2013

Simulation of Textured Audio Harmonics Using Random Fractal Phaselets

Jonathan Blackledge
Technological University Dublin, jonathan.blackledge@tudublin.ie

Derry Fitzgerald
Technological University Dublin, derry.fitzgerald@tudublin.ie

Ruairi Hickson
Technological University Dublin

Recommended Citation

Blackledge, J., Fitzgerald, D., Hickson, R.:Simulation of Textured Audio Harmonics Using Random Fractal Phaselets. Digital Audio Effects (DAFx2013),NUI Maynooth, Ireland, September, 2-6, 2013. - Accepted for Publication, pages:1 - 8.

SIMULATION OF TEXTURED AUDIO HARMONICS USING RANDOM FRACTAL PHASELETS

J Blackledge, *

Audio Research Group
Dublin Institute of Technology
jonathan.blackledge@dit.ie

D Fitzgerald, †

Audio Research Group
Dublin Institute of Technology
derry.fitzgerald@dit.ie

R Hickson,‡

Audio Research Group
Dublin Institute of Technology
notruairi@gmail.com

ABSTRACT

We present a method of simulating audio signals using the principles of random fractal geometry which, in the context of this paper, is concerned with the analysis of statistically self-affine 'phaselets'. The approach is used to generate audio signals that are characterised by texture and timbre through the Fractal Dimension such as those associated with bowed stringed instruments. The paper provides a short overview on potential simulation methods using Artificial Neural Networks and Evolutionary Computing and on the problems associated with using a deterministic approach based on solutions to the acoustic wave equation. This serves to quantify the origins of the 'noise' associated with multiple scattering events that often characterises texture and timbre in an audio signal. We then explore a method to compute the phaselet of a phase signal which is the primary phase function from which a phase signal is, to a good approximation, a periodic replica and show that, by modelling the phaselet as a random fractal signal, it can be characterised by the Fractal Dimension. The Fractal Dimension is then used to synthesise a phaselet from which the phase function is computed through multiple concatenations of the phaselet. The paper provides details of the principal steps associated with the method considered and examines some example results, providing a URL to m-coded functions for interested readers to repeat the results obtained and develop the algorithms further.

1. INTRODUCTION

The digital simulation of audio signals produced by different musical instruments is a standard problem with many example solutions having been considered and implemented over many decades e.g. [1] and [2]. However, these successes tend to be related to the simulation of instruments that are monotonic in the sense that the sound field they generate is the result of a resonance with a relatively small band-width. This includes instruments such as the trumpet, trombone, clarinet, flute, the forte-piano and so on. The simulation of primarily bowed string instruments such as the violin, viola and cello relies on sound archives and libraries such as those available from the Vienna Symphonic Library [3] and the algorithmic simulation of solo stringed instruments remains an elusive problem to-date. This is because the sound field these instuments produce is the result of a resonance involving complex interactions (multiple reflections) of the sound waves inside a resonator with a relatively complex topology. The result of this is to produce a sound source that, compared with many other instruments, may be loosely classified by the term 'texture'. This is reflected in the onomatopoeic terms used to describe the violin, for example, in the slavonic languages, i.e. *Skrzypce* in Polish and *Skripka* in Russian, both words being synonymous with the English phrase, 'to scrape'.

Texture and Timbre is fundamental to playing stringed instruments [4] and can significantly differentiate one individual player from another which is often based on the traditions of the 'School' in which they have been trained, e.g. the tone of Jasha Heifetz (with its highly textured and 'gritty' timbre) [5] verses that of Nathan Milstein known as the 'man with the silver bow' (a phrase used to describe the silky transparency of a sound with a 'silver shade') [6]. The word associations used to describe the subtleties of an audio signal generated by a bowed stringed instrument are of little value with regard to the issue of how such a sound can be synthesised. This is especially true with regard to the algorithmic synthesis of a solo violin (as opposed to developing a library of samples associated with a string ensemble). There are a variety of potential approaches that can be used which are briefly discussed in the following sections.

1.1. Synthesis using Artificial Neural Networks

An Artificial Neural Network (ANN) aims, through iterative processes, to compute a set of optimal weights that determine the flow of information (the amplitude of a signal at a given node) through a network that simulates a simple output subject to a complex input. In this sense, an ANN simulates a high entropy input with the aim of transforming the result into a low entropy output. However, this process can be reversed to generate a high entropy output from a low entropy input. In this sense, a ANN can be used to generate a textured harmonic by simulating signals once it has been trained to do so. To use a ANN in this way, the audio engineer requires knowledge of the ANN algorithm and the weights that have been

[*] Stokes Professor, Science Foundation Ireland

[†] Stokes Lecturer, Science Foundation Ireland

[‡] Research Assistant, School of Electrical and Electronic Engineering

generated through the training process (i.e. the input of the signals used to generate the weights). For this purpose the type and architecture of the ANN that provides a best simulation is crucial [7].

1.2. Synthesis using Evolutionary Algorithms

Like artificial intelligence, evolutionary computing involves the process of continuous optimisation. However, using ANNs to generating textured harmonics can be time consuming and inflexible compared to using a formulaic approach to simulating sound textures via an iterated (nonlinear) function, for example, whose output can be filtered as required. To do this, an evolutionary algorithms approach is required in which a population-based, stochastic search engine is required that mimics natural selection. Due to their ability to find excellent solutions for conventionally difficult and dynamic problems within acceptable time, evolutionary algorithms have attracted interest from many areas of science and engineering [8]. Like the use of ANNs, the application of evolutionary algorithms to simulate textured sound fields lies beyond the scope of the paper and will be considered in a future publication. However, it can be considered as an extension of the noise filtering approach considered in this paper where the iterator used to generated random numbers coupled with a low-pass filter is replaced by a single nonlinear iteration function that has been evolved to simulate the audio signal directly.

1.3. Synthesis using Fractal Texture Analysis

In a very general context, we may expect a textured sound signal to be the result of filtered noise. As with all filtered noise models, the problem is to determine what type of noise and what type of filter provides the closest simulation to the sound. In this paper, we focus on a $1/\mid\omega\mid^q$ fractal noise model [9] where ω is the (angular) frequency and q is some exponent which is taken to characterise the real signal in some 'best fit' sense. This approach is consistent with self-affine stochastic systems theory. As discussed later on in the paper, although such a model is relatively trivial, it does provide surprisingly good results when applied to the simulation of the phaselet associated with a textured monotonic harmonic subject to parameter optimisation. This filter based model is easy to implement on any audio engineering and post-production platform and can be extended further to include a range of stochastic field models However, in this paper we focus on the application of the simplest filtered-noise based random fractal model on the understanding that interested readers can easily extend such a filter-based model. For this purpose, two m-code functions are given in Appendix A and Appendix B.

2. THE PHYSICAL ORIGINS OF TEXTURE IN SOUND SIGNALS

The synthesis of an acoustic field from complex sound sources such as stringed instruments is computationally intensive using a deterministic approach based on the application of conventional methods for modelling the propagation and scattering of sound waves. This is because the sound waves undergo many complex scattering interactions to generate a 'resonance' that is characteristic of a particular instrument such as violin, for example. This 'resonance' is not a simple one-dimensional standing wave pattern but a three-dimensional quasi-standing wave pattern that outputs a complex phase signal. By understanding the physical background to the problem, which is compounded in solutions to the wave equation, we can assess which physical aspect of the sound source is responsible for the complexity that we associated with the term texture [10]. This is the purpose for the material presented in the following section.

2.1. Model for a Simple Source

It is well known that the fundamental model for an acoustic field $u(\mathbf{r},t)$ (i.e. three-dimensional pressure waves) as a function of three-dimensional space $\mathbf{r} = \hat{\mathbf{x}}x + \hat{\mathbf{y}}y + \hat{\mathbf{z}}z$ and time t, generated by an acoustic source denoted by a 'source function' $f(\mathbf{r},t)$ is given by [11]

$$\left(\nabla^2 - \frac{1}{c_0^2}\frac{\partial^2}{\partial t^2}\right)u(\mathbf{r},t) = -f(\mathbf{r},t) \tag{1}$$

where ∇^2 is the Laplacian operator and c_0 is the (constant) acoustic wave speed. If we consider the source to be ideally localised in space so that $f(\mathbf{r},t) = \delta^3(\mathbf{r})f(t)$ then, with $U(\mathbf{r},\omega) = \mathcal{F}_1[u(\mathbf{r},t)]$ and $F(\omega) = \mathcal{F}_1[f(t)]$ where $\mathcal{F}_1$ denotes the one-dimensional Fourier transform operator, equation (1) can be written in terms of the wavenumber $k = \omega/c_0$ as

$$\left(\nabla^2 + k^2\right)U(\mathbf{r},k) = -\delta^3(\mathbf{r})F(k)$$

This equation has the general Green's function solution [1]

$$U(\mathbf{r},k) = g(r,k) \otimes_{\mathbf{r}} \delta^3(\mathbf{r})F(k) = g(r,k)F(k)$$

where

$$g(r,k) = \frac{\exp(ikr)}{4\pi r}, \quad r = \mid \mathbf{r} \mid$$

and $\otimes_{\mathbf{r}}$ denotes the convolution integral over $\mathbf{r}$. Inverse Fourier transforming

$$u(\mathbf{r},t) = \frac{f(t + r/c_0)}{4\pi r}$$

and the time variations of the sound field are simply related to the source function subject to an intensity $\mid u \mid^2$ determined by the inverse square law. Ignoring the scaling and translation effects by r (and c_0), the detected signal at some point in space $\mathbf{r}$ can be taken to be given by

$$s(t) = p(t) \otimes_t f(t)$$

where $\otimes_t$ denotes the convolution integral over time and p is the Impulse Response Function of the 'detector' (which may include the audio environment in which the source is placed). This result provides a standard linear time invariant model for an audio signal.

2.2. Model for a Complex Resonator

Acoustic fields are rarely generated by the direct propagation of a field from a source but through reflection or scattering of the field in a resonator. The scattering effects can characterise both the source itself (the local resonator) and the environment (e.g. a concert hall) in which the source is placed. The effects of a concert hall typical filter out the higher frequency components of the source. In either case, the scattering of an acoustic field can be taken to be generated by spatial variations in the wave speed so that the wave equation (1) now becomes

$$\left(\nabla^2 - \frac{1}{c^2(\mathbf{r})}\frac{\partial^2}{\partial t^2}\right)u(\mathbf{r},t) = -f(\mathbf{r},t)$$

With

$$\frac{1}{c^2(\mathbf{r})} = \frac{1}{c_0^2}[1+\gamma(r)]$$

and taking the Fourier transform with respect to time, we then obtain the inhomogeneous Helmholtz equation

$$\left(\nabla^2 + k^2\right) U(\mathbf{r},k) = -k^2\gamma(\mathbf{r})U(\mathbf{r},k) - F(\mathbf{r},k)$$

whose Green's function 'transformation' is given by [13], [14]

$$U(\mathbf{r},k) = U_i(\mathbf{r},k) + k^2 g(r,k) \otimes_{\mathbf{r}} \gamma(\mathbf{r})U(\mathbf{r},k) \qquad (2)$$

where

$$U_i(\mathbf{r},k) = g(r,k) \otimes_r F(\mathbf{r},k)$$

It is important to stress that, unlike the 'source problem' considered in the previous section, equation (2) is not a solution since the field we require a solution for is on both the left- and right-hand sides of the equation, i.e. equation (2) is transcendental. This issue is a principal problem with regard to computing exact solutions to the scattering problem in order to simulate the complex wave patterns generated by a scattered acoustic field. The 'formal approach' is to consider the 'iterative solution' [11]

$$U = U_i + k^2 g \otimes_{\mathbf{r}} \gamma U_i + k^4 g \otimes_{\mathbf{r}} \gamma(g \otimes_{\mathbf{r}} \gamma U_i) + ...$$

which converges provided $k^2\|\gamma\| < 1$. Each term in this series solution characterises the effect of increasingly higher order scattering effects, i.e. the total field is equal to the incident field plus single scattering processes plus second order scattering effects and so on. We consider the field pattern associated with the higher order terms to contribute to a 'noise function' N associated with the generation of a complex (multiple scattered) audio signal (through the 'resonator' of a string instrument, for example) and write

$$U(\mathbf{r},k) = U_i(\mathbf{r},k) + k^2 g(r,k) \otimes_{\mathbf{r}} \gamma(\mathbf{r})U_i(\mathbf{r},k) + N(\mathbf{r},k)$$

The second term in this equation describes weak scattering under the Born approximation [11] and the third term the texture which we are required to model. The sound signal detected at a point in space $\mathbf{r}$ is taken to be given by [12]

$$s(\mathbf{r},t) = p(t) \otimes_t u(\mathbf{r},t) = p(t) \otimes_t u_i(\mathbf{r},t)$$

$$-p(t) \otimes_t \frac{\partial^2}{\partial t^2} G(r,t) \otimes_{\mathbf{r}} \gamma(\mathbf{r}) \otimes_t u_i(\mathbf{r},t) + p(t) \otimes_t n(\mathbf{r},t)$$

and is the result of contributions from the initial source, single scattering events and the 'noise' generated by multiple scattering processes. In this sense, the development of a model for the (temporal) noise function $n(t)$ of a complex resonator changes from an approach based on a deterministic model involving a series solution to one that is based on a stochastic model. The problem is then to consider the most suitable stochastic model in terms of its statistical properties and spectral characteristics subject to an evaluation based on known audio signals.

2.3. Diffusion Based Model

Diffusion models are based on an understanding that multiple scattering effects can be taken to be analogous to the effect of diffusion via a random walk model. In this context, it is possible to show that the diffusion equation is a special case of the wave equation. Consider the three-dimensional homogeneous time dependent wave equation [13]

$$\left(\nabla^2 - \frac{1}{c_0^2}\frac{\partial^2}{\partial t^2}\right) u(\mathbf{r},t) = 0$$

and

$$u(x,y,z,t) = U(x,y,z,t)\exp(i\omega t)$$

where it is assumed that the field u varies significantly slowly in time compared with $\exp(i\omega t)$ and we note that

$$u^*(x,y,z,t) = U^*(x,y,z,t)\exp(-i\omega t)$$

is also a solution to the wave equation. Differentiating

$$\nabla^2 U = \exp(i\omega t)\nabla^2 u,$$

and

$$\frac{\partial^2}{\partial t^2}U = \exp(i\omega t)\left(\frac{\partial^2}{\partial t^2}u + 2i\omega\frac{\partial u}{\partial t} - \omega^2 u\right)$$

$$\simeq \exp(i\omega t)\left(2i\omega\frac{\partial u}{\partial t} - \omega^2 u\right)$$

when

$$\left|\frac{\partial^2 u}{\partial t^2}\right| << 2\omega\left|\frac{\partial u}{\partial t}\right|.$$

Under this condition, the wave equation reduces to

$$(\nabla^2 + k^2)u = \frac{2ik}{c_0}\frac{\partial u}{\partial t}$$

where $k = \omega/c_0$. However, since u^* is also a solution,

$$(\nabla^2 + k^2)u^* = -\frac{2ik}{c_0}\frac{\partial u^*}{\partial t}$$

and thus,

$$u^*\nabla^2 u - u\nabla^2 u^* = \frac{2ik}{c_0}\left(u^*\frac{\partial u}{\partial t} + u\frac{\partial u^*}{\partial t}\right)$$

which can be written in the form

$$\nabla^2 I - 2\nabla\cdot(u\nabla u^*) = \frac{2ik}{c_0}\frac{\partial I}{\partial t}$$

where $I = uu^* =| u |^2$. Let u be given by

$$u(\mathbf{r},t) = A(\mathbf{r},t)\exp(ik\hat{\mathbf{n}}\cdot\mathbf{r})$$

where $\hat{\mathbf{n}}$ is a unit vector and A is the amplitude function. Differentiating, and noting that $I = A^2$, we obtain

$$\hat{\mathbf{n}}\cdot\nabla A = \frac{2}{c_0}\frac{\partial A}{\partial t}$$

or

$$\left(\frac{\partial}{\partial x} + \frac{\partial}{\partial y} + \frac{\partial}{\partial z}\right)A(x,y,z,t) = \frac{2}{c_0}\frac{\partial}{\partial t}A(x,y,z,t)$$

which is the unconditional continuity equation for the amplitude A of a wavefield

$$u(\mathbf{r},t) = A(\mathbf{r},t)\exp[i(k\hat{\mathbf{n}}\cdot\mathbf{r} + \omega t)]$$

where A varies slowly with time.

The equation

$$\nabla^2 I - 2\nabla\cdot(u\nabla u^*) = \frac{2ik}{c_0}\frac{\partial I}{\partial t}$$

is valid for $k = k_0 - i\kappa$ (i.e. $\omega = \omega_0 - i\kappa c_0$) and so, by equating the real and imaginary parts, we have

$$D\nabla^2 I + 2\text{Re}[\nabla\cdot(u\nabla u^*)] = \frac{\partial I}{\partial t}$$

and

$$\text{Im}[\nabla\cdot(u\nabla u^*)] = -\frac{k_0}{c_0}\frac{\partial I}{\partial t}$$

respectively, where $D = c_0/2\kappa$, so that under the condition

$$\text{Re}[\nabla\cdot(u\nabla u^*)] = 0$$

we obtain

$$D\nabla^2 I = \frac{\partial I}{\partial t}.$$

This is the diffusion equation for the intensity of sound I. The condition required to obtain this result can be justified by applying a boundary condition on the surface S of a volume V over which the equation is taken to conform. Using the divergence theorem

$$\text{Re}\int\limits_V \nabla\cdot(u\nabla u^*)d^3\mathbf{r} = \text{Re}\oint\limits_S u\nabla u^*\cdot\hat{\mathbf{n}}d^2\mathbf{r} = 0$$

under the homogenous boundary condition

$$u = 0,\quad \nabla u^* = 0\ \ \forall\ \ \mathbf{r}\in S$$

Physically, diffusion based models consider the multiple scattering of an acoustic wavefield to be analogous to a random walk with a uniformly distributed phase.

If we consider the diffusion equation for a source $f(\mathbf{r},t)$ to be given by

$$\left(\nabla^2 - \sigma\frac{\partial}{\partial t}\right)I(\mathbf{r},t) = -f(\mathbf{r},t),\quad I_0(\mathbf{r}) = I(\mathbf{r},t=0)$$

where $\sigma = D^{-1}$ and f is of compact support, the Green's function solution to this equation for homogenous boundary conditions is given by [1]

$$I(\mathbf{r},t) = f(\mathbf{r},t)\otimes_{\mathbf{r}}\otimes_t G(r,t) + \sigma I_0(\mathbf{r})\otimes_{\mathbf{r}} G(r,t)$$

where

$$G(r,t) = \frac{1}{\sigma}\left(\frac{\sigma}{4\pi t}\right)^{\frac{3}{2}}\exp\left[-\left(\frac{\sigma r^2}{4t}\right)\right],\quad t>0$$

which satisfies the property:

$$\int_{-\infty}^{\infty} G(r,t)d^3\mathbf{r} = \frac{1}{\sigma},\quad t>0$$

The terms involved in the equation for the sound intensity are convolutions of the Green's function with the source function and the initial condition $I_0(\mathbf{r})$ respectively. Thus, for a localised source $f(\mathbf{r},t) = \delta^3 f(t)$ and with $I_0 = 0$, the signal recorded a some point in space $\mathbf{r}$ is given by

$$s(\mathbf{r},t) = p(t)\otimes_t f(t)\otimes_t G(r,t)$$

Classically diffusive systems are thus characterised by an Impulse Response Function that is a Gaussian, a scenario that is compatible with the purpose of an anechoic chamber (i.e. a room designed to completely absorb reflections of sound) but incompatible with the concept of a resonator.

2.4. Fractal Noise Model

The models considered in the previous sections reveal some major problems with regard to using a deterministic approach to simulating an audio signal generated by an instrument and/or an environment whose audio response is the result of multiple scattering effects. These problems include: (i) the three-dimensional nature of the models used, that, while of physical significance, are incompatible with the direct simulation of a time signature using a linear filtering approach; (ii) diffusion based models assume a uniform phase distribution for the scattering of sound which is rarely the case, especially with regard to instruments that are specifically designed to generate complex resonance effects; (iii) multiple scattering models 'naturally iterative' (as should be expected on a causal basis, i.e. an $(n+1)^{\text{th}}$ order reflection can only occur after a n^{th} order event) and therefore require complex simulations to be undertaken in light of point (i) above. Thus, given the inadequacy of the simple sound-source model, the complexity associated with the multiple scattering model, the incompatibility associated with a fully diffusive model and the incompatibility of using fully three-dimensional models to simulate time-dependent signals, we consider an approach to modelling the texture associated with multiple scattering which is based on the one-dimensional fractional diffusion equation for a localised source given by

$$\left(\frac{\partial^2}{\partial x^2} - \frac{\partial^q}{\partial t^q}\right)u(x,t) = \delta(x)w(t)$$

where $w(t)$ is white noise (with a uniformly distributed power spectrum). This phenomenological equation represents the intermediate case between a wave model $q=2$ and a classical diffusion model $q=1$ in a one-dimensional sense, and, moreover, has the self-affine temporal solution [14]

$$u(t) = \frac{1}{\Gamma(q)t^{1-q/2}}\otimes_t w(t),\quad t>0 \tag{3}$$

where $q = 5-2D$ and $D\in[1,2]$ is the Fractal Dimension. A fundamental property of this solution is that it is characterised by a Power Spectral Density Function $\hat{P}(\omega)$ given by [12]

$$\log\hat{P}(\omega) = C - q\log\omega,\quad \omega>0 \tag{4}$$

where C is a real scaling constant and the remaining focus of this paper is the use of equation (4) to determine the Fractal Dimension of a phaselet and equation (3) to simulate a phaselet and thereby, harmonic texture.

3. FRACTAL PHASELET MODEL

A phaselet is the smallest phase signature of a single harmonic. If a perfect harmonic is taken to have a simple linear phase function that can be cut into a sequence of smaller linear phase functions concatenated together, then the phase of a textured harmonic is taken to be the concatenation of a many phaselets of compact support $t = (0,T]$. Each phaselet is computed using equation (3), subject to a given fractal dimension that changes the texture of the output. Thus, we consider a fractal phaselet to be given by

$$\theta(t) = \frac{1}{\Gamma(q)t^{1-q/2}}\otimes_t w(t),\quad t\in[0,T] \tag{5}$$

The phase function $\Theta(t)$ is then given by the N^{th} concatenation of replicas of $\theta(t)$, i.e.

$$\Theta(t) = \mathop{\Big\|}_{j=1}^{N} \theta_j(t) \equiv \theta_1(t) \parallel \theta_2(t) \parallel \cdots \parallel \theta_N(t) \qquad (6)$$

and the (complex) signal given by

$$s(t) = \exp(i[\omega_0 t + \Theta(t)])$$

whose real (or imaginary) component is used to compute the audio harmonic.

3.1. Phase Evaluation of a Harmonic Audio Signal

To demonstrate the nature of a phaselet for a textured harmonic, we consider the computation of the phase function for a concert A (44.1kHz) generated by a violin. Figure 1 shows the unwrapped and de-trended phase function for the signal together with a smaller window of the signal illustrating a series of replica phaselets associated with the primary resonance phenomena. The macro-trends (with both positive and negative gradient) associated with this data are generated by the vibrato which is quasi-periodic but the phaselets from which the entire phase signal is composed (and as illustrated in Figure 1) are the basis functions for the texture of the sound generated.

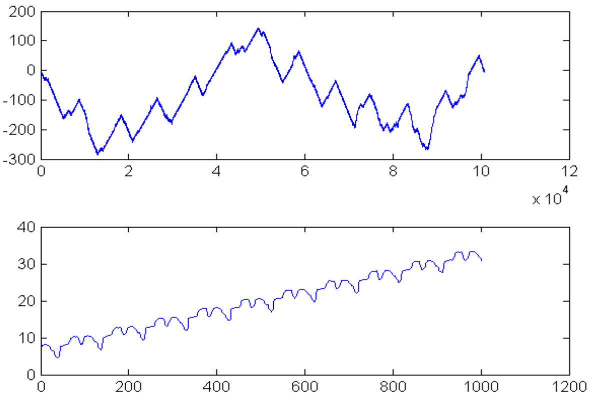

Figure 1: *Unwrapped and de-trended phase for a concert A recorded from a violin (above) and a windowed sample of the signal (below) showing a series of replica phaselets associated with the primary resonance characteristics of the instrument.*

The phase function shown in Figure 1 is the perturbation to an otherwise near-linear function that must be extracted from the (unwrapped) phase signal. This is easily accomplished by evaluating the gradient of the phase ω_0 and computing the (de-trended) phase function

$$\theta(t) = \Theta(t) - \omega_0 t$$

the gradient being obtained by applying a least square fit to the data (when the unwrapped phase is quasi-linear) or, for a quasi-harmonic phrase function, via the equation

$$\omega_0 = \frac{\Theta(\tau) - \Theta(0)}{\tau}$$

where τ defines the window of time over which the un-wrapped phase is considered.

3.2. Phaselet Identification

With regard to automating the period of a phaselet and thereby identifying the primary resonance signature, this can be undertaken by autocorrelating the de-trended unwrapped phase which is composed of a sequence of ‘spikes’ whose width represents the periodicity of the phaselet. By locating the positions of the zero crossings in this autocorrelation function and rounding the average distance between them (for one half of the autocorrelation function) an estimate of the phaselet periodicity can be obtained from which a single phaselet can then be extracted. However, this approach assumes the absence of vibrato, which, as shown in Figure 1, generates a distortion in terms of quasi-periodic trends. To over come this problem, the de-trended unwrapped phase is first differentiated (using a simple forward differencing scheme) to remove the isolated trending effects caused by vibrato. The result of this computation is conveyed in Figure 2 which shows a single isolated phaselet together with the associated log-log power spectrum (for the first 50% of the positive half-space data) that is clearly characteristic of the scaling relationship compounded in equation (4), i.e. a linear relationship between the logarithm of the power spectrum and the logarithm of the frequency with a negative gradient characterised by q.

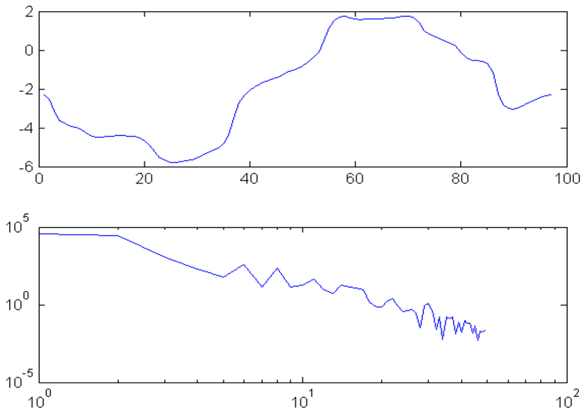

Figure 2: *Isolated phaselet (above) and the associated log-log power spectrum (below).*

3.3. Computation of the Fractal Dimension of a Phaselet

Numerical methods for computing the Fractal Dimension of a random scaling fractal are well known. The power spectrum method is particularly popular and involves the generation of a best fit estimation based on the scaling model associated with equation (4) and thereby obtaining an estimate for q (and C). Application of a least squares method for estimating q, which is based on minimising the error function

$$e(q, C) = \| \log P(\omega) - \log \hat{P}(\omega, q, C) \|_2^2$$

where $P(\omega)$ is the input power spectrum of the phaselet, generally leads to errors in estimates for q. The reason for this is that the application of a least squares approach is very sensitive to statistical heterogeneity and may therefore provide values of the Fractal Dimension that are not compatible with the rationale associated with the model. For this reason, Orthogonal Linear Regression (OLR) is used to compute an estimate of q based on the algorithm available at [15]. In Appendix A an m-code function is provided to compute the Fractal Dimension of a phaselet for a single textured harmonic input from a (mono) .wav file. Note that this function is only suitable for applications to single harmonic inputs alone. Its sole purpose is to identify the fractal dimension of the phaselet associated with a single harmonic characterised by texture, the Fractal Dimension being a measure of this texture. The function uses the OLR algorithm given by function OLR and in the case of the phaselet shown in Figure 2 yields a Fractal Dimension of 1.3945.

4. SIMULATION OF A PHASELET USING RANDOM FRACTALS

In principle, once the Fractal Dimension has been estimated for a phaselet of a harmonic texture, the phaselet can be simulated based on the application of equation (5), in particular, using the Fourier filtering operation

$$\theta(t,q,T) = \text{Re}\mathcal{F}_1^{-1}\left(\frac{[W(\omega)]}{\mid \omega \mid^q}\right), \quad \omega > 0, \quad t \in [0,T] \qquad (7)$$

where $w(t)$ is generated using a Pseudo Random Number Generator and where the phaselet may now depend upon the input Fractal Dimension and the period T (user parameters). The phase function is then computed using equation (6) which depends upon another user defined parameter N. An example m-code function for simulating a harmonic texture using this approach is provide in Appendix B which depends upon user parameter D, T, N. The value of N determines the length of the signal and the parameter T determines the pitch subject to the sampling rate chosen to output an audio signal. However, in the context of the remit for this paper, the 'key parameter' is D which changes the texture of the sound depending upon the value used subject to the pitch being considered.

4.1. Incorporation of a Deterministic Sound Source

To simulate an audio signal we need to include a deterministic source term. This can be achieved by replacing the stochastic source function $w(t)$ as given in equation (5) with a hybrid term that consists of two components including a deterministic source function $f(t)$, i.e.

$$w(t) := (1-r)w(t) + rf(t), \quad r \in [0,1]$$

where $\|n(t)\|_\infty = 1$ and $\|f(t)\|_\infty = 1$. Here, r defines the relative 'strength' of each term in relaton to its contribution to the solution of

$$\left(\frac{\partial^2}{\partial x^2} - \frac{\partial^q}{\partial t^q}\right) u(x,t) = \delta(x)[(1-r)w(t) + rf(t)]$$

which models a signal now determined by the equation

$$s(t) = p(t) \otimes_t [(1-r)w(t) + rf(t)] \otimes_t \frac{1}{\Gamma(q)t^{1-q/2}}$$

Note that, for $r = 1$ and $q = 2$ we obtain a signal that is, in effect the same as that obtained under the 'simple source model' presented in Section 2.1, i.e. $s(t) = p(t) \otimes_t f(t)$.

5. DISCUSSION

To quote B. Mandelbrot, 'fractal geometry is an intrinsic study of texture' and in this paper, we have developed a model for simulating texture in audio signals based on random scaling fractals. However, the model has been introduced in a way that attempts to relate it to the problem of evaluating multiple scattering effects that occur in instruments such as the violin when a simple sound-source model is entirely inadequate and a deterministic approach is not practically realisable.

The principal original contribution we have made to the field of digital audio effects has been to illustrate how a fractional diffusion equation can be used to model a digital audio signal that is taken to be generated from a complex resonance. This approach has specific applications in the simulation of audio signals generated by instruments such as the violin which is notoriously difficult to synthesise because of the complex acoustic scattering processes that occur in this instrument. The primary controlling parameter is the Fractal Dimension D. This parameter can be used to characterise the phaselet of audio signals using the algorithm presented in Appendix A which is then used to synthesise signals based on the application of equation (7).

The purpose of this paper has been to introduce a fractal based model for the synthesis of audio signals using a texture associated with the generation of sound fields from instruments that generate complex resonance patterns due to multiple scattering effects. This has been considered through a study of conventional scattering theory which is too computationally difficult to use effectively for audio simulation and a classical diffusion based approach which produces a model that is 'phase limited'. A fractional diffusion approach has therefore been considered in which each scattering process is considered to be a random walk with a directional bias to the phase (which is determined by the Fractal Dimension). This is the essential link between attempting to model a multiple scattered sound field and using random scaling fractal signals to model audio texture produced from instruments such as a violin.

6. CONCLUDING REMARKS

With regard to the simulation method proposed, the audio signal is still a relatively poor representation of a stringed instrument. First, no vibrato is considered and second, the harmonic characteristics of the sound are weak. With regard to vibrato, it is observed that the quasi-regular trending behaviour of the unwrapped phase function given in Figure 1 resembles a deterministic fractal structure consistent with a Sierpinski triangle, for example. Thus application of deterministic fractals for simulating vibrato may provide a route to modelling this quasi-regular effect.

The use of ANNs and evolutionary algorithms to simulate sound textures as briefly discussed earlier in this paper is the subject of future work and lies beyond the scope of this paper. However, it is noted that the evolution of nonlinear functions for the generation of phase signals of the type given in Figure 1 is practically viable within the context of applications packages such as *Eureqa* [8], for example, especially if an output can be configured, through application of an iterated function, for sound simulation over a range of frequencies. In contrast, although ANNs provide the potential for a more subtle demarcation of a sound textures, the result is predicated on training data which makes the approach time and data demanding.

The simplest and most cost effective approach is to further develop the texture synthesis based on the approach considered here which has introduced only the most basic method for fractal sound synthesis using a simple $1/\omega^q$ model. This model can be further improved to incorporate generalised self-affine models and multifractal methods coupled with a more in-depth analysis of stochastic time series applied to sound textures using, for example, the Ornstein-Uhlenbeck process [16]. Irrespective of the stochastic model assumed, the identification and regeneration of a phaselet for this purpose appears to be crucial. Thus, another approach is to categorise these functions, thereby developing a library of phaselets that can be concatenated for sound simulation. In this context, the use of evolutionary algorithms could prove to be advantageous. However, in the context of the approach considered in this paper, there would appear to be value in further exploring the role of random fractal geometry and fractal time random walks [17] and stochastic field theory focusing on the theory and applications of fractional dynamics (e.g. [18] and [19]) for the purpose of simulating textured audio signals.

Appendix A: m-Code for Computing the Fractal Dimension of a Phaselet

The following code has been written to accommodate a two-column format and consequently uses the MATLAB continuation syntax ‘...’. The code is somewhat condensed to minimize space.

```
function D=FDPhaselet
%Function to compute Fractal Dimension D
%for a phaselet extracted from a textured
% harmonic.
%START:
%Read signal from mono .wav file.
signal=wavread(’newvio.wav’);
signal=signal./max(abs(signal));%Normalise.
%Compute Hilbert transform of signal &
%its angle and write unwrapped phase to
%array ’phase’.
phase=unwrap(angle(hilbert(signal)));
%Compute gradient of unwrapped phase
grad=(phase(length(phase))-phase(1));
grad=grad/length(phase);
%and de-trend
for i=1:length(phase)
phase(i)=phase(i)-grad*i;
end
%Compute gradient of phase
for i=1:length(phase)-1
dphase(i)=phase(i+1)-phase(i);
end
dphase(length(phase))=...
dphase(length(phase)-1);
dphase=dphase’;
%Compute autocorrelation function
autocor=xcorr(dphase,dphase);
autocor=autocor./max(autocor);
%Compute zero crossings metric to
%extract a phaselet.
k=1;
zero(k)=0;%initialize
%only fist half of autocor required
for i=1:round(size(autocor)/2)
%find a zero crossing and repeat process
  if autocor(i)>0.0 & autocor(i+1)<=0.0
     zero(k)=i; k=k+1;
  end
end
%Compute distance between zero crossings
for i=1:round(length(zero))-1
     zc(i)=zero(i+1)-zero(i);
end
%Compute the (integer) average of the
%distanec between the zero crossings.
avzc=round(sum(zc));
avzc=round(avzc/(length(zero)-1));
%extract phaselet from phase array centre
phaselet=phase(round(length(phase)/2)...
-round(avzc/2):round(length(phase)/2)...
+round(avzc/2));
%Compute Power Spectrum of phaselet
powspec=abs(fft(phaselet)).^2;
%Extract data for OLR estimation
%(50% of positive half space of spectrum).
for i=1:round(size(powspec)/2)
    if powspec(i)>0
       ydata(i)=log(powspec(i));
       xdata(i)=log(i);
    end
end
%Compute Fractal Dimension using OLR.
 p=OLR(xdata,ydata);
 q=-p(2)/2; D=(5-q)/2;
 %Output of function is D.
%FINISH
%
%-------------------------
%
%OLR Function
%
function x=OLR(xdata,ydata)
fun=inline(’sum((p(1)+p(2)*xdata-ydata).^2)...
/(1+p(2)^2)’,’p’,’xdata’,’ydata’);
x0=flipdim(polyfit(xdata,ydata,1),2);
options=optimset(’TolX’,1e-6,’TolFun’,1e-6);
x=fminsearch(fun,x0,options,xdata,ydata);
```

Appendix B: m-Code for Simulating a Fractal Textured Harmonic

The following code has been written to accommodate a two-column format and consequently uses the MATLAB continuation syntax ‘...’. The code is somewhat condensed to minimize space.

```
function Sig=FHTS(D,T,N)
%Example Parameters:
%D=1.2; T=100; N=1000;
%Function to simulate harmonic texture
```

```
%Fractal Harmonic Texture Simulator 'FGTS'.
%INPUTS: D-Fractal Diomension
%        T-Length of Phaselet
%        N-Number of cancatinations
%        =length of phase signal
%OUTPUT: Sig-Simulated signal.
q=(5-2*D);%Comput q from D
%Compute noise using function rand.
w=rand(1,T);
%Comput complex spectrum of noise
%with shift.
w=fftshift(fft(w));
%Filter the spectrum.
for i=1:T
    filter(i)=abs(i-round(T/2))^(q/2);
    if filter(i)==0%Check if singularity
        w(i)=w(i); else
        w(i)=w(i)/filter(i);
    end
end
%Compute phaselet and normalise.
theta=real(ifft(fftshift(w)))';
theta=theta./max(theta);
%Cancatenate phaselets.
Theta=theta;%initilise
for i=1:N
Theta=cat(1,theta,Theta);
end
%Compute output signal
for i=1:size(Theta)
   Sig(i)=cos(Theta(i)); end
%FINISH
```

7. ACKNOWLEDGMENTS

Jonathan Blackledge and Derry Fitzgerald are supported by the Science Foundation Ireland Stokes Professorship and Stokes Lectureship Programmes, respectively. Ruairi Hickson is funded by the Telecommunications Graduate Initiative, Ireland. The authors are grateful to Prof Eugene Coyle, Dr Marek Rebow and Prof Max Ammann, all of Dublin Institute of Technology for the continued encouragement and support.

8. REFERENCES

[1] J. Chowning, J. M. Grey, L. Rush and J. A. Moorer, *Computer Simulation of Music Instrument Tones in Reverberant Environments*, Report, Stanford University Department of Music, STAN-M-1, 1974 `http://ccrma.stanford.edu/files/papers/stanm1.pdf`.

[2] V. Välimäki, J. Pakarinen, C. Erkut and M. Karjalainen, *Discrete-time modelling of musical instruments*, Rep. Prog. Phys. 69, Institute of Physics Publishing, 1-78, 2006, `http://www.ece.uvic.ca/~bctill/papers/numacoust/V%C3%A4lim%C3%A4ki_etal_2006.pdf`.

[3] Wienna Symphonic Library, 2013, `http://vsl.co.at/`.

[4] J. Charles, *Playing Technique and Violin Timbre: Detecting Bad Playing*, PhD Thesis, Dublin Institute of Technology, 2010.

[5] J. Heifetz, *The Official Website*, 2013, `http://www.jaschaheifetz.com/`.

[6] N Milstein, *Reflection*, 2013 `http://www.andromeda.at/mus/mil/refl_e.html`.

[7] E. R. Miranda and J. Matthias, *Music Neurotechnology for Sound Synthesis: Sound Synthesis with Spiking Neuronal Networks*, Leonardo, MIT Press Journals, Vol. 42, No. 5, 439-442, 2009.

[8] Eureqa, *A software tool for detecting equations and hidden mathematical relationships in your data*, Cornell Creative Machine Lab, USA, 2013, `http://creativemachines.cornell.edu/eureqa`.

[9] M. J. Turner, J. M. Blackledge and P. R. Andrews, *Fractal Geometry in Digital Imaging*, Academic Press, 1998; ISBN-10: 0127039708; ISBN-13: 978-0127039701.

[10] P. Polotti, *Fractal Additive Synthesis via Harmonic-Band Wavelets*, Journal of Computer Music, Vol. 25, No. 3, MIT Press, Cambridge, 22-37, 2001.

[11] P. M. Morse and K. U. Ingard, *Theoretical Acoustics*, Prinston University Press, 1968.

[12] J. M. Blackledge, *Digital Signal Processing*, Horwood, 2006; ISBN: 1-904275-26-5, `http://eleceng.dit.ie/papers/102.pdf`.

[13] J. M. Blackledge, *Digital Image Processing*, Second Edition, Horwood, 2005; ISBN:1-898563-49-7, `http://eleceng.dit.ie/papers/103.pdf`.

[14] J. M. Blackledge, *The Fractal Market Hypothesis: Applications to Financial Forecasting*, Canter for Advanced Studies, Warsaw University of Technology, 2010; ISBN-13: 978-83-61993-02-5, `http://eleceng.dit.ie/papers/182.pdf`.

[15] P. Sundqvist, *Orthogonal Linear Regression*, MATLAB Central, 2005, `http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=6716&objectType=File`.

[16] E. Bibbona, G. Panfilo and P. Tavella, *The Ornstein-Uhlenbeck Process as a Model of a Lowpass Filtered White Noise*, Metrologia, Vol. 45 117-126, 2008.

[17] R. Hilfer, *Exact Solutions for a Class of Fractal Time Random Walks*, Fractals, Vol. 3(1), 211-216, 1995.

[18] R. Hilfer, *Foundations of Fractional Dynamics*, Fractals Vol. 3(3), 549-556, 1995.

[19] A. Compte, *Stochastic Foundations of Fractional Dynamics*, Phys. Rev E, Vol. 53(4), 4191-4193, 1996.