A Unified Time-Frequency Method for Synthesizing Noisy Sounds with Short Transients and Narrow Spectral Components

Damián Marelli[1], Mitsuko Aramaki[2], Richard Kronland-Martinet[3] and Charles Verron[4]
[1]School of Electrical Engineering and Computer Science, University of Newcastle, Australia.
[2]CNRS-Institut de Neurosciences Cognitives de la Méditerranée, France.
[3]CNRS-Laboratoire de Mécanique et d'Acoustique, France.
[4]Orange Labs, Department of Sound and Speech Technologies and Processing, France.

***Abstract*—The inverse FFT method was proposed to alleviate the complexity of the additive sound synthesis method in real time applications, and consists in synthesizing overlapping blocks of samples in the frequency domain. However, its application is limited by its inherent trade-off between time and frequency resolution. In this paper we propose an alternative method for overcoming this limitation. The proposed method generates time-frequency noise with an auto-correlation function such that the sound obtained after converting it to time domain has the desired time-varying power spectral density. We present synthesis examples illustrating the simultaneously good time and frequency resolution of the proposed method and study its complexity.**

***Index Terms*—IFFT sound synthesis, Colored noisy signal, short transient signal.**

I. Introduction

Various methods are available to generate both realistic and artificial sounds [1]. Methods based on linear models such as additive and subtractive synthesis play an important role since their synthesis parameters can be easily obtained through an analysis stage [2]. An efficient algorithm for additive synthesis is called Inverse Fast Fourier Transform (IFFT) [3], and consists in approximating the time domain signal in the time-frequency domain. By providing a clever choice of the analysis window, this method permitted a computational cost gain of approximately ten times over a time domain implementation [4].

In the synthesis of audio signals such as musical or environmental sounds, the stochastic component is modeled as a random process with time-varying statistics [5]. Using the IFFT method, this component is synthesized by generating time-frequency noise whose envelope equals the instantaneous power spectral density (PSD) of the target signal [5], [6], [4]. These methods require that the synthesis window length matches that of the noise auto-correlation function. Thus, the synthesis of narrowband noises requires a long synthesis window, which is incompatible with the generation of short transient signals. This is a severe issue that limits the use of time-frequency approaches for a wide class of audio signals such as impacts.

In this paper we propose a time-frequency synthesis method which achieves frequency and time resolutions, beyond the inherent trade-off of the IFFT method, while preserving a low computational cost. We present simulation results illustrating the performance using practical sound examples.

Financial support for this work was partially provided by the project "senSons" from the French National Research Agency (ANR).

II. Noisy Sound Synthesis using the IFFT Method

A noisy sound $y(t)$ is generally modeled as a stochastic process with a time-varying spectrum

$$\phi_y(z,t) = \mathcal{Z}_1\left\{r_y(\tau,t)\right\},$$

where $\mathcal{Z}_1\{\cdot\}$ denotes the z-transform with respect to the first variable, and

$$r_y(\tau,t) = \mathcal{E}\left\{y(t)y(t-\tau)\right\},$$

with $\mathcal{E}\{\cdot\}$ denoting expected value [5]. Using the IFFT method, the sound $y(t)$ is synthesized by the following overlap-add procedure:

$$y(t) = \sum_{k=-\infty}^{\infty} f(t-kD)v^{(k)}(t-kD) \tag{1}$$

where $f(t)$, $t \in \mathbb{Z}$ is a synthesis window of tap size M (i.e., $f(t) = 0$ if $t < 0$ or $t \geq M$), which is assumed to satisfy

$$\sum_{\tau=-\infty}^{\infty} f^2(t-\tau D) = 1 \text{ for all } t \in \mathbb{Z}, \tag{2}$$

to conserve energy, and $D \leq M$ is the synthesis hop size. The k-th block of M samples $v^{(k)}(t)$, $t = 0, \cdots, M-1$ is obtained doing the inverse discrete Fourier transform (DFT) of an M-dimensional random vector $\mathbf{v}(k)$, whose m-th component $\mathbf{v}_m(k)$ is given by

$$\mathbf{v}_m(k) = \phi_y(e^{j2\pi\frac{m-1}{M}}, kD)\mathbf{w}_m(k) \tag{3}$$

where $\mathbf{w}_m(k)$ is the m-th entry of a white complex random vector $\mathbf{w}(k)$ with Gaussian distribution.

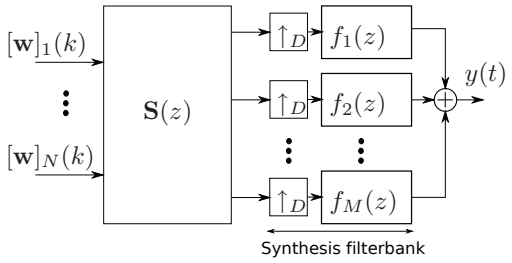

Figure 1. Scheme for achieving an arbitrary spectral shape.

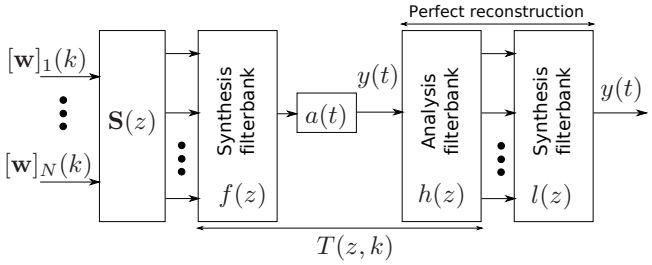

Figure 2. Scheme for achieving arbitrary time resolution.

Using the result in [7, eq. 10], it is straightforward to verify that (1) is equivalent to a synthesis filterbank operation, i.e.,

$$y(z) = \sum_{m=1}^{M} f_m(z) \uparrow_D \{\mathbf{v}_m\}(z) \tag{4}$$

where the filters $f_m(z) = f(e^{j2\pi\frac{m-1}{M}}z)$, $m = 1, \cdots, M$ are frequency-shifted versions of the synthesis window $f(z)$, and $\uparrow_D\{\mathbf{v}_m\}(k)$ denotes the upsampling operation with factor D (i.e., inserting $D-1$ zeros between every two samples) applied to the signal $\mathbf{v}_m(k)$.

It follows from (4) that the frequency resolution of the IFFT method is determined by the spectral shape of the synthesis window $f(t)$, and that the time-resolution is given by its time domain concentration. Hence, this method suffers from an inherent tradeoff between time and frequency resolution, turning the synthesis of narrowband noises incompatible with the generation of short transient signals.

III. Proposed Synthesis Method

In this section we propose an alternative approach which overcomes the aforementioned tradeoff. In Sections III-A and III-B we explain how to achieve arbitrary frequency and time resolutions, respectively, and in Section III-C we use these results to describe the proposed method.

A. Achieving Arbitrary Frequency Resolution

In this section we assume that we want to synthesize a stationary random process with an arbitrary spectral shape. As depicted in Figure 1, the idea consists of processing an N-dimensional white random vector $\mathbf{w}(k)$ (the value of N is to be determined by the the number of columns of the transfer matrix $\mathbf{R}(z)$ in (6) below) by an $M \times N$ transfer matrix $\mathbf{S}(z)$ so that the signal obtained after synthesis has the desired spectrum.

Using polyphase representation [8], we can write

$$\mathbf{y}(z) = \mathbf{F}(z)\mathbf{S}(z)\mathbf{w}(z)$$

where $\mathbf{y}(z)$ and $\mathbf{F}(z)$ are the polyphase representations of $y(t)$ and the synthesis filterbank, respectively, having impulse responses

$$\begin{aligned} [\mathbf{y}(k)]_d &= y(kD+1-d), \\ [\mathbf{F}(k)]_{d,m} &= f_m(kD-d+1), \end{aligned}$$

for all $m = 1, \cdots, M$, and $d = 1, \cdots, D$. Let $\phi_{\mathbf{y}}(z) = \mathcal{E}\{\mathbf{y}(z)\mathbf{y}^*(z)\}$ be the spectrum of $\mathbf{y}(z)$, where the superscript * denotes transpose conjugate. We have that

$$\begin{aligned} \phi_{\mathbf{y}}(z) &= \mathbf{F}(z)\mathbf{S}(z)\mathcal{E}\{\mathbf{w}(z)\mathbf{w}^*(z)\}\mathbf{S}^*(z)\mathbf{F}^*(z) \\ &= \mathbf{F}(z)\mathbf{S}(z)\mathbf{S}^*(z)\mathbf{F}^*(z). \end{aligned} \tag{5}$$

Let $\phi_{\mathbf{y}}(z) = \mathbf{R}(z)\mathbf{R}^*(z)$ be a spectral factorization of $\phi_{\mathbf{y}}(z)$ [9]. Then, from (5), the required matrix $\mathbf{S}(z)$ needs to satisfy

$$\mathbf{R}(z) = \mathbf{F}(z)\mathbf{S}(z) \tag{6}$$

The minimum-norm solution of (6) is given by:

$$\mathbf{S}(z) = \tilde{\mathbf{F}}^*(z)\mathbf{R}(z)$$

where $\tilde{\mathbf{F}}(z)$ is the polyphase representation of the dual window $\tilde{f}(z)$ [10] of $f(z)$. To improve efficiency, the number of non-zero entries in $\mathbf{S}(z)$ can be reduced by solving (6) using sparse approximation techniques [11].

B. Achieving Arbitrary Time Resolution

Now suppose that we want to change the amplitude of the spectrum of the random process synthesized in Section III-A, following an arbitrary law $a(t)$. We can do this in the time domain by multiplying by $a(t)$ the output of the synthesis filterbank. To transpose this operation to the time-frequency domain we add after $a(t)$ a perfect-reconstructing pair of analysis and synthesis filterbanks (i.e., a pair which achieves perfect reconstruction), as shown in Figure 2.

The analysis filterbank operation consists of filtering $y(t)$ using an array of filters $h_m(z)$, $m = 1, \cdots, M$ followed by a downsampling operation of factor D (by keeping one out of D samples). Then, the impulse response $[\mathbf{T}]_{m,n}(l,k)$ of the m,n-th entry of the $M \times M$ transfer matrix $\mathbf{T}(z,k)$ shown in Figure 2 is

$$[\mathbf{T}]_{m,n}(l,k) = (a_k h_m * f_n)(lD) \tag{7}$$

where $a_k(t) = a(kD-t)$. Let C be the impulse response length of $h_m(z)$, $m = 1, \cdots, M$. For each k, we can write $a_k(t)$, $t = 0, \cdots, C-1$ using DFT as follows

$$a_k(t) = \sum_{c=1}^{C} \alpha^{(c)}(k) e^{j\frac{2\pi(c-1)}{C}t}. \tag{8}$$

Then, from (7) and (8) we have

$$\mathbf{T}(l,k) = \sum_{c=1}^{C} \alpha^{(c)}(k)\mathbf{T}^{(c)}(l), \tag{9}$$

where the impulse response $[\mathbf{T}^{(c)}]_{m,n}(l)$ of the m,n-th entry of $\mathbf{T}^{(c)}(z)$ is given by

$$[\mathbf{T}^{(c)}]_{m,n}(l) = (h_m^{(c)} * f_n)(lD), \tag{10}$$

$$h_m^{(c)}(t) = h_m(t)e^{j\frac{2\pi(c-1)}{C}t}. \tag{11}$$

Remark 1: Since the filters $h_m(z)$, $m = 1, \cdots, M$ are frequency-shifted versions of the same prototype $h(z)$, if $R = C/M$ is integer, then the output of $\mathbf{T}_{m,n}^{(c+rR)}(l)$ is obtained from that of $\mathbf{T}_{m,n}^{(c)}(l)$ by doing an r-step circular shift. Hence, only the R transfer matrices $\mathbf{T}_{m,n}^{(c)}(l)$, $c = 0, \cdots, R-1$ need to be computed to generate (9).

C. Proposed Synthesis Scheme

Using the results above we can propose a time-frequency method for synthesizing noisy sounds with the desired frequency and time resolutions. The idea is depicted in Figure 3. The number P of frequency bands is chosen to achieve the desired frequency resolution. Notice that P does not need to be equal to the number M of frequency bands used in the time-frequency representation. Then, for each $p = 1, \cdots, P$, a white, N-dimensional vector random process $\mathbf{w}_p(k)$ is applied to the input of an $M \times N$ transfer matrix $\mathbf{S}_p(z)$. The matrix $\mathbf{S}_p(z)$ is designed as described in Section III-A, so that it generates at the output a narrow frequency band with the desired spectral shape. Then, for each $k \in \mathbb{Z}$, the coefficients $\alpha_p^{(c)}(k)$, $c = 1, \cdots, C$ are computed from the desired time-varying amplitude of the p-th frequency band using (8). The constants $\alpha_p^{(c)}(k)$ are then used to multiply the output of the $M \times M$ transfer matrices $\mathbf{T}^{(c)}(z)$, $c = 1, \cdots, C$, which are computed as explained in Section III-B.

Remark 2: The scheme in Figure 3 requires the computation of P transfer matrices of dimension $M \times N$ and $P \times R$ transfer matrices of dimension $M \times M$. Hence, it seems to suffer from a very high complexity. However, notice that the non-zero entries of the matrices $\mathbf{S}_p(z)$ concentrate in a neighborhood of the row corresponding to the center of the p-th frequency band. Also, in view of (10), the non-zero entries of the matrices $\mathbf{T}^{(c)}(z)$ concentrate towards its main diagonal. Hence, the composition $\mathbf{T}^{(c)}(z)\mathbf{S}_p(z)$ can be very efficiently implemented.

IV. A Practical Design

In this section we use the method proposed in Section III-C to design a time-frequency sound synthesizer with prescribed time and frequency resolutions. We use a sampling frequency of $f_s = 44.1$kHz. We want a frequency resolution of $P = 1024$ frequencies over the range $[-f_s/2, f_s/2]$, and we want the amplitude of the signal in each frequency band to vary once every 64 samples $= 1.4\,ms$.

To simplify the design, we choose $M = P = 1024$ frequency bands, and we let the spectral shape of the m-th frequency band equal that of the synthesis filter $f_m(z)$. With these choices, we have that $N = 1$, and the entries of the $M \times 1$ spectral shaping transfer matrix $\mathbf{S}_m(z)$ are all zero except for the $m,1$-th entry which is $[\mathbf{S}_m]_{m,1}(z) = 1$. We choose the synthesis hop size $D = 3/4M = 768$ which minimizes the overall complexity. We design the prototype filter $f(z)$ using a root raised cosine filter with roll-off factor $\beta = M/D - 1 = 1/3$ and bandwidth $BW = \pi(1+\beta)/M = 4\pi/3M$ [12], that guarantees

$$\sum_{m=1}^{M} |f_m(e^{j\omega})|^2 = 1 \text{ for all } \omega \in [-\pi, \pi],$$

hence a flat spectrum is obtained when all bands have the same amplitude. The frequency response of the resulting synthesis filters is shown in Figure 4.

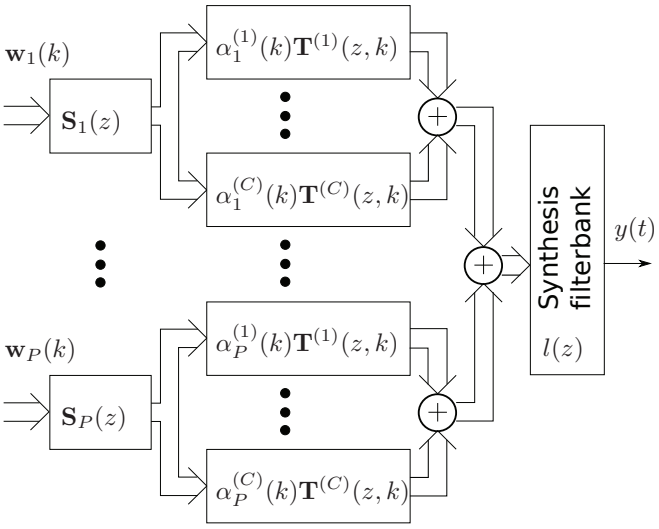

Figure 3. Proposed time-frequency synthesis method.

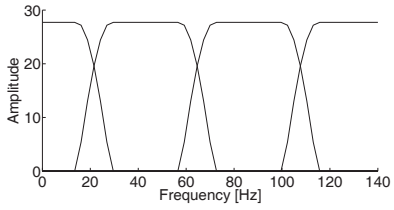

Figure 4. Frequency response of the synthesis filters $f_m(z)$, $m = 1, \cdots, M$.

Remark 3: The choice of a root raised cosine filter to design $f(z)$ results in the synthesis filter $f_m(z)$ having infinite impulse response. However, notice that these filters are only used to build the transfer matrices $\mathbf{T}^{(c)}$, $c = 1, \cdots, C$ in (10), and their influence in the overall complexity is only implicit in the computation of these matrices.

It follows from equation (10) that the filters $h_m(z)$, $m = 1, \cdots, M$ need to be concentrated in frequency, so that the off-diagonal terms of $\mathbf{T}^{(c)}(z)$, $c = 1, \cdots, C$ vanish quickly. However, if their frequency response is too concentrated their

impulse response length C becomes too big, and therefore, a large number R of transfer matrices $\mathbf{T}^{(c)}(z)$ need to be computed (recall Remark 1). We found a good compromise by choosing $R = 3$ and designing the prototype $h(z)$ as the FIR filter having impulse response length $1024 \times R = 3072$, which best approximates in a least-squares a raised cosine window with roll-off factor $\beta = 1$ and $BW = 2\pi/M$. The prototype $l(z)$ of the synthesis filters also has impulse response length 3072 and is computed using the method described in [13], achieving an analysis/synthesis reconstruction error of -80dB.

As mentioned before, the amplitude of each frequency band is specified once every 64 samples. In order to build the coefficients $\alpha_p^{(c)}(k)$ using (8), we need to interpolate these values to obtain one value per sample. We do so using a raised cosine window with $\beta = 1$, which is concentrated in frequency and hence minimizes the number of terms required in the expansion (9). The interpolated amplitude function has bandwidth 690Hz, and therefore only $\bar{C} = 1+2MR/64 = 97$ terms need to be computed in the expansion (9).

In order to reduce the implementation complexity, we truncate the transfer matrices $\mathbf{T}^{(r)}(z)$, $r = 1, \cdots, R$ by zero-rounding their entries having absolute values smaller than a threshold chosen 40dB smaller than the maximum absolute value. After doing so, the computation of $\mathbf{T}^{(r)}(z)$, $r = 1, \cdots, R$ requires 95 (real) multiplications per sample[1]. The computation of $\bar{C}$ terms of (9) requires 192 multiplications per sample. Finally, the synthesis filterbank with filters $l_m(z)$, $m = 1, \cdots, M$ is computed using the method [7], which requires 31 multiplications per sample. Hence, the overall complexity is 318 multiplications per sample.

In order to illustrate the proposed method we synthesize a blurry sound effect on a glass impact whose model consists of narrow frequency bands at 1051 Hz, 1849 Hz, 3388 Hz, 5339 Hz, 7606 Hz and 10163 Hz, whose amplitude decay exponentially with time constants 4.948, 6.397, 10.78, 21.26, 47.49 and 110.4, respectively. For comparison purposes we consider the *time domain* method obtained by multiplying the output of the filters $f_m(z)$ in (4) by the desired amplitude values before addition. In this method, Remark 3 does not apply, hence, we design $f_m(z)$ to have impulse response length of 16384 samples. Then, since only 512 frequency bands need to be computed, and each $f_m(z)$ needs to be computed only once every $D = 768$ samples, the overall complexity (including the amplitude multiplication at the output of each $f_m(z)$) is of 11435 multiplications per sample, i.e., about 36 times more complex than the proposed time-frequency method. The synthesized signals and their spectra are shown in Figures 5 and 6, respectively, and their relative square error difference is -27.12dB. For a perceptual evaluation, this glass impact sound example, as well as other examples including waves, wind, whoosh, stones, etc., can be found at http://www.lma.cnrs-mrs.fr/~kronland/ICASSP2010/sounds.html.

[1]Notice that, since the synthesized sound is real-valued, only half of its spectrum needs to be synthesized.

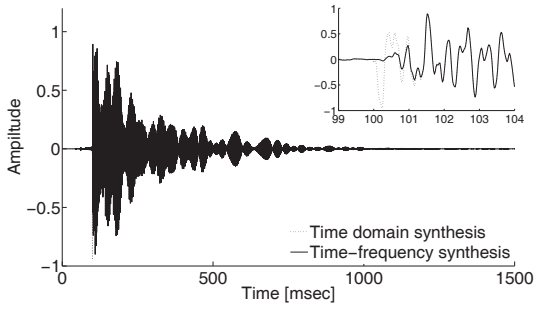

Figure 5. Synthesis of a glass impact starting from 100 msec., and detail of the attack showing a settling time of about 1.5 msec. for the time-frequency method.

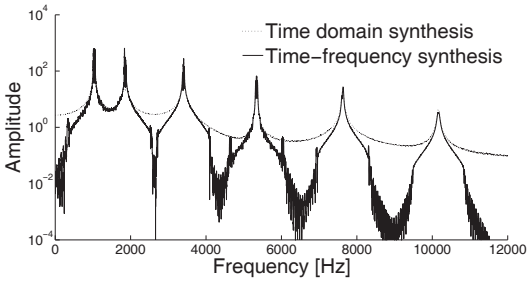

Figure 6. Spectra of the synthesized glass impact sounds.

REFERENCES

[1] C. Roads, *The Computer Music Tutorial*, Fifth, Ed. MIT Press, 2000.

[2] R. Kronland-Martinet, P. Guillemain, and S. Ystad, "Modelling of natural sounds by time-frequency and wavelet representations," *Organised Sound*, vol. 2, no. 3, pp. 179–191, 1997.

[3] X. Rodet and P. Depalle, "Spectral envelopes and inverse fft synthesis," in *Proc. of the 93rd AES Conv.*, 1992.

[4] X. Rodet and D. Schwarz, *Analysis, Synthesis, and Perception of Musical Sounds: Sound of Music*. Springer, 2007, ch. Spectral Envelopes and Additive + Residual Analysis/Synthesis.

[5] X. Serra and J. O. Smith, "Spectral modeling synthesis: A sound analysis/synthesis system based on a deterministic plus stochastic decomposition," *Comp. Music. J.*, vol. 14, no. 4, 1990.

[6] X. Amatriain, J. Bonada, A. Loscos, and X. Serra, *DAFX: Digital Audio Effects*. John Wiley & Sons Publishers, 2002, ch. Spectral Processing.

[7] S. Weiss and R. Stewart, "Fast implementation of oversampled modulated filter banks," *Electronics Letters*, vol. 36, no. 17, pp. 1502–1503, August 2000.

[8] P. Vaidyanathan, *Multirate Systems and Filterbanks*. Englewood Cliffs, N.J.: Prentice Hall, 1993.

[9] A. Sayed and T. Kailath, "A survey of spectral factorization methods," *Numerical Linear Algebra with Applications*, vol. 8, no. 6-7, pp. 467–496, 2001.

[10] K. Gröchenig, *Foundations of time-frequency analysis*, ser. Applied and Numerical Harmonic Analysis. Boston, MA: Birkhäuser Boston Inc., 2001.

[11] J. Tropp, "Greed is good: algorithmic results for sparse approximation," *IEEE Tr. on Information Theory*, vol. 50, no. 10, pp. 2231–2242, 2004.

[12] J. Proakis, *Digital Communications*, 4th ed. McGraw-Hill, 8 2000.

[13] J. Morris and Y. Lu, "Generalized Gabor expansions of discrete-time signals in $l^2(Z)$ via biorthogonal-like sequences," *IEEE Transactions on Signal Processing*, vol. 44, no. 6, pp. 1378–1391, 1996.