\mathrm{TRT}} M_{\mathrm{TRT}}(\theta_h) N_{\mathrm{TRT\text{-}g}}(\phi) / \cos^2\theta_d + \\ & I_{\mathrm{TRT}} M_{\mathrm{TRT}}(\theta_h) I_{\mathrm{g}} N_{\mathrm{g}}(\phi) / \cos^2\theta_d \end{aligned} \quad (1)$$

	M(longitudinal terms)	N(azimuthal terms)	I(intensity)
R	$g(\beta_{\mathrm{R}}^2, \theta_h - \alpha_{\mathrm{R}})$	$\cos(\frac{\phi}{2})$	color vector
TT	$g(\beta_{\mathrm{TT}}^2, \theta_h - \alpha_{\mathrm{TT}})$	$g(\gamma_{\mathrm{TT}}^2, \pi - \phi)$	color vector
TRT-g	$g(\beta_{\mathrm{TRT}}^2, \theta_h - \alpha_{\mathrm{TRT}})$	$\cos(\frac{\phi}{2})$	color vector
g	—	$g(\gamma_{\mathrm{g}}^2, \lvert\phi\rvert - \phi_{\mathrm{g}})$	scalar

Table 2: *Definition of components in equation (1).*

$I_{[\mathrm{R}|\mathrm{TT}|\mathrm{TRT}]}$ is the color/intensity of its corresponding lobe and I_{g} is the additional intensity of the *Glint* lobe. $M_{[\mathrm{R}|\mathrm{TT}|\mathrm{TRT}]}$ are the longitudinal terms. They model the longitudinal variation of each lobe.

All of them are defined as *gaussian* functions of longitudinal half angle θ_h.

$$M_{[\mathrm{R|TT|TRT}]} = g(\beta^2_{[\mathrm{R|TT|TRT}]}, \theta_h - \alpha_{[\mathrm{R|TT|TRT}]})$$

, where $\beta_{[\mathrm{R|TT|TRT}]}$ and $\alpha_{[\mathrm{R|TT|TRT}]}$ are the width and median of corresponding *gaussian* function. $\alpha_{[\mathrm{R|TT|TRT}]}$ controls the highlight shift of each lobe, while modifying $\beta_{[\mathrm{R|TT|TRT}]}$ changes the roughness of the hair. $N_{[\mathrm{R|TT|TRT\text{-}g|g}]}$ are the azimuthal terms. They model the azimuthal variation of each lobe. All azimuthal terms are a function of relative azimuthal angle $\phi = \phi_r - \phi_i$. N_{R} is defined as $\cos(\phi/2)$. N_{TT} is defined as a *gaussian* function with a user controllable azimuthal width γ_{TT}. $N_{\mathrm{TRT\text{-}g}}$ is approximated as $\cos(\phi/2)$ and N_{g} is defined as two *gaussian* functions with width γ_{g} and symmetric about axis $\phi = 0$, where ϕ_{g} is the half angle between the peak of two *Glint*.

3.2 Importance Sampling

To efficiently reduce variance of *Monte Carlo* integration, we want to draw samples from a *pdf* that is proportional to the function we are estimating. In the context of hair rendering, we want to draw sampling of ω_i so that $p(\omega_i) \propto S(\theta_i, \phi_i, \theta_r, \phi_r)$

Because the hair *bsdf* model is a multiple-lobe model, it is impractical to sample all the lobes at the same time. In this section, we will first describe how to efficiently sample each lobe individually. Then, we show how to combine all the lobes by randomly select a lobe base on its energy estimation. Noting that the longitudinal terms and azimuthal terms depend on different variables, they can be sampled independently. Specifically, we sample spherical angle θ_i and ϕ_i separately, then convert them into direction ω_i. The *pdf* of the sample is a product of the longitudinal *pdf* and the azimuthal *pdf*, $p(\omega_i) = p(\theta_i)p(\phi_i)$. We used the invert *cdf* technique [Pharr and Humphreys 2010] to derive our analytic sampling functions.

Sampling *Gaussian* Function Equation (1) use different *gaussian* functions to model the variation in longitudinal and azimuthal scattering (See Table 2). To derive a efficient sampling algorithm, we would need to draw samples from a *gaussian* distribution. However, deriving a sampling algorithm for the *gaussian* distribution is nontrivial; because *gaussian*s do not have closed-form antiderivatives, making it infeasible to use the inverse *cdf* technique directly (See Figure 3). Although there exists approximation for the *pdf* and *cdf* of *gaussian*, they require the evaluation of Error Functions or Taylor series[Pressa et al. 2007]. These methods are either computationally expensive or unstable at the tail of the *gaussian* function

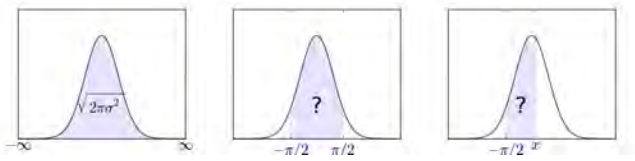

Figure 3: *Gaussian functions only have a closed-form integral in* $\sqrt{2\pi\beta^2}$ *of domain* $(-\infty, \infty)$. *In domain* $[-\pi/2, \pi/2]$ *and* $[-\pi/2, x]$, *Gaussian has no closed-form antiderivative, making it infeasible to compute the* pdf *or the* cdf *needed for deriving importance sampling.*

To overcome this limitation, we would like to draw samples from a *pdf* that has a similar shape to the *gaussian* function and a closed-form antiderivative. Observing that *gaussian* is a bell-shape function with varying width and offset, we can use approximate it using another bell-shape function.

***Cauchy* distribution** *Cauchy* distribution is a probability distribution mainly used physics research. It is defined as follows:

$$f(\gamma, x - x_0) = \frac{1}{\pi}\left[\frac{\gamma}{(x - x_0)^2 + \gamma^2}\right]$$

Similar to *gaussian*, *cauchy* distribution is a bell shape function with offset x_0 and width γ. In contrast to *gaussian*, *cauchy* has an analytic antiderivative.

$$P(x) = \frac{1}{\pi}\tan^{-1}\left(\frac{x - x_0}{\gamma}\right)$$

This simple form of antiderivative make it very easy to derive a sampling algorithm using the inverse *cdf* technique. Also, the parameter conversion between two distributions is trival. The offset and width of a *gaussian* distribution can be directly used as the offset and width of a *cauchy* distribution. Figure 4 shows the plot of a set *gaussian* functions and *cauchy* functions with same widths and offsets. The fact that *cauchy* distributions have wider tails than *gaussian* guarantees that using *cauchy* to approximate the *gaussian* in importance sampling will not increase variance. Using this approximation, we derive our sampling method for each lobe.

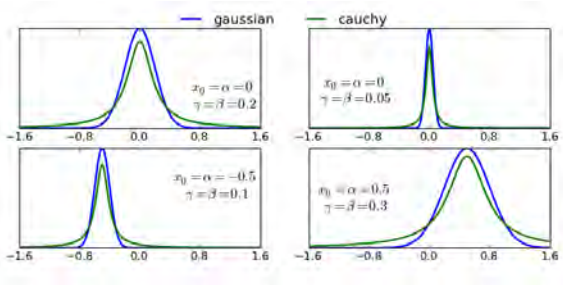

Figure 4: *cauchy and gaussian distributions with same width and offset. Both of distributions are normalized in domain* $[-\pi/2, \pi/2]$

3.2.1 Sampling Longitudinal Terms

Since three longitudinal terms have the same form, we generalize the approach by using symbols M, β and α instead of $M_{[\mathrm{R|TT|TRT}]}$, $\beta_{[\mathrm{R|TT|TRT}]}$ and $\alpha_{[\mathrm{R|TT|TRT}]}$. Note that we ignore the $1/\cos^2\theta_d$ terms for the simplicity, since $M_{[\mathrm{R|TT|TRT}]}$ alone accounts for most of the variation of the longitudinal term. Substituting the *gaussian* functions in the M terms with *cauchy* distributions allows us to derive the sampling functions for incoming inclination θ_i(See derivation in appendix A). Given a random variable ξ uniformly drawn from range $[0, 1]$, we can sample θ_i as:

$$\theta_i = 2\beta\tan(\xi(A - B) + B) + 2\alpha - \theta_r$$

, where $A = \tan^{-1}\left(\frac{\frac{\pi/2 + \theta_r}{2} - \alpha}{\beta}\right)$ and $B = \tan^{-1}\left(\frac{\frac{-\pi/2 + \theta_r}{2} - \alpha}{\beta}\right)$. A and B are the generalization of variables $A_{[\mathrm{R|TT|TRT}]}$ and $B_{[\mathrm{R|TT|TRT}]}$. The longitudinal *pdf* can be computed as

$$p(\theta_i) = \frac{1}{2\cos\theta_i(A - B)}\frac{\beta_x}{(\theta_h - \alpha)^2 + \beta^2}$$

3.2.2 Sampling Azimuthal Terms

All azimuthal terms are a function of relative azimuthal angle $\phi = \phi_r - \phi_i$. In our approach, we first sample ϕ, then compute $\phi_i =$

range of ξ	sign of ϕ	remapping
$\xi < \frac{1}{2}$	$+$	$\xi = 2\xi$
$\xi \geq \frac{1}{2}$	$-$	$\xi = 2(1-\xi)$

Table 3: *Selecting the* Glint *lobe base on the range of random variable ξ_2 and remap it back to range* $[0, 1]$

$\phi_r - \phi$. We also need to transform the *pdf* of ϕ to the *pdf* of ϕ_i, but it can be proved that

$$p(\phi_i) = p(\phi) \left| \frac{d\phi_i}{d\phi} \right|^{-1} = p(\phi)$$

Sampling $N_{\mathbf{R}}$. N_{R} is evaluated as $\cos(\phi/2)$. Deriving a sampling function for this term is trivial (See Appendix B for derivation). Given a uniform random variable ξ from $[0, 1]$, we can sample ϕ as

$$\phi = 2\sin^{-1}(2\xi - 1)$$

then we can compute $\phi_i = \phi_r - \phi$ and azimuthal *pdf*, $p(\phi_i) = p(\phi) = \frac{1}{4}\cos\frac{\phi}{2}$.

Sampling $N_{\mathbf{TT}}$. N_{TT} is driven by a *gaussian* function, which has positive value in range $[0, 2\pi]$. We take an approach similar to the longitudinal terms (See derivation in Appendix C). Given a uniform random variable ξ from $[0, 1]$, we can to draw a sample of ϕ as

$$\phi = \gamma_{\mathrm{TT}} \tan\left[C_{\mathrm{TT}}\left(\xi - \frac{1}{2}\right)\right] + \pi$$

, where $C_{\mathrm{TT}} = 2\tan^{-1}\left(\frac{\pi}{\gamma_{\mathrm{TT}}}\right)$. Then we can compute $\phi_i = \phi_r - \phi$ and the azimuthal *pdf*, $p(\phi_i) = p(\phi) = \frac{1}{C_{\mathrm{TT}}}\left[\frac{\gamma_{\mathrm{TT}}}{(\phi-\pi)^2+\gamma_{\mathrm{TT}}^2}\right]$

Sampling $N_{\mathbf{TRT\text{-}g}}$ $N_{\text{TRT-g}}$ is approximated as $\cos(\phi/2)$. Since it is the same as the N_{R} term, we follow the same approach as sampling N_{R}.

Sampling *Glint* *Glint* lobe models the lighting effect cause by the caustic light path inside hair strands. The azimuthal term of *Glint* is defined as two *gaussian* functions symmetric about axis $\phi = 0$. For every random sample, we only choose one *Glint* to sample. Specifically, given a uniform random variable ξ from $[0, 1]$, we first choose one of two *Glint*s by setting the sign of ϕ base on the range of ξ. Then, we remap the ξ back to range $[0, 1]$ as shown in Table 3. After that we can sample ϕ using the remapped ξ (see derivation in Appendix D).

$$\phi = \gamma_{\mathrm{g}} \tan(\xi(C_{\mathrm{g}} - D_{\mathrm{g}}) + D_{\mathrm{g}}) + \phi_{\mathrm{g}}$$

, where $C_{\mathrm{g}} = \tan^{-1}\left(\frac{\pi/2-\phi_{\mathrm{g}}}{\gamma_{\mathrm{g}}}\right)$ and $D_{\mathrm{g}} = \tan^{-1}\left(\frac{-\phi_{\mathrm{g}}}{\gamma_{\mathrm{g}}}\right)$

Once we have ϕ, we can compute $\phi_i = \phi_r \pm \phi$. The sign of ϕ is determined by the lobe we selected in the remapping stage (Table **??**). We also need to transform the *pdf* to account for the fact that we remapped the random variable. The resulting azimuthal *pdf* is $p(\phi_i) = \frac{1}{2}p(\phi) = \frac{1}{2(C_{\mathrm{g}}-D_{\mathrm{g}})}\left[\frac{\gamma_{\mathrm{g}}}{(|\phi|-\phi_{\mathrm{g}})^2+\gamma_{\mathrm{g}}^2}\right]$

3.2.3 Energy based lobe selection

We have discussed how to sample each lobe individually. To sample the complete *bsdf*, we need to distribute samples according to the energy distribution of the *bsdf*. For each random sample, this is achieved by selecting a lobe with a probability proportional to its total energy estimation. We constructing a discrete *cdf* by using the energy estimation of each lobe. The energy of each lobe is approximately computed as a product of the integral of each lobe's longitudinal term and azimuthal term. The following is the estimated energy for each lobe.

$$\mathrm{E_R} = 4\sqrt{2\pi}\beta_{\mathrm{R}} I_{\mathrm{R}} \qquad \mathrm{E_{TT}} = 2\pi\beta_{\mathrm{TT}}\gamma_{\mathrm{TT}} I_{\mathrm{TT}}$$
$$\mathrm{E_{TRT\text{-}g}} = 4\sqrt{2\pi}\beta_{\mathrm{TRT}} I_{\mathrm{TRT}} \qquad \mathrm{E_g} = 4\pi\beta_{TRT}\gamma_{\mathrm{g}} I_{\mathrm{TRT}} I_{\mathrm{g}}$$

Noting that we are using the *gaussian* integral of domain $[-\infty, \infty]$ instead of $[-\pi/2, \pi/2]$ to compute the energy estimation. Although it is not accurate, but it is easy to compute and works fine as an estimation. The approximation error is less than 1% for $\beta < 30^\circ$ and $|\alpha| < 20^\circ$.

3.3 Implementation Details

Amortizing constants computation It is important to note that, $A_{[\mathrm{R}|\mathrm{TT}|\mathrm{TRT}]}, B_{[\mathrm{R}|\mathrm{TT}|\mathrm{TRT}]}, C_{\mathrm{TT}}, C_{\mathrm{g}}$ and D_{g} in the sampling functions are constants for all the samples of the same gather point and reflective direction ω_r,. We only need to compute those constants once and amortize the cost for all the samples.

Longitudinal grazing angle *pdf* Notice that in the longitudinal *pdf* has a singularity when θ_i approaches $-\frac{\pi}{2}$ or $\frac{\pi}{2}$. The sample evaluation become numerically unstable at grazing angles. To avoid this problem, in our implementation, we discard the sample if angle between ω_i and $\mathbf{u}$ or $-\mathbf{u}$ is smaller then a predefined epsilon (10^{-5} in our case). Although in theory this may bias the result when the *gaussian* is wide, in practice this is not a problem since most longitudinal lobe for hair tend to be sharp.

Replacing *Gaussian* with *Cauchy* In this paper, we derive our *pdf* based on a artist friendly model used *gaussian* extensively. Although *cauchy* distribution can provide a good sample distribution for *gaussian* function(Figure 5), the shape of two distribution do not match exactly (see Figure 4). As an extension of our work, we propose a new hair *bsdf* by replacing all the *gaussian* functions in the old model with *cauchy* distributions. This new *bsdf* model introduces some very minor visual differences, but it is able to achieve a better sampling performance. The comparison between the new *bsdf* and the published artist friendly model is out of the scope of this paper. Interested readers can find comparisons using the new model in our supplemental material. All the result in this paper is generated using the published artist friendly model.

4 Results

4.1 Sample Distribution

Figure 5 shows the sample distributions using the described importance sampling scheme. All the random number is generated using a *Halton Sequence*, which is already stratified[Pharr and Humphreys 2010]. Compare to uniform sampling (Figure 5.A), our importance sampling method (Figure 5.B) is able to concentrate samples to the area with high energy. Figure 5.C-F show the sample distribution of each individual lobe.

4.2 Rendering Result

Overview We implemented our importance sampling scheme for hair *bsdf* in a raytracing renderer written in C++. The source code

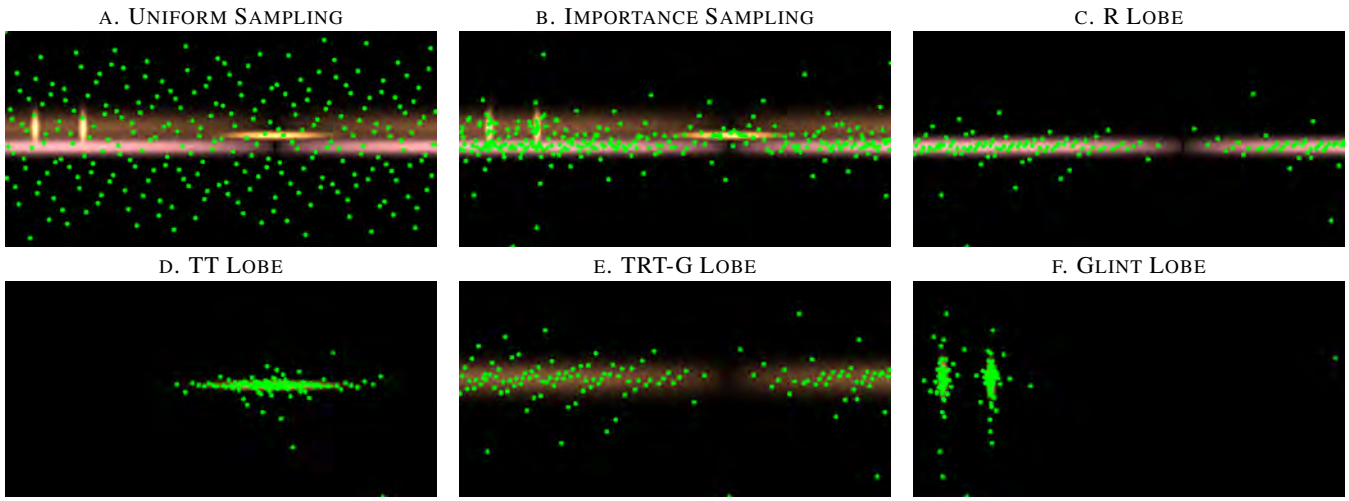

Figure 5: *Comparison of sample distributions using uniform sampling (a) and hair* bsdf *importance sampling (b). We also shown sample distributions of each individual lobes. (c) Importance samples of R lobe. (d) Importance samples of TT lobe, (e) Importance samples of TRT lobe without Glint. (f) Importance samples of Glint. Sample distributions show that our approach can place samples match the energy distribution of the* bsdf.

of this C++ implementation is available in the supplemental material. Moreover, to test our approach in a movie production environment, we also implemented the described algorithm in a production pipeline based on rasterization and micro-polygon rendering. Figure 6 shows a comparison of our method to stratified uniform sampling. All the image are rendered using multiple importance sampling (MIS). Therefore sample count 16 means 16 *bsdf* samples and 16 light samples. In supplemental material, we included all the rendering images with its native resolution and animation sequences.

Area Lighting Area light is a surface that emits radiance. It is essential to computing physically plausible images. Computing direct lighting with area light can be done by sampling the light, the surface *bsdf*, or both (*MIS*). When the area light is large or the surface *bsdf* is high glossy, light sampling approach become inefficient. Moreover, the present of occluders can also make light sampling insufficient. In these cases, it is important to have efficient *bsdf* importance sampling.

Figure 6.A is a simple scene with a large area light above the hair geometry rendered with our raytracing implementation. We can see the significant noise is shown using uniform sampling with small number of samples(32 samples), while the importance sampling result is relatively smooth. With 256 samples, importance sampling image has no visible noise, while uniform sampling image still has some distracting noise.

Figure 7.A is a production model render with the production pipeline. Noting that with sophisticated anti-aliasing and filtering algorithm in the production pipeline, it requires significantly less samples then our simple raytracing implementation to get a visually pleasant image. The comparison still shows obvious improvement when using our importance sampling scheme.

Environment Lighting Environment light is able to provide realistic illumination to the scene. Importance sampling the environment map can be done by building a 2D *cdf* [Pharr and Humphreys 2010]. However, light sampling can become in efficient when the illumination in the environment map is smooth.

Figure 6.B is a simple scene with a environment map of *Grace Church*. In this case, while light sampling is able to quickly capture the highlight on top of the hair, it because quite inefficient for the highlight at the lower part of the hair cloud. While uniform sampling is not able to clean up the noise of cause by the *Glint* with 256 samples, our importance sampling method is able to provide a smooth result using with relatively small amount of samples (64 samples).

Figure 7.B is a production scene rendered with a environment map of *Ennis-Brown House*. While uniform sampling is not able to clean up the noise of cause by the *Glint* with X samples, our importance sampling method is able to provide a smooth result using with relatively small samples (X samples).

Indirect Lighting Indirect illumination is an important component of global illumination that provides realistic appearance of a scene. Unlike direct illumination, indirect lighting is unstructured. Therefore light sampling and *MIS* cannot apply in this situation. Normally, scattering ray is generated by only *bsdf* sampling.

Figure 6.C is a simple scene with a hair model inside the cornell box. In the scene, we turn off the multiple scattering inside the hair cloud. The comparison shows our importance sampling algorithm is able to generate images converge much quicker than uniform sampling. In the setting of Figure 6.D, we allow multiple scattering inside the hair cloud. Although that requires a longer rendering time and higher number of samples for both methods. It is clear that our importance sampling method is able to converge to the correct result much quicker than stratified uniform sampling.

5 Discussion and Limiation

Sampling Efficiency Although our importance sampling method always yields better sample efficiency than uniform sampling, the magnitude of improvement highly depend on the setting of the scene and the hair *bsdf*. In the case of rough hair and high frequency environment map, the improvement is less obvious because in that case the light sampling become dominant in the MIS weighting.

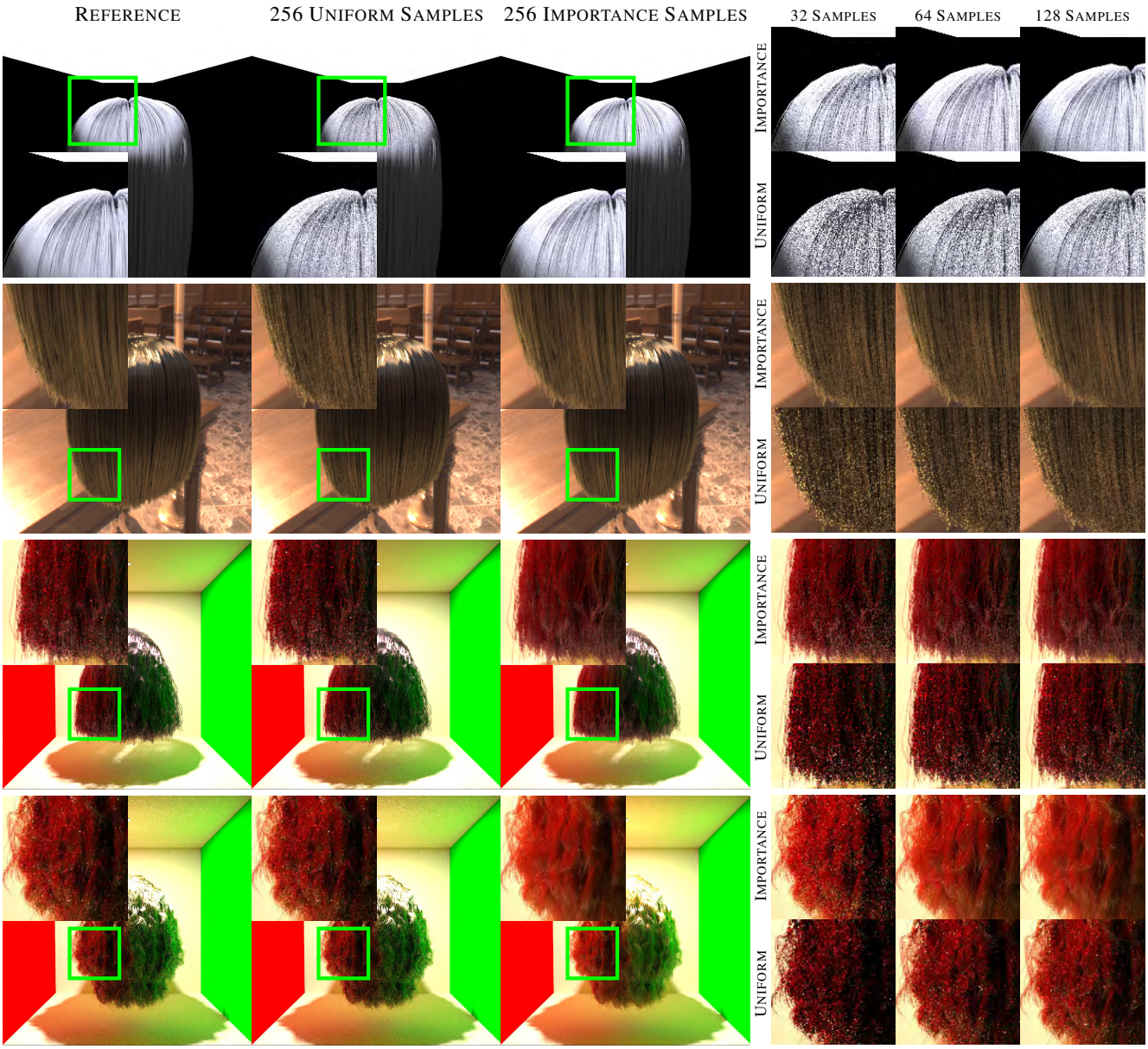

Figure 6: *Comparison of our importance sampling approach and simple uniform sample. All the images are render with MIS turned on. (A) Raytracing direct Illumination with a large area light lies along the direction of the hair strands. Light sampling is inefficient in this case. Our method capture the longitudinal variance better than uniform sampling. (B) Raytracing direct Illumination with environment lighting. Our method generate smooth result with small number of samples while uniform sampling failed to capture the glint and transmission highlight. (C) Indirect lighting without multiple scattering inside hair cloud. (D) Indirect lighting with multiple scattering inside hair cloud.*

Multiple Scattering Our importance sampling algorithm is derived for the single scattering function. We shown multiple scattering results rendered by path tracing, which performance is drastically improved by using our importance sampling algorithm. Approximation algorithms for multiple scattering is out of the scope of this paper. Although our sampling algorithm is not specifically designed for multiple scattering approximation algorithm, we believe those algorithms will still benefit from our work. For example, our importance sampling can be used to drive the photon shooting of [Moon and Marschner 2006] and light tracing of [Moon et al. 2008].

Integration with Other Sampling Techniques Since our sampling algorithm does not require any additional data structures, it can be easily integrated with other sampling techniques. We have only shown our method using in a raytracer with multiple importance sampling, but it can also be used with other *Monte Carlo* techniques, e.g. photon mapping, bidirectional path tracing, or even more sophisticated unstructured illumination sampling technique[Wang and Åkerlund 2009].

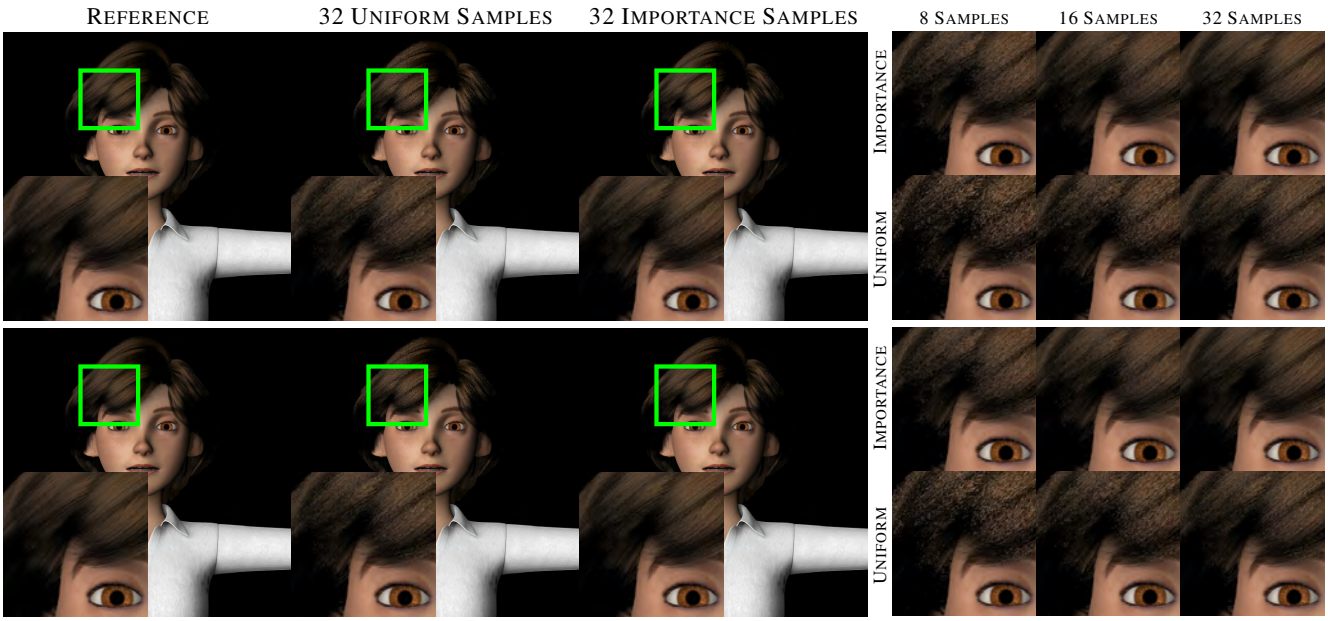

Figure 7: *Comparison of our importance sampling approach and simple uniform sample. Production models with pipeline.*

6 Conclusions and Future Work

In this paper, we presented a practical importance sampling algorithm for hair *bsdf*, which is simple to implement and efficient to evaluate. By approximating the *gaussian* functions in the *bsdf* function using *cauchy* distribution, we are able to derive an analytic sampling algorithm, which in many cases reduces variance and sampling times compare to stratified uniform sampling. We use our importance sampling method to render scenes with area lighting, environment lighting, indirect lighting and multiple scattering.

In future work, we would like to extend our technique to perform importance sampling for multiple scattering lobes. We would also like to extend it to support different hair *bsdf* models, e.g. the Marschner's model and the energy conservative model. We also interested in integrating our work with different sampling techniques e.g. Metropolis Raytracing. Generally, we see our work as a first step towards efficient sampling technique for hair *bsdf*. With the increasing importance of physically based technique and accurate hair shading in production rendering, this problem promises to have growing significance.

References

CLARBERG, P., JAROSZ, W., AKENINE-MÖLLER, T., AND JENSEN, H. W. 2005. Wavelet importance sampling: efficiently evaluating products of complex functions. *ACM Transactions on Graphics 24*, 3 (Aug.), 1166–1175.

D'EON, E., FRANCOIS, G., HILL, M., LETTERI, J., AND AUBRY, J. 2011. An energy-conserving hair reflectance model. *Eurographics Symposium on Rendering 2011 30*, 4, 1181–1187.

JAROSZ, W., CARR, N. A., AND JENSEN, H. W. 2009. Importance sampling spherical harmonics. *Computer Graphics Forum 28*, 2 (Apr.), 577–586.

KAJIYA, J. T., AND KAY, T. L. 1989. Rendering fur with three dimensional textures. In *Computer Graphics (Proceedings of SIGGRAPH 89)*, 271–280.

LAFORTUNE, E. P. F., FOO, S.-C., TORRANCE, K. E., AND GREENBERG, D. P. 1997. Non-linear approximation of reflectance functions. In *Proceedings of SIGGRAPH 97*, Computer Graphics Proceedings, Annual Conference Series, 117–126.

LARSON, G. J. W. 1992. Measuring and modeling anisotropic reflection. In *Computer Graphics (Proceedings of SIGGRAPH 92)*, 265–272.

LAWRENCE, J., RUSINKIEWICZ, S., AND RAMAMOORTHI, R. 2004. Efficient brdf importance sampling using a factored representation. *ACM Transactions on Graphics 23*, 3 (Aug.), 496–505.

MARSCHNER, S. R., JENSEN, H. W., CAMMARANO, M., WORLEY, S., AND HANRAHAN, P. 2003. Light scattering from human hair fibers. *ACM Transactions on Graphics 22*, 3 (July), 780–791.

MOON, J. T., AND MARSCHNER, S. R. 2006. Simulating multiple scattering in hair using a photon mapping approach. *ACM Transactions on Graphics 25*, 3 (July), 1067–1074.

MOON, J. T., WALTER, B., AND MARSCHNER, S. 2008. Efficient multiple scattering in hair using spherical harmonics. *ACM Transactions on Graphics 27*, 3 (Aug.), 31:1–31:7.

NEULANDER, I. 2010. Fast furry ray gathering. In *ACM SIGGRAPH 2010 Talks*, ACM, 2.

PHARR, M., AND HUMPHREYS, G. 2010. *Physically Based Rendering, Second Edition: From Theory To Implementation*, 2nd ed. Morgan Kaufmann Publishers Inc.

PHONG, B. T. 1975. Illumination for computer generated pictures. *Commun. ACM 18* (June), 311–317.

PRESSA, W., TEUKOLSKY, S., VETTERLING, W., AND FLANNERY, B. 2007. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, The Edinburgh Building, Cambridge, UK.

REN, Z., ZHOU, K., LI, T., HUA, W., AND GUO, B. 2010. Interactive hair rendering under environment lighting. *ACM Transactions on Graphics 29*, 4 (July), 55:1–55:8.

SADEGHI, I., PRITCHETT, H., JENSEN, H. W., AND TAMSTORF, R. 2010. An artist friendly hair shading system. *ACM Transactions on Graphics 29*, 4 (July), 56:1–56:10.

WANG, R., AND ÅKERLUND, O. 2009. Bidirectional importance sampling for unstructured direct illumination. *Computer Graphics Forum 28*, 2 (Apr.), 269–278.

WARD, K., BERTAILS, F., KIM, T.-Y., MARSCHNER, S. R., CANI, M.-P., AND LIN, M. C. 2007. A survey on hair modeling: Styling, simulation, and rendering. *IEEE Transactions on Visualization and Computer Graphics 13*, 213–234.

XU, K., MA, L.-Q., REN, B., WANG, R., AND HU, S.-M. 2011. Interactive hair rendering and appearance editing under environment lighting. *ACM Trans. Graph. 30* (Dec.), 173:1–173:10.

ZINKE, A., YUKSEL, C., WEBER, A., AND KEYSER, J. 2008. Dual scattering approximation for fast multiple scattering in hair. *ACM Transactions on Graphics 27*, 3 (Aug.), 32:1–32:10.

A Longitudinal Sampling Derivation

Given a uniform random variable ξ from $[0,1]$, we want to draw a sample of θ_i from *pdf*.

$$p(\theta_i) \propto \left[\frac{\beta}{(\frac{\theta_i+\theta_r}{2}-\alpha)^2+\beta^2}\right]\frac{1}{\cos\theta_i}$$

Noting that $1/\cos\theta_i$ term is there to compensate the correcting factor for transforming the integral over solid angle into integrals over spherical coordinates. The normalization gives that

$$\begin{aligned}&\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} c\left[\frac{\beta}{(\frac{\theta_i+\theta_r}{2}-\alpha)^2+\beta^2}\right]\frac{1}{\cos\theta_i}\cos\theta_i d\theta_i\\ =\;& 2c\tan^{-1}\left(\frac{\theta_i-\alpha}{\beta}\right)\Big|_{\frac{-\pi/2+\theta_r}{2}}^{\frac{\pi/2+\theta_r}{2}} = 1\end{aligned}$$

Therefore $c = \frac{1}{2(A-B)}$, where $A = \tan^{-1}\left(\frac{\frac{\pi/2+\theta_r}{2}-\alpha}{\beta}\right)$ and $B = \tan^{-1}\left(\frac{\frac{-\pi/2+\theta_r}{2}-\alpha}{\beta}\right)$. The *pdf* of the θ_i is:

$$p(\theta_i) = \frac{1}{2\cos\theta_i(A-B)}\frac{\beta}{\left(\frac{\theta_i+\theta_r}{2}-\alpha\right)^2+\beta^2}$$

The *cdf* can be computed by integrating the *pdf*

$$\begin{aligned}P(\theta_i) &= \int_{-\frac{\pi}{2}}^{\theta_i} c\left[\frac{\beta}{(\frac{\theta_i'+\theta_r}{2}-\alpha)^2+\beta^2}\right]\frac{1}{\cos\theta_i'}\cos\theta_i' d\theta_i'\\ &= \frac{\tan^{-1}\left(\frac{\frac{\theta_i+\theta_r}{2}-\alpha}{\beta}\right)-B}{A-B}\end{aligned}$$

By inverting the *cdf*, we can sample θ_i, given a uniform random variable ξ from $[0,1]$

$$\theta_i = 2\beta\tan(\xi(A-B)+B)+2\alpha-\theta_r$$

B N_{R} and $N_{\mathrm{TRT\text{-}g}}$ Sampling Derivation

Given a uniform random variable ξ from $[0,1]$, we want to draw a sample of ϕ from *pdf*.

$$p(\phi) \propto \cos\frac{\phi}{2}$$

The normalization gives that

$$\int_{-\pi}^{\pi} c\cos\frac{\phi}{2}d\phi = c\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} 2\cos x dx = 2c\sin x\Big|_{-\frac{\pi}{2}}^{\frac{\pi}{2}} = 4c = 1$$

Therefore, $c = 1/4$. The *pdf* of ϕ is

$$p(\phi) = \frac{1}{4}\cos\frac{\phi}{2}$$

The *cdf* can be computed by integrating the *pdf*

$$\int_{-\pi}^{\phi}\frac{1}{4}\cos\frac{\phi'}{2}d\phi = \frac{1}{2}\sin x\Big|_{-\frac{\pi}{2}}^{\frac{\phi}{2}} = \frac{1}{2}\left(\sin\frac{\phi}{2}+1\right)$$

By inverting the *cdf*, we can sample ϕ, given a uniform random variable ξ_2 from $[0,1]$

$$\phi = 2\sin^{-1}(2\xi_2-1)$$

4 Then we can compute $\phi_i = \phi_r - \phi$. We also need to transform the *pdf* $p(\phi)$ to $p(\phi_i)$ and it can be proved that $p(\phi_i) = \left|\frac{d\phi_i}{d\phi}\right|^{-1} p(\phi) = p(\phi)$

C N_{TT} Sampling Derivation

Given a uniform random variable ξ from $[0,1]$, we want to draw a sample of ϕ from *pdf*.

$$p(\phi) \propto \frac{\gamma_{\mathrm{TT}}}{(\phi-\pi)^2+\gamma_{\mathrm{TT}}^2}$$

The normalization gives that

$$\int_0^{2\pi} c\left[\frac{\gamma_{\mathrm{TT}}}{(\phi-\pi)^2+\gamma_{\mathrm{TT}}^2}\right]d\phi = c\left[\tan^{-1}\left(\frac{\phi-\pi}{\gamma_{\mathrm{TT}}}\right)\right]\Big|_0^{2\pi} = 1$$

Therefore $c = \frac{1}{C_{\mathrm{TT}}}$ where $C_{\mathrm{TT}} = 2\tan^{-1}(\pi/\gamma_{\mathrm{TT}})$. Then we can compute the *pdf* of ϕ

$$p(\phi) = \frac{1}{C_{\mathrm{TT}}}\left[\frac{\gamma_{\mathrm{TT}}}{(\phi-\pi)^2+\gamma_{\mathrm{TT}}^2}\right]$$

The *cdf* can be computed by integrating the *pdf*

$$\begin{aligned}\int_0^{\phi} c\left[\frac{\gamma_{\mathrm{TT}}}{(\phi'-\pi)^2+\gamma_{\mathrm{TT}}^2}\right]d\phi' &= \frac{1}{C_{\mathrm{TT}}}\left[\tan^{-1}\left(\frac{\phi'-\pi}{\gamma_{\mathrm{TT}}}\right)\right]\Big|_0^{\phi}\\ &= \frac{\tan^{-1}\left(\frac{\phi-\pi}{\gamma_{\mathrm{TT}}}\right)}{C_{\mathrm{TT}}}+\frac{1}{2}\end{aligned}$$

By inverting the *cdf*, we can sample ϕ, given a uniform random variable ξ from $[0,1]$

$$\phi = \gamma_{\mathrm{TT}}\tan\left[C_{\mathrm{TT}}\left(\xi-\frac{1}{2}\right)\right]+\pi$$

Then we can compute $\phi_i = \phi_r - \phi$ and $p(\phi_i) = p(\phi)$

D N_{g} Sampling Derivation

Given a uniform random variable ξ from $[0,1]$, we want to draw samples of ϕ from a *pdf*.

$$p(\phi) \propto \frac{\gamma_{\mathrm{g}}}{(|\phi|-\phi_{\mathrm{g}})^2+\gamma_{\mathrm{g}}^2}$$

We first use ξ to randomly pick a half of the lobe and remap the random variable ξ_2 back to $[0,1]$. Then we sample ϕ in the domain $[0,\pi/2]$, and change its sign to its corresponding half of the lobe. The normalization gives

$$\int_0^{\pi/2} c\left[\frac{\gamma_{\mathrm{g}}}{(|\phi|-\phi_{\mathrm{g}})^2+\gamma_{\mathrm{g}}^2}\right]d\phi = c\left[\tan^{-1}\left(\frac{\phi-\phi_{\mathrm{g}}}{\gamma_{\mathrm{g}}}\right)\right]\Big|_0^{\pi/2} = 1$$

Therefore $c = \frac{1}{C_{\mathrm{g}}-D_{\mathrm{g}}}$ where $C_{\mathrm{g}} = \tan^{-1}\left(\frac{\pi/2-\phi_{\mathrm{g}}}{\gamma_{\mathrm{g}}}\right)$ and $D_{\mathrm{g}} = \tan^{-1}\left(\frac{-\phi_{\mathrm{g}}}{\gamma_{\mathrm{g}}}\right)$. we can compute the *pdf* of ϕ

$$p(\phi) = \frac{1}{C_{\mathrm{g}}-D_{\mathrm{g}}}\left[\frac{\gamma_{\mathrm{g}}}{(\phi-\phi_{\mathrm{g}})^2+\gamma_{\mathrm{g}}^2}\right]$$

The *cdf* can be computed by integrating the *pdf*

$$\begin{aligned}\int_0^{\phi} c\left[\frac{\gamma_{\mathrm{g}}}{(\phi'-\phi_{\mathrm{g}})^2+\gamma_{\mathrm{g}}^2}\right] &= \frac{1}{C_{\mathrm{g}}-D_{\mathrm{g}}}\left[\tan^{-1}\left(\frac{\phi'-\phi_{\mathrm{g}}}{\gamma_{\mathrm{g}}}\right)\right]\Big|_0^{\phi}\\ &= \frac{\tan^{-1}\left(\frac{\phi-\phi_{\mathrm{g}}}{\gamma_{\mathrm{g}}}\right)-D_{\mathrm{g}}}{C_{\mathrm{g}}-D_{\mathrm{g}}}\end{aligned}$$

We can sample ϕ, given a uniform random variable ξ from $[0, 1]$

$$\phi = \gamma_{\mathrm{g}} \tan(\xi(C_{\mathrm{g}} - D_{\mathrm{g}}) + D_{\mathrm{g}}) + \phi_{\mathrm{g}}$$

Then we can compute $\phi_i = \phi_r \pm \phi$. The sign of ϕ is determined by the value of the original random variable ξ before remapping. Moreover we also need to transform the *pdf* to account for the fact that we remapped the random variable.

$$p(\phi_i) = \frac{1}{2} p(|\phi|) = \frac{1}{2(C_{\mathrm{g}} - D_{\mathrm{g}})} \left[\frac{\gamma_{\mathrm{g}}}{(|\phi_r - \phi_i| - \phi_g)^2 + \gamma_{\mathrm{g}}^2} \right]$$