A spectral analysis of function concatenations and its implications for sampling in direct volume visualization

Steven Bergner   Torsten Möller   Daniel Weiskopf[*]   David J. Muraki[†]

GrUVi Lab / Simon Fraser University   Department of Mathematics / Simon Fraser University
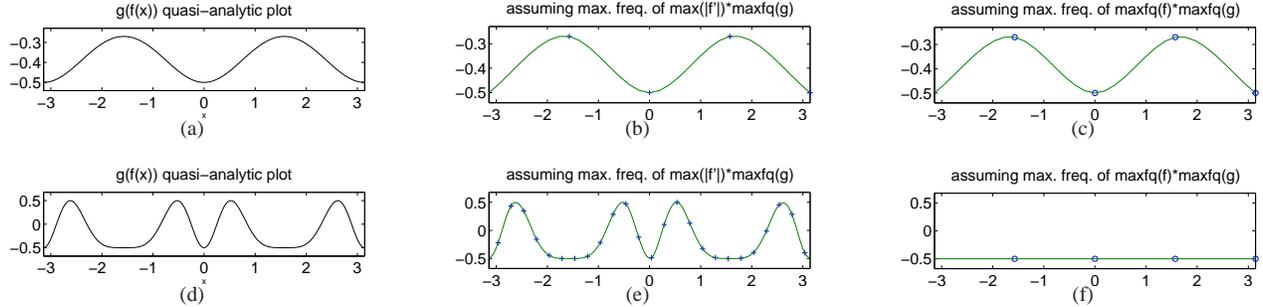
Figure 1: Images (a) and (d) show an analytic plot of $g(a_{1,2}f(x))$ using two different scalings of f. Illustrations (b) and (e) picture a spline-interpolated sampling $g(f(x))$ assuming a maximum frequency of $\max|f'| \max_{\text{frequency}}(g)$ according to derivation of this paper. For comparison, images (c) and (f) use the estimate $\pi \max_{\text{frequency}}(f) \max_{\text{frequency}}(g)$. Images (a)–(c) and (b)–(e) differ in the scaling of the magnitude of f. Notice that (f) is undersampled while (e) still truthfully reflects the composite function.

ABSTRACT

In this paper we investigate the effects of function composition in the form $g(f(x)) = h(x)$ by means of a spectral analysis of h. We decompose the spectral description of $h(x)$ into a scalar product of the spectral description of $g(x)$ and a term that solely depends on $f(x)$ and that is independent of $g(x)$. We then use the method of stationary phase to derive the Nyquist limit of $g(f(x))$. This limit is the product of the Nyquist limit of $g(x)$ and the maximum derivative of $f(x)$. This leads to a proper sampling of the composition h of the two functions g and f. We then apply our theoretical results to a fundamental open problem in volume rendering—the proper sampling of the rendering integral after the application of a transfer function. In particular, we demonstrate how the sampling criterion can be incorporated in adaptive ray integration, visualization with multi-dimensional transfer functions, and pre-integrated volume rendering.

CR Categories: I.4.5 [Image Processing and Computer Vision]: Reconstruction—Transform methods; I.3.3 [Computer Graphics]: Picture/Image Generation—Antialiasing

Keywords: volume rendering, transfer function, classification, signal processing, fourier transform, sampling

1 INTRODUCTION

The fundamental problem in image synthesis is the evaluation of the rendering integral [13]. In particular, volume rendering is based on the volume rendering integral, which requires the assignment or mapping of optical properties to given data values $f(x)$. By slightly abstracting the integrand within the volume rendering integral, this mapping can be viewed as a composite function $g(f(x)) = h(x)$, where g is the transfer function assigning opacities to values of the data f and where h is the resulting function that is to be rendered. It is the signal $g(f(x)) = (g \circ f)(x)$ that is the input to the rendering algorithm. Despite the chosen quadrature formula for evaluating the integral, a crucial parameter determining the accuracy of the numerical solution to the integral is the sampling distance. While it is common to use linear interpolation, it is important to use at least twice the Nyquist rate in order to guarantee an accurate evaluation of the integral.

Despite the common use of this approach it has not yet undergone a satisfactory mathematical analysis. In particular, there were no clear statements on how the mapped function is to be sampled appropriately.

It has been previously suggested [1, 6, 17] that the proper Nyquist limit of $(g \circ f)$ is proportional to the product of the respective Nyquist limits. However, we found this estimate too restricted for many data models, as demonstrated in Figure 1. The knowledge of the proper sampling rate of the function $(g \circ f)(x)$ will enable us to not only predict a proper error behavior, but to allow us to accelerate rendering algorithms by skipping over regions that need less sampling in order to guarantee a particular error behavior.

While solutions for quantized 8-bit data exist in form of pre-integrated transfer functions, adequate sampling of high dynamic range volumes and multi-modal or multi-dimensional data, such as $(f, |f'|)$, is yet unknown. Typical estimates are based on a proper sampling of f alone, which is neglecting the effect of the transfer function. In the following we will present an estimate for suitable sampling that takes the effect of the transfer function into account.

After a summary of the related work in Section 2 we will present a rigorous mathematical treatment in Section 3. Section 4 will then discuss the implications of our analysis for applications in volume rendering. We will also suggest a solution for multi-dimensional transfer functions. We summarize our contributions in Section 5 and give some directions for possible future explorations.

[*]e-mail: {sbergner,torsten,weiskopf}@cs.sfu.ca
[†]e-mail: muraki@math.sfu.ca

2 RELATED WORK

In this paper, we consider sampling in the context of volume rendering: what is the appropriate sampling rate for the combined function $(g \circ f)$, when f represents the scalar data and g the transfer function?

The task on how to properly evaluate the function under the rendering integral has been debated since the beginnings of volume graphics. Wittenbrink et al. [20] made the observation that it is important to interpolate f in order to properly super-sample $(g \circ f)$, while recently Younesy et al. [22] pointed out that it is important to low-pass filter g in order to sub-sample $(g \circ f)$.

To our knowledge, the work by Kraus [6] and Schulze and Kraus [17] is the only previous work that investigates the sampling of the volume rendering integral by means of Fourier analysis and the sampling theorem. For the function models they use in their derivations, the Nyquist frequency of $(g \circ f)$ is $\pi \nu_g \nu_f$, where ν_g and ν_f are the maximum frequencies in g and f, respectively. This statement is in accordance with a similar conjecture by Engel et al. [1].

2.1 Adaptive sampling

The main benefit of understanding the required sampling rate for the volume rendering integral is that the sampling rate can be adapted to the lowest possible value in order to reduce the computational load. Various approaches to adaptive sampling are known in the literature. A simple example is empty space skipping, which identifies regions of vanishing contribution to the integral and skips those regions, as for example in [5, 7, 11, 18, 21]. More advanced methods flexibly adapt the sampling rate to the requirements of volume rendering. For example, adaptive sampling can be employed for hierarchical splatting [9], GPU ray casting [15], or texture-based volume rendering [8].

2.2 Pre-integrated transfer functions

Pre-integrated volume rendering separates the computation of the volume rendering integral for a small ray segment from the sampling of the scalar field. Therefore, pre-integration is effective in reducing the required sampling rate. Pre-integration can be employed for various volume rendering algorithms. Its first uses were for the cell projection of tetrahedra [16] (with a simpler predecessor [19]) and 2D texture-based volume rendering of uniform grids [1]. One of the issues of pre-integration is the need to calculate and store large tables with pre-computed ray segments. Although there exist methods to accelerate this pre-computation [12, 14], the required computations and memory increase at least quadratically with the number of distinct scalar values. Therefore, pre-integration becomes less useful for data with high-resolution quantization, such as 12-bit CT scans or simulation data with floating-point accuracy. Today's trend to high dynamic range volume visualization [23] will increase the demand for appropriate volume rendering methods.

2.3 Multi-dimensional transfer functions

Another trend in volume rendering is the use of multi-dimensional transfer functions. Levoy [10] considers both the scalar value and its gradient magnitude to model a transfer function that extracts isosurface-like structures. Kindlmann [2] and Kniss et al. [3] extend this idea to include higher-order derivatives in transfer function design. Unfortunately, multi-dimensional transfer functions are difficult to use in combination with pre-integration: the larger number of parameters for the transfer function leads to impractically huge pre-integration tables. One solution is the on-the-fly computation of ray segments [4]. However, this approach is restricted to Gaussian transfer functions.

3 SPECTRAL ANALYSIS

In the subsequent analysis, the data is represented by $\mathbf{f}(\mathbf{x})$, which maps from $\mathbb{R}^3$ to $\mathbb{R}^m$, with m being the number of modalities. Our transfer function g maps $\mathbb{R}^m$ to a scalar value in $\mathbb{R}$, which could be one channel of the optical properties, such as opacity. The composite function is

$$h(\mathbf{x}) = g(\mathbf{f}(\mathbf{x})) \qquad (1)$$

Considering $G(\mathbf{l})$ to be the Fourier domain expansion of $g(\mathbf{y})$, $h(\mathbf{x})$ results from the inverse transform of G as

$$h(\mathbf{x}) = g(\mathbf{f}(\mathbf{x})) = \left(\frac{1}{\sqrt{2\pi}}\right)^m \int_{\mathbb{R}^m} G(\mathbf{l}) e^{i\mathbf{l}\cdot\mathbf{f}(\mathbf{x})} d\mathbf{l}. \qquad (2)$$

This is the inverse Fourier transform giving $g(\mathbf{y})$ for $\mathbf{y} = \mathbf{f}(\mathbf{x})$. The Fourier transform of $h(\mathbf{x})$ can be written as

$$H(\mathbf{k}) = \left(\frac{1}{\sqrt{2\pi}}\right)^{m+3} \int_{\mathbb{R}^3} \int_{\mathbb{R}^m} G(\mathbf{l}) e^{i\mathbf{l}\cdot\mathbf{f}(\mathbf{x})} d\mathbf{l}\, e^{-i\mathbf{k}\cdot\mathbf{x}} d\mathbf{x}. \qquad (3)$$

Switching the order of integration yields

$$H(\mathbf{k}) = \left(\frac{1}{\sqrt{2\pi}}\right)^{m+3} \int_{\mathbb{R}^m} G(\mathbf{l}) \int_{\mathbb{R}^3} e^{i\mathbf{l}\cdot\mathbf{f}(\mathbf{x})} e^{-i\mathbf{k}\cdot\mathbf{x}} d\mathbf{x} d\mathbf{l}. \qquad (4)$$

Noticing that the inner integral is independent of G, we give it its own name, $\mathbf{P}(\mathbf{k},\mathbf{l})$, and continue

$$\mathbf{P}(\mathbf{k},\mathbf{l}) = \int_{\mathbb{R}^3} e^{i(\mathbf{l}\cdot\mathbf{f}(\mathbf{x}) - \mathbf{k}\cdot\mathbf{x})} d\mathbf{x} \qquad (5)$$

$$H(\mathbf{k}) = \left(\frac{1}{\sqrt{2\pi}}\right)^{m+3} \int_{\mathbb{R}^m} G(\mathbf{l}) \mathbf{P}(\mathbf{k},\mathbf{l}) d\mathbf{l} \qquad (6)$$

$$H(\mathbf{k}) = \left(\frac{1}{\sqrt{2\pi}}\right)^3 < G(\cdot), \mathbf{P}(\mathbf{k},\cdot) >, \qquad (7)$$

This shows that forming the spectrum $H(\mathbf{k})$ of the composite function can be interpreted as a linear operation on the spectrum $G(\mathbf{l})$, implemented by means of the scalar product $< \cdot, \cdot >$. In the following we will take a closer look at the properties of $\mathbf{P}(\mathbf{k},\mathbf{l})$.

3.1 Visual inspection of the frequency map $\mathbf{P}(\mathbf{k},\mathbf{l})$

It is important to point out that $\mathbf{P}(\mathbf{k},\mathbf{l})$ is independent of the properties of g, which in our application is the transfer function, and solely depends on $\mathbf{f}(\mathbf{x})$. Further, in its role as a linear operator it can be interpreted as a map telling how a certain frequency component of G of wavenumber $\mathbf{l}$ is mapped to a frequency of index $\mathbf{k}$ in the target spectrum of H.

To get an intuition about the properties of this function we will first inspect it visually. Figure 2 shows $P(k,l)$ for different one-dimensional scalar functions $f(x)$. In particular we have chosen a pure Gaussian function and a combination of two Gaussian functions. We have computed $P(k,l)$ as the discrete Fourier transform of $e^{ilf(x)}$, which is a possible interpretation of Equation 5. The picture is horizontally periodic in k, which is due to the discretization and does not apply to the continuous case that we deal with in the subsequent analysis. Further, the function is point symmetric as $P(k,l) = \overline{P(-k,-l)}$, where $\overline{a}$ denotes the complex conjugate of a. This results from Equation 5.

A significant property apparent from Figure 2 is the non-marginal valued wedge in the middle, starting narrow at $l = k = 0$ and increasing in size towards larger k. According to Equation 7 the spectrum of the composite function $H(k)$ is formed by column-wise dot products with the spectrum $G(l)$. In order to determine at which maximum wavenumber k the function $H(k)$ has a significant contribution, we have to figure out for which k the main spectrum of $G(l)$ overlaps with the high-valued wedge to produce a non-negligible contribution. This determines the Nyquist rate of $H(k)$.
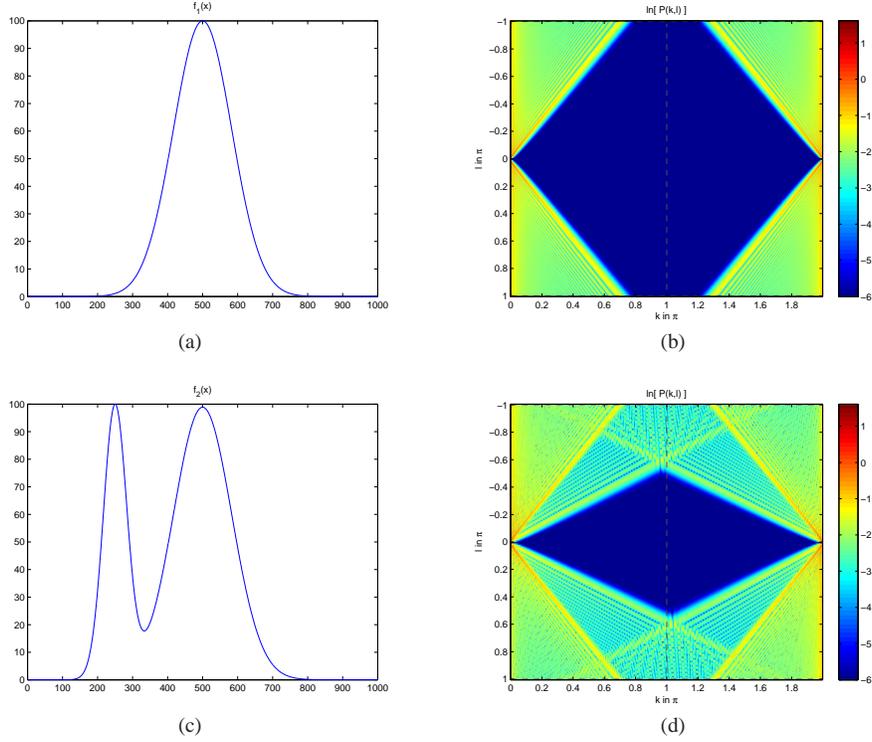
Figure 2: $P(k,l)$ for different functions $f(x)$. (a) single and (c) mixed Gaussians and their corresponding $P(k,l)$ in (b) and (d), respectively. The upper and lower slope of the low-valued wedge is given by the reciprocal of the minimum and the maximum value of f', respectively.

3.2 Analytic interpretation

In Equation 5 it is apparent that $\mathbf{P}(\mathbf{k},\mathbf{l})$ is an integral over an oscillating function $e^{iu(\mathbf{x})}$ with unit magnitude and phase $u(\mathbf{x}) = \mathbf{l} \cdot \mathbf{f}(\mathbf{x}) - \mathbf{k} \cdot \mathbf{x}$. For the following analysis we will restrict ourselves to the one-dimensional case. This is appropriate when performing the analysis along a single ray. Further we assume $f(x)$ to be a scalar-valued function.

As an introductory example, consider a linear function $f(x) = ax$. This simplifies $I = P(k,l) = \int_{-\infty}^{\infty} e^{i(la-k)x} dx = \delta(la - k)$, which is well known. If the phase is zero, the integral is infinite. However, if the phase is non-zero (changing constantly), the integral is zero. This behavior is well-known as Dirac's delta function.

For general functions $f(x)$ it can be said that the integral has significant cancellations in intervals where the phase $u(x) = lf(x) - kx$ is changing rapidly. The largest contributions occur where the phase of the integrand varies slowest, in particular where its derivative $u'(x_s) = 0$. An approximate solution for the integral can be obtained by only considering the neighborhood around x_s, which are the so-called *points of stationary phase*.

The previous statement only applies if the term $u(x)$ for the phase can be split up into the product of a large scalar and a function in the order $O(1)$. To facilitate this split, we change the parameterization of the integrand from $P(k,l)$ to polar coordinates $P(\kappa,\theta)$. Hence, the phase becomes $u(x) = \kappa(f(x)\sin\theta - x\cos\theta)$. The points x_s of stationary phase are then given by

$$\frac{du}{dx} = \frac{d}{dx}\kappa(f(x)\sin\theta - x\cos\theta) = 0 \tag{8}$$

$$f'(x_s)\sin\theta - \cos\theta = 0 \tag{9}$$

$$\frac{1}{f'(x_s)} = \tan\theta. \tag{10}$$

Around each x_s we replace the integrand by a second-order Taylor approximation resulting in[1]

$$I_{x_s} \sim \int_{-\infty}^{\infty} e^{i\kappa(f(x_s)\sin\theta - x_s\cos\theta + \frac{1}{2}f''(x_s)x^2\sin\theta)} dx \tag{11}$$

$$I_{x_s} \sim e^{i\kappa(f(x_s)\sin\theta - x_s\cos\theta)} \left(\frac{2\pi}{\kappa|f''(x_s)\sin\theta|}\right)^{1/2} e^{i\frac{\pi}{4}sgn\{f''(x_s)\sin\theta\}}. \tag{12}$$

For $f''(x_s)$ considerably different from zero the integrand vanishes quickly as $(x - x_s)^2$ increases. The full integral is obtained by summing all I_{x_s} for all x_s fulfilling Equation 10. This case is relevant for points $\min(f') < \frac{1}{\tan\theta} < \max(f')$. Outside this range we do not have any points of stationary phase and the overall integral forming $P(\kappa,\theta)$ is close to zero.

This observation establishes the main insight of our analysis: the extremal slopes of f form the boundary of the wedge observed in Figure 2. Therefore, the primary result of this paper is that the composite function has a maximum frequency of

$$v_h = v_g \max_x |f'(x)|, \tag{13}$$

where v_g is the maximum frequency of g. The corresponding sampling rate should be chosen just above the Nyquist rate $2v_h$.

An interesting case arises if one considers the boundaries of this interval. They form the boundaries of the wedge. To inspect the range around this band edge we define a critical angle θ_e fulfilling $\sin\theta_e f'(x_e) = \cos\theta_e$ and $f''(x_e) = 0$ with x_e being a maximum point of $f'(x)$. Here, the second derivative vanishes, which requires

---

[1]We do not need to consider $(x - x_s)$, because we can substitute x with $x = x' + x_s$ (and then rename x' back to x).

a third-order Taylor approximation of $u(x)$. In the vicinity of the band edge for $\theta \approx \theta_e$ the resulting integral is

$$P(\kappa, \theta) \sim \int_{-\infty}^{\infty} \exp[i\kappa(f(x_e)\sin\theta - x_e\cos\theta + (f'(x_e)\sin\theta - \cos\theta)x$$
$$+ \frac{1}{6}f'''(x_e)x^3\sin\theta)]dx \quad (14)$$

substituting $x = \alpha\bar{x}$ using $\alpha = \left(\frac{2}{\kappa f'''(x_e)\sin\theta}\right)^{1/3}$

$$= e^{i\kappa(f(x_e)\sin\theta - x_e\cos\theta)}\alpha$$
$$\cdot \int_{-\infty}^{\infty} \exp\left[i(\kappa\alpha(f'(x_e)\sin\theta - \cos\theta)\bar{x} + \frac{\bar{x}^3}{3}\right]d\bar{x} \quad (15)$$

considering $e^{is} + e^{-is} = 2\cos(s)$

$$= 2\pi e^{i\kappa(f(x_e)\sin\theta - x_e\cos\theta)}\alpha$$
$$\cdot \frac{1}{\pi}\int_0^{\infty} \cos\left[\alpha\kappa(f'(x_e)\sin\theta - \cos\theta)\bar{x} + \frac{\bar{x}^3}{3}\right]d\bar{x} \quad (16)$$

$$= 2\pi e^{i(f(x_e)\kappa\sin\theta - x_e\kappa\cos\theta)}\left(\frac{2}{f'''(x_e)\kappa\sin\theta}\right)^{1/3}$$
$$\cdot Ai\left(\kappa(f'(x_e)\sin\theta - \cos\theta)\left(\frac{2}{f'''(x_e)\kappa\sin\theta}\right)^{1/3}\right) \quad (17)$$

which has a solution involving the Airy function[2], whose graph is shown in Figure 3.
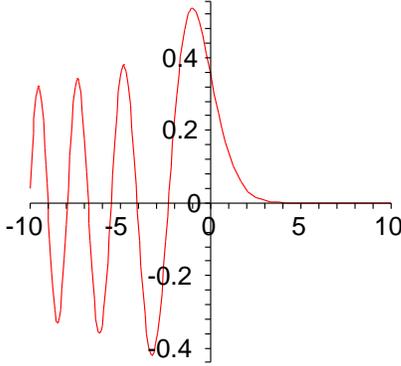
Figure 3: The graph of the Airy function $Ai(t)$. It decays algebraically toward positive t with $t^{-1/4}$. Also notice that its maximum for negative t. The value for $t = 0$ in Eq. 17 is attained at the band edge $\theta = \theta_e$.

3.3 Error analysis

The result in Eq. 17 gives us an idea of how $P(k,l)$ behaves near the band edge[3]. The first factor is a complex exponential changing in phase with k and l and is fixed in magnitude 2π. The second term is decaying in $O(l^{-1/3})$. The rapid decay of the third term is indicated in Figure 3 toward increasing κ. Important to note is that the main contributions from $Ai(t)$, including its maximum, occur for $t < 0$. That means by choosing our cutoff to be at the location,

[2]It is defined as $Ai(t) = \frac{1}{\pi}\int_0^{\infty}\cos(tx + \frac{x^3}{3})dx$ with $Ai(0) = 0.355028\ldots$
[3]For the interpretation recall that $l = \kappa\sin\theta$ and $k = \kappa\cos\theta$.

$\tan\theta = 1/\max|f'|$, we obtain an estimate for the band-limitedness of H.

3.4 Limits of the model

Since the above derivation is based on approximations, it is important to be aware of the limitations, arising from the assumptions made to facilitate the analysis. The most important one is that the method of stationary phase is only applicable if the phase is amplified by a large constant. In our case this means that the derivation does not necessarily hold for small κ. This is a reasonable assumption as long as we consider (k,l) not too close to $(0,0)$, which is the case for the band-limits $v_g = l_{max}$ of practical transfer functions or $v_h = k_{max}$ for typical rendered volumes h.

3.5 Extension to multi-modal data and multi-dimensional transfer functions

In the case of multi-modal transfer functions applied to 3D data, our function $\mathbf{f}(\mathbf{x})$ would be a mapping from $\mathbb{R}^3$ to $\mathbb{R}^m$ with $m > 1$. Hence, we cannot simplify the multi-dimensional description of $\mathbf{P}(\mathbf{k},\mathbf{l})$. We can, however, assume that our analysis is along one ray (which we will use for ray-tracing) and assume a mapping from $\mathbb{R}$ to $\mathbb{R}^m$.

In this case, the Taylor series expansion of $u(x) = \mathbf{l} \cdot \mathbf{f}(x) - kx$ will simply be a dot product of the Taylor series of each component. I.e.

$$u'(x) = \mathbf{l} \cdot \mathbf{f}'(x) - k \quad (18)$$
$$= l_1 f_1'(x) + l_2 f_2'(x) + \ldots + l_m f_m'(x) - k \quad (19)$$
$$= 0. \quad (20)$$

This is indeed a description of an m-dimensional line in the space $(\mathbf{l}, k)$ which starts in the origin. Given a particular direction $\mathbf{l}$, we are interested in the maximum $\max(\mathbf{l} \cdot \mathbf{f}'(x))$ over all x. This will again be the border of our $(m+1)$-dimensional wedge, where the Airy function takes over.

Looking at just one particular direction $\mathbf{l}$ this problem is analogous to our 1D problem in the space spanned by the k-axis and the $\mathbf{l}$ axis. Let us assume, that the maximum frequency of our transfer function in this particular direction is denoted by $v_\mathbf{l}$. In analogy to our 1D treatment we would look for minimal k for which the length of $\mathbf{l}$ is equal to that particular maximum frequency. I.e. $v_\mathbf{l} \max(\mathbf{l} \cdot \mathbf{f}'(x)/||\mathbf{l}||) = k$. Naturally we are looking at all possible directions $\mathbf{l}$ and will have to pick the maximum k, since this will be the maximum frequency of our $\mathbf{P}(\mathbf{k}, \mathbf{l})$, denoted by v_{Pkl}. Therefore, we get

$$v_{Pkl} = \max_{|\mathbf{l}|=1}(v_\mathbf{l}\max_x(\mathbf{l} \cdot \mathbf{f}'(x))) \quad (21)$$

3.6 Relationship to $v_d \cdot v_g$ bounding frequency

A previous analysis [6, 17] suggests that the maximum frequency to be expected in transfer function composition is given by $\pi v_f v_g$, multiplying the two band-limiting frequencies, of the data and the transfer function, respectively. Using Carson's rule the statement was derived that over 98% of the energy are preserved within this cutoff frequency.

Considering the example in Figure 1 where $f(x) = a\sin(v_c x)$, the statement really only holds for $a = 1$. The result for other values of a could be obtained by extending the previous derivation [6, 17]. However, their previous discussion has to assume some kind of sinusoidal modeling of the input function—a restriction that does not occur in our derivation.

3.7 Other features of f(x) to be found in P(k,l)

There are a few more properties of $\mathbf{P}(\mathbf{k},\mathbf{l})$ that we have come across during our analysis, that we thought worthwhile to be mentioned here.

We can extract the histogram of $\mathbf{f}(\mathbf{x})$. Using a transfer function $g(\mathbf{y}) = \delta(\mathbf{y} - \mathbf{y}_0)$ with frequency response $G_{\mathbf{y}_0}(\mathbf{l}) = e^{i\mathbf{l}\mathbf{y}_0}$. The histogram is given by counting how often the value $\mathbf{y}_0$ occurs in $\mathbf{f}(\mathbf{x})$, which amounts to the DC value $H_{\mathbf{y}_0}(\mathbf{0})$.

Another observation comes up when applying sifting[4] to Eq. 5 yielding

$$\mathbf{P}(\mathbf{k},\mathbf{l}) = \int_{\mathbb{R}^3} \int_{\mathbb{R}} \delta\left(\mathbf{l}\cdot\mathbf{f}(\mathbf{x}) - \mathbf{k}\cdot\mathbf{x} - y\right) e^{iy} dy d\mathbf{x} \qquad (22)$$

$$\mathbf{P}(\mathbf{k},\mathbf{l}) = \int_{\mathbb{R}} \int_{\mathbb{R}^3} \delta\left(\mathbf{l}\cdot\mathbf{f}(\mathbf{x}) - \mathbf{k}\cdot\mathbf{x} - y\right) d\mathbf{x} e^{iy} dy \qquad (23)$$

$$\mathcal{H}\{u(\mathbf{x})\}(y) = \int_{\mathbb{R}^3} \delta\left(\mathbf{l}\cdot\mathbf{f}(\mathbf{x}) - \mathbf{k}\cdot\mathbf{x} - y\right) d\mathbf{x} \qquad (24)$$

$$\mathbf{P}(\mathbf{k},\mathbf{l}) = \int_{-\infty}^{\infty} \mathcal{H}_{u(\mathbf{x})}(y) e^{iy} dy, \qquad (25)$$

where $\mathcal{H}_{u(\mathbf{x})}(y)$ can be regarded as the analytic histogram of the phase $u(\mathbf{x})$.

4 APPLICATION TO VOLUME RENDERING

In the following we are going to lay out implications the above theory has when applied in volume visualization.

4.1 Adaptive sampling

A direct application is to use the maximum frequency of $(g \circ f)$ in order to determine the sampling rate for the volume rendering integral. Here, the maximum value of f' is computed in the whole volume to calculate a fixed, overall sampling rate. Unfortunately, the (possibly small) region of the data set with the maximum of f' would solely determine the sampling, even if the data set were slowly changing in most parts. A better solution is adaptive sampling: the rate is chosen space-variant to reflect the local behavior of the data set.

The space-variant step size can be determined by identifying the maximum value of f' in a finite neighborhood around the current sampling point. In other words, the discussion from Section 3 is applied only to a window of the full domain of f. The step size in this window region is equal or greater than the step size for a global treatment. Therefore, we typically obtain less sample points, without degrading the sampling quality.

There are numerous previous papers on adaptive volume rendering, a few of which are mentioned in Section 2. Most of the adaptive approaches need some kind of data structure that controls the space-variant steps size. Our approach also follows this strategy. The distinctive feature of our approach is not the fact that an adaptive step size is used, but that we provide a mathematically based criterion for choosing the step size. In fact, most of the existing adaptive rendering methods could be enriched by this criterion.

Our implementation consists of the following parts. First, a volume of gradient magnitudes is computed for the scalar data set. Second, the gradient-magnitude volume is filtered using a rank-order filter that picks out the maximum in a given neighborhood around a grid point. The size of the neighborhood is user-defined; its shape is a cube (based on a tensor-product filter). The size of the neighborhood is a 3D version of the ray-oriented window size that is used to derive the step size criterion. By using the maximum gradient magnitude in a 3D neighborhood, the isotropic step

---

[4]Sifting refers to the property $r(s) = \int_{-\infty}^{\infty} \delta(s-t) r(t) dt$.

---

size is chosen conservatively in this neighborhood. The third step is the actual volume rendering. We currently use a CPU ray caster that selects the sampling distance at a point based on the filtered gradient-magnitude volume. The maximum step size is clamped to the size of the neighborhood to avoid sampling artifacts that may arrive through the construction of the gradient-magnitude volume. If the sampling rate were to exceed a certain user-defined threshold (e.g., a hundred times the frequency of the data grid), it will be artificially clamped to that threshold value to avoid excessive sampling.

Note that, for a fixed transfer function, steps one and two of the above pipeline are pre-processing steps that do not have to be re-computed during rendering. To speed up the change of transfer function, additional acceleration data structures should be considered. For example, ideas for the efficient computation of space-leaping (see [7]) could be explored.

4.2 Relationship to pre-integration

The rationale for pre-integration is to separate the influences of the transfer function and of the scalar data field on the sampling rate of the complete volume rendering integral [1]. The separating is achieved by pre-computing the contributions of small ray segments to the rendering integral. Typically, a linear interpolation of scalar values is assumed within a ray segment. In this way, the pre-integration table absorbs the effects of the transfer function, while the actual volume rendering process only needs to reconstruct the scalar field faithfully.

Our derivation of the Nyquist rate for sampling $(g \circ f)$ is another support for the usefulness of pre-integration. We have shown that the Nyquist frequency for the volume rendering integral is proportional to the Nyquist frequency of the transfer function. Therefore, pre-integration is especially useful for high-frequency transfer functions (see, e.g., the extreme case of a random transfer function [1]).

In fact, we would like to demonstrate how the computation of pre-integration tables can be related to our description of sampling rates. The volume rendering integral can be expressed as

$$\int_{x_0}^{x_0+L} c(f(x)) e^{-\int_x^{x_0+L} \tau(f(\tilde{x})) d\tilde{x}} dx, \qquad (26)$$

with emission c, extinction coefficient τ, the start point x_0, and the segment length L. The scalar field f is assumed to be linear within a single ray segment. If f_0 and f_1 are the scalar values at the start and end points of that segment, the corresponding first derivative is constant in the segment: $f'(x) = (f_1 - f_0)/L$. Therefore, the number of sample steps should be given by Eq. 13. Equation 26 can be re-written as an integral in the scalar-value domain by using a substitution of variables:

$$\int_{f_0}^{f_1} c(f) e^{-\int_f^{f_1} \tau(\tilde{f}) \frac{L}{f_1-f_0} d\tilde{f}} \frac{L}{f_1 - f_0} df. \qquad (27)$$

In this domain, the sampling distance is determined by Eq. 13 and depends only on the characteristics of the transfer function, not on f'. As expected, the generation of pre-integration tables is independent of the behavior of the scalar field. In fact, it has been common practice to compute 2D pre-integration tables with a constant step size in the scalar-value domain (e.g., for the original version [1] and the subrange integration approach [12]). For an accurate computation, the frequency of the transfer function should be taken into account to determine the integration step size.

<center>(a)   (b)</center>
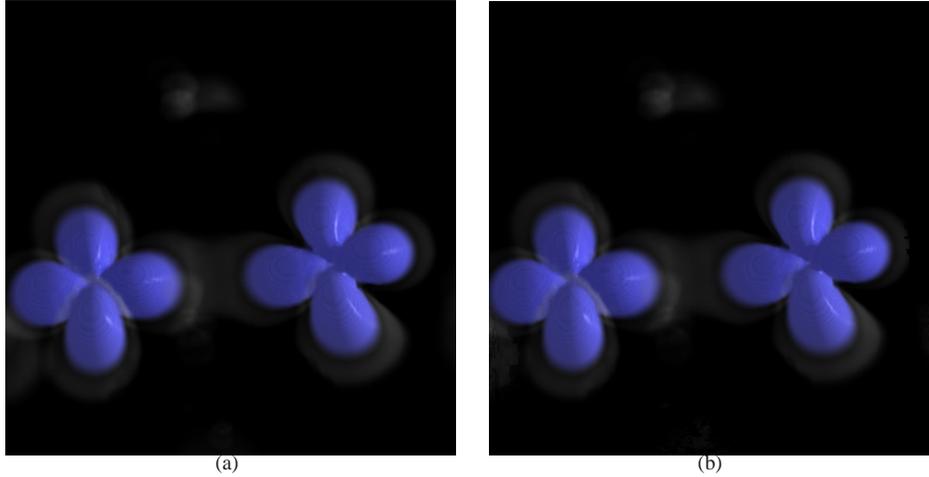
Figure 4: Examples of hipiph data set (a,c) sampled at a fixed rate (0.5) and (b,d) sampled with adaptive stepping. The adaptive method in (b) is using about 25% less samples than (a) only measuring in areas of non-zero opacity to not account for effects of empty-space skipping, which would skew the advantage even more towards the adaptive approach.

The main issue of pre-integrated volume rendering is are the computational and memory costs for generating and storing the pre-integration tables. Even with accelerated pre-computation [12], the computational and memory requirements for a 2D table increase quadratically with the number of distinct scalar values. Therefore, pre-integration is not suitable for data sets with finely quantized scalar values. An extreme case is floating-point scalar data, which can for example be visualized by high dynamic rending volume rendering [23]. Another problem is caused by the increase of parameters in multi-dimensional transfer functions, which makes the use of pre-integration prohibitive in these applications. In general, the problem can be viewed as an imbalance between the number of pre-computed ray segments and the actually used segment information during volume rendering: for high-resolution data, most of the ray segments are pre-computed without using them for volume rendering. Therefore, we see an increasing demand for direct sampling of the full volume rendering integral without any pre-integration. Here, adaptive on-the-fly sampling will be revived as a most important acceleration mechanism (see Section 4.1).

5   Discussion and Outlook

This paper closes a gap in the understanding and accurate estimation of the volume rendering integral. Namely it closes perhaps the most important theoretical gap still existing – the proper sampling rate to be used during the rendering step. Hence, the main contribution of this paper is an analysis of the frequency behavior applying a transfer function to spatial data. The resulting rule is that the band-limiting frequency of the composite function $h(x) = g(f(x))$ is given by $v_h = v_g \max_x |f'(x)|$. This is not a strict band-limit, but frequency components decay algebraically beyond v_h.

Further we demonstrated an extension to multi-dimensional transfer functions. In this case a similar band-limit is computed by $v_{Pkl} = \max_{|\mathbf{l}|=1}(v_{\mathbf{l}} \max_x(\mathbf{l} \cdot \mathbf{f}'(x)))$. This is again very simple to compute in a preprocessing step.

The treatment in this paper is independent of the application and hence can be applied in other fields of signal processing and applied mathematics. However, the focus of our application has been rendering. In addition to the theoretical findings we have applied the result to a method for adaptive sampling based on the maximum gradient magnitude. We were able to apply our theoretical re-sults for an adaptive rendering algorithms, which achieved the same quality in the rendered images by reducing the number of samples needed drastically.

5.1   Future work

The above analysis and discussion has considered $g(f(x))$ from the perspective of applying a transfer function to given data. Another interesting interpretation is to view $\mathbf{f}$ as a change in parameterization of g, where $\mathbf{P}(\mathbf{k},\mathbf{l})$ reflects the change in the spectrum of g. This might be of use when investigating the effect of a change in parameterization of a function defined on a surface.

A further application is photo-realistic Fourier-domain rendering. Knowing the Fourier domain decomposition of f and g we can now apply the slicing theorem to $\mathbf{P}(\mathbf{k},\mathbf{l})$ and only have to recompute G whenever the user changes the transfer function. This is a rather fast computation.

6   Acknowledgments

References

[1] K. Engel, M. Kraus, and T. Ertl. High-quality pre-integrated volume rendering using hardware-accelerated pixel shading. In *Eurographics / SIGGRAPH Workshop on Graphics Hardware 2001*, pages 9–16, 2001.

[2] G. Kindlmann and J. W. Durkin. Semi-automatic generation of transfer functions for direct volume rendering. In *Proc. Symposium on Volume Visualization*, pages 79–86, 1998.

[3] J. Kniss, G. Kindlmann, and C. Hansen. Interactive volume rendering using multi-dimensional transfer functions and direct manipulation widgets. In *Proc. IEEE Visualization 2001*, pages 255–262, 2001.

[4] J. Kniss, S. Premoze, M. Ikits, A. E. Lefohn, C. Hansen, and E. Praun. Gaussian transfer functions for multi-field volume visualization. In *Proc. IEEE Visualization 2003*, pages 497–504, 2003.

[5] G. Knittel. The UltraVis system. In *Proc. Symposium on Volume Visualization*, pages 71–79, 2000.

[6] M. Kraus. *Direct Volume Visualization of Geometrically Unpleasant Meshes*. PhD thesis, University of Stuttgart, 2003.

[7] J. Krüger and R. Westermann. Acceleration techniques for GPU-based volume rendering. In *Proc. IEEE Visualization 2003*, pages 287–292, 2003.

[8] E. LaMar, B. Hamann, and K. I. Joy. Multiresolution techniques for interactive texture-based volume visualization. In *Proc. IEEE Visualization 1999*, pages 355–361, 1999.

[9] D. Laur and P. Hanrahan. Hierarchical splatting: A progressive refinement algorithm for volume rendering. *Computer Graphics (Proc. ACM SIGGRAPH '91)*, 25(4):285–288, 1991.

[10] M. Levoy. Display of surfaces from volume data. *IEEE Computer Graphics and Applications*, 8(3):29–37, 1988.

[11] W. Li, K. Mueller, and A. Kaufman. Empty space skipping and occlusion clipping for texture-based volume rendering. In *Proc. IEEE Visualization 2003*, pages 317–324, 2003.

[12] E. B. Lum, B. Wilson, and K.-L. Ma. High-quality lighting for pre-integrated volume rendering. In *Proc. EG/IEEE TCVG Symposium on Visualization VisSym 2004*, pages 25–34, 2004.

[13] M. Pharr and G. Humphreys. *Physically Based Rendering: From Theory to Implementation*. Morgan Kaufmann, August 2004.

[14] S. Roettger and T. Ertl. A two-step approach for interactive pre-integrated volume rendering of unstructured grids. In *Proc. Symposium on Volume Visualization 2002*, pages 23–28, 2002.

[15] S. Roettger, S. Guthe, D. Weiskopf, and T. Ertl. Smart hardware-accelerated volume rendering. In *Proc. EG/IEEE TCVG Symposium on Visualization VisSym 2003*, pages 231–238, 2003.

[16] S. Roettger, M. Kraus, and T. Ertl. Hardware-accelerated volume and isosurface rendering based on cell-projection. In *Proc. IEEE Visualization 2000*, pages 109–116, 2000.

[17] J. P. Schulze, M. Kraus, U. Lang, and T. Ertl. Integrating pre-integration into the shear-warp algorithm. In *Proc. Eurographics/IEEE TCVG Workshop on Volume Graphics*, pages 109–118, 2003.

[18] M. Sramek and A. Kaufman. Fast ray-tracing of rectilinear volume data using distance transforms. *IEEE Transactions on Visualization and Computer Graphics*, 6(3):236–252, 2000.

[19] P. L. Williams, N. L. Max, and C. M. Stein. A high accuracy volume renderer for unstructured data. *IEEE Transactions on Visualization and Computer Graphics*, 4(1):37–54, 1998.

[20] C. M. Wittenbrink, T. Malzbender, and M. E. Goss. Opacity-weighted color interpolation for volume sampling. In *Proc. Symposium on Volume Visualization 1998*, pages 135–142, 1998.

[21] R. Yagel and Z. Shi. Accelerating volume animation by space-leaping. In *Proc. IEEE Visualization 1993*, pages 62–69, 1993.

[22] H. Younesy, T. Möller, and H. Carr. Improving the quality of multi-resolution volume rendering. In *Proc. EuroVis 2006*, 2006.

[23] X. Yuan, M. X. Nguyen, B. Chen, and D. H. Porter. High dynamic range volume visualization. In *Proc. IEEE Visualization 2005*, pages 327–334, 2005.