Procedural Phasor Noise

Thibault Tricard, Semyon Efremov, Cédric Zanni, Fabrice Neyret, Jonàs Martínez, Sylvain Lefebvre

▶ **To cite this version:**

Thibault Tricard, Semyon Efremov, Cédric Zanni, Fabrice Neyret, Jonàs Martínez, et al.. Procedural Phasor Noise. ACM Transactions on Graphics, 2019, 38 (4), pp.Article No. 57:1-13. 10.1145/3306346.3322990 . hal-02118508

HAL Id: hal-02118508
https://hal.science/hal-02118508

Submitted on 3 May 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Procedural Phasor Noise

THIBAULT TRICARD* and SEMYON EFREMOV*, Université de Lorraine, CNRS, Inria, LORIA
CÉDRIC ZANNI, Université de Lorraine, CNRS, Inria, LORIA
FABRICE NEYRET, CNRS, Université Grenoble Alpes, Grenoble INP, LJK, INRIA
JONÀS MARTÍNEZ, Université de Lorraine, CNRS, Inria, LORIA
SYLVAIN LEFEBVRE, Université de Lorraine, CNRS, Inria, LORIA

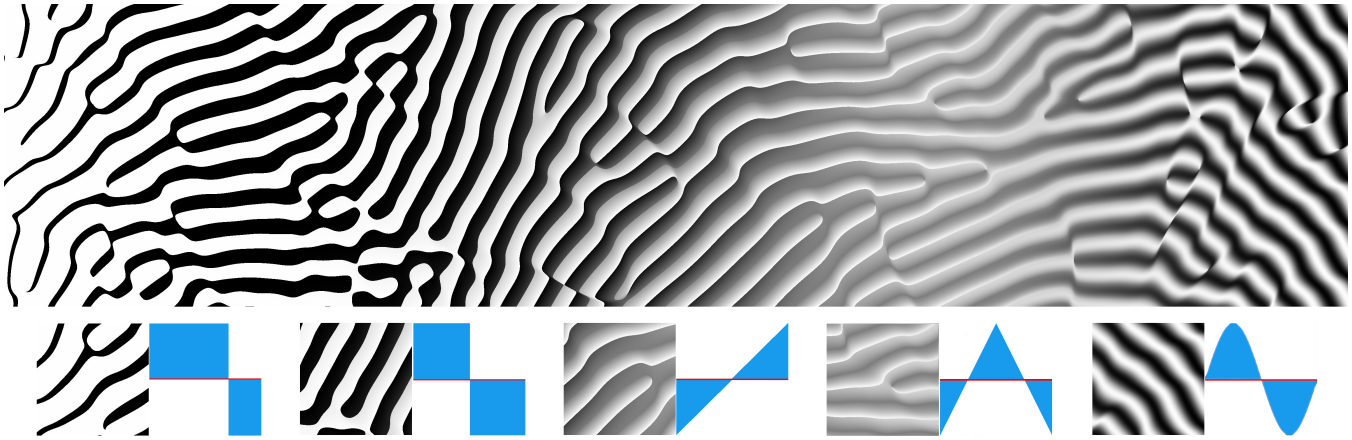


Fig. 1. High-contrast patterns produced by our approach. Note how the profile of the oscillations smoothly transition from a rectangular wave (20% black), to a square wave, to a triangular profile and finally a sine wave. At the same time, the orientation of the waves changes from left to right. The field visualized here is purely procedural. It is obtained by feeding our phasor noise into periodic profile functions (shown in blue), that are interpolated from left to right.

Procedural pattern synthesis is a fundamental tool of Computer Graphics, ubiquitous in games and special effects. By calling a single procedure in every pixel – or voxel – large quantities of details are generated at low cost, enhancing textures, producing complex structures within and along surfaces. Such procedures are typically implemented as pixel shaders.

We propose a novel procedural pattern synthesis technique that exhibits desirable properties for modeling highly contrasted patterns, that are especially well suited to produce surface and microstructure details. In particular, our synthesizer affords for a precise control over the profile, orientation and distribution of the produced stochastic patterns, while allowing to grade all these parameters spatially.

Our technique defines a stochastic smooth phase field – *a phasor noise* – that is then fed into a periodic function (e.g. a sine wave), producing an oscillating field with prescribed main frequencies and preserved contrast oscillations. In addition, the profile of each oscillation is directly controllable (e.g. sine wave, sawtooth, rectangular or any 1D profile). Our technique builds upon a reformulation of Gabor noise in terms of a *phasor field* that affords for a clear separation between local intensity and phase.

Applications range from texturing to modeling surface displacements, as well as multi-material microstructures in the context of additive manufacturing.

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

Additional Key Words and Phrases: procedural, textures, noise, Gabor, pattern, texture synthesis

ACM Reference Format:
Thibault Tricard, Semyon Efremov, Cédric Zanni, Fabrice Neyret, Jonàs Martínez, and Sylvain Lefebvre. 2019. Procedural Phasor Noise. *ACM Trans. Graph.* 38, 4, Article 57 (July 2019), 13 pages. https://doi.org/10.1145/3306346.3322990

1 INTRODUCTION

The Computer Graphics community is in a perpetual quest to extend the range of patterns that can be generated at low memory and computational costs, from a single procedure. Most techniques rely on *procedural noises* [Lagae et al. 2010], which generate scalar fields with prescribed frequency content. These base noises are then combined through various functions to produce interesting patterns [Ebert et al. 2003].

A key limitation of standard procedural noises is the lack of direct control over the local characteristics of the patterns, such as contrast,

*Joint first authors

Authors' addresses: Thibault Tricard; Semyon Efremov, Université de Lorraine, CNRS, Inria, LORIA, Nancy, France, F-54000; Cédric Zanni, Université de Lorraine, CNRS, Inria, LORIA, Nancy, France, F-54000; Fabrice Neyret, CNRS, Université Grenoble Alpes, Grenoble INP, LJK, INRIA; Jonàs Martínez, Université de Lorraine, CNRS, Inria, LORIA, Nancy, France, F-54000; Sylvain Lefebvre, Université de Lorraine, CNRS, Inria, LORIA, Nancy, France, F-54000.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2019 Copyright held by the owner/author(s). Publication rights licensed to ACM.
0730-0301/2019/7-ART57 $15.00
https://doi.org/10.1145/3306346.3322990

sharpness, as well as the range and histogram of values within a neighborhood. A degree of control is obtained though thresholding a base noise, or through lookup tables (color maps) reshaping the global histogram. Unfortunately, noise patterns generally exhibit variations such as local loss of contrast and undesired variations in scales and orientations. These variations make it very difficult to obtain an overall consistent appearance through lookup tables, and it is rarely possible to find parameters producing a consistent, stable aspect, such as the waves shown in Figure 1. Lookup tables also cannot distinguish between the 'sides' of an oscillation, preventing the synthesis of patterns with asymmetric aspects, such as the sawtooth wave in Figure 1, middle.

In this paper we propose a procedural synthesizer that has the ability to generate patterns exhibiting precisely controllable oscillations, in terms of their profile shape, minimum/maximum values, orientation and frequencies (scale). The synthesizer can freely grade the pattern parameters, within planes (2D) and volumes (3D). Our noise also works along surfaces – see *surface noise* in [Lagae et al. 2009] – which affords for the synthesis of patterns along objects, changing their appearance and surface roughness profiles. We explore the various possibilities in Section 4. Thanks to its procedural nature, our approach is implemented as an efficient OpenGL shader.

Contrary to most existing techniques, our procedure does not directly generate a scalar field of values. Instead it generates a phase field designed to be subsequently modulated through a periodic function. This phase field – our *phasor noise* – is such that, once seen through the periodic function, it produces a pattern with the specified frequency and orientation. In addition, the shape of the oscillation profile remains that of the periodic function without any undesirable fluctuations. This affords for a precise control over the shape of the produced stochastic structures, illustrated in Figure 1.

We achieve this by reformulating Gabor noise [Lagae et al. 2009] as the explicit product between an intensity field, and a modulated sine wave. The phasor noise is obtained as the instantenous phase of the Gabor oscillations. Its name originates from our reformulation, which exposes a sum of phasors[1] within Gabor noise.

Feeding a phasor noise into a sine wave produces a signal that ressembles Gabor noise, has a similar spectrum but oscillates with a corrected, uniform local contrast (see Section 3). This can serve as a fundamental harmonic to reproduce any 1D periodic profile by an approach akin to inverse Fourier transform (i.e. additive synthesis). Profiles are synthesized as weighted sums of integer harmonics of the base pattern, see Figure 13.

Being able to orient synthesized patterns while controlling their profile and distribution is especially well suited for the definition of patterns such as hatches, stripes, cracks, ridges, scales and ripples. Our technique also finds applications in modeling for additive manufacturing, for sculpting grip patterns or for precise control of proportions in multi-material mixtures.

Contributions. In summary, our contributions are:

- The definition of phasor noises, enabling pattern synthesizers with precisely controlled oscillation's profiles.
- An analytical separation of Gabor noise into an intensity and a modulated sine wave component.
- A numerical analysis of the resulting signals, in terms of their spectrum and spectrum of variance.
- A complete implementation with available source code.

[1] See e.g. https://en.wikipedia.org/wiki/Phasor.

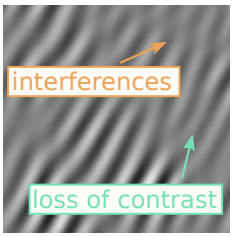
2 PREVIOUS WORK

Procedural noises [Lagae et al. 2010] have a long history in Computer Graphics and are widely used in applications ranging from games to movie special effects. They allow for arbitrary amounts of content to be synthesized at low memory and computational costs. Procedural noise methods focus on producing seemingly random patterns in the spatial domain, while providing an accurate control over their frequency content. A variety of 2D patterns – textures – are obtained by combining layers of noises having different spectral contents [Ebert et al. 2003].

One of the most recent development in this area are techniques based on the so-called Gabor noise [Lagae et al. 2009]. A Gabor noise is an instance of a sparse convolution noise [Lewis 1984, 1986], where a kernel is convolved in the spatial domain with a random impulse noise. An important property is that the spectrum of the obtained noise is that of the kernel. Thus, precisely controlling the kernel spectral content translates to a similar control over the noise. The Gabor kernel offers such a precise, elementary control.

Several improvements and variants of Gabor noise have been proposed, for isotropic noises [Lagae et al. 2011], filtering of solid (3D) noises [Lagae and Drettakis 2011], fitting Gabor noises to examples [Galerne et al. 2012; Gilet et al. 2010], as well as modifying the underlying formulation for faster synthesis [Tavernier et al. 2019] and explicit manipulation of phase [Gilet et al. 2014].

The original Gabor noise [Lagae et al. 2009] supports two important features in our context. First, the ability to spatially grade the noise content, freely changing frequencies and orientations. Second, the basic formulation can be applied in 2D and 3D, but also along any surface equipped with a tangent field. This is achieved without the need for a global parameterization (texture coordinates). Preserving these properties while manipulating phase is especially difficult. For instance, the formulation by Gilet et al. [2014] localizes kernels at the nodes of a grid to synchronize their phases. While this provides excellent results in the context of 2D texture synthesis, the phase synchronization cannot be preserved through an orientable field or along a surface.



When it comes to synthesizing highly contrasted patterns, Gabor noise suffers from two interrelated defects: local loss of contrast and interferences, as illustrated in the Figure inset. To our knowledge, these have been identified and studied for the first time by Neyret and Heitz [2016], who provide deep insights regarding these defects and focus on analyzing and filtering them. In particular, they show that the *spectrum of variance* of the noise – the Fourier transform of the squared signal – characterizes the local loss of contrast. We follow their insight and rely on the spectrum of variance to analyze the properties of our noise. We provide additional discussion later in the paper.

Procedural textures in additive manufacturing. The procedural texture synthesizer we propose is general purpose, but one application we aim at is additive manufacturing, and more specifically for the synthesis of multi-material patterns.

There are several major challenges when considering pattern synthesis for AM. First, the patterns should afford for a wide range of effects that the user should be able to freely choose from. In addition, it is expected that the patterns can be progressively spatially graded, triggering different effects in different regions of space.

Second, the patterns should be very efficient to compute. In most cases, thousands of high-resolution slices have to be generated for a single part and sent to the printer. It is also important for the user to be able to preview the result beforehand.

To address these challenges, one line of research advocates for the use of procedural approaches, inspired from procedural texturing in Computer Graphics [Ebert et al. 2003; Livesu et al. 2017]. This is especially interesting to grade materials having anisotropies: the absence of underlying grid affords for complete freedom in orientation [Martínez et al. 2017]. Several frameworks have shown how to exploit procedural synthesis to achieve efficient process planning in AM [Vidimce et al. 2016; Vidimče et al. 2013].

Our work fits well within this trend. However, rather than proposing a specific type of structure, we seek to develop a novel type of procedural texture generator well suited for pattern synthesis in modeling for additive manufacturing. In particular, we seek to design multi-material patterns akin to laminates, with precise control over the material mixtures. We also consider surface displacements, with the ability to produce different roughness profiles.

3 PHASOR NOISE

We now describe our approach in details, and analyze its properties. Applications and results are described in Section 4.

The crux of our technique is to rewrite a Gabor noise as a single sine wave, with a clear separation of intensity and modulation:

$$GaborNoise\,(\mathbf{x}) = I(\mathbf{x}) \sin\left(\phi(\mathbf{x})\right) \tag{1}$$

where $\mathbf{x}$ is the point of evaluation. For conciseness we denote Gabor noise as $G(\mathbf{x})$ in the remainder of this paper. The *phasor noise* is the instantaneous phase ϕ of the rewritten Gabor noise:

$$PhasorNoise\,(\mathbf{x}) = \phi(\mathbf{x}) \tag{2}$$

As we will see, the phasor noise is the argument (angle) of a complex valued version of Gabor noise, which is obtained by summing *phasors*[2] representing the sine waves of the Gabor kernels.

This change of formulation is defined in the most general case of Gabor noise, where kernels of different frequencies, orientation, bandwidth and amplitudes may be summed and where all these parameters can be spatially graded. Thus, phasor noise works in 2D, 3D and for surface noise [Lagae et al. 2009]. It is a pure procedural function that can be implemented in a GLSL pixel shader.

Having clearly separated the instantaneous phase from the noise intensity has two direct benefits. First, it affords for the definition of a noise that perfectly oscillates without local loss of constrast,

[2]See e.g. https://en.wikipedia.org/wiki/Phasor.

Symbol	Definition	Introduced
$G(\mathbf{x})$	Gabor noise	Eqn (1)
$\phi(\mathbf{x})$	phasor noise (instantaneous phase)	Eqn (2)
$\varphi(\mathbf{x})$	phase field - phase shift	Eqn (7)
$I(\mathbf{x})$	intensity field	Eqn (8)
$P(\mathbf{x})$	phasor sinewave $\sin(\phi(\mathbf{x}))$	Eqn (9)
$\mathcal{G}(\mathbf{x})$	complex Gabor noise	Eqn (10)
$\mathbf{x}_j$	center of kernel j	Eqn (5)
F	main frequency of the noise	Eqn (5)
b	bandwith of the noise and Gaussian	Eqn (6)
$\mathbf{u}_{\{j,k,m\}}$	direction of anisotropy	Eqn (15)
$a(\mathbf{x}), a_j(\mathbf{x})$	Gaussian of bandwith b	Eqns (6,8,9)
θ	angle between $\mathbf{u}_k$ and $\mathbf{u}_m$	Sec 3.2.1
φ_j	phase of kernel j	Eqn (8,9)
$\phi_j(\mathbf{x})$	instantaneous phase of kernel j	Eqn (14)

Table 1. Summary of the main notations.

akin to a form of normalized Gabor noise:

$$PhasorSinewave(\mathbf{x}) = \sin(PhasorNoise(\mathbf{x})) \tag{3}$$

We call the result a *phasor sine wave* and denote it by $P(\mathbf{x})$ for conciseness. We further analyze its properties in the following Sections.

The second benefit is to afford for controlling the shape (profile) of the noise oscillations themselves. A direct way to achieve this is to replace the sine wave by another periodic function:

$$ProfiledNoise(\mathbf{x}) = f(PhasorNoise(\mathbf{x})) \tag{4}$$

where f is a function of period 2π.

Alternatively, phasor noise can be used to generate harmonics of a base phasor sine wave, allowing to produce different profiles through an inverse Fourier transform (additive synthesis):

$$Noise\,(\mathbf{x}) = \sum_{k=1}^{K} w_k \cdot \sin\left(k \cdot PhasorNoise(\mathbf{x}) + \phi_k\right)$$

This is a weighted sum of $K > 0$ integer harmonics with synchronized phases. The weights w_k and phase shifts ϕ_k control the produced profile, allowing to synthesize any noise profile, e.g. rectangular, sawtooth, or any other user controlled profile. While similar to specifying a profile function, this offers novel possibilities such as optimizing the weights — profile synthesis — or dynamically adjusting them for filtering purposes (e.g. anti-aliasing).

To understand our approach and its properties we will study noises of increasing complexities, starting by the simplest phasor noise using a single main frequency and orientation, to richer noises.

For convenience we summarize the main notations introduced throughout the paper in Table 1.

3.1 The bi-lobe case

The simplest form of phasor noise is obtained from a Gabor noise using a single kernel. As a reminder, a Gabor noise is obtained by convolving a Gabor kernel with a random impulse process. The spectrum of the noise is then the spectrum of the kernel [Lagae et al. 2009].

In the frequency domain, a Gabor kernel is two symmetric Gaussians, produced by the convolution of two delta functions and a Gaussian. This is illustrated in Figure 2. The delta functions come

from the sine wave in the spatial domain, and are precisely positioned in the spectrum by changing the frequency and orientation of the wave. The Gaussian defines the bandwidth of the noise, e.g. its frequency spread around the perfect sine wave. As in [Neyret and Heitz 2016] we call this simplest form of noise the *bi-lobe case*.

The main observation making our work possible is that, in the bi-lobe case, the noise is obtained by summing in the spatial domain a large number of sine waves *of same frequency* weighted by Gaussians. This can be rewritten as a single weighted sine wave, with an explicit separation of intensity and phase. This operation is akin to phasor addition in signal processing.

We start from a bi-lobe Gabor noise, defined in 2D (the same derivations apply in 3D). It is parameterized by a main frequency $\frac{F}{2\pi}$, along direction $\mathbf{u}$. In every position $\mathbf{x}$, using the simplified expression of Gabor noise with sine waves and no random weights [Tavernier et al. 2019][3]. It evaluates as:

$$G(\mathbf{x}) = \sum_{j=0}^{n} a(\mathbf{x} - \mathbf{x}_j) \sin\left(F \cdot (\mathbf{x} - \mathbf{x}_j) \cdot \mathbf{u}\right) \tag{5}$$

where a is a centered Gaussian of bandwidth b:

$$a(\mathbf{x}) = e^{-\pi b^2 ||\mathbf{x}||^2} \tag{6}$$

Through phasor addition, we rewrite the noise as:

$$G(\mathbf{x}) = I(\mathbf{x}) \sin\left(F \cdot \mathbf{x} \cdot \mathbf{u} + \varphi(\mathbf{x})\right) \tag{7}$$

where I is the intensity and φ the phase. This rewritten Gabor noise is strictly equivalent to the initial one. In this case the phasor noise is the instantaneous phase field $\phi(\mathbf{x}) = F \cdot \mathbf{x} \cdot \mathbf{u} + \varphi(\mathbf{x})$, and we denote the *bi-lobe phasor sine wave* as $P(\mathbf{x}) = \sin(\phi(\mathbf{x}))$.

Our claim is that, in the bi-lobe case, removing the intensity I – keeping only the sine wave – creates a noise oscillating with perfect local contrast, i.e. there are no low frequencies in the spectrum of variance. Yet, the local shape of the sine wave is preserved, and the overall frequency remains F. This means that harmonics of the phasor sine wave are preserved as well, and thus the characteristics of the profile function — that can be seen as a Fourier combination of harmonics of the base phasor sine wave — are preserved.

[3]This is not mandatory for our reformulation to be well defined, we adopt this point of view for the sake of simplicity.

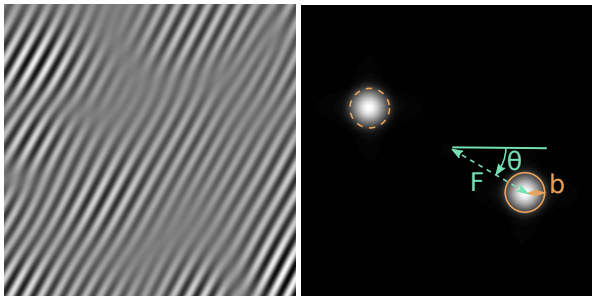


Fig. 2. A bi-lobe Gabor noise and its Fourier transform. The orientation is θ, the frequency of the sine wave F and the bandwith b. They are all clearly visible in the spectrum. Note that the image of the Fourier transform has been cropped for readability.

While this might seem obvious when looking at Equation 7, the difficulty comes from the fact that the new phase $\varphi(\mathbf{x})$ does depend on $\mathbf{x}$. By varying it could reshape the sine wave arbitrarily, significantly changing the frequency content of the result.

Before entering the details, let us give the full expression of I and φ obtained from the sum of sine waves. This is a direct application of the phasor addition formulas[4]. The intensity is obtained as:

$$I(\mathbf{x}) = \sqrt{\left(\sum_j a_j(\mathbf{x}) \sin(\varphi_j)\right)^2 + \left(\sum_j a_j(\mathbf{x}) \cos(\varphi_j)\right)^2} \tag{8}$$

and the phase is obtained as:

$$\varphi(\mathbf{x}) = \operatorname{atan2}\left(\sum_j a_j(\mathbf{x}) \sin(\varphi_j), \sum_j a_j(\mathbf{x}) \cos(\varphi_j)\right) \tag{9}$$

where $a_j(\mathbf{x}) = a(\mathbf{x} - \mathbf{x}_j)$ and $\varphi_j = -\mathbf{x}_j \cdot \mathbf{u} \cdot F$, and where $\operatorname{atan2}(y, x)$ returns the polar angle of the vector (x, y).

It is worth noting that atan2 is only defined if the vector norm is non zero ($I(\mathbf{x}) \neq 0$). We later discuss singularities occurring when I vanishes (Section 3.1.4).

In the next sections we will verify that by omitting the intensity we eliminate the local loss of contrast in the bi-lobe case of the noise. At the same time, the overall frequency content is preserved: the sine wave still oscillates at the same main frequency.

3.1.1 Phasor field. The rewrite of Gabor noise in terms of a single sine wave exposes an underlying complex valued field. Indeed, $I(\mathbf{x})$ and $\varphi(\mathbf{x})$ — Equations 8 and 9 — can be directly interpreted as the modulus and argument of a sum of complex numbers:

$$\sum_{j=1}^{n} a_j(\mathbf{x}) \left(\cos(\varphi_j) + i \sin(\varphi_j)\right) = \sum_{j=1}^{n} a_j(\mathbf{x}) e^{i\varphi_j}$$

where $e^{i\varphi_j}$ are random complex numbers of unit modulus. This is the *phasor field*, illustrated in Figure 3.

To understand its effect, let us consider the complex version of Gabor noise underlying the phasor addition:

$$\mathcal{G}(\mathbf{x}) = e^{iF \cdot \mathbf{x} \cdot \mathbf{u}} \sum_{j=1}^{n} a_j(\mathbf{x}) e^{i\varphi_j} \tag{10}$$

The real valued Gabor noise is obtained as $G(\mathbf{x}) = \operatorname{Im}(\mathcal{G}(\mathbf{x}))$.

$\mathcal{G}$ is the product of two terms: a perfectly oscillating complex wave ($e^{iF \cdot \mathbf{x} \cdot \mathbf{u}}$) and the phasor field. Thus, in a bi-lobe Gabor noise, the phasor field is the source for both the local loss of contrast — via its varying modulus $I(\mathbf{x})$ — and the phase shifts $\varphi(\mathbf{x})$ — via its argument. A key question is whether the phase shifts $\varphi(\mathbf{x})$ could be strong enough to introduce frequencies lower than F.

Interestingly, the phasor field is a sparse convolution obtained by summing spatially located Gaussians (a_j) multiplied by random, unit complex numbers. Therefore, the produced random field has the bandwidth b of the Gaussians, and the speed at which φ changes is driven by b. When b is much smaller that F, the variations in phase due to φ are much slower than that of the base frequency F.

[4]see e.g. https://ccrma.stanford.edu/~jos/filters/Proof_Using_Trigonometry.html

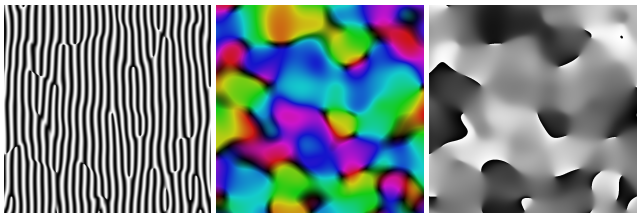


Fig. 3. **From left to right:** a phasor sine wave, its underlying phasor field (directions as hue, modulus as luminance) and associated phase field φ.

This implies that φ has little impact on the local profile of the sine function, while its impact on the overall frequency is limited by b.

This hints at a first result: the spectrum of the phasor sine wave is similar to that of the Gabor noise — since φ has little impact on F — while its spectrum of variance exhibit no low frequency (no local loss of contrast in the signal) — since φ cannot reshape the sine wave profile significantly. Note that this is only true for b smaller than F.

The phasor field also reveals that the phase will exhibit singularities. Indeed, in some singular locations the modulus of the phasor field vanishes: this is where the sum of complex numbers becomes null. Around these singularities φ rotates abruptly. We discuss this effect in Section 3.1.4.

3.1.2 Spectrum of the phasor sine wave. In this section and the following we only outline the main results of our analysis of phasor noise. Please refer to the supplemental material for full details and derivations. We perform the analysis on periodic instances of the phasor sinewave. This is achieved by sampling kernels in a periodic domain. In this case, the Fourier transform of P is :

$$\mathcal{F}_x[P(\mathbf{x})](\boldsymbol{\omega}) = \boldsymbol{\omega}_0 \sum_{k=-\infty}^{+\infty} \mathcal{F}_x[B(\mathbf{x})\,P(\mathbf{x})](k\boldsymbol{\omega}_0)\,\delta_{k\boldsymbol{\omega}_0}(\boldsymbol{\omega}) \qquad (11)$$

with B a windowing function verifying the partition of unity property and where $\boldsymbol{\omega}_0 = \frac{1}{T_0}$ with T_0 the analysis period. The amplitude of the deltas δ in $\mathcal{F}_x[P(\mathbf{x})]$ is driven by the spectrum of $\mathcal{F}_x[B(\mathbf{x})\,P(\mathbf{x})]$ which is:

$$\begin{aligned}\mathcal{F}_x[B(\mathbf{x})\,P(\mathbf{x})](\boldsymbol{\omega}) =& \frac{1}{2}\mathcal{F}_x[B(\mathbf{x})\,(\sin(\varphi(\mathbf{x})) - i\cos(\varphi(\mathbf{x})))]\left(\boldsymbol{\omega} - \frac{F}{2\pi}\mathbf{u}\right)\\ &+\frac{1}{2}\mathcal{F}_x[B(\mathbf{x})\,(\sin(\varphi(\mathbf{x})) + i\cos(\varphi(\mathbf{x})))]\left(\boldsymbol{\omega} + \frac{F}{2\pi}\mathbf{u}\right)\end{aligned}$$

The two terms are functions convoluted with two symmetric deltas located at the main frequency of the noise ($\boldsymbol{\omega} \pm \frac{F}{2\pi}$). These two functions are smooth and their Fourier transforms are rapidly decaying (holds for one dimension, verified experimentally in 2D and 3D due to singularities – please refer to the supplemental material for details). This explains the presence of two lobes in the spectrum.

3.1.3 Spectrum of variance the phasor sine wave. We use the same approach to derive the spectrum of variance, noting that if P is periodic, then P^2 is also periodic. The spectrum of variance is the spectrum of the squared signal, $\mathcal{F}_x[B(\mathbf{x})\,P^2(\mathbf{x})]$. In this case, the amplitude of the deltas in the Fourier transform are obtained as

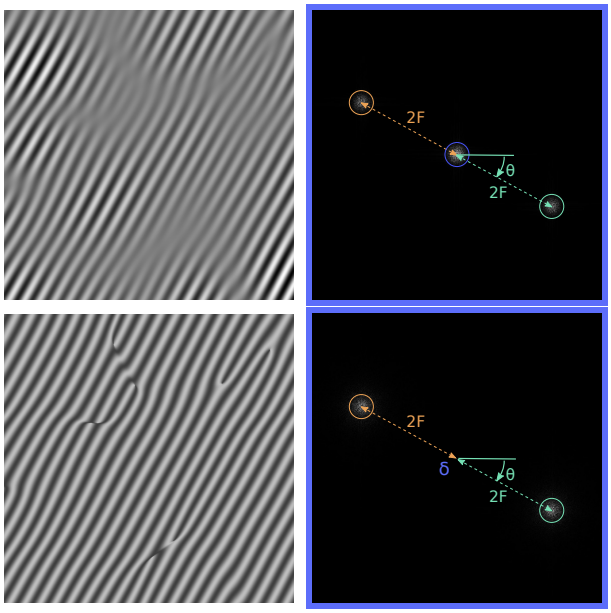


Fig. 4. Noise (left) and spectrum of variance (right). **First row:** Gabor noise. **Second row:** Phasor sine wave. The colors are the same as the ones used for equation 12. Note the absence of low frequency – no center lobe – in the spectrum of variance of the phasor sine wave.

(coloring the main terms to help the description):

$$\begin{aligned}\mathcal{F}_x[B(\mathbf{x})\,P^2(\mathbf{x})](\boldsymbol{\omega}) =& \frac{1}{2}\delta_0(\boldsymbol{\omega})\\ &+ \frac{1}{4}\mathcal{F}_x[B(\mathbf{x})\,(1-2\cos^2(\varphi(\mathbf{x})))]\left(\boldsymbol{\omega} - \frac{2F}{2\pi}\mathbf{u}\right)\\ &- \frac{i}{2}\mathcal{F}_x[B(\mathbf{x})\sin(\varphi(\mathbf{x}))\cos(\varphi(\mathbf{x}))]\left(\boldsymbol{\omega} - \frac{2F}{2\pi}\mathbf{u}\right)\\ &+ \frac{1}{4}\mathcal{F}_x[B(\mathbf{x})\,(1-2\cos^2(\varphi(\mathbf{x})))]\left(\boldsymbol{\omega} + \frac{2F}{2\pi}\mathbf{u}\right)\\ &+ \frac{i}{2}\mathcal{F}_x[B(\mathbf{x})\sin(\varphi(\mathbf{x}))\cos(\varphi(\mathbf{x}))]\left(\boldsymbol{\omega} + \frac{2F}{2\pi}\mathbf{u}\right)\end{aligned} \qquad (12)$$

where the terms in orange and green are again two functions that are convolved with symmetric deltas located at twice the main frequency of the noise ($\pm\frac{2F}{2\pi}\mathbf{u}$). These terms correspond to the lobes shown in Figure 4. The central dirac δ_0 corresponds to the average value of the variance, which is non zero. The most important observation is the absence of any central lobe, which indicates that the phasor sine wave suffers no local loss of contrast.

The non-centered lobes could however contribute low frequencies in the spectrum of variance, depending on the ratio between F and b (the bandwidth of the Gaussians a_j). In this case removing the intensity terms from the Gabor noise does not remove all low frequencies from the spectrum of variance. Going back to the interpretation of the bi-lobe noise in terms of underlying random phasor field (Section 3.1), this is an instance where the phasor field varies

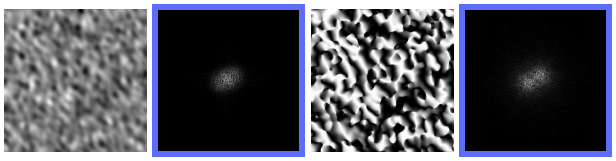


Fig. 5. Noise and spectrum of variance. (**Left:** Gabor noise, **Right:** phasor sine wave). In this case the ratio between the bandwith of the gaussian and the frequency is large enough for φ to significantly reshape the main sine wave.

more rapidly than the base sine wave frequency, thereby changing its profile. Such a case is illustrated in Figure 5.

3.1.4 When the phasor modulus goes to zero. The intensity $I(\mathbf{x})$ of the Gabor noise goes to zero in a few singular locations. Note that we do not suffer from numerical issues since we define the phasor sine wave P by omitting to multiply by I. The only potential source of trouble (atan2) is well identified. In the implementation null phasors never truly occur thanks to the small rounding errors.

In such locations, we can observe singularities in the phase φ: abrupt rotations around a singular point or abrupt variations across ridges (see Figure 6). By going to zero I hides these occurrences in the original Gabor noise. When removing I we bring these singularities into the signal. This creates sharp points and edges, which correspond to locations in the Gabor noise where the sine wave oscillations are in phase opposition.

There are two main types of issues (see Figure 6). First, those which occur at a point where one sine wave band disappears. Second, those which occur along abrupt variations in the field, where the wave fronts on both sides oppose. These correspond to curved ridges that are connecting two point singularities.

The singularities only add high frequencies in the direction orthogonal to the waves. As they are relatively few and localized, their influence on the overall signal is limited. The number of singularities is impacted by the choice of Gabor bandwidth b – the smaller the bandwidth the closer to an ideal sine wave.

It is worth noting that Neyret and Heitz [2016] proposed an iterated scheme to filter away these issues – paving the way for potential solutions. However, the several iterations of direct and inverse Fourier transforms cannot be done in a procedural manner. Also, such a filtering approach does not give access to an explicit separation between a base sine wave and its argument, precluding the ability to control the oscillation's profiles.

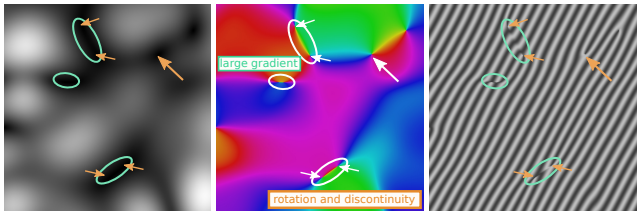


Fig. 6. **Left:** Intensity field. **Middle:** Color coded phase field (angled mapped on hue). **Right:** Phasor sine wave. Two types of artifacts appear in the phase field. The first are point singularities (in orange) around which the phase rotates rapidly. It corresponds to the disappearance of a wave front. When two such singularities are close to one another in the direction orthogonal to the noise (in cyan), a ridge appears in the phase with opposing wave fronts across.

While we currently do not address these singularities, this is a clear venue of future work. We believe the phasor field interpretation to open interesting possibilities by directly exposing the source of these defects.

3.2 General case

We now present our approach to produce general phasor noise patterns with richer frequency content.

As with Gabor noise, kernels of different orientations and frequencies are combined together. We then extract the instantaneous phase modulation $\phi(\mathbf{x})$ and evaluate a single sine wave at the end. Thus, in the most general case we define the noise as:

$$G(\mathbf{x}) = I(\mathbf{x}) \sin(\phi(\mathbf{x})) \tag{13}$$

It contains Gabor kernels at arbitrary frequencies F_j and direction $\mathbf{u}_j$ (unit vectors), that is:

$$G(\mathbf{x}) = I(\mathbf{x}) \sin(\phi(\mathbf{x})) = \sum_{j=1}^{n} a_j(\mathbf{x}) \sin(\phi_j(\mathbf{x})) \tag{14}$$

where

$$a_j(\mathbf{x}) = a\left(||\mathbf{x} - \mathbf{x_j}||\right) \text{ and } \phi_j(\mathbf{x}) = F_j \cdot (\mathbf{x} - \mathbf{x}_j) \cdot \mathbf{u}_j \tag{15}$$

and the phasors are $a_j(\mathbf{x}) \cdot \left(\cos(\phi_j(\mathbf{x})), \sin(\phi_j(\mathbf{x}))\right)$.
Through phasor addition, we obtain I and ϕ as:

$$I(\mathbf{x}) = \sqrt{\left(\sum_{j=1}^{n} a_j(\mathbf{x}) \cos(\phi_j(\mathbf{x}))\right)^2 + \left(\sum_{j=1}^{n} a_j(\mathbf{x}) \sin(\phi_j(\mathbf{x}))\right)^2}$$

$$\phi(\mathbf{x}) = \text{atan2}\left(\sum_{j=1}^{n} a_j(\mathbf{x}) \sin(\phi_j(\mathbf{x})), \sum_{j=1}^{n} a_j(\mathbf{x}) \cos(\phi_j(\mathbf{x}))\right)$$

Equivalently, I and ϕ are the modulus and argument of a complex version of the noise $\mathcal{G}$, defined similarly as in Section 3.1.1.

Note that Equation 13 remains strictly equivalent to the original Gabor noise, but rewritten as a single sine wave. In the following we will study the properties of the phasor sine wave $P = \sin(\phi(\mathbf{x}))$ obtained by omitting I.

To understand the behavior in the general case, we start by discussing the case of *two* bi-lobes in Section 3.2.1. We then proceed to the general case combining kernels of different frequencies and orientations in Section 3.2.2. It can be understood as a generalization of the two bi-lobe case.

3.2.1 Two bi-lobes. Let us consider a Gabor noise combining two bi-lobe noises of same frequency F but with different orientations $\mathbf{u}_k$ and $\mathbf{u}_m$. Applying the phasor addition we obtain:

$$\begin{aligned} G(\mathbf{x}) &= G_{\mathbf{u}_k}(\mathbf{x}) + G_{\mathbf{u}_m}(\mathbf{x}) \\ &= I_{\mathbf{u}_k}(\mathbf{x}) \sin(\phi_{\mathbf{u}_k}(\mathbf{x})) + I_{\mathbf{u}_m}(\mathbf{x}) \sin(\phi_{\mathbf{u}_m}(\mathbf{x})) \\ &= I(\mathbf{x}) \sin(\phi(\mathbf{x})) \end{aligned}$$

with:

$$I(\mathbf{x}) = \sqrt{(I_{\mathbf{u}_k}\cos\phi_{\mathbf{u}_k} + I_{\mathbf{u}_m}\cos\phi_{\mathbf{u}_m})^2 + (I_{\mathbf{u}_k}\sin\phi_{\mathbf{u}_k} + I_{\mathbf{u}_m}\sin\phi_{\mathbf{u}_m})^2}$$

$$\phi(\mathbf{x}) = atan2(I_{\mathbf{u}_k}\sin\phi_{\mathbf{u}_k} + I_{\mathbf{u}_m}\sin\phi_{\mathbf{u}_m}, I_{\mathbf{u}_k}\cos\phi_{\mathbf{u}_k} + I_{\mathbf{u}_m}\cos\phi_{\mathbf{u}_m})$$

This is the sum of two phasors, one for each initial Gabor noise $G_{\mathbf{u}_k}$ and $G_{\mathbf{u}_m}$. These phasors vary in space and have different behaviors along different directions.

In the following we discuss the spectrum of variance and spectrum of the two bi-lobe phasor sine wave. Detailed derivations, obtained from the complex version of the noises, are provided in the accompanying supplemental material.

Please note that the angle θ *between* the two bi-lobes, that is between directions $\mathbf{u}_k$ and $\mathbf{u}_m$, plays a role in the analysis.

Spectrum of variance: two bi-lobe Gabor noises. In total, the spectrum of variance for Gabor noise in the two bi-lobes case exhibits Gaussians in nine locations, shown in Figure 8, left. Five of these Gaussians correspond to the squared bi-lobe noises: centered Gaussians and Gaussians at $\pm 2\mathbf{u}_k$ and $\pm 2\mathbf{u}_m$ coming from respectively $G^2_{\mathbf{u}_k}$ and $G^2_{\mathbf{u}_m}$ (recall that the spectrum of variance is the spectrum of the squared noise). The four other Gaussians emerge from interactions between the two bi-lobes. They appear when squaring the Gabor noise: $G^2(\mathbf{x}) = G^2_{\mathbf{u}_k}(\mathbf{x}) + G^2_{\mathbf{u}_m}(\mathbf{x}) + 2G_{\mathbf{u}_k}G_{\mathbf{u}_m}$. The last product term leads to the remaining four Gaussians located at $\pm(\mathbf{u}_k - \mathbf{u}_m)$ and $\pm(\mathbf{u}_k + \mathbf{u}_m)$: the product in the spatial domain is a convolution in the spectral domain, resulting in the diagonal lobes.

Spectrum of variance: two bi-lobe phasor sine waves. We now consider the spectrum of variance of the phasor sine wave resulting from omitting I, see Figure 7 middle. We observe the absence of centered Gaussians — implying there is no loss of local contrast — and the presence of lobes at $\pm 2\mathbf{u}_k$ and $\pm 2\mathbf{u}_m$. The Gaussians at $\pm(\mathbf{u}_k - \mathbf{u}_m)$ have disappeared ; however, those at $\pm(\mathbf{u}_k + \mathbf{u}_m)$ remain.

This is only partially satisfying as the Gaussians at $\pm(\mathbf{u}_k + \mathbf{u}_m)$ may fold back onto low frequencies depending on the angle θ between $\mathbf{u}_k, \mathbf{u}_m$. It is possible to obtain the symmetric effect, ensuring that Gaussians at $\pm(\mathbf{u}_k + \mathbf{u}_m)$ are removed in place of those that $\pm(\mathbf{u}_k - \mathbf{u}_m)$, see Figure 7, right. This is achieved by inverting the direction of *one* of the base noises. This may be used to select the approach depending on the angle θ between $\mathbf{u}_k, \mathbf{u}_m$, avoiding low frequencies to be introduced by the diagonal bi-lobes (e.g. in Figure 7 selecting the middle case instead of the right one where the lobes get closer to the center). The switch should occur when $\mathbf{u}_k \cdot \mathbf{u}_m$ changes sign, that is when the angle θ between both directions exceeds $\frac{\pi}{2}$.

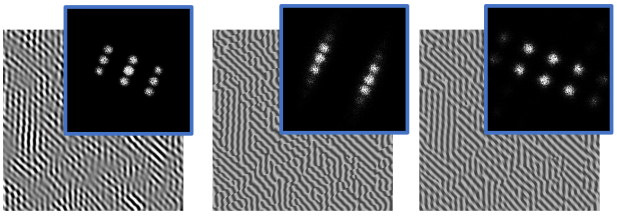


Fig. 7. Spectrum of variance of two bilobes. **Left:** Gabor noise exhibiting local loss of contrast, resulting in a central lobe in the spectrum of variance. **Middle:** Phasor sine wave with the two bi-lobes separated by $\frac{\pi}{4}$. The local loss of contrast is completely eliminated: no central lobe. Note the two-sided spectrum of variance. **Right:** Same phasor sine wave using the dual formulation to eliminate the other lobes instead.

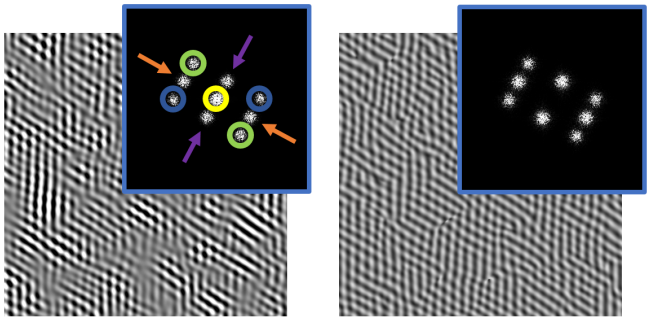


Fig. 8. Two bi-lobes noise. **Left:** Gabor noise and its spectrum of variance. The nine Gaussian locations are clearly visible; in yellow the center Gaussians, in blue and green the bi-lobes Gaussians at $\pm 2\mathbf{u}_k$ and $\pm 2\mathbf{u}_m$ and outlined by orange and purple arrows the interaction Gaussians at (respectively) $\pm(\mathbf{u}_k + \mathbf{u}_m)$ and $\pm(\mathbf{u}_k - \mathbf{u}_m)$. **Right:** Normalized noise where the center lobe is removed. This is not a phasor sine wave as an intensity term remains, please refer to the text.

As a side note, we can also choose to cancel only the centered Gaussians in the Gabor spectrum of variance. To achieve this, we keep the noise intensity I but divide it by $\sqrt{I^2_{\mathbf{u}_k} + I^2_{\mathbf{u}_m}}$. The full noise expression becomes $\frac{I(\mathbf{x})}{\sqrt{I^2_{\mathbf{u}_k}(\mathbf{x}) + I^2_{\mathbf{u}_m}(\mathbf{x})}}\sin\phi(\mathbf{x})$. An example is shown in Figure 8, right. The rationale is that the central lobe originates from that of both individual Gabor noises. Hence, by rescaling the noise with $\frac{I(\mathbf{x})}{\sqrt{I^2_{\mathbf{u}_k}(\mathbf{x}) + I^2_{\mathbf{u}_m}(\mathbf{x})}}$ we cancel the center lobes but preserve the others. However, this renormalization is different from our initial intent with phasor noise: it only applies to a sine wave profile.

Phase singularities. As with the bi-lobe case, phase singularities appear where intensity vanishes. However, another type of phase defect appears, which depends on the angle θ between the two bi-lobe directions.

To understand this effect let us consider — without loss of generality — a single direction. For simplicity we select $\mathbf{u}_k$. Along this direction the phasors take different frequencies, respectively F and $F \cdot \mathbf{u}_m \cdot \mathbf{u}_k$ (this stems from the dot product with $\mathbf{u}_j$s in Equation 15). Now, depending on the angle θ between $\mathbf{u}_m$ and $\mathbf{u}_k$, the two phasors will interfere with one another in different manners. The different cases we are about to discuss are illustrated in Figure 9.

If $\mathbf{u}_m = \mathbf{u}_k$ the pattern is exactly similar to the bi-lobe case. For small angles θ, we obtain a smooth pattern with gentle orientation changes. For larger angles however, interferences start to introduce sharp turns in the phase. This is most visible when the angle θ between the two bi-lobes is $\frac{\pi}{2}$. Visually this results in an abrupt interruption of the waves ('chopping'), shown in Figure 9, rightmost. High frequencies appear in the spectrum reflecting the perceivable edges. This defect is easily explained by considering the phasor diagram of two sine waves in the $\frac{\pi}{2}$ case, as detailed in Figure 10.

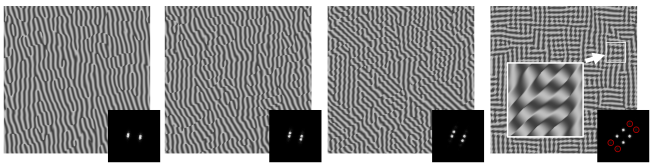


Fig. 9. Phasor sine wave in the two bi-lobes case, with separation angles θ of respectively (left to right) $\frac{\pi}{10}$, $\frac{\pi}{6}$, $\frac{\pi}{4}$, $\frac{\pi}{2}$. The insets at the bottom right show the power spectrum of the patterns. As the angle θ increases, high frequencies due to interferences between the two bi-lobes appear.

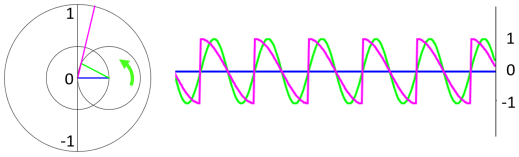


Fig. 10. *Phasor diagram* illustrating the interactions between two orthogonal 2D sine waves $\sin(\mathbf{x} \cdot \mathbf{u_k})$ (green) and $\sin(\mathbf{x} \cdot \mathbf{u_m})$ (blue), along direction $\mathbf{u_k}$. The diagram on the left shows the two sine wave phasors as blue and green vectors: the green phasor rotates while the blue phasor remains fixed: the blue sine wave does not oscillate along $\mathbf{u_k}$. The right plot shows the resulting waves along $\mathbf{u_k}$. The purple is the result obtained by phasor addition, omitting intensity. Since both sine waves have the same intensity, the sum of the green and blue phasors vanish in specific locations, and the purple phasor abruptly changes from $\frac{\pi}{2}$ to $-\frac{\pi}{2}$, interrupting the purple sine wave. Waves of different intensities do not exhibit this issue.

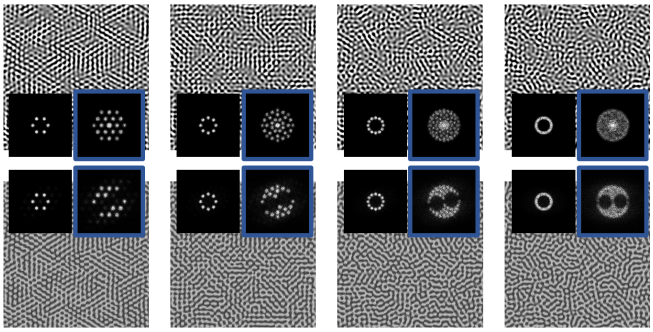


Fig. 11. **Top row:** Gabor noises mixing an increasing number of bi-lobes (3, 4, 6, 32) with orientations regularly distributed over π. **Bottom row:** Corresponding phasor sine wave. Both the spectrum and spectrum of variance (outlined in blue) are shown. The achieved power spectrum are similar. The phasor noise has some additional energy in higher frequencies due to singularities. The phasor sine wave eliminates a large part of the low frequency disc in the spectrum of variance. The parts that remain are due to the two-sided normalization observed with the two bi-lobes case, here generalized for large number of bi-lobes. We discuss this further in Section 3.2.2.

Whether the visual artifact is problematic depends on the target application. However, these defects appear progressively. This hints at a possibility to reduce their impact by locally avoiding mixing patterns with large differences, which we discuss in Section 3.2.2.

3.2.2 Mixed kernels. The most general case involves summing kernels of randomly sampled orientations and/or frequencies [Lagae et al. 2009]. This affords for the synthesis of richer patterns. Isotropic patterns are of particular interest, as well as progressive spatial gradation between different orientations and frequencies.

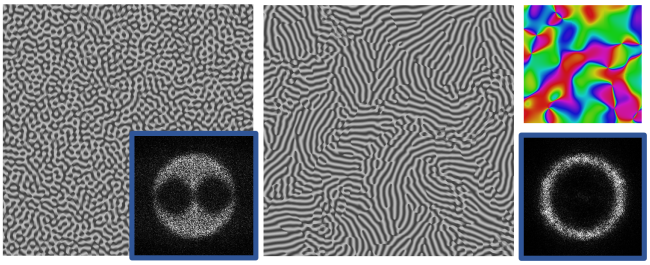


Fig. 12. **Left:** Phasor sine wave with randomly sampled orientations in $[-\frac{\pi}{2}, \frac{\pi}{2}]$. The spectrum of variance has a two-sided aspect. **Middle:** Phasor noise with locally coherent random direction field. The spectrum of variance (bottom right) is now close to an ideal ring, with little residual energy in the low frequencies. The top right image shows the direction field with angle mapped on hue.

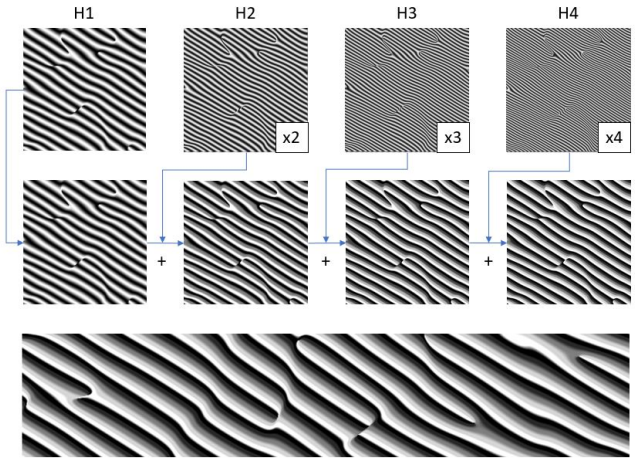


Fig. 13. Additive synthesis of a sawtooth profile. **Top left**: Base phasor sine wave. **Top row**: integer harmonics (without their weight for clarity). **Middle row** : Progressive summation of the weighted harmonics. **Bottom**: Obtained pattern with sawtooth profile ; a few more harmonics would be required to properly converge.

Figure 11 illustrates the patterns obtained when mixing an increasing number of bi-lobes. Each additional bi-lobe introduces new pairwise interactions, which are as many additional Gaussians in the spectrum of variance (the number of Gaussians doubles for each new bi-lobe).

We observe a generalization of the behavior of two bi-lobes: A large portion of the low frequency disk in the spectrum of variance of the phasor sine wave is removed, albeit in a two-sided manner. The remaining low frequencies are due to pairs of bi-lobes separated by an angle θ above $\frac{\pi}{2}$. However, if the angle between all bi-lobe pairs remains below $\frac{\pi}{2}$, all low frequencies are suppressed. We exploit this to formulate an isotropic phasor noise that strongly reduces the mixing of incompatible bi-lobes.

Locally coherent isotropic noise. In the standard approach, the isotropic noise is produced by randomly sampling orientations in every kernel, i.e. $\mathbf{u}_k = random_unit_vector(\mathbf{x}_k)$. To better control which kernels mix together in the noise we modify the sampling of directions. Instead of using random independent sampling, we rely on a Gaussian sparse convolution noise with controlled bandwidth:

$$u_j = \frac{\sum_{j=1}^{n} g\left(\mathbf{x}-\mathbf{x}_j\right)\mathbf{r}_j(\omega_f)}{||\sum_{j=1}^{n} g\left(\mathbf{x}-\mathbf{x}_j\right)\mathbf{r}_j(\omega_f)||}$$

where $g(\mathbf{x}) = e^{-\pi\cdot b_f^2\cdot||\mathbf{x}||^2}$ and $\mathbf{r}_j(\omega_f)$ are random unit vectors sampled in the range of angles $[-\omega_f, \omega_f]$. The impulses $\mathbf{x}_j$ are independent from those of the phasor noise. By changing b_f we control how locally coherent the direction field is. By increasing ω_f we smoothly go from the bi-lobe case to arcs, and ultimately to a full ring. To obtain an isotropic noise we set $\omega_f = 2.4$ and $b_f = \frac{b}{2}$ with b the bandwidth of the phasor noise. This does not remove all areas of abrupt changes, but strongly reduces the spurious interferences. As shown in Figure 12 the ring in the spectrum of variance is now close to perfect. Of course, while the spectral content is now satisfactory, this significantly changes the aspect of the field. Nevertheless, this provides an additional control to trade-off between the more standard isotropic aspect and an aspect which, while being locally similar to the bi-lobe case, is globally isotropic.

Finally, let us note that this incurs a performance penalty, as the direction field has to be sampled at every kernel.

3.3 Oscillation's profiles

Phasor noise provides an instantaneous phase modulation that affords for the synthesis of noises with arbitrarily shaped oscillation profiles.

This can be achieved in two ways. The most direct is to feed the phasor noise into a periodic function having a 2π period. This generates a field where the main oscillations follow that of a phasor sine wave, but where the oscillations are shaped by the new periodic function. In addition, the periodic function itself may vary spatially, offering a unique control over the patterns, as shown in Figure 1.

The second way is to perform weighted sums of integer harmonics of a base phasor sine wave, as illustrated in Figure 13. This is a form of inverse Fourier transform, using the phasor sine wave as the fundamental harmonic. The linear summation has the advantage of being analytic and offering a direct control over introduced frequencies. It also paves the way for optimizing the profiles or dynamically adjusting them during rendering (e.g. anti-aliasing).

Figure 23 shows various profile shapes and how they modify the spectrum and spectrum of variance. We further explore profile reshaping in Section 4.

4 RESULTS

Our technique is especially well suited to produce highly contrasted patterns and strongly oriented, laminate-like stochastic structures. Direct control over the profile enables further spatial gradations.

We start by showing the high degree of control made possible by phasor noise in Section 4.1. We continue with applications in modeling surface displacements and textures in Section 4.2. We finally illustrate potential applications for synthesizing multi-material laminates, in the context of additive manufacturing, in Section 4.3.

Throughout the section, please keep in mind that the shown effects are all purely procedural [Lagae et al. 2010], and computed in a few milliseconds from a pixel shader.

4.1 Controls

Figure 15 illustrates the variety of controls enabled by phasor noise. In this particular case the profile being used is a rectangular signal which 'up' time in each period can be modified – visually modifying the white/black ratio and producing thicker or thinner lines. We refer to this profile as a Pulse Width Modulation (PWM) profile.

As can be seen in Figure 15 all parameters can be freely spatially graded, including the profile parameters themselves. Figure 14 shows a similar control using a grayscale image as a source.

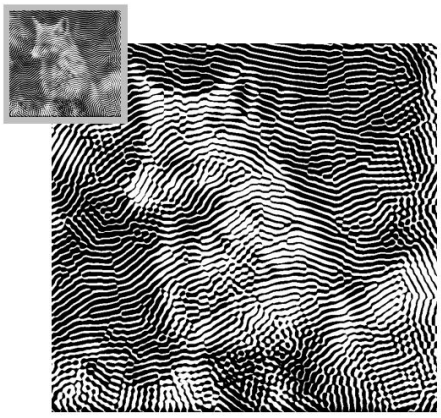


Fig. 14. Pattern produced from phasor noise, controlling a PWM profile width from the image gray level and the orientations from the gradients.

Reshaping the profiles is a powerful tool, especially when targeting a precise histogram and homogeneous appearance. For instance, Figure 16 compares the synthesis of a noise with a fixed, chosen width (25% high). With phasor noise we immediately get the corresponding profiled pattern, which exhibits the chosen average histogram. On the contrary, thresholding a Gabor noise involves manually tuning the threshold, and the achieved result is not spatially homogeneous.

Figure 1 illustrates how different oscillation profiles may be blended together while controlling other parameters. The profile functions are linearly interpolated based on the abscissa: the two profiles enclosing the lookup point are accessed and the result is linearly interpolated.

4.2 Surface displacements and texturing

Another promising application of phasor noise is its use for modeling surface displacements. The profile, frequency and orientation controls allow to produce different displacements along the parts, changing significantly the appearance of surfaces, and possibly their friction and tactile properties.

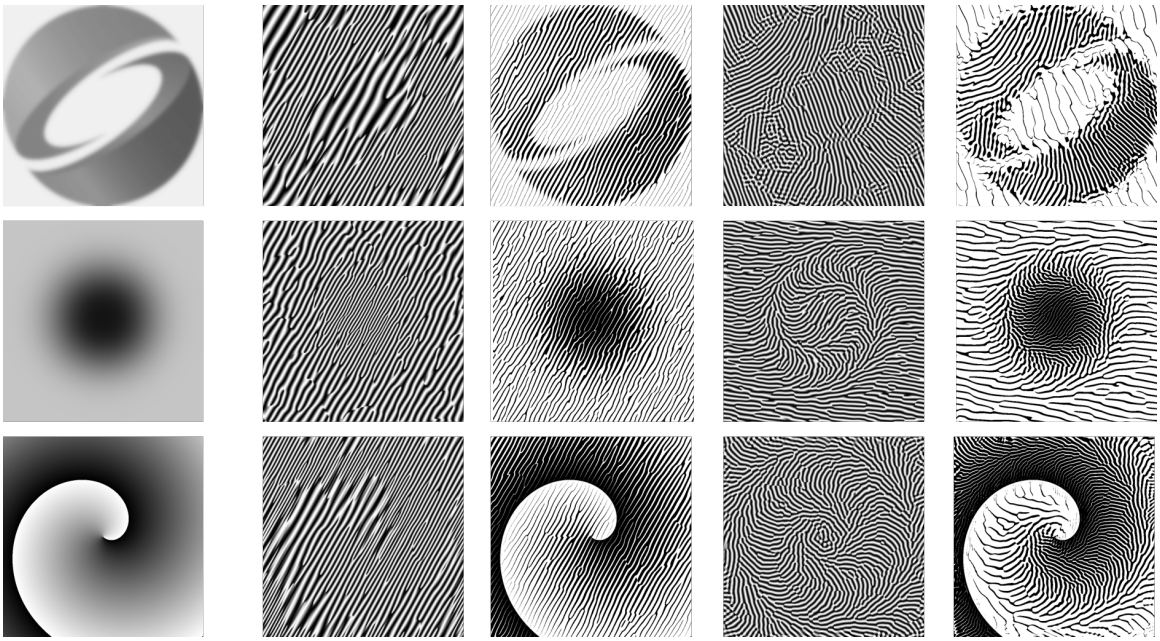


Fig. 15. **Columns, from left to right:** Control fields, frequency control, PWM 'width' control, orientation control, all together.

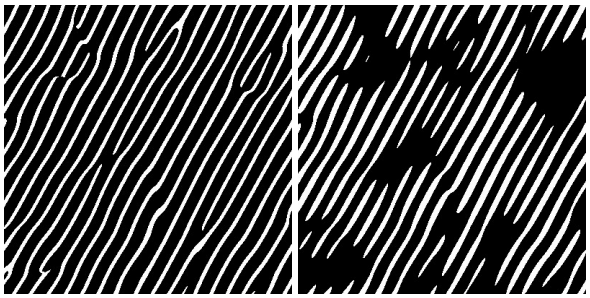


Fig. 16. **Left**: phasor noise through a PWM profile having 25% white (global average verified at 25.007 %). **Right**: Gabor noise thresholded to obtain an average of 25% (final at 0.25902%) the threshold was manually found as it depends on the noise parameters.

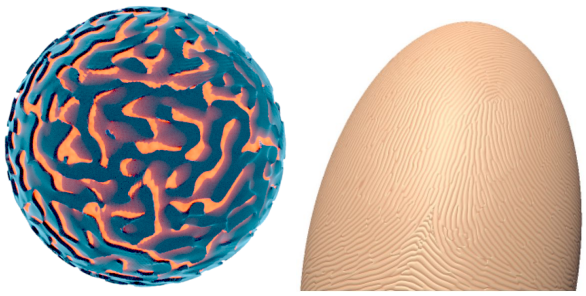


Fig. 17. **Left:** A lava sphere which surface is synthesized by an isotropic phasor noise with sawtooth profile. **Right:** Fingerprints modeled with a phasor sine wave following a user specified direction field.

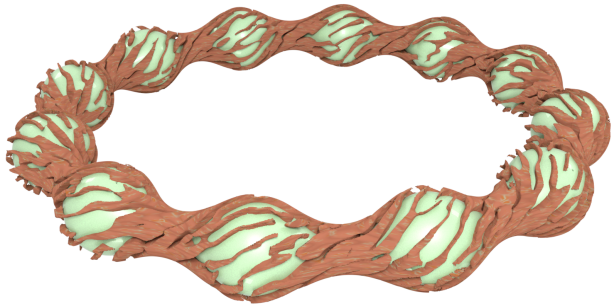


Fig. 18. Bracelet modeled with a phasor noise displacement. By varying the profile according to the local section of the torus, a tearing effect is achieved.

Figure 17 shows a lava sphere which surface is modeled by an isotropic phasor noise with sawtooth profile, as well as fingerprints generated by a phasor sine wave from a user specified direction field. Changing the random seed generates new sets of unique fingerprints. Figure 18 shows a complex 3D bracelet which outer crust is modeled with a phasor noise displacement.

Phasor noise also provides novel possibilities in terms of procedural texturing. For instance, Figure 19 shows cracked tree bark textures that answer different requirements along cone cross-sections: the bark stripes are controlled independently in their width and number. Such precise controls would not be possible through simple lookup tables applied to existing noise patterns. These effects being produced from a pixel shader they could also easily evolve with a dynamic geometry. This provides a fully procedural alternative to

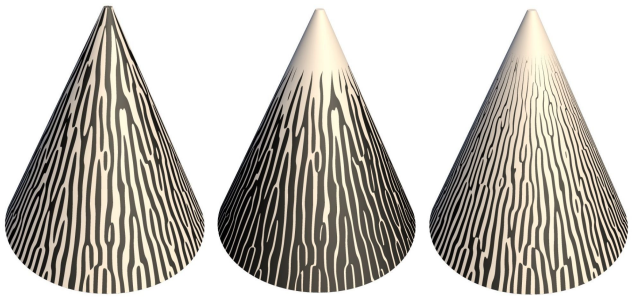


Fig. 19. Cones covered with a cracked tree bark pattern, using a PWM profile. Each cone achieves a different effect through different controls. **Left:** constant width and constant frequency, resulting in less stripes at the top. **Middle:** varying width and constant frequency, preserves the amount of white along cross-sections. **Right:** constant width and varying frequency.

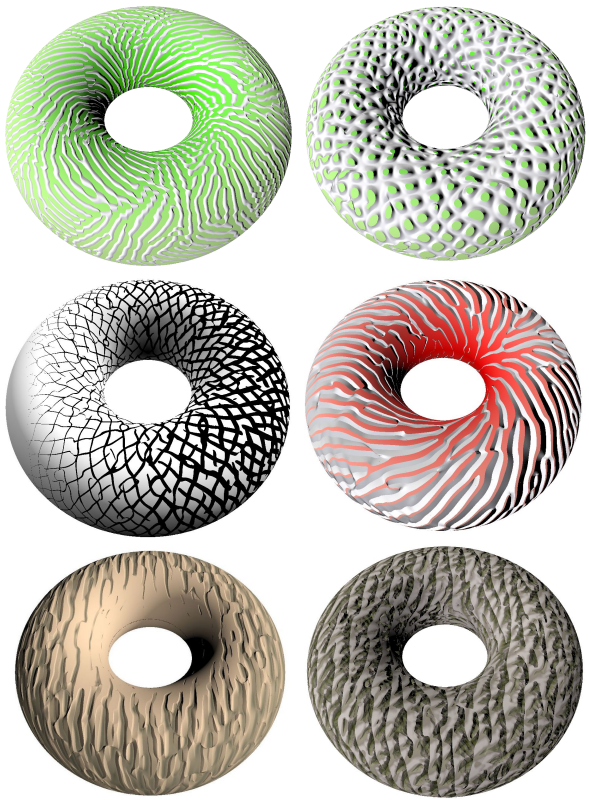


Fig. 20. **Top:** Left torus is covered by ridges produced with a locally coherent isoptropic phasor noise and a triangular profile; right torus is covered by a grip pattern, obtained as the product of two correlated (same kernels distribution), orthogonal phasor sine waves. **Middle:** Cracked patterns. The left texture is obtained by crossing two phasor noises using PWM patterns of varying width. The right texture is obtained from a single triangular profile, clamped at a varying height. **Bottom:** Bark patterns. These barks combine several phasor noises with triangular and sawtooth profiles to achieve the final appearance, with varying degrees of isotropy.

optimization-based parameterization methods for generating stripe patterns [Knöppel et al. 2015].

Figure 21 illustrates how the oscillation's profile can directly mimic some naturally occurring phenomena such as the asymmetric sand ripples along desert dunes.

Finally, Figure 20 shows a variety of textures such as cracked paints and tree barks applied to tori. These texture combine several layers of phasor noises using rectangular and sawtooth profiles.

4.3 Multi-material laminates

Multi-material laminates are typically made of two sandwiched materials, one more rigid than the other. Such laminates are of high interest for additive manufacturing, as they produce orthotropic elastic behaviors [Martínez et al. 2017]: the part is more flexible orthogonally to the laminate than in the other two directions. Ideally, for maximal design flexibility, the laminates should be freely orientable in space, and the proportions between both materials should be precisely controlled.

Phasor noise provides such flexibility through PWM profiles. Materials A and B are directly mapped to the up/down parts of the profile. This affords for a direct control of the orientation and material proportions. Figure 22 illustrates a cross shaped plate where we can observe that the laminate pattern triggers different bending behaviors in the extremities. The Figure also shows a miniature 3D shoe insole with a spatially varying 3D laminate pattern. Both models are driven by control fields allowing to paint orientation, mixture ratios and degree of isotropy.

Note that this could easily accommodate for more than two materials by creating a profile with multiple steps.

5 LIMITATIONS

In the bi-lobe case the noise is overall well behaved. Point singularities are acceptable, however we would like to reduce the impact of ridges between them. These produce waves in opposition and 'chopping' of the sine waves. Now that we have exposed the phasor field underlying the noise, it would be interesting to consider how to filter singularities. However, doing this in a procedural manner seems difficult.

Some undesirable interactions exist when mixing kernels using very different directions, with sine waves being abruptly cut off. Controlling the underlying direction field to obtain a locally coherent noise helps greatly and in practice, most interesting profiles have themselves high frequencies and mask these issues. However, ultimately we would like to reduce or remove these defects.

6 CONCLUDING REMARKS

There are links between phasor noise and the work of Neyret and Heitz [2016]. In particular, they introduce the notion of a *normalized* Gabor noise, which is filtered to suffer no local loss of contrast. This actually defines a full family of noises, as the normalization embeds a notion of window size (the smallest window resulting in a zero-thresholded noise — a noise normalized at all scales and hence with a square profile). Our phasor noise is normalized but has no notion of normalization scale. The normalization is an intrinsic property of its definition.

Having a window size to chose from offers additional controls to the user, which is often desirable for texturing applications. One benefit of our approach, however, is that no a-posteriori filtering is required, while the filtering approaches in [Neyret and Heitz 2016] involve either a local averaging through super-sampling, or an iterative Fast Fourier Transform algorithm. Finally, the ability to reshape the noise oscillation's profiles is quite unique to our method.

In a sense, the phasor sine wave is a generalized sine wave, which scale and orientations can be freely controlled in space. This leads to the possibility of performing additive synthesis to generate various profiles. We believe this is a novel tool in procedural synthesis that will lead to further developments. In particular, procedural noise is typically used to add fine scale unstructured details [Guingo et al. 2017]. Phasor noise paves the way to the ability to synthesize increasingly structured patterns from purely procedural synthesizers.

A remarkable property of phasor noise is that it is purely analytical. Even in the case of using a PWM there is no thresholding: the result is obtained as a weighted sum of harmonics. This opens interesting venues of future work, such as optimizing the pattern profile directly under specific objectives (elasticity, appearance).

ACKNOWLEDGMENTS

This work was supported partly by the french PIA project « Lorraine Université d'Excellence » (ANR-15-IDEX-04-LUE), by the ANR MuFFin (ANR-17-CE10-0002), and the project IMPRIMA(ANR-18-CE46-0004). We thank Samuel Hornus for proofreading.

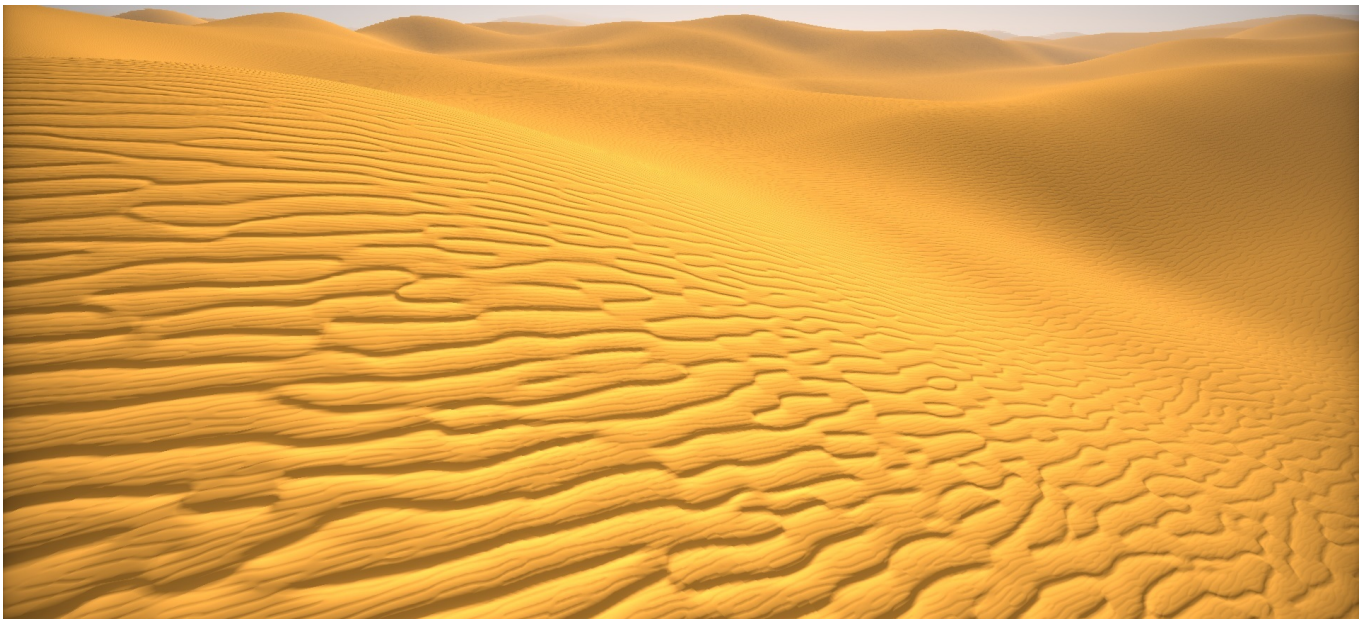


Fig. 21. A desert scene. The sand surface pattern is a phasor noise fed into a sawtooth profile. A second higher frequency phasor noise adds details to the surface. The color is also modulated by the profile.

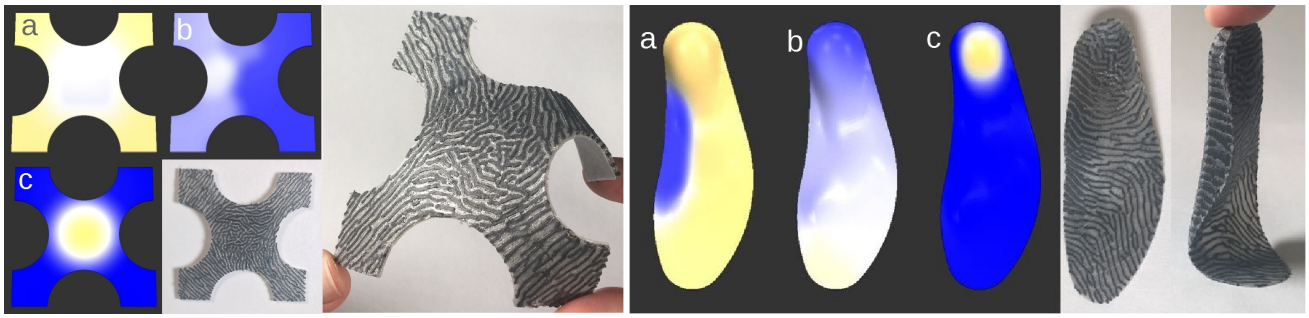


Fig. 22. **Left:** Cross shaped plate patterned with a phasor noise through a PWM profile. Printed using two materials (one flexible, one rigid). The user controlled fields are shown to the side: **a** ratios of material, **b** orientation and **c** isotropy. The patterns vary across the plate. **Right:** Shoe insole using a multimaterial pattern procedural synthesized with phasor noise. Both objects are printed on an Ultimaker 3.

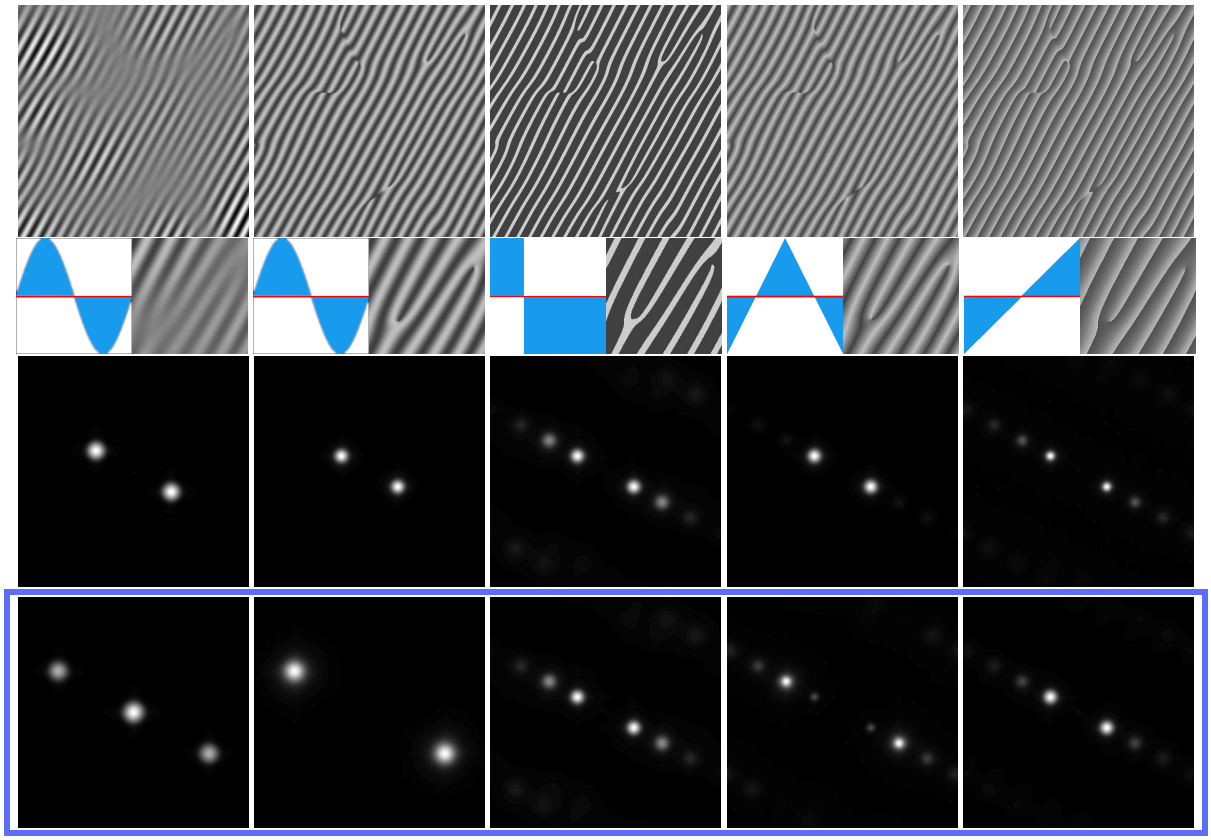


Fig. 23. Profile synthesis using phasor noise, and effect on the spectrum and spectrum of variance. **Top row**: Noise patterns. **Middle row**: Noise spectrum. **Bottom row**: Noise spectrum of variance. From **Left** to **Rigth**: Gabor noise followed by phasor noises with sine, square, triangular and sawtooth profiles.

REFERENCES

David S Ebert, F Kenton Musgrave, Darwyn Peachey, Ken Perlin, and Steven Worley. 2003. *Texturing & modeling: a procedural approach.* Morgan Kaufmann.

Bruno Galerne, Ares Lagae, Sylvain Lefebvre, and George Drettakis. 2012. Gabor Noise by Example. *ACM Trans. Graph.* 31, 4, Article 73 (July 2012), 9 pages. https://doi.org/10.1145/2185520.2185569

Guillaume Gilet, Jean-Michel Dischler, and Luc Soler. 2010. Procedural Descriptions of Anisotropic Noisy Textures by Example. In *Eurographics 2010 - Short Papers.* The Eurographics Association. https://doi.org/10.2312/egsh.20101052

Guillaume Gilet, Basile Sauvage, Kenneth Vanhoey, Jean-Michel Dischler, and Djamchid Ghazanfarpour. 2014. Local Random-phase Noise for Procedural Texturing. *ACM Trans. Graph.* 33, 6, Article 195 (Nov. 2014), 11 pages. https://doi.org/10.1145/2661229.2661249

Geoffrey Guingo, Basile Sauvage, Jean-Michel Dischler, and Marie-Paule Cani. 2017. Bi-Layer textures: a Model for Synthesis and Deformation of Composite Textures. *Computer Graphics Forum* 36, 4 (2017), 111–122. https://doi.org/10.1111/cgf.13229

Felix Knöppel, Keenan Crane, Ulrich Pinkall, and Peter Schröder. 2015. Stripe Patterns on Surfaces. *ACM Transactions on Graphics* 34 (2015). Issue 4.

Ares Lagae and George Drettakis. 2011. Filtering Solid Gabor Noise. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2011)* 30, 4 (July 2011), 51:1–51:6. https://doi.org/10.1145/1964921.1964946

Ares Lagae, Sylvain Lefebvre, Rob Cook, Tony Derose, George Drettakis, David S. Ebert, J.P. Lewis, Ken Perlin, and Matthias Zwicker. 2010. A Survey of Procedural Noise Functions. *Computer Graphics Forum* 29, 8 (2010), 2579–2600. https://doi.org/10.1111/j.1467-8659.2010.01827.x

Ares Lagae, Sylvain Lefebvre, George Drettakis, and Philip Dutré. 2009. Procedural Noise using Sparse Gabor Convolution. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2009)* 28, 3 (July 2009), 54–64. https://doi.org/10.1145/1531326.1531360

Ares Lagae, Sylvain Lefebvre, and Philip Dutré. 2011. Improving Gabor noise. *IEEE Transactions on Visualization and Computer Graphics* (2011). http://www-sop.inria.fr/reves/Basilic/2011/LLD11

John-Peter Lewis. 1984. Texture Synthesis for Digital Painting. In *Proceedings of the 11th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '84).* ACM, New York, NY, USA, 245–252. https://doi.org/10.1145/800031.808605

J P Lewis. 1986. Methods for Stochastic Spectral Synthesis. In *Proceedings on Graphics Interface '86/Vision Interface '86.* Canadian Information Processing Society, Toronto, Ont., Canada, Canada, 173–179. http://dl.acm.org/citation.cfm?id=16564.16594

Marco Livesu, Stefano Ellero, Jonàs Martínez, Sylvain Lefebvre, and Marco Attene. 2017. From 3D Models to 3D Prints: An Overview of the Processing Pipeline. *Computer Graphics Forum* 36 (2017), 537–564.

Jonàs Martínez, Haichuan Song, Jérémie Dumas, and Sylvain Lefebvre. 2017. Orthotropic K-nearest Foams for Additive Manufacturing. *ACM Trans. Graph.* 36, 4, Article 121 (July 2017), 12 pages. https://doi.org/10.1145/3072959.3073638

Fabrice Neyret and Eric Heitz. 2016. *Understanding and controlling contrast oscillations in stochastic texture algorithms using Spectrum of Variance.* Research Report. LJK / Grenoble University - INRIA. 8 pages. https://hal.inria.fr/hal-01349134

Vincent Tavernier, Fabrice Neyret, Romain Vergne, and Joëlle Thollot. 2019. Making Gabor Noise Fast and Normalized. In *EG 2019 - Short Papers.* The Eurographics Association. https://hal.inria.fr/hal-02104389

Kiril Vidimce, Alexandre Kaspar, Ye Wang, and Wojciech Matusik. 2016. Foundry: Hierarchical material design for multi-material fabrication. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology.* ACM, 563–574.

Kiril Vidimče, Szu-Po Wang, Jonathan Ragan-Kelley, and Wojciech Matusik. 2013. OpenFab: a programmable pipeline for multi-material fabrication. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 136.