

Importance Sampling with Floyd-Steinberg Halftoning

László Szirmay-Kalos, László Szécsi, and Anton Penzov

Department of Control Engineering and Information Technology, Budapest University of Technology, Hungary
Email: szirmay@iit.bme.hu

Abstract

This paper proposes a deterministic importance sampling algorithm for complex integrands. The idea is based on the recognition that halftoning algorithms are equivalent to importance sampling if the gray-scale image and a resulting white pixel are considered as the target importance function and the sampling position, respectively. We adopt the Floyd-Steinberg halftoning algorithm, extend it to higher dimensions, and rephrase it as a sampling method. As the Floyd-Steinberg halftoning places a sample also considering where other samples are located, our sampling algorithm distributes samples in a stratified way. In order to demonstrate the power of the method, we present an environment mapping application where the sampling mimics the product of the cosine weighted BRDF, environment radiance, and the environment visibility.

1. Introduction

Monte-Carlo and quasi-Monte Carlo quadrature rules generate M samples $\mathbf{z}_1, \dots, \mathbf{z}_M$ with density $p(\mathbf{z})$ and approximate an integrand as follows:

$$\Phi = \int_{\mathcal{P}} F(\mathbf{z}) d\mathbf{z} \approx \frac{1}{M} \sum_{i=1}^M \frac{F(\mathbf{z}_i)}{p(\mathbf{z}_i)},$$

where \mathcal{P} is the domain of the integration, and $F(\mathbf{z})$ is the integrand at sample \mathbf{z} . In order to generate samples, we should take uniformly distributed deterministic or pseudo-random numbers in the unit cube and transform them to the domain of the integration. Let us call this cube \mathcal{U} as the *primary sample space* [KSKAC02], and denote the transformation from here to domain \mathcal{P} by $\mathbf{z} = \mathbf{z}(\mathbf{u})$. Our original integral can also be obtained as an integral in the primary sample space:

$$\Phi = \int_{\mathcal{U}} F(\mathbf{z}(\mathbf{u})) \cdot J(\mathbf{u}) d\mathbf{u}, \quad J(\mathbf{u}) = \left| \frac{d\mathbf{z}(\mathbf{u})}{d\mathbf{u}} \right|$$

where $J(\mathbf{u})$ is the Jacobi determinant of the mapping. Intuitively, the Jacobi determinant expresses the local expansion between two corresponding spaces \mathcal{U} and \mathcal{P} .

The error of the quadrature can be reduced if the samples are generated with a density that is at least approximately proportional to the integrand. This variance reduction technique is called *importance sampling*.

In order to specify what approximate proportionality means and to extend importance sampling for vector valued functions, we define a *scalar importance function* $I(\mathbf{u})$ for which exact proportionality is expected. For vector valued functions, this scalar contribution function can represent the average of the vector elements, which is denoted by $\mathcal{L}(F)$. The goal is then to find $I(\mathbf{u})$ that mimics the integrand, i.e.

$$I(\mathbf{u}) \approx \mathcal{L}(F(\mathbf{z}(\mathbf{u}))) \cdot J(\mathbf{u})$$

as accurately as possible, and sample the primary sample space \mathcal{U} with a density $p(\mathbf{u})$ that is proportional to the scalar contribution function: $p(\mathbf{u}) = I(\mathbf{u})/b$. Scalar b comes from the requirement of normalization for the density, resulting in $b = \int_{\mathcal{U}} I(\mathbf{u}) d\mathbf{u}$.

The goals of making the scalar importance mimic the integrand and sampling proportionally with the scalar importance function are often contradicting. If the first objective is met, then the scalar importance function is far too complex to allow the transformation of uniformly distributed samples. Note that this would require the integration of the required density and the symbolic inverse of the integral.

When solving the rendering equation, the integrand is the product of the emission radiance of the path at one end, the cosine weighted BRDFs of the visited points, and the measuring function of the eye at the other end. BRDF sampling obtains a random direction that mimics the cosine weighted

BRDF, light source sampling finds a point with a probability that is proportional to the emission. Note the BRDF sampling and light source sampling consider just a single factor from the integrand and ignore the others.

Rejection sampling and their special versions as the *Metropolis method* [VG97], *Bi-directional importance sampling* [BGH05], *Importance resampling* [BGH04, TCE05] or *Hierarchical thresholding* [ARBJ03, ODJ04, RCL⁺08] generate samples with an easy density, e.g. with BRDF sampling, then the samples are randomly rejected and re-weighted to better mimic the target distribution.

The other problem of classical importance sampling is that it distorts the original distribution, thus its appealing properties like stratification are also corrupted, which is made even worth by additional random rejections. To address this, in the context of environment mapping Wan et al. [WWL05] proposed sampling directly on the sphere. Agarwal et al. [ARBJ03] ensured stratification by an additional clustering step.

1.1. Halftoning

Halftoning is a technique to render gray-scale images on a black and white display. The idea is to put more white points at brighter areas and less points at darker parts. The spatial density of white points in a region around a pixel is expected to be proportional to the gray level of that particular pixel. If we consider the gray level of the original image to be a scalar importance function and the white pixels of the resulting image to be sample locations, then we can realize that halftoning is equivalent to a deterministic importance sampling algorithm. This holds for an arbitrary halftoning algorithm, including the random and ordered halftoning methods that add random noise or a periodic pattern to the original image before quantization, or, for example, the Floyd-Steinberg algorithm [FS75]. The Floyd-Steinberg halftoning provides better results than random or ordered halftoning, because it makes not only local decisions, but the gathered information is also distributed in neighboring pixels. It means that it takes other samples into account as well, so the sample positions are stratified, making the resulting image smoother and reducing the noise compared to random or dithered approaches.

Due to its good properties and automatic stratification, we developed our sampler based on the Floyd-Steinberg method. Random dithering, which is similar to importance resampling, was implemented for comparison. We expected the same improvement in importance sampling as provided by the Floyd-Steinberg halftoning over random dithering.

The paper is organized as follows. In Section 2 we revisit the Floyd-Steinberg algorithm, extend it to arbitrary dimensions, and reformulate it to a sampling approach. Then a rendering application is presented in Section 3, which is environment mapping with shadow computation.

2. The Floyd-Steinberg sampler

Let us take the D -dimensional cube of the primary sample space and define an $R_1 \times R_2 \times \dots \times R_D$ resolution grid in it. The number of grid points is denoted by N . It is not necessary to make the same number of subdivisions along each axis. The grid is processed twice by iterating the subdivision points along the coordinate axes one-by-one. In the first run, scalar importance $I(c_1, c_2, \dots, c_D)$ is evaluated and stored as the initial importance of the grid point $\mathbf{u}_i = (c_1, c_2, \dots, c_D)$. From the sum of these values, the integral of the scalar importance function is estimated:

$$b = \int_{\mathcal{U}} I(\mathbf{u}) d\mathbf{u} \approx \frac{1}{N} \cdot \sum_{c_1=1}^{R_1} \dots \sum_{c_D=1}^{R_D} I(c_1, \dots, c_D).$$

The samples are generated in the second run. At grid point \mathbf{u}_i its scalar importance I is compared to a threshold value I_T .

- If the scalar importance is greater than the threshold, then the integrand is evaluated at sample \mathbf{u}_i and is included in the quadrature:

$$\Phi \approx \frac{1}{M} \sum_{i=1}^M \frac{F(\mathbf{z}(\mathbf{u}_i)) \cdot J(\mathbf{u}_i)}{p(\mathbf{u}_i)} = \frac{b}{M} \sum_{i=1}^M \frac{F(\mathbf{z}(\mathbf{u}_i)) \cdot J(\mathbf{u}_i)}{I(\mathbf{u}_i)}.$$

Simultaneously, the importance of the current grid point is decreased with I_T .

- If the scalar importance is not greater than the threshold, then no sample is generated here and the scalar importance of the grid point is left unchanged.

In both cases, before stepping onto the next grid point, the remaining importance of the current grid point is distributed to its unvisited neighbors. When we are at a grid point $\mathbf{u}_i = (c_1, c_2, \dots, c_D)$, the unvisited neighboring grid points have coordinates $(c_1 + \delta_1, c_2 + \delta_2, \dots, c_D + \delta_D)$, where δ_i may have values 0 or 1, while they cannot be all zeros simultaneously. Unvisited grid point $(c_1 + \delta_1, c_2 + \delta_2, \dots, c_D + \delta_D)$ receives $Iw(\delta_1, \delta_2, \dots, \delta_D)$ importance where the total sum of weights w is equal to 1. For example, in 2D the following weighting scheme may be used, which makes the weights roughly proportional to the associated points' Euclidean distances.

$$w(0, 1) = w(1, 0) = 3/8, \quad w(1, 1) = 2/8.$$

The number of samples M can be controlled by the used threshold I_T . As the total sum of the scalar importance values is Nb , the following correspondence can be established between them:

$$I_T = \frac{N \cdot b}{M}.$$

The number of original samples (N) and final samples (M) should be carefully selected since they affect both the computational cost and the accuracy. The total cost of sampling is proportional to N , which means that the method is practical only if the cost of importance function computation is much less than the cost of the integrand computation.

3. Application to environment mapping

In environment mapping [Deb98] we compute the direct lighting of a distant hemispherical light source. The radiance of point \vec{x} of a virtual object is the reflection of the illumination provided by the environment map, which can be computed as

$$L^r(\vec{x}, \vec{\omega}) = \int_{\Omega} L^{env}(\vec{\omega}') \cdot f_r(\vec{\omega}', \vec{x}, \vec{\omega}) \cdot \cos \theta' \cdot v(\vec{x}, \vec{\omega}') d\omega',$$

where Ω is the set of all directions, $L^{env}(\vec{\omega}')$ is the radiance of the environment map at direction $\vec{\omega}'$, f_r is the BRDF, and $v(\vec{x}, \vec{\omega}')$ is the indicator function checking whether no virtual object is seen from \vec{x} at direction $\vec{\omega}'$ (that is, the environment map can illuminate this point of the virtual object from the given direction).

Note that the integral is the product of three factors, the light intensity, the cosine weighted BRDF, and the visibility indicator:

$$F(\vec{\omega}) = L^{env}(\vec{\omega}') \cdot f_r(\vec{\omega}', \vec{x}, \vec{\omega}) \cdot \cos \theta' \cdot v(\vec{x}, \vec{\omega}').$$

Unfortunately, it is not possible to directly find a probability density that is proportional to this product, so classical techniques consider only a single factor. Unlike classical techniques, the proposed Floyd-Steinberg sampler can mimic this complex product as well.

As we need to evaluate the scalar importance function for every grid point, its computation must be much cheaper than that of the original integrand F . In environment mapping, the expensive part is visibility checking, that is the evaluation of function v . So in the scalar importance function the visibility is omitted. The illumination and the BRDF terms are wavelength dependent, so the scalar contribution should map these vectors to scalars. We may use the luminance of the product. Thus, the scalar importance function is:

$$I(\mathbf{u}) = \mathcal{L}(L^{env}(\vec{\omega}') \cdot f_r(\vec{\omega}', \vec{x}, \vec{\omega})) \cdot \cos \theta' \cdot J(\mathbf{u}),$$

where direction $\vec{\omega}'$ and angle θ' are functions of \mathbf{u} .

If we transform the unit rectangle onto the hemisphere by BRDF sampling, then the Jacobi determinant will compensate the angular variation of $f_r(\vec{\omega}', \vec{x}, \vec{\omega}) \cos \theta'$, making their product equal to the albedo $a(\vec{x}, \vec{\omega})$, thus the importance function will be proportional only to the remaining factors:

$$I(\mathbf{u}) = \mathcal{L}(L^{env}(\vec{\omega}'(\mathbf{u})) \cdot a(\vec{x}, \vec{\omega})).$$

The final form of the integral quadrature is:

$$\Phi \approx \frac{b}{M} \sum_{i=1}^M \frac{L^{env}(\vec{\omega}'(\mathbf{u}_i)) \cdot a(\vec{x}, \vec{\omega}) \cdot v(\vec{x}, \vec{\omega}'(\mathbf{u}_i))}{\mathcal{L}(L^{env}(\vec{\omega}'(\mathbf{u}_i)) \cdot a(\vec{x}, \vec{\omega}))}.$$

3.1. Results

We have compared three sampling techniques: BRDF sampling, importance resampling, and the Floyd-Steinberg sampler. All three were implemented as GPU algorithms, which

run on nVidia GeForce 8800 GFX graphics hardware. The test scenes consisted of diffuse objects illuminated by an environment map. All methods generated 32 directional samples per pixel. Both importance resampling and the Floyd-Steinberg sampler used 16×64 sample grids.

The top row of Figure 1 compares the images rendered with the three techniques when illuminated by a fairly smooth environment map. We can see that importance resampling does not improve image quality significantly, but the Floyd-Steinberg sampler has practically eliminated the noise which was not due to visibility. In the middle and bottom rows of Figure 1 we can see the same experiment with a high variation environment map. This scenario is particularly challenging for the BRDF sampler, as the high intensity range is sampled with a low probability. Importance resampling is already useful in this case, choosing the meaningful samples with a high probability, but some pepper-style noise remains. The Floyd-Steinberg sampler is successful at removing the noise, proving that it accomplishes practically perfect importance sampling. The only area where noise remains is where the bright part of the environment is occluded by shadowing geometry.

4. Discussion and conclusions

This paper presented a new importance sampling strategy that is based on the Floyd-Steinberg halftoning algorithm. Halftoning trades off spatial resolution to color resolution. In other words, it assumes that the image is relatively smooth. From the point of view of importance sampling, it means that the method is good if the scalar contribution function is relatively smooth. The power of the method comes from this property. When the decision is made whether or not we sample a particular value, not only this point is evaluated, but the information of the already generated samples is also taken account. Comparing to other sophisticated importance sampling methods, like *importance resampling* [BGH04], this is the key difference and also the main advantage in our method. Unlike in *hierarchical thresholding* [ARBJ03], we do not enforce an expensive explicit stratification step, but the algorithm delivers well distributed samples automatically.

Acknowledgement

This work has been supported by the National Scientific Research Fund (OTKA ref. No.: T042735) and by Hungarian and Bulgarian Academies of Sciences.

References

- [ARBJ03] S. Agarwal, R. Ramamoorthi, S. Belongie, and H. W. Jensen. Structured importance sampling of environment maps. *ACM Trans. Graph.*, 22(3):605–612, 2003.
- [BGH04] D. Burke, A. Ghosh, and W. Heidrich. Bidirectional importance sampling for illumination from environment maps. In *ACM SIGGRAPH 2004 Sketches*, page 112, 2004.

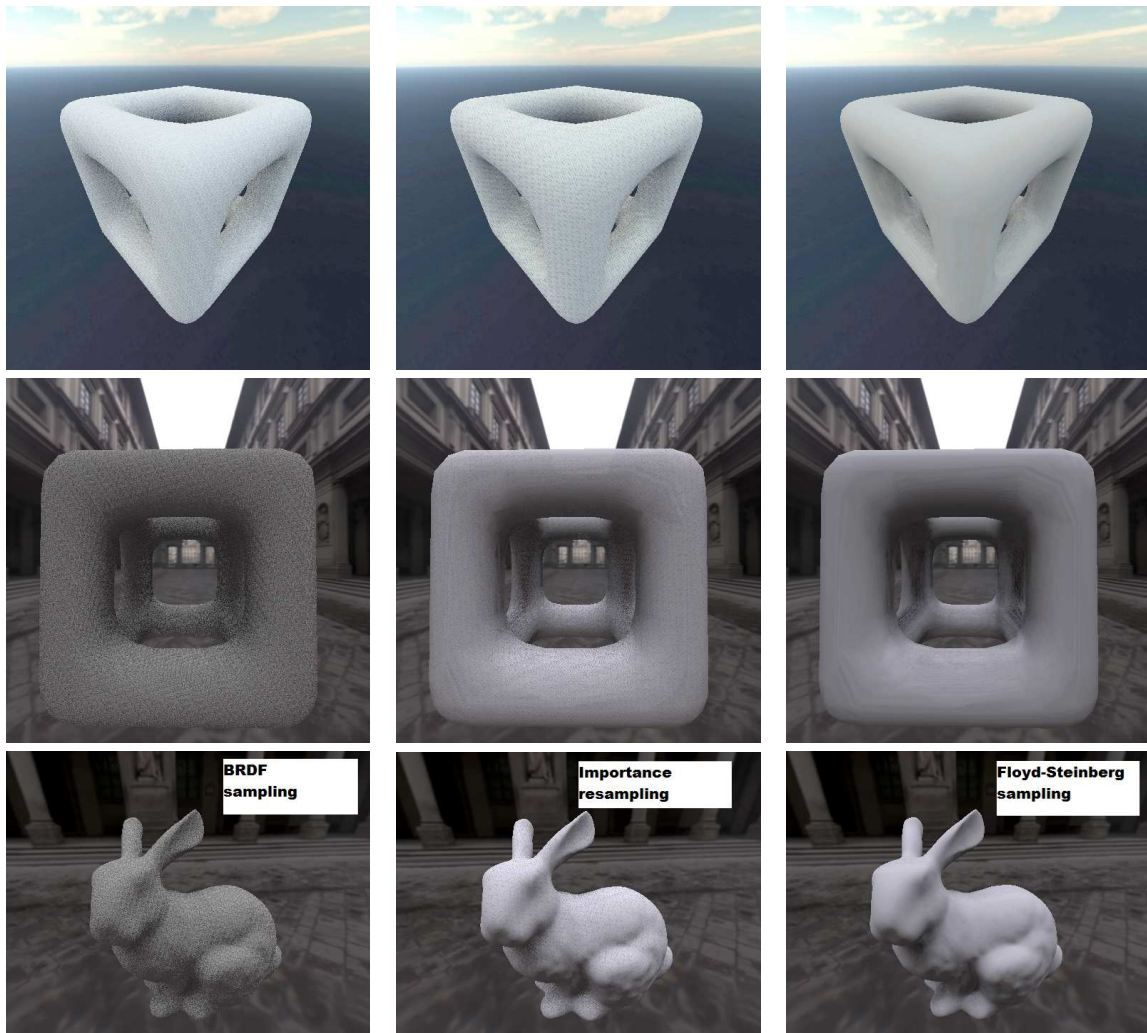


Figure 1: BRDF sampling (left, rendered at 17 sec), importance resampling (middle, rendered at 20 sec), and Floyd-Steinberg sampling (right, rendered at 20 sec) with a smooth (top row) and a high variance (middle and bottom rows) environment maps.

- [BGH05] D. Burke, A. Ghosh, and W. Heidrich. Bidirectional importance sampling for direct illumination. In *Rendering Techniques*, pages 147–156, 2005.
- [Deb98] P. Debevec. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *SIGGRAPH '98*, pages 189–198, 1998.
- [FS75] R. Floyd and L. Steinberg. An adaptive algorithm for spatial gray scale. In *Society for Information Display 1975 Symposium Digest of Technical Papers*, page 36, 1975.
- [KSKAC02] C. Kelemen, L. Szirmay-Kalos, G. Antal, and F. Csonka. A simple and robust mutation strategy for the Metropolis light transport algorithm. *Comput. Graph. Forum*, 21(3), pages 531–540, 2002.
- [ODJ04] V. Ostromoukhov, C. Donohue, and P.-M. Jodoin. Fast hierarchical importance sampling with blue noise properties. *ACM Transactions on Graphics*, 23(3):488–498, 2004.
- [RCL⁺08] F. Rousselle, P. Clarberg, L. Leblanc, V. Ostromoukhov, and P. Poulin. Efficient product sampling using hierarchical thresholding. In *Computer Graphics International 2008*, pages 465–474, June 2008.
- [TCE05] J. Talbot, D. Cline, and P. K. Egbert. Importance resampling for global illumination. In *Rendering Techniques*, pages 139–146, 2005.
- [VG97] E. Veach and L. Guibas. Metropolis light transport. *SIGGRAPH '97 Proceedings*, pages 65–76, 1997.
- [WWL05] L. Wan, T.T. Wong, and C.S. Leung. Spherical Q2-tree for sampling dynamic environment sequences. In *Eurographics Symposium on Rendering 2005 (EGSR 2005)*, pages 21–30, 2005.